Add a permission의

Amazon Lightsail

Copyright © 2026 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Amazon Lightsail: Add a permission의

Copyright © 2026 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Amazon의 상표 및 트레이드 드레스는 Amazon 외 제품 또는 서비스와 함께, Amazon 브랜드 이미지를 떨어뜨리거나 고객에게 혼동을 일으킬 수 있는 방식으로 사용할 수 없습니다. Amazon이 소유하지 않은 기타 모든 상표는 Amazon과 제휴 관계이거나 관련이 있거나 후원 관계와 관계없이 해당 소유자의 자산입니다.

Table of Contents

Lightsail란 무엇인가요? 1
특성 1
Lightsail은 누구를 대상으로 합니까? 3
시작하기 3
관련 서비스 3
추정, 결제 및 비용 최적화 4
설정 5
에 가입 AWS 계정 5
관리자 액세스 권한이 있는 사용자 생성 5
Lightsail 액세스 7
Lightsail 서비스 인터페이스 7
Lightsail 서비스 엔드포인트 8
IPv4 엔드포인트 8
이중 스택(IPv4 및 IPv6) 엔드포인트 9
리전별 서비스 엔드포인트 9
엔드포인트 지정의 예 11
시작하기 14
1단계: 필수 구성 요소 완성 14
2단계: 인스턴스 만들기 14
3단계: 인스턴스에 연결 15
4단계: 인스턴스에 스토리지 추가 18
5단계: 스냅샷 생성 19
6단계: 정리 19
다음 단계 20
와 Lightsail 함께 사용 AWS CLI 20
사전 조건 21
SSH 키 페어 생성 21
인스턴스 생성 및 관리 22
인스턴스에 연결합니다 26
인스턴스에 스토리지 추가 28
스냅샷 생성 및 사용 30
리소스 정리 32
다음 단계 35
Lightsail 리셀러 36

Lightsail 재판매의 이점 36
Lightsail 리셀러 혜택 및 증가된 기본 할당량이 계정에 적용되는 방식 37
Lightsail 리셀러가 되는 방법 39
Lightsail 리셀러 되기 39
Lightsail 리셀러가 되는 데 필요한 정보 40
Lightsail 리셀러 자격 신청 40
추가 계정의 Lightsail 리셀러 신청 43
서비스 할당량 증가 44
리셀러로서 Lightsail에 문의하기 46
인스턴스 49
인스턴스 생성 49
Linux 인스턴스 50
Windows 인스턴스 54
블루프린트 61
운영 체제 61
데이터베이스 애플리케이션 65
CMS 애플리케이션 66
애플리케이션 스택 및 서버 69
전자 상거래 애플리케이션 73
프로젝트 관리 애플리케이션 73
번들 74
Linux/Unix 번들(퍼블릭 IPv4 주소 포함) 74
Linux/Unix 번들(IPv6 전용) 76
Windows 번들(퍼블릭 IPv4 주소 포함) 78
Windows 번들(IPv6 전용) 80
인스턴스 방화벽 82
Lightsail 방화벽 83
방화벽 규칙 생성 84
프로토콜 지정 84
포트 지정 85
애플리케이션 계층 프로토콜 유형 지정 86
소스 IP 주소 지정 88
기본 Lightsail 방화벽 규칙 88
방화벽 규칙 추가 90
방화벽 규칙 삭제 92
인스턴스 방화벽 규칙 93

버스트 용량 및 성능 96
CPU 성능 97
버스트 용량 누적 99
인스턴스 버스트 식별 101
버스트 용량 모니터링 102
버스트 용량 보기 104
높은 CPU 문제 해결 107
인스턴스 관리 107
인스턴스 시작, 중지 또는 다시 시작 108
인스턴스 강제 중지 110
향상된 네트워킹 112
Lightsail에서 Windows Serve 파일 시스템 확장 114
Linux shell 스크립트 117
PowerShell 스크립트 119
Windows 보안 모범 사례 121
인스턴스 삭제 125
Lightsail 콘솔 홈 페이지에서 인스턴스 삭제 125
Lightsail 콘솔 인스턴스 관리 페이지에서 인스턴스 삭제 126
를 사용하여 인스턴스 삭제 AWS CLI 127
다음 단계 129
SSH와 인스턴스에 대한 연결 130
키 페어 옵션 선택 131
인스턴스에 연결 131
인스턴스에 저장된 키 관리 132
SSH 키 설정 133
SSH 키 관리 136
인스턴스 SSH 키 관리 150
Linux 인스턴스에 연결 155
Windows 인스턴스에 연결 174
인스턴스 메타데이터 서비스 190
인스턴스 메타데이터 서비스 사용 190
추가 IMDS 문서 191
IMDS 구성 191
디스크 198
블록 스토리지 디스크 198
디스크 할당량 199

Linux 인스턴스에 디스크 연결 199
1단계: 새 디스크 생성 및 인스턴스에 연결 199
2단계: 인스턴스에 연결하여 디스크 포맷 및 탑재 201
3단계: 인스턴스를 재부팅할 때마다 디스크 탑재 206
디스크를 Windows 인스턴스에 연결 206
1단계: 새 블록 스토리지 디스크 생성 및 인스턴스에 연결 207
2단계: 인스턴스에 연결하고 블록 스토리지 디스크를 온라인 상태로 만들기 209
3단계: 블록 스토리지 디스크 초기화 211
4단계: 파일 시스템으로 디스크 포맷 213
디스크 분리 및 삭제 215
사전 조건 215
디스크 분리 및 삭제 215
스냅샷 217
수동 스냅샷 수 217
자동 스냅샷 218
시스템 디스크 스냅샷 218
스냅샷에서 새 리소스 생성 218
스냅샷 복사 219
Amazon EC2로 스냅샷 내보내기 219
스냅샷 삭제 219
자동 스냅샷 219
자동 스냅샷 제한 220
자동 스냅샷 보존 220
Lightsail 콘솔을 사용하여 자동 인스턴스 스냅샷 활성화 또는 비활성화 221
를 사용하여 인스턴스 또는 블록 스토리지 디스크에 대한 자동 스냅샷 활성화 또는 비활성화
AWS CLI 222
스냅샷 시간 변경 226
자동 스냅샷 삭제 230
자동 스냅샷 유지 234
Linux 스냅샷 240
Windows 스냅샷 및 sysprep 241
1단계: Sysprep을 실행하기 전에 백업 스냅샷 생성 241
2단계: Sysprep을 사용하여 인스턴스에 연결 및 종료 243
3단계: Sysprep 실행 후 스냅샷 생성 245
다음 단계 246
블록 스토리지 디스크 스냅샷 생성 247

스냅샷에서 디스크를 생성합니다. 248
1단계: 디스크 스냅샷을 찾아서 새 디스크 만들기 248
2단계: 디스크 스냅샷에서 새 디스크 만들기 250
루트 볼륨 스냅샷 생성 251
1단계: 필수 구성 요소 완성 252
2단계: 인스턴스 루트 볼륨 스냅샷 생성 252
3단계: 스냅샷에서 블록 스토리지 디스크를 생성하고 이를 인스턴스에 연결 254
4단계: 인스턴스에서 블록 스토리지 디스크 액세스 256
스냅샷에서 인스턴스 생성 261
스냅샷으로 더 큰 리소스 생성 263
사전 조건 263
리소스 생성 263
를 사용하여 스냅샷에서 더 큰 리소스 생성 AWS CLI 265
사전 조건 265
1단계: 스냅샷 이름 가져오기 265
2단계: 번들 선택 265
3단계: AWS CLI 명령 작성 및 새 인스턴스 생성 269
다음 단계 270
스냅샷 삭제 270
리전 간 스냅샷 복사 272
사전 조건 272
스냅샷 복사 272
다음 단계 274
EC2로 스냅샷 내보내기 275
내보낸 Lightsail 스냅샷에서 Amazon EC2 리소스 생성 276
Amazon EC2 인스턴스 유형 선택 278
Amazon EC2 인스턴스에 연결 278
Amazon EC2 인스턴스 보호 279
스냅샷을 내보내는 방법 279
내보내기 모니터링 283
내보낸 스냅샷에서 EC2 인스턴스 생성 284
내보낸 스냅샷에서 EBS 볼륨 생성 292
Linux EC2 인스턴스에 연결 294
Linux/Unix EC2 인스턴스 보호 302
Windows EC2 인스턴스에 연결 310
Windows EC2 인스턴스 보호 316

CloudFormation 스택 ... 318
도메인 및 DNS ... 320
도메인 등록 방식 ... 320
에 등록할 수 있는 도메인Lightsail ... 321
도메인 등록 요금 ... 321
도메인에 대한 추가 정보 ... 322
Lightsail의 DNS ... 322
DNS 용어 ... 323
Lightsail DNS 영역에서 지원되는 DNS 레코드 유형 ... 324
DNS 영역 생성 ... 326
DNS 영역 편집 ... 333
DNS 영역 삭제 ... 334
인터넷 트래픽 라우팅 ... 334
인스턴스로 도메인 연결 ... 337
로드 밸런서로 도메인 연결 ... 339
전송 DNS 관리 ... 343
Route 53 사용 ... 344
도메인 등록 ... 347
Lightsail을 사용하여 새 도메인 등록 ... 348
도메인 세부 정보 ... 351
도메인 이름 형식 지정 ... 352
도메인 이름 등록에 대한 도메인 이름 형식 지정 ... 352
DNS 영역 및 레코드에 대한 도메인 이름 형식 지정 ... 353
DNS 영역 및 레코드의 이름에 별표(*) 사용 ... 353
다음 단계 ... 354
R53에서 도메인 관리 ... 354
도메인 등록 상태 보기 ... 355
다른 등록 대행자로의 무단 이전을 방지하기 위해 도메인 잠그기 ... 355
만료되거나 삭제된 도메인 복원 ... 355
도메인 등록 이전 ... 356
도메인 이름 등록 삭제 ... 356
등록 정보 ... 356
Term ... 357
자동 도메인 갱신 ... 357
등록자, 관리자, 기술 담당자 및 청구 담당자 연락처 ... 357
연락처 유형 ... 358

이름, 성 ... 358
조직 ... 358
이메일 ... 359
전화번호 ... 359
주소 1 ... 359
주소 2 ... 359
국가 ... 359
시/도 ... 360
구/군/시 ... 360
우편번호 ... 360
개인 정보 보호 ... 360
등록 갱신 ... 361
자동 갱신 ... 361
도메인 등록 도중 도메인에 대한 자동 갱신 구성 ... 362
이미 등록된 도메인에 대한 자동 갱신 구성 ... 363
개인 정보 보호 ... 363
사전 조건 완료 ... 364
도메인의 개인 정보 보호 관리 ... 364
도메인 연락처 정보 업데이트 ... 364
도메인 소유자는 누구입니까? ... 364
도메인의 연락처 정보 업데이트 ... 365
데이터베이스 수 ... 366
데이터베이스 생성 ... 366
다음 단계 ... 370
데이터베이스 엔진 선택 ... 370
Lightsail에서 관리형 데이터베이스 비교 ... 370
데이터 가져오기 최적화 ... 372
고가용성 데이터베이스 ... 373
MySQL에 연결 ... 373
1단계: MySQL 데이터베이스 연결 세부 정보 가져오기 ... 373
2단계: MySQL 데이터베이스의 퍼블릭 가용성 구성 ... 374
3단계: 데이터베이스 클라이언트를 구성하여 MySQL 데이터베이스에 연결 ... 375
다음 단계 ... 377
SSL을 사용하여 MySQL에 연결 ... 377
지원되는 연결 ... 378
사전 조건 ... 378

SSL을 사용하여 MySQL 데이터베이스에 연결 ... 379
PostgreSQL에 연결 ... 380
1단계: PostgreSQL 데이터베이스 연결 세부 정보 가져오기 ... 381
2단계: PostgreSQL 데이터베이스의 퍼블릭 가용성 구성 ... 382
3단계: 데이터베이스 클라이언트를 구성하여 PostgreSQL 데이터베이스에 연결 ... 382
다음 단계 ... 385
SSL을 사용하여 PostgreSQL에 연결 ... 385
사전 조건 ... 386
SSL을 사용하여 Postgres 데이터베이스에 연결 ... 386
데이터베이스 삭제 ... 387
데이터 가져오기 모드 ... 388
SQL 데이터 가져오기 ... 389
데이터 PostgreSQL 가져오기 ... 391
데이터베이스 로그 ... 393
MySQL 쿼리 로그 ... 395
특정 시점 백업 비활성화 ... 398
사전 조건 ... 399
데이터베이스 특정 시점 백업 비활성화 ... 399
데이터베이스 스냅샷 ... 400
다음 단계 ... 402
데이터베이스 복원 ... 402
스냅샷에서 데이터베이스 생성 ... 404
SSL 인증서 다운로드 ... 407
모든 AWS 리전에 대한 인증서 번들 ... 408
특정 AWS 리전에 대한 인증서 번들 ... 408
CA 인증서 업데이트 ... 408
유지 관리 및 백업 기간 ... 412
사전 조건 ... 412
데이터베이스 유지 관리 기간 변경 ... 412
다음 단계 ... 415
데이터베이스 암호 관리 ... 416
다음 단계 ... 417
퍼블릭 모드 ... 417
다음 단계 ... 418
파라미터 업데이트 ... 419
사전 조건 ... 419

사용 가능한 데이터베이스 파라미터 목록 가져오기 ... 419
데이터베이스 파라미터 업데이트 ... 421
메이저 버전 업그레이드 ... 423
사전 조건 ... 423
데이터베이스 메이저 버전 업데이트 ... 424
다음 단계 ... 427
MySQL 5.6에서 마이그레이션 ... 427
1단계: 변경 사항 이해 ... 427
2단계: 사전 조건 완료 ... 428
3단계: MySQL 5.6 데이터베이스에 연결 및 데이터 내보내기 ... 428
4단계: MySQL 5.7 데이터베이스에 연결 및 데이터 가져오기 ... 432
5단계: 애플리케이션 테스트 및 마이그레이션 완료 ... 435
로드 밸런서 ... 436
로드 밸런서 기능 ... 436
로드 밸런서의 사용 시점 ... 437
로드 밸런싱에 권장되는 애플리케이션 ... 437
로드 밸런서 사용 시작하기 ... 437
로드 밸런서 생성 ... 438
사전 조건 ... 438
로드 밸런서 생성 ... 438
로드 밸런서에 인스턴스 연결 ... 439
다음 단계 ... 440
로드 밸런서 설정 업데이트 ... 440
상태 확인 ... 440
암호화된 트래픽(HTTPS) ... 441
세션 지속성 ... 441
인스턴스 로드 밸런싱 ... 442
일반 지침: 데이터베이스를 사용하는 애플리케이션 ... 442
WordPress ... 442
Node.js ... 442
Magento ... 443
GitLab ... 443
Drupal ... 444
LAMP 스택 ... 444
MEAN 스택 ... 445
Redmine ... 445

Nginx ... 445
Joomla! ... 445
TLS 보안 정책 구성 ... 446
보안 정책 개요 ... 446
지원되는 보안 정책 및 프로토콜 ... 446
사전 조건 완료 ... 448
Lightsail 콘솔을 사용하여 보안 정책 구성 ... 448
를 사용하여 보안 정책 구성 AWS CLI ... 448
HTTP-HTTPS 리디렉션 ... 449
사전 조건 완료 ... 450
Lightsail 콘솔을 사용하여 로드 밸런서에 HTTPS 리디렉션 구성 ... 450
를 사용하여 로드 밸런서에 대한 HTTP에서 HTTPS로의 리디렉션 구성 AWS CLI ... 451
세션 지속성 ... 452
세션 지속성 활성화 ... 452
쿠키 지속 기간 조정 ... 453
건전성 체크 ... 454
상태 확인 경로 사용자 지정 ... 454
상태 확인 지표 ... 455
건전성 체크 ... 457
인스턴스 분리 ... 458
로드 밸런서 삭제 ... 458
배포 ... 459
사용 사례 ... 461
배포 구성 ... 462
엣지 로케이션 및 IP 주소 범위 ... 463
배포 생성 ... 463
사전 조건 ... 464
오리진 리소스 ... 465
오리진 프로토콜 정책 ... 466
캐싱 동작 및 캐싱 사전 설정 ... 466
WordPress 캐싱 사전 설정 모범 사례 ... 467
기본 동작 ... 468
디렉터리 및 파일 재정의 ... 468
고급 캐시 설정 ... 470
배포 플랜 ... 473
배포 생성 ... 473

다음 단계 476
배포 삭제 476
배포 삭제 476
캐싱 동작 477
캐싱 사전 설정 477
WordPress 캐싱 사전 설정 모범 사례 478
기본 동작 479
디렉터리 및 파일 재정의 479
고급 캐시 설정 480
배포의 캐시 동작 변경 483
캐시 재설정 484
오리진 변경 484
오리진 프로토콜 정책 485
배포 오리진 변경 485
배포와 함께 버킷 사용 487
1단계: 필수 구성 요소 완성 488
2단계: 버킷 권한 수정 488
3단계: 버킷을 오리진으로 하는 배포 생성 491
4단계: 배포용 사용자 지정 하위 도메인 활성화 493
5단계: WordPress 웹 사이트에 WP Offload Media Lite 플러그 인 설치 493
6단계: WordPress 웹 사이트와 Lightsail 버킷 및 배포 간의 연결 테스트 500
버킷 및 객체 관리 503
플랜 변경 505
배포 플랜 변경하기 505
배포 사용자 지정 도메인 506
사전 조건 506
배포용 사용자 지정 도메인 활성화 506
배포로 도메인 연결 507
사용자 지정 도메인 변경 509
배포 사용자 지정 도메인 사용 중지 510
배포 도메인을 컨테이너 서비스에 추가 511
요청 및 응답 동작 513
배포에서 요청을 처리하고 오리진에 요청을 전달하는 방법 513
배포에서 오리진의 응답을 처리하는 방법 527
배포 테스트 531
배포 테스트 531

네트워킹 533
로드 밸런서 533
고정 IP 533
IP 주소 533
인스턴스의 프라이빗 및 퍼블릭 IPv4 주소 534
인스턴스의 고정 IPv4 주소 536
인스턴스, 컨테이너 서비스, CDN 배포 및 로드 밸런서의 IPv6 537
정적 IP 주소 540
듀얼 스택 네트워킹 545
IPv6 전용 네트워킹 549
리전 및 가용 영역 553
Lightsail 리전 554
SSH 키 및 Lightsail 리전 555
Lightsail 리전으로 작업하기 위한 팁 555
Lightsail 가용 영역 555
가용 영역과 Lightsail 애플리케이션 556
리전 활성화 556
리전 비활성화 558
VPC 피어링 560
다른 AWS 서비스와의 통신 허용 561
SSL/TLS 인증서 562
HTTPS를 사용하는 이유 562
프로세스 개요 563
배포 또는 컨테이너 서비스와 함께 SSL/TLS 인증서 사용 563
로드 밸런서와 함께 SSL/TLS 인증서 사용 564
컨테이너 인증서 565
배포 인증서 570
로드 밸런서 인증서 581
역방향 DNS 구성 590
사전 조건 590
AWS Support에 요청을 제출하여 역방향 DNS 구성 591
버킷 593
객체 스토리지 개념 593
버킷 및 객체 관리 595
버킷 생성 596
버킷 생성 596

버킷 및 객체 관리 597
버킷 삭제 598
버킷 강제 삭제 599
Lightsail 콘솔을 사용하여 버킷 삭제 599
를 사용하여 버킷 삭제 AWS CLI 600
버킷 및 객체 관리 601
액세스 키 생성 603
버킷용 액세스 키 생성 604
액세스 키 삭제 605
버킷에 대한 액세스 키 삭제 605
퍼블릭 액세스 차단 605
계정에 대한 퍼블릭 액세스 차단 설정 구성 606
버킷 및 객체 관리 609
버킷 액세스 로그 611
로그 전송을 활성화하려면 필요한 사항이 무엇입니까? 611
로그 객체 키 형식 612
로그 전송 방법 612
최선의 액세스 로그 전송 612
버킷 로깅 상태 변경 시 일정 기간에 걸쳐 단계적으로 반영됨 613
액세스 로그 형식 613
액세스 로그 관리 626
액세스 로그 사용 630
버킷 객체 635
Lightsail 콘솔을 사용한 객체 필터링 635
를 사용하여 객체 보기 AWS CLI 637
버킷 및 객체 관리 639
객체 복사 및 이동 641
객체 삭제 645
객체 다운로드 653
객체 필터링 657
객체 버전 관리 661
객체 버전 복원 666
객체에 태그 지정 670
버킷 리소스 액세스 674
버킷에 대한 리소스 액세스 구성 675
버킷 플랜 변경 675

Lightsail 콘솔을 사용하여 버킷의 스토리지 플랜 변경 676
를 사용하여 버킷의 스토리지 플랜 변경 AWS CLI 676
액세스 권한 구성 678
버킷 액세스 권한 구성 678
크로스 계정 액세스 680
버킷에 대한 교차 계정 액세스 구성 680
개별 객체 액세스 권한 680
개별 객체 액세스 권한 구성 681
멀티파트 업로드 682
멀티파트 업로드 프로세스 683
동시 멀티파트 업로드 작업 686
멀티파트 업로드 보존 686
Amazon Simple Storage Service 멀티파트 업로드 제한 686
업로드할 파일 분할 687
를 사용하여 멀티파트 업로드 시작 AWS CLI 687
를 사용하여 파트 업로드 AWS CLI 688
AWS CLI를 사용하여 멀티파트 업로드의 파트 나열 689
멀티파트 업로드 .json 파일 생성 691
AWS CLI를 사용하여 멀티파트 업로드 완료 692
AWS CLI를 사용하여 버킷에 대한 멀티파트 업로드 나열 693
AWS CLI를 사용하여 멀티파트 업로드 중지 694
이름 지정 규칙 695
예제 버킷 이름 696
객체 키 이름 697
키 이름 697
객체 키 명명 지침 697
XML 관련 객체 키 제한 사항 700
객체 스토리지 보안 모범 사례 701
예방적 보안 모범 사례 701
모니터링 및 감사 모범 사례 706
버킷 권한 708
버킷 액세스 권한 709
개별 객체 액세스 권한 709
크로스 계정 액세스 710
액세스 키 710
리소스 액세스 710

Amazon S3 퍼블릭 액세스 차단 710
버킷에 파일 업로드 711
객체 키 이름 및 버전 관리 711
Lightsail 콘솔을 사용하여 버킷에 파일 업로드 712
를 사용하여 버킷에 파일 업로드 AWS CLI 712
IPv6 전용 요청에 대한 AWS CLI 구성 714
Lightsail에서 버킷 및 객체 관리 715
Lightsail의 CORS 717
CORS 사용 사례 717
Lightsail이 CORS 구성을 평가하는 방법 717
AWS CLI를 사용하여 CORS 구성 719
CORS 문제 해결 721
컨테이너 서비스 수 723
컨테이너 724
Lightsail 컨테이너 서비스 요소 724
Lightsail 컨테이너 서비스 724
컨테이너 서비스 용량(규모 및 성능) 725
가격 책정 726
배포 726
배포 버전 727
컨테이너 이미지 소스 727
컨테이너 서비스 ARN 728
퍼블릭 엔드포인트 및 기본 도메인 728
사용자 지정 도메인 및 SSL/TLS 인증서 729
컨테이너 로그 730
Metrics 730
Lightsail 컨테이너 서비스 사용 730
컨테이너 생성 731
컨테이너 서비스 용량(규모 및 성능) 731
가격 책정 732
컨테이너 서비스 상태 732
컨테이너 서비스 생성 733
컨테이너 이미지 736
1단계: 필수 구성 요소 완성 736
2단계: Dockerfile 생성 및 컨테이너 이미지 구축 737
3단계: 새 컨테이너 이미지 실행 739

(선택 사항) 4단계: 로컬 시스템에서 실행 중인 컨테이너 정리 739
컨테이너 이미지를 생성한 후의 다음 단계 740
컨테이너 이미지 관리 740
컨테이너 서비스 플러그인 설치 745
ECR 프라이빗 리포지토리 액세스 752
컨테이너 및 배포 관리 769
사전 조건 770
배포 파라미터 770
컨테이너 간 통신 774
컨테이너 로그 775
배포 버전 775
배포 상태 775
배포 실패 776
현재 컨테이너 서비스 배포 확인 776
컨테이너 서비스 배포 생성 또는 수정 776
컨테이너 용량 변경 778
배포 버전 확인 및 관리 779
컨테이너 로그 확인 781
컨테이너 서비스 사용자 지정 도메인 수 783
컨테이너 서비스 사용자 지정 도메인 제한 784
사전 조건 784
컨테이너 서비스용 사용자 지정 도메인 확인 784
컨테이너 서비스용 사용자 지정 도메인 사용 785
컨테이너 서비스용 사용자 지정 도메인 사용 중지 786
Lightsail 도메인을 컨테이너에 지정 787
Route 53 도메인을 컨테이너에 지정 789
컨테이너 삭제 794
컨테이너 서비스 삭제 794
보안 796
인프라 보안 796
복원성 797
Identity and Access Management(IAM) 797
대상 797
자격 증명을 통한 인증 798
정책을 사용하여 액세스 관리 800
AWS 관리형 정책 803

Lightsail 정책 및 역할 806
IAM 사용자 액세스 관리 827
업데이트 관리 833
인스턴스 블루프린트 소프트웨어 지원 833
규정 준수 확인 835
AWS PrivateLink 835
고려 사항 836
인터페이스 엔드포인트 생성 836
AWS CLI 예제 836
엔드포인트 정책을 생성 837
성능 모니터링 839
효과적으로 리소스 모니터링 839
지표 개념 및 용어 840
Metrics 840
지표 보존 기간 840
통계 840
단위 841
기간 841
경보 841
Lightsail에서 사용할 수 있는 지표 842
인스턴스 지표 842
데이터베이스 지표 843
배포 지표 843
로드 밸런서 지표 844
컨테이너 서비스 지표 845
버킷 지표 845
리소스 상태 지표 846
인스턴스 지표 846
데이터베이스 지표 847
배포 지표 848
로드 밸런서 지표 848
컨테이너 서비스 지표 849
버킷 지표 849
지표 알림 850
인스턴스 지표 확인 851
지표 경보 855

인스턴스 경보 생성 ... 865
경보 삭제 또는 비활성화 ... 870
버킷 지표 ... 871
버킷 지표 ... 871
Lightsail 콘솔에서 버킷 지표 보기 ... 872
버킷 및 객체 관리 ... 872
경보 생성 ... 874
컨테이너 지표 ... 878
컨테이너 서비스 지표 ... 879
Lightsail 콘솔에서 컨테이너 서비스 지표 확인 ... 879
데이터베이스 지표 ... 880
데이터베이스 지표 ... 880
Lightsail 콘솔에서 데이터베이스 지표 보기 ... 880
데이터베이스 지표 확인 후의 다음 단계 ... 881
데이터베이스 경보 생성 ... 881
배포 지표 ... 887
배포 지표 ... 887
Lightsail 콘솔에서 배포용 지표 확인 ... 888
배포 지표 확인 후의 다음 단계 ... 888
배포 경보 생성 ... 889
로드 밸런서 지표 ... 894
로드 밸런서 지표 ... 894
로드 밸런서 지표 확인 ... 895
다음 단계 ... 895
로드 밸런서 경보 ... 896
알림 연락처 추가 ... 901
리전별 알림 연락처 제한 ... 902
SMS 문자 메시지 지원 ... 902
이메일 연락처 확인 ... 903
Lightsail 콘솔을 사용하여 알림 연락처 추가 ... 904
를 사용하여 알림 연락처 추가 AWS CLI ... 909
알림 연락처 추가 후의 다음 단계 ... 911
알림 연락처 삭제 ... 911
Lightsail 콘솔을 사용하여 알림 연락처 삭제 ... 912
를 사용하여 알림 연락처 삭제 AWS CLI ... 912
알림 연락처 삭제 후의 다음 단계 ... 913

Lightsail 경보 알림 검토 ... 913
활성 경보에 대한 경보 알림 검토 ... 914
확인 보류 중인 이메일 연락처 검토 ... 914
Tags ... 915
태그를 사용하여 결제 구성 및 액세스 제어 ... 915
태그 지정을 지원하는 Lightsail 리소스 ... 916
태그 제한 ... 917
태그 추가 ... 917
다음 단계 ... 919
태그 삭제 ... 920
권한 및 태그 기반 권한 부여 ... 921
태그를 사용하여 액세스 제어 ... 921
1단계: IAM 정책 생성 ... 922
2단계: 사용자 또는 그룹에 정책 연결 ... 923
태그를 사용하여 비용 구성 ... 924
1단계: 리소스에 키-값 태그 추가 ... 924
2단계: 사용자 정의 비용 할당 태그 활성화 ... 925
3단계: 비용 할당 보고서 설정 및 확인 ... 925
태그를 사용하여 리소스 구성 ... 925
리소스에 대한 태그 보기 ... 925
태그를 사용하여 리소스 필터링 ... 926
문제 해결 ... 928
WordPress 설정 ... 928
일반적인 오류 ... 929
설정 실패 ... 932
403 오류(권한 없음) ... 941
블록 스토리지 디스크 ... 941
일반 디스크 오류 ... 941
브라우저 기반 SSH 및 RDP 클라이언트 ... 943
오류 메시지: 연결할 수 없음 ... 943
오류 메시지: 현재 연결할 수 없음 ... 945
Ghost 서비스 사용 불가 ... 946
Ghost 서비스 시작 ... 946
IAM 문제 ... 948
Lightsail에서 작업을 수행할 권한이 없음 ... 949
iam:PassRole을 수행하도록 인증되지 않음 ... 949

액세스 키를 보아야 합니다. 950
관리자인데, 다른 사용자가 Lightsail에 액세스할 수 있게 허용하려고 합니다 950
내 AWS 계정 외부의 사람이 내 Lightsail 리소스에 액세스하도록 허용하고 싶습니다. 951
IPv6 도달 가능성 951
이중 스택 인스턴스에 IPv6 활성화 952
인스턴스의 방화벽 구성 953
인스턴스에 대한 연결 가능성 테스트 954
인스턴스 용량 부족 오류 957
새 인스턴스 시작 시 용량 부족 957
중지된 인스턴스 시작 시 용량 부족 958
관련 정보 958
로드 밸런서 958
일반적인 로드 밸런서 오류 958
알림 959
SSL/TLS 인증서 961
자습서 962
빠른 시작 설명서 963
AlmaLinux 963
cPanel & WHM 972
Drupal 986
Ghost 995
GitLab CE 1008
Joomla! 1019
LAMP 1030
Magento 1037
Nginx 1054
Node.js 1062
OpenClaw 1070
Plesk 1090
PrestaShop 1098
Redmine 1114
Ruby on Rails 1123
WordPress 1126
WordPress 멀티사이트 1137
Bitnami 1145
Bitnami 사용자 이름 및 암호 1146

Bitnami 배너 제거 1153
WordPress 1156
WordPress 구성 1157
Amazon S3에 연결 1170
Aurora DB에 연결 1180
MySQL에 연결 1188
스토리지 버킷에 연결 1193
CDN 구성 1207
이메일 활성화 1211
HTTPS 활성화 1223
Lightsail으로 마이그레이션 1233
WordPress 멀티사이트 1242
WordPress Multisite: 블로그를 도메인으로 추가 1242
WordPress Multisite: 하위 도메인으로 블로그 추가 1249
WordPress 멀티사이트: 도메인 정의 1253
Let's Encrypt 1255
LAMP Let's Encrypt 인증서 1256
NGINX Let's Encrypt 인증서 1269
WordPress Let's Encrypt 인증서 1296
IPv6 네트워킹 1311
cPanel 및 WHM용 IPv6 1311
GitLab용 IPv6 1318
Nginx용 IPv6 1321
Plesk용 IPv6 1325
Lightsail의 경우AWS CLI 1328
1단계: 설치 AWS CLI 1329
2단계: 새 액세스 키 생성 1329
3단계: 구성 AWS CLI 1330
다음 단계 1331
AWS CloudShell 1331
영구 스토리지 1331
AWS 리전 1332
시작 및 사용 AWS CloudShell 1333
추가 정보 1335
LAMP 시작 및 구성 1336
1단계: AWS에 가입 1336

2단계: LAMP 인스턴스 생성 1336
3단계: SSH를 통해 인스턴스에 연결하고 LAMP 인스턴스에 대한 애플리케이션 암호 가져오기 1339
4단계: LAMP 인스턴스에 애플리케이션 설치 1341
5단계: 고정 IP 주소 생성 및 LAMP 인스턴스에 연결 1341
6단계: DNS 영역 생성 및 LAMP 인스턴스에 도메인 매핑 1342
다음 단계 1343
LAMP 인스턴스를 Aurora 데이터베이스에 연결 1343
Windows Server 2016 시작 및 구성 1348
1단계: AWS에 가입 1349
2단계: Lightsail에서 Windows Server 2016 인스턴스 만들기 1349
3단계: RDP로 Windows Server 2016 인스턴스에 연결 1351
4단계: 고정 IP 주소 생성 및 Windows Server 2016 인스턴스 연결 1353
5단계: DNS 영역 생성 및 Windows Server 2016 인스턴스에 도메인 매핑 1355
다음 단계 1356
CloudTrail 로깅 1356
CloudTrail의 Lightsail 정보 1356
Lightsail 로그 파일 항목 이해 1357
HAR 파일 생성 1358
1단계: 브라우저에서 HAR 파일 생성 1358
2단계: HAR 파일을 편집하여 민감한 정보 제거 1360
3단계: 검토를 위해 HAR 파일 제출 1360
Prometheus 설치 1360
1단계: 필수 구성 요소 완성 1361
2단계: Lightsail 인스턴스에 사용자와 로컬 시스템 디렉터리 추가 1361
3단계: Prometheus 바이너리 패키지 다운로드 1362
4단계: Prometheus 구성 1365
5단계: Prometheus 시작 1367
6단계: Node Exporter 시작 1370
7단계: Node Exporter 데이터 수집기로 Prometheus 구성 1371
scp를 사용하여 파일 전송 1374
사전 조건 1374
1단계: 프라이빗 키(.pem) 파일을 로컬 컴퓨터에 저장 1375
2단계: 프라이빗 키의 권한 변경 1376
3단계: 프라이빗 키를 인스턴스로 전송 1376
4단계: Lightsail Linux 인스턴스와 Unix 인스턴스 간에 파일을 안전하게 전송 1378

다른 AWS 서비스와 함께 작동 1379
가상 머신(가상 프라이빗 서버) 1379
서버리스 컴퓨팅 1380
데이터베이스 수 1381
로드 밸런서 1381
빅 데이터 1382
스토리지 1383
모니터링 및 경보 1384
애플리케이션 배포 1384
애플리케이션 컨테이너 1384
보안 및 사용자 로그인 1385
소스 제어 및 애플리케이션 수명 주기 관리 1386
대기열 및 메시징 1386
워크플로 1387
스트리밍 애플리케이션 1387
AWS CloudFormation 리소스 1388
Lightsail 및 CloudFormation 템플릿 1388
에 대해 자세히 알아보기 CloudFormation 1388
Lightsail에 대한 추가 정보 1389
블로그 1389
튜토리얼 1391
동영상 1394
결제 1396
세부 Lightsail 청구서 보기 1396
결제 사용 유형 1397
청구서의 리전 코드 1398
FAQ 1400
Lightsail 정보 1400
Amazon Lightsail이란 무엇인가요? 1400
Lightsail으로 할 수 있는 작업은 무엇입니까? 1401
Lightsail이 API를 제공하나요? 1401
Lightsail에 가입해야 하나요? 1401
어떤에서 AWS 리전 를 Lightsail 사용할 수 있나요? 1401
가용 영역이란? 1401
Lightsail 서비스 할당량이란 무엇인가요? 1401
어떻게 더 많은 도움을 구할 수 있습니까? 1402

결제 및 계정 관리 1402
Lightsail 요금제에 따른 가격은 얼마나 되나요? 1402
어떤 요금제에 가입한 경우 언제 요금이 청구됩니까? 1402
Lightsail 인스턴스를 무료로 사용해 볼 수 있나요? 1403
Lightsail 무료 체험판은 언제 시작되나요? 1403
Lightsail 관리형 데이터베이스 비용은 어떻게 되나요? 1403
Lightsail 관리형 데이터베이스를 무료로 사용해 볼 수 있나요? 1403
Lightsail 블록 스토리지의 요금은 어떻게 되나요? 1404
Lightsail 로드 밸런서 요금은 어떻게 되나요? 1404
인증서 관리의 비용은 어떻게 됩니까? 1404
Lightsail 고정 IPv4 주소의 사용 요금은 얼마인가요? 1404
Lightsail 도메인 비용은 어떻게 되나요? 1404
Lightsail DNS 관리 비용은 얼마나 되나요? 1404
Lightsail 스냅샷 비용은 얼마나 되나요? 1404
내 AWS 계정을 관리하려면 어떻게 해야 하나요? 1405
활성화 및 비활성화된 옵트인 리전은 어떻게 관리할 수 있나요? 1405
비활성화된 옵트인 리전의 리소스는 어떻게 되나요? 1405
비활성화된 옵트인 리전에서 리소스는 어떻게 삭제할 수 있나요? 1405
Lightsail의 법적 사용 약관은 무엇인가요? 1405
Lightsail 요금은 어떻게 결제할 수 있나요? 1405
데이터 전송 1406
내 데이터 전송 플랜 허용량을 초과할 경우에는 어떻게 됩니까? 1406
인스턴스에는 어떤 유형의 데이터 전송에 대해 요금이 청구됩니까? 1406
인스턴스의 내 데이터 전송 허용량은 어떤 식으로 운영됩니까? 1407
인스턴스의 데이터 전송 허용량은 AWS 리전에 따라 어떻게 달라지나요? 1408
데이터 전송 비용은 얼마나 됩니까? 1408
내 데이터 전송 허용량은 로드 밸런서에서 어떤 식으로 운영됩니까? 1409
내 데이터 전송 허용량은 객체 스토리지에서 어떻게 사용하나요? 1409
배포에는 어떤 유형의 데이터 전송에 대해 요금이 청구됩니까? 1409
Lightsail의 인스턴스 데이터 전송 할당량과 배포 데이터 전송 할당량의 차이점은 무엇입니까? 1410
컨테이너 서비스 내/외부로 데이터를 전송할 때 요금이 부과되나요? 1410
블록 스토리지(디스크) 1411
Lightsail 블록 스토리지로 어떤 작업을 수행할 수 있나요? 1411
연결된 디스크는 내 Lightsail 플랜에 포함된 스토리지와 어떻게 다른가요? 1411
내 연결 디스크의 최대 크기는 어떻게 됩니까? 1412

Lightsail 인스턴스당 몇 개의 디스크를 연결할 수 있나요? 1412
하나의 디스크를 2개 이상의 인스턴스에 연결할 수 있습니까? 1412
내 디스크를 인스턴스에 연결해야 합니까? 1412
내 연결된 디스크의 크기를 늘릴 수 있습니까? 1412
Lightsail 블록 스토리지는 암호화를 제공하나요? 1412
Lightsail 블록 스토리지에서 기대할 수 있는 가용성 수준은 어떻게 되나요? 1412
내 연결된 디스크를 백업하려면 어떻게 해야 합니까? 1413
인증서 1413
Lightsail에서 제공하는 인증서를 어떻게 사용할 수 있나요? 1413
내 인증서를 검증하려면 어떻게 해야 합니까? 1413
내 도메인을 검증할 수 없는 경우에는 어떻게 됩니까? 1413
내 인증서에는 몇 개의 도메인과 하위 도메인을 추가할 수 있습니까? 1413
내 인증서에 연결된 도메인을 변경하려면 어떻게 해야 합니까? 1414
내 인증서를 갱신하려면 어떻게 해야 합니까? 1414
내 로드 밸런서를 삭제하면 대상 인증서는 어떻게 됩니까? 1414
Lightsail에서 제공한 내 인증서를 다운로드할 수 있나요? 1414
연락처 및 모니터링 알림 1414
알림이란 무엇입니까? 1414
연락처를 몇 개까지 추가할 수 있습니까? 1414
컨테이너 서비스 수 1415
Lightsail 컨테이너 서비스로 할 수 있는 작업은 무엇인가요? 1415
Lightsail 컨테이너 서비스에서 Docker 컨테이너를 실행할 수 있나요? 1415
퍼블릭 컨테이너 이미지를 Lightsail 컨테이너 서비스와 함께 사용하려면 어떻게 해야 하나요? 1415
프라이빗 컨테이너 레지스트리에서 컨테이너 이미지를 가져올 수 있나요? 1415
온디맨드 방식으로 서비스의 성능 및 규모를 변경할 수 있나요? 1415
Lightsail 컨테이너 서비스에서 생성한 HTTPS 엔드포인트 이름을 사용자 지정할 수 있나요? 1415
Lightsail 컨테이너 서비스의 HTTPS 엔드포인트에 사용자 지정 도메인을 사용할 수 있나요? 1416
Lightsail 컨테이너 서비스의 사용 요금은 얼마인가요? 1416
컨테이너 서비스를 며칠만 사용해도 한 달 요금이 부과되나요? 1416
컨테이너 서비스를 중지하는 것과 삭제하는 것에는 어떤 차이가 있나요? 1416
컨테이너 서비스가 비활성화된 상태인 경우에도 요금이 청구되나요? 1417
Lightsail 콘텐츠 전송 네트워크(CDN) 배포의 오리진으로 컨테이너 서비스를 사용할 수 있나요? 1417

컨테이너 서비스를 Lightsail 로드 밸런서의 대상으로 사용할 수 있나요? 1417
HTTP 요청을 HTTPS로 리디렉션하도록 컨테이너 서비스의 퍼블릭 엔드포인트를 구성할 수 있나요? 1417
컨테이너 서비스가 모니터링 및 알림을 지원하나요? 1417
Lightsail 컨테이너 서비스는 IPv6를 지원하나요? 1417
콘텐츠 전송 네트워크 배포 1417
Lightsail CDN 배포로 수행할 수 있는 작업은 무엇인가요? 1417
배포의 오리진으로 사용할 수 있는 리소스 유형은 무엇인가요? 1418
고정 IPv4 주소를 Lightsail 배포의 오리진으로 사용하려면 Lightsail 인스턴스에 연결해야 하나요? 1418
WordPress 웹 사이트에서 Lightsail 배포를 설정하려면 어떻게 해야 하나요? 1418
여러 오리진을 연결할 수 있나요? 1418
Lightsail 배포에서 인증서 생성을 지원하나요? 1418
인증서가 필요하나요? 1418
생성할 수 있는 인증서 수에 제한이 있나요? 1418
HTTP 요청을 HTTPS로 리디렉션하도록 배포를 구성하려면 어떻게 해야 하나요? 1419
Apex 도메인이 내 Lightsail 배포를 가리키도록 구성하려면 어떻게 해야 하나요? 1419
배포와 연결된 플랜을 변경할 수 있나요? 1419
배포가 작동하는지 어떻게 알 수 있나요? 1419
Lightsail 배포에서 캐시된 콘텐츠를 삭제할 수 있나요? 1419
Lightsail 배포와 Amazon CloudFront 배포는 언제 사용해야 하나요? 1419
Lightsail 콘텐츠 전송 네트워크(CDN) 배포를 Amazon CloudFront로 이동할 수 있나요? 1420
Lightsail CDN의 용도는 무엇인가요? 1420
Lightsail CDN 배포는 IPv6를 지원하나요? 1421
Lightsail CDN 배포를 사용하려면 오리진에서 IPv6를 활성화해야 하나요? 1421
데이터베이스 수 1421
Lightsail 관리형 데이터베이스란 무엇인가요? 1421
Lightsail 관리형 데이터베이스로 무엇을 할 수 있나요? 1421
Lightsail에서 무엇을 자동으로 관리해 주나요? 1422
Lightsail은 어떤 종류와 버전의 데이터베이스를 지원하나요? 1422
Lightsail이 제공하는 관리형 데이터베이스 플랜은 무엇인가요? 1422
고가용성 플랜이란 무엇입니까? 1422
Lightsail 관리형 데이터베이스는 어떻게 확장 또는 축소하나요? 1423
Lightsail 관리형 데이터베이스는 어떻게 백업할 수 있나요? 1423
Lightsail 관리형 데이터베이스를 삭제하면 데이터는 어떻게 되나요? 1423

다른 AWS 리전 가용 영역에서 실행 중인 Lightsail 관리형 데이터베이스에 인스턴스(들)를 연결할 수 있나요? 1423
내 Lightsail 관리형 데이터베이스에 데이터를 어떻게 로드하나요? 1424
내 Lightsail 관리형 데이터베이스의 데이터에 어떻게 액세스하나요? 1424
Lightsail 관리형 데이터베이스는 Lightsail 인스턴스와 어떻게 연동되나요? 1424
내 AWS 계정에서 실행 중인 EC2 인스턴스에 Lightsail 관리형 데이터베이스를 연결하려면 어떻게 해야 하나요? 1424
내 Lightsail 관리형 데이터베이스의 퍼블릭 모드와 프라이빗 모드의 차이는 무엇인가요? ... 1425
내 Lightsail 관리형 데이터베이스에서 사용하는 포트를 관리할 수 있나요? 1425
Lightsail 관리형 데이터베이스 서비스는 IPv6를 지원하나요? 1425
도메인 1425
Lightsail 도메인에서 무엇을 할 수 있나요? 1425
어떤 최상위 도메인(TLD)을 사용할 수 있습니까? 1425
Lightsail을 기존 도메인에 대한 DNS 서비스로 설정할 수 있나요? 1425
Lightsail에서 도메인 등록을 어떻게 시작하나요? 1426
Lightsail 또는 Route 53 중에서 도메인 등록 선택의 기준은 무엇인가요? 1426
도메인을 Lightsail로 이전할 수 있나요? 1426
도메인에 사용할 수 있는 Lightsail 리소스에는 무엇이 있나요? 1426
Amazon EC2로 리소스 내보내기 1426
Amazon EC2로 내보내기란? 1426
Amazon EC2로 내보내는 이유는 무엇입니까? 1426
Amazon EC2로 내보내기는 어떻게 작동하나요? 1427
요금은 어떻게 청구됩니까? 1427
관리형 데이터베이스나 디스크 스냅샷을 내보낼 수 있습니까? 1427
어떤 Lightsail 리소스를 내보낼 수 있나요? 1427
인스턴스 1428
Lightsail 인스턴스란 무엇인가요? 1428
Lightsail 요금제란 무엇인가요? 1428
내 인스턴스에서는 어떤 소프트웨어를 실행할 수 있습니까? 1428
어떤 운영 체제에서 Lightsail을 사용할 수 있나요? 1428
Lightsail 인스턴스를 사용하려면 자체 라이선스를 사용해야 하나요? 1428
Lightsail 인스턴스는 어떻게 만들 수 있나요? 1429
Lightsail 인스턴스는 어떻게 실행되나요? 1429
인스턴스가 버스팅 중인지 어떻게 알 수 있습니까? 1429
Lightsail 인스턴스에 어떻게 연결하나요? 1429
내 인스턴스는 어떻게 백업할 수 있습니까? 1430

내 요금제를 업그레이드할 수 있습니까? 1430
내 AWS 계정의 다른 리소스에 Lightsail 인스턴스를 연결하려면 어떻게 해야 하나요? 1430
내 인스턴스를 중지하는 것과 삭제하는 것은 어떤 차이가 있나요? 1430
로드 밸런서 1431
Lightsail 로드 밸런서로 어떤 작업을 수행할 수 있나요? 1431
서로 다른 AWS 리전 가용 영역에 있는 인스턴스에서 로드 밸런서를 사용할 수 있나요? 1431
Lightsail 로드 밸런서는 트래픽 스파이크를 어떻게 처리하나요? 1431
Lightsail 로드 밸런서는 어떻게 트래픽을 내 대상 인스턴스로 라우팅하나요? 1432
Lightsail은 내 대상 인스턴스가 정상인지 어떻게 알 수 있나요? 1432
내 로드 밸런서에 인스턴스를 몇 개나 연결할 수 있습니까? 1432
하나의 인스턴스를 여러 로드 밸런서에 할당할 수 있습니까? 1432
내 로드 밸런서를 삭제하면 대상 인스턴스는 어떻게 됩니까? 1432
세션 지속성이란 무엇입니까? 1432
Lightsail 로드 밸런서에는 어떤 종류의 연결을 지원하나요? 1433
Lightsail 로드 밸런서가 IPv6를 지원하나요? 1433
IPv6를 활성화한 로드 밸런서를 사용하려면 로드 밸런서 뒤에 인스턴스에 IPv6를 활성화해야 하나요? 1433
스냅샷 1433
스냅샷이란 무엇입니까? 1433
자동 스냅샷이란 무엇입니까? 1434
수동 스냅샷과 자동 스냅샷의 차이는 무엇입니까? 1434
어떤 리소스가 스냅샷을 지원합니까? 1434
스냅샷을 얼마나 오래 저장할 수 있습니까? 1434
자동 스냅샷을 어떻게 활성화합니까? 1434
언제 자동 스냅샷이 생성됩니까? 1435
몇 개의 스냅샷을 저장할 수 있습니까? 1435
스냅샷 요금은 어떻게 청구됩니까? 1435
자동 스냅샷을 비활성화하면 스냅샷이 손실됩니까? 1435
자동 스냅샷이 교체되지 않도록 하려면 어떻게 해야 합니까? 1435
자동 스냅샷을 삭제할 수 있습니까? 1435
스냅샷을 사용하려면 어떻게 해야 합니까? 1435
지표 및 경보 1436
지표란 무엇입니까? 1436
경보란 무엇입니까? 1436
경보를 몇 개까지 추가할 수 있습니까? 1436
네트워킹 1436

Lightsail에서 IP 주소를 사용하려면 어떻게 해야 하나요? 1436
Lightsail은 IPv6 전용 인스턴스를 지원하나요? 1437
고정 IP란 무엇입니까? 1437
인스턴스에 연결할 수 있는 고정 IP는 몇 개인가요? 1437
DNS 레코드란? 1437
내 인스턴스에 대한 방화벽 설정을 관리할 수 있습니까? 1437
객체 스토리지(버킷) 1438
Lightsail 객체 스토리지로 할 수 있는 작업은 무엇인가요? 1438
Lightsail 객체 스토리지의 비용은 어떻게 됩니까? 1438
Lightsail 버킷과 연결된 플랜을 변경할 수 있나요? 1438
Lightsail 객체 스토리지에서 Amazon S3로 객체를 복사할 수 있나요? 1438
Lightsail 객체 스토리지를 시작하려면 어떻게 해야 하나요? 1438
버킷에 객체를 업로드하려면 어떻게 해야 하나요? 1439
내 버킷에 대한 퍼블릭 액세스를 차단할 수 있나요? 1439
버킷에 프로그래밍 방식으로 액세스하려면 어떻게 해야 하나요? 1439
버킷을 다른 AWS 계정과 공유하려면 어떻게 해야 하나요? 1439
버전 관리란 무엇인가요? 1439
Lightsail 버킷을 Lightsail CDN 배포에 연결하려면 어떻게 해야 하나요? 1439
Lightsail 객체 스토리지 서비스에는 어떤 제한이 있나요? 1440
Lightsail 객체 스토리지가 모니터링 및 알림을 지원하나요? 1440
Lightsail의 태그 1440
태그란 무엇입니까? 1440
Lightsail에서 태그를 사용하려면 어떻게 해야 하나요? 1440
어떤 리소스에 태그를 지정할 수 있습니까? 1441
Lightsail 스냅샷에 태그를 지정하려면 어떻게 해야 하나요? 1441
키-값 태그와 키 전용 태그의 차이점은 무엇입니까? 1441
문서 기록 1442
지원 받기 1443
상황에 맞는 도움말 패널 1443
사용 설명서 정보 1443
검색 사용 1444
Lightsail CLI 및 API 사용 1444
AWS 포럼 및 기타 커뮤니티 리소스 1444
..... mcdxlv

Amazon Lightsail이란?

Amazon Lightsail은 웹 사이트 또는 웹 애플리케이션을 구축해야 하는 개발자가 Amazon Web Services(AWS)를 시작하는 가장 쉬운 방법입니다. 이를 통해 인스턴스(가상 프라이빗 서버), 컨테이너 서비스, 관리형 데이터베이스, 콘텐츠 전송 네트워크(CDN) 배포, 로드 밸런서, SSD 기반 블록 스토리지, 고정 IP 주소, 등록된 도메인의 DNS 관리, 리소스 스냅샷(백업) 등 프로젝트를 신속하게 시작하는 데 필요한 모든 것을 저렴하고 예측 가능한 요금으로 사용할 수 있습니다.

Lightsail은 Amazon Lightsail for Research도 제공합니다. Lightsail for Research를 사용하면 학계와 연구원은 AWS 클라우드에서 강력한 가상 컴퓨터를 생성할 수 있습니다. 이러한 가상 컴퓨터에는 RStudio 및 Scilab과 같은 연구 애플리케이션이 사전 설치되어 있습니다. 자세한 내용은 Amazon Lightsail에 대한 Research User Guide를 참조하세요.

주제

- Lightsail의 기능
- Lightsail은 누구를 대상으로 합니까?
- Lightsail 시작하기
- 관련 서비스
- 추정, 결제 및 비용 최적화

Lightsail의 기능

Lightsail은 다음의 대략적인 기능을 제공합니다.

인스턴스

Lightsail는의 성능과 안정성으로 쉽게 설정하고 지원할 수 있는 가상 프라이빗 서버(인스턴스)를 제공합니다 AWS. 웹 사이트, 웹 애플리케이션 또는 프로젝트를 몇 분 안에 시작하고 직관적인 Lightsail 콘솔 또는 API에서 인스턴스를 관리할 수 있습니다.

인스턴스를 생성할 때 WordPress, Windows, Plesk, LAMP, Nginx 등과 같은 간단한 운영 체제(OS), 사전 구성된 애플리케이션 또는 개발 스택을 클릭하여 시작할 수 있습니다. 모든 Lightsail 인스턴스에는 소스 IP, 포트 및 프로토콜을 기반으로 인스턴스에 대한 트래픽을 허용하거나 제한하는 데 사용할 수 있는 방화벽이 내장되어 있습니다. 자세히 알아보기

컨테이너

클라우드에서 컨테이너화된 애플리케이션을 실행하고 안전하게 액세스합니다. 컨테이너는 애플리케이션이 한 컴퓨팅 환경에서 다른 컴퓨팅 환경으로 빠르고 안정적으로 실행되도록 코드와 종속성을 함께 패키징하는 표준 소프트웨어 단위입니다. 자세히 알아보기

로드 밸런서

웹 사이트와 애플리케이션이 트래픽의 변동을 수용하고 중단으로부터 보호하며 원활한 방문자 경험을 제공할 수 있도록 인스턴스 간에 웹 트래픽을 라우팅합니다. 자세히 알아보기

관리형 데이터베이스

Lightsail은 메모리, 처리, 스토리지 및 전송 허용량을 포함하는 완전히 구성된 MySQL 또는 PostgreSQL 데이터베이스 계획을 제공합니다. Lightsail 관리형 데이터베이스를 사용하면 가상 서버와 관계없이 데이터베이스를 쉽게 확장하거나, 애플리케이션 가용성을 개선하거나, 클라우드에서 독립 실행형 데이터베이스를 실행할 수 있습니다. 자세히 알아보기

블록 및 객체 스토리지

Lightsail은 블록 스토리지와 객체 스토리지를 모두 제공합니다. Linux 또는 Windows 가상 서버를 위한 고가용성 SSD 지원 스토리지를 사용하여 스토리지를 빠르고 쉽게 확장할 수 있습니다. 자세히 알아보기

Lightsail 객체 스토리지 버킷으로 인터넷상의 어디서나 언제든지 객체를 저장하고 검색할 수 있습니다. 클라우드에서 정적 콘텐츠를 호스팅할 수도 있습니다. 자세히 알아보기

CDN 배포

Lightsail은 Amazon CloudFront와 동일한 인프라에 구축된 콘텐츠 전송 네트워크(CDN) 배포를 활성화합니다. 전 세계에 프록시 서버를 설정하여 사용자가 웹 사이트에 지리적으로 더 가깝게 액세스할 수 있도록 하여 콘텐츠를 글로벌 대상에게 쉽게 배포하여 지연 시간을 줄일 수 있습니다. 자세히 알아보기

AWS 제품 액세스

Lightsail은 인스턴스, 관리형 데이터베이스 및 로드 밸런서와 같은 집중된 기능 세트를 사용하여 더 쉽게 시작할 수 있습니다. 그러나 이러한 옵션으로 제한되지는 않습니다. Amazon VPC 피어링을 AWS 통해 프로젝트를의 90개 이상의 다른 서비스 중 일부Lightsail와 통합할 수 있습니다. 자세히 알아보기

Lightsail에 대한 자세한 내용은 Amazon Lightsail 섹션을 참조하세요.

Lightsail은 누구를 대상으로 합니까?

Lightsail은 모두를 위한 것입니다. 소프트웨어나 프레임워크를 설치하는 만큼 많은 시간을 쓸 필요가 없도록 개발 프로젝트를 촉진시키는 Lightsail 인스턴스의 이미지를 선택할 수 있습니다.

개인으로 활동하는 개발자이거나 취미로 개인적 프로젝트를 진행하는 분이라면 Lightsail이 기본 클라우드 리소스를 배포하고 관리하는 데 도움이 될 수 있습니다. 가상 머신, 도메인 또는 네트워킹과 같은 클라우드 서비스로 실험하거나 학습하는 데 관심을 가진 분에게도 유용합니다. Lightsail을 통해 즉시 시작할 수 있습니다.

Lightsail에는 기본 운영 체제, LAMP/LEMP(Nginx)/SQL Server Express 같은 개발 스택, 그리고 WordPress/Drupal/Magento 등의 애플리케이션이 포함된 이미지가 있습니다. 각 이미지에 설치된 소프트웨어에 대한 자세한 내용은 Lightsail 인스턴스 이미지 선택을 참조하세요.

프로젝트의 규모가 커짐에 따라 블록 스토리지 디스크를 추가하고 이를 Lightsail 인스턴스에 연결할 수 있습니다. 이러한 인스턴스와 디스크의 스냅샷을 캡처한 다음 그 스냅샷을 토대로 새 인스턴스를 손쉽게 만들 수 있습니다. Lightsail 인스턴스가 외부의 다른 AWS 리소스를 사용할 수 있도록 VPC를 피어링할 수도 있습니다Lightsail.

또한 Lightsail 로드 밸런서를 생성하고 대상 인스턴스를 연결하여 고가용성 애플리케이션을 구축할 수 있습니다. 암호화된(HTTPS) 트래픽, 세션 지속성, 상태 확인 등을 처리하도록 로드 밸런서를 구성할 수도 있습니다.

Lightsail 시작하기

Lightsail을 사용하도록 설정한 후에는 Lightsail에서 가상 프라이빗 서버 시작하기의 내용에 따라 인스턴스를 시작, 연결 및 정리할 수 있습니다. Lightsail에 액세스하는 방법에 대한 자세한 내용은 Lightsail 액세스 섹션을 참조하세요.

관련 서비스

Lightsail를 사용하여 인스턴스, 디스크 같은 Lightsail 리소스를 직접 프로비저닝할 수 있습니다. 또한 다음과 같은 다른 AWS 서비스를 사용하여 리소스를 프로비저닝할 수 있습니다.

- Amazon EC2

  소프트웨어 시스템의 구축 및 호스팅에 사용할 수 있도록 규모를 변경할 수 있는 컴퓨팅 용량, 즉 Amazon 데이터 센터의 서버를 제공합니다. Amazon EC2와 Lightsail을 비교하려면 Amazon Lightsail 또는 Amazon EC2 섹션을 참조하세요.

- Amazon EC2 Auto Scaling

  애플리케이션의 로드를 처리할 수 있는 정확한 수의 Amazon EC2 인스턴스를 유지하는 데 도움이 됩니다.

- Elastic Load Balancing

  수신되는 애플리케이션 트래픽을 여러 인스턴스로 자동 분산합니다.

- Amazon Relational Database Service(RDS)

  클라우드에서 관리되는 관계형 데이터베이스를 쉽게 생성, 운영하고 규모를 조정합니다.

- Amazon Elastic Container Service(Amazon ECS)

  Amazon EC2 인스턴스 클러스터에서 컨테이너화된 애플리케이션을 배포, 관리하고 규모를 조정합니다.

추정, 결제 및 비용 최적화

AWS 사용 사례에 대한 추정치를 생성하려면를 사용합니다AWS Pricing Calculator.

청구 요금은 AWS 결제 및 비용 관리 콘솔의 청구 및 비용 관리 대시보드에서 확인할 수 있습니다. 청구서에는 요금 내역을 자세하게 확인할 수 있는 사용 보고서 링크가 포함됩니다. AWS 계정 결제에 대한 자세한 내용은 AWS Billing and Cost Management 사용 설명서를 참조하세요.

AWS 결제, 계정 및 이벤트에 대한 질문이 있는 경우 AWS Support에 문의하세요.

를 사용하여 AWS 환경의 비용, 보안 및 성능을 최적화할 수 있습니다AWS Trusted Advisor.

에 대한 AWS 계정 및 관리 사용자 설정 Lightsail

신규 AWS 고객인 경우 사용을 시작하기 전에이 페이지에 나열된 설정 사전 조건을 완료합니다 Amazon Lightsail. 이러한 설정 절차에는 (AWS Identity and Access Management IAM) 서비스를 사용합니다. IAM에 대한 전체 내용은 IAM 사용 설명서를 참조하십시오.

에 가입 AWS 계정

이 없는 경우 다음 단계를 AWS 계정완료하여 생성합니다.

에 가입하려면 AWS 계정

1. https://portal.aws.amazon.com/billing/signup을 엽니다.
2. 온라인 지시 사항을 따르세요.

   등록 절차 중 전화 또는 텍스트 메시지를 받고 전화 키패드로 확인 코드를 입력하는 과정이 있습니다.

   에 가입하면 AWS 계정AWS 계정 루트 사용자이 생성됩니다. 루트 사용자에게는 계정의 모든 AWS 서비스 및 리소스에 액세스할 권한이 있습니다. 보안 모범 사례는 사용자에게 관리 액세스 권한을 할당하고, 루트 사용자만 사용하여 루트 사용자 액세스 권한이 필요한 작업을 수행하는 것입니다.

AWS 는 가입 프로세스가 완료된 후 확인 이메일을 보냅니다. 언제든지 https://aws.amazon.com/으로 이동하고 내 계정을 선택하여 현재 계정 활동을 확인하고 계정을 관리할 수 있습니다.

관리자 액세스 권한이 있는 사용자 생성

에 가입한 후 일상적인 작업에 루트 사용자를 사용하지 않도록 관리 사용자를 AWS 계정보호 AWS IAM Identity Center, AWS 계정 루트 사용자활성화 및 생성합니다.

보안 AWS 계정 루트 사용자

1. 루트 사용자를 선택하고 AWS 계정 이메일 주소를 입력하여 계정 소유자AWS Management Console로에 로그인합니다. 다음 페이지에서 비밀번호를 입력합니다.

   루트 사용자를 사용하여 로그인하는 데 도움이 필요하면 AWS 로그인 사용 설명서의 루트 사용자로 로그인을 참조하세요.

2. 루트 사용자의 다중 인증(MFA)을 활성화합니다.

   지침은 IAM 사용 설명서의 AWS 계정 루트 사용자(콘솔)에 대한 가상 MFA 디바이스 활성화를 참조하세요.

관리자 액세스 권한이 있는 사용자 생성

1. IAM Identity Center를 활성화합니다.

   지침은 AWS IAM Identity Center 사용 설명서의 AWS IAM Identity Center설정을 참조하세요.
2. IAM Identity Center에서 사용자에게 관리 액세스 권한을 부여합니다.

   를 자격 증명 소스 IAM Identity Center 디렉터리 로 사용하는 방법에 대한 자습서는 사용 AWS IAM Identity Center 설명서의 기본값으로 사용자 액세스 구성을 IAM Identity Center 디렉터리 참조하세요.

관리 액세스 권한이 있는 사용자로 로그인

- IAM IDentity Center 사용자로 로그인하려면 IAM Identity Center 사용자를 생성할 때 이메일 주소로 전송된 로그인 URL을 사용합니다.

  IAM Identity Center 사용자를 사용하여 로그인하는 데 도움이 필요하면 AWS 로그인 사용 설명서의 AWS 액세스 포털에 로그인을 참조하세요.

추가 사용자에게 액세스 권한 할당

1. IAM Identity Center에서 최소 권한 적용 모범 사례를 따르는 권한 세트를 생성합니다.

   지침은 AWS IAM Identity Center 사용 설명서의 Create a permission set를 참조하세요.
2. 사용자를 그룹에 할당하고, 그룹에 Single Sign-On 액세스 권한을 할당합니다.

   지침은 AWS IAM Identity Center 사용 설명서의 그룹 추가를 참조하세요.

Lightsail 액세스

다양한 인터페이스 및 서비스 엔드포인트를 사용하여 Amazon Lightsail에 액세스할 수 있습니다.

주제

- Lightsail 서비스 인터페이스
- Lightsail 서비스 엔드포인트
- 엔드포인트 지정의 예

Lightsail 서비스 인터페이스

다음과 같은 인터페이스로 Lightsail 리소스를 생성하고 관리할 수 있습니다.

Amazon Lightsail 콘솔

Lightsail 인스턴스 및 리소스를 생성하고 관리하는 간단한 웹 인터페이스입니다. AWS 계정에 가입한 경우 Lightsail 콘솔에 직접 액세스하거나에 로그인하여 콘솔 홈 페이지에서 AWS Management Console 선택할 수 Lightsail 있습니다.

AWS Command Line Interface

명령줄 셸에서 명령을 사용하여 AWS 서비스와 상호 작용할 수 있습니다. Windows, Mac, Linux에서 지원됩니다. 에 대한 자세한 내용은 AWS Command Line Interface 사용 설명서를 AWS CLI참조하세요. CLI Lightsail 명령 참조의 섹션에서 명령을 찾을 수 있습니다. Lightsail AWS

AWS CloudShell

CloudShell은 브라우저 기반의 사전 인증된 셸로,에서 직접 시작할 수 있습니다 AWS Management Console. Bash, PowerShell 또는 Z 쉘과 같은 원하는 쉘을 사용하여 AWS CLI 명령을 실행할 수 있습니다. 를 사용하여 Lightsail 리소스를 AWS CloudShell 관리하는 방법에 대한 예는 섹션을 참조하세요를 사용하여 Lightsail 리소스 관리 AWS CloudShell.

Query API

Lightsail에서는 쿼리 API를 제공합니다. 이러한 요청은 HTTP나 HTTPS의 메시지 교환 방식인 GET이나 POST이며, 미리 정해진 이름인 "`Action`"을 쿼리 변수로 사용합니다. Lightsail의 API 작업에 대한 자세한 내용은 Amazon Lightsail API 레퍼런스의 작업(Actions)을 참조하세요.

AWS SDKs

HTTP 또는 HTTPS를 통해 요청을 제출하는 대신 언어별 APIs를 사용하여 애플리케이션을 빌드하려는 경우는 소프트웨어 개발자에게 라이브러리, 샘플 코드, 자습서 및 기타 리소스를 AWS 제공합니다. 이러한 라이브러리는 요청 암호화 서명, 요청 재시도, 오류 응답 처리와 같은 작업을 자동화하는 기본 기능을 제공하므로 더 쉽게 시작할 수 있습니다. 자세한 내용은 AWS기반의 도구를 참조하세요.

AWS Tools for PowerShell

SDK for .NET에서 공개하는 기능을 기반으로 하는 PowerShell 모듈 세트입니다. Tools for PowerShell을 사용하면 PowerShell 명령줄에서 AWS 리소스에 대한 작업을 스크립트로 작성할 수 있습니다. 시작하려면 AWS Tools for PowerShell 사용 설명서를 참조하세요. Lightsail용 cmdlet은 AWS Tools for PowerShell cmdlet 참조에서 찾아볼 수 있습니다.

Lightsail 서비스 엔드포인트

엔드포인트는 AWS 웹 서비스의 진입점 역할을 하는 URL입니다. 앞서 설명한 인터페이스의 프로그래밍 방식 액세스 방법의 경우, Lightsail은 다음 엔드포인트 유형을 지원합니다.

- IPv4 엔드포인트
- 듀얼 스택 엔드포인트(IPv4 및 IPv6를 모두 지원)

요청 시에, 사용할 엔드포인트를 지정할 수 있습니다. 엔드포인트를 지정하지 않으면 기본적으로 IPv4 엔드포인트가 사용됩니다. 다른 엔드포인트 유형을 사용하려면 요청에서 이를 지정해야 합니다.

IPv4 엔드포인트

IPv4 엔드포인트는 IPv4 트래픽만 지원합니다. IPv4 엔드포인트는 모든 리전에 사용할 수 있습니다. 리전의 서비스 엔드포인트에 대한 자세한 내용은 리전별 서비스 엔드포인트 섹션을 참조하세요.

IPv4 엔드포인트 이름에는 다음 명명 규칙이 사용됩니다.

- *service*.*region*.amazonaws.com

예를 들어 us-east-2 리전의 IPv4 엔드포인트 이름은 lightsail.us-east-2.amazonaws.com입니다.

이중 스택(IPv4 및 IPv6) 엔드포인트

이중 스택 엔드포인트는 IPv4 트래픽과 IPv6 트래픽을 모두 지원합니다. 이중 스택 엔드포인트에 요청하는 경우, 엔드포인트 URL이 네트워크 및 클라이언트에서 사용하는 프로토콜에 따라 IPv6 또는 IPv4 주소로 확인됩니다.

- `lightsail.region.api.aws`

예를 들어 `us-east-2` 리전의 이중 스택 엔드포인트 이름은 `lightsail.us-east-2.api.aws`입니다.

리전별 서비스 엔드포인트

다음은 Lightsail에 대한 서비스 엔드포인트입니다. Lightsail을 사용할 수 있는 리전에 대한 자세한 내용은 Lightsail의 리전 및 가용 영역 섹션을 참조하세요.

리전 이름	리전	엔드포인트	프로토콜
미국 동부(오하이오)	us-east-2	lightsail.us-east-2.amazonaws.com	HTTPS
		lightsail.us-east-2.api.aws	HTTPS
미국 동부(버지니아 북부)	us-east-1	lightsail.us-east-1.amazonaws.com	HTTPS
		lightsail.us-east-1.api.aws	HTTPS
미국 서부(오리곤)	us-west-2	lightsail.us-west-2.amazonaws.com	HTTPS
		lightsail.us-west-2.api.aws	HTTPS
아시아 태평양(자카르타)	ap-southeast-3	lightsail.ap-southeast-3.amazonaws.com	HTTPS
		lightsail.ap-southeast-3.api.aws	HTTPS
아시아 태평양(뭄바이)	ap-south-1	lightsail.ap-south-1.amazonaws.com	HTTPS
		lightsail.ap-south-1.api.aws	HTTPS

리전 이름	리전	엔드포인트	프로토콜
아시아 태평양(서울)	ap-northeast-2	lightsail.ap-northeast-2.amazonaws.com lightsail.ap-northeast-2.api.aws	HTTPS HTTPS
아시아 태평양(싱가포르)	ap-southeast-1	lightsail.ap-southeast-1.amazonaws.com lightsail.ap-southeast-1.api.aws	HTTPS HTTPS
아시아 태평양(시드니)	ap-southeast-2	lightsail.ap-southeast-2.amazonaws.com lightsail.ap-southeast-2.api.aws	HTTPS HTTPS
아시아 태평양(도쿄)	ap-northeast-1	lightsail.ap-northeast-1.amazonaws.com lightsail.ap-northeast-1.api.aws	HTTPS HTTPS
캐나다(중부)	ca-central-1	lightsail.ca-central-1.amazonaws.com lightsail.ca-central-1.api.aws	HTTPS HTTPS
유럽(프랑크푸르트)	eu-central-1	lightsail.eu-central-1.amazonaws.com lightsail.eu-central-1.api.aws	HTTPS HTTPS
유럽(아일랜드)	eu-west-1	lightsail.eu-west-1.amazonaws.com lightsail.eu-west-1.api.aws	HTTPS HTTPS
유럽(런던)	eu-west-2	lightsail.eu-west-2.amazonaws.com lightsail.eu-west-2.api.aws	HTTPS HTTPS
유럽(파리)	eu-west-3	lightsail.eu-west-3.amazonaws.com lightsail.eu-west-3.api.aws	HTTPS HTTPS
유럽(스톡홀름)	eu-north-1	lightsail.eu-north-1.amazonaws.com lightsail.eu-north-1.api.aws	HTTPS HTTPS

엔드포인트 지정의 예

이 섹션에서는 요청 시에 엔드포인트를 지정하는 방법을 몇 가지 예로 보여줍니다.

> **Note**
>
> 엔드포인트를 지정하지 않으면 기본적으로 IPv4 엔드포인트가 사용됩니다.

AWS CLI

다음 예는 AWS CLI를 사용하여 `us-east-2` 리전의 엔드포인트를 지정하는 방법을 보여줍니다.

- IPv4

```
aws lightsail get-regions --region us-east-2 --endpoint-url https://lightsail.us-east-2.amazonaws.com
```

- 이중 스택

```
aws lightsail get-regions --region us-east-2 --endpoint-url https://lightsail.us-east-2.api.aws
```

AWS SDK for Java 2.x

다음 예는 AWS SDK for Java 2.x를 사용하여 `us-east-2` 리전의 엔드포인트를 지정하는 방법을 보여줍니다.

- IPv4

```
LightsailClient client = LightsailClient.builder()
    .region(Region.US_EAST_2)
    .endpointOverride(URI.create("https://lightsail.us-east-2.amazonaws.com"))
    .build();
```

- 이중 스택

```
LightsailClient client = LightsailClient.builder()
    .region(Region.US_EAST_2)
    .endpointOverride(URI.create("https://lightsail.us-east-2.api.aws"))
```

```
    .build();
```

AWS SDK for Java 1.x

다음 예제에서는 AWS SDK for Java 1.x를 사용하여 us-east-2 리전의 엔드포인트를 지정하는 방법을 보여줍니다.

- IPv4

```
AmazonLightsail lightsail = AmazonLightsailClientBuilder.standard()
     .withEndpointConfiguration(new EndpointConfiguration(
          "https://lightsail.us-east-2.amazonaws.com",
          "us-east-2"))
     .build();
```

- 이중 스택

```
AmazonLightsail lightsail = AmazonLightsailClientBuilder.standard()
     .withEndpointConfiguration(new EndpointConfiguration(
          "https://lightsail.us-east-2.api.aws",
          "us-east-2"))
     .build();
```

AWS SDK for Go

다음 예는 AWS SDK for Go를 사용하여 us-east-2 리전의 엔드포인트를 지정하는 방법을 보여줍니다.

- IPv4

```
sess := session.Must(session.NewSession())
svc := lightsail.New(sess, &aws.Config{
    Region: aws.String(endpoints.UsEast2RegionID),
    Endpoint: aws.String("https://lightsail.us-east-2.amazonaws.com")
})
```

- 이중 스택

```
sess := session.Must(session.NewSession())
svc := lightsail.New(sess, &aws.Config{
```

```
    Region: aws.String(endpoints.UsEast2RegionID),
    Endpoint: aws.String("https://lightsail.us-east-2.api.aws")
})
```

Lightsail에서 가상 프라이빗 서버 시작하기

Lightsail에서 인스턴스는 가상 머신이라고도 하는 가상 프라이빗 서버입니다. AWS 클라우드에서 Lightsail 인스턴스를 생성하고 관리합니다. 인스턴스를 생성할 때 운영 체제(OS)를 가진 이미지를 선택합니다. 기본 OS를 포함하여, 애플리케이션 또는 개발 스택을 가진 인스턴스 이미지를 선택할 수도 있습니다.

이 자습서에서 생성하는 인스턴스에는 인스턴스를 생성한 시점부터 삭제할 때까지 사용 요금이 부과됩니다. 삭제는 이 자습서의 마지막 단계입니다. 요금에 대한 자세한 내용은 Lightsail 요금을 참조하세요.

주제

- 1단계: 필수 구성 요소 완성
- 2단계: 인스턴스 만들기
- 3단계: 인스턴스에 연결
- 4단계: 인스턴스에 스토리지 추가
- 5단계: 스냅샷 생성
- 6단계: 정리
- 다음 단계
- 와 Amazon Lightsail 함께 사용 AWS CLI

1단계: 필수 구성 요소 완성

신규 AWS 고객인 경우 사용을 시작하기 전에 설정 사전 조건을 완료하세요Amazon Lightsail. 자세한 내용은 에 대한 AWS 계정 및 관리 사용자 설정 Lightsail 단원을 참조하십시오.

2단계: 인스턴스 만들기

다음 절차의 설명에 따라 Lightsail 콘솔을 사용하여 인스턴스를 생성할 수 있습니다. 이 자습서는 첫 번째 인스턴스를 빠르게 시작하도록 돕기 위한 것입니다. 또한 사용 가능한 애플리케이션과 하드웨어 플랜을 살펴보는 것이 좋습니다. 자세한 내용은 Lightsail 인스턴스 블루프린트 제품 검토 단원을 참조하십시오.

1. Lightsail 콘솔에 로그인합니다.

2. 홈 페이지에서 인스턴스 생성을 선택합니다.
3. 인스턴스의 위치(AWS 리전 및 가용 영역)를 선택합니다. 지연 시간을 줄이려면 물리적 위치와 가장 가까운 AWS 리전 를 선택합니다.

   변경 AWS 리전 및 가용 영역을 선택하여 다른 위치에 인스턴스를 생성합니다.
4. 애플리케이션(앱 + OS) 또는 운영 체제(OS 전용)를 선택할 수 있습니다.

   Lightsail 인스턴스 이미지에 대한 자세한 내용은 Lightsail 인스턴스 블루프린트 제품 검토 섹션을 참조하세요.
5. 인스턴스 플랜을 선택합니다.

   인스턴스가 이중 스택(IPv4 및 IPv6)을 사용할지 IPv6 전용을 사용할지 선택합니다. 일부 Lightsail 블루프린트는 현재 IPv6 전용 네트워킹을 지원하지 않습니다. IPv6 전용 네트워킹을 지원하는 블루프린트를 확인하려면 Lightsail 인스턴스 블루프린트 제품 검토 섹션을 참조하세요.

   5 USD Lightsail 플랜을 한 달 동안 무료로 체험할 수 있습니다(최대 750시간). 귀하의 계정에 1개월 무료 이용권을 드리겠습니다. Lightsail 요금 페이지에 대해 자세히 알아보세요.
6. 인스턴스 이름을 입력합니다.

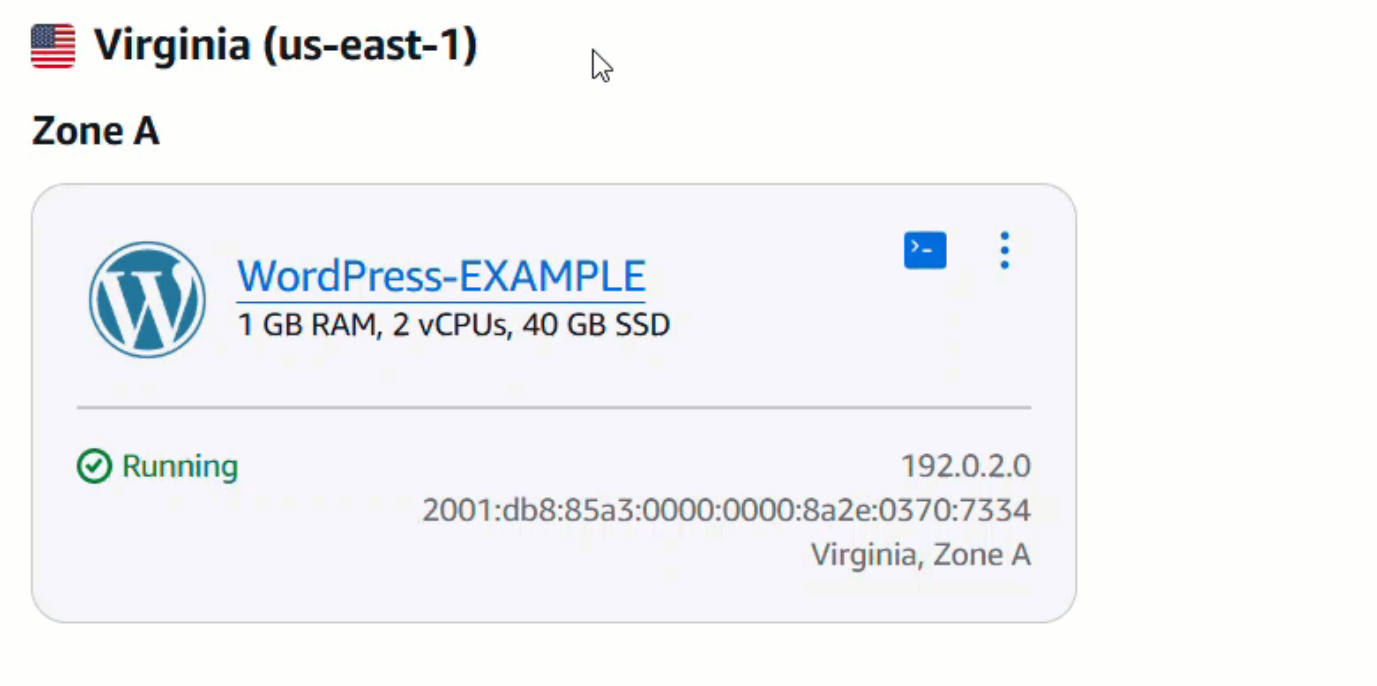
   리소스 이름:

   - Lightsail 계정 AWS 리전 의 각 내에서 고유해야 합니다.
   - 2~255자의 문자로 구성되어야 합니다.
   - 영숫자 문자 또는 숫자로 시작하고 끝나야 합니다.
   - 영숫자 문자, 숫자, 마침표, 대시, 밑줄이 포함될 수 있습니다.
7. 인스턴스 생성을 선택합니다.

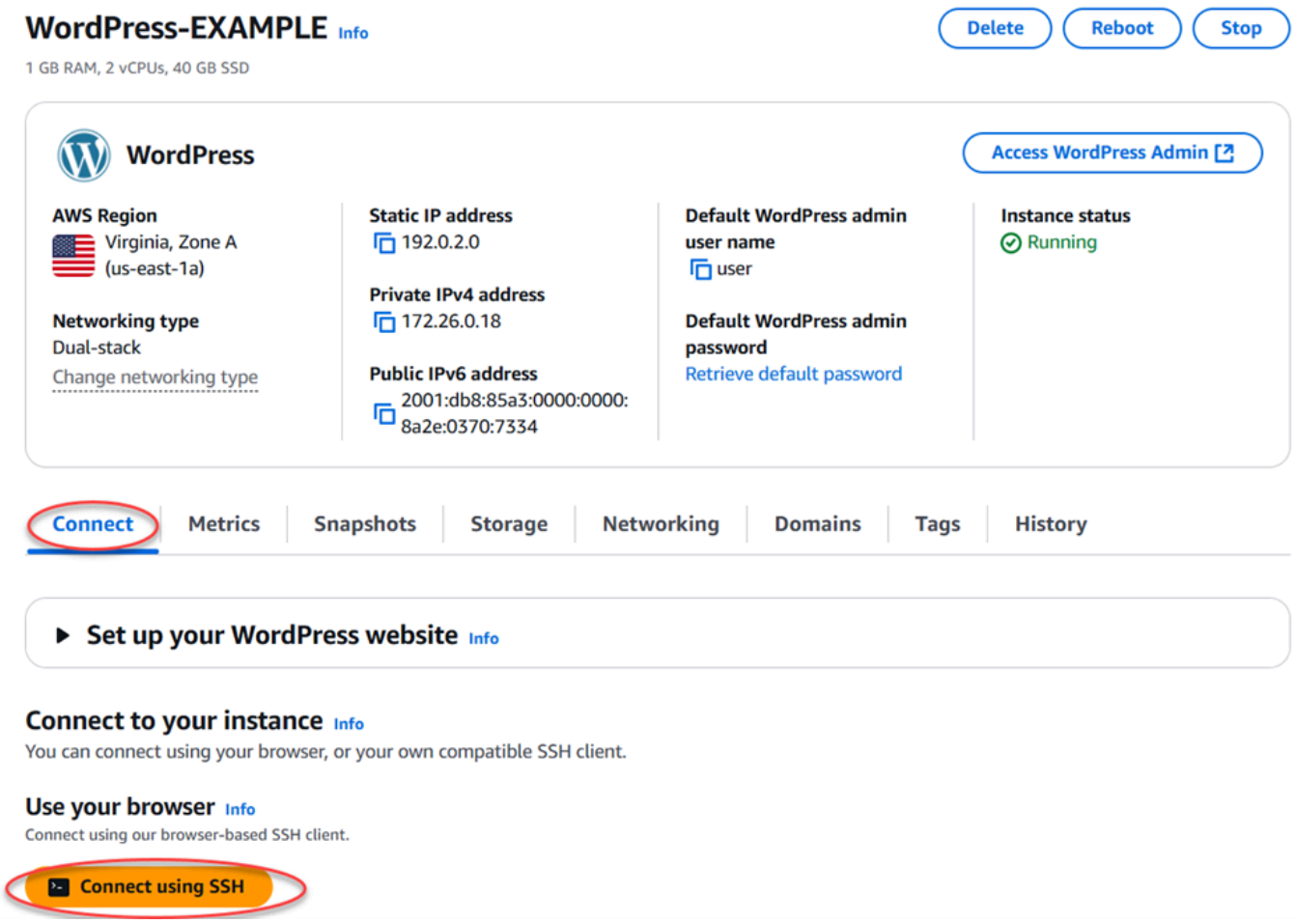
Lightsail 인스턴스는 몇 분 내에 준비되며 연결할 수 있습니다.

3단계: 인스턴스에 연결

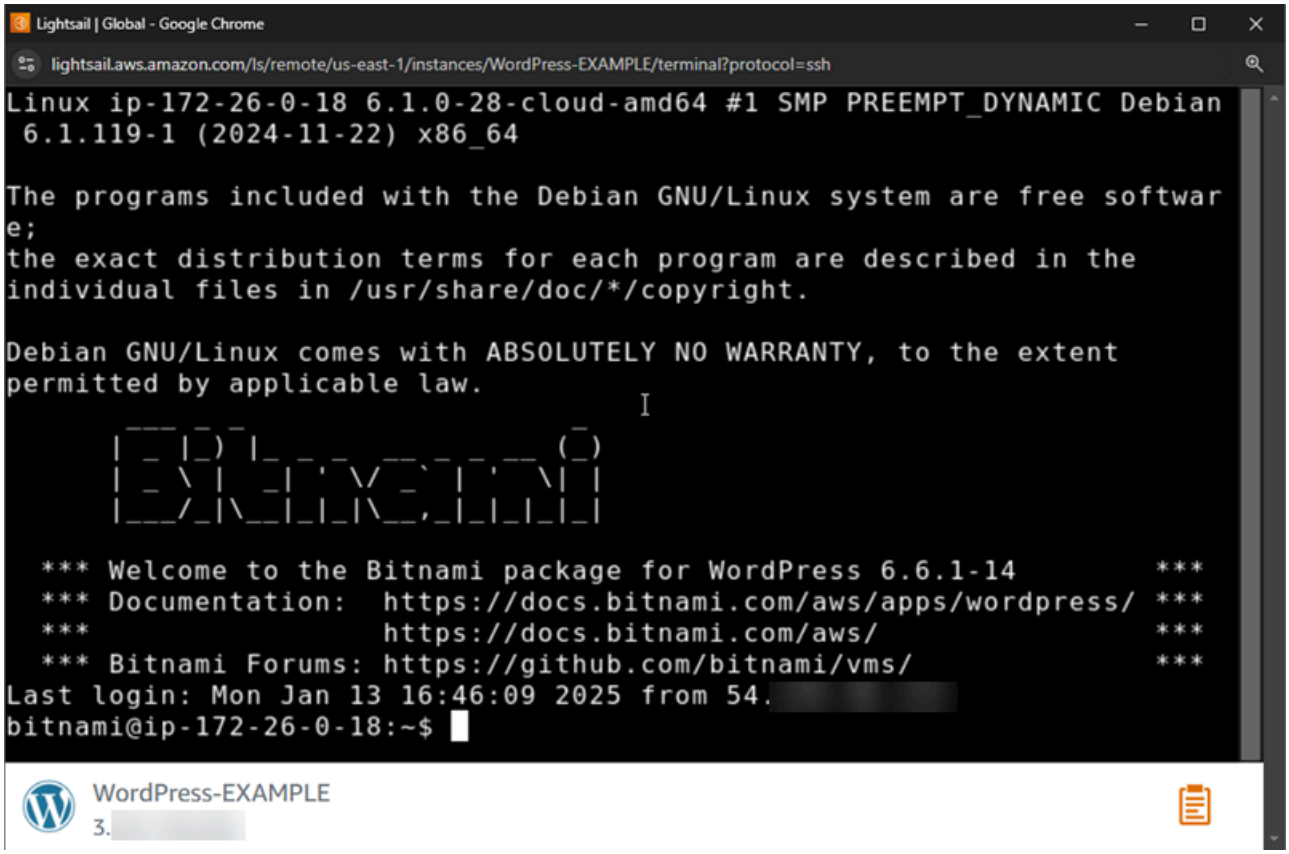
1. Lightsail 홈 페이지에서 작업 메뉴 아이콘(:)을 선택한 다음 연결을 선택합니다.



또는 인스턴스의 관리 페이지에서 연결할 수 있습니다. 인스턴스 이름을 선택하고 연결 탭을 선택한 다음 SSH를 사용하여 연결을 선택합니다.



2. 이제 터미널에 명령을 입력한 다음, SSH 클라이언트를 설치하지 않고도 Lightsail 인스턴스를 관리할 수 있습니다.



가상 컴퓨터에 추가 스토리지를 연결하는 방법에 대해 알아보려면 이 자습서의 다음 단계를 계속 진행합니다.

4단계: 인스턴스에 스토리지 추가

Lightsail은 블록 수준 스토리지 볼륨(디스크)을 제공하여 인스턴스에 연결하는 것이 가능합니다. 인스턴스가 시스템 디스크와 함께 제공되더라도 요구 사항의 변화에 따라 추가 스토리지 디스크를 연결할 수 있습니다. 한 인스턴스에서 디스크를 분리한 다음 다른 인스턴스에 연결하는 것도 가능합니다.

추가 디스크를 생성한 후 Lightsail 인스턴스에 연결하여 디스크를 포맷하고 탑재해야 합니다.

디스크 생성, 연결, 관리에 대한 자세한 내용은 Lightsail 블록 스토리지 디스크를 생성하고 Linux 인스턴스에 연결 섹션을 참조하세요.

가상 컴퓨터 백업에 대해 알아보려면 이 자습서의 다음 단계를 계속 진행합니다.

5단계: 스냅샷 생성

스냅샷은 데이터의 특정 시점 복사본입니다. 인스턴스의 스냅샷을 생성하고 이를 기준으로 사용하여 새 인스턴스를 생성하거나 데이터 백업을 수행할 수 있습니다. 스냅샷은 스냅샷을 생성한 시점부터 인스턴스를 복원하는 데 필요한 모든 데이터를 포함합니다.

스냅샷 생성 및 관리에 대한 자세한 내용은 스냅샷으로 Linux/Unix Lightsail 인스턴스 백업 섹션을 참조하세요.

가상 컴퓨터 리소스 정리에 대해 알아보려면 이 자습서의 다음 단계를 계속 진행합니다.

6단계: 정리

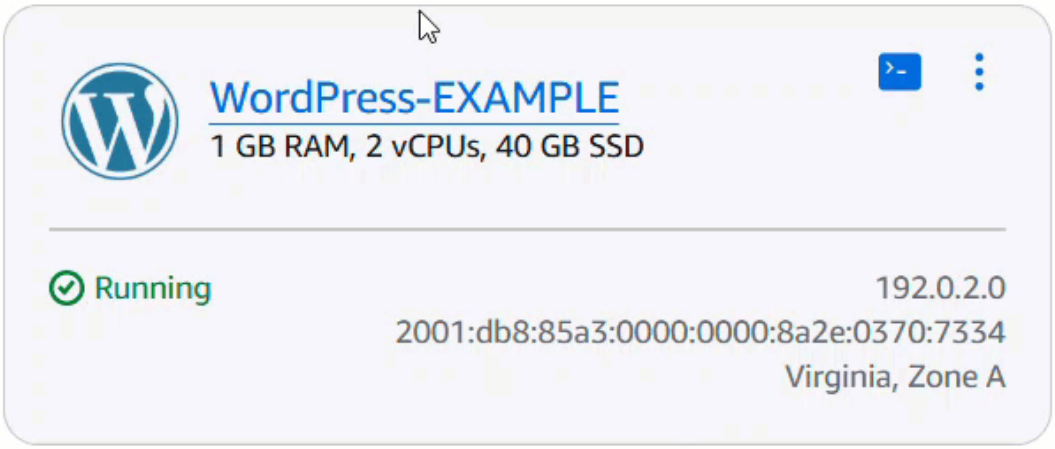
이 자습서용으로 위해 생성한 인스턴스 작업을 마친 후에는 인스턴스를 삭제할 수 있습니다. 이렇게 하면 필요하지 않은 인스턴스에 대한 요금이 더 이상 부과되지 않습니다.

인스턴스를 삭제해도 관련 스냅샷이나 연결된 디스크는 삭제되지 않습니다. 이 자습서용으로 스냅샷과 디스크를 생성한 경우 해당 항목도 삭제해야 합니다.

나중에 사용할 수 있도록 인스턴스를 저장하되 비용이 발생하지 않도록 하려면 인스턴스를 삭제하는 대신 중지하면 됩니다. 그런 다음 나중에 다시 시작할 수 있습니다. 요금에 대한 자세한 내용은 Lightsail 요금을 참조하세요.

> **Important**
>
> Lightsail 리소스 삭제는 영구적인 작업입니다. 삭제된 데이터는 복구할 수 없습니다. 나중에 데이터가 필요할 수 있는 경우 삭제하기 전에 가상 컴퓨터의 스냅샷을 생성합니다. 자세한 내용은 스냅샷으로 Linux/Unix Lightsail 인스턴스 백업 단원을 참조하십시오.

1. Lightsail 콘솔에 로그인합니다.
2. 탐색 창에서 인스턴스를 선택합니다.
3. 삭제할 인스턴스의 경우, 작업 메뉴 아이콘(⋮)을 선택한 다음 삭제를 선택합니다.



4. 예, 삭제를 선택하여 삭제를 확인합니다.

다음 단계

Amazon Lightsail Linux 및 Windows 기반 인스턴스를 시작하려면 다음 주제를 사용합니다.

- Lightsail의 앱을 사용하여 Linux/Unix 인스턴스 생성
- Lightsail에서 Windows Server 인스턴스 생성

와 Amazon Lightsail 함께 사용 AWS CLI

이 자습서에서는 AWS Command Line Interface (AWS CLI)를 사용하는 일반적인 Amazon Lightsail 작업을 안내합니다. 키 페어, 인스턴스, 스토리지 및 스냅샷을 포함한 Lightsail 리소스를 생성하고 관리하는 방법을 알아봅니다.

주제

- 사전 조건
- SSH 키 페어 생성
- 인스턴스 생성 및 관리
- 인스턴스에 연결합니다
- 인스턴스에 스토리지 추가
- 스냅샷 생성 및 사용
- 리소스 정리
- 다음 단계

사전 조건

이 튜토리얼을 시작하기 전에 다음 사항을 확인해야 합니다.

1. AWS CLI. 설치해야 하는 경우 AWS CLI 설치 안내서를 따르세요. 이 포함된를 사용할 AWS CloudShell 수도 있습니다 AWS CLI.
2. 적절한 자격 증명 AWS CLI 으로를 구성했습니다. 자격 증명을 아직 설정하지 않은 경우 `aws configure`를 실행합니다.
3. 명령줄 인터페이스 및 SSH 개념에 대한 기본 지식.
4. AWS 계정에서 Lightsail 리소스를 생성하고 관리할 수 있는 충분한 권한.

아직 설정하지 않은 경우 `AWS_REGION` 환경 변수를 시작하기 전에 사용하도록를 구성한 리전과 동일한 리전 AWS CLI 으로 설정합니다. 이 환경 변수는 예제 명령에서 Lightsail 리소스에 대한 가용 영역을 지정하는 데 사용됩니다.

```
$ [ -z "${AWS_REGION}" ] && export AWS_REGION=$(aws configure get region)
```

CLI를 사용하여 Amazon Lightsail 리소스의 생성 및 관리를 시작해 보겠습니다.

SSH 키 페어 생성

SSH 키 페어를 사용하면 암호를 사용하지 않고 Lightsail 인스턴스에 안전하게 연결할 수 있습니다. 이 섹션에서는 새 키 페어를 생성하고 해당 정보를 검색합니다.

Example– 새 키 페어 생성

다음 명령은 "cli-tutorial-keys"라는 이름의 새 SSH 키 페어를 생성하고 로컬 시스템에 프라이빗 키를 저장합니다.

```
$ aws lightsail create-key-pair --key-pair-name cli-tutorial-keys \
        --query privateKeyBase64 --output text > ~/.ssh/cli-tutorial-keys.pem
$ chmod 400 ~/.ssh/cli-tutorial-keys.pem
```

이 명령을 실행하면 적절한 권한이 있는 `~/.ssh` 디렉터리에 프라이빗 키가 저장됩니다. `chmod` 명령은 SSH의 보안 요구 사항으로 사용자만 프라이빗 키 파일을 읽을 수 있도록 합니다.

Example- 키 페어 정보 검색

정보를 검색하여 키 페어가 성공적으로 생성되었는지 확인할 수 있습니다.

```
$ aws lightsail get-key-pair --key-pair-name cli-tutorial-keys
{
    "keyPair": {
        "name": "cli-tutorial-keys",
        "arn": "arn:aws:lightsail:us-east-2:123456789012:KeyPair/e00xmpl-6a6a-434a-
bff1-87f2bb815e21",
        "supportCode": "123456789012/cli-tutorial-keys",
        "createdAt": 1673596800.000,
        "location": {
            "availabilityZone": "all",
            "regionName": "us-east-2"
        },
        "resourceType": "KeyPair",
        "tags": [],
        "fingerprint": "d0:0d:30:db:5a:24:df:f6:17:f0:e2:15:45:77:3d:bb:d0:6d:fc:81"
    }
}
```

출력에는 이름, ARN, 생성 시간, 리전, 지문을 포함하여 키 페어에 대한 세부 정보가 표시됩니다. 이 지문을 사용하여 인스턴스에 연결할 때 키의 신뢰성을 확인할 수 있습니다.

인스턴스 생성 및 관리

Lightsail 인스턴스는 애플리케이션 또는 웹 사이트를 실행하는 가상 프라이빗 서버입니다. 이 섹션에서는 WordPress 인스턴스를 생성하고 해당 세부 정보를 검색합니다.

Example- WordPress 인스턴스 생성

다음 명령은 nano_3_0 번들(가장 작은 Lightsail 인스턴스 크기)을 사용하여 새 WordPress 인스턴스를 생성하고 이를 키 페어와 연결합니다. 명령은 AWS_REGION 환경 변수를 사용하여 구성된 리전의 가용 영역에 인스턴스를 생성합니다.

```
$ aws lightsail create-instances --instance-names cli-tutorial \
        --availability-zone ${AWS_REGION}a --blueprint-id wordpress \
        --bundle-id nano_3_0 --key-pair-name cli-tutorial-keys
{
    "operations": [
        {
            "id": "f30xmpl-3727-492a-9d42-5c94ad3ef9a8",
            "resourceName": "cli-tutorial",
            "resourceType": "Instance",
            "createdAt": 1673596800.000,
```

```
            "location": {
                "availabilityZone": "us-east-2a",
                "regionName": "us-east-2"
            },
            "isTerminal": false,
            "operationType": "CreateInstance",
            "status": "Started",
            "statusChangedAt": 1673596800.000
        }
    ]
}
```

응답은 인스턴스 생성 작업이 시작되었음을 나타냅니다. 인스턴스가 사용 가능한 상태가 되는 데 몇 분 정도 걸릴 수 있습니다.

Example- 인스턴스 세부 정보 가져오기

인스턴스가 생성되면 다음 명령을 사용하여 해당 세부 정보를 검색할 수 있습니다.

```
$ aws lightsail get-instance --instance-name cli-tutorial
{
    "instance": {
        "name": "cli-tutorial",
        "arn": "arn:aws:lightsail:us-east-2:123456789012:Instance/7d3xmpl-ae2e-44d5-
bbd9-22f9ec2abe1f",
        "supportCode": "123456789012/i-099cxmpl5dad5923c",
        "createdAt": 1673596800.000,
        "location": {
            "availabilityZone": "us-east-2a",
            "regionName": "us-east-2"
        },
        "resourceType": "Instance",
        "tags": [],
        "blueprintId": "wordpress",
        "blueprintName": "WordPress",
        "bundleId": "nano_3_0",
        "isStaticIp": false,
        "privateIpAddress": "172.26.6.136",
        "publicIpAddress": "192.0.2.1",
        "ipv6Addresses": [
            "2001:db8:85a3:0000:0000:8a2e:0370:7334"
        ],
        "ipAddressType": "dualstack",
        "hardware": {
```

```
            "cpuCount": 2,
            "disks": [
                {
                    "createdAt": 1673596800.000,
                    "sizeInGb": 20,
                    "isSystemDisk": true,
                    "iops": 100,
                    "path": "/dev/xvda",
                    "attachedTo": "cli-tutorial",
                    "attachmentState": "attached"
                }
            ],
            "ramSizeInGb": 0.5
        },
        "networking": {
            "monthlyTransfer": {
                "gbPerMonthAllocated": 1024
            },
            "ports": [
                {
                    "fromPort": 80,
                    "toPort": 80,
                    "protocol": "tcp",
                    "accessFrom": "Anywhere (0.0.0.0/0 and ::/0)",
                    "accessType": "public",
                    "commonName": "",
                    "accessDirection": "inbound",
                    "cidrs": [
                        "0.0.0.0/0"
                    ],
                    "ipv6Cidrs": [
                        "::/0"
                    ],
                    "cidrListAliases": []
                },
                {
                    "fromPort": 22,
                    "toPort": 22,
                    "protocol": "tcp",
                    "accessFrom": "Anywhere (0.0.0.0/0 and ::/0)",
                    "accessType": "public",
                    "commonName": "",
                    "accessDirection": "inbound",
                    "cidrs": [
```

```
                        "0.0.0.0/0"
                    ],
                    "ipv6Cidrs": [
                        "::/0"
                    ],
                    "cidrListAliases": []
                },
                {
                    "fromPort": 443,
                    "toPort": 443,
                    "protocol": "tcp",
                    "accessFrom": "Anywhere (0.0.0.0/0 and ::/0)",
                    "accessType": "public",
                    "commonName": "",
                    "accessDirection": "inbound",
                    "cidrs": [
                        "0.0.0.0/0"
                    ],
                    "ipv6Cidrs": [
                        "::/0"
                    ],
                    "cidrListAliases": []
                }
            ]
        },
        "state": {
            "code": 16,
            "name": "running"
        },
        "username": "bitnami",
        "sshKeyName": "cli-tutorial-keys",
        "metadataOptions": {
            "state": "applied",
            "httpTokens": "optional",
            "httpEndpoint": "enabled",
            "httpPutResponseHopLimit": 1,
            "httpProtocolIpv6": "disabled"
        }
    }
}
```

출력은 IP 주소, 하드웨어 사양, 네트워킹 구성 및 상태를 포함하여 인스턴스에 대한 포괄적인 정보를 제공합니다. 인스턴스에 연결하는 데 필요하므로 퍼블릭 IP 주소와 사용자 이름을 기록해 둡니다.

인스턴스에 연결합니다

인스턴스를 생성한 후 이전에 생성한 키 페어와 함께 SSH를 사용하여 인스턴스에 연결할 수 있습니다. 이 섹션에서는 SSH 연결을 설정하고 보안 설정을 관리하는 방법을 보여줍니다.

Example- 인스턴스에 대한 SSH

다음 명령을 사용하여 SSH를 통해 인스턴스에 연결하여 IP 주소를 인스턴스의 퍼블릭 IP로 바꿉니다.

```
$ ssh -i ~/.ssh/cli-tutorial-keys.pem bitnami@192.0.2.1
Linux ip-172-26-6-136 6.1.0-32-cloud-amd64 #1 SMP PREEMPT_DYNAMIC Debian 6.1.129-1
 (2025-03-06) x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
       ___ _ _                   _
      | _ |_) |_ _ _  __ _ _ __ (_)
      | _ \ |  _| ' \/ _` | '  \| |
      |___/_|\__|_|_|\__,_|_|_|_|_|

  *** Welcome to the Bitnami package for WordPress 6.7.2           ***
  *** Documentation:  https://docs.bitnami.com/aws/apps/wordpress/ ***
  ***                 https://docs.bitnami.com/aws/                ***
  *** Bitnami Forums: https://github.com/bitnami/vms/              ***

bitnami@ip-172-26-6-136:~$ df
Filesystem      1K-blocks    Used Available Use% Mounted on
udev               217920       0    217920   0% /dev
tmpfs               45860     480     45380   2% /run
/dev/nvme0n1p1   20403592 3328832  16142256  18% /
tmpfs              229292       0    229292   0% /dev/shm
tmpfs                5120       0      5120   0% /run/lock
/dev/nvme0n1p15    126678   11840    114838  10% /boot/efi
tmpfs               45856       0     45856   0% /run/user/1000
```

연결되면 WordPress 설치를 관리하거나, 서버를 구성하거나, 추가 소프트웨어를 설치할 수 있습니다. 위 예제는 df 명령을 사용하여 인스턴스의 디스크 사용량을 보여줍니다.

Example- 퍼블릭 포트 닫기

SSH를 사용하지 않을 때는 인스턴스의 퍼블릭 포트를 닫을 수 있습니다. 이렇게 하면 인스턴스를 무단 액세스 시도로부터 보호할 수 있습니다.

```
$ aws lightsail close-instance-public-ports --instance-name cli-tutorial \
        --port-info fromPort=22,protocol=TCP,toPort=22
{
    "operation": {
        "id": "6cdxmpl-9f39-4357-a66d-230096140b4f",
        "resourceName": "cli-tutorial",
        "resourceType": "Instance",
        "createdAt": 1673596800.000,
        "location": {
            "availabilityZone": "us-east-2a",
            "regionName": "us-east-2"
        },
        "isTerminal": true,
        "operationDetails": "22/tcp",
        "operationType": "CloseInstancePublicPorts",
        "status": "Succeeded",
        "statusChangedAt": 1673596800.000
    }
}
```

> **Note**
>
> 포트 22를 닫으면 Lightsail 콘솔에서 시작된 연결을 포함하여 모든 SSH 연결이 차단됩니다. 자세한 내용은 다음 항목을 참조하세요.
>
> - SSH 키 페어 관리 및 Lightsail 인스턴스 연결
> - Lightsail의 방화벽을 사용하여 인스턴스 트래픽 제어

응답은 포트 22가 성공적으로 닫혔음을 확인합니다. SSH를 통해 다시 연결해야 할 때는 `open-instance-public-ports` 명령을 사용하여 포트를 다시 열 수 있습니다.

인스턴스에 스토리지 추가

애플리케이션이 확장됨에 따라 추가 스토리지 공간이 필요할 수 있습니다. Lightsail을 사용하면 추가 디스크를 생성하여 인스턴스에 연결할 수 있습니다. 이 섹션에서는 스토리지를 추가하는 방법을 보여 줍니다.

Example- 디스크 생성

다음 명령은 새로운 32GB 디스크를 생성합니다.

```
$ aws lightsail create-disk --disk-name cli-tutorial-disk \
        --availability-zone ${AWS_REGION}a --size-in-gb 32
{
    "operations": [
        {
            "id": "070xmpl-3364-4aa2-bff2-3c589de832fc",
            "resourceName": "cli-tutorial-disk",
            "resourceType": "Disk",
            "createdAt": 1673596800.000,
            "location": {
                "availabilityZone": "us-east-2a",
                "regionName": "us-east-2"
            },
            "isTerminal": false,
            "operationType": "CreateDisk",
            "status": "Started",
            "statusChangedAt": 1673596800.000
        }
    ]
}
```

응답은 디스크 생성 작업이 시작되었음을 나타냅니다. 디스크를 사용할 수 있게 되는 데 몇 분 정도 걸릴 수 있습니다.

Example- 인스턴스에 디스크 연결

디스크가 생성되면 다음 명령을 사용하여 인스턴스에 연결할 수 있습니다.

```
$ aws lightsail attach-disk --disk-name cli-tutorial-disk \
        --disk-path /dev/xvdf --instance-name cli-tutorial
{
    "operations": [
```

```
        {
            "id": "d17xmpl-2bdb-4292-ac63-ba5537522cea",
            "resourceName": "cli-tutorial-disk",
            "resourceType": "Disk",
            "createdAt": 1673596800.000,
            "location": {
                "availabilityZone": "us-east-2a",
                "regionName": "us-east-2"
            },
            "isTerminal": false,
            "operationDetails": "cli-tutorial",
            "operationType": "AttachDisk",
            "status": "Started",
            "statusChangedAt": 1673596800.000
        },
        {
            "id": "01exmpl-c04e-42d4-aa6b-45ce50562a54",
            "resourceName": "cli-tutorial",
            "resourceType": "Instance",
            "createdAt": 1673596800.000,
            "location": {
                "availabilityZone": "us-east-2a",
                "regionName": "us-east-2"
            },
            "isTerminal": false,
            "operationDetails": "cli-tutorial-disk",
            "operationType": "AttachDisk",
            "status": "Started",
            "statusChangedAt": 1673596800.000
        }
    ]
}
```

disk-path 파라미터는 Linux 파일 시스템에서 디스크를 연결할 위치를 지정합니다. 디스크를 연결한 후에는 인스턴스 내에서 디스크를 포맷하고 탑재해야 합니다.

Example- 디스크 연결 확인

세부 정보를 검색하여 디스크가 제대로 연결되었는지 확인할 수 있습니다.

```
$ aws lightsail get-disk --disk-name cli-tutorial-disk
{
    "disk": {
```

```
        "name": "cli-tutorial-disk",
        "arn": "arn:aws:lightsail:us-east-2:123456789012:Disk/1a9xmpl-8a34-46a4-
b87e-19184f0cca9c",
        "supportCode": "123456789012/vol-0dacxmplc1c3108e2",
        "createdAt": 1673596800.000,
        "location": {
            "availabilityZone": "us-east-2a",
            "regionName": "us-east-2"
        },
        "resourceType": "Disk",
        "tags": [],
        "sizeInGb": 32,
        "isSystemDisk": false,
        "iops": 100,
        "path": "/dev/xvdf",
        "state": "in-use",
        "attachedTo": "cli-tutorial",
        "isAttached": true,
        "attachmentState": "attached"
    }
}
```

출력은 디스크가 인스턴스에 연결되어 있음을 확인합니다. 'state' 필드에 'in-use'가 표시되고 'isAttached'가 true로 설정되어 성공적인 연결을 나타냅니다.

스냅샷 생성 및 사용

스냅샷은 인스턴스를 백업하고 백업에서 새 인스턴스를 생성하는 방법을 제공합니다. 이는 재해 복구, 테스트 또는 중복 환경 생성에 유용합니다.

Example- 인스턴스 스냅샷 생성

다음 명령은 인스턴스의 스냅샷을 생성합니다.

```
$ aws lightsail create-instance-snapshot --instance-name cli-tutorial \
         --instance-snapshot-name cli-tutorial-snapshot
{
    "operations": [
        {
            "id": "41bxmpl-7824-4591-bfcc-1b1c341613a4",
            "resourceName": "cli-tutorial-snapshot",
            "resourceType": "InstanceSnapshot",
            "createdAt": 1673596800.000,
```

```
            "location": {
                "availabilityZone": "all",
                "regionName": "us-east-2"
            },
            "isTerminal": false,
            "operationDetails": "cli-tutorial",
            "operationType": "CreateInstanceSnapshot",
            "status": "Started",
            "statusChangedAt": 1673596800.000
        },
        {
            "id": "725xmpl-158e-46f6-bd49-27b0e6805aa2",
            "resourceName": "cli-tutorial",
            "resourceType": "Instance",
            "createdAt": 1673596800.000,
            "location": {
                "availabilityZone": "us-east-2a",
                "regionName": "us-east-2"
            },
            "isTerminal": false,
            "operationDetails": "cli-tutorial-snapshot",
            "operationType": "CreateInstanceSnapshot",
            "status": "Started",
            "statusChangedAt": 1673596800.000
        }
    ]
}
```

응답은 스냅샷 프로세스가 시작되었음을 나타냅니다. 스냅샷을 가져오는 인스턴스에 대한 하나의 비동기 작업이 있고, 생성 중인 스냅샷에 대한 하나의 작업이 있습니다. 스냅샷은 인스턴스에 연결된 모든 디스크를 포함합니다.

Example- 스냅샷에서 새 인스턴스 생성

스냅샷이 완료되면 이를 사용하여 새 인스턴스를 생성할 수 있습니다.

```
$ aws lightsail create-instances-from-snapshot --availability-zone ${AWS_REGION}b \
        --instance-snapshot-name cli-tutorial-snapshot --instance-name cli-tutorial-bup
 --bundle-id small_3_0
{
    "operations": [
        {
            "id": "a35xmpl-efa1-4d6c-958e-9d58fd258f5f",
```

```
            "resourceName": "cli-tutorial-bup",
            "resourceType": "Instance",
            "createdAt": 1673596800.000,
            "location": {
                "availabilityZone": "us-east-2b",
                "regionName": "us-east-2"
            },
            "isTerminal": false,
            "operationType": "CreateInstancesFromSnapshot",
            "status": "Started",
            "statusChangedAt": 1673596800.000
        }
    ]
}
```

이 명령은 small_3_0 번들 크기를 사용하여 가용 영역 cli-tutorial-bup에 us-east-2b라는 이름의 새 인스턴스를 생성합니다. 새 인스턴스에 대해 다른 번들 크기를 선택할 수 있으며, 이는 스케일업 또는 스케일 다운에 유용할 수 있습니다.

리소스 정리

Lightsail 리소스 사용을 마쳤으면 추가 요금이 발생하지 않도록 리소스를 삭제해야 합니다. 이 섹션은 이 자습서에서 생성한 리소스를 정리하는 방법을 보여줍니다.

Example- 인스턴스 스냅샷 삭제

더 이상 필요하지 않은 스냅샷을 삭제하려면 다음 명령을 사용합니다.

```
$ aws lightsail delete-instance-snapshot --instance-snapshot-name cli-tutorial-snapshot
{
    "operations": [
        {
            "id": "cf8xmpl-0ec7-43ec-9cbc-6dedd9d8eda8",
            "resourceName": "cli-tutorial-snapshot",
            "resourceType": "InstanceSnapshot",
            "createdAt": 1673596800.000,
            "location": {
                "availabilityZone": "all",
                "regionName": "us-east-2"
            },
            "isTerminal": true,
            "operationDetails": "",
```

```
            "operationType": "DeleteInstanceSnapshot",
            "status": "Succeeded",
            "statusChangedAt": 1673596800.000
        }
    ]
}
```

응답은 스냅샷 삭제 작업이 성공했음을 확인합니다.

Example- 인스턴스 삭제

인스턴스를 삭제하려면 다음 명령을 사용합니다.

```
$ aws lightsail delete-instance --instance-name cli-tutorial
{
    "operations": [
        {
            "id": "f4bxmpl-2df1-4740-90d7-e30adaf7e3a1",
            "resourceName": "cli-tutorial",
            "resourceType": "Instance",
            "createdAt": 1673596800.000,
            "location": {
                "availabilityZone": "us-east-2a",
                "regionName": "us-east-2"
            },
            "isTerminal": true,
            "operationDetails": "",
            "operationType": "DeleteInstance",
            "status": "Succeeded",
            "statusChangedAt": 1673596800.000
        }
    ]
}
```

스냅샷에서 생성된 인스턴스를 포함하여 생성한 모든 인스턴스를 삭제해야 합니다.

Example- 디스크 삭제

더 이상 필요하지 않은 디스크를 삭제하려면 다음 명령을 사용합니다.

```
$ aws lightsail delete-disk --disk-name cli-tutorial-disk
{
    "operations": [
```

```
        {
            "id": "aacxmpl-8626-4edd-8b3b-bf108d6b279c",
            "resourceName": "cli-tutorial-disk",
            "resourceType": "Disk",
            "createdAt": 1673596800.000,
            "location": {
                "availabilityZone": "us-east-2a",
                "regionName": "us-east-2"
            },
            "isTerminal": true,
            "operationDetails": "",
            "operationType": "DeleteDisk",
            "status": "Succeeded",
            "statusChangedAt": 1673596800.000
        }
    ]
}
```

디스크가 인스턴스에 연결되어 있는 경우 먼저 `detach-disk` 명령을 사용하여 디스크를 분리해야 합니다.

Example- 키 페어 삭제

마지막으로 이 자습서의 시작 부분에서 생성한 키 페어를 삭제합니다.

```
$ aws lightsail delete-key-pair --key-pair-name cli-tutorial-keys
{
    "operation": {
        "id": "dbfxmpl-c954-4a45-93a4-ab3e627d2c23",
        "resourceName": "cli-tutorial-keys",
        "resourceType": "KeyPair",
        "createdAt": 1673596800.000,
        "location": {
            "availabilityZone": "all",
            "regionName": "us-east-2"
        },
        "isTerminal": true,
        "operationDetails": "",
        "operationType": "DeleteKeyPair",
        "status": "Succeeded",
        "statusChangedAt": 1673596800.000
    }
}
```

이 명령은 키 페어만 삭제합니다 AWS. 이제 로컬 복사본도 삭제할 수 있습니다.

```
$ rm ~/.ssh/cli-tutorial-keys.pem
```

다음 단계

이제를 사용하여 Lightsail 리소스를 관리하는 기본 사항을 배웠으므로 다른 Lightsail 기능을 AWS CLI 살펴봅니다.

1. 도메인 - 애플리케이션에 도메인 이름을 할당합니다.
2. 로드 밸런서 - 트래픽을 여러 인스턴스로 라우팅하여 용량과 복원력을 높입니다.
3. 자동 스냅샷 - 애플리케이션 데이터를 자동으로 백업합니다.
4. 지표 - 리소스 상태를 모니터링하고, 알림을 받고, 경보를 설정합니다.
5. 데이터베이스 - 애플리케이션을 관계형 데이터베이스에 연결합니다.

사용 가능한 AWS CLI 명령에 대한 자세한 내용은 AWS CLI 에 대한 명령 참조Lightsail를 참조하세요.

Lightsail 리셀러

Amazon Lightsail 리셀러로 등록되면 자체 고객에게 Lightsail 제품을 제공할 수 있습니다. Lightsail 리셀러가 되면 Lightsail 인스턴스의 기본 할당량이 높아질 뿐만 아니라 콘솔 내의 등록 리셀러 전용 피드백 양식을 사용할 수 있습니다.

주제

- Lightsail 재판매의 이점
- Lightsail 리셀러 혜택 및 증가된 기본 할당량이 계정에 적용되는 방식
- Lightsail 리셀러가 되는 방법
- Lightsail 리셀러 되기
- 리셀러 계정에 대한 서비스 할당량 증가 요청
- 리셀러로서 Lightsail에 문의하기

Lightsail 재판매의 이점

Lightsail 리셀러가 되면 Lightsail 리소스에 대해 규모 조정, 예산 책정 및 지원과 관련된 다양한 혜택을 얻을 수 있습니다.

Lightsail 비즈니스 확장

리셀러가 되면 Lightsail의 글로벌 클라우드 인프라에서 비즈니스를 더 빠르게 확장할 수 있습니다. 리셀러 혜택을 사용하면 AWS 리전 기본적으로 각의 등록된 계정에서 Lightsail 인스턴스에 대한 서비스 할당량이 높아집니다.

예산 책정 간소화

Lightsail에는 메모리, vCPU 및 솔리드 스테이트 드라이브(SSD) 스토리지가 번들 플랜으로 제공되는 예측 가능한 요금 모델이 있습니다. 이 모델을 사용하면 대규모로 Lightsail 리소스 비즈니스를 성장시키고 관리할 때 비용을 쉽게 예측할 수 있습니다.

신뢰성

데이터의 자동 스냅샷, 구성된 임계값을 위반하는 리소스에 대한 알림이 포함된 경보, IPv6 네트워킹 지원과 같은 기능을 통해 더 높은 리소스 효율성 및 신뢰성으로 운영하세요.

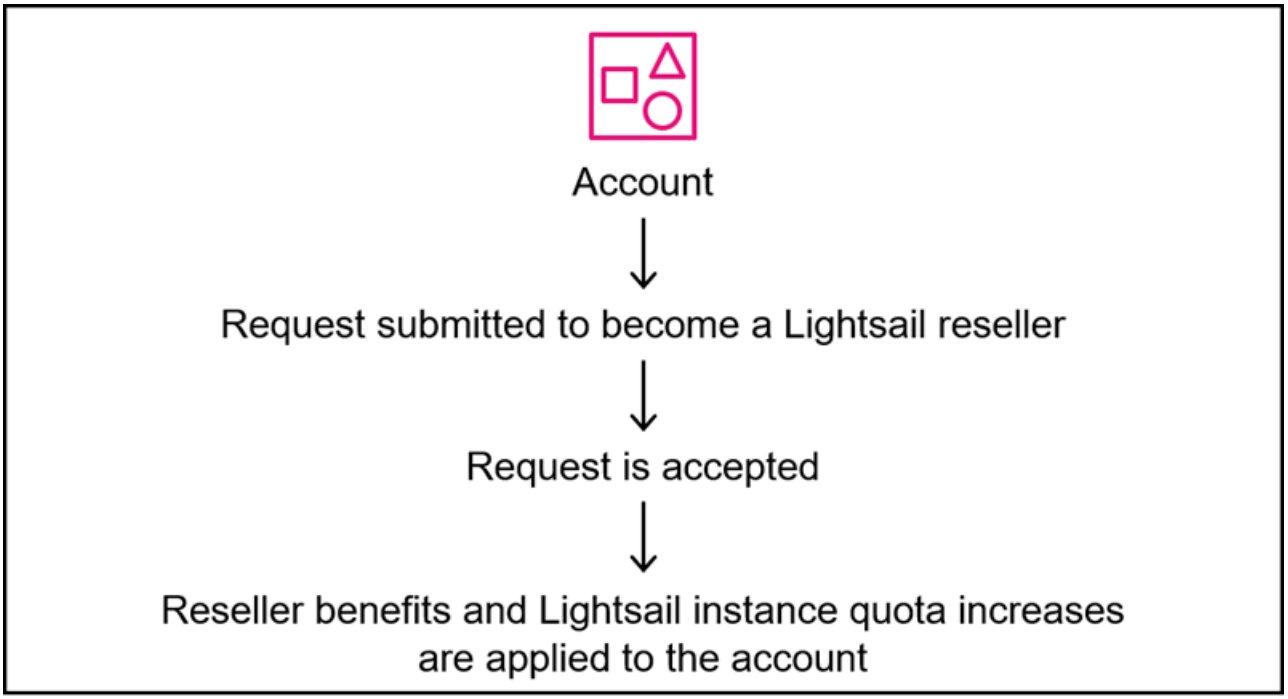
Lightsail 리셀러 혜택 및 증가된 기본 할당량이 계정에 적용되는 방식

리셀러 혜택은 요청을 제출하는 AWS 계정 에 적용됩니다. 요청이 승인되면 기본 Lightsail 인스턴스 할당량을 늘리도록 추가 AWS 계정 를 요청할 수 있습니다. 를 사용하면 AWS Organizations리셀러 혜택과 늘어난 기본 Lightsail 인스턴스 할당량이 관리 계정에 적용됩니다. 멤버 계정의 경우 Lightsail 인스턴스에 대한 기본 할당량이 증가합니다. 조직에 대한 자세한 내용은 AWS Organizations 사용 설명서의 What is AWS Organizations?를 참조하세요.

다음 다이어그램은 Lightsail 리셀러의 이점과 늘어난 기본 Lightsail 인스턴스 할당량이 적용되는 방법을 보여줍니다 AWS 계정.

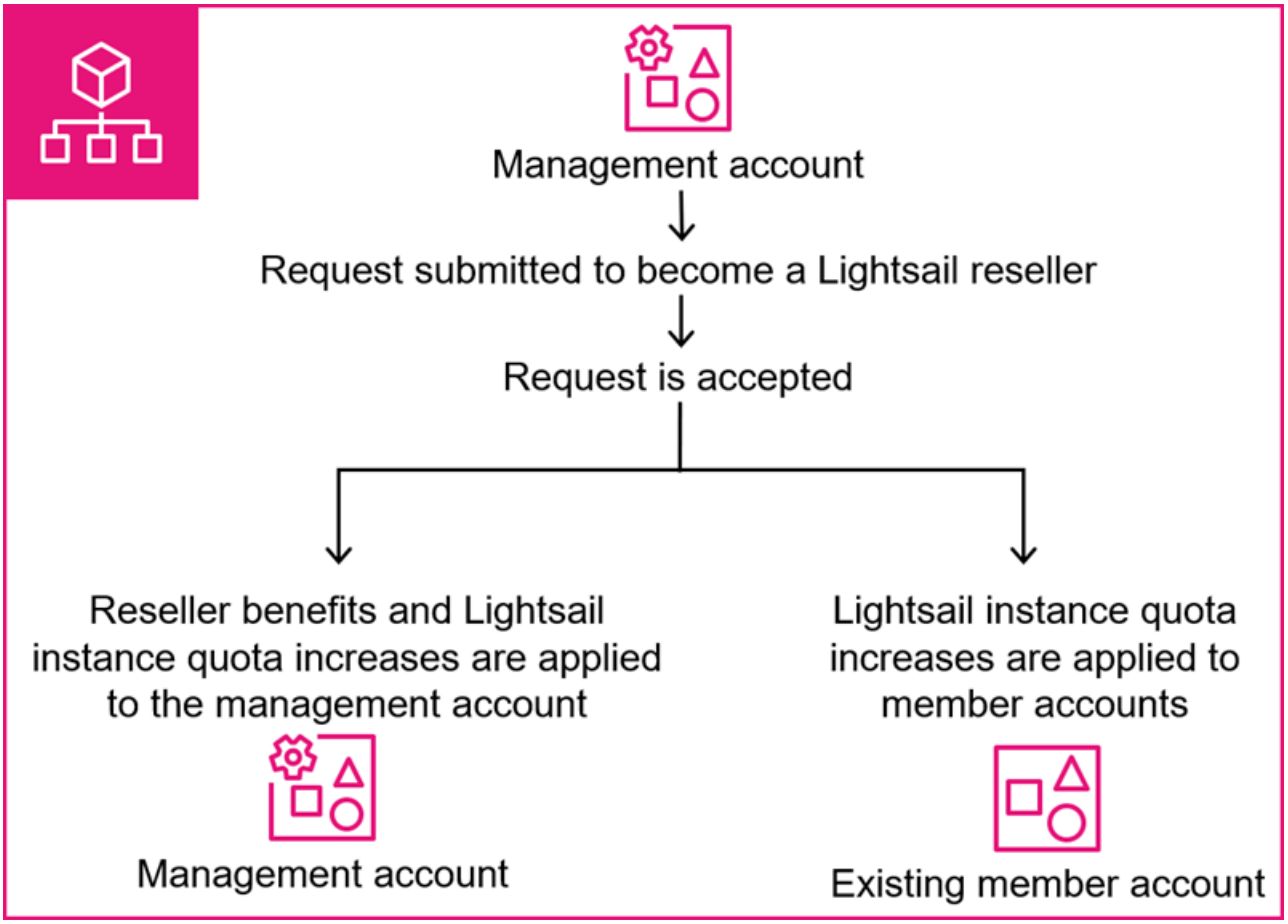
단일 AWS 계정

다음 다이어그램은 외부의 단일 계정이 Lightsail 리셀러 AWS Organizations 가 될 때 발생하는 상황을 자세히 설명합니다.



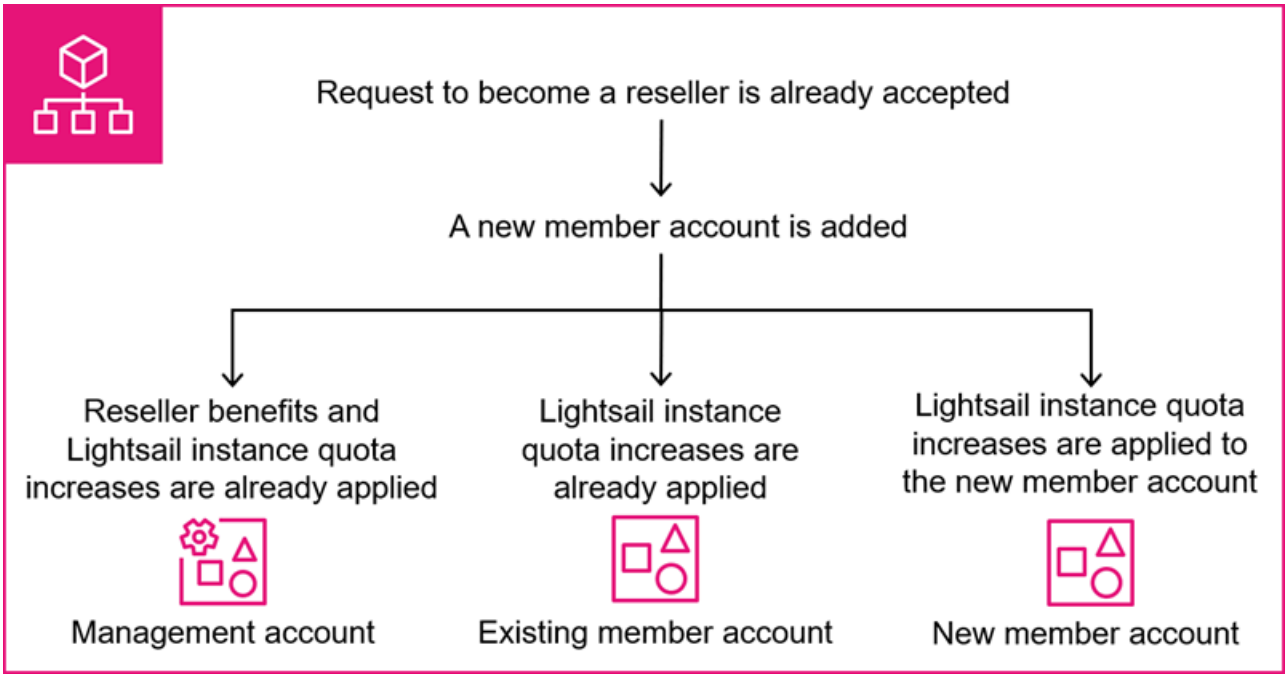
AWS 계정 Organizations의

다음 다이어그램은 AWS Organizations 의 관리 계정이 Lightsail 리셀러가 될 때 발생하는 상황을 자세히 설명합니다.



AWS 계정 리셀러가 된 후 추가된 조직의

다음 다이어그램은 관리 계정이 이미 Lightsail 리셀러로 등록된 조직에 새 멤버 계정이 추가될 때 발생하는 상황을 자세히 설명합니다.



Lightsail 리셀러가 되는 방법

계속하려면 Lightsail 리셀러가 되어야 하는 비즈니스 요구 사항에 대한 세부 정보가 포함된 양식을 제출해야 합니다. 자세한 내용은 Lightsail 리셀러 되기 단원을 참조하십시오.

Lightsail 리셀러 되기

Amazon Lightsail 리셀러 심사를 받으려면 양식을 제출해야 합니다. 요청은 양식을 작성할 때 로 로그인 AWS 계정 한를 사용하여 제출됩니다. AWS Organizations 를 사용하여를 중앙에서 관리하는 경우 관리 계정을 사용하여 리셀러가 되는 동안 요청을 제출해야 AWS 계정합니다. 관리 계정을 사용하면 조직의 멤버 계정 전체에 대해 기본 Lightsail 인스턴스 할당량이 증가합니다. Lightsail 리셀러 혜택이에 미치는 영향에 대한 자세한 내용은 섹션을 AWS 계정참조하세요Lightsail 리셀러 혜택 및 증가된 기본 할당량이 계정에 적용되는 방식.

요청이 승인되고 여러 조직이 있는 경우 각 조직의 관리 계정 AWS 계정 ID를 추가하여 늘어난 기본 Lightsail 인스턴스 할당량을 해당 조직의 멤버 계정으로 확장하는 추가 요청을 제출할 수 있습니다. 조직에 대한 자세한 내용은 AWS Organizations 사용 설명서의 What is AWS Organizations?를 참조하세요.

주제

- Lightsail 리셀러가 되는 데 필요한 정보

- Lightsail 리셀러 자격 신청
- 추가 계정의 Lightsail 리셀러 신청

Lightsail 리셀러가 되는 데 필요한 정보

Amazon Lightsail 리셀러 자격 신청을 심사하려면 예상 사용량 및 사용 사례에 대한 몇 가지 정보가 필요합니다. 작성하여 제출할 신청 양식은 Lightsail 콘솔에서 찾을 수 있습니다. 양식을 작성하려면 비즈니스에 대한 세부 정보 외에도 다음과 같은 정보가 있어야 합니다.

- 사용하려는 Lightsail 리소스에 대한 인스턴스 번들의 크기 및 수량. 사용 가능한 번들에 대한 자세한 내용은 Amazon Lightsail 요금을 참조하세요.
- 등록하려는AWS 계정 IDs. 를 사용하는 경우 요청에서 관리 계정만 지정해야 AWS Organizations합니다. 이렇게 하면 조직의 각 멤버 계정도 등록됩니다. 자세한 내용은 AWS Organizations 사용 설명서의 AWS Organizations용어 및 개념을 참조하세요.

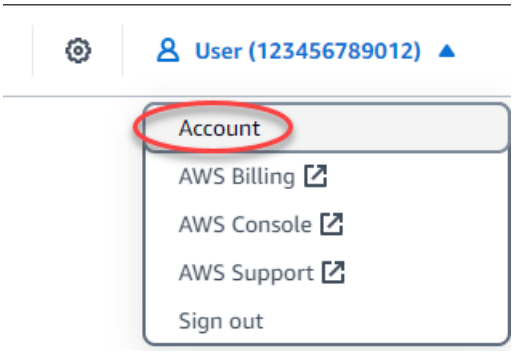
Lightsail 리셀러 자격 신청

다음 단계에서는 리셀러 자격 신청을 제출합니다. 인증된 AWS 계정 ID는 리셀러 혜택을 받고자 하는 계정으로 사용됩니다. 신청이 승인된 다음에는 추가 계정을 추가 신청할 수 있습니다.

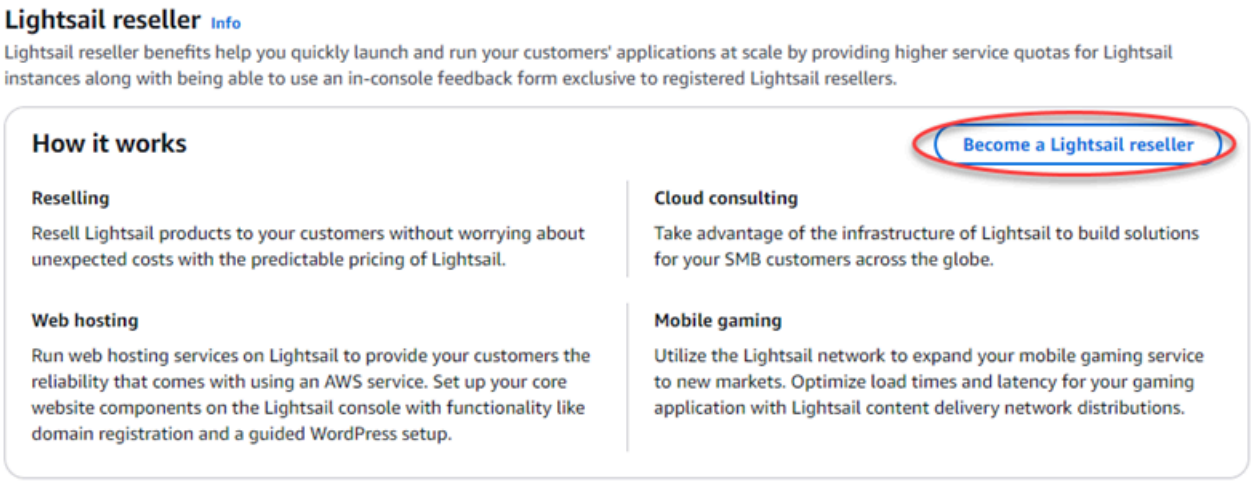
> **Tip**
>
> 를 사용하는 경우 멤버 계정에 기본 Lightsail 인스턴스 할당량 증가도 적용되도록 조직의 관리 계정으로이 절차를 수행해야 AWS Organizations합니다.

리셀러 자격을 신청하려면

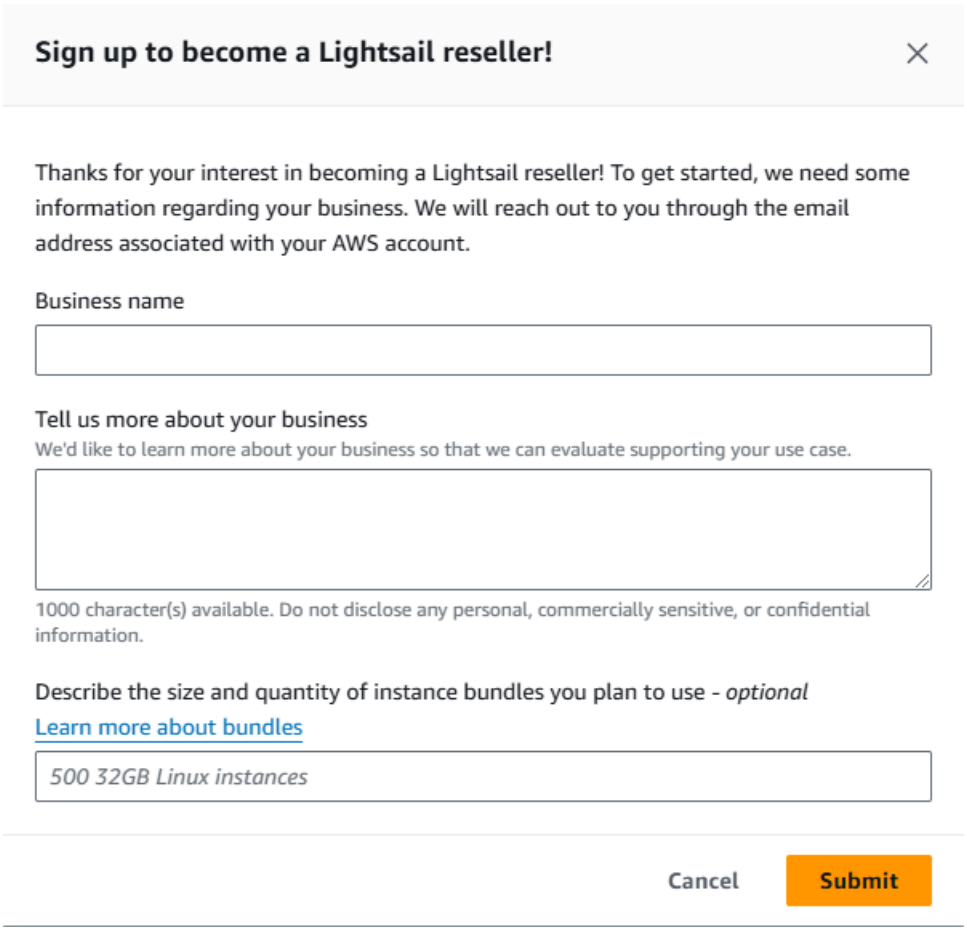
1. Lightsail 콘솔에 로그인합니다.
2. Lightsail 홈 페이지의 상단 탐색 메뉴에서 계정을 선택합니다.
3. 드롭다운 메뉴에서 계정(Account)을 선택합니다.



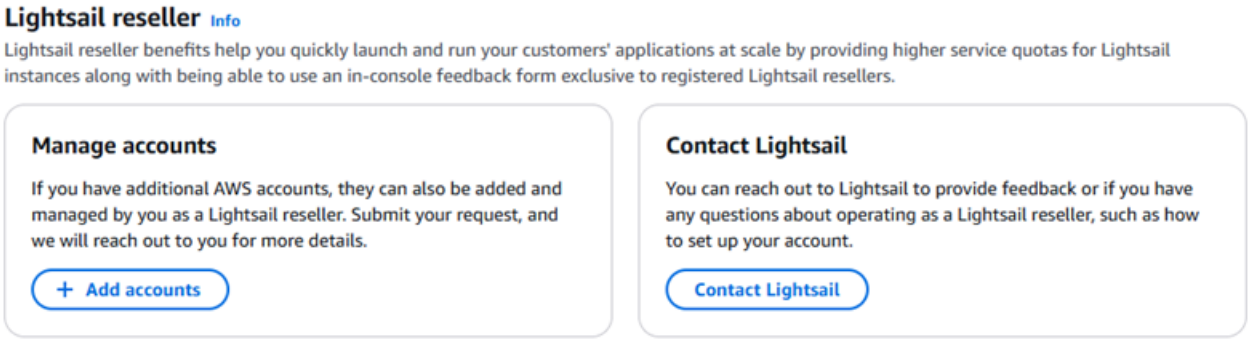
4. 프로필 탭의 Lightsail 리셀러 섹션에서 Lightsail 리셀러 되기를 선택합니다.



5. 등록 양식에서 필드에 정보를 입력하고 제출을 선택합니다.



리셀러 자격 획득에 대한 의도와 관련한 내용이 제출되었음을 확인하는 확인 메시지가 계정 이메일로 수신됩니다. 신청이 승인되면 Lightsail 콘솔의 계정 페이지에 Lightsail 리셀러로서 리셀러 계정을 관리하고 피드백 또는 쿼리를 위해 Lightsail 팀에 연락할 수 있는 옵션이 포함된 개정된 Lightsail 리셀러 섹션이 표시됩니다. 이 섹션은 Lightsail 리셀러 자격 신청을 제출한 계정에만 표시됩니다. 또한 Lightsail 인스턴스에 대해 더 높은 서비스 할당량을 받게 되며 추가 AWS 계정 에 대한 Lightsail 리셀러 신청을 할 수 있습니다.



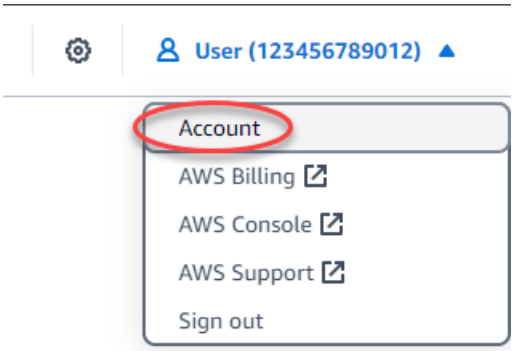
추가 계정의 Lightsail 리셀러 신청

다음 단계에서는 리셀러가 AWS 계정 되기 위한 추가 요청을 제출합니다.

> **Tip**
>
> 를 사용하는 경우 추가할 로 관리 계정을 지정 AWS 계정 해야 AWS Organizations합니다. 이 접근 방식은 증가된 기본 Lightsail 인스턴스 할당량을 관리 계정 조직의 모든 멤버 계정으로 확대합니다.

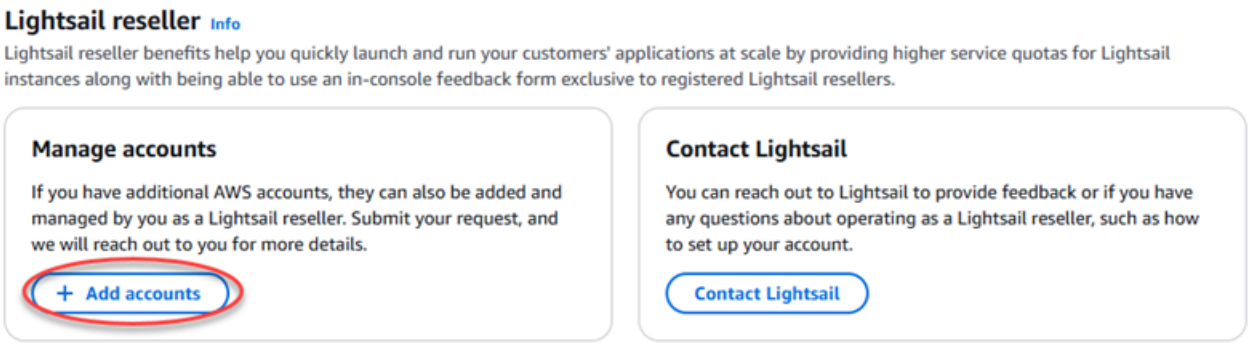
추가 계정의 Lightsail 리셀러 자격을 신청하려면

1. Lightsail 콘솔에 로그인합니다.
2. Lightsail 홈 페이지의 상단 탐색 메뉴에서 계정을 선택합니다.
3. 드롭다운 메뉴에서 계정(Account)을 선택합니다.



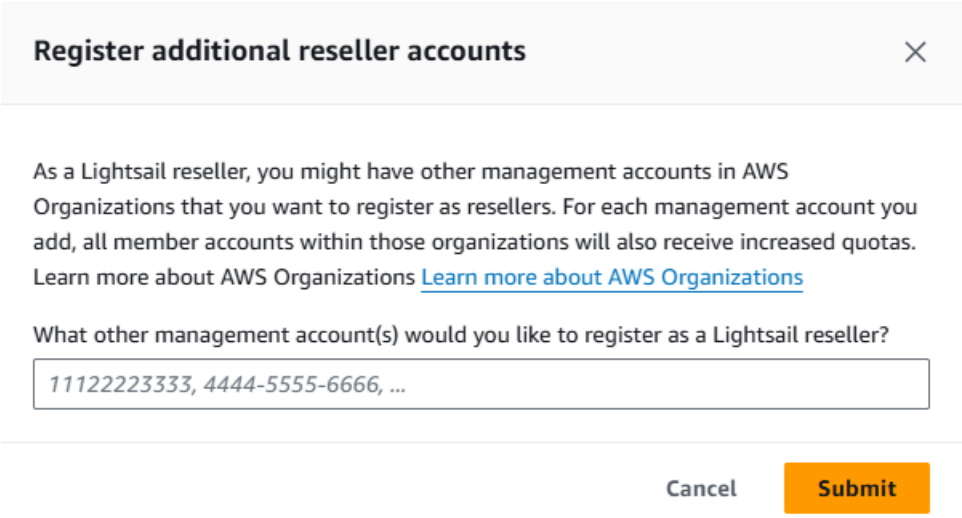
4. 프로필 탭의 Lightsail 리셀러 섹션에서 계정 추가를 선택합니다.

> **Important**
>
> 계정 추가 작업은 Lightsail 리셀러 자격을 신청하고 수락된 계정에서만 사용할 수 있습니다.



5. 등록 양식에 등록하려는 조직의 추가 AWS 계정 IDs 또는 관리 계정을 입력합니다.

> **Note**
>
> Organizations를 사용하는 경우, 멤버 계정을 신청할 필요가 없습니다.



6. 제출을 선택합니다.

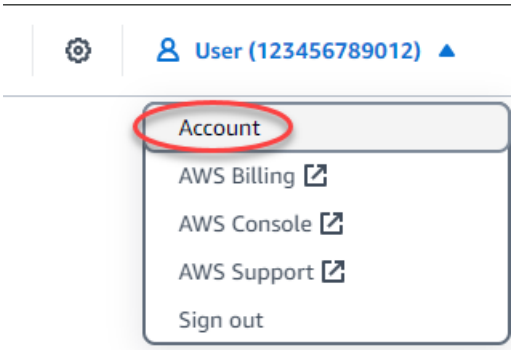
리셀러 계정에 대한 서비스 할당량 증가 요청

Amazon Lightsail 리셀러가 되면 현재 계정과 조직 전체의 멤버 계정에 대해 Lightsail 인스턴스의 기본 서비스 할당량이 증가됩니다. 멤버 계정에 대한 한도를 추가로 늘리려는 경우, 다음 프로세스를 사용하여 할당량 증가를 요청해야 합니다. Lightsail 콘솔에서 현재 할당량을 확인하고 증가를 요청할 수 있습니다.

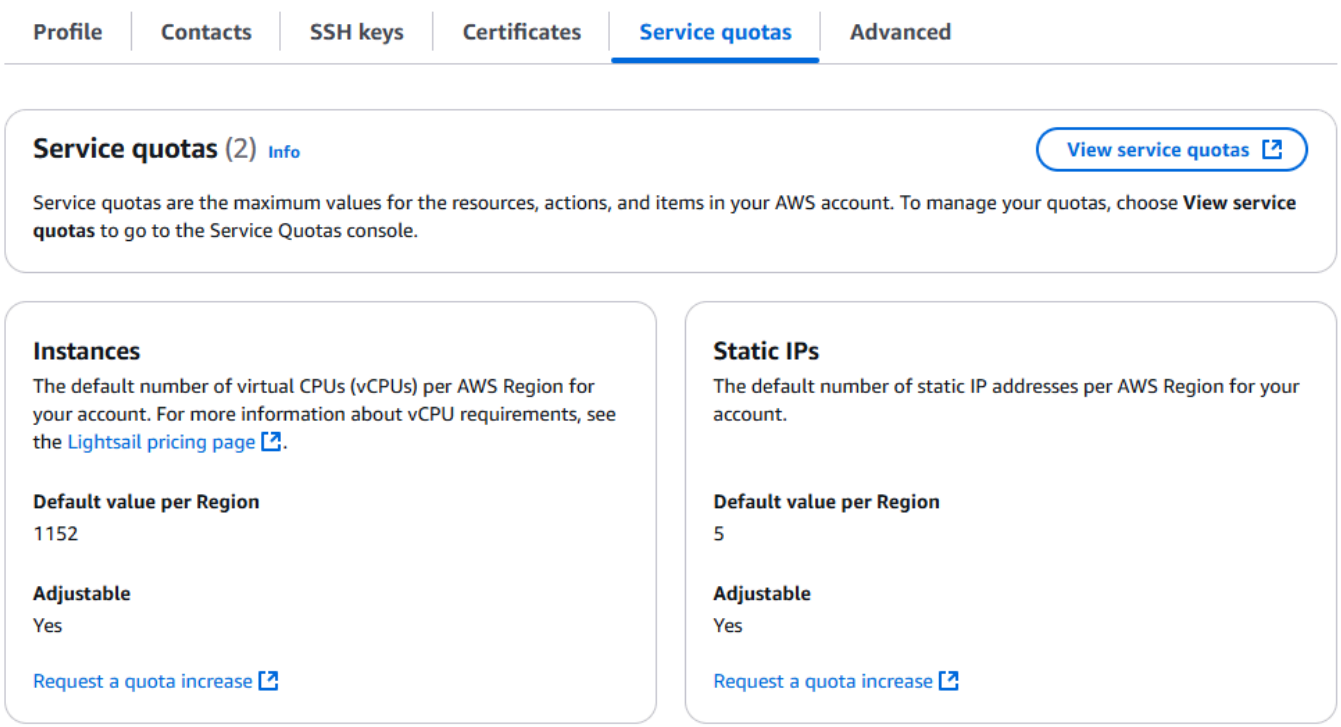
> **Note**
>
> Lightsail 리셀러 계정에 연결된 여러 멤버 계정의 경우, 리셀러 피드백 양식을 사용하여 서비스 할당량 증가를 요청해야 합니다. 자세한 내용은 리셀러로서 Lightsail에 문의하기 단원을 참조하십시오.

리셀러 계정에 대한 서비스 할당량 증가를 요청하려면

1. Lightsail 콘솔에 로그인합니다.
2. Lightsail 홈 페이지의 상단 탐색 메뉴에서 계정을 선택합니다.
3. 드롭다운 메뉴에서 계정(Account)을 선택합니다.



4. 서비스 할당량 탭을 선택합니다.
5. 늘리려는 할당량에 대해 할당량 증가 요청을 선택합니다.



6. Service Quotas 콘솔에서 계정 수준의 증가 요청을 선택합니다.
7. 할당량 값 증가에 증가량을 입력합니다.
8. 요청을 제출하려면 요청을 선택합니다.

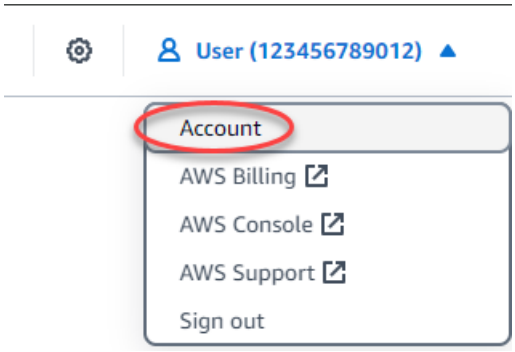
할당량 증가 요청이 제출되면 업데이트를 모니터링할 수 있는 지원 사례가 생성될 수 있습니다. 증가가 승인되면 리전의 모든 리셀러 계정에 적용됩니다. 목록에 없는 할당량 증가는 리셀러로서 Lightsail에 문의하기 섹션을 참조하세요.

리셀러로서 Lightsail에 문의하기

Amazon Lightsail 리셀러는 Lightsail 콘솔에서 바로 Lightsail 팀에 문의하여 리셀러 업무에 대한 질문이나 피드백을 제출할 수 있습니다. 이는 조직의 멤버 계정 전체에 대한 Lightsail 서비스 할당량 증가를 요청하는 방법이기도 합니다.

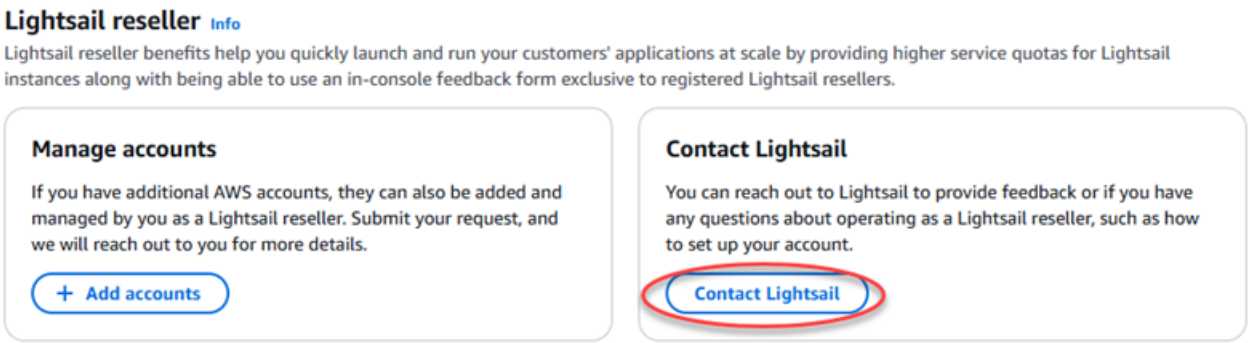
Lightsail 팀에 문의하려면

1. Lightsail 콘솔에 로그인합니다.
2. Lightsail 홈 페이지의 상단 탐색 메뉴에서 계정을 선택합니다.
3. 드롭다운 메뉴에서 계정(Account)을 선택합니다.

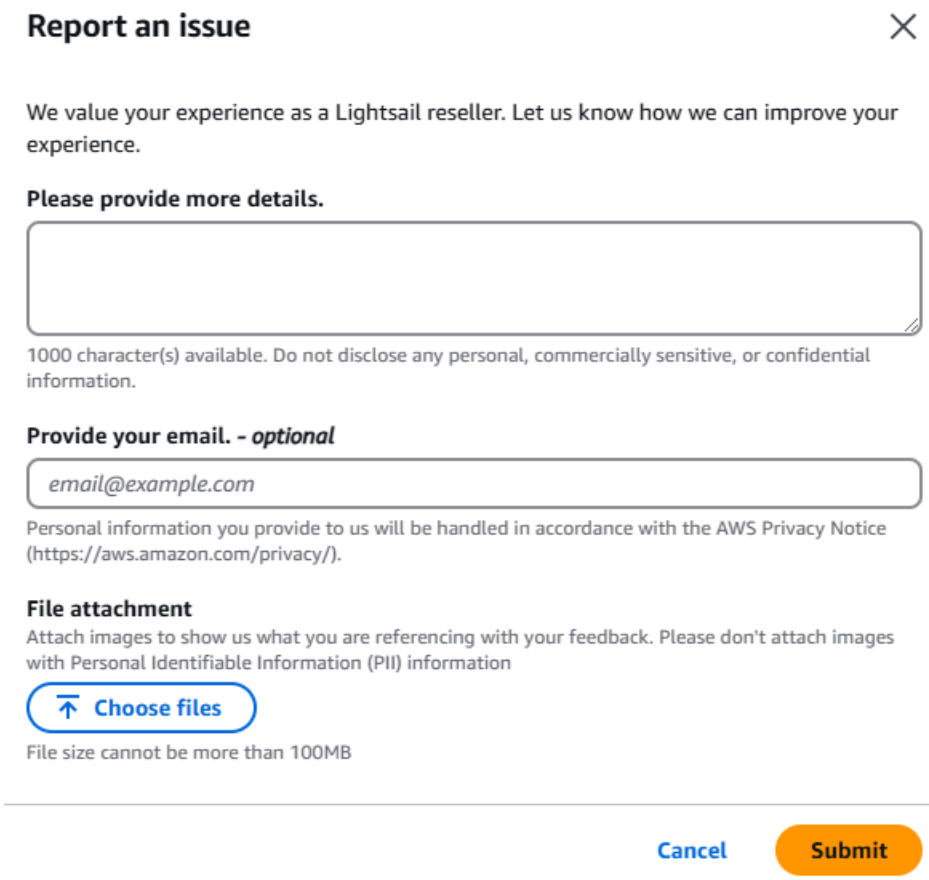


4. 프로필 탭의 Lightsail 리셀러 섹션에서 Lightsail 문의를 선택합니다.

> **Important**
>
> Lightsail 문의 작업은 Lightsail 리셀러 자격을 신청하고 수락된 계정에서만 사용할 수 있습니다. 자세한 내용은 Lightsail 리셀러 되기 섹션을 참조하세요.



5. 요청과 관련된 필수 필드를 작성합니다. Lightsail에 대한 서비스 할당량 증가를 요청하는 경우 여러 멤버 계정을 지정할 수 있습니다.



6. 제출을 선택합니다.

이메일 주소를 제공하는 경우, 피드백에 대한 연락을 받게 될 수 있습니다.

Lightsail의 가상 프라이빗 서버 인스턴스

Lightsail 인스턴스는 가상 머신이라고도 하는 가상 프라이빗 서버입니다. 인스턴스를 생성할 때 OS(운영 체제)을 가진 이미지를 선택합니다. 기본 OS를 포함하여, 애플리케이션 또는 개발 스택을 가진 인스턴스 이미지를 선택할 수도 있습니다.

운영 체제, 애플리케이션 및 개발 프레임워크의 전체 목록은 Lightsail 인스턴스 이미지 선택을 참조하세요.

인스턴스에 대한 자세한 내용은 다음 주제를 참조하세요.

주제

- Lightsail 인스턴스 생성
- Lightsail 인스턴스 블루프린트 제품 검토
- Lightsail 인스턴스 번들
- Lightsail의 방화벽을 사용하여 인스턴스 트래픽 제어
- 최적의 성능을 위해 Lightsail 인스턴스 버스팅 감지
- Lightsail 인스턴스 연결 및 관리
- Lightsail 인스턴스 삭제
- SSH 키 페어 관리 및 Lightsail 인스턴스 연결
- Lightsail의 인스턴스 메타데이터 서비스(IMDS) 및 사용자 데이터 액세스

Lightsail 인스턴스 생성

이 섹션에서는 Amazon Lightsail에서 인스턴스 생성과 관련된 다음 주제를 다룹니다.

주제

- Lightsail의 앱을 사용하여 Linux/Unix 인스턴스 생성
- Lightsail에서 Windows Server 인스턴스 생성

Lightsail의 앱을 사용하여 Linux/Unix 인스턴스 생성

> 알고 계셨나요?
>
> Lightsail는 일별 스냅샷 7개를 저장하고 인스턴스에 대해 자동 스냅샷을 활성화하면 가장 오래된 스냅샷을 최신 스냅샷으로 자동 대체합니다. 자세한 내용은 Lightsail 인스턴스 및 디스크의 자동 스냅샷 구성을 참조하세요.

WordPress 같은 애플리케이션이나 LAMP 같은 개발 스택을 실행하는 Linux/Unix 기반의 Amazon Lightsail 인스턴스(가상 프라이빗 서버)를 생성합니다. 인스턴스 실행이 시작된 후 Lightsail에서 나가지 않고 SSH를 통해 인스턴스에 연결할 수 있습니다. 그 방법은 다음과 같습니다.

Windows 기반의 인스턴스를 생성하는 방법은 Amazon Lightsail에서 Windows 기반 인스턴스 시작하기를 참조하세요.

Linux 기반의 인스턴스 생성

1. 홈 페이지에서 인스턴스 생성을 선택합니다.
2. 인스턴스의 위치(AWS 리전 및 가용 영역)를 선택합니다.

   변경 AWS 리전 및 가용 영역을 선택하여 다른 위치에 인스턴스를 생성합니다.
3. 선택적으로, 가용 영역을 변경할 수 있습니다.

   가용 영역 변경을 선택합니다.
4. Linux 플랫폼을 선택합니다.
5. 애플리케이션(앱 + OS) 또는 운영 체제(OS 전용)를 선택합니다.

   Lightsail 인스턴스 이미지에 대해 자세히 알아보려면 Amazon Lightsail 인스턴스 이미지 선택 섹션을 참조하세요.
6. 인스턴스 플랜을 선택합니다.

   인스턴스가 이중 스택(IPv4 및 IPv6)을 사용할지 IPv6 전용을 사용할지 선택합니다. 일부 Lightsail 블루프린트는 현재 IPv6 전용 네트워킹을 지원하지 않습니다. IPv6 전용 네트워킹을 지원하는 블루프린트를 확인하려면 Lightsail 인스턴스 블루프린트 제품 검토 섹션을 참조하세요.

   5 USD Lightsail 플랜을 한 달 동안 무료로 체험할 수 있습니다(최대 750시간). 귀하의 계정에 1개월 무료 이용권을 드리겠습니다. Lightsail 요금 페이지에 대해 자세히 알아보세요.

> **Note**
>
> AWS 프리 티어의 일부로 일부 인스턴스 번들에서 Amazon Lightsail를 무료로 시작할 수 있습니다. 자세한 내용은 Amazon Lightsail 요금 페이지의 AWS 프리 티어를 참조하세요.

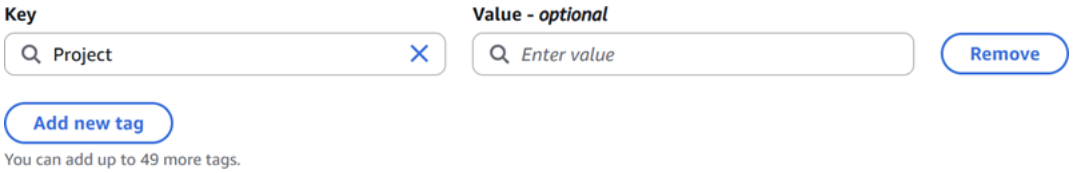
7. 인스턴스 이름을 입력합니다.

   리소스 이름:

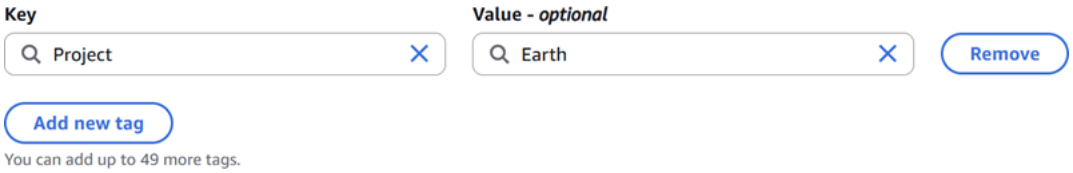
   - Lightsail 계정 AWS 리전 의 각 내에서 고유해야 합니다.
   - 2~255자의 문자로 구성되어야 합니다.
   - 영숫자 문자 또는 숫자로 시작하고 끝나야 합니다.
   - 영숫자 문자, 숫자, 마침표, 대시, 밑줄이 포함될 수 있습니다.

8. (선택 사항) 인스턴스에 태그를 추가하려면 새 태그 추가를 선택합니다. 필요에 따라 이 단계를 반복하여 모든 태그를 추가합니다. 태그 사용에 대한 자세한 내용은 태그를 참조하세요.

   a. 키에는 태그 키를 입력합니다.

   b. (선택 사항) 값에 태그 값을 입력합니다.

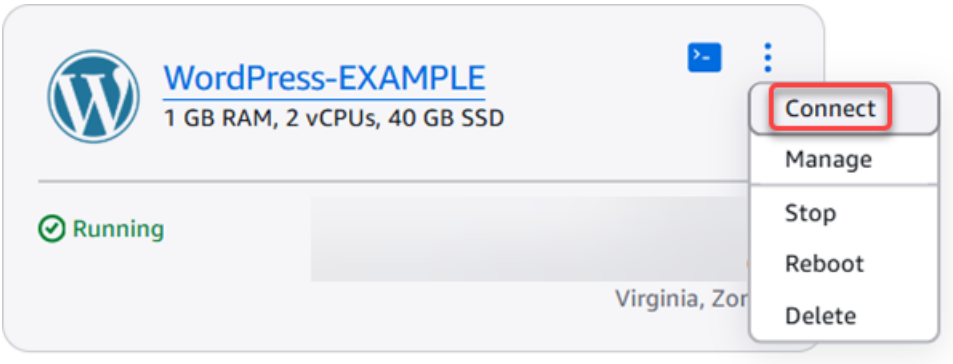
9. 인스턴스 생성을 선택합니다.

   고급 생성 옵션에 대해서는 인스턴스가 시작되면 시작 스크립트를 사용하여 Amazon Lightsail 인스턴스 구성 또는 Linux/Unix 기반 Lightsail 인스턴스를 위한 SSH 설정을 참조하세요.

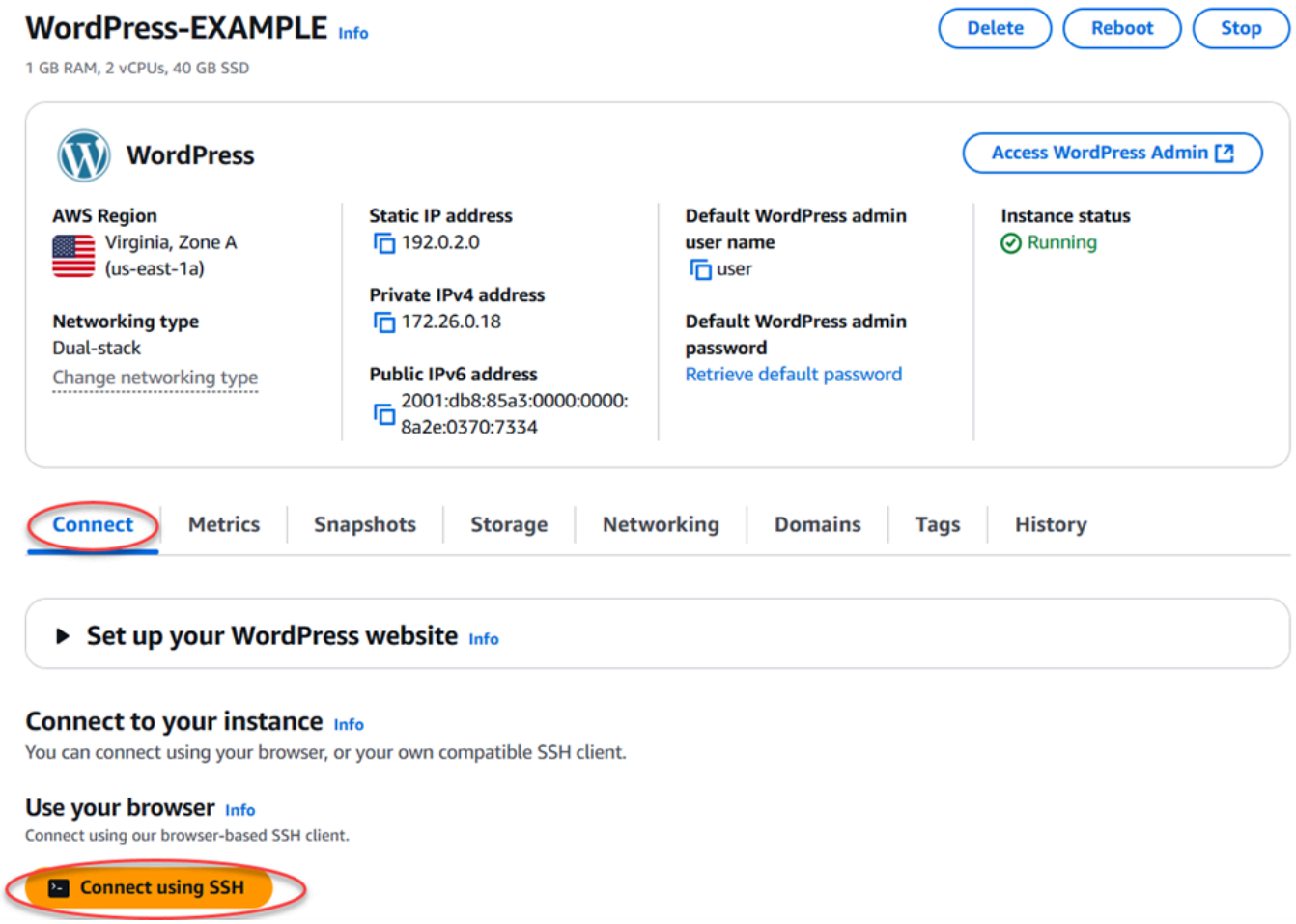
Lightsail 인스턴스는 몇 분 내에 준비되며 Lightsail에서 나가지 않고도 SSH를 통해 인스턴스에 연결할 수 있습니다.

인스턴스에 연결합니다

1. Lightsail 홈 페이지에서 인스턴스의 이름 오른쪽에 있는 메뉴를 선택하고 연결을 선택합니다.

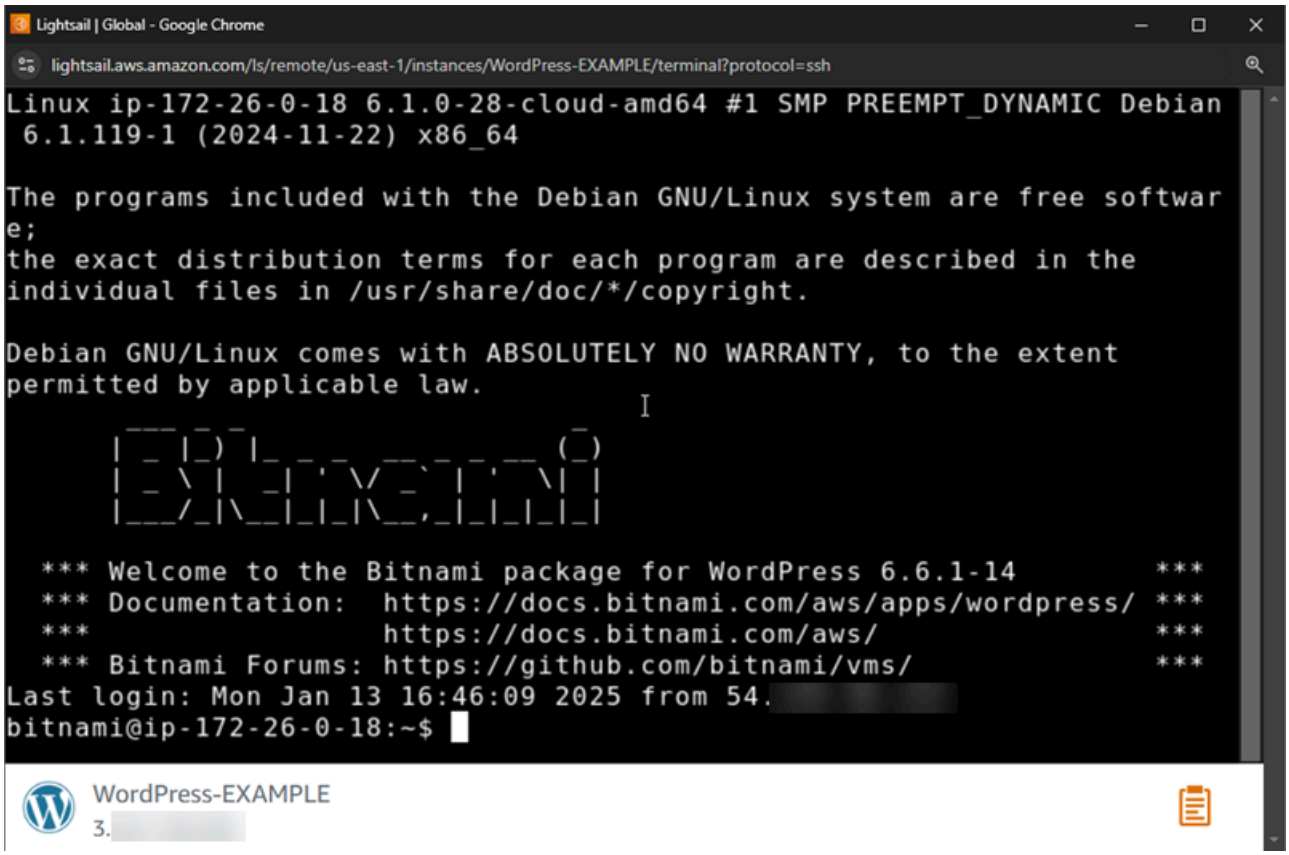


다른 방법으로, 인스턴스 관리 페이지를 열고 연결 탭을 선택한 다음 SSH를 사용하여 연결을 선택할 수 있습니다.



> **Note**
>
> PuTTY와 같은 SSH 클라이언트를 사용하여 인스턴스에 연결하려면 PuTTY를 설정하여 Lightsail 인스턴스에 연결 절차에 따르면 됩니다.

2. 이제 터미널에 명령을 입력한 다음, SSH 클라이언트를 설치하지 않고도 Lightsail 인스턴스를 관리할 수 있습니다.



다음 단계

인스턴스에 연결할 수 있으므로, 다음으로 할 일은 인스턴스 사용 플랜에 따라 다릅니다. 예제:

- the section called “WordPress” 블로그를 생성하는 경우.
- Lightsail 인스턴스를 다시 시작할 때마다 동일한 IP 주소를 유지할 고정 IP 주소를 생성합니다.
- 백업으로 사용할 인스턴스의 스냅샷을 생성합니다.

Lightsail에서 Windows Server 인스턴스 생성

> 알고 계셨나요?
>
> Lightsail는 일별 스냅샷 7개를 저장하고 인스턴스에 대해 자동 스냅샷을 활성화하면 가장 오래된 스냅샷을 최신 스냅샷으로 자동 대체합니다. 자세한 내용은 Lightsail 인스턴스 및 디스크의 자동 스냅샷 구성을 참조하세요.

Windows Server 운영 체제(OS)를 실행하는 Lightsail 인스턴스를 생성합니다. 사용할 수 있는 OS 블루프린트는 Windows Server 2022, Windows Server 2019, Windows Server 2016 등 3가지입니다. 이와 함께 SQL Server 2022, 2019 및 2016 Express로 미리 구성된 블루프린트도 있습니다.

이 주제에서는 소프트웨어를 선택하는 방법과 Windows Server 기반의 인스턴스를 생성하고 연결하는 방법을 다룹니다.

AWS의 Windows Server에 대해 자세히 알아보기

Windows Server 기반의 인스턴스 선택

Lightsail에서 Windows Server 기반의 인스턴스를 생성할 때 세 가지 옵션을 선택할 수 있습니다.

Windows Server 2022

Windows Server를 실행하는 Lightsail은 Microsoft 웹 플랫폼을 사용하여 애플리케이션을 빠르고 안정적으로 배포할 수 있는 환경입니다. Lightsail을 사용하면 호환 가능한 Windows 기반 솔루션을 비용 효율성이 높은 안정적인 고성능 AWS 클라우드 컴퓨팅 플랫폼에서 실행할 수 있습니다. 일반적인 Windows 사용 사례에는 Enterprise Windows 기반 애플리케이션 호스팅, 웹 사이트 및 웹 서비스 호스팅, 데이터 처리, 분산 시험, ASP.NET 애플리케이션 호스팅 및 Windows 소프트웨어가 필요한 기타 애플리케이션 등이 포함됩니다.

Windows Server 2022 이미지에 대해 자세히 알아보기

Windows Server 2019

이유가 있어서 Windows Server 2016 또는 Windows Server 2019을 실행해야 하는 경우가 아니라면, 최신 버전의 Windows Server 2022를 사용하는 것이 좋습니다.

Lightsail Windows Server 실행은 Microsoft 웹 플랫폼을 사용하여 애플리케이션을 배포하기 위한 빠르고 신뢰할 수 있는 환경입니다.를 Lightsail 사용하면 안정적이고 비용 효율적인 AWS고성능 클

라우드 컴퓨팅 플랫폼에서 호환되는 모든 Windows 기반 솔루션을 실행할 수 있습니다. 일반적인 Windows 사용 사례에는 엔터프라이즈 Windows 기반 애플리케이션 호스팅, 웹 사이트 및 웹 서비스 호스팅, 데이터 처리, 분산 시험, ASP.NET 애플리케이션 호스팅 및 Windows 소프트웨어가 필요한 기타 애플리케이션 등이 포함됩니다.

Windows Server 2019 이미지에 대해 자세히 알아보기

Windows Server 2016

이유가 있어서 Windows Server 2016 또는 Windows Server 2019을 실행해야 하는 경우가 아니라면, 최신 버전의 Windows Server 2022를 사용하는 것이 좋습니다.

Windows Server를 실행하는 Lightsail은 Microsoft 웹 플랫폼을 사용하여 애플리케이션을 배포할 수 있는 빠르고 믿음직한 환경입니다. Lightsail을 사용하면 AWS의 안정적이고 비용 효율적인 고성능 클라우드 컴퓨팅 플랫폼에서 호환 가능한 모든 Windows 기반 솔루션을 실행할 수 있습니다. 일반적인 Windows 사용 사례에는 엔터프라이즈 Windows 기반 애플리케이션 호스팅, 웹 사이트 및 웹 서비스 호스팅, 데이터 처리, 분산 시험, ASP.NET 애플리케이션 호스팅 및 Windows 소프트웨어가 필요한 기타 애플리케이션 등이 포함됩니다.

Windows Server 2016 이미지에 대해 자세히 알아보기

SQL Server Express 2022

SQL Server Express는 무료로 다운로드, 배포, 사용할 수 있는 관계형 데이터베이스 관리 시스템으로, 특히 임베디드 애플리케이션 및 소규모 애플리케이션용 데이터베이스에 사용됩니다. 이 Lightsail 이미지는 Windows Server 2022의 기본 OS에서 실행됩니다.

SQL Server Express 2022 이미지에 대해 자세히 알아보기

SQL Server Express 2019

SQL Server Express는 무료로 다운로드, 배포, 사용할 수 있는 관계형 데이터베이스 관리 시스템으로, 특히 임베디드 애플리케이션 및 소규모 애플리케이션용 데이터베이스에 사용됩니다. 이 Lightsail 이미지는 Windows Server 2022의 기본 OS에서 실행됩니다.

SQL Server Express 2019 이미지에 대해 자세히 알아보기

SQL Server Express 2016

SQL Server Express는 무료로 다운로드, 배포, 사용할 수 있는 관계형 데이터베이스 관리 시스템으로, 특히 임베디드 애플리케이션 및 소규모 애플리케이션용 데이터베이스에 사용됩니다. 이 Lightsail 이미지는 Windows Server 2016의 기본 OS에서 실행됩니다.

SQL Server Express 이미지에 대해 자세히 알아보기

Windows Server 기반의 인스턴스 생성

Lightsail 콘솔을 사용하거나 AWS Command Line Interface ()를 사용하여 Windows Server 기반 인스턴스를 생성할 수 있습니다AWS CLI.

콘솔을 사용하여 인스턴스를 생성하려면

1. Lightsail에 로그인한 다음 홈 페이지로 이동합니다.
2. 인스턴스 생성을 선택합니다.
3. Windows Server 기반 Lightsail 인스턴스를 생성할 AWS 리전 를 선택합니다.

   예를 들어 `Ohio (us-east-2)`입니다.
4. Microsoft Windows 플랫폼을 선택합니다.
5. Windows Server 2022, Windows Server 2019, Windows Server 2016 블루프린트를 원한다면 OS 전용을 선택합니다.

   SQL Server Express 블루프린트를 원한다면 앱 + OS를 선택합니다.
6. 인스턴스 플랜을 선택합니다.

   인스턴스가 이중 스택(IPv4 및 IPv6)을 사용할지 IPv6 전용을 사용할지 선택합니다. 일부 Lightsail 블루프린트는 현재 IPv6 전용 네트워킹을 지원하지 않습니다. IPv6 전용 네트워킹을 지원하는 블루프린트를 확인하려면 Lightsail 인스턴스 블루프린트 제품 검토 섹션을 참조하세요.

   요금제에는 데이터 전송은 물론 머신 구성(RAM, SSD, vCPU)도 포함되며, 예측 가능하고 저렴한 비용이 특징입니다.

   > **Note**
   >
   > 일부 블루프린트에는 사용할 수 없는 인스턴스 요금제도 있습니다. 예를 들어 SQL Server Express 블루프린트에서는 최소 4GB의 메모리와 80GB의 SSD 스토리지가 있는 계획을 사용해야 합니다.
7. 인스턴스 이름을 입력합니다.

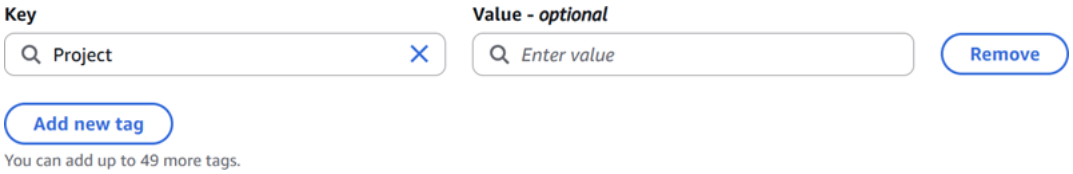
   리소스 이름:

   - Lightsail 계정 AWS 리전 의 각 내에서 고유해야 합니다.

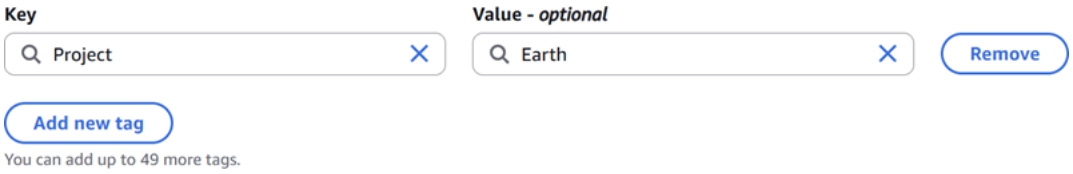
- 2~255자의 문자로 구성되어야 합니다.
- 영숫자 문자 또는 숫자로 시작하고 끝나야 합니다.
- 영숫자 문자, 숫자, 마침표, 대시, 밑줄이 포함될 수 있습니다.

8. (선택 사항) 인스턴스에 태그를 추가하려면 새 태그 추가를 선택합니다. 필요에 따라 이 단계를 반복하여 모든 태그를 추가합니다. 태그 사용에 대한 자세한 내용은 태그를 참조하세요.

   a. 키에는 태그 키를 입력합니다.

   b. (선택 사항) 값에 태그 값을 입력합니다.

9. 인스턴스 생성을 선택합니다.

를 사용하여 인스턴스를 생성하려면 AWS CLI

1. AWS CLI를 아직 설치 및 구성하지 않았다면 설치하고 구성합니다.

   자세한 내용은 AWS Command Line Interface 가 Amazon Lightsail과 함께 작동하도록 구성을 참조하세요.

2. 명령 프롬프트 또는 터미널 창을 엽니다.
3. 아직 구성하지 않은 경우를 AWS CLI 사용하여를 구성`aws configure`하고 Lightsail 리소스를 생성할 AWS 리전 를 선택합니다.
4. 다음 AWS CLI 명령을 입력하여 오하이오 리전에서 실행되는 월별 44 USD Windows Server 2022 인스턴스를 생성합니다.

```
aws lightsail create-instances --instance-names InstanceName --availability-zone
 us-east-2a --blueprint-id windows_server_2022 --bundle-id medium_win_3_0
```

   명령에서 *InstanceName*을 새 인스턴스의 이름으로 바꿉니다.

   성공하면 AWS CLI에 다음과 같은 결과가 출력됩니다.

```
{
    "operations": [
        {
            "status": "Started",
            "resourceType": "Instance",
            "isTerminal": false,
            "statusChangedAt": 1508086226.4,
            "location": {
                "availabilityZone": "us-east-2a",
                "regionName": "us-east-2"
            },
            "operationType": "CreateInstance",
            "resourceName": "my-windows-instance",
            "id": "344acdc8-f9c4-4eda-8232-12345EXAMPLE",
            "createdAt": 1508086225.467
        }
    ]
}
```


> **Note**
>
> 사용 가능한 블루프린트 목록을 보려면 get-blueprints 명령을 사용하세요. 사용 가능한 번들 목록을 보려면 get-bundles 명령을 사용하세요. 인스턴스의 암호를 가져오는 방법을 자세히 알아보려면 get-instance-access-details 명령을 사용하세요.

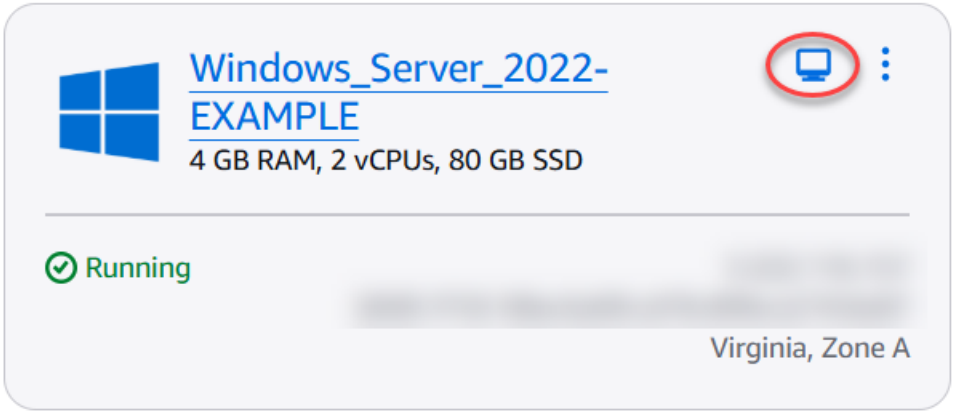
인스턴스에 연결합니다

Windows Server 기반의 Lightsail 인스턴스가 생성되면 브라우저 기반의 RDP 클라이언트나 원하는 원격 데스크톱 클라이언트를 통해 그 인스턴스에 연결할 수 있습니다.

> **Note**
>
> 생성된 인스턴스에 연결할 수 있게 되기까지 최대 15분간 기다려야 할 수 있습니다.

Lightsail의 브라우저 기반 RDP 클라이언트를 사용하여 연결하려면

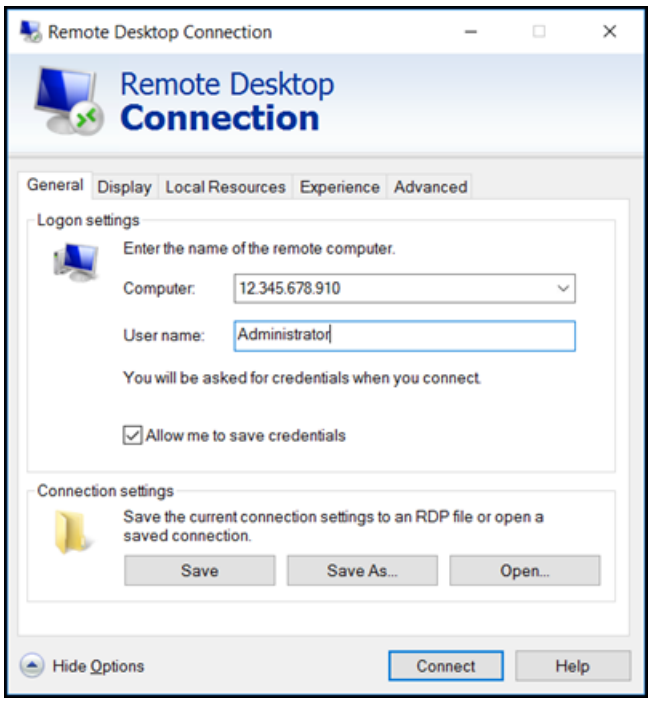
1. 홈 페이지에서 인스턴스 옆에 있는 RDP를 사용하여 연결 아이콘을 선택합니다.



2. 아니면 바로 가기 메뉴나 인스턴스 관리 페이지에서 인스턴스에 연결하는 방법도 있습니다.

자체 RDP 클라이언트를 사용하여 연결하려면

1. IP 주소를 얻기 위해 Lightsail 홈 페이지로 이동합니다.
2. IP 주소를 클립보드에 복사합니다.
3. Windows의 원격 데스크톱 연결 같은 RDP 클라이언트를 엽니다.
4. 컴퓨터 필드에 IP 주소를 붙여넣습니다.
5. 옵션 표시(Show Options)를 선택한 다음 사용자 이름(User name)에 `Administrator`를 입력합니다.

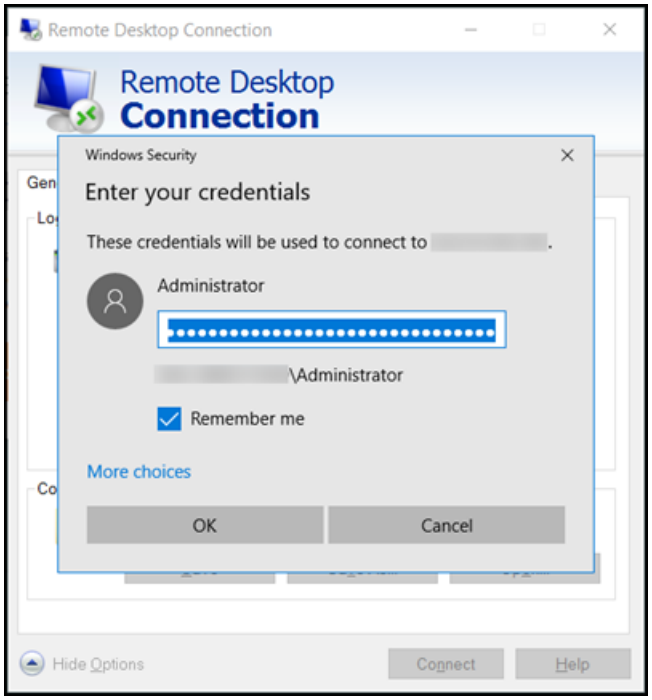


6. 연결을 선택합니다.

7. 암호를 얻기 위해 Lightsail의 인스턴스 관리 페이지로 이동합니다.

   Lightsail 홈 페이지에서 인스턴스 이름을 선택하거나 바로 가기 메뉴에서 관리(Manage)를 선택하여 인스턴스 관리 페이지로 이동할 수 있습니다.

8. 기본 암호 표시를 선택합니다.
9. 기본 암호를 클립보드에 복사합니다.
10. 원격 데스크톱 연결에 암호를 붙여넣은 다음, 자격 증명 저장을 선택하여 앞으로 이 대화 상자가 표시되지 않게 합니다.



11. 확인을 선택합니다.
12. 이 컴퓨터에 대한 연결 여부를 다시 묻지 않음을 선택하고 예를 선택합니다.

다음 단계

인스턴스에 연결할 수 있으므로, 다음으로 할 일은 인스턴스 사용 플랜에 따라 다릅니다. 예제:

- Lightsail 인스턴스를 다시 시작할 때마다 동일한 IP 주소를 유지할 고정 IP 주소를 생성합니다.
- Lightsail Windows Server 인스턴스의 스냅샷을 생성합니다.

단계별 지침에 따라 Amazon Linux, Ubuntu, Debian 또는 Windows Server 2022, 2019, 2016 등의 Windows Server 운영 체제와 같은 Linux 및 Unix 배포를 실행하는 인스턴스를 생성합니다.

Linux 및 Unix 인스턴스의 경우 WordPress, LAMP, LEMP와 같은 다양한 애플리케이션 블루프린트 중에서 선택하거나 운영 체제만 선택할 수 있습니다. Windows Server 인스턴스의 경우 Windows Server 블루프린트 또는 SQL Server Express 블루프린트 중에서 선택할 수 있습니다.

이 가이드에서는 AWS 리전 및 가용 영역 선택, 원하는 컴퓨팅 및 스토리지 리소스로 인스턴스 계획(번들) 선택, IPv4 및 IPv6와 같은 네트워킹 옵션 구성, 인스턴스 이름 지정, 태그 추가를 다룹니다. 인스턴스를 생성한 후 Lightsail 브라우저 기반 SSH 또는 RDP 클라이언트를 사용하여 인스턴스에 연결하거나 제공된 연결 세부 정보와 함께 자체 SSH 또는 RDP 클라이언트를 사용할 수 있습니다. 이 가이드를 따르면 특정 요구 사항에 맞게 Lightsail에서 Linux 및 Unix 또는 Windows Server 인스턴스를 빠르게 시작하고 액세스할 수 있습니다.

Lightsail 인스턴스 블루프린트 제품 검토

Lightsail은 가상 프라이빗 서버를 만들 수 있는 여러 가지 옵션을 제공합니다. 이 주제는 프로젝트에 알맞은 OS(운영 체제), 애플리케이션 또는 개발 스택을 결정하는 데 도움이 됩니다. Amazon은 애플리케이션을 기능별 영역(CMS와 전자 상거래 등) 기준으로 구성했습니다.

운영 체제

Lightsail에는 선택할 수 있는 여러 Linux/Unix 기반 또는 Windows 기반 운영 체제가 있습니다.

Windows Server 2022

Windows Server를 실행하는 Lightsail은 Microsoft 웹 플랫폼을 사용하여 애플리케이션을 빠르고 안정적으로 배포할 수 있는 환경입니다. Lightsail을 사용하면 호환 가능한 Windows 기반 솔루션을 비용 효율성이 높은 안정적인 고성능 AWS 클라우드 컴퓨팅 플랫폼에서 실행할 수 있습니다. 일반적인 Windows 사용 사례에는 Enterprise Windows 기반 애플리케이션 호스팅, 웹 사이트 및 웹 서비스 호스팅, 데이터 처리, 분산 시험, ASP.NET 애플리케이션 호스팅 및 Windows 소프트웨어가 필요한 기타 애플리케이션 등이 포함됩니다. 지원 종료 정보는 Microsoft 웹 사이트를 참조하세요.

이 블루프린트는 Lightsail IPv6 전용 인스턴스 플랜과 호환됩니다.

Windows Server 2022에 대해 자세히 알아보기

Windows Server 2019

Windows Server를 실행하는 Lightsail은 Microsoft 웹 플랫폼을 사용하여 애플리케이션을 빠르고 안정적으로 배포할 수 있는 환경입니다. Lightsail을 사용하면 호환 가능한 Windows 기반 솔루션을 비용 효율성이 높은 안정적인 고성능 AWS 클라우드 컴퓨팅 플랫폼에서 실행할 수 있습니다. 일반적인 Windows 사용 사례에는 Enterprise Windows 기반 애플리케이션 호스팅, 웹 사이트 및 웹 서비

스 호스팅, 데이터 처리, 분산 시험, ASP.NET 애플리케이션 호스팅 및 Windows 소프트웨어가 필요한 기타 애플리케이션 등이 포함됩니다. 지원 종료 정보는 Microsoft 웹 사이트를 참조하세요.

이 블루프린트는 Lightsail IPv6 전용 인스턴스 플랜과 호환됩니다.

Windows Server 2019에 대해 자세히 알아보기

Windows Server 2016

Windows Server를 실행하는 Lightsail은 Microsoft 웹 플랫폼을 사용하여 애플리케이션을 빠르고 안정적으로 배포할 수 있는 환경입니다. Lightsail을 사용하면 호환 가능한 Windows 기반 솔루션을 비용 효율성이 높은 안정적인 고성능 AWS 클라우드 컴퓨팅 플랫폼에서 실행할 수 있습니다. 일반적인 Windows 사용 사례에는 Enterprise Windows 기반 애플리케이션 호스팅, 웹 사이트 및 웹 서비스 호스팅, 데이터 처리, 분산 시험, ASP.NET 애플리케이션 호스팅 및 Windows 소프트웨어가 필요한 기타 애플리케이션 등이 포함됩니다. 지원 종료 정보는 Microsoft 웹 사이트를 참조하세요.

이 블루프린트는 Lightsail IPv6 전용 인스턴스 플랜과 호환됩니다.

Windows Server 2016에 대해 자세히 알아보기

Amazon Linux 2023

Amazon Linux 2023(AL2023)은 차세대 Amazon Linux로 AWS의 범용 워크로드에 적합합니다. AL2023은 정식 출시 후 5년 동안 지원됩니다. AL2023은 Amazon Linux 패키지 리포지토리의 특정 버전에 고정되므로 업데이트를 흡수하는 방법과 시기를 제어할 수 있습니다. 또한 AL2023은 수시로 기능 업데이트를 제공하며 규정 준수 요구 사항을 충족하는 데 도움이 되는 기능도 제공합니다.

AL2023에서 실행된 Lightsail 인스턴스는 인스턴스 메타데이터 서비스 버전 2(IMDSv2)가 기본으로 적용됩니다. 자세한 내용은 인스턴스 메타데이터 서비스 버전 2 작동 방식 단원을 참조하십시오.

이 블루프린트는 Lightsail IPv6 전용 인스턴스 플랜과 호환됩니다.

Amazon Linux 2023에 대해 자세히 알아보세요.

Amazon Linux 2

> **Important**
>
> Amazon Linux 2는 2026년 6월 30일에 장기 지원 종료에 도달합니다. 2026년 6월 30일 이후에는이 블루프린트를 사용하여 새 Lightsail 인스턴스를 생성할 수 없습니다. 자세한 내용은 Amazon Linux 2 웹 사이트를 참조하세요.

Amazon Linux 2는 AWS의 Linux 서버 운영 체제인 Amazon Linux의 전 세대입니다. 클라우드 및 엔터프라이즈 애플리케이션을 개발하고 실행할 수 있는 안전하고 안정적인 고성능 실행 환경을 제공합니다. Amazon Linux 2를 사용하면 Linux에서 최신 혁신에 액세스할 수 있는 장기적인 지원을 제공하는 애플리케이션 환경을 사용할 수 있습니다. Amazon Linux 2는 추가 요금 없이 제공됩니다. 지원 종료 정보는 Amazon Linux 2 FAQ를 참조하세요.

이 블루프린트는 Lightsail IPv6 전용 인스턴스 플랜과 호환됩니다.

Amazon Linux 2에 대해 자세히 알아보세요.

AlmaLinux OS 9

AlmaLinux OS 9는 장기 안정성에 중점을 둔 커뮤니티 소유 및 관리형 오픈 소스의 영원히 자유로운 엔터프라이즈 Linux 배포로, 강력한 프로덕션 등급 플랫폼을 제공합니다. AlmaLinux는 RHEL® 및 사전 스트림 CentOS와 호환됩니다. 지원 종료 정보는 AlmaLinux OS Foundation 웹 사이트를 참조하세요.

이 블루프린트는 Lightsail IPv6 전용 인스턴스 플랜과 호환됩니다.

AlmaLinux OS 9에 대해 자세히 알아보세요.

CentOS Stream 9

CentOS Stream 9은 CentOS Stream 배포판의 차기 메이저 릴리스 버전입니다. CentOS Stream 9은 RHEL(Red Hat Enterprise Linux) 개발 직전까지 추적하며 지속적으로 제공되는 배포판으로, Fedora Linux와 RHEL의 중간 위치입니다. RHEL과 기능적으로 호환되도록 설계되었으며 안정적이고 예측 가능하며 관리 및 재현 가능한 Linux 환경을 제공합니다. 지원 종료 정보는 CentOS 웹 사이트를 참조하세요.

이 블루프린트는 Lightsail IPv6 전용 인스턴스 플랜과 호환됩니다.

CentOS Stream 웹 사이트에서 자세히 알아보세요.

Debian 11, 12 및 13

> **Important**
>
> Debian 11은 2026년 8월 31일에 장기 지원 종료에 도달합니다. 2026년 8월 31일 이후에는이 블루프린트를 사용하여 새 Lightsail 인스턴스를 생성할 수 없습니다. 자세한 내용은 Debian 웹 사이트를 참조하십시오.

Debian은 인터넷을 통해 협력하는 전 세계 수천 명의 자원봉사자들이 개발한 무료 운영 체제입니다. Debian 프로젝트의 주요 강점은 자원봉사자 기반, Debian 사회 계약과 자유 소프트웨어에 대한 헌신, 가능한 한 최고의 운영 체제를 제공하겠다는 약속입니다. 이번 새 릴리스는 이러한 방향으로 나아가는 또 다른 중요한 단계입니다. 지원 종료 정보는 Debian 웹 사이트를 참조하세요.

이 블루프린트는 Lightsail IPv6 전용 인스턴스 플랜과 호환됩니다.

Debian 웹 사이트에서 자세히 알아보세요.

FreeBSD 13, 14 및 15

> **Important**
>
> FreeBSD 13은 2026년 4월 30일에 수명 종료에 도달합니다. 2026년 4월 30일 이후에는 이 블루프린트를 사용하여 새 Lightsail 인스턴스를 생성할 수 없습니다. 자세한 내용은 FreeBSD 웹 사이트를 참조하십시오.

FreeBSD는 서버, 데스크톱 및 임베디드 시스템을 구동하는 데 사용되는 운영 체제입니다. 캘리포니아 대학교 버클리에서 개발된 UNIX 버전인 BSD에서 파생된 FreeBSD는 30년 이상 대규모 커뮤니티에서 지속적으로 개발되었습니다. pf 방화벽, Capsicum 및 CloudABI 기능 프레임워크, ZFS 파일 시스템 및 DTrace 동적 추적 프레임워크를 포함한 FreeBSD의 네트워킹, 보안, 스토리지 및 모니터링 기능은 FreeBSD를 가장 많은 웹 사이트와 가장 널리 퍼져 있는 임베디드 네트워킹 및 스토리지 시스템을 위한 플랫폼으로 선택하게 합니다. 지원 종료 정보는 FreeBSD 웹 사이트를 참조하세요.

이 블루프린트는 Lightsail IPv6 전용 인스턴스 플랜과 호환됩니다.

자세한 내용은 FreeBSD 웹 사이트에서 확인하세요.

openSUSE 15

openSUSE 배포 버전은 안정적이고 사용하기 쉬우며 완전한 다목적 Linux 배포 버전입니다. 이 버전은 데스크톱이나 서버로 작업하는 사용자와 개발자를 대상으로 합니다. 초보자, 능숙한 사용자, 최고급 사용자에게 모두 훌륭한 버전입니다. 한 마디로 모든 이에게 완벽한 솔루션이라 할 수 있습니다. 지원 종료 정보는 openSUSE 웹 사이트를 참조하세요.

암호 인증은 이 운영 체제에 대해 기본적으로 비활성화되어 있습니다. 이는 암호 인증이 활성화된 인스턴스의 스냅샷에서 인스턴스를 생성하는 경우에도 새 인스턴스의 암호 인증이 비활성화됨을 의미합니다. SUSE Linux의 암호 인증에 대한 자세한 내용은 SUSE 설명서의 문서 3404214를 참조하세요.

암호 인증이 비활성화된 상태로 인스턴스에 로그인하려면 Lightsail 콘솔 또는 키 페어에서 브라우저 기반 SSH 클라이언트를 사용할 수 있습니다. 로그인에 대한 자세한 내용은 Lightsail에서 Linux 또는 Unix 인스턴스에 연결 또는 SSH 명령을 사용하여 Lightsail Linux 또는 Unix 인스턴스에 연결을 참조하세요.

이 블루프린트는 Lightsail IPv6 전용 인스턴스 플랜과 호환됩니다.

자세한 내용은 openSUSE 웹 사이트에서 확인하세요.

Ubuntu 22 및 24

Ubuntu Server는 가상 서버용으로 사용되는 Debian 기반 Linux 운영 체제입니다. Ubuntu의 기본 설치 프로그램에는 LibreOffice, Firefox, Thunderbird 및 Transmission을 포함한 폭넓고 다양한 소프트웨어가 들어 있습니다. APT 기반 패키지 관리 도구(`apt-get`)를 사용하여 Evolution, GIMP, Pidgin 및 Synaptic 같은 추가 소프트웨어 패키지를 다수 설치할 수 있습니다. 지원 종료 정보는 Ubuntu 웹 사이트를 참조하세요.

LightsailUbuntu 24 블루프린트로 생성된 인스턴스는 기본적으로 인스턴스 메타데이터 서비스 버전 2(IMDSv2)가 적용됩니다. 자세한 내용은 인스턴스 메타데이터 서비스 버전 2 작동 방식 단원을 참조하십시오.

이 블루프린트는 Lightsail IPv6 전용 인스턴스 플랜과 호환됩니다.

자세한 내용은 Ubuntu 웹 사이트에서 확인하세요.

데이터베이스 애플리케이션

다음은 Lightsail에서 사용할 수 있는 데이터베이스 애플리케이션입니다.

SQL Server 2022 Express

SQL Server Express는 무료로 다운로드, 배포, 사용할 수 있는 관계형 데이터베이스 관리 시스템으로, 특히 임베디드 애플리케이션 및 소규모 애플리케이션용 데이터베이스에 사용됩니다. 이 Lightsail 이미지는 Windows Server 2022의 기본 OS에서 실행됩니다.

이 블루프린트는 Lightsail IPv6 전용 인스턴스 플랜과 호환됩니다.

SQL Server 2022 Express에 대해 자세히 알아보세요.

SQL Server 2019 Express

SQL Server Express는 무료로 다운로드, 배포, 사용할 수 있는 관계형 데이터베이스 관리 시스템으로, 특히 임베디드 애플리케이션 및 소규모 애플리케이션용 데이터베이스에 사용됩니다. 이 Lightsail 이미지는 Windows Server 2022의 기본 OS에서 실행됩니다.

이 블루프린트는 Lightsail IPv6 전용 인스턴스 플랜과 호환됩니다.

SQL Server 2019 Express에 대해 자세히 알아보세요.

SQL Server 2016 Express

> **Important**
>
> SQL Server 2016 Express는 2026년 7월 14일에 추가 지원이 종료됩니다. 2026년 7월 14일 이후에는이 블루프린트를 사용하여 새 Lightsail 인스턴스를 생성할 수 없습니다. 자세한 내용은 Microsoft 웹 사이트를 참조하십시오.

SQL Server Express는 무료로 다운로드, 배포, 사용할 수 있는 관계형 데이터베이스 관리 시스템으로, 특히 임베디드 애플리케이션 및 소규모 애플리케이션용 데이터베이스에 사용됩니다. 이 Lightsail 이미지는 Windows Server 2016의 기본 OS에서 실행됩니다.

이 블루프린트는 Lightsail IPv6 전용 인스턴스 플랜과 호환됩니다.

SQL Server 2016 Express에 대해 자세히 알아보세요.

CMS 애플리케이션

Lightsail에서 사용할 수 있는 콘텐츠 관리 시스템(CMS) 애플리케이션은 다음과 같습니다.

WordPress

WordPress 블루프린트는 PHP, MariaDB, phpMyAdmin 및 WordPress로 완전한 프로덕션 환경을 제공합니다.는 업계 모범 사례를 사용하여 블루프린트를 안전하고 up-to-date Lightsail 패키징합니다.

Lightsail WordPress에서 시작된 인스턴스에는 기본적으로 인스턴스 메타데이터 서비스 버전 2(IMDSv2)가 적용됩니다. 자세한 내용은 인스턴스 메타데이터 서비스 버전 2 작동 방식 단원을 참조하십시오.

에서 WordPress 배포 및 관리 Lightsail

WordPress 웹 사이트에서 WordPress 스택에 대해 자세히 알아봅니다. WordPress

Bitnami 인증 WordPress

Bitnami WordPress는 Lightsail에서 WordPress를 실행할 수 있도록 미리 구성되어 바로 사용 가능한 이미지입니다. WordPress는 블로그와 웹사이트를 빌드하기 위해 널리 사용되는 웹 게시 플랫폼입니다. 다양한 테마, 확장 프로그램, 플러그인 및 위젯을 사용하여 WordPress를 맞춤형으로 구성할 수 있습니다.

WordPress는 완전한 테마 시스템을 갖추고 있어 몇 번의 클릭으로 사이트의 모양과 느낌을 바꿀 수 있습니다. 기존의 무료 또는 유료 WordPress 테마도 사용할 수 있습니다. WordPress는 World Wide Web Consortium (W3C)의 표준을 완벽히 준수합니다.

에서 WordPress 배포 및 관리 Lightsail

Bitnami 웹 사이트에서 WordPress에 대해 자세히 알아보세요.

Bitnami 인증 WordPress Multisite

WordPress Multisite를 사용하면 관리자가 동일한 WordPress 인스턴스에서 여러 웹 사이트를 호스팅하고 관리할 수 있습니다. 이러한 웹 사이트에는 모두 고유의 도메인 이름이 있을 수 있으며 소유자가 해당 웹 사이트를 사용자 지정할 수 있는 반면, 테마 및 플러그인과 같은 공유 자산은 서버 관리자가 제공합니다. 모든 사이트에 대한 업데이트가 한 번에 푸시되므로 모든 사이트에서 안전성과 보안을 유지할 수 있습니다.

WordPress Multisite는 많은 사람들이 자신의 웹 사이트를 호스팅하고 중앙 관리자에게 전반적인 제어 권한을 부여해야 하는 대학, 기업 및 대행사와 같은 조직에 적합합니다.

Lightsail에서 WordPress 멀티사이트 설정

Bitnami 웹 사이트에서 WordPress 멀티사이트에 대해 자세히 알아보세요.

cPanel & WebHost 관리자(WHM)

cPanel & WHM은 간단한 그래픽 사용자 인터페이스를 통해 웹 호스팅 작업을 자동화하는 기능을 제공하는 Linux OS용 도구 모음입니다. 목표는 사용자가 서버를, 고객이 웹 사이트를 더 쉽게 관리하도록 지원하는 것입니다.

Lightsail에서 cPanel 및 WHM을 사용하여 웹 사이트, 이메일 및 서비스 호스팅

cPanel 웹 사이트에서 cPanel 및 WHM에 대해 자세히 알아보세요.

Bitnami에서 패키징한 PrestaShop

PrestaShop은 세계에서 특히 많이 사용되는 전자 상거래 솔루션 중 하나입니다. 이 소프트웨어는 무료 오픈 소스 소프트웨어이며, 1백만 명 이상의 활성 멤버가 있습니다. 이 제품은 사전 구성된 테마를 통해 온라인 상점을 신속하게 시작하고 실행할 수 있도록 설계되었으며 Live Configurator와 함께 판매를 시작하여 손쉽게 사이트 디자인을 사용자 지정할 수 있습니다. PrestaShop은 다중 스토어 지원, 사용자 지정 가능한 URL, 다중 결제 게이트웨이 옵션(PayPal 및 Stripe 포함), Amazon, eBay, Facebook 등과 마켓플레이스 통합을 제공합니다.

Lightsail에서 PrestaShop 웹 사이트 설정

PrestaShop 웹 사이트에서 PrestaShop에 대해 자세히 알아보세요.

Bitnami에서 패키징한 Ghost

Ghost는 개인 블로그에서 주요 뉴스 웹 사이트에 이르기까지 모든 것에 적합한 출판 플랫폼입니다. Node.js를 기반으로 구축된 최신 기술 스택은 콘텐츠 제작자의 사용 편의성을 유지하면서 다른 애플리케이션 및 도구와 통합하고 싶어하는 개발자가 다양한 용도로 유연하게 사용할 수 있습니다.

Lightsail에 Ghost 웹 사이트 배포

Bitnami 웹 사이트에서 Bitnami Ghost에 대해 자세히 알아보세요.

Bitnami에서 패키징한 Joomla!

Bitnami Joomla!는 Lightsail에서 Joomla!를 실행할 수 있도록 미리 구성되어 바로 사용 가능한 이미지입니다. Joomla!는 다양한 웹사이트나 포털을 빌드하기 위해 사용할 수 있는 CMS입니다. 개인, 법인, 소규모 사업체, 비영리 기관 및 기타 조직의 웹사이트가 포함됩니다.

Joomla!는 사용자가 개인 옵션을 구성할 수 있는 등록 시스템도 갖추고 있습니다. 인증은 사용자 관리의 중요한 부분이며, Joomla!는 LDAP, OpenID 등의 여러 프로토콜을 지원합니다. Joomla!는 다양한 언어를 지원하며 웹사이트 및 관리 패널에 이런 언어를 사용하기 위한 지침을 제공합니다. 또한, 배너 관리자를 사용하면 사이트에서 배너를 쉽게 설정하고 관리할 수 있습니다. 인쇄 번호 설정, 특수 URL 등을 포함한 지표를 추적할 수 있습니다.

Lightsail에서 Joomla! 시작하기

Bitnami 웹 사이트에서 Joomla!에 대해 자세히 알아보세요.

Bitnami에서 패키징한 Drupal

Bitnami Drupal은 Lightsail에서 Drupal을 실행할 수 있도록 미리 구성되어 바로 사용 가능한 이미지입니다. Drupal은 사용자가 콘텐츠를 쉽게 게시, 관리 및 구성하는 데 도움이 되는 콘텐츠 관리 플랫폼입니다. 커뮤니티 웹 포털, 토론 사이트, 회사 웹사이트 등에 사용됩니다. 모듈 플러그인을 통해

Drupal을 쉽게 확장할 수 있습니다. Drupal은 높은 성능을 목적으로 빌드되어 많은 서버로 확장 가능하고 REST, JSON, SOAP 및 기타 형식과 손쉽게 통합됩니다.

Drupal용으로 수천 가지의 추가 기능 모듈과 디자인이 무료로 제공됩니다. 또한, Drupal은 여러 언어로 제공됩니다.

Lightsail에서 Drupal 웹 사이트 설정 및 사용자 지정

Bitnami 웹 사이트에서 Drupal에 대해 자세히 알아보세요.

애플리케이션 스택 및 서버

Lightsail 에는 다양한 개발 프로젝트를 위한 여러 애플리케이션 스택과 서버가 있습니다. 각 이미지는 Linux/Unix를 기본 운영 체제로 사용합니다.

OpenClaw

OpenClaw는 오픈 소스 자율 AI 에이전트(이전 Clawdbot/Moltbot)입니다. 자체 서버의 백그라운드에서 지속적으로 실행되어 Slack, Telegram, WhatsApp 및 Discord와 같은 메시징 플랫폼에 기본 인터페이스로 연결됩니다. OpenClaw는 선제적 작업 실행, 다중 채널 통합, 코드를 실행하고, 파일을 관리하고, 웹을 탐색하는 기능을 제공합니다.

Lightsail OpenClaw에서 시작된 인스턴스에는 기본적으로 인스턴스 메타데이터 서비스 버전 2(IMDSv2)가 적용됩니다. 자세한 내용은 인스턴스 메타데이터 서비스 버전 2 작동 방식 단원을 참조하십시오.

에서 OpenClaw 시작하기 Lightsail

OpenClaw 웹 사이트에서 OpenClaw에 대해 자세히 알아보세요.

Bitnami에서 패키징한 LAMP 스택(PHP 8)

Bitnami LAMP 스택은 PHP 애플리케이션의 개발과 배포를 단순화합니다. 이 스택에는 실행 준비가 완료된 Apache, MySQL, PHP 및 phpMyAdmin 버전뿐만 아니라, 이런 구성 요소를 각각 실행하는 데 필요한 다른 소프트웨어도 포함되어 있습니다. Bitnami LAMP 스택은 완전히 통합 및 구성되어 있으므로, Lightsail에서 인스턴스를 생성하자마자 애플리케이션 개발을 시작할 수 있습니다. 번들로 제공되는 각 구성 요소를 위해 안정적인 최신 릴리스에 언제든 액세스할 수 있도록, Bitnami LAMP 스택은 정기적으로 업데이트됩니다.

이 블루프린트는 Lightsail IPv6 전용 인스턴스 플랜과 호환됩니다.

에서 LAMP 스택 배포 및 관리 Lightsail

Bitnami 웹 사이트에서 Bitnami LAMP 스택에 대해 자세히 알아보세요.

램프

LAMP 블루프린트는 Linux에서 PHP, Apache 및 MariaDB를 사용하는 완전한 프로덕션 환경을 제공합니다. 이 블루프린트에는 phpMyAdmin, PHP 코어 모듈 및 Composer도 포함됩니다.

Lightsail LAMP에서 시작된 인스턴스에는 기본적으로 인스턴스 메타데이터 서비스 버전 2(IMDSv2)가 적용됩니다. 자세한 내용은 인스턴스 메타데이터 서비스 버전 2 작동 방식 단원을 참조하십시오.

에서 LAMP 스택 배포 및 관리 Lightsail

Bitnami에서 패키징한 Django

Django는 신속한 개발과 깔끔하고 실용적인 설계를 장려하는 고급 Python Web 프레임워크입니다. Python은 다양한 종류의 소프트웨어 개발에 사용할 수 있는 동적인 객체 지향 프로그래밍 언어이다. Bitnami Django 스택은 Django와 런타임 종속성의 배포를 획기적으로 단순화하고 바로 실행할 수 있는 Python, Django, MySQL 및 Apache 버전을 포함합니다.

Bitnami 웹 사이트에서 Bitnami Django 스택에 대해 자세히 알아보세요.

Bitnami에서 패키징한 Node.js

Bitnami Node.js는 Lightsail에서 Node.js를 실행할 수 있도록 미리 구성되어 바로 사용 가능한 이미지입니다. Node.js는 빠르고 확장 가능한 네트워크 애플리케이션을 쉽게 만들기 위해 Chrome의 JavaScript 런타임을 기반으로 빌드된 플랫폼입니다. 가볍고 효율적으로 만들어주는 이벤트 기반의 비차단 I/O 모델을 사용합니다. Node.js는 데이터 집약적인 실시간 애플리케이션에 매우 적합합니다.

에서 Node.js 스택 배포 및 관리 Lightsail

Bitnami 웹 사이트에서 Node.js 스택에 대해 자세히 알아보세요.

Node.js

Node.js 블루프린트는 MariaDB 및 Node.js가 포함된 완전한 프로덕션 환경을 제공합니다.는 업계 모범 사례를 사용하여 블루프린트를 안전하고 up-to-date Lightsail 패키징합니다.

Lightsail Node.js에서 시작된 인스턴스에는 기본적으로 인스턴스 메타데이터 서비스 버전 2(IMDSv2)가 적용됩니다. 자세한 내용은 인스턴스 메타데이터 서비스 버전 2 작동 방식 단원을 참조하십시오.

에서 Node.js 스택 배포 및 관리 Lightsail

Node.js 웹 사이트에서 Node.js 스택에 대해 자세히 알아봅니다.

Bitnami에서 패키징한 MEAN 스택

Bitnami MEAN 스택은 한 번의 클릭으로 배포할 수 있는 MongoDB 및 Node.js를 위한 완전한 개발 환경을 제공합니다. 이 스택에는 MongoDB, Express, Angular, Node.js, Git, PHP 및 RockMongo의 안정적인 최신 릴리스가 포함됩니다.

이 블루프린트는 Lightsail IPv6 전용 인스턴스 플랜과 호환됩니다.

Bitnami 웹 사이트에서 MEAN 스택에 대해 자세히 알아보세요.

Bitnami에서 패키징한 GitLab CE

Bitnami GitLab Community Edition(CE)은 Lightsail에서 GitLab을 실행할 수 있도록 미리 구성되어 바로 사용 가능한 이미지입니다. GitLab은 자체적으로 호스트된 Git 관리 소프트웨어로서 빠르고 안전하며 Ruby on Rails를 기반으로 합니다. GitLab CI(역시 포함되어 있음)는 Git 및 GitLab과 긴밀히 통합된 오픈소스 CI(Continuous Integration) 서버입니다.

GitLab을 사용하면 자체 서버에 코드를 안전하게 보호하고 리포지토리, 사용자 및 액세스 권한을 관리할 수 있습니다. GitLab은 독립적으로 실행되므로 설치 프로그램을 다른 서버로 쉽게 복제하거나 이동할 수 있습니다.

Lightsail에서 GitLab CE 인스턴스 설정 및 구성

Bitnami 웹 사이트에서 GitLab 스택에 대해 자세히 알아보세요.

Bitnami에서 패키징한 Nginx(LEMP 스택)

Bitnami NGINX 스택은 한 번의 클릭으로 시작할 수 있는 완전한 PHP, MySQL 및 NGINX 개발 환경을 제공합니다. 또한, phpMyAdmin, SQLite, ImageMagick, FastCGI, Memcache, GD, CURL, PEAR, PECL 및 기타 구성 요소를 번들로 묶어 제공합니다.

NGINX는 비동기 서버로서 주요 장점은 확장성입니다. NGINX 스택은 LEMP(Linux, NGINX, MySQL, PHP)라고도 알려져 있습니다.

Lightsail에서 Nginx 웹 서버 배포 및 관리

Bitnami 웹 사이트에서 NGINX 스택에 대해 자세히 알아보세요.

Nginx

Nginx 블루프린트는 PHP, MariaDB, phpMyAdmin 및 NGINX로 완전한 프로덕션 환경을 제공합니다.는 업계 모범 사례를 사용하여 블루프린트를 안전하고 up-to-date Lightsail 패키징합니다.

Lightsail Nginx에서 시작된 인스턴스에는 기본적으로 인스턴스 메타데이터 서비스 버전 2(IMDSv2)가 적용됩니다. 자세한 내용은 인스턴스 메타데이터 서비스 버전 2 작동 방식 단원을 참조하십시오.

이 블루프린트는 Lightsail IPv6 전용 인스턴스 플랜과 호환됩니다.

Lightsail에서 Nginx 웹 서버 배포 및 관리

NGINX 웹 사이트에서 NGINX 스택에 대해 자세히 알아보세요.

Plesk Hosting Stack on Ubuntu, Plesk Hosting Stack on Ubuntu(BYOL)

> **Important**
>
> 2024년 8월 1일부로 Plesk는 유료 라이선스 모델로 전환되었습니다. 다음 라이선스 동작은 Plesk를 실행하는 Lightsail 인스턴스에 적용됩니다.
>
> - 2025년 2월 1일부터, 이전 Plesk Hosting Stack on Ubuntu 블루프린트를 사용하는 모든 인스턴스에 대해 유료 라이선스가 필요합니다.
> - Plesk Hosting Stack on Ubuntu(BYOL) 블루프린트로 시작된 인스턴스에는 30일 평가판 라이선스가 있습니다. 30일이 지나면 Plesk에서 라이선스를 구매해야 Plesk 애플리케이션을 계속 사용할 수 있습니다.
>
> 자세한 내용은 Plesk 라이선스 구매를 참조하세요.

Plesk로 구동되는 호스팅 스택을 사용하여 Lightsail과 AWS에서 웹 사이트 및 애플리케이션을 구축하고, 보호하고, 실행할 수 있습니다. 여기에는 모든 웹 기반 서버 관리 및 보안 도구 외에도 그래픽 사용자 인터페이스의 WordPress 자동화가 포함됩니다. 이 기능은 웹 전문가의 작업을 간소화하며 고객에게 필요한 확장성, 보안 및 성능을 제공합니다.

Plesk를 설정하고 구성합니다.

Plesk 웹 사이트의 Plesk 스택에 대해 자세히 알아보세요.

Ruby on Rails

Ruby on Rails 블루프린트는 Amazon Linux 2023의 Rails로 사전 구성되어 있으므로 수동 프레임워크 설치 및 설정이 필요하지 않습니다. Ruby on Rails 블루프린트를 사용하면에서 최신 웹 애플리케이션을 빌드하기 위한 강력하고 확장 가능하며 비용 효율적인 솔루션을 배포할 수 있습니다 Lightsail.

에서 Ruby on Rails 설정 Lightsail

Ruby on Rails 및 Amazon Linux 2023에 대해 자세히 알아보세요.

전자 상거래 애플리케이션

Lightsail에는 현재 Magento라는 전자 상거래 애플리케이션 이미지가 하나 있습니다. 이 Magento 이미지는 Linux/Unix(Ubuntu)를 기본 운영 체제로 사용합니다.

Bitnami에서 패키징한 Magento

Bitnami Magento는 Lightsail에서 Magento를 실행할 수 있도록 미리 구성되어 바로 사용 가능한 이미지입니다. Magento를 사용하여 매력적이고 응답성이 뛰어나며 안전한 사이트를 빌드할 수 있습니다. Magento는 트랜잭션 옵션, 멀티스토어 기능, 충성도 프로그램, 제품 분류, 쇼핑객 필터링, 프로모션 규칙 등을 포함하는 기능이 풍부하고 유연한 전자 상거래 솔루션입니다.

Magento를 사용하여 브랜드를 반영하는 고도의 맞춤형 전자 상거래 사이트를 만들 수 있습니다. Magento는 비즈니스 운영과 통합되므로 비즈니스 요구 사항에 따라 전자 상거래 사이트를 관리할 수 있습니다.

Lightsail에 Magento를 설정 및 구성

Bitnami 웹 사이트의 Magento 스택에 대해 자세히 알아보세요.

프로젝트 관리 애플리케이션

Lightsail에는 현재 Redmine이라는 프로젝트 관리 애플리케이션 이미지가 하나 있습니다. 이 이미지는 Linux/Unix(Ubuntu)를 기본 운영 체제로 사용합니다.

Bitnami에서 패키징한 Redmine

Bitnami Redmine은 Lightsail에서 Redmine을 실행할 수 있도록 미리 구성되어 바로 사용 가능한 이미지입니다. Redmine은 유연한 프로젝트 관리 웹 애플리케이션입니다. Redmine은 여러 가지 프로젝트 지원, 역할 기반 액세스 제어, Gantt 차트와 달력, 뉴스, 문서 및 파일의 관리, 프로젝트별 위키와 포럼, SCM 통합 등을 포함합니다.

이 블루프린트는 Lightsail IPv6 전용 인스턴스 플랜과 호환됩니다.

Lightsail에서 Redmine 인스턴스 구성 및 보호

Bitnami 웹 사이트의 Redmine 스택에 대해 자세히 알아보세요.

Lightsail 인스턴스 번들

Lightsail은 서로 다른 워크로드 요구 사항을 충족하기 위해 다양한 인스턴스 번들(인스턴스 플랜이라고도 함)을 제공합니다. 각 번들은 컴퓨팅 성능(vCPU), 메모리(RAM), 스토리지 및 데이터 전송 허용량을 제공합니다. 번들은 온디맨드 시간 요율에 따라 요금이 청구되므로 사용하는 만큼만 결제하면 됩니다. 고객이 이용하는 모든 번들에 대해 월간 최대 요금까지 고정된 시간당 가격으로 계산해 요금이 청구됩니다.

범용 계획은 웹 및 애플리케이션 서버, 가상 데스크톱, 마이크로서비스, 데이터베이스 및 배치 처리와 같은 대부분의 애플리케이션에 적합한 컴퓨팅, 메모리 및 네트워킹 리소스를 균형 있게 혼합하여 제공합니다. 메모리 최적화 계획은 인 메모리 캐싱, 실시간 빅 데이터 분석 또는 고성능 데이터베이스와 같은 워크로드에 도움이 되는 더 높은 memory-to-CPU 비율을 제공합니다.

각 번들에는 퍼블릭 IP 주소가 포함됩니다. 번들은 퍼블릭 IPv4 주소, IPv6 주소가 있는 퍼블릭 IPv4 주소(이중 스택이라고 함) 또는 IPv6 주소만(IPv6 전용이라고 함)을 제공합니다. 번들 가격은 퍼블릭 IPv4 주소를 포함하는지 여부에 따라 다릅니다.

주제

- Linux/Unix 번들(퍼블릭 IPv4 주소 포함)
- Linux/Unix 번들(IPv6 전용)
- Windows 번들(퍼블릭 IPv4 주소 포함)
- Windows 번들(IPv6 전용)

Linux/Unix 번들(퍼블릭 IPv4 주소 포함)

다음 표에는 IPv4 주소를 포함하는 Linux/Unix 인스턴스 번들의 사양이 나열되어 있습니다.

번들 이름	가격 (USD/월)	vCPU	Memory	스토리지	데이터 전송 *
퍼블릭 IPv4를 사용하는 Nano-0.5GB Linux	5.00 USD	2	0.5GB	20GB	1TB
퍼블릭 IPv4를 사용하는 Micro-1GB Linux	7.00 USD	2	1GB	40GB	2TB

번들 이름	가격 (USD/월)	vCPU	Memory	스토리지	데이터 전송 *
퍼블릭 IPv4를 사용하는 Small-2GB Linux	12.00 USD	2	2GB	60GB	3TB
퍼블릭 IPv4가 있는 Medium-4GB Linux	24.00 USD	2	4GB	80GB	4TB
퍼블릭 IPv4를 사용하는 Large-8GB Linux	44.00 USD	2	8GB	160GB	5TB
퍼블릭 IPv4를 사용하는 Xlarge-16GB Linux	84.00 USD	4	16 GB	320GB	6TB
퍼블릭 IPv4를 사용하는 2Xlarge-32GB Linux	164.00 USD	8	32GB	640GB	7TB
퍼블릭 IPv4를 사용하는 4Xlarge-64GB Linux	384.00 USD	16	64GB	1,280GB	8TB
퍼블릭 IPv4를 사용하는 8Xlarge-128GB Linux	884.00 USD	32	128GB	1,280GB	9TB
퍼블릭 IPv4를 사용하는 12Xlarge-192GB Linux	1,324.00 USD	48	192GB	1,280GB	10TB
퍼블릭 IPv4를 사용하는 16Xlarge-256GB Linux	1,764.00 USD	64	256GB	1,280GB	10TB
퍼블릭 IPv4를 사용하는 메모리 최적화 Large-16GB Linux	74.00 USD	2	16 GB	160GB	5TB
퍼블릭 IPv4를 사용하는 메모리 최적화 Xlarge-32GB Linux	144.00 USD	4	32GB	320GB	6TB

번들 이름	가격 (USD/월)	vCPU	Memory	스토리지	데이터 전송 *
퍼블릭 IPv4를 사용하는 메모리 최적화 2Xlarge-64GB Linux	294.00 USD	8	64GB	640GB	7TB
퍼블릭 IPv4를 사용하는 메모리 최적화 4Xlarge-128GB Linux	584.00 USD	16	128GB	1,280GB	8TB
퍼블릭 IPv4를 사용하는 메모리 최적화 8Xlarge-256GB Linux	1,174.00 USD	32	256GB	1,280GB	9TB
퍼블릭 IPv4를 사용하는 메모리 최적화 12Xlarge-384GB Linux	1,764.00 USD	48	384GB	1,280GB	10TB
퍼블릭 IPv4를 사용하는 메모리 최적화 16Xlarge-512GB Linux	2,344.00 USD	64	512GB	1,280GB	10TB

* 데이터 전송 허용량은 리전에 따라 달라질 수 있습니다. 자세한 내용은 인스턴스의 데이터 전송 허용량은 AWS 리전에 따라 어떻게 달라지나요? 단원을 참조하십시오.

Linux/Unix 번들(IPv6 전용)

다음 표에는 IPv6 주소만 포함하는 Linux/Unix 인스턴스 번들의 사양이 나열되어 있습니다.

번들 이름	가격 (USD/월)	vCPU	Memory	스토리지	데이터 전송 *
Nano-0.5GB Linux IPv6-only	3.50 USD	2	0.5GB	20GB	1TB
Micro-1GB Linux IPv6-only	5.00 USD	2	1GB	40GB	2TB

번들 이름	가격 (USD/월)	vCPU	Memory	스토리지	데이터 전송 *
Small-2GB Linux IPv6-only	10.00 USD	2	2GB	60GB	3TB
Medium-4GB Linux IPv6-only	20.00 USD	2	4GB	80GB	4TB
Large-8GB Linux IPv6-only	40.00 USD	2	8GB	160GB	5TB
Xlarge-16GB Linux IPv6-only	80.00 USD	4	16 GB	320GB	6TB
2Xlarge-32GB Linux IPv6-only	160.00 USD	8	32GB	640GB	7TB
4Xlarge-64GB Linux IPv6-only	380.00 USD	16	64GB	1,280GB	8TB
8Xlarge-128GB Linux IPv6-only	880.00 USD	32	128GB	1,280GB	9TB
12Xlarge-192GB Linux IPv6-only	1,320.00 USD	48	192GB	1,280GB	10TB
16Xlarge-256GB Linux IPv6-only	1,760.00 USD	64	256GB	1,280GB	10TB
메모리 최적화 Large-16GB Linux IPv6-only	70.00 USD	2	16 GB	160GB	5TB
메모리 최적화 Xlarge-32GB Linux IPv6-only	140.00 USD	4	32GB	320GB	6TB

번들 이름	가격 (USD/월)	vCPU	Memory	스토리지	데이터 전송 *
메모리 최적화 2Xlarge-64GB Linux IPv6-only	290.00 USD	8	64GB	640GB	7TB
메모리 최적화 4Xlarge-128GB Linux IPv6-only	580.00 USD	16	128GB	1,280GB	8TB
메모리 최적화 8Xlarge-256GB Linux IPv6-only	1170.00 USD	32	256GB	1,280GB	9TB
메모리 최적화 12Xlarge-384GB Linux IPv6-only	1,760.00 USD	48	384GB	1,280GB	10TB
메모리 최적화 16Xlarge-512GB Linux IPv6-only	2,340.00 USD	64	512GB	1,280GB	10TB

* 데이터 전송 허용량은 리전에 따라 달라질 수 있습니다. 자세한 내용은 인스턴스의 데이터 전송 허용량은 AWS 리전에 따라 어떻게 달라지나요? 단원을 참조하십시오.

Windows 번들(퍼블릭 IPv4 주소 포함)

다음 표에는 IPv4 주소를 포함하는 Windows 인스턴스 번들의 사양이 나열되어 있습니다.

번들 이름	가격 (USD/월)	vCPU	Memory	스토리지	데이터 전송 *
퍼블릭 IPv4가 있는 Nano-0.5GB Windows	9.50 USD	2	0.5GB	30GB	1TB
퍼블릭 IPv4가 있는 Micro-1GB Windows	14.00 USD	2	1GB	40GB	2TB

번들 이름	가격 (USD/월)	vCPU	Memory	스토리지	데이터 전송 *
퍼블릭 IPv4가 있는 Small-2GB Windows	22.00 USD	2	2GB	60GB	3TB
퍼블릭 IPv4Mediu m-4GB Windows	44.00 USD	2	4GB	80GB	4TB
퍼블릭 IPv4가 있는 Large-8GB Windows	74.00 USD	2	8GB	160GB	5TB
퍼블릭 IPv4가 있는 Xlarge-16GB Windows	124.00 USD	4	16 GB	320GB	6TB
퍼블릭 IPv4가 있는 2Xlarge-32GB Windows	244.00 USD	8	32GB	640GB	7TB
퍼블릭 IPv4가 있는 4Xlarge-64GB Windows	574.00 USD	16	64GB	1,280GB	8TB
퍼블릭 IPv4가 있 는 8Xlarge-128GB Windows	1,254.00 USD	32	128GB	1,280GB	9TB
퍼블릭 IPv4가 있 는 12Xlarge-192GB Windows	1,884.00 USD	48	192GB	1,280GB	10TB
퍼블릭 IPv4가 있 는 16Xlarge-256GB Windows	2,504.00 USD	64	256GB	1,280GB	10TB
퍼블릭 IPv4를 사용 하는 메모리 최적화 Large-16GB Windows	134.00 USD	2	16 GB	160GB	5TB

번들 이름	가격 (USD/월)	vCPU	Memory	스토리지	데이터 전송 *
퍼블릭 IPv4를 사용하는 메모리 최적화 Xlarge-32GB Windows	264.00 USD	4	32GB	320GB	6TB
퍼블릭 IPv4를 사용하는 메모리 최적화 2Xlarge-64GB Windows	524.00 USD	8	64GB	640GB	7TB
퍼블릭 IPv4를 사용하는 메모리 최적화 4Xlarge-128GB Windows	1044.00 USD	16	128GB	1,280GB	8TB
퍼블릭 IPv4를 사용하는 메모리 최적화 8Xlarge-256GB Windows	2,104.00 USD	32	256GB	1,280GB	9TB
퍼블릭 IPv4를 사용하는 메모리 최적화 12Xlarge-384GB Windows	3,164.00 USD	48	384GB	1,280GB	10TB
퍼블릭 IPv4를 사용하는 메모리 최적화 16Xlarge-512GB Windows	4,204.00 USD	64	512GB	1,280GB	10TB

* 데이터 전송 허용량은 리전에 따라 달라질 수 있습니다. 자세한 내용은 인스턴스의 데이터 전송 허용량은 AWS 리전에 따라 어떻게 달라지나요? 단원을 참조하십시오.

Windows 번들(IPv6 전용)

다음 표에는 IPv6 주소만 포함하는 Windows 인스턴스 번들의 사양이 나열되어 있습니다.

번들 이름	가격 (USD/월)	vCPU	Memory	스토리지	데이터 전송 *
Nano-0.5GB Windows IPv6-only	8.00 USD	2	0.5GB	30GB	1TB
Micro-1GB Windows IPv6-only	12.00 USD	2	1GB	40GB	2TB
Small-2GB Windows IPv6-only	20.00 USD	2	2GB	60GB	3TB
Medium-4GB Windows IPv6-only	40.00 USD	2	4GB	80GB	4TB
Large-8GB Windows IPv6-only	70.00 USD	2	8GB	160GB	5TB
Xlarge-16GB Windows IPv6-only	120.00 USD	4	16 GB	320GB	6TB
2Xlarge-32GB Windows IPv6-only	240.00 USD	8	32GB	640GB	7TB
4Xlarge-64GB Windows IPv6-only	570.00 USD	16	64GB	1,280GB	8TB
8Xlarge-128GB Windows IPv6-only	1,250.00 USD	32	128GB	1,280GB	9TB
12Xlarge-192GB Windows IPv6-only	1,880.00 USD	48	192GB	1,280GB	10TB
16Xlarge-256GB Windows IPv6-only	2,500.00 USD	64	256GB	1,280GB	10TB
메모리 최적화 Large-16GB Windows IPv6-only	130.00 USD	2	16 GB	160GB	5TB

번들 이름	가격 (USD/월)	vCPU	Memory	스토리지	데이터 전송 *
메모리 최적화 Xlarge-32GB Windows IPv6-only	260.00 USD	4	32GB	320GB	6TB
메모리 최적화 2Xlarge-64GB Windows IPv6-only	520.00 USD	8	64GB	640GB	7TB
메모리 최적화 4Xlarge-128GB Windows IPv6-only	1,040.00 USD	16	128GB	1,280GB	8TB
메모리 최적화 8Xlarge-256GB Windows IPv6-only	2,100.00 USD	32	256GB	1,280GB	9TB
메모리 최적화 12Xlarge-384GB Windows IPv6-only	3,160.00 USD	48	384GB	1,280GB	10TB
메모리 최적화 16Xlarge-512GB Windows IPv6-only	4,200.00 USD	64	512GB	1,280GB	10TB

* 데이터 전송 허용량은 리전에 따라 달라질 수 있습니다. 자세한 내용은 인스턴스의 데이터 전송 허용량은 AWS 리전에 따라 어떻게 달라지나요? 단원을 참조하십시오.

Lightsail의 방화벽을 사용하여 인스턴스 트래픽 제어

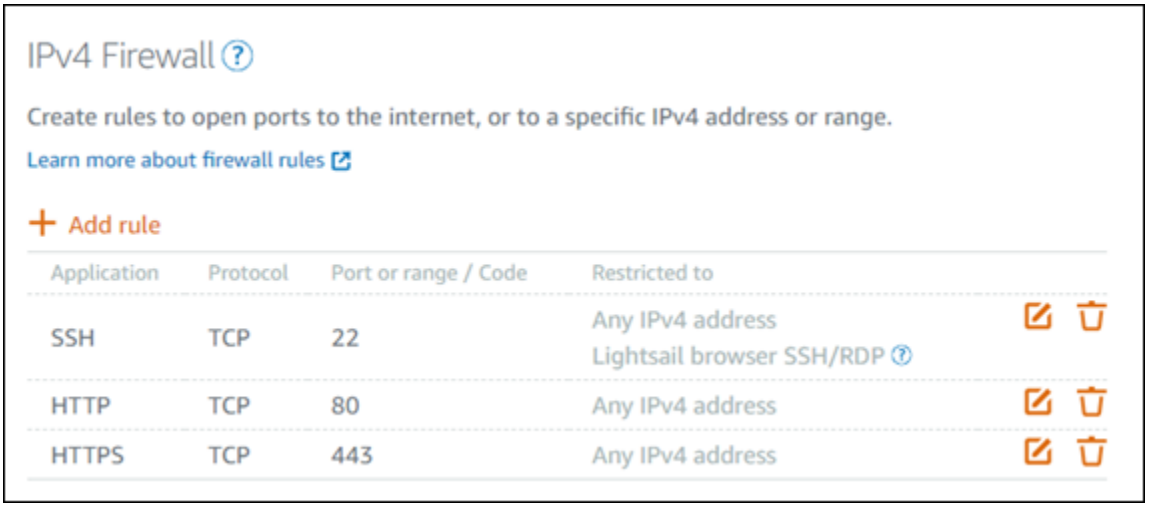
Amazon Lightsail 콘솔의 방화벽은 퍼블릭 IP 주소를 통해 인스턴스에 연결할 수 있는 트래픽을 제어하는 가상 방화벽 역할을 합니다. Lightsail에서 생성하는 각 인스턴스에는 IPv4 주소용과 IPv6 주소용의 방화벽 2개가 있습니다. 각 방화벽에는 인스턴스로 들어오는 트래픽을 필터링하는 규칙 집합이 포함되어 있습니다. 2개의 방화벽은 서로 독립적이므로, IPv4 및 IPv6 각각에 대해 별도로 방화벽 규칙을 구성

해야 합니다. 트래픽을 허용하거나 제한하는 규칙을 추가 및 삭제하여 언제든지 인스턴스의 방화벽을 편집합니다.

Lightsail 방화벽

각 Lightsail 인스턴스에는 IPv4 주소용과 IPv6 주소용의 방화벽 2개가 있습니다. Lightsail 인스턴스에 들어오고 나가는 모든 인터넷 트래픽은 방화벽을 통과합니다. 인스턴스의 방화벽은 인스턴스로 유입될 수 있는 인터넷 트래픽을 제어합니다. 그러나 방화벽은 유출되는 트래픽을 제어하지 않고 모든 아웃바운드 트래픽을 허용합니다. 수신 트래픽을 허용하거나 제한하는 규칙을 추가 및 삭제하여 언제든지 인스턴스의 방화벽을 편집할 수 있습니다. 2개의 방화벽은 서로 독립적이므로, IPv4 및 IPv6 각각에 대해 별도로 방화벽 규칙을 구성해야 합니다.

방화벽 규칙은 항상 허용적입니다. 따라서 액세스를 거부하는 규칙을 생성할 수 없습니다. 인스턴스의 방화벽에 규칙을 추가하여 트래픽이 인스턴스에 도달하도록 허용합니다. 인스턴스의 방화벽에 규칙을 추가할 때 다음 예제(IPv4용)와 같이 사용할 프로토콜, 열려는 포트 및 인스턴스에 연결할 수 있는 IPv4 및 IPv6 주소를 지정합니다. 인스턴스에서 사용하려는 서비스에 따라 프로토콜 및 포트 범위를 지정하는 사전 설정인 애플리케이션 계층 프로토콜 유형을 지정할 수도 있습니다.



> **Important**
>
> 방화벽 규칙은 인스턴스의 퍼블릭 IP 주소를 통해 흐르는 트래픽에만 영향을 미칩니다. 동일한 AWS 리전리전에 있는 계정의 Lightsail 리소스 또는 동일한 AWS 리전리전에 있는 피어링된 Virtual Private Cloud(VPC)의 리소스로부터 시작되어 인스턴스의 프라이빗 IP 주소를 통해 흐르는 트래픽에 영향을 미치지 않습니다.

방화벽 규칙 및 구성 가능한 파라미터는 이 안내서의 다음 몇 섹션에서 설명합니다.

방화벽 규칙 생성

클라이언트가 인스턴스 또는 인스턴스에서 실행 중인 애플리케이션과의 연결을 설정할 수 있도록 방화벽 규칙을 생성합니다. 예를 들어 모든 웹 브라우저가 인스턴스의 WordPress 애플리케이션에 연결되도록 하려면 IP 주소에서 포트 80을 통해 TCP(Transmission Control Protocol)를 활성화하는 방화벽 규칙을 구성합니다. 이 규칙이 인스턴스의 방화벽에 이미 구성되어 있는 경우 이를 삭제하여 웹 브라우저가 인스턴스의 WordPress 애플리케이션에 연결할 수 없도록 차단할 수 있습니다.

Important

Lightsail 콘솔을 사용하여 한 번에 최대 30개의 소스 IP 주소를 추가할 수 있습니다. 한 번에 최대 60개의 IP 주소를 추가하려면 Lightsail API, AWS Command Line Interface (AWS CLI) 또는 SDK를 AWS 사용합니다. 이 할당량은 IPv4 및 IPv6 규칙에 개별적으로 적용됩니다. 예를 들어 방화벽은 IPv4 트래픽에 대한 인바운드 규칙 60개 및 IPv6 트래픽에 대한 인바운드 규칙 60개를 보유할 수 있습니다. 개별 IP 주소는 CIDR 범위로 통합하는 것이 좋습니다. 자세한 내용은 이 설명서의 소스 IP 주소 지정 섹션을 참조하세요.

연결을 설정해야 하는 컴퓨터의 IP 주소에서만 포트 22를 통한 TCP를 활성화하는 방화벽 규칙을 구성하여 SSH 클라이언트가 인스턴스에 연결하고 서버에서 관리 작업을 수행하도록 설정할 수도 있습니다. 이 경우 인스턴스에 대한 SSH 연결을 설정하는 IP 주소를 허용하지 않을 것입니다. 그렇게 하면 인스턴스에 보안 위험이 발생할 수 있습니다.

Note

이 섹션에서 설명하는 방화벽 규칙 예제는 기본적으로 인스턴스의 방화벽에 존재할 수 있습니다. 자세한 내용은 이 안내서의 뒷부분에 있는 기본 방화벽 규칙을 참조하십시오.

특정 포트에 대한 규칙이 여러 개 있는 경우 최대 허용 규칙을 적용합니다. 예를 들어 IP 주소 192.0.2.1에서 TCP 포트 22(SSH)에 대한 액세스를 허용하는 규칙을 추가하는 경우. 그러면 모든 사람으로부터 TCP 포트 22에 대한 액세스를 허용하는 다른 규칙을 추가합니다. 결과적으로 모든 사람이 TCP 포트 22에 액세스할 수 있습니다.

프로토콜 지정

프로토콜은 두 컴퓨터 간에 데이터가 전송되는 형식입니다. Lightsail을 사용하면 방화벽 규칙에서 다음 프로토콜을 지정할 수 있습니다.

- TCP(Transmission Control Protocol)는 데이터 교환이 완료될 때까지 인스턴스에서 실행되는 클라이언트와 애플리케이션 간의 연결을 설정하고 유지 관리하는 데 주로 사용됩니다. 이 프로토콜은 널리 사용되는 프로토콜이며 방화벽 규칙에서 자주 지정할 수 있습니다. TCP는 전송된 데이터가 누락되지 않고 전송된 모든 데이터가 의도한 수신자에게 전송되도록 보장합니다. 높은 안정성이 요구되는 네트워크 애플리케이션과 웹 브라우징, 금융 거래 및 문자 메시징과 같이 전송 시간이 상대적으로 덜 중요한 네트워크 애플리케이션에 적합합니다. 데이터의 일부가 손실되면 이러한 사용 사례가 상당한 가치를 잃게 됩니다.
- UDP(User Datagram Protocol)는 인스턴스에서 실행되는 클라이언트와 애플리케이션 간에 지연 시간이 짧고 손실 허용 연결을 설정하는 데 주로 사용됩니다. 게임, 음성 및 비디오 통신과 같이 인식되는 지연 시간이 중요한 네트워크 애플리케이션에 적합합니다. 이러한 사용 사례는 인식된 품질에 악영향을 주지 않고 일부 데이터 손실을 경험할 수 있습니다.
- ICMP(Internet Control Message Protocol)는 데이터가 적시에 의도한 대상에 도달하는지 확인하는 등의 네트워크 통신 문제를 진단하는 데 주로 사용됩니다. 로컬 컴퓨터와 인스턴스 간의 연결 속도를 테스트하는 데 사용할 수 있는 Ping 유틸리티에 적합합니다. 데이터가 인스턴스에 도달한 후 로컬 컴퓨터로 돌아오는 데 걸리는 시간을 보고합니다.

  > **Note**
  >
  > Lightsail 콘솔을 사용하여 인스턴스의 IPv6 방화벽에 ICMP 규칙을 추가하면 ICMPv6를 사용하도록 규칙이 자동으로 구성됩니다. 자세한 내용은 Wikipedia의 IPv6용 인터넷 제어 메시지 프로토콜을 참조하세요.

- 모두는 모든 프로토콜 트래픽이 인스턴스로 유입되도록 허용하는 데 사용됩니다. 지정할 프로토콜을 잘 모르는 경우 이 프로토콜을 지정합니다. 여기에는 위에 지정된 프로토콜뿐만 아니라 모든 인터넷 프로토콜이 포함됩니다. 자세한 내용은 Internet Assigned Numbers Authority 웹 사이트의 Protocol Numbers를 참조하세요.

포트 지정

컴퓨터가 키보드 및 마우스 같은 주변 장치와 통신할 수 있는 컴퓨터의 물리적 포트와 마찬가지로, 네트워크 포트는 인스턴스의 인터넷 통신 엔드포인트 역할을 합니다. 컴퓨터가 인스턴스에 연결하려고 할 때 통신을 설정하기 위해 포트를 노출합니다.

방화벽 규칙에서 지정할 수 있는 포트의 범위는 0에서 65535 사이입니다. 클라이언트가 인스턴스와의 연결을 설정할 수 있도록 방화벽 규칙을 생성할 때 사용할 프로토콜(이 안내서의 앞부분에서 설명)과 연결을 설정할 수 있는 포트 번호를 지정합니다. 프로토콜 및 포트를 사용하여 설정할 수 있는 IP 주소를 지정할 수도 있습니다. 이에 대해서는 이 안내서의 다음 섹션에서 설명합니다.

다음은 일반적으로 사용되는 포트 중 일부와 함께 해당 포트를 사용하는 서비스입니다.

- FTP(File Transfer Protocol)를 통한 데이터 전송은 포트 20을 사용합니다.
- FTP를 통한 명령 제어는 포트 21을 사용합니다.
- SSH(Secure Shell)는 포트 22를 사용합니다.
- Telnet 원격 로그인 서비스 및 암호화되지 않은 텍스트 메시지는 포트 23을 사용합니다.
- SMTP(Simple Mail Transfer Protocol) 이메일 라우팅은 포트 25를 사용합니다.

> **Important**
>
> 인스턴스에 SMTP를 활성화하려면 인스턴스의 역방향 DNS도 구성해야 합니다. 그렇지 않으면 TCP 포트 25를 통한 이메일이 제한될 수 있습니다. 자세한 내용은 Amazon Lightsail 인스턴스에서 이메일 서버에 대한 역방향 DNS 구성을 참조하십시오.

- DNS(Domain Name System) 서비스는 포트 53을 사용합니다.
- 웹 브라우저에서 웹 사이트에 연결하는 데 사용하는 HTTP(Hypertext Transfer Protocol)는 포트 80을 사용합니다.
- 이메일 클라이언트가 서버에서 이메일을 검색하는 데 사용하는 POP3(Post Office Protocol)는 포트 110을 사용합니다.
- NNTP(Network News Transfer Protocol)는 포트 119를 사용합니다.
- NTP(Network Time Protocol)는 포트 123을 사용합니다.
- 디지털 메일을 관리하는 데 사용되는 IMAP(Internet Message Access Protocol)는 포트 143을 사용합니다.
- SNMP(Simple Network Management Protocol)는 포트 161을 사용합니다.
- 웹 브라우저에서 웹 사이트에 대해 암호화된 연결을 설정하기 위해 사용하는 TLS/SSL을 통한 HTTPS(HTTP Secure) HTTP는 포트 443을 사용합니다.

자세한 내용은 Internet Assigned Numbers Authority 웹 사이트의 Service Name and Transport Protocol Port Number Registry를 참조하세요.

애플리케이션 계층 프로토콜 유형 지정

방화벽 규칙을 생성할 때 인스턴스에서 활성화하려는 서비스에 따라 규칙의 프로토콜 및 포트 범위를 지정하는 사전 설정인 애플리케이션 계층 프로토콜 유형을 지정할 수 있습니다. 이렇게 하면 SSH, RDP, HTTP 등의 서비스에 사용할 공통 프로토콜과 포트를 검색할 필요가 없습니다. 이러한 애플리케

이션 계층 프로토콜 유형을 선택하기만 하면 프로토콜과 포트가 자동으로 지정됩니다. 고유한 프로토콜과 포트를 지정하려면 사용자 지정 규칙 애플리케이션 계층 프로토콜 유형을 선택하여 해당 파라미터를 제어할 수 있습니다.

> **Note**
>
> 애플리케이션 계층 프로토콜 유형은 Lightsail 콘솔을 통해서만 지정할 수 있습니다. Lightsail API, AWS Command Line Interface (AWS CLI) 또는 SDK를 사용하여 애플리케이션 계층 프로토콜 유형을 지정할 수 없습니다.

Lightsail 콘솔에서는 다음 애플리케이션 계층 프로토콜 유형을 사용할 수 있습니다.

- 사용자 지정 – 사용자 고유의 프로토콜과 포트를 지정하려면 이 옵션을 선택합니다.
- 모든 프로토콜 – 모든 프로토콜을 지정하고 고유한 포트를 지정하려면 이 옵션을 선택합니다.
- 모든 TCP – TCP 프로토콜을 사용하지만 어떤 포트를 열어야 할지 확실하지 않으면 이 옵션을 선택합니다. 이렇게 하면 모든 포트(0-65535)에서 TCP가 활성화됩니다.
- 모든 UDP – UDP 프로토콜을 사용하지만 어떤 포트를 열어야 할지 확실하지 않으면 이 옵션을 선택합니다. 이렇게 하면 모든 포트(0-65535)에서 UDP가 활성화됩니다.
- 모든 ICMP – 모든 ICMP 유형 및 코드를 지정하려면 이 옵션을 선택합니다.
- 사용자 지정 ICMP – ICMP 프로토콜을 사용하고 ICMP 유형 및 코드를 정의하려면 이 옵션을 선택합니다. ICMP 유형 및 코드에 대한 자세한 내용은 Wikipedia의 Control Messages를 참조하세요.
- DNS – 인스턴스가 DNS를 활성화하도록 하려면 이 옵션을 선택합니다. 이렇게 하면 포트 53를 통해 TCP 및 UDP를 활성화할 수 있습니다.
- HTTP – 웹 브라우저가 인스턴스에서 호스팅되는 웹 사이트에 연결하도록 하려면 이 옵션을 선택합니다. 이렇게 하면 포트 80을 통해 TCP가 활성화됩니다.
- HTTPS – 웹 브라우저가 인스턴스에서 호스팅되는 웹 사이트에 대해 암호화된 연결을 설정하도록 하려면 이 옵션을 선택합니다. 이렇게 하면 포트 443을 통해 TCP가 활성화됩니다.
- MySQL/Aurora – 클라이언트가 인스턴스에서 호스팅되는 MySQL 또는 Aurora 데이터베이스에 연결하도록 하려면 이 옵션을 선택합니다. 이렇게 하면 포트 3306을 통해 TCP가 활성화됩니다.
- Oracle-RDS – 클라이언트가 인스턴스에서 호스팅되는 Oracle 또는 RDS 데이터베이스에 연결하도록 하려면 이 옵션을 선택합니다. 이렇게 하면 포트 1521을 통해 TCP가 활성화됩니다.
- Ping(ICMP) – 인스턴스가 Ping 유틸리티를 사용하여 요청에 응답하도록 하려면 이 옵션을 선택합니다. IPv4 방화벽에서 ICMP 유형 8(에코) 및 코드-1(모든 코드)이 활성화됩니다. IPv6 방화벽에서 ICMP 유형 129(에코 응답) 및 코드 0이 활성화됩니다.

- RDP – RDP 클라이언트가 인스턴스에 연결하도록 하려면 이 옵션을 선택합니다. 이렇게 하면 포트 3389를 통해 TCP가 활성화됩니다.
- SSH – SSH 클라이언트가 인스턴스에 연결하도록 하려면 이 옵션을 선택합니다. 이렇게 하면 포트 22를 통해 TCP가 활성화됩니다.

소스 IP 주소 지정

기본적으로 방화벽 규칙은 모든 IP 주소가 지정된 프로토콜과 포트를 통해 인스턴스에 연결되도록 허용합니다. 이는 HTTP 및 HTTPS를 통한 웹 브라우저와 같은 트래픽에 이상적입니다. 그러나 모든 IP 주소가 해당 애플리케이션을 사용하여 인스턴스에 연결하는 것을 허용하지 않기 때문에 SSH 및 RDP 같은 트래픽에 대한 보안 위험이 있습니다. 이러한 이유로 방화벽 규칙을 IPv4 또는 IPv6 주소나 IP 주소 범위로 제한하도록 선택할 수 있습니다.

- IPv4 방화벽의 경우 - 단일 IPv4 주소(예: 203.0.113.1) 또는 IPv4 주소 범위를 지정할 수 있습니다. Lightsail 콘솔에서 범위는 대시(예: 192.0.2.0-192.0.2.255) 또는 CIDR 블록 표기법(예: 192.0.2.0/24)을 사용하여 지정할 수 있습니다. CIDR 블록 표기법에 대한 자세한 내용은 Wikipedia의 Classless Inter-Domain Routing을 참조하세요.
- IPv6 방화벽의 경우 - 단일 IPv6 주소(예: 2001:0db8:85a3:0000:0000:8a2e:0370:7334) 또는 IPv6 주소 범위를 지정할 수 있습니다. Lightsail 콘솔에서 IPv6 범위는 CIDR 블록 표기법(예: 2001:db8::/32)을 사용해서만 지정할 수 있습니다. IPv6 CIDR 블록 표기법에 대한 자세한 내용은 Wikipedia의 IPv6 CIDR blocks를 참조하세요.

기본 Lightsail 방화벽 규칙

새 인스턴스를 생성하면 인스턴스에 대한 기본 액세스를 허용하는 다음 기본 규칙 집합으로 IPv4 및 IPv6 방화벽이 미리 구성됩니다. 기본 규칙은 생성하는 인스턴스 유형에 따라 다릅니다. 이러한 규칙은 애플리케이션, 프로토콜, 포트 및 소스 IP 주소(예: 애플리케이션 - 프로토콜 - 포트 - 소스 IP 주소)로 나열됩니다.

AlmaLinux, Amazon Linux 2, Amazon Linux 2023, CentOS, Debian, openSUSE, FreeBSD 및 Ubuntu(기본 운영 체제)

SSH - TCP - 22 - 모든 IP 주소

HTTP - TCP - 80 - 모든 IP 주소

WordPress, Ghost, Joomla!, PrestaShop 및 Drupal(CMS 애플리케이션)

SSH - TCP - 22 - 모든 IP 주소

HTTP - TCP - 80 - 모든 IP 주소

HTTPS - TCP - 443 - 모든 IP 주소

cPanel & WHM(CMS 애플리케이션)

SSH - TCP - 22 - 모든 IP 주소

DNS (UDP) - UDP - 53 - 모든 IP 주소

DNS (TCP) - TCP - 53 - 모든 IP 주소

HTTP - TCP - 80 - 모든 IP 주소

HTTPS - TCP - 443 - 모든 IP 주소

사용자 지정 - TCP - 2078 - 모든 IP 주소

사용자 지정 - TCP - 2083 - 모든 IP 주소

사용자 지정 - TCP - 2087 - 모든 IP 주소

사용자 지정 - TCP - 2089 - 모든 IP 주소

LAMP, Django, Node.js, MEAN, GitLab 및 Nginx(개발 스택)

SSH - TCP - 22 - 모든 IP 주소

HTTP - TCP - 80 - 모든 IP 주소

HTTPS - TCP - 443 - 모든 IP 주소

Magento(전자 상거래 애플리케이션)

SSH - TCP - 22 - 모든 IP 주소

HTTP - TCP - 80 - 모든 IP 주소

HTTPS - TCP - 443 - 모든 IP 주소

Redmine(프로젝트 관리 애플리케이션)

SSH - TCP - 22 - 모든 IP 주소

HTTP - TCP - 80 - 모든 IP 주소

HTTPS - TCP - 443 - 모든 IP 주소

Plesk(호스팅 스택)

SSH - TCP - 22 - 모든 IP 주소

HTTP - TCP - 80 - 모든 IP 주소

HTTPS - TCP - 443 - 모든 IP 주소

사용자 지정 - TCP - 53 - 모든 IP 주소

사용자 지정 - UDP - 53 - 모든 IP 주소

사용자 지정 - TCP - 8443 - 모든 IP 주소

사용자 지정 - TCP - 8447 - 모든 IP 주소

Windows Server 2022, Windows Server 2019 및 Windows Server 2016

SSH - TCP - 22 - 모든 IP 주소

HTTP - TCP - 80 - 모든 IP 주소

RDP - TCP - 3389 - 모든 IP 주소

SQL Server Express 2022, SQL Server Express 2019 및 SQL Server Express 2016

SSH - TCP - 22 - 모든 IP 주소

HTTP - TCP - 80 - 모든 IP 주소

RDP - TCP - 3389 - 모든 IP 주소

Lightsail 인스턴스에 방화벽 규칙 추가

Amazon Lightsail 인스턴스의 IPv4 및 IPv6 방화벽에 규칙을 추가하여 연결 가능한 트래픽을 제어할 수 있습니다. 방화벽 규칙을 추가할 때 인스턴스에 연결 가능한 애플리케이션 계층 프로토콜 유형, 프로토콜, 포트 및 소스 IPv4 또는 IPv6 주소를 지정할 수 있습니다. 방화벽에 대한 자세한 내용은 방화벽 및 포트를 참조하세요.

인스턴스 방화벽 규칙 추가 및 편집

Lightsail 콘솔에서 방화벽 규칙을 추가하거나 편집하려면 다음 단계를 수행하세요.

1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 인스턴스를 선택합니다.
3. 방화벽 규칙을 추가하거나 편집할 인스턴스의 이름을 선택합니다.
4. 인스턴스 관리 페이지에서 네트워킹 탭을 선택합니다.

   네트워킹(Networking) 탭에는 인스턴스의 퍼블릭 및 프라이빗 IP 주소와 인스턴스에 대해 구성된 IPv4 또는 IPv6 방화벽이 표시됩니다.

   > **Note**
   >
   > IPv6 방화벽은 인스턴스에 대해 IPv6를 사용하도록 설정한 경우에만 표시됩니다. 자세한 내용은 IPv6 사용 또는 사용 중지를 참조하세요.

5. 규칙의 소스 IP가 IPv4 주소인지 IPv6 주소인지에 따라 다음 단계 중 하나를 완료합니다.

   - IPv4 방화벽 규칙을 추가하려면 페이지에서 아래로 스크롤하여 IPv4 방화벽(IPv4 Firewall) 섹션에서 규칙 추가(Add rule)를 선택합니다.
   - IPv6 방화벽 규칙을 추가하려면 페이지에서 아래로 스크롤하여 IPv6 방화벽(IPv6 Firewall) 섹션에서 규칙 추가(Add rule)를 선택합니다.

   방화벽 중 하나의 기존 규칙 옆에 있는 편집(Edit)(연필 아이콘)을 선택하여 편집할 수도 있습니다.
6. 애플리케이션 드롭다운 메뉴에서 애플리케이션 계층 프로토콜 유형을 선택합니다.

   애플리케이션 계층 프로토콜 유형을 선택하면 프로토콜 및 포트 사전 설정이 자동으로 지정됩니다. 예제 값은 사용자 지정 규칙, 모든 TCP, 모든 UDP, 사용자 지정 ICMP, SSH, RDP입니다.

   선택한 애플리케이션 계층 프로토콜 유형에 따라 다음과 같은 선택적 설정을 구성할 수 있습니다.

   - (선택 사항) 사용자 지정 옵션을 선택한 경우 프로토콜 드롭다운 메뉴에서 값을 선택할 수 있습니다. 사용 가능한 프로토콜 값은 TCP 및 UDP입니다.

     포트 필드에 단일 포트 번호 또는 포트 번호 범위(예: 7000-8000)를 입력할 수도 있습니다.
   - (선택 사항) 사용자 지정 ICMP 옵션을 선택한 경우 유형 필드에서 ICMP 유형을, 코드 필드에서 ICMP 코드를 지정할 수 있습니다. ICMP 유형 및 코드에 대한 자세한 내용은 Wikipedia의 메시지 제어를 참조하세요.

> **Note**
>
> Lightsail 콘솔을 사용하여 인스턴스의 IPv6 방화벽에 ICMP 규칙을 추가하면 ICMPv6를 사용하도록 규칙이 자동으로 구성됩니다. 자세한 내용은 Wikipedia의 IPv6용 인터넷 제어 메시지 프로토콜을 참조하세요.

- (선택 사항) 지정된 프로토콜 및 포트에 대한 액세스를 특정 IP 주소 또는 IP 주소 범위로 제한하려면 IP 주소로 제한을 선택합니다. 지정된 프로토콜 및 포트에 대해 모든 IP 주소를 허용하려면 이 옵션을 선택하지 않은 상태로 둡니다.

  단일 IPv4 주소(예: `203.0.113.1`) 또는 IPv4 주소 범위를 입력할 수 있습니다. 범위는 대시(예: `192.0.2.0-192.0.2.255`) 또는 CIDR 블록 표기법(예: `192.0.2.0/24`)을 사용하여 지정할 수 있습니다. CIDR 블록 표기법에 대한 자세한 내용은 Wikipedia의 클래스 없는 도메인 간 라우팅을 참조하세요.

- (선택 사항) SSH 또는 RDP 애플리케이션 계층 프로토콜 유형을 선택한 다음 IP 주소로 제한을 선택하는 경우, Lightsail 브라우저 SSH/RDP 허용을 선택하여 Lightsail 콘솔에서 사용 가능한 브라우저 기반 SSH 및 RDP 클라이언트를 사용하여 인스턴스에 대한 연결을 허용할 수 있습니다. 이러한 브라우저 기반 클라이언트를 통한 액세스를 차단하려면 이 옵션을 선택하지 않은 상태로 둡니다.

7. 생성을 선택하여 방화벽에 규칙을 추가합니다.

   방화벽 규칙은 잠시 후 추가됩니다.

방화벽 규칙 삭제

방화벽 규칙을 추가하고 편집하고 Amazon Lightsail 인스턴스에 대한 기존 규칙을 삭제할 수도 있습니다. 인스턴스에 특정 인바운드 트래픽을 더 이상 허용하지 않는 경우 방화벽 규칙을 제거해야 할 수 있습니다. IPv4 및 IPv6 방화벽 규칙을 삭제하는 프로세스는 간단하며 Lightsail 콘솔을 통해 직접 수행할 수 있습니다. Lightsail 콘솔에서 인스턴스 방화벽 규칙을 삭제하려면 다음 단계를 수행하십시오.

1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 인스턴스를 선택합니다.
3. 방화벽 규칙을 삭제할 인스턴스의 이름을 선택합니다.
4. 인스턴스 관리 페이지에서 네트워킹 탭을 선택합니다.
5. 규칙의 소스 IP가 IPv4 주소인지 IPv6 주소인지에 따라 다음 단계 중 하나를 완료합니다.

- IPv4 방화벽 규칙을 삭제하려면 페이지에서 아래로 스크롤하여 IPv4 방화벽(IPv4 Firewall) 섹션에서 기존 규칙 옆에 있는 삭제(Delete)(휴지통 아이콘)를 선택하여 삭제합니다.
- IPv6 방화벽 규칙을 삭제하려면 페이지에서 아래로 스크롤하여 IPv6 방화벽(IPv6 Firewall) 섹션에서 기존 규칙 옆에 있는 삭제(Delete)(휴지통 아이콘)를 선택하여 삭제합니다.

> **Important**
>
> 방화벽 규칙은 인스턴스의 퍼블릭 IP 주소를 통해 흐르는 트래픽에만 영향을 미칩니다. 동일한 AWS 리전 리전에 있는 계정의 Lightsail 리소스 또는 동일한 AWS 리전 리전에 있는 피어링된 Virtual Private Cloud(VPC)의 리소스로부터 시작되어 인스턴스의 프라이빗 IP 주소를 통해 흐르는 트래픽에 영향을 미치지 않습니다. 예를 들어, 인스턴스 방화벽에서 SSH 규칙(TCP 포트 22)을 삭제하는 경우 동일한 Lightsail 계정, 그리고 동일한 AWS 리전 리전에 있는 다른 인스턴스는 인스턴스의 프라이빗 IP 주소 지정을 통해 SSH를 사용하여 계속 연결할 수 있습니다.

방화벽 규칙은 잠시 후 삭제됩니다.

Lightsail 인스턴스에 대한 방화벽 규칙 참조

인스턴스의 역할을 반영하는 규칙을 Amazon Lightsail 인스턴스의 방화벽에 추가할 수 있습니다. 예를 들어 웹 서버로 구성된 인스턴스에는 인바운드 HTTP 및 HTTPS 액세스를 허용하는 방화벽 규칙이 필요합니다. 데이터베이스 인스턴스에는 데이터베이스 유형에 대한 액세스(예: MySQL의 경우 포트 3306을 통한 액세스)를 허용하는 규칙이 필요합니다. 방화벽에 대한 자세한 내용은 Lightsail의 인스턴스 방화벽을 참조하세요.

이 안내서에서는 특정 유형의 액세스를 위해 인스턴스 방화벽에 추가할 수 있는 방화벽 규칙의 예를 제공합니다. 달리 명시되지 않는 한 규칙은 애플리케이션, 프로토콜, 포트 및 소스 IP 주소(예: 애플리케이션 - 프로토콜 - 포트 - 소스 IP 주소)로 나열됩니다.

목차

- 웹 서버 규칙
- 컴퓨터에서 인스턴스에 연결하는 규칙
- 데이터베이스 서버 규칙
- DNS 서버 규칙

- SMTP 이메일

웹 서버 규칙

다음 인바운드 규칙은 HTTP 및 HTTPS 액세스를 허용합니다.

> **Note**
>
> 일부 Lightsail 인스턴스에는 기본적으로 다음과 같은 방화벽 규칙이 구성되어 있습니다. 자세한 내용은 방화벽 및 포트를 참조하세요.

HTTP

HTTP - TCP - 80 - 모든 IP 주소

HTTPS

HTTPS - TCP - 443 - 모든 IP 주소

컴퓨터에서 인스턴스에 연결하는 규칙

인스턴스에 연결하려면 SSH 액세스(Linux 인스턴스의 경우) 또는 RDP 액세스(Windows 인스턴스의 경우)를 허용하는 규칙을 추가합니다.

> **Note**
>
> 모든 Lightsail 인스턴스에는 기본적으로 다음과 같은 방화벽 규칙이 구성되어 있습니다. 자세한 내용은 방화벽 및 포트를 참조하세요.

SSH

SSH - TCP - 22 - 컴퓨터의 퍼블릭 IP 주소 또는 로컬 네트워크의 IP 주소 범위(CIDR 블록 표기법)

RDP

RDP - TCP - 3389 - 컴퓨터의 퍼블릭 IP 주소 또는 로컬 네트워크의 IP 주소 범위(CIDR 블록 표기법)

데이터베이스 서버 규칙

다음의 인바운드 규칙은 인스턴스에서 실행 중인 데이터베이스 유형에 따라 데이터베이스 액세스를 위해 추가할 수 있는 규칙을 예로 든 것입니다.

SQL Server

사용자 지정 - TCP - 1433 - 컴퓨터의 퍼블릭 IP 주소 또는 로컬 네트워크의 IP 주소 범위(CIDR 블록 표기법)

MySQL/Aurora

MySQL/Aurora - TCP - 3306 - 컴퓨터의 퍼블릭 IP 주소 또는 로컬 네트워크의 IP 주소 범위(CIDR 블록 표기법)

PostgreSQL

PostgreSQL - TCP - 5432 - 컴퓨터의 퍼블릭 IP 주소 또는 로컬 네트워크의 IP 주소 범위(CIDR 블록 표기법)

Oracle-RDS

Oracle-RDS - TCP - 1521 - 컴퓨터의 퍼블릭 IP 주소 또는 로컬 네트워크의 IP 주소 범위(CIDR 블록 표기법)

Amazon Redshift

사용자 지정 - TCP - 5439 - 컴퓨터의 퍼블릭 IP 주소 또는 로컬 네트워크의 IP 주소 범위(CIDR 블록 표기법)

DNS 서버 규칙

인스턴스를 DNS 서버로 설정한 경우 TCP 및 UDP 트래픽이 포트 53을 통해 DNS 서버에 연결할 수 있는지 확인해야 합니다.

DNS (TCP)

DNS(TCP) - TCP - 53 - 컴퓨터의 IP 주소 또는 로컬 네트워크의 IP 주소 범위(CIDR 블록 표기법)

DNS (UDP)

DNS(UDP) - UDP - 53 - 컴퓨터의 IP 주소 또는 로컬 네트워크의 IP 주소 범위(CIDR 블록 표기법)

SMTP 이메일

인스턴스에서 SMTP를 활성화하려면 다음 방화벽 규칙을 구성해야 합니다.

> **Important**
>
> 다음 규칙을 구성한 후에는 인스턴스에 대한 역방향 DNS도 구성해야 합니다. 그렇지 않으면 TCP 포트 25를 통해 이메일이 제한될 수 있습니다. 자세한 내용은 이메일 서버에 대한 역방향 DNS 구성을 참조하세요.

SMTP

사용자 지정 - TCP - 25 - 인스턴스와 통신하는 호스트의 IP 주소

최적의 성능을 위해 Lightsail 인스턴스 버스팅 감지

> **알고 계셨나요?**
>
> 인스턴스 스냅샷에서 인스턴스를 생성할 때 인스턴스 번들을 더 큰 크기로 변경할 수 있습니다. 자세한 내용은 스냅샷에서 Lightsail 인스턴스, 스토리지 또는 데이터베이스 크기를 참조하세요.

Amazon Lightsail 인스턴스는 기준 CPU 성능을 제공하지만, 필요에 따라 기준 이상의 추가 CPU 성능을 일시적으로 제공할 수도 있습니다. 이를 버스팅이라고 합니다. 기준 성능과 버스트 기능은 다음 인스턴스 지표에 의해 좌우됩니다.

- CPU 사용률 – 인스턴스에서 사용 중인 할당된 컴퓨팅 유닛의 백분율입니다. 이 지표는 인스턴스에서 애플리케이션을 실행하는 데 사용되는 처리 능력을 식별합니다.
- CPU 버스트 용량 백분율 – 인스턴스에서 사용할 수 있는 CPU 성능의 백분율입니다.
- CPU 버스트 용량(분) – 인스턴스가 100% CPU 사용률로 버스트할 수 있는 시간입니다.

다음 주제에서는 이러한 지표를 모니터링하여 인스턴스의 가용성을 극대화하는 방법을 알아봅니다.

주제

- Lightsail 인스턴스의 기준 CPU 성능 및 버스트 용량 누적 이해

- Lightsail 인스턴스의 CPU 버스트 용량 발생 보기
- Lightsail 인스턴스가 버스트되는 시점 식별
- Lightsail 인스턴스의 CPU 버스트 용량 모니터링
- Lightsail 인스턴스의 CPU 사용률 및 버스트 용량 보기
- Lightsail 인스턴스의 높은 CPU 사용률 문제 해결

Lightsail 인스턴스의 기준 CPU 성능 및 버스트 용량 누적 이해

> 알고 계셨나요?
>
> 인스턴스 스냅샷에서 인스턴스를 생성할 때 인스턴스 번들을 더 큰 크기로 변경할 수 있습니다. 자세한 내용은 스냅샷에서 Lightsail 인스턴스, 스토리지 또는 데이터베이스 크기를 참조하세요.

Lightsail 인스턴스는 시간당 특정 비율의 CPU 버스트 용량을 (밀리초 수준의 시간 정밀도로) 지속적으로 획득하며, 이는 인스턴스의 CPU 사용률이 0%보다 클 때도 사용됩니다. 버스트 용량이 누적되는지 아니면 소비되는지를 결정하는 산정 프로세스도 밀리초 수준의 시간 정밀도로 수행되므로 CPU 버스트 용량 과소비를 염려할 필요는 없습니다. 즉, 짧은 CPU 버스트는 약간의 버스트 용량만을 소비합니다.

인스턴스에서 기준 성능에 필요한 것보다 더 적은 CPU 리소스를 사용하는 경우(예: 유휴 상태) 사용하지 않은 CPU 버스트 용량은 CPU 버스트 용량 백분율 및 시간(분)의 형태로 누적됩니다. 인스턴스가 기준 성능 수준 이상으로 버스트해야 할 경우 누적된 CPU 버스트 용량을 소모합니다. 인스턴스가 획득한 CPU 버스트 용량이 많을수록 추가 성능이 필요할 때 기준을 초과하여 버스트할 수 있는 시간이 증가합니다.

기준 CPU 성능

다음 테이블에 Lightsail의 이중 스택 인스턴스 플랜에 대한 성능 기준이 요약되어 있습니다. IPv6 전용 플랜의 가격은 다르지만 성능 기준은 동일합니다.

인스턴스 요금제	vCPU	Memory	스토리지	성능 기준
Linux 또는 Unix $5 및 Windows $9.50	2	512MB	20GB	5%
Linux 또는 Unix $7 및 Windows $14	2	1GB	40GB	10%

인스턴스 요금제	vCPU	Memory	스토리지	성능 기준
Linux 또는 Unix $12 및 Windows $22	2	2GB	60GB	20%
Linux 또는 Unix $24 및 Windows $44	2	4GB	80GB	20%
Linux 또는 Unix $44 및 Windows $74	2	8GB	160GB	30%
Linux 또는 Unix $84 및 Windows $124	4	16 GB	320GB	40%
Linux 또는 Unix $164 및 Windows $244	8	32GB	640GB	40%
* Linux 또는 Unix $384 및 Windows $574	16	64GB	1,280GB	40%
* Linux 또는 Unix $884 및 Windows $1,254	32	128GB	1,280GB	40%
* Linux 또는 Unix $1,324 및 Windows $1,884	48	192GB	1,280GB	40%
* Linux 또는 Unix $1,764 및 Windows $2,504	64	256GB	1,280GB	40%

* 이러한 인스턴스 플랜은 필요에 따라 자동으로 버스트되며 버스트 용량을 활용하지 않습니다.

이러한 성능 기준은 vCPU를 기준으로 합니다. Lightsail 콘솔의 CPU 사용률 지표 그래프는 vCPU가 두 개 이상인 인스턴스에 대한 CPU 사용률 및 기준의 평균을 나타냅니다. 예를 들어, Linux 또는 Unix 월 44 USD 인스턴스에는 vCPU가 2개 있고 평균 CPU 사용률 기준은 30%입니다. 따라서 만약:

- 한 vCPU가 50%에서 작동하고 다른 vCPU는 0%에서 작동하면 25%의 평균 CPU 사용률이 그래프에 표시됩니다. 이렇게 하면 인스턴스의 CPU 사용률이 30% 기준 미만으로 지속 가능 영역에 있게 됩니다.
- 한 vCPU가 30%에서 작동하고 다른 vCPU는 20%에서 작동하면 25%의 평균 CPU 사용률이 그래프에 표시됩니다. 이렇게 하면 인스턴스의 CPU 사용률이 30% 기준 미만으로 지속 가능 영역에 있게 됩니다.

- 한 vCPU가 35%에서 작동하고 다른 vCPU는 25%에서 작동하면 30%의 평균 CPU 사용률이 그래프에 표시됩니다. 이렇게 하면 인스턴스의 CPU 사용률이 30% 기준에 있게 됩니다.
- 한 vCPU가 100%에서 작동하고 다른 vCPU는 90%에서 작동하면 95%의 평균 CPU 사용률이 그래프에 표시됩니다. 이렇게 하면 인스턴스의 CPU 사용률이 30% 기준을 초과하여 버스트 가능 영역에 있게 됩니다.

지속 가능 영역과 버스트 가능 영역에 대한 자세한 내용은 이 설명서의 뒷부분에서 인스턴스가 버스트되는 시점 식별을 참조하세요.

이전 세대 CPU 성능

다음 표에는 2023년 6월 29일 이전에 생성된 Lightsail 인스턴스의 성능 기준이 요약되어 있습니다. 이러한 성능 기준은 vCPU를 기준으로 합니다.

인스턴스 요금제	vCPU	Memory	스토리지	성능 기준
Linux 또는 Unix $5 및 Windows $9.50	1	512MB	20GB	5%
Linux 또는 Unix $7 및 Windows $14	1	1GB	40GB	10%
Linux 또는 Unix $12 및 Windows $22	1	2GB	60GB	20%
Linux 또는 Unix $24 및 Windows $44	2	4GB	80GB	20%
Linux 또는 Unix $44 및 Windows $74	2	8GB	160GB	30%
Linux 또는 Unix $84 및 Windows $124	4	16 GB	320GB	22.5%
Linux 또는 Unix $164 및 Windows $244	8	32GB	640GB	17%

Lightsail 인스턴스의 CPU 버스트 용량 발생 보기

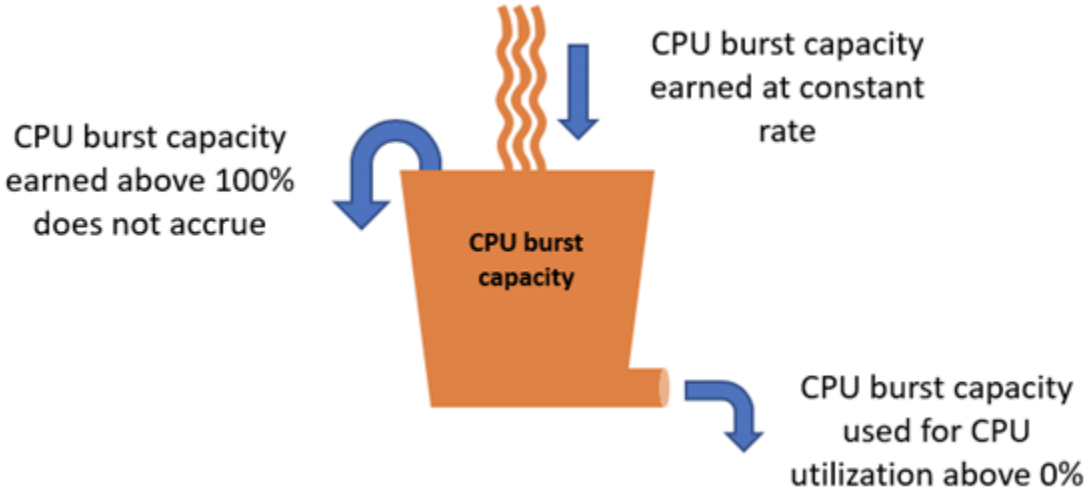
> 알고 계셨나요?
>
> 인스턴스 스냅샷에서 인스턴스를 생성할 때 인스턴스 번들을 더 큰 크기로 변경할 수 있습니다. 자세한 내용은 스냅샷에서 Lightsail 인스턴스, 스토리지 또는 데이터베이스 크기를 참조하세요.

Linux 또는 Unix 380 USD 이상 플랜 및 Windows 570 USD 이상 플랜을 제외한 Amazon Lightsail 인스턴스 플랜에는 시간당 CPU 버스트 용량의 4.17%가 누적됩니다. 누적될 수 있는 최대 CPU 버스트 용량은 24시간 동안 얻을 수 있는 CPU 버스트 용량 백분율과 같습니다. CPU 버스트 용량 백분율이 100%에 도달하면 인스턴스에서 CPU 버스트 용량 생성을 중지합니다.

Important

누적된 CPU 버스트 용량

- Linux 또는 Unix 380 USD 및 Windows 570 USD 이상의 인스턴스 플랜 - 이러한 플랜에는 CPU 버스트 용량이 누적되지 않습니다. 필요에 따라 자동으로 버스트됩니다.
- 2023년 6월 29일 이전에 생성된 인스턴스 - 인스턴스가 중지되면 CPU 버스트 용량이 지속되지 않습니다. 인스턴스를 중지하면 누적된 버스트 용량이 모두 손실됩니다.
- 2023년 6월 29일 이후에 생성된 인스턴스 - CPU 버스트 용량은 인스턴스 중지와 시작 사이에 7일 동안 지속됩니다.
- 실행 중인 인스턴스에서 누적된 CPU 버스트 용량은 만료되지 않습니다.



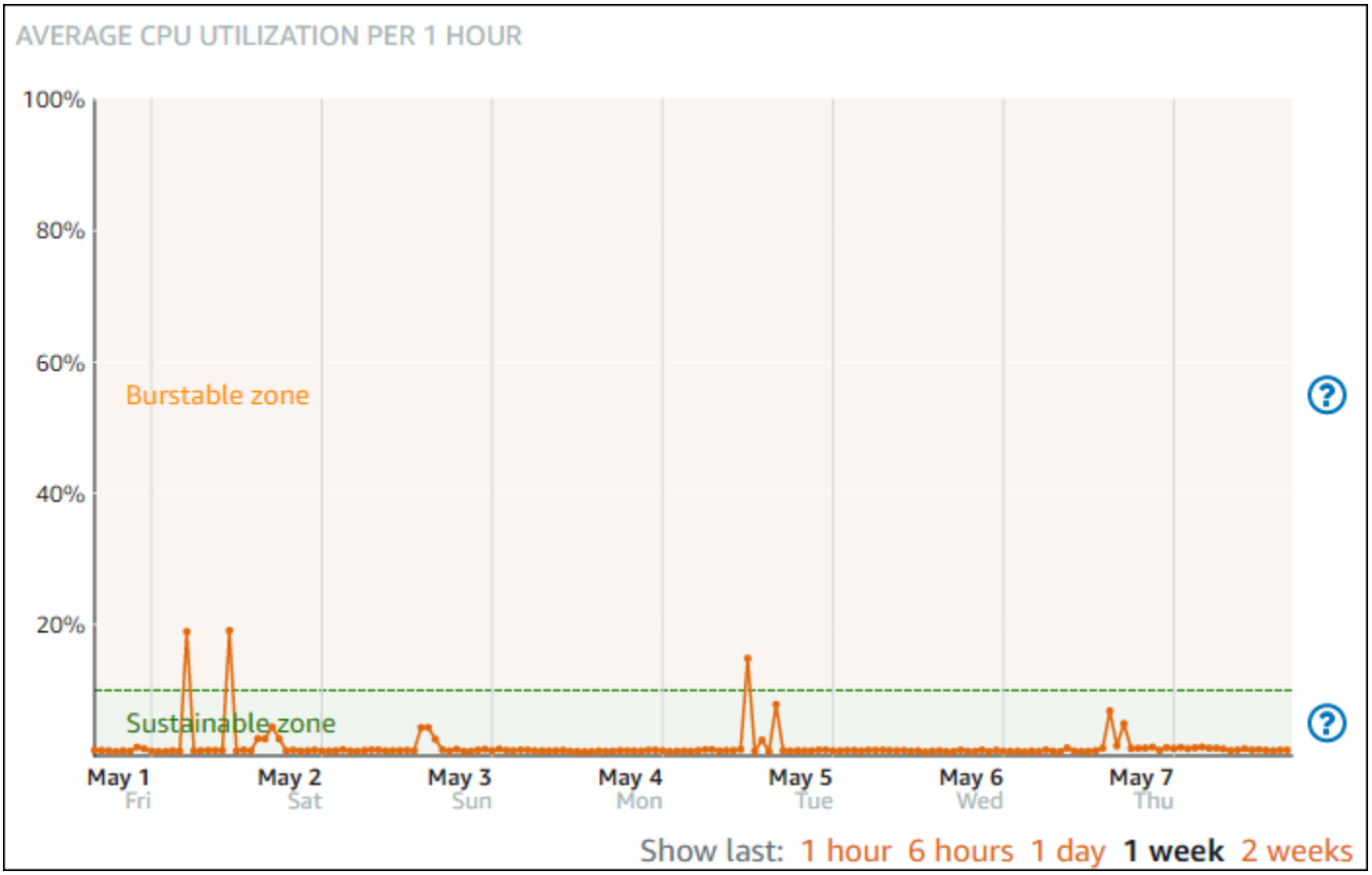
Lightsail 인스턴스는 시작 시 추가 CPU 버스트 용량을 받는데, 이를 시작 CPU 버스트 용량이라고 합니다. 시작 CPU 버스트 용량을 사용하면 인스턴스가 추가 버스트 용량을 누적하기 전에 시작 직후 버스트할 수 있습니다. 시작 CPU 버스트 용량은 버스트 용량 한도에 포함되지 않습니다. 인스턴스가 시작 CPU 버스트 용량을 사용하지 않았고 더 많은 버스트 용량을 누적하면서 24시간 동안 유휴 상태를 유지하는 경우, 해당 CPU 버스트 용량(백분율) 지표 그래프는 100% 이상으로 나타납니다.

또한, 일부 Lightsail 인스턴스는 시작 모드에서 시작되므로 일반적으로 버스트 가능한 인스턴스에 있는 일부 성능 제한이 일시적으로 제거됩니다. 실행 모드를 사용하면 인스턴스의 전체 성능에 영향을 미치지 않고 리소스 집약적인 스크립트를 시작할 때 실행할 수 있습니다.

Lightsail 인스턴스가 버스트되는 시점 식별

> 알고 계셨나요?
>
> 인스턴스 스냅샷에서 인스턴스를 생성할 때 인스턴스 번들을 더 큰 크기로 변경할 수 있습니다. 자세한 내용은 스냅샷에서 Lightsail 인스턴스, 스토리지 또는 데이터베이스 크기를 참조하세요.

인스턴스의 CPU 사용률 지표 그래프에는 지속 가능 영역과 버스트 가능 영역이 표시됩니다. 다음 CPU 사용률 지표 그래프 예에서 인스턴스는 Linux 또는 Unix 기반 월 7 USD 인스턴스 플랜을 사용하기 때문에 성능 기준은 10%입니다.



Lightsail 인스턴스는 시스템 운영에 영향을 주지 않고 지속 가능 영역에서 무기한으로 작동할 수 있습니다. 코드 컴파일, 새 소프트웨어 설치, 배치 작업 실행 또는 최대 로드 요청 처리 등으로 부하가 큰 경

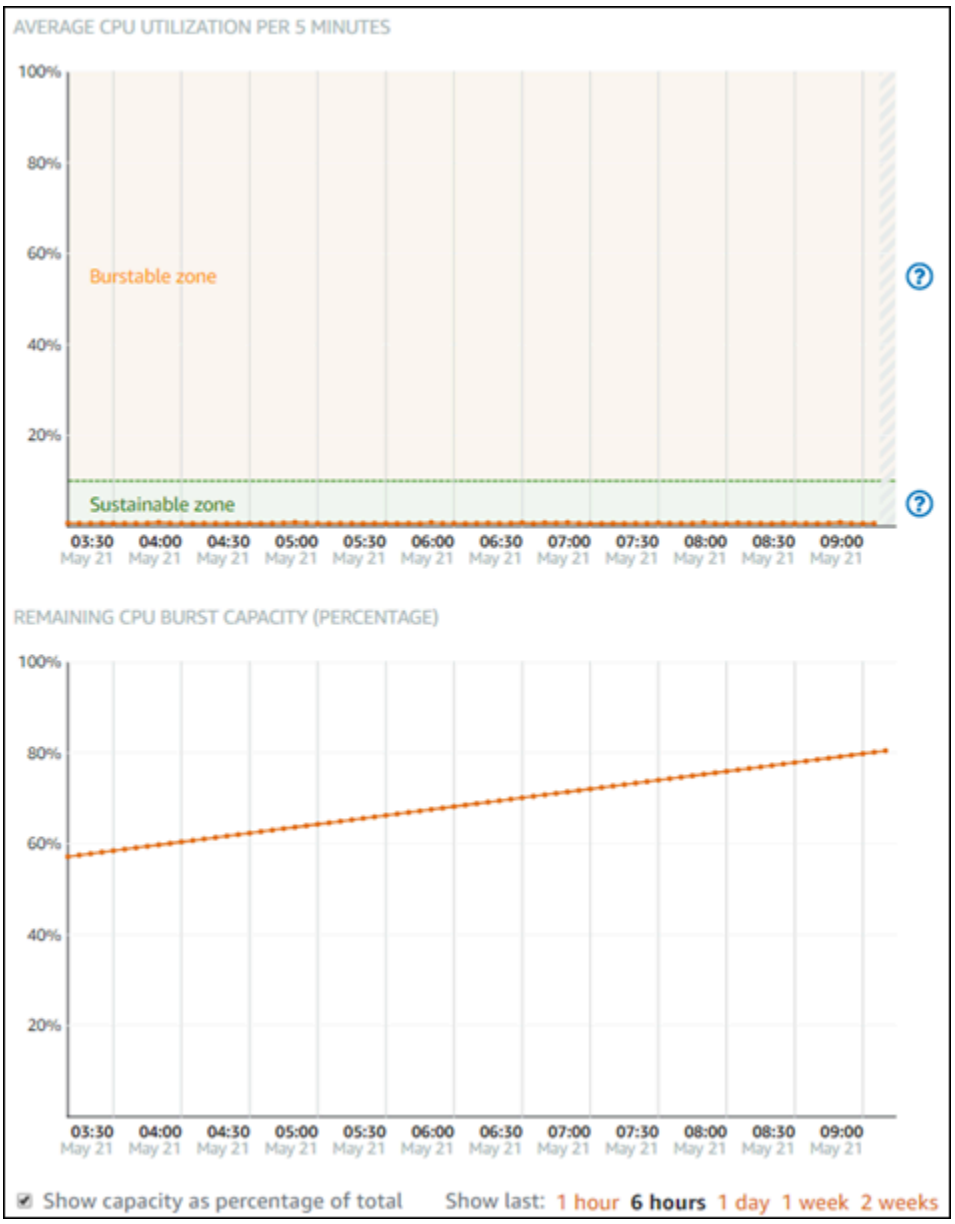
우 버스트 가능 영역에서 인스턴스가 작동할 수 있습니다. 버스트 가능 영역에서 작동하는 동안에는 인스턴스가 상대적으로 더 많은 CPU 주기를 사용합니다. 따라서 제한된 기간 동안만 이 영역에서 작동할 수 있습니다.

인스턴스가 버스트 가능 영역에서 작동할 수 있는 기간은 인스턴스가 버스트 가능 영역 내의 어느 지점에 있는지에 따라 달라집니다. 버스트 가능 영역의 하단에서 작동하는 인스턴스는 버스트 가능 영역의 상단에서 작동하는 인스턴스보다 오랜 기간 동안 버스트될 수 있습니다. 그러나 장기간 버스트 가능 영역에 있는 인스턴스는 지속 가능 영역에서 다시 작동할 때까지 결국 모든 CPU 용량을 소모하게 됩니다. 따라서 이 안내서의 다음 단원에서 설명하는 나머지 CPU 버스트 용량도 모니터링하는 것이 중요합니다.

Lightsail 인스턴스의 CPU 버스트 용량 모니터링

> 알고 계셨나요?
>
> 인스턴스 스냅샷에서 인스턴스를 생성할 때 인스턴스 번들을 더 큰 크기로 변경할 수 있습니다. 자세한 내용은 스냅샷에서 Lightsail 인스턴스, 스토리지 또는 데이터베이스 크기를 참조하세요.

Lightsail 콘솔의 CPU 개요 페이지에는 사용 가능한 CPU 버스트 용량과 비교하여 인스턴스의 CPU 사용률이 표시됩니다. 다음 CPU 개요 예에서는 인스턴스가 지속 가능 영역에서 계속 기준 미만으로 작동했기 때문에 CPU 버스트 용량 백분율이 증가했습니다.



나머지 CPU 버스트 용량 그래프 보기는 CPU 버스트 용량 백분율과 시간(분) 간에 전환할 수 있습니다. 인스턴스는 버스트 영역에서 작동할 때 더 많은 CPU 버스트 용량을 소비합니다. CPU 버스트 용량 시간(분) 지표는 인스턴스가 100% CPU 사용률로 버스트할 수 있는 시간이며, 버스트 가능 영역에서 작동할 때 인스턴스의 현재 CPU 사용률(%)과 동일한 비율로 소비됩니다. 예를 들어, Linux 또는 Unix 기반 월 7 USD 인스턴스의 경우 CPU 사용률 기준이 10%이고, 시간당 6분의 CPU 버스트 용량을 누적합니다. 따라서 인스턴스가 다음 조건에서 작동하는 경우를 예로 들어보겠습니다.

- 60분 동안의 버스트 가능 영역에서 100% CPU 사용률인 경우 해당 기간 동안 100% 비율로 CPU 버스트 용량 시간(분)을 소비합니다. 인스턴스는 60분의 CPU 버스트 용량을 소비하고 6분을 누적하므로 총 54분을 소비합니다.

- 60분 동안의 버스트 가능 영역에서 50% CPU 사용률인 경우 해당 기간 동안 50% 비율로 CPU 버스트 용량 시간(분)을 소비합니다. 인스턴스는 30분의 CPU 버스트 용량을 소비하고 6분을 누적하므로 총 24분을 소비합니다.
- 60분 동안 인스턴스의 기준에서 10% CPU 사용률인 경우 해당 기간 동안 10% 비율로 CPU 버스트 용량 시간(분)을 소비합니다. 인스턴스는 6분의 CPU 버스트 용량을 소비하고 6분을 누적합니다. 인스턴스가 해당 기준에서 작동할 때 CPU 버스트 용량 시간(분)은 증가하거나 감소하지 않습니다.
- 60분 동안의 지속 가능 영역에서 5% CPU 사용률인 경우 해당 기간 동안 5% 비율로 CPU 버스트 용량 시간(분)을 소비합니다. 인스턴스는 3분의 CPU 버스트 용량을 소비했고 6분을 누적했으므로 총 3분을 누적합니다.

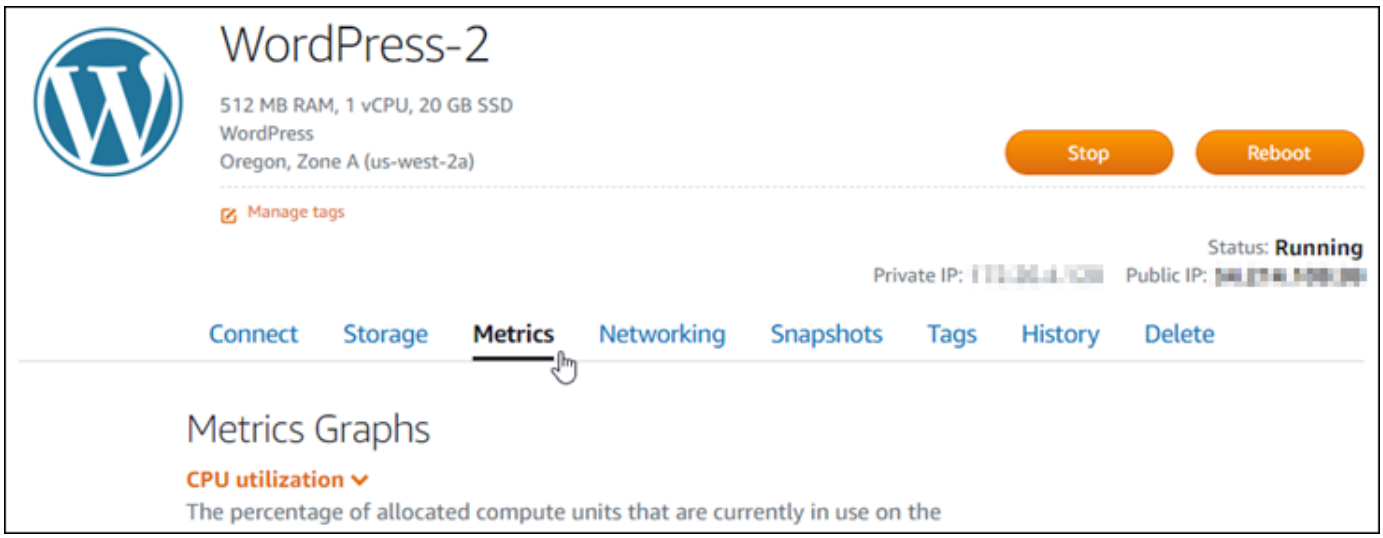
또는 인스턴스에서 60분의 CPU 버스트 용량을 누적한 경우, 인스턴스는 60분 동안 100% CPU 사용률, 120분 동안 50% CPU 사용률 또는 150분 동안 25% CPU 사용률로 작동할 수 있습니다.

Lightsail 인스턴스의 CPU 사용률 및 버스트 용량 보기

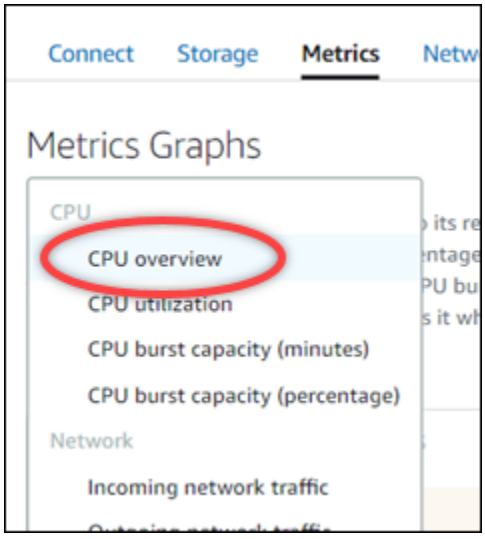
> 알고 계셨나요?
>
> 인스턴스 스냅샷에서 인스턴스를 생성할 때 인스턴스 번들을 더 큰 크기로 변경할 수 있습니다. 자세한 내용은 스냅샷에서 Lightsail 인스턴스, 스토리지 또는 데이터베이스 크기를 참조하세요.

다음 단계를 완료하여 CPU 개요 페이지에 액세스하고, 인스턴스의 CPU 사용률 및 남은 CPU 버스트 용량을 확인합니다.

1. Lightsail 콘솔에 로그인합니다.
2. Lightsail 홈 페이지에서 CPU 사용률 및 버스트 용량을 보려는 인스턴스의 이름을 선택합니다.
3. 인스턴스 관리 페이지에서 지표 탭을 선택합니다.



4. 지표 그래프(Metrics graphs) 머리말 아래 드롭다운 메뉴의 CPU 개요(CPU overview)를 선택합니다.



이 페이지에는 5분당 평균 CPU 사용률 및 남은 CPU 버스트 용량 그래프가 표시됩니다.

> **Note**
>
> 남은 CPU 버스트 용량 그래프에는 인스턴스를 만든 후 짧은 시간 동안 그래프에 실행 모드(Launch mode) 영역이 표시될 수 있습니다. 일부 Lightsail 인스턴스는 시작 모드에서 시작되므로 일반적으로 버스트 가능한 인스턴스에 있는 일부 성능 제한이 일시적으로 제거됩니다. 실행 모드를 사용하면 인스턴스의 전체 성능에 영향을 미치지 않고 리소스 집약적인 스크립트를 시작할 때 실행할 수 있습니다.

5. 지표 그래프에 대해 다음 작업을 수행할 수 있습니다.

   - 버스트 용량 그래프에서 Show capacity as percentage of total(용량을 총 백분율로 표시)을 선택하여 사용 가능한 버스트 용량 시간(분)에서 사용 가능한 버스트 용량 백분율로 보기를 변경합니다.
   - 1시간, 6시간, 1일, 1주 및 2주 동안의 데이터를 표시하도록 그래프 보기를 변경합니다.
   - 데이터 포인트에 커서를 놓으면 해당 데이터 포인트에 대한 자세한 정보를 볼 수 있습니다.
   - CPU 사용률 및 버스트 용량이 지정한 임계값을 초과하면 알림을 받을 경보를 추가합니다. CPU 개요 페이지에서는 경보를 추가할 수 없습니다. 개별 CPU 사용률, CPU 버스트 용량 백분율 및

CPU 버스트 용량 시간(분) 지표 그래프 페이지에서 추가해야 합니다. 자세한 내용은 경보 및 인스턴스 지표 경보 생성을 참조하세요.

Lightsail 인스턴스의 높은 CPU 사용률 문제 해결

> 알고 계셨나요?
>
> 인스턴스 스냅샷에서 인스턴스를 생성할 때 인스턴스 번들을 더 큰 크기로 변경할 수 있습니다. 자세한 내용은 스냅샷에서 Lightsail 인스턴스, 스토리지 또는 데이터베이스 크기를 참조하세요.

인스턴스가 버스트 영역에서 자주 작동하거나 오랜 시간 동안 작동할 경우 인스턴스의 버스트 용량을 모두 사용하게 됩니다. 이는 인스턴스가 과소 프로비저닝되었음을 의미할 수 있습니다. 서비스가 너무 자주 실행되거나 인스턴스에서 불필요한 소프트웨어를 실행 중일 수도 있습니다.

Linux/Unix 인스턴스의 top 및 Windows Server 인스턴스의 작업 관리자와 같은 도구를 사용하여 인스턴스가 버스트되는 원인을 조사합니다. 이러한 도구는 인스턴스에서 리소스를 소비하는 서비스를 보여 줍니다. 가장 많은 리소스를 소비하는 서비스를 확인하고, 인스턴스의 워크로드에 영향을 주지 않고 비활성화할 수 있는지 확인합니다. 서비스 비활성화나 소프트웨어 제거를 통해 인스턴스의 버스트를 줄일 수 있으므로 인스턴스 크기를 늘리지 않아도 됩니다.

인스턴스가 실제로 과소 프로비저닝되고 CPU 사용률을 낮출 수 없는 경우 처리 능력을 추가하여 버스트 용량 소비를 줄일 수 있습니다. 이렇게 하려면 인스턴스의 스냅샷을 생성한 다음 더 큰 Lightsail 인스턴스 요금제를 사용하여 스냅샷에서 새 인스턴스를 생성합니다. 예를 들어, 이전 인스턴스에 사용한 Linux 또는 Unix 기반 월 12 USD 요금제 대신 새 인스턴스에 월 Linux 또는 Unix 기반 24 USD 플랜을 사용합니다. 새 인스턴스가 실행 중일 때 필요에 따라 워크로드의 DNS를 변경하여 이전 인스턴스를 새 인스턴스로 바꿉니다. 트래픽이 새 인스턴스로 라우팅되기 시작한 후 과소 프로비저닝된 이전 인스턴스를 삭제합니다. 자세한 내용은 스냅샷을 참조하세요.

Lightsail 인스턴스 연결 및 관리

이 가이드에서는 Amazon Lightsail 인스턴스 관리 및 연결과 관련된 다음 주제를 다룹니다.

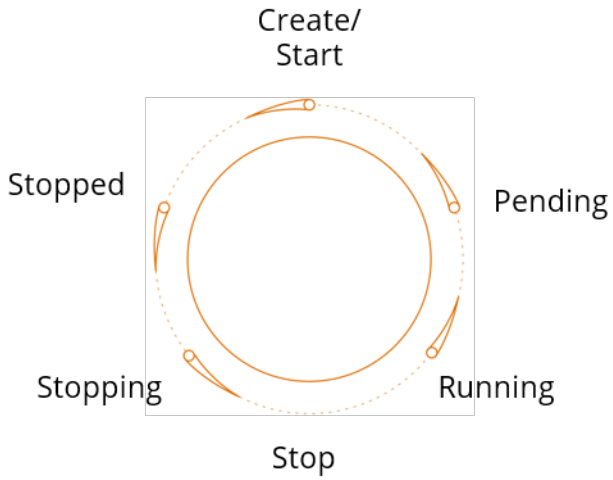
주제

- Lightsail 인스턴스 시작, 중지 또는 재부팅
- 강제 중지 멈춤 Lightsail 인스턴스

- Amazon EC2 인스턴스를 위한 활성화된 향상된 네트워킹
- Lightsail에서 Windows Server 인스턴스의 파일 시스템 확장
- Lightsail에서 시작 스크립트를 사용하여 Linux/Unix 인스턴스 구성
- Windows PowerShell 및 배치 스크립트를 사용하여 Windows Lightsail 인스턴스 구성
- Lightsail의 Windows Server 인스턴스 보호

Lightsail 인스턴스 시작, 중지 또는 재부팅

Amazon Lightsail에서 인스턴스를 생성할 때 시스템은 대기 중 상태로 들어간 후 실행을 시작합니다. 인스턴스 실행 후, 인스턴스를 재부팅하거나 중지했다가 다시 시작할 수 있습니다. 이 주기는 다음과 같습니다.

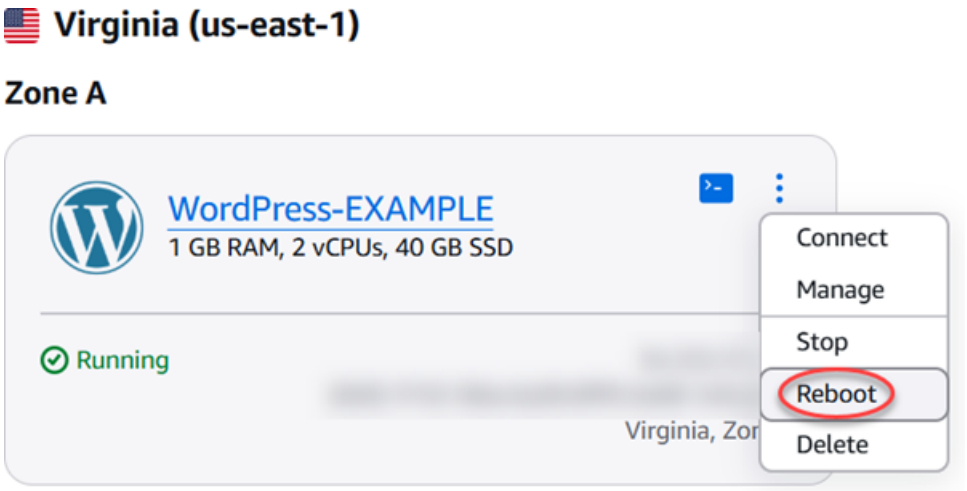


인스턴스를 관리하거나 홈 페이지에서 인스턴스를 볼 때 인스턴스 상태를 확인할 수 있습니다.

> **Important**
>
> 인스턴스를 생성할 때 인스턴스에 할당된 기본 퍼블릭 IPv4 주소는 인스턴스를 중지하고 시작할 때 변경됩니다. 고정 IPv4 주소를 생성하여 인스턴스에 연결할 수도 있습니다. 고정 IPv4 주소는 인스턴스의 기본 퍼블릭 IPv4 주소를 대체하며 인스턴스를 중지하고 시작할 때 동일하게 유지됩니다. 자세한 내용은 고정 IP를 생성하여 인스턴스에 연결을 참조하세요.

실행 중인 인스턴스 재부팅

- 홈 페이지에서 재부팅하려는 인스턴스를 선택하거나 인스턴스 관리 메뉴에서 재부팅을 선택합니다.

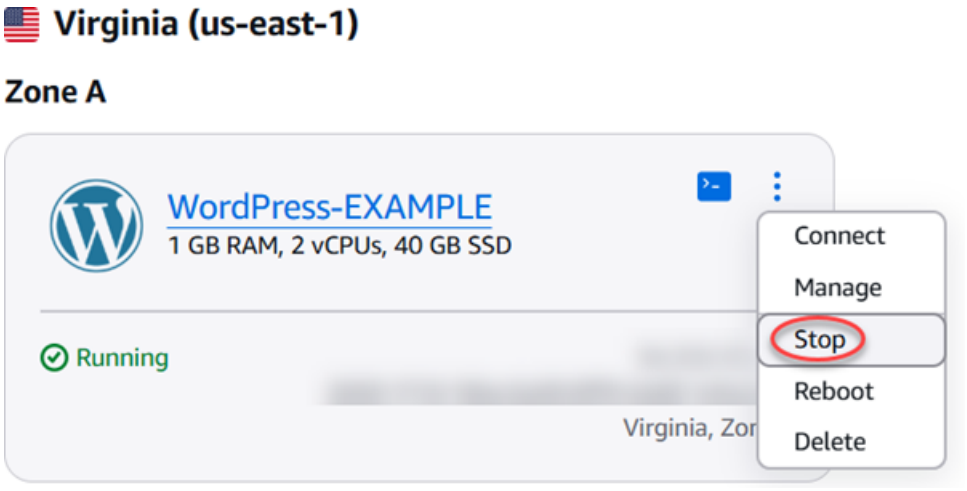


인스턴스 관리 페이지에서 인스턴스를 보고 있는 경우 재부팅을 선택한 후 메시지가 표시되면 확인을 선택합니다.

> **Note**
>
> 인스턴스를 재부팅하려면 실행 중 상태에 있어야 합니다.

실행 중인 인스턴스 중지

- 홈 페이지에서 중지하려는 인스턴스를 선택하거나 인스턴스 관리 메뉴에서 중지를 선택합니다.

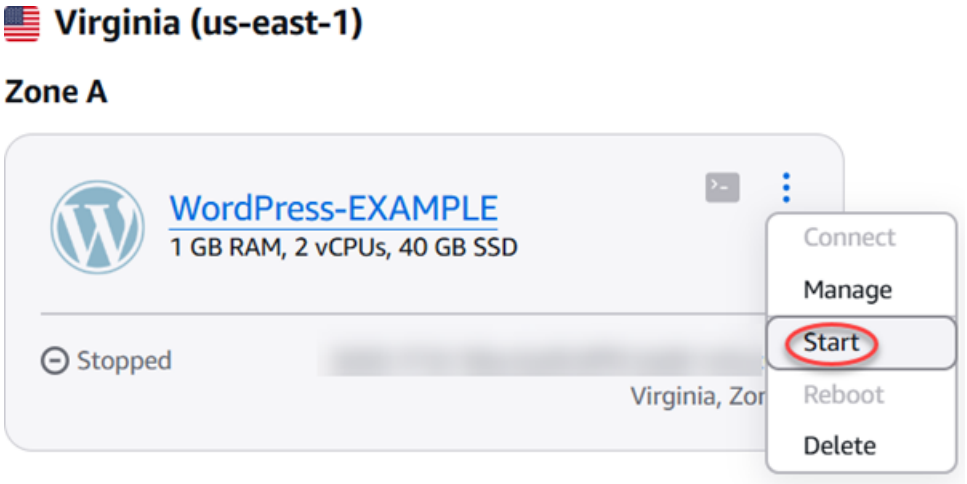


인스턴스 관리 페이지에서 인스턴스를 볼 경우 중지를 선택한 후 메시지가 표시되면 확인을 선택합니다.

> **Note**
>
> 인스턴스를 중지하려면 실행 중 상태에 있어야 합니다.

중지된 인스턴스 시작

- 홈 페이지에서 시작하려는 인스턴스를 선택하거나 인스턴스 관리 메뉴에서 시작을 선택합니다.



인스턴스 관리 페이지에서 인스턴스를 볼 경우 시작을 선택합니다.

> **Note**
>
> 인스턴스를 시작하려면 중지됨 상태에 있어야 합니다.

강제 중지 멈춤 Lightsail 인스턴스

드문 경우지만 인스턴스가 `Stopping` 상태에서 멈출 수 있습니다. 이 경우 Amazon Lightsail 인스턴스를 호스팅하는 기본 하드웨어에 문제가 있을 수 있습니다. 이 가이드에서는 `stopping` 상태에 멈춘 인스턴스를 강제 중지하는 방법을 알아봅니다. 인스턴스 상태에 대한 자세한 내용은 Lightsail 인스턴스 시작, 중지 또는 재시작을 참조하세요.

인스턴스를 강제 중지하는 방법

Lightsail 콘솔을 사용하여 인스턴스를 강제 중지할 수 있습니다. 단, `stopping` 상태에 있는 인스턴스만 가능합니다. 또는 AWS Command Line Interface (AWS CLI)를 사용하여 인스턴스가 `shutting-`

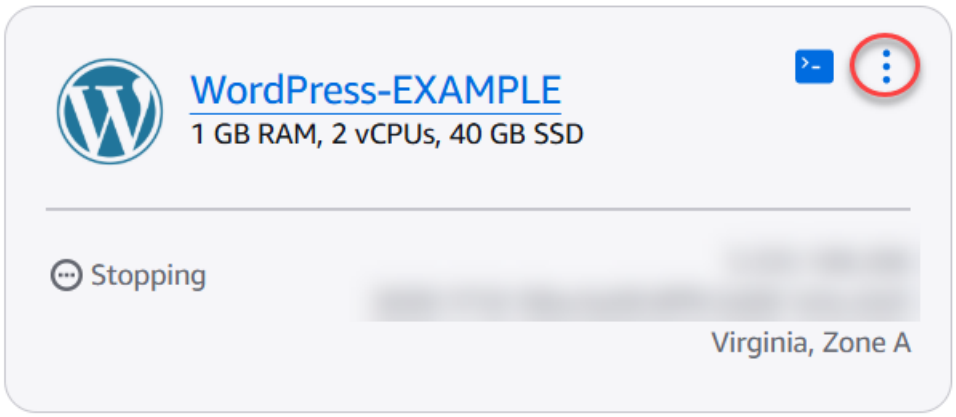
down 및 terminated 이외의 상태에 있는 동안 인스턴스를 강제 중지할 수 있습니다. 강제 중지가 완료되는 데 몇 분 정도 걸릴 수 있습니다. 10분이 지나도 인스턴스가 중지되지 않으면 다시 강제 중지하세요.

인스턴스를 강제 중지해도 파일 시스템 캐시 또는 파일 시스템 메타데이터를 플러시하지 않습니다. 인스턴스를 강제 중지한 후에는 파일 시스템 검사 및 복구 절차를 수행해야 합니다.

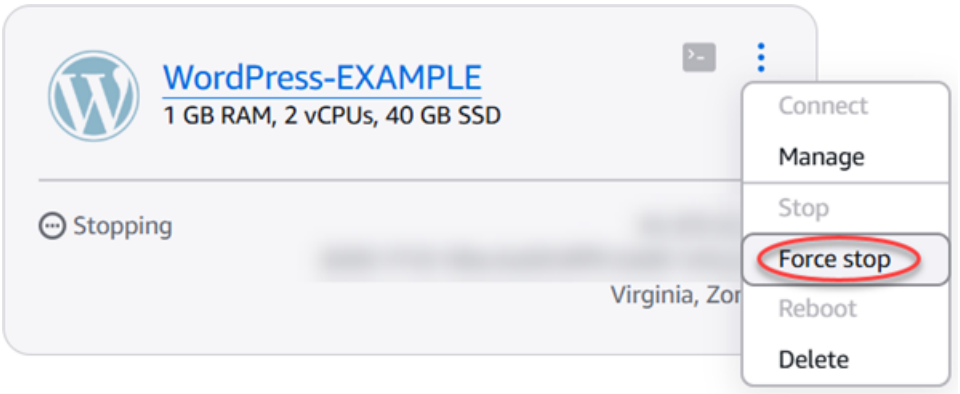
다음 절차에서는 Lightsail 인스턴스를 강제 중지할 수 있는 여러 가지 방법을 설명합니다.

Lightsail 콘솔에서 인스턴스 강제 중지

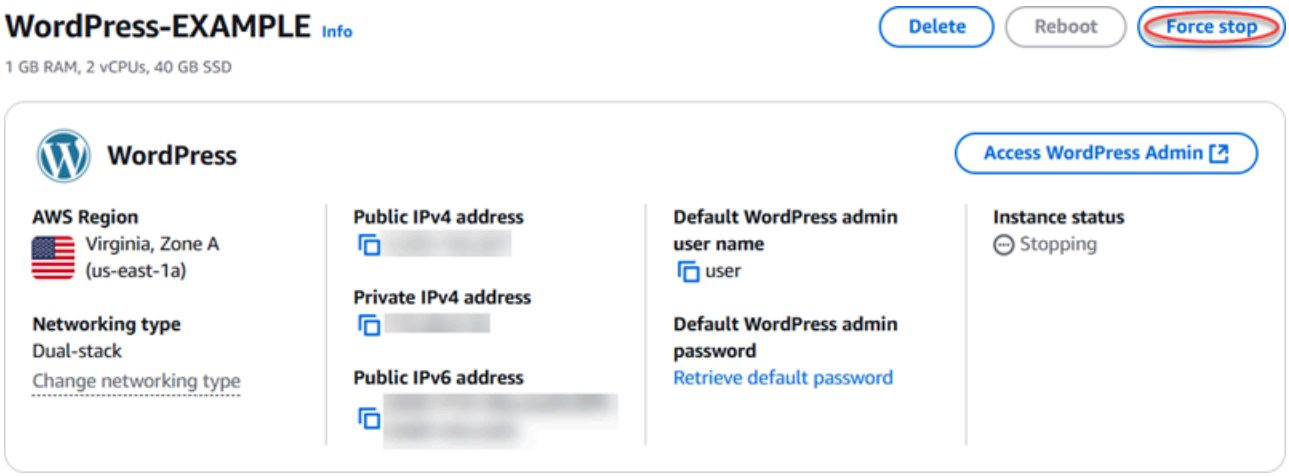
1. Lightsail 콘솔에 로그인합니다.
2. [Instances] 탭을 선택합니다.
3. Stopping 상태에서 멈춘 인스턴스를 찾습니다. 그런 다음 인스턴스 이름 옆에 표시된 작업 메뉴 아이콘(:)을 선택합니다.



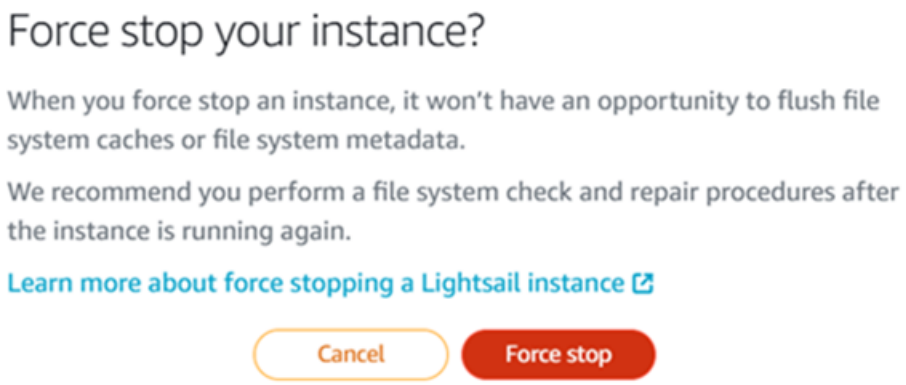
4. 표시되는 드롭다운 목록에서 강제 중지를 선택합니다.



또는 인스턴스 이름을 선택하여 인스턴스 관리 페이지에 액세스할 수 있습니다. 그런 다음 강제 중지 버튼을 선택합니다.



5. 이 작업에 대한 고려 사항을 검토합니다. 계속하려면 강제 중지를 선택합니다.



를 사용하여 인스턴스 강제 중지 AWS CLI

1. 시작하기 전에 AWS CLI를 설치해야 합니다. 자세한 내용은 AWS Command Line Interface설치 섹션을 참조하세요. AWS CLI를 설치한 후 구성해야 합니다.
2. 다음과 같이 stop-instances 명령과 `--force` 파라미터를 사용합니다.

```
aws lightsail stop-instance --instance-name Wordpress-1 --force
```

Amazon EC2 인스턴스를 위한 활성화된 향상된 네트워킹

일부 Lightsail 인스턴스는 향상된 네트워킹에 대해 활성화되지 않았으므로 현재 세대 EC2 인스턴스 유형(T3, M5, C5 또는 R5)과 호환되지 않습니다. 소스 Lightsail 인스턴스가 호환되지 않는 경우 내보낸 스냅샷에서 EC2 인스턴스를 생성할 때 이전 세대 인스턴스 유형(T2, M4, C4 또는 R4)을 선택해야 합

니다. 이러한 인스턴스 유형 옵션은 Lightsail 콘솔에서 Amazon EC2 인스턴스 생성 페이지를 사용하여 EC2 인스턴스를 생성할 때 표시됩니다.

> **Note**
>
> 향상된 네트워킹에 대한 자세한 내용은 Amazon EC2 설명서의 Linux의 향상된 네트워킹 또는 Windows의 향상된 네트워킹을 참조하세요.

소스 Lightsail 인스턴스가 호환되지 않는 경우 최신 세대 EC2 인스턴스 유형을 사용하려면 이전 세대 인스턴스 유형(T2, M4, C4 또는 R4)을 사용하여 새 EC2 인스턴스를 생성하고, 인스턴스에서 네트워킹 드라이버를 업데이트하고, 인스턴스를 원하는 현재 세대 인스턴스 유형으로 업그레이드해야 합니다.

사전 조건

내보낸 Lightsail 스냅샷에서 Amazon EC2 인스턴스를 생성해야 합니다. Lightsail 인스턴스가 호환되지 않는 경우 Amazon EC2 인스턴스를 생성할 때 이전 세대 인스턴스 유형(T2, M4, C4 또는 R4)을 선택합니다. 자세한 내용은 Lightsail의 내보낸 스냅샷에서 Amazon EC2 인스턴스 생성을 참조하세요.

새 EC2 인스턴스가 실행된 후 이 안내서의Elastic Network Adapter를 사용하여 향상된 네트워킹 활성화 단원으로 계속 진행하여, 향상된 네트워킹 활성화 방법을 알아봅니다.

ENA(Elastic Network Adapter)를 사용하여 향상된 네트워킹 활성화

새 인스턴스가 실행되면 Amazon EC2 설명서의 다음 가이드 중 하나를 참조해 Elastic Network Adapter(ENA)를 사용하여 향상된 네트워킹을 활성화합니다.

- Linux 인스턴스에서 ENA를 사용하여 향상된 네트워킹 활성화
- Windows 인스턴스에서 ENA를 사용하여 향상된 네트워킹 활성화

인스턴스 유형 업그레이드

향상된 네트워킹을 활성화한 후에는 다음 가이드 중 하나의 지침에 따라 인스턴스 유형을 업그레이드할 수 있습니다.

- Windows Server 인스턴스의 경우 - 최신 세대 인스턴스 유형으로 마이그레이션
- Linux 또는 Unix 인스턴스의 경우 - 인스턴스 유형 변경

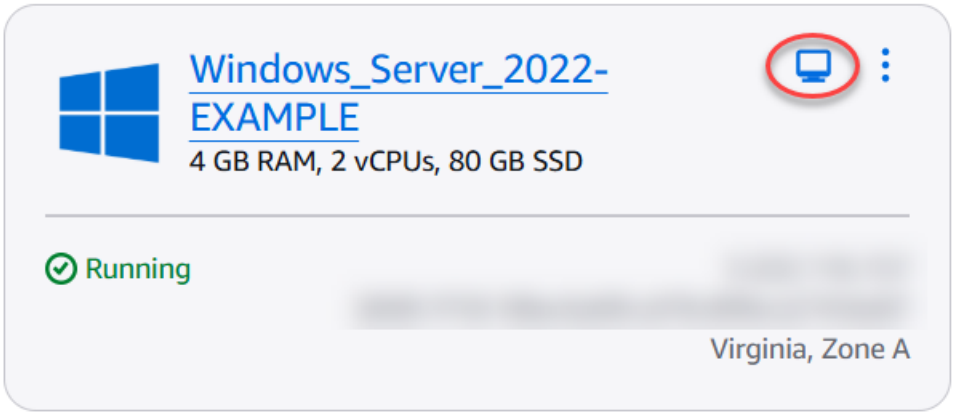
Lightsail에서 Windows Server 인스턴스의 파일 시스템 확장

스냅샷을 사용하여 더 큰 규모의 플랜이 있는 새 Windows Server 인스턴스를 만든 후에는 사용 가능한 스토리지 공간이 플랜에서 지정된 것보다 작을 수 있습니다. 이는 대개 더 큰 규모의 플랜에서 제공하는 추가 스토리지 공간이 할당되지 않아 활성 볼륨에서 사용되고 있지 않기 때문입니다. 이 항목의 단계에서는 사용 가능한 최대 스토리지 공간을 사용하도록 Windows Server 인스턴스의 파일 시스템을 확장하는 방법을 보여줍니다.

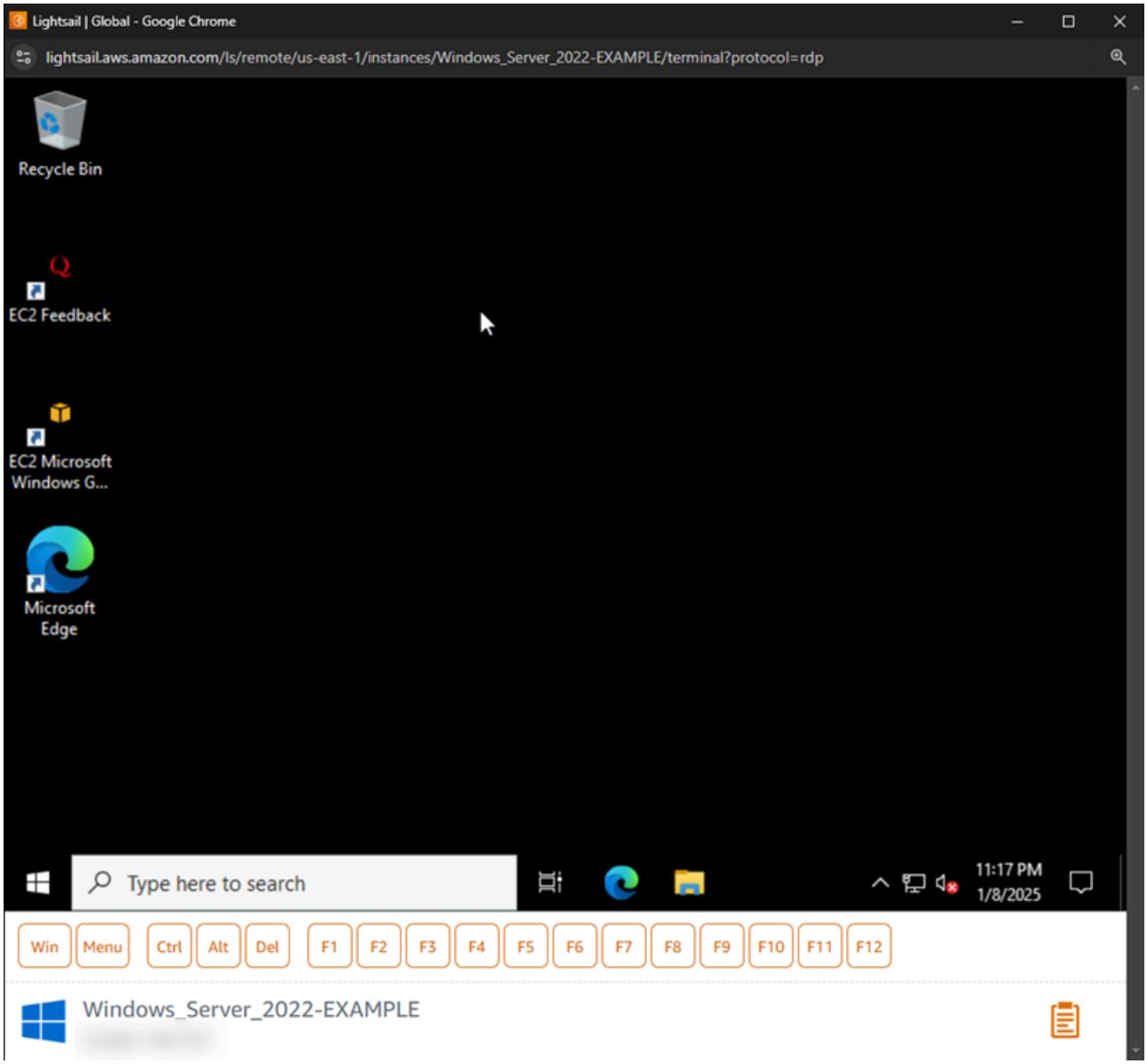
> **Note**
>
> 이 시나리오는 시스템 준비(Sysprep) 유틸리티를 실행하기 전에 생성된 스냅샷을 사용하여 Windows Server 인스턴스를 생성하는 경우에만 발생합니다. 자세한 내용은 Windows Server 인스턴스의 스냅샷 생성을 참조하세요.

Windows Server 인스턴스에 대한 파일 시스템을 확장하려면

1. Lightsail 콘솔에 로그인합니다.
2. Lightsail 홈 페이지에서 연결할 인스턴스에 대한 RDP 클라이언트 아이콘을 선택합니다.



브라우저 기반 RDP 클라이언트 창이 열리고 다음 예처럼 표시됩니다.

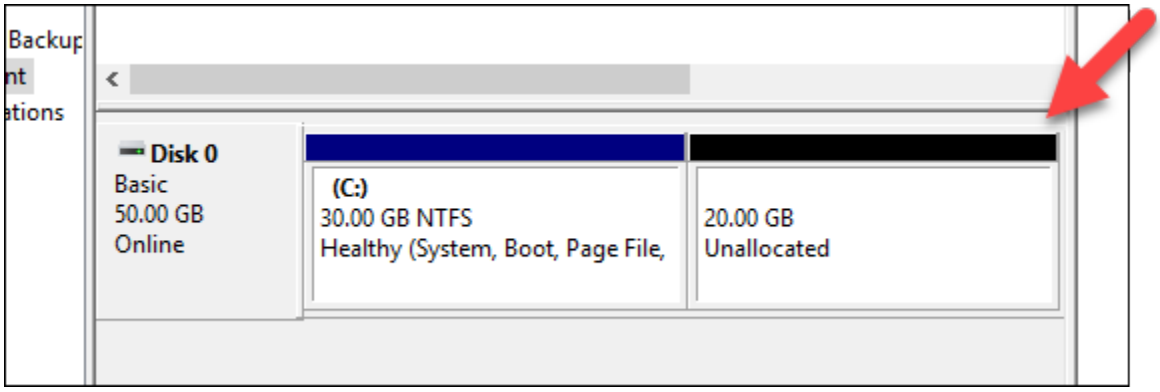


3. 작업 표시줄에서 Windows 아이콘을 선택하고 다음 옵션 중 하나를 선택합니다.

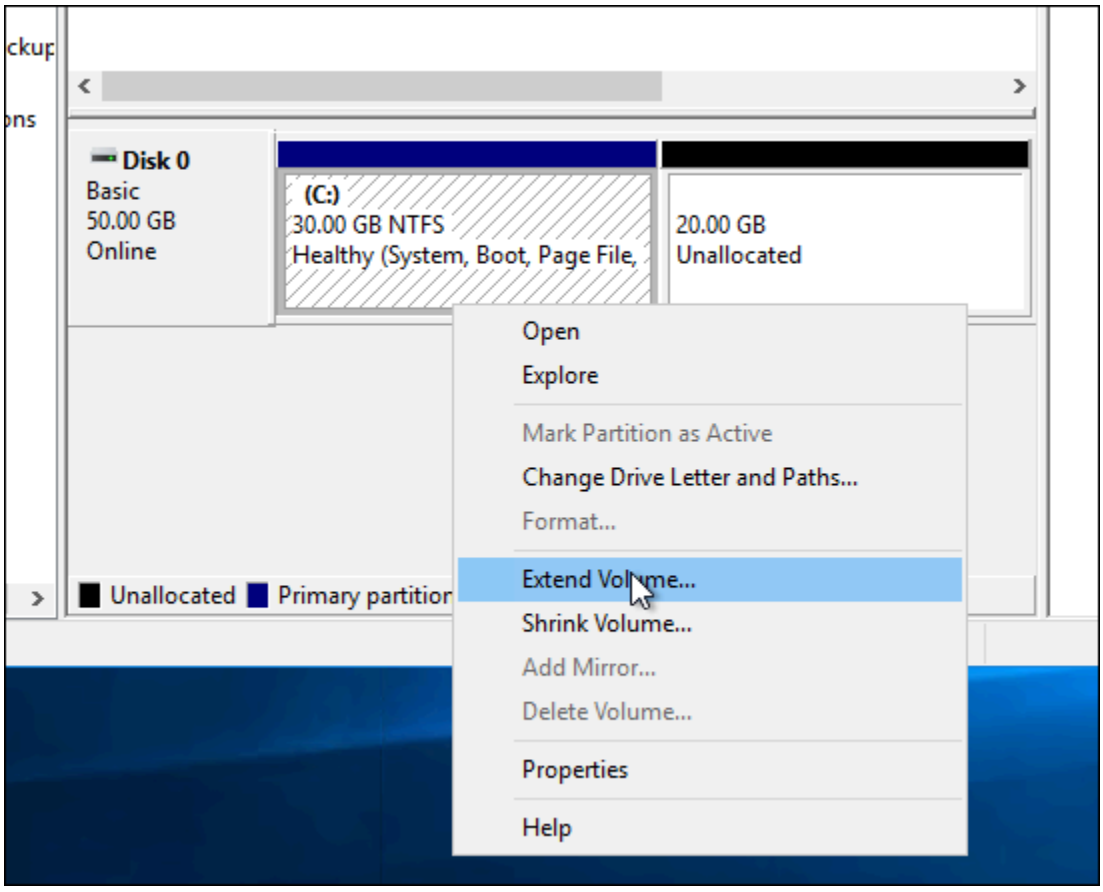
   - Windows Server 2022, Windows Server 2019 및 Windows Server 2016 인스턴스에서는 시작을 선택하고 Windows 관리 도구를 선택합니다.

4. 컴퓨터 관리를 닫습니다.
5. 컴퓨터 관리 콘솔의 왼쪽 창에서 디스크 관리를 선택합니다.
6. 작업 메뉴에서 디스크 다시 스캔을 선택합니다.

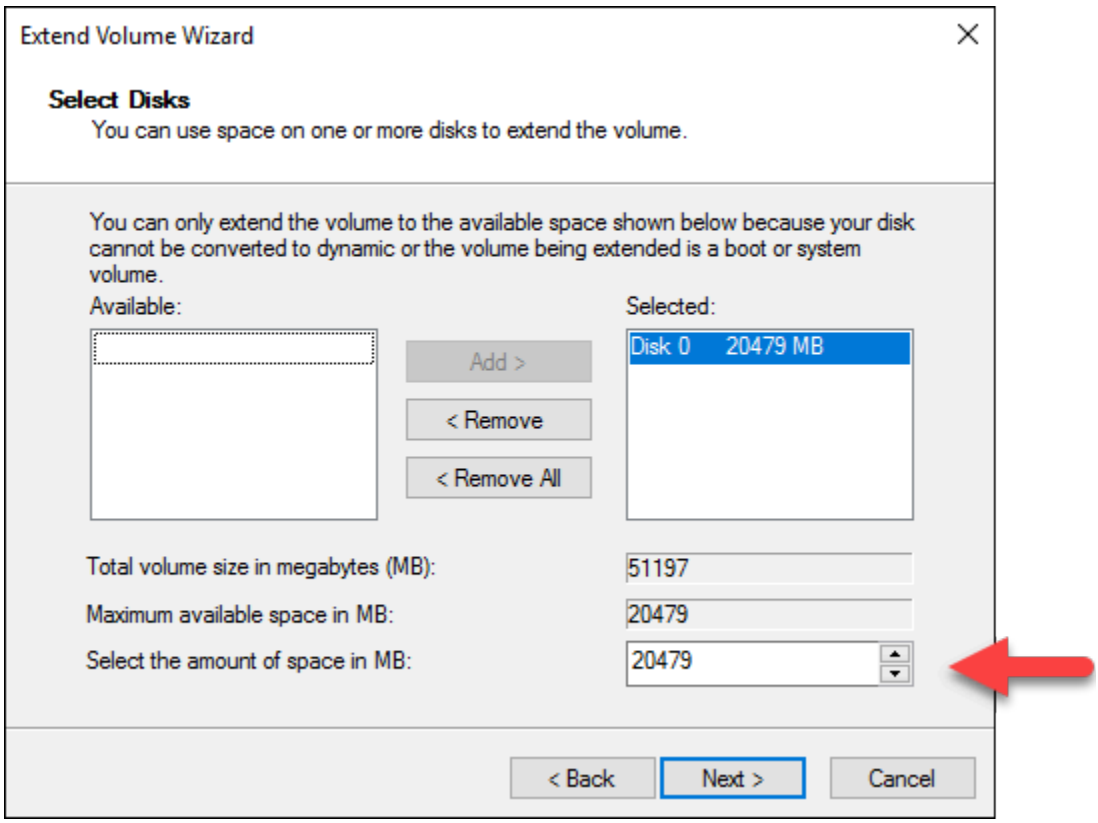
   디스크와 관련된 할당되지 않은 공간이 나타날 수 있습니다. 할당되지 않은 공간을 사용하려면 디스크의 활성 볼륨을 확장합니다.



7. 할당되지 않은 공간과 동일한 디스크의 활성 볼륨을 마우스 오른쪽 버튼으로 클릭한 다음 볼륨 확장을 선택합니다.

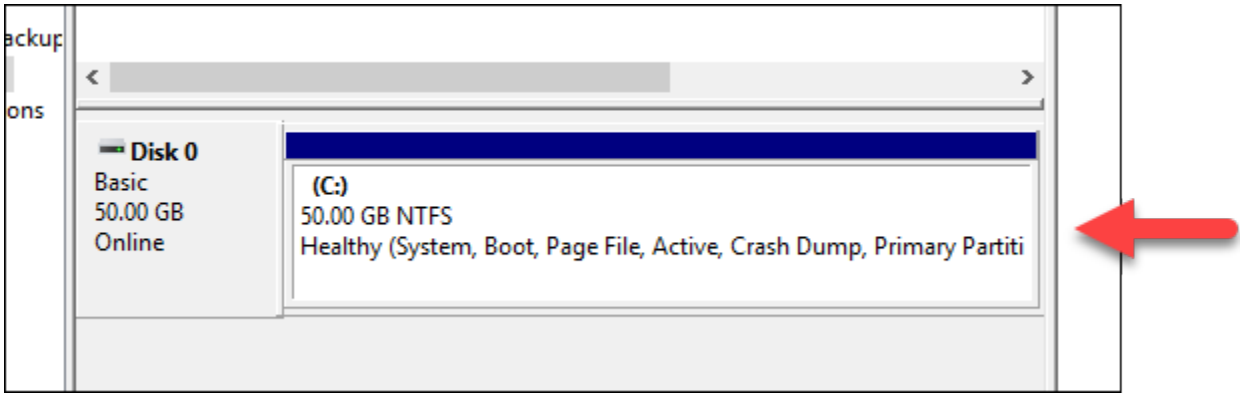


8. 볼륨 확장 마법사가 열리면 다음을 선택합니다.
9. MB 단위로 공간 크기 선택 필드에 볼륨 확장에 적용할 메가바이트 수를 입력합니다. 일반적으로 이 값을 최대 미할당 공간으로 설정합니다. 입력하는 값은 볼륨의 최종 크기가 아니라 추가할 공간의 크기입니다.



10. 볼륨 확장 마법사를 완료합니다.

    지정한 미할당 공간을 사용하도록 활성 볼륨이 확장됩니다. 다음 예시에서는 선택한 미할당 공간을 모두 보여줍니다.



Lightsail에서 시작 스크립트를 사용하여 Linux/Unix 인스턴스 구성

Linux/Unix 기반의 인스턴스를 생성할 때 시작 스크립트를 추가하여 소프트웨어 추가 및 업데이트 또는 인스턴스 구성 등의 작업을 다른 방식으로 수행할 수 있습니다. 추가 데이터가 있는 Windows 기반

의 인스턴스를 구성하는 방법은 Windows PowerShell을 사용하여 새 Lightsail 인스턴스 구성을 참조하십시오.

> **Note**
>
> 선택하는 시스템 이미지에 따라 인스턴스에서 소프트웨어를 가져오는 명령이 달라집니다. Amazon Linux는 `yum`을 사용하는 반면, Debian과 Ubuntu는 둘 다 `apt-get`을 사용합니다. WordPress 및 기타 애플리케이션 이미지는 Debian을 운영 체제로 실행하기 때문에 `apt-get`을 사용합니다. FreeBSD 및 openSUSE는 `freebsd-update` 또는 `zypper`(openSUSE)와 같은 사용자 지정 도구를 사용하려면 추가 사용자 구성이 필요합니다.

예: Node.js를 설치하도록 Ubuntu 서버 구성

다음 예에서는 패키지 목록을 업데이트한 다음에 `apt-get` 명령을 통해 Node.js를 설치합니다.

1. 인스턴스 생성(Create an instance) 페이지에서 OS 전용(OS Only) 탭의 Ubuntu를 선택합니다.
2. 아래로 스크롤하여 시작 스크립트 추가를 선택합니다.
3. 다음을 입력합니다.

```
# update package list
apt-get update -y
# install some of my favorite tools
apt-get install nodejs -y
```

> **Note**
>
> 서버를 구성하려고 보내는 명령은 루트로서 실행되므로, 명령 앞에 `sudo`를 포함할 필요가 없습니다.

4. 인스턴스 생성을 선택합니다.

예: 플러그인을 다운로드하고 설치하도록 WordPress 서버 구성

다음 예에서는 패키지 목록을 업데이트한 다음에 WordPress용 BuddyPress 플러그인을 다운로드하고 설치합니다.

1. 인스턴스 생성 페이지에서 WordPress를 선택합니다.

2. 시작 스크립트 추가를 선택합니다.
3. 다음을 입력합니다.

```
# update package list
apt-get update
# download wordpress plugin
wget "https://downloads.wordpress.org/plugin/buddypress.14.0.0.zip"
apt-get install unzip
# unzip into wordpress plugin directory
unzip buddypress.14.0.0.zip -d /bitnami/wordpress/wp-content/plugins
```

4. 인스턴스 생성을 선택합니다.

Windows PowerShell 및 배치 스크립트를 사용하여 Windows Lightsail 인스턴스 구성

Windows 기반의 인스턴스를 생성할 때 Windows PowerShell 스크립트나 기타 배치 스크립트를 사용하여 이를 구성할 수 있습니다. 이것은 인스턴스를 시작하는 즉시 실행되는 일회용 스크립트입니다. 이 주제에서는 이러한 스크립트의 구문을 소개하고 예제를 제시하여 사용자의 시작 과정을 돕습니다. 이와 함께 스크립트가 제대로 실행되는지 확인하기 위한 테스트 방법도 알려 드립니다.

PowerShell 스크립트를 시작 및 실행하는 인스턴스 생성

다음은 새 인스턴스가 시작된 직후 그 인스턴스에 chocolatey라는 도구를 설치하는 절차입니다.

1. 왼쪽 탐색 창에서 인스턴스 생성을 선택합니다.
2. 인스턴스를 생성할 AWS 리전 및 가용 영역을 선택합니다.
3. 플랫폼 선택에서 Microsoft Windows를 선택합니다.
4. OS 전용을 선택한 다음 Windows Server 2022, Windows Server 2019, Windows Server 2016을 선택합니다.
5. 시작 스크립트 추가를 선택합니다.
6. 다음을 입력합니다.

```
<powershell>
iex ((New-Object System.Net.WebClient).DownloadString('https://chocolatey.org/
install.ps1'))
</powershell>
```

> **Note**
>
> PowerShell 스크립트는 항상 <powershell></powershell> 태그로 묶어야 합니다. PowerShell이 아닌 명령이나 배치 스크립트는 <script></script> 태그를 사용하거나 아예 태그를 사용하지 않고 입력할 수 있습니다.

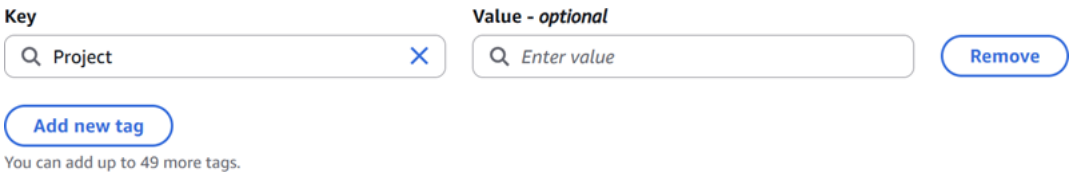
7. 인스턴스 이름을 입력합니다.

   리소스 이름:

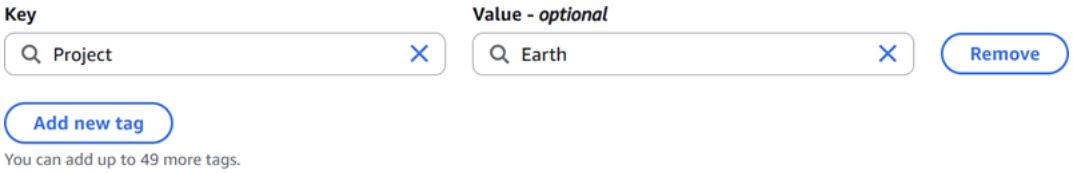
   - Lightsail 계정 AWS 리전 의 각 내에서 고유해야 합니다.
   - 2~255자의 문자로 구성되어야 합니다.
   - 영숫자 문자 또는 숫자로 시작하고 끝나야 합니다.
   - 영숫자 문자, 숫자, 마침표, 대시, 밑줄이 포함될 수 있습니다.

8. (선택 사항) 인스턴스에 태그를 추가하려면 새 태그 추가를 선택합니다. 필요에 따라 이 단계를 반복하여 모든 태그를 추가합니다. 태그 사용에 대한 자세한 내용은 태그를 참조하세요.

   a. 키에는 태그 키를 입력합니다.

   b. (선택 사항) 값에 태그 값을 입력합니다.

9. 인스턴스 생성을 선택합니다.

스크립트가 제대로 실행되었는지 확인

인스턴스에 로그인하여 스크립트가 제대로 실행되었는지 확인할 수 있습니다. Windows 기반의 인스턴스가 RDP 연결을 수락할 준비가 될 때까지는 최대 15분이 걸릴 수 있습니다. 준비가 되면 브라우저 기반의 RDP 클라이언트를 사용하여 로그인하거나 자체 RDP 클라이언트를 구성하세요. 자세한 내용은 Windows 기반 인스턴스에 연결을 참조하세요.

1. Lightsail 인스턴스에 연결되었으면 명령 프롬프트(또는 Windows 탐색기)를 엽니다.
2. 다음을 입력하여 Log 디렉터리로 이동합니다.

```
cd C:\ProgramData\Amazon\EC2-Windows\Launch\Log
```

3. 텍스트 편집기에서 `UserdataExecution.log`를 열거나 `type UserdataExecution.log`를 입력합니다.

   로그 파일에는 다음 내용이 표시됩니다.

```
2017/10/11 20:32:12Z: <powershell> tag was provided.. running powershell content
2017/10/11 20:32:13Z: Message: The output from user scripts: iex ((New-Object
 System.Net.WebClient).DownloadString('https://chocolatey.org/install.ps1'))

2017/10/11 20:32:13Z: Userdata execution done
```

Lightsail의 Windows Server 인스턴스 보호

이 문서에서는 Lightsail 인스턴스로 Windows Server를 실행할 때 보안 위험을 피할 수 있는 팁과 요령을 제공합니다.

Lightsail 암호 정보

사용자가 Windows Server 기반의 인스턴스를 생성하면 Lightsail은 짐작하기 어려운 긴 암호를 무작위로 작성합니다. 이 암호를 새 인스턴스의 고유 암호로 사용하면 됩니다. RDP(원격 데스크톱)을 사용하여 인스턴스에 신속하게 기본 암호로 연결할 수 있습니다. 사용자는 항상 Lightsail 인스턴스에서 관리자로 로그인합니다.

암호 변경

Windows Server 기반 인스턴스의 암호를 변경할 수 있습니다. 이 기능은 원격 데스크톱 클라이언트를 사용하여 Lightsail 인스턴스에 액세스하려는 경우 유용합니다. Lightsail은 결코 사용자가 만든 암호를 저장하지 않습니다.

> **Note**
>
> Lightsail의 브라우저 기반 RDP 클라이언트에 Lightsail이 생성한 암호를 사용하거나 사용자 지정 자체 암호를 사용할 수 있습니다. 사용자 지정 암호를 사용하는 경우, 로그인할 때마다 암호

를 묻는 메시지가 나타납니다. 그러므로 인스턴스에 빠르게 액세스하려면 Lightsail에서 생성된 기본 암호를 브라우저 기반 RDP 클라이언트에 사용하는 것이 좋습니다.

암호를 안전하게 변경하려면 Windows Server 암호 관리자를 사용하십시오. `Ctrl+Alt+Del`를 누른 다음 암호 변경을 선택하면 됩니다. Lightsail에는 암호가 저장되지 않으므로 암호를 기록해 두십시오. 암호를 검색해야 하는 경우 Windows 기반 인스턴스의 관리자 암호 변경을 참조하세요.

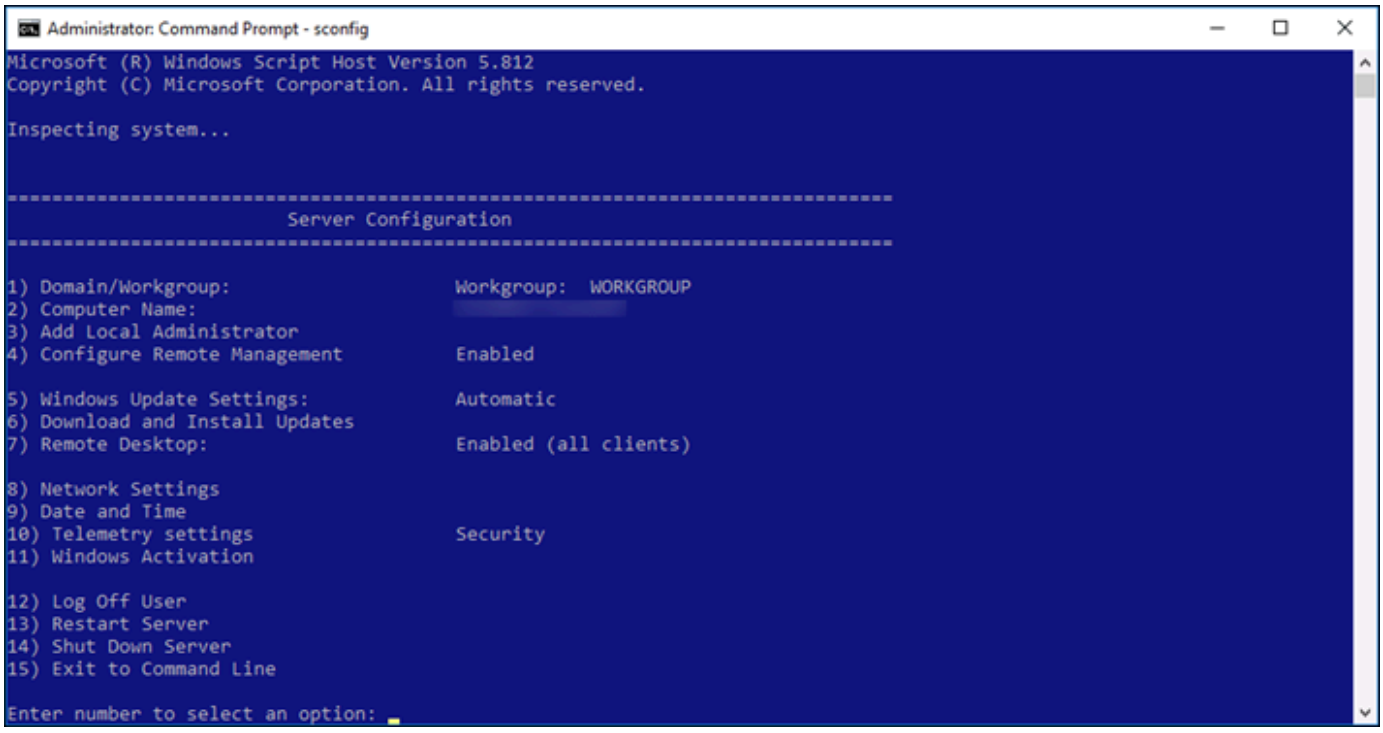
고유한 기본 암호에서 다른 암호로 변경하려면 강력한 암호를 사용하십시오. 이름이나 디렉터리의 단어, 글자 조합의 반복으로 이루어진 암호는 피해야 합니다.

보안 패치

Windows Server 기반의 Lightsail 인스턴스를 항상 최신 보안 패치로 업데이트해 두는 것이 좋습니다. 업데이트를 다운로드하고 설치하도록 서버를 구성하십시오. 다음 절차는 Windows Server를 실행하는 Lightsail 인스턴스에서 직접 이 작업을 수행하는 방법입니다.

1. Windows Server 기반의 인스턴스에서 명령 프롬프트를 엽니다.
2. `sconfig`를 입력하고 `Enter`를 누릅니다.

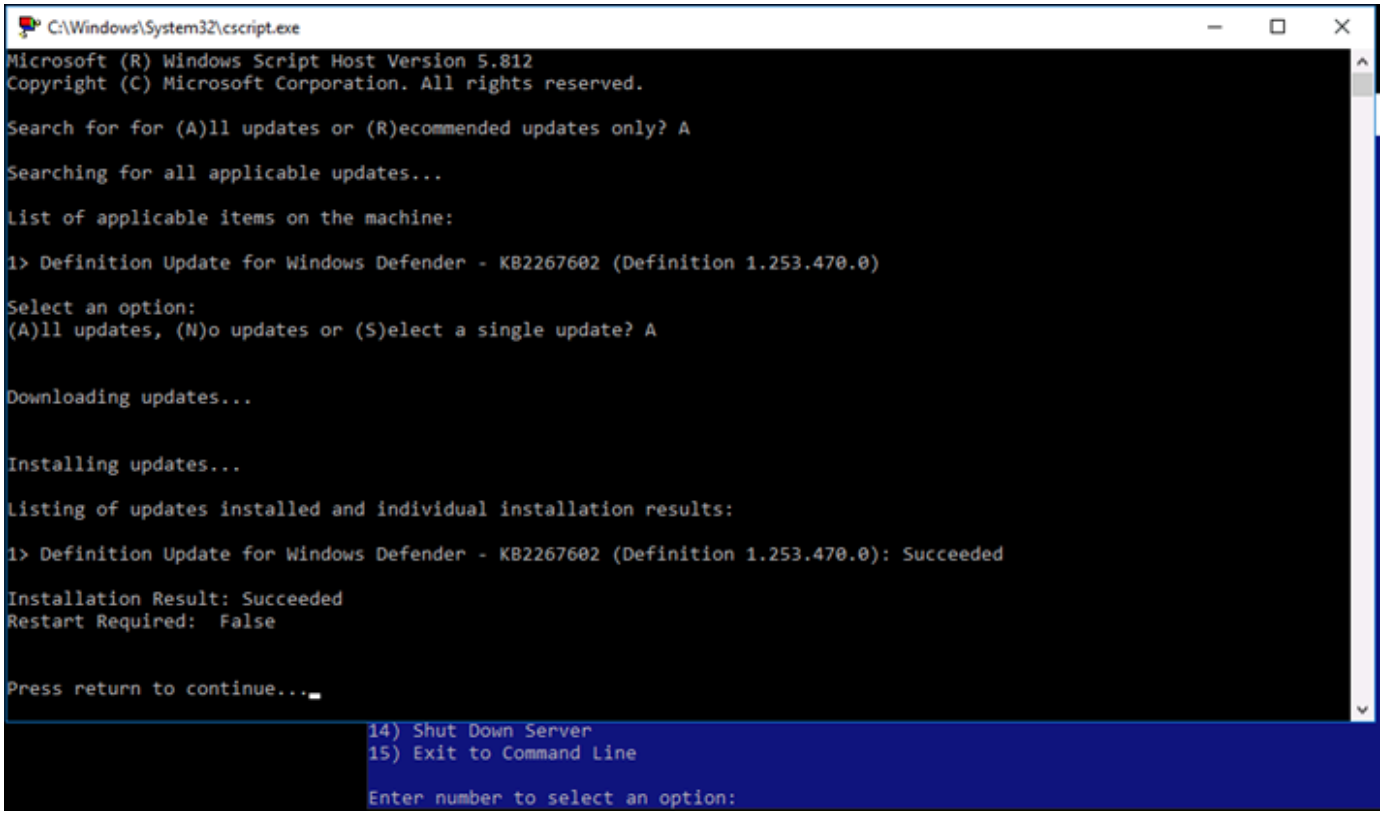
   Windows 업데이트 설정(5번)은 기본적으로 `Automatic` 상태입니다.



3. 새 업데이트를 다운로드하여 설치하려면 6을 입력하고 `Enter`를 누릅니다.

4. 새 명령 창에서 A를 입력하여 (A)ll updates(모든 업데이트)를 검색한 다음 Enter를 누릅니다.
5. 다시 A를 입력하여 (A)ll updates(모든 업데이트)를 설치한 다음 Enter를 누릅니다.

   작업이 끝나면 설치 결과 및 추가 지침(해당하는 경우)을 담은 메시지가 표시됩니다.



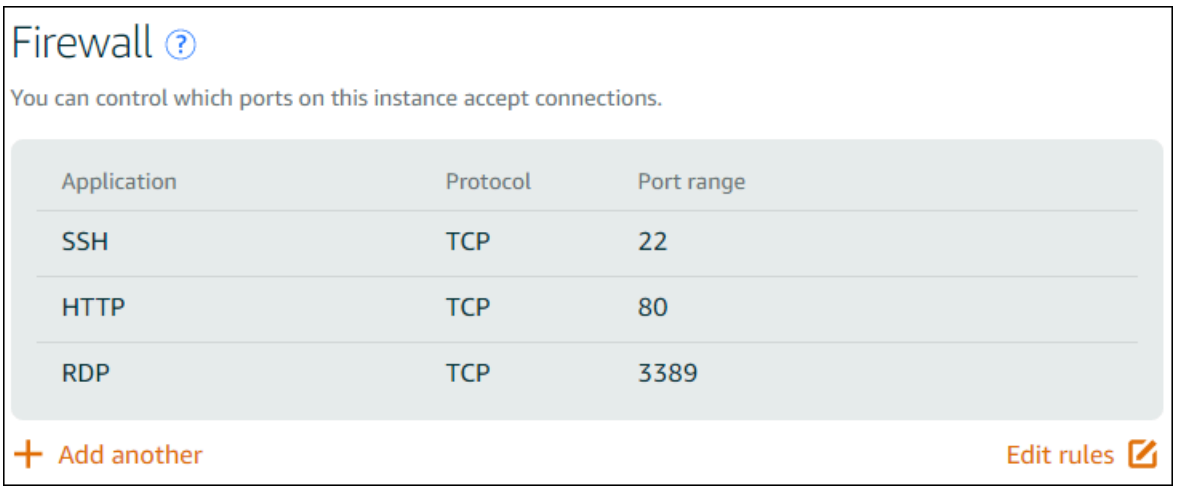
Windows Server에서 계정 잠금 정책 활성화

정해진 횟수만큼 로그인 시도에 실패하면 일시적으로 또는 영구적으로 계정을 비활성화하도록 Windows Server를 구성할 수 있습니다. 예를 들어, 잘못된 암호로 세 번이나 인스턴스에 로그인하려고 한 사람을 잠금 처리할 수 있습니다.

자세한 내용은 Windows Server 문서의 계정 잠금 정책을 참조하세요.

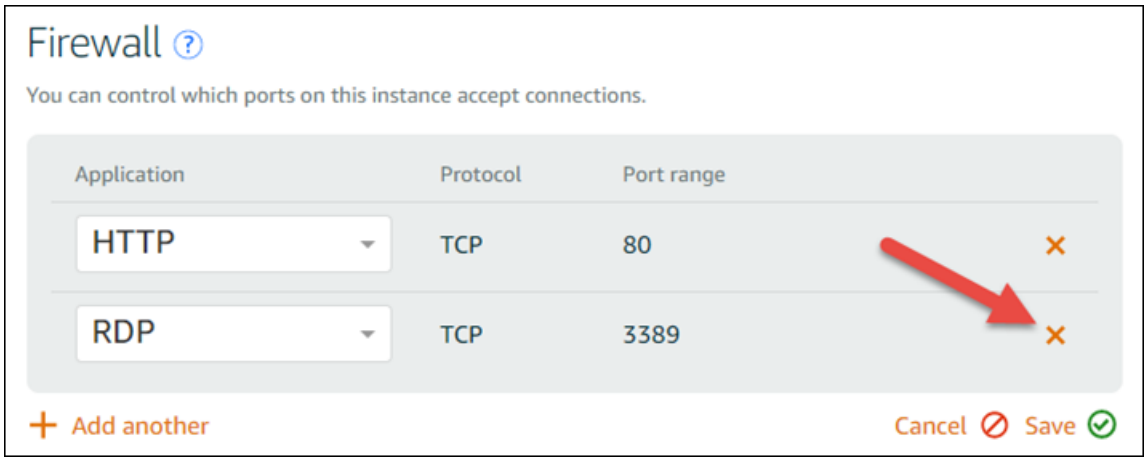
포트 및 방화벽 설정

기본적으로, Windows Server 기반의 인스턴스에서 다음 포트를 열어 둡니다.



활성화한 포트는 전 세계에 노출되며, 소스 IP를 기준으로 제한할 수 없습니다. 인스턴스에 대한 액세스를 제한하려면 이러한 포트를 꺼 두었다가, 인스턴스에 액세스해야 할 때만 활성화하는 방법이 있습니다. 그 방법은 다음과 같습니다.

1. Lightsail에서 관리할 인스턴스를 찾은 다음 관리를 선택합니다.
2. 네트워킹을 선택합니다.
3. 인스턴스의 네트워킹 페이지에서 규칙 편집을 선택합니다.
4. RDP/TCP/3389 규칙 옆에 있는 주황색 "x"를 선택하여 이 규칙을 삭제합니다.



5. 저장을 선택합니다.

단계별 지침에 따라 인스턴스의 상태 제어, 중단된 인스턴스 강제 중지, 향상된 네트워킹을 위해 인스턴스 업데이트, Windows Server 인스턴스의 파일 시스템 확장, 스크립트를 사용하여 시작 시 인스턴스 구성, Windows Server 인스턴스를 보호하는 방법을 알아봅니다.

이 가이드에서는 Linux나 Unix 및 Windows Server 인스턴스를 모두 다루며, 소프트웨어 설치, 구성 업데이트, 암호 관리, 보안 패치 활성화, 방화벽 설정 구성과 같은 작업에 대한 팁과 모범 사례를 제공합니다. 이 가이드를 따라하면 Lightsail 인스턴스를 효과적으로 관리하고 보호하여 특정 사용 사례에 대해 최적의 성능, 보안 및 사용자 지정을 보장할 수 있습니다.

Lightsail 인스턴스 삭제

인스턴스가 더 이상 필요하지 않은 경우 Amazon Lightsail 콘솔 또는 AWS Command Line Interface ()를 사용하여 삭제할 수 있습니다AWS CLI. 삭제하는 즉시 인스턴스에 대한 요금 발생이 중지됩니다. 하지만 삭제된 인스턴스에 연결된 리소스에는 마찬가지로 삭제할 때까지 계속 요금이 부과됩니다. 이러한 리소스와 인스턴스를 삭제한 후 해당 리소스를 삭제하는 방법에 대한 자세한 내용은 다음 단계 섹션을 참조하세요.

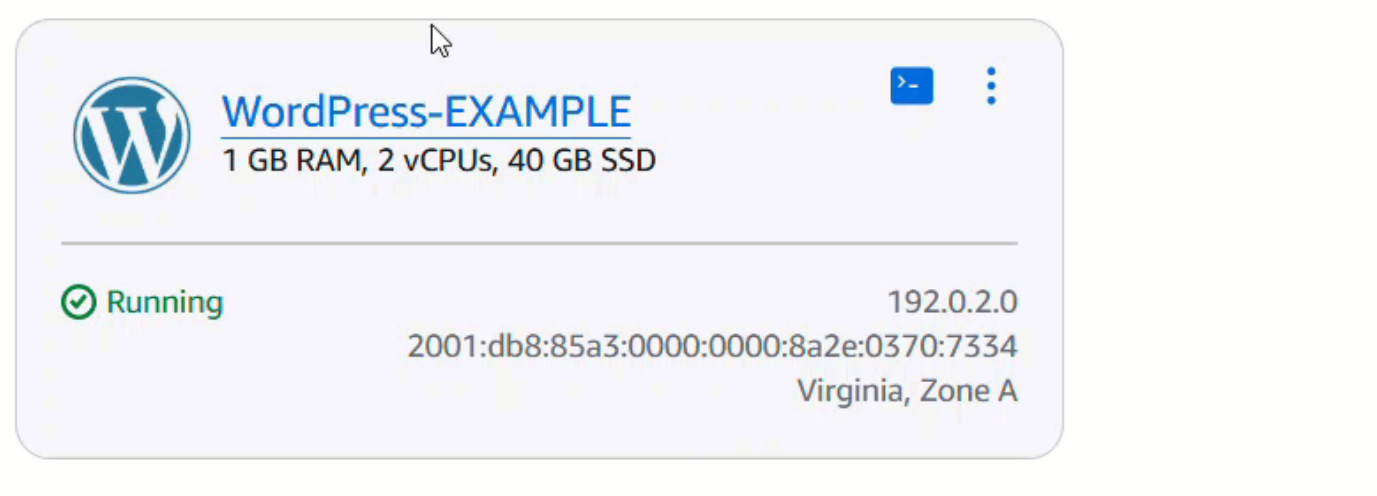
Warning

인스턴스를 삭제하면 복구할 수 없습니다. 인스턴스의 자동 스냅샷도 이 작업의 일부로 삭제됩니다. 나중에 사용할 수 있도록 데이터를 보존하려는 경우, 먼저 인스턴스의 스냅샷을 생성하거나 기존 자동 스냅샷을 유지하도록 선택해야 합니다. 자세한 내용은 다음 설명서를 참조하세요.

- Lightsail에서 자동 스냅샷이 교체되는 것을 방지
- 스냅샷으로 Linux/Unix Lightsail 인스턴스 백업
- Lightsail Windows Server 인스턴스의 스냅샷 생성

Lightsail 콘솔 홈 페이지에서 인스턴스 삭제

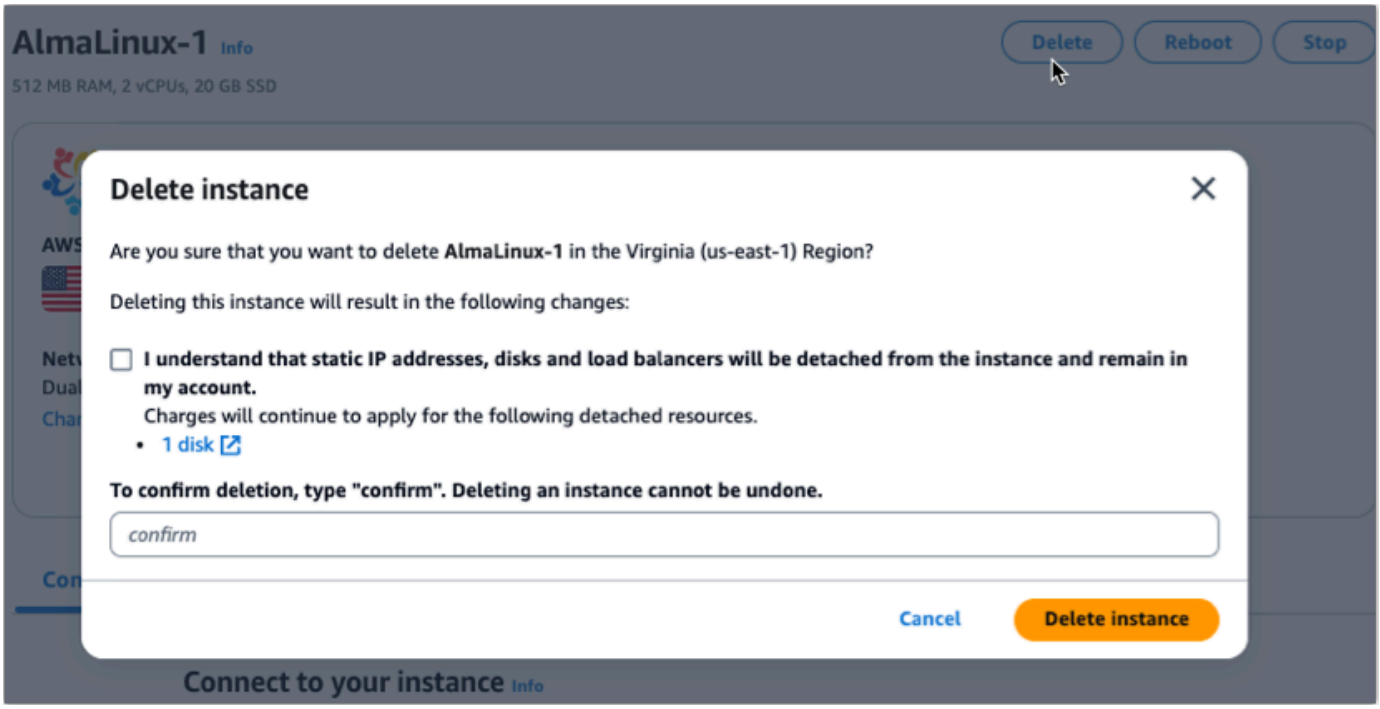
1. Lightsail 콘솔에 로그인합니다.
2. 삭제할 인스턴스의 경우, 작업 메뉴 아이콘(⋮)을 선택한 다음 삭제를 선택합니다.



3. 예, 삭제를 선택하여 삭제를 확인합니다.

Lightsail 콘솔 인스턴스 관리 페이지에서 인스턴스 삭제

1. 홈 페이지의 Lightsail 콘솔에서 삭제하려는 인스턴스를 선택합니다.
2. 삭제 버튼을 선택한 다음 인스턴스 삭제를 선택합니다.



3. 확인란을 선택한 다음 입력 필드에 확인을 입력하여 인스턴스를 삭제할지 확인합니다.
4. [인스턴스 삭제]를 선택하여 삭제를 확인합니다.

를 사용하여 인스턴스 삭제 AWS CLI

1. 아직 수행하지 않은 경우 다음 사전 조건을 완료합니다.

   a. 를 설치합니다 AWS CLI. 자세한 내용은 AWS CLI설치를 참조하세요.

   b. AWS CLI구성. 자세한 내용은 AWS CLI구성 섹션을 참조하세요.

   c. (선택 사항)를 사용합니다 AWS CloudShell. 자세한 내용은 ??? 단원을 참조하십시오.

2. 터미널 또는 명령 프롬프트 창 또는 CloudShell 창을 열고 다음 명령을 입력하여 삭제할 인스턴스의 이름을 가져옵니다.

   ```
   aws lightsail get-instances
   ```

   다음과 유사한 결과가 출력됩니다.

3. 다음 단계에서 사용할 수 있도록 삭제하려는 인스턴스의 이름을 선택하고 복사합니다.

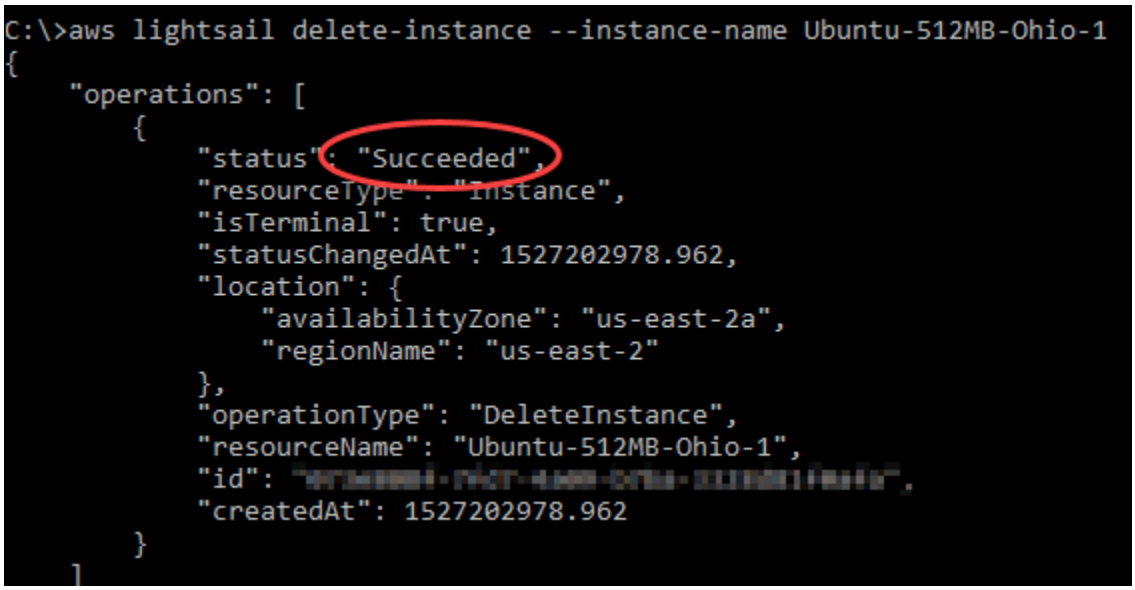
> **Note**
>
> 삭제하려는 인스턴스가 표시되지 않으면 인스턴스 AWS CLI 가 AWS 리전 있는에 대해가 구성되어 있는지 확인합니다. 자세한 내용은 AWS CLI구성 섹션을 참조하세요.

4. 다음 명령을 입력하여 인스턴스를 삭제합니다.

```
aws lightsail delete-instance --instance-name InstanceName
```

명령에서 *InstanceName*을 인스턴스의 이름으로 바꿉니다.

삭제에 성공하면 다음과 유사한 확인 메시지가 표시됩니다.



> **Note**
>
> 삭제에 실패하면 오류 메시지가 나타납니다. 인스턴스의 정확한 이름을 복사하여 붙여넣었는지 확인하고 다시 시도하세요.

다음 단계

인스턴스를 삭제해도 고정 IP, 스냅샷, 블록 스토리지 디스크 및 인스턴스에 연결된 로드 밸런서는 Lightsail에 남아 있으며 추가 요금이 발생합니다. 이러한 리소스를 삭제하는 방법에 대한 자세한 내용은 다음 문서를 참조하세요.

- 고정 IP 삭제
- 스냅샷 삭제
- 블록 스토리지 디스크 분리 및 삭제
- 로드 밸런서 삭제

SSH 키 페어 관리 및 Lightsail 인스턴스 연결

키 페어는 Amazon Lightsail 인스턴스에 연결할 때 자격 증명을 입증하는 데 사용하는 보안 자격 증명 집합입니다. 키 페어는 퍼블릭 키와 프라이빗 키로 구성됩니다. Lightsail은 퍼블릭 키를 인스턴스에 저장하고, 사용자는 프라이빗 키를 저장합니다.

키 페어 파일에는 다음 텍스트가 포함됩니다.

Public key example:

```
ssh-rsa
EXAMPLEzaC1yc2EAAAADAQABAAABAQCoYFOS10yNQ2AoRuvt2uM2LpuZXLGpNoHFxCAmXZjNIZ6t6s
sHCAWgiqzbp5fzRSZnPXjeuxQo2KsGkZCD6f81YHfEIBTSPWoiA6HPWAlAOR6K7E4ZGBkpYhOJKDK1
BYzCKUTgyRUvemmNmGme/c5O4ts50se0A/8m26YNt8TYgKqLV7mjl+Q1uMix0qS3wOim4x
+Iq5eV3cdTa0v0iuQJd01aXoCdJ1cdMW6qEDxZ5ILEMtle8FoLvvMe67JLqjCTxy8i/6x
+SiBWVITOgBKfeePPHsq2PceOQN/XfajeLd+CMAXYyRrvUo4HIiR443BJG1zevIvKYA7+yEXAMPLE
```

Private key example:

```
-----BEGIN RSA PRIVATE KEY-----
EXAMPLEBAAKCAQEAqGBTktdMjUNgKEbr7drjNi6bmVyxqTaBxcQgJl2YzSGererL
BwgFoIqs26eX80UmZz143rsUKNirBpGQg+n/JWB3xCAU0j1qIgOhz1gJQDkeiuxO
GRgZKWITiSgypQWMwilE4MkVL3ppjZhpnv3OTuLbOdLHtAP/JtumDbfE2ICqi1e5
o5fkNbjIsdKkt8DopuMfiKuXld3HU2tL9IrkCXdNW16AnSdXHTFuqhA8WeSCxDLZ
XvBaC77zHuuyS6owk8cvIv+sfkogVlSEzoASn3njzx7Ktj3HjkDf132o3i3fgjAF
2Mka71KOByIkeONwSRtc3ryLymAO/smHHNQRzwIDAQABAoIBAGoipiu2uVOGd/OL
mSaKxpSd1olaq8atTCo8kcN9V1df7OVWTnp1LQ7gu0uOnjLDkQyc7DcCGBgTU+NF
GKJ+es21vGkNi/JmsiMUxQetR8+K8dzCTgx1a07xurzHcP0ivXKajwde2ZLfB/Aw
dcu50zVYvLX7TtUDe++jn02gXF3X3q98lqWmSPV+dtlZPctQqcmemjQg3onUdpZo
4yrAKUKJdrchIMHhBD0jisom86Zl3jEPXRY7iuOfa1bB76cmErja18rijUhMH5Pn
mjAsbvZ0CTxU7QGx5yHnFtSK73oLN4LoYKek0TA7JARc41p0MELtkOTn9mj2IeEw
h2yygPECgYEA37mi3uGVBBAVLEU3Z2sAS/thF0+L2y6qcuBxjY/HeyPnwvuXied0
xJhb9wPpODRTShDkKLfHPiVYD7H6bXZLetZfNLjIu/IKvseL85zCX8fWz6cJ6IeS
3QKRYu2VdpQW2prs+58QyKDlDqQ0hfE3dHZvSayLmm/9/sBZ24+G/WcCgYEAwKqb
yYkDOZtXIHZyTt1UUHvKFzo9LFuuMwlHQdNpvy2QbNNw4iE706DzVjy9FNuMXzIs
Skhhn7m+wredBP+r8udX3+gA1vY329wJ/+c7W8IPN21RiWIT4VtawmoHgMeJHOv4
4mdxqMo6L44Nkny/4KLtGAuZCUrJzoLr+d+Fn1kCgYEAyA7MIdo+0r8+77OFc6kv
PsKvc5TiT0FPkiI56IilrOvS1307aUncFODZe+23Y1cHE7g/DloohN4H/SD9+1xI
6rM/t3l1pvstuKPf9hw7hELDSDTqm1CAd7mQIJKrkLmkJh9bwzXeYEngClOz1AJ7
wF0X7x2oSJXU3zVKJRgXcgkCgYEAn504DxC5YUI2Piiirn9iWIMVe4S+JT+W46Uu
KXSSSNXgrqfE/zH1NHBE6A7NvrfcZQlV8/xfFEp3pS0kon2F4GiUPmUgPPYidLyo
dB8G6A+vN4YTZLOiMLLUT/gzWxbzmshLmpWEbgeLiNYwnElJVTrlHWSOVkplQfbo
tEvfkZECgYAayAwDXa2gbZBmqInwCTNJyqu8XW/Kc4JBT6mugXzQqxMr6ZnXM70h
Fq0EAT7kAHt4wKfZyPkcgrrmj0Mej6VoL2GlJejPykNa20nxrPIi8ecJDYhjiaIp
zoO5rFDVcZhMctewa70OL3c1q+nDGf7Sd9pqw0q31K6MiJwEXAMPLE==
-----END RSA PRIVATE KEY-----
```

Linux 및 Unix 인스턴스에서는 프라이빗 키를 사용하여 인스턴스에 대한 보안 SSH 연결을 설정할 수 있습니다. Windows 인스턴스에서 프라이빗 키는 인스턴스에 대한 보안 RDP 연결을 설정하는 데 사용하는 기본 관리자 암호를 해독합니다.

프라이빗 키에 액세스하는 사람은 누구나 인스턴스에 연결할 수 있으므로 보안된 위치에 프라이빗 키를 저장해 두는 것이 중요합니다.

목차

- 키 페어 옵션 선택
- 인스턴스에 연결
- 인스턴스에 저장된 키 관리

키 페어 옵션 선택

Lightsail을 인스턴스를 생성할 때 다음 키 페어 옵션 중 하나를 선택할 수 있습니다. Windows 인스턴스는 항상 기본 키를 사용하므로 Windows 인스턴스를 만들 때 키 페어를 생성하거나 키를 업로드할 수 없습니다.

- 기본 SSH 키 -는 인스턴스를 생성하는 각에 기본 키 페어 AWS 리전 를 Lightsail 자동으로 생성합니다. 인스턴스에 기본 키 페어를 사용하는 경우 Lightsail은 퍼블릭 키를 인스턴스에 저장합니다. 기본 키 페어의 프라이빗 키는 Lightsail 콘솔의 계정 페이지에서 언제든 다운로드할 수 있습니다. 각각에 최대 하나의 기본 키 페어를 가질 수 있습니다 AWS 리전.
- 키 페어 생성(Linux 및 Unix 인스턴스) - Lightsail 콘솔을 사용하여 인스턴스에 사용할 새 사용자 지정 키 페어를 생성할 수 있습니다. 사용자 지정 키 페어를 생성할 때는 고유한 이름을 제공하고, Lightsail은 퍼블릭 키를 인스턴스에 저장합니다. 사용자 지정 키 페어를 처음 생성할 때만 사용자 지정 키 페어의 프라이빗 키를 다운로드할 수 있습니다.
- 키 업로드(Linux 및 Unix 인스턴스) – 기존 키 페어를 사용하려면 Lightsail에 퍼블릭 키를 업로드할 수 있습니다. 인스턴스에 사용할 퍼블릭 키를 업로드할 때 고유한 이름을 제공하면 Lightsail이 이를 인스턴스에 저장합니다. 키 페어의 프라이빗 키를 보관하고 저장합니다.

여러 인스턴스에 단일 퍼블릭 키를 구성하는 경우 키 페어의 동일한 프라이빗 키를 사용하여 해당 인스턴스에 연결할 수 있습니다. 키 페어 관리에 대한 자세한 내용은 Amazon Lightsail의 키 페어 관리를 참조하세요.

인스턴스에 연결

다음 옵션 중 하나를 사용하여 Lightsail 인스턴스에 연결할 수 있습니다.

Lightsail 브라우저 기반 SSH 및 RDP 클라이언트

Lightsail 콘솔에서 브라우저 기반 SSH 클라이언트를 사용하여 Linux 및 Unix 인스턴스에 즉시 연결하고, 브라우저 기반 RDP 클라이언트를 사용하여 Windows 인스턴스에 연결할 수 있습니다. 브라우저 기반 클라이언트를 사용하여 인스턴스에 연결할 때 컴퓨터에 SSH 클라이언트를 설치하거나 키 페어를 구성하거나 관리자 암호를 지정할 필요가 없습니다. 이는 인스턴스에 연결하는 가장 빠른 방법입니다. 자세한 내용은 Amazon Lightsail에서 Linux 또는 Unix 인스턴스에 연결과 Amazon Lightsail에서 Windows 인스턴스에 연결을 참조하세요.

브라우저 기반 클라이언트는 인스턴스를 생성할 때 구성하는 키 페어와 다른 키 페어를 사용합니다(예: 기본 키 또는 생성하거나 업로드하는 키). 따라서 원래 구성한 키 중 하나를 삭제하거나 분실하더라도 브라우저 기반 클라이언트를 사용하여 인스턴스에 계속 연결할 수 있습니다.

서드 파티 SSH 및 RDP 클라이언트

서드 파티 SSH 클라이언트를 사용하여 Linux 및 Unix 인스턴스에 연결하고 서드 파티 RDP 클라이언트를 사용하여 Windows 인스턴스에 연결할 수 있습니다. SSH 클라이언트를 사용하는 경우 인스턴스에서 구성한 키 페어의 프라이빗 키를 사용하도록 클라이언트를 구성해야 합니다. RDP 클라이언트를 사용하는 경우 Windows 인스턴스의 관리자 암호를 지정해야 합니다.

Windows 컴퓨터를 로컬로 사용하는 경우 다음 클라이언트를 사용하여 Lightsail 인스턴스에 연결할 수 있습니다.

- PuTTY - PuTTY를 사용하여 SSH를 사용하는 Linux 또는 Unix 인스턴스에 연결합니다. 자세한 내용은 PuTTY를 설정하여 인스턴스에 연결을 참조하세요.
- 원격 데스크톱 연결 - 원격 데스크톱 연결을 사용하여 RDP를 사용하는 Windows 인스턴스에 연결합니다. 자세한 내용은 Windows 컴퓨터에서 원격 데스크톱 연결 클라이언트를 사용하여 Windows 인스턴스에 연결을 참조하세요.

Mac 컴퓨터를 로컬로 사용하는 경우 다음 클라이언트를 사용하여 Lightsail 인스턴스에 연결합니다.

- 터미널의 기본 SSH 클라이언트 - 터미널에서 기본 SSH 클라이언트를 사용하여 Linux 및 Unix 인스턴스에 연결합니다. 자세한 내용은 터미널에서 SSH를 사용하여 Linux 또는 Unix 인스턴스에 연결을 참조하세요.
- Microsoft 원격 데스크톱 - macOS용 Microsoft 원격 데스크톱을 통해 RDP를 사용하는 Windows 인스턴스에 연결합니다. 자세한 내용은 Mac에서 Microsoft 원격 데스크톱을 사용하여 Windows 인스턴스에 연결을 참조하세요.

인스턴스에 저장된 키 관리

인스턴스가 시작 및 실행되면 인스턴스에 새 키를 추가하거나 원래 할당한 키를 교체할 수 있습니다. 예를 들어, 조직 내 사용자가 별도의 키를 사용하여 인스턴스에 액세스해야 할 경우 인스턴스에 키를 추가할 수 있습니다. 다른 예로는 누군가가 조직을 떠나고 프라이빗 키(.PEM) 파일의 복사본이 있는 경우일 수 있습니다. 키를 새 키로 바꾸거나 완전히 제거하여 인스턴스에 연결하지 못하게 할 수 있습니다. 자세한 내용은 Amazon Lightsail의 인스턴스에 저장된 키 관리를 참조하세요.

주제

- Lightsail에 대한 SSH 키 설정
- Lightsail SSH 키를 사용한 보안 인스턴스 연결 제어
- Lightsail Linux 인스턴스에서 SSH 키 관리

- Lightsail에 Linux 또는 Unix 인스턴스에 연결
- RDP를 사용하여 Lightsail Windows 인스턴스에 연결

Lightsail에 대한 SSH 키 설정

Secure SHell(SSH)은 가상 프라이빗 서버(또는 Lightsail인스턴스)에 안전하게 연결하는 프로토콜입니다. SSH는 원격 서버를 권한이 있는 사용자에게 연결시키는 프라이빗 키 및 퍼블릭 키를 생성하여 작동합니다. 이 키 페어를 사용하여 브라우저 기반 SSH 터미널에서 Lightsail 인스턴스에 연결할 수 있습니다.

SSH에 대한 자세한 내용은 SSH 이해를 참조하세요.

Lightsail 인스턴스를 생성할 때 Lightsail에서 SSH 키를 관리하도록 하는 것이 기본 옵션입니다. Lightsail에서는 Linux 기반 인스턴스에 안전하게 연결하기 위해 브라우저 기반 SSH 클라이언트를 제공합니다. 이 터미널에서 명령을 입력하고 인스턴스를 변경할 수 있습니다.

Windows 기반 인스턴스는 SSH 대신 원격 데스크톱(RDP) 프로토콜을 사용합니다. Lightsail의 Windows 기반 인스턴스에 대한 자세한 내용은 Lightsail에서 Windows 기반 인스턴스 시작하기를 참조하세요.

> **Important**
>
> SSH 키 관리는 리전을 기준으로 수행됩니다. 새 AWS 리전에서 인스턴스를 생성하면 이 리전에 기본 키 페어를 사용하는 옵션이 주어집니다. 그 리전에서 사용자 지정 키를 사용할 수도 있습니다. 자체 키를 업로드하는 경우 Lightsail 인스턴스가 있는 각 리전에 대해 업로드해야 합니다.

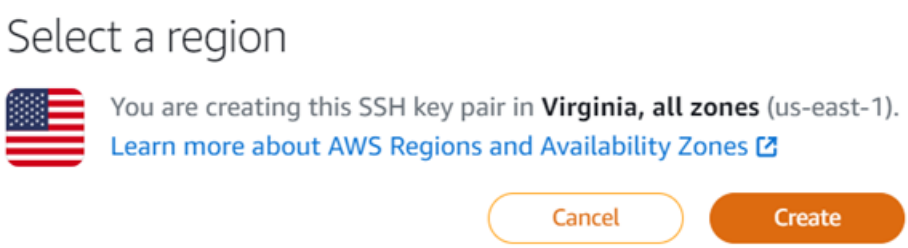
기본 키를 사용하는 경우에도 프라이빗 키를 다운로드하여 보관할 수 있습니다. 인스턴스를 생성하는 시점이나 그 이후에 위 작업을 수행할 수 있습니다. 인스턴스를 만든 후 키를 다운로드하도록 선택한 경우 계정(Account) 페이지의 SSH 키(SSH keys)에서 다운로드할 수 있습니다.

새 키 생성

기본 키를 사용하도록 선택하지 않으면 Lightsail 인스턴스를 생성할 때 새 키 페어를 생성할 수 있습니다.

1. 아직 선택하지 않았다면 인스턴스 생성을 선택합니다.
2. 인스턴스 생성 페이지에서 사용자 지정 키 생성을 선택합니다.

3. Lightsail에 새 키가 생성되는 리전이 표시됩니다.



   생성을 선택합니다.

4. 키 페어 이름을 입력합니다.

   리소스 이름:

   - Lightsail 계정의 각 AWS 리전 내에서 고유해야 합니다.
   - 2~255자의 문자로 구성되어야 합니다.
   - 영숫자 문자 또는 숫자로 시작하고 끝나야 합니다.
   - 영숫자 문자, 숫자, 마침표, 대시, 밑줄이 포함될 수 있습니다.

5. 키 페어 생성을 선택합니다.

   > **Important**
   >
   > 쉽게 찾을 수 있는 위치에 키를 저장하세요. 권한을 다른 사람이 읽을 수 없도록 설정하는 것도 좋은 방법입니다.

6. 인스턴스 생성 절차를 계속 진행합니다.

기존 키 업로드

Lightsail 인스턴스를 생성할 때 기존 키를 업로드하도록 선택할 수도 있습니다.

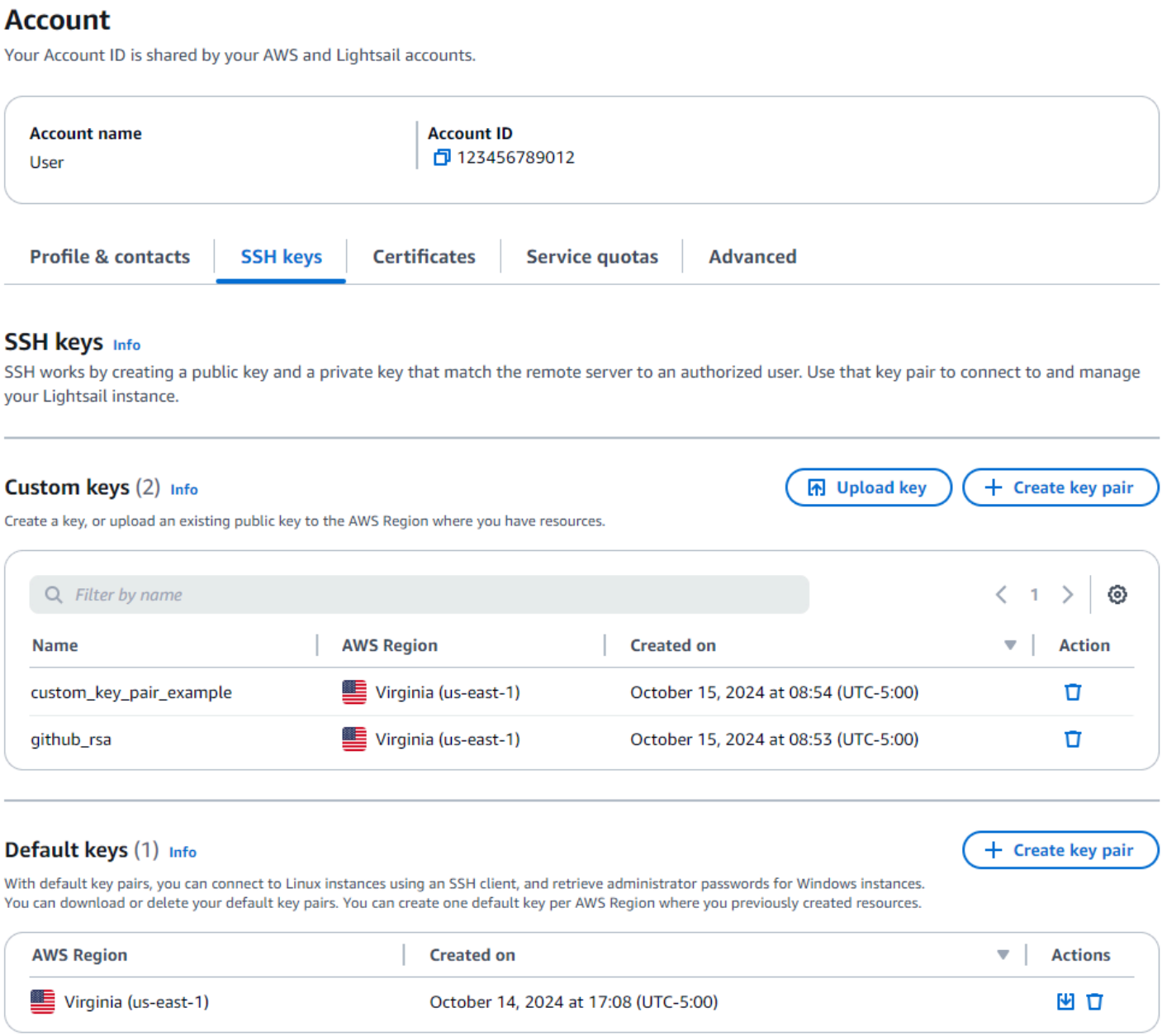
1. 아직 선택하지 않았다면 인스턴스 생성을 선택합니다.
2. 인스턴스 생성 페이지에서 키 업로드를 선택합니다.
3. 업로드를 선택합니다.
4. 새 키를 업로드하는 리전이 Lightsail에 표시됩니다.
5. 파일 선택을 선택하여 로컬 시스템에서 키를 찾습니다.

(프라이빗 키가 아니라) 반드시 퍼블릭 키를 업로드하세요. 예를 들어 `github_rsa.pub`입니다.

6. 키 업로드를 선택합니다.
7. 인스턴스 생성 절차를 계속 진행합니다.

키 관리

계정(Account) 페이지의 SSH 키(SSH keys) 탭에서 키를 관리할 수 있습니다. 각 리전에서 사용하는 각각의 키 페어가 보일 것입니다.



이 페이지에서 새 키를 생성하거나, 기존 키를 삭제하거나, 기존 키를 업로드하거나, 프라이빗 키를 다운로드할 수 있습니다. PuTTY 같은 SSH 클라이언트를 사용하여 연결할 수도 있는데, 이때는 키의 프라이빗 부분 중 절반이 필요합니다. 계정 페이지에서 키를 다운로드할 수 있습니다. Lightsail 인스턴스에 연결하도록 PuTTY를 설정하는 방법을 자세히 알아봅니다..

Lightsail SSH 키를 사용한 보안 인스턴스 연결 제어

키 페어를 사용하여 Amazon Lightsail 인스턴스에 보안 연결을 설정할 수 있습니다. Amazon Lightsail 인스턴스를 처음 생성할 때 Lightsail에서 대신 생성한 키 페어(Lightsail 기본 키 페어) 또는 직접 생성한 사용자 지정 키 페어를 사용할 수 있습니다. 자세한 내용은 Amazon Lightsail의 키 페어 및 인스턴스 연결을 참조하세요.

Linux 및 Unix 인스턴스에서는 프라이빗 키를 사용하여 인스턴스에 대한 보안 SSH 연결을 설정할 수 있습니다. Windows 인스턴스에서 프라이빗 키는 인스턴스에 대한 보안 RDP 연결을 설정하는 데 사용하는 기본 관리자 암호를 해독합니다.

이 가이드에서는 Lightsail 인스턴스에서 사용할 수 있는 키를 관리하는 방법을 안내합니다. 키를 보고, 기존 키를 삭제하고, 새 키를 생성하거나 업로드할 수 있습니다.

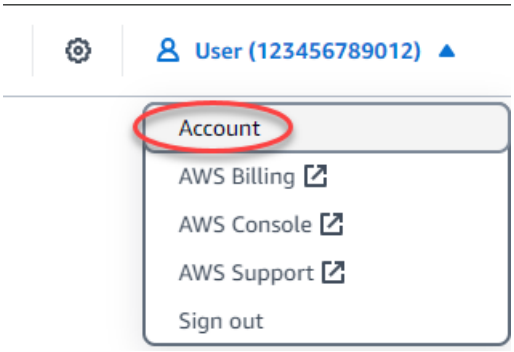
목차

- 기본 키 및 사용자 지정 키 보기
- Lightsail 콘솔에서 기본 키의 프라이빗 키 다운로드
- Lightsail 콘솔의 사용자 지정 키 삭제
- Lightsail 콘솔에서 기본 키를 삭제하고 새 키 생성
- Lightsail 콘솔을 사용하여 사용자 지정 키 생성
- ssh-keygen을 사용하여 사용자 지정 키를 생성하고 Lightsail에 업로드

기본 키 및 사용자 지정 키 보기

다음 절차를 완료하여 Lightsail 콘솔에서 기본 키와 사용자 지정 키를 봅니다.

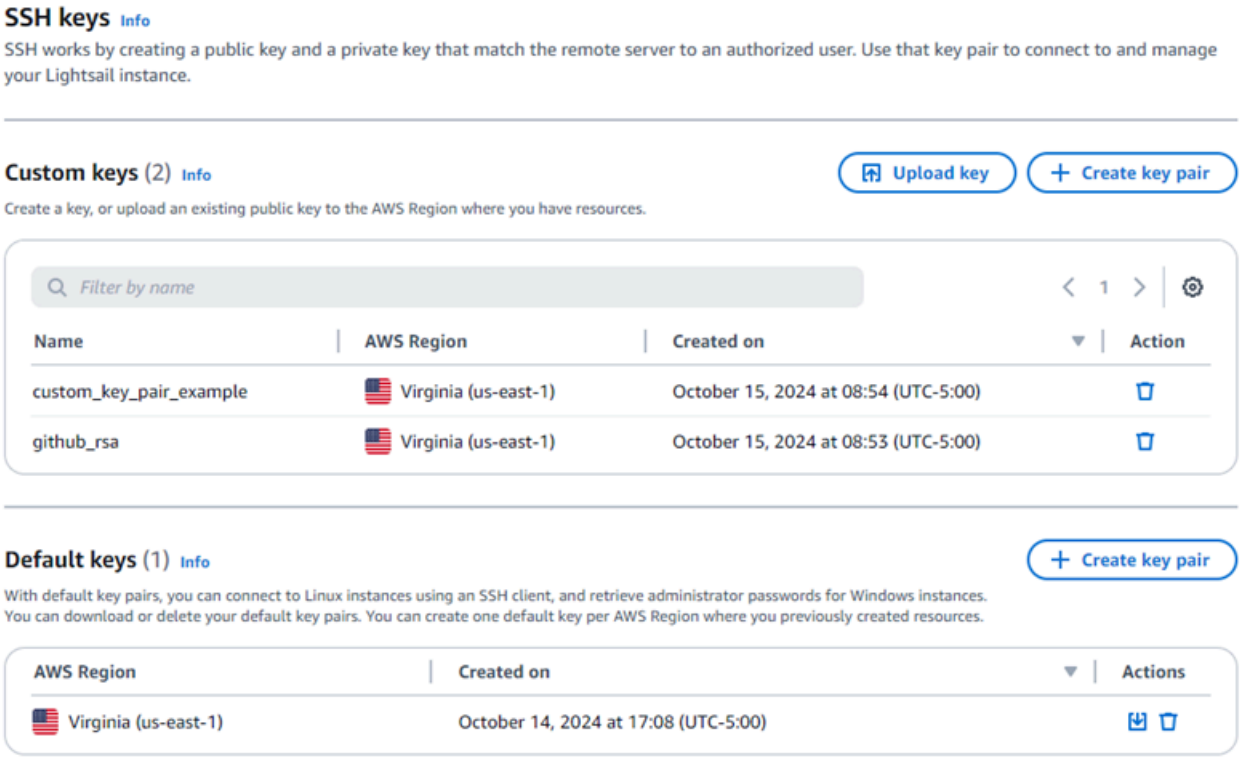
1. Lightsail 콘솔에 로그인합니다.
2. Lightsail 홈 페이지의 상단 탐색 메뉴에서 계정을 선택합니다.
3. 드롭다운 메뉴에서 계정(Account)을 선택합니다.



4. SSH 키(SSH keys) 탭을 선택합니다.

   SSH 키 페이지에 다음이 나열됩니다.

   - 사용자 지정 키 - Lightsail 콘솔 또는 ssh-keygen과 같은 서드 파티 도구를 사용하여 생성한 키입니다. 각각에 많은 사용자 지정 키가 있을 수 있습니다 AWS 리전.
   - 기본 키 - Lightsail에서 생성한 키입니다. 각 AWS 리전에 1개의 기본 키만 있을 수 있습니다.



사용자 지정 키와 기본 키는 리전별로 다릅니다. 예를 들어, 미국 서부(오레곤) AWS 리전 의 키는 해당 리전에서 생성된 인스턴스에서만 구성할 수 있습니다. 키에 대한 자세한 내용은 Amazon Lightsail의 키 페어 및 인스턴스 연결을 참조하세요.

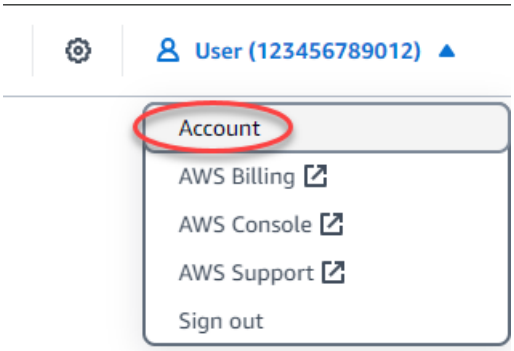
SSH 키 페이지에서 키 페어를 생성하고, 키를 업로드하고, 삭제하고, Lightsail 기본 키 페어의 프라이빗 키를 다운로드할 수 있습니다.

> **Note**
>
> Lightsail은 해당 키를 저장하지 않으므로 사용자 지정 키 페어의 프라이빗 키를 다운로드할 수 없습니다. 사용자 지정 키 페어의 프라이빗 키를 분실한 경우 새로 생성하고, 인스턴스에서 이를 구성해야 합니다. 그런 다음 손실된 키를 삭제합니다. 자세한 내용은 Lightsail 콘솔을 사용하여 사용자 지정 키 생성 또는 ssh-keygen을 사용하여 사용자 지정 키를 생성하고 Lightsail에 업로드를 참조하세요.

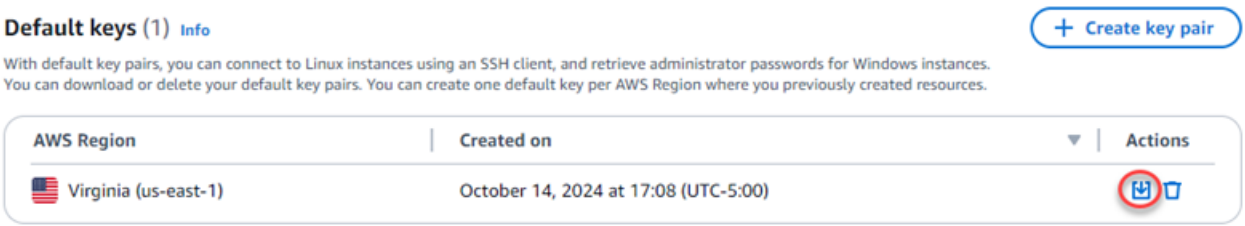
Lightsail 콘솔에서 기본 키의 프라이빗 키 다운로드

다음 절차를 완료하여 Lightsail 콘솔에서 기본 키 페어의 프라이빗 키를 다운로드합니다.

1. Lightsail 콘솔에 로그인합니다.
2. Lightsail 홈 페이지의 상단 탐색 메뉴에서 계정을 선택합니다.
3. 드롭다운 메뉴에서 계정(Account)을 선택합니다.



4. SSH 키(SSH keys) 탭을 선택합니다.
5. 페이지의 기본 키 섹션에서 다운로드할 키의 다운로드 아이콘을 선택합니다.



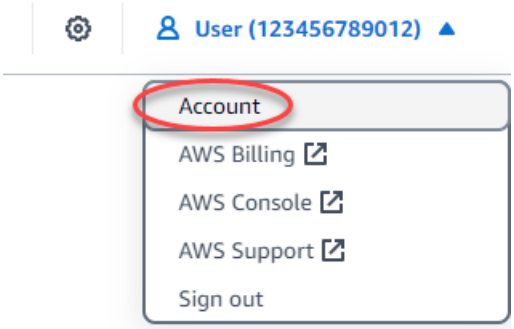
> **Important**
>
> 안전한 위치에 프라이빗 키를 저장합니다. 인스턴스에 연결하는 데 사용할 수 있으므로 공개적으로 공유하지 마십시오.

프라이빗 키를 사용하여 인스턴스에 연결하도록 SSH 클라이언트를 구성할 수 있습니다. 자세한 내용은 인스턴스에 연결을 참조하세요.

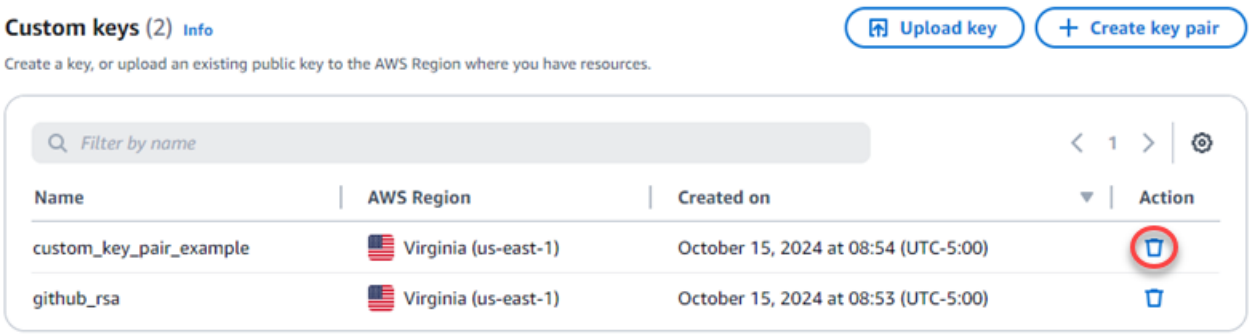
Lightsail 콘솔의 사용자 지정 키 삭제

다음 절차에 따라 Lightsail 콘솔에서 사용자 지정 키를 삭제합니다. 이렇게 하면 Lightsail에서 생성한 새 인스턴스에 사용자 지정 키가 구성되지 않습니다.

1. Lightsail 콘솔에 로그인합니다.
2. Lightsail 홈 페이지의 상단 탐색 메뉴에서 계정을 선택합니다.
3. 드롭다운 메뉴에서 계정(Account)을 선택합니다.



4. SSH 키(SSH keys) 탭을 선택합니다.
5. 페이지의 사용자 지정 키 섹션에서 삭제할 키의 삭제 아이콘을 선택합니다.

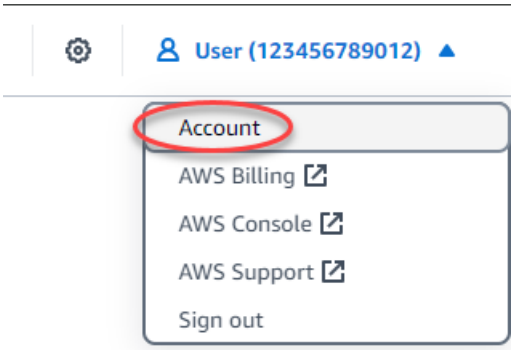


이는 이전에 생성되어 현재 실행 중인 인스턴스에서 사용자 지정 키 페어의 퍼블릭 키를 제거하지는 않습니다. 실행 중인 인스턴스에 저장된 이전에 구성한 퍼블릭 키를 제거하려면 Amazon Lightsail의 인스턴스에 저장된 키 관리를 참조하세요.

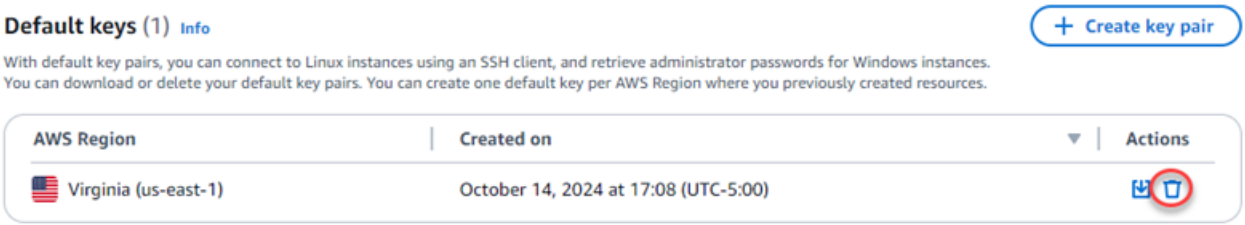
Lightsail 콘솔에서 기본 키를 삭제하고 새 키 생성

다음 절차에 따라 Lightsail 콘솔에서 기본 키를 삭제합니다. 이렇게 하면 Lightsail에서 생성한 새 인스턴스에 기본 키가 구성되지 않습니다. 그런 다음 새 기본 키를 생성하여 삭제한 키를 대체할 수 있습니다. Lightsail에서 생성한 새 인스턴스에 새 기본 키를 구성할 수 있습니다.

1. Lightsail 콘솔에 로그인합니다.
2. Lightsail 홈 페이지의 상단 탐색 메뉴에서 계정을 선택합니다.
3. 드롭다운 메뉴에서 계정(Account)을 선택합니다.



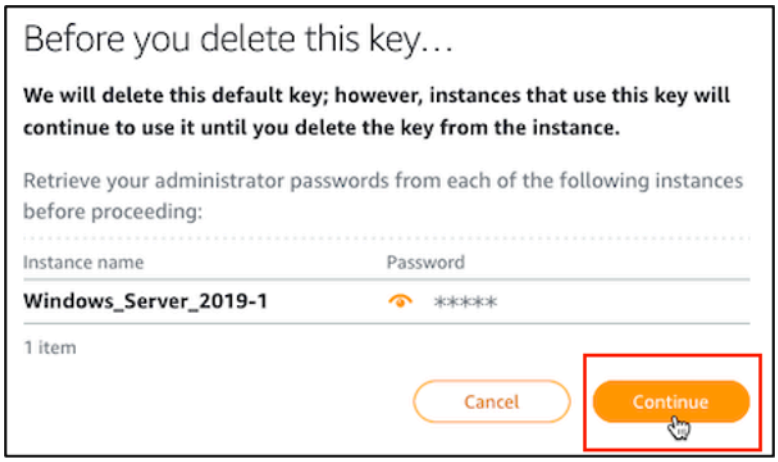
4. SSH 키(SSH keys) 탭을 선택합니다.
5. 페이지의 기본 키 섹션에서 삭제할 기본 키의 삭제 아이콘을 선택합니다.



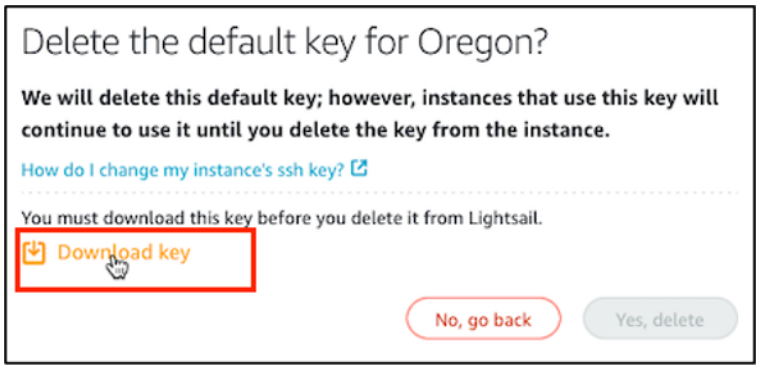
> **Important**
>
> 기본 키를 삭제해도 이전에 생성되었고 현재 실행 중인 인스턴스에서 사용자 지정 키 페어의 퍼블릭 키가 제거되지는 않습니다. 자세한 내용은 Amazon Lightsail의 인스턴스에 저장된 키 관리를 참조하세요.

6. Windows 인스턴스의 경우 관리자 암호를 생성하는 데 기본 키가 사용됩니다. 기본 키를 삭제하기 전에 삭제할 기본 키를 사용하는 모든 Windows 인스턴스에서 관리자 암호를 검색하고 저장해야 합니다.

7. 계속(Continue)을 선택하여 기본 키를 삭제합니다.



8. 삭제하기 전에 기본 키를 다운로드해야 합니다. 기본 키를 다운로드한 후 예, 삭제합니다.(Yes, delete)를 선택하여 기본 키를 영구적으로 삭제할 수 있습니다.



9. 기본 키가 삭제되었습니다. 확인을 선택합니다.

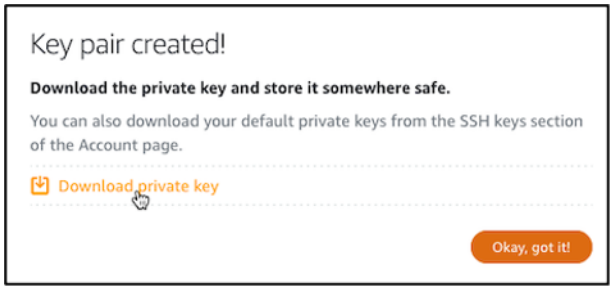
다음 단계는 선택 사항이며 삭제한 기본 키 페어를 교체하려는 경우에만 완료해야 합니다.

10. 페이지의 기본 키 섹션에서 키 페어 생성을 선택합니다.

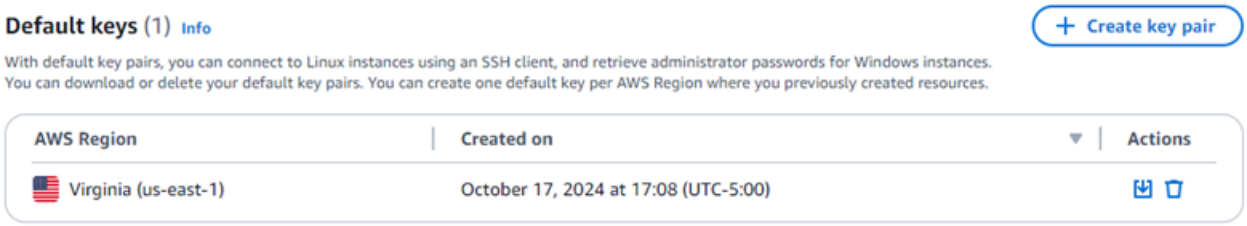
11. 표시되는 리전 선택 프롬프트 AWS 리전 에서 새 기본 키를 생성할를 선택합니다. 동일한 AWS 리전의 새 인스턴스에서 새 기본 키를 구성할 수 있습니다.

> **Note**
>
> 이 단계를 사용하면 Lightsail 리소스를 생성한 AWS 리전에서만 기본 키 페어를 생성할 수 있습니다. 새 리전에서 기본 키 페어를 생성하려면 해당 리전에서 Lightsail 리소스를 생성해야 합니다. 리소스를 생성하면 기본 키 페어도 생성됩니다.

12. 프라이빗 키를 다운로드하고 안전한 위치에 보관합니다.
13. 확인되었습니다!(Ok, got it!)를 선택하고 계속 진행합니다.



14. Lightsail 콘솔 SSH 키 페이지에서 새 기본 키를 확인합니다.

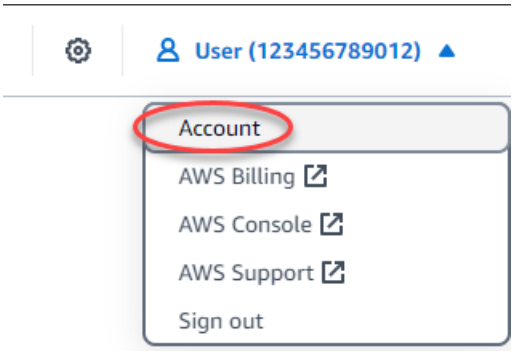


Lightsail에서 생성한 새 인스턴스에 새 기본 키를 구성할 수 있습니다. 이전에 생성되었고 현재 실행 중인 인스턴스에 새 기본 키를 구성하려면 Amazon Lightsail의 인스턴스에 저장된 키 관리를 참조하세요.

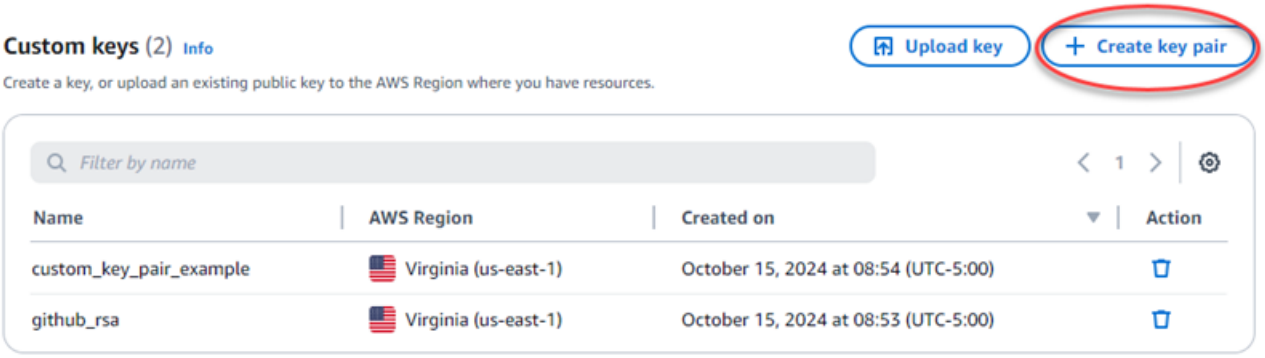
Lightsail 콘솔을 사용하여 사용자 지정 키 생성

Lightsail 콘솔을 사용하여 사용자 지정 키 페어를 생성하려면 다음 절차를 완료하세요. Lightsail에서 생성한 새 인스턴스에 새 사용자 지정 키를 구성할 수 있습니다.

1. Lightsail 콘솔에 로그인합니다.
2. Lightsail 홈 페이지의 상단 탐색 메뉴에서 계정을 선택합니다.
3. 드롭다운 메뉴에서 계정(Account)을 선택합니다.

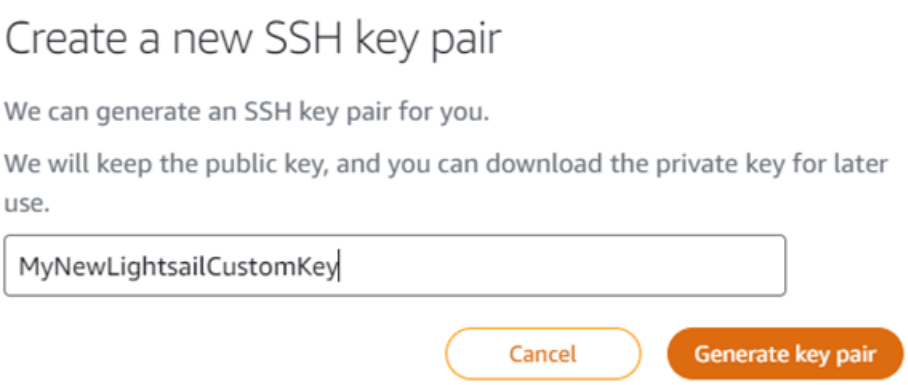


4. SSH 키(SSH keys) 탭을 선택합니다.
5. 페이지의 사용자 지정 키 섹션에서 키 페어 생성을 선택합니다.



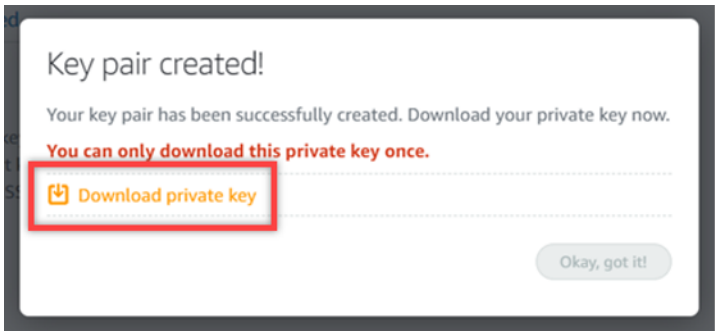
6. 표시되는 리전 선택 프롬프트에서 새 사용자 지정 키를 생성하려는 AWS 리전 을 선택합니다. 동일한 AWS 리전의 새 인스턴스에서 새 사용자 지정 키를 구성할 수 있습니다.

7. 표시되는 새 SSH 키 페어 생성(Create a new SSH key pair) 프롬프트에서 사용자 지정 키 이름을 지정한 다음 키 페어 생성(Generate key pair)을 선택합니다.

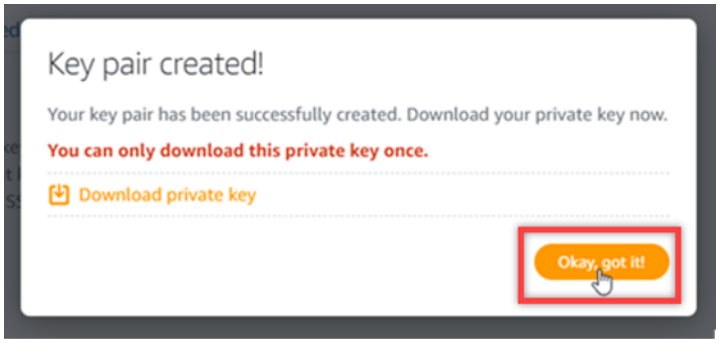


8. 키 페어가 생성되었습니다!(Key pair created!) 프롬프트가 표시되면 프라이빗 키 다운로드(Download private key)를 선택하여 로컬 컴퓨터에 프라이빗 키를 저장합니다.

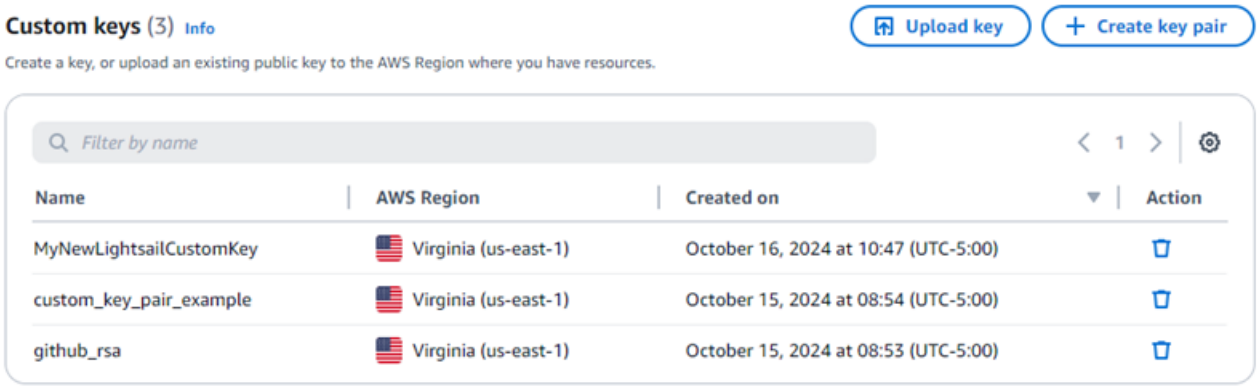
> **Important**
>
> 안전한 위치에 프라이빗 키를 저장합니다. 인스턴스에 연결하는 데 사용할 수 있으므로 공개적으로 공유하지 마십시오.
> 이때에만 사용자 지정 키 페어의 프라이빗 키를 다운로드할 수 있습니다. Lightsail은 사용자 지정 키 페어의 프라이빗 키를 저장하지 않습니다. 이 프롬프트를 닫은 후 다시 다운로드할 수 없습니다.



9. 확인되었습니다!(Ok, got it!)를 선택하여 프롬프트를 닫습니다.



10. 새 사용자 지정 키는 페이지의 사용자 지정 키 섹션에 나열됩니다.



Lightsail에서 생성한 새 인스턴스에 새 사용자 지정 키를 구성할 수 있습니다. 이전에 생성되었고 현재 실행 중인 인스턴스에 새 사용자 지정 키를 구성하려면 Amazon Lightsail의 인스턴스에 저장된 키 관리를 참조하세요.

ssh-keygen을 사용하여 사용자 지정 키를 생성하고 Lightsail에 업로드

다음 절차에 따라 ssh-keygen과 같은 서드 파티 도구를 사용하여 로컬 컴퓨터에 사용자 지정 키 페어를 생성합니다. 키를 생성한 후에는 Lightsail 콘솔에 업로드할 수 있습니다. Lightsail에서 생성한 새 인스턴스에 새 사용자 지정 키를 구성할 수 있습니다.

1. 로컬 컴퓨터에서 명령 프롬프트 또는 터미널을 엽니다.
2. 다음 명령을 입력하여 키 페어를 생성합니다.

```
ssh-keygen -t rsa
```

3. 컴퓨터에서 키 페어를 저장할 디렉터리 위치를 지정합니다.

   예를 들어 다음 디렉터리 중 하나를 지정할 수 있습니다.

a. Windows: `C:\Users\<UserName>\.ssh\<KeyPairName>`

b. macOS, Linux 또는 Unix: `/home/<UserName>/.ssh/<KeyPairName>`

*<UserName>*을(를) 현재 로그인한 사용자의 이름으로 바꾸고, *<KeyPairName>*을(를) 새 키 페어의 이름으로 바꿉니다.

다음 예제에서는 Windows 컴퓨터에서 `C:\Keys` 디렉터리를 지정하고, 새 키에 `MyNewLightsailCustomKey`라는 이름을 지정했습니다.

```
C:\Users\[redacted]>ssh-keygen -t rsa
Generating public/private rsa key pair.
Enter file in which to save the key (C:\Users\[redacted]/.ssh/id_rsa): C:\Keys\MyNewLighstailCustomKey
```

4. 키의 암호를 입력하고 Enter 키를 누릅니다. 암호를 입력할 때 암호를 볼 수 없습니다.

   나중에 SSH 클라이언트에서 키 페어의 프라이빗 키를 구성하여 키 페어의 퍼블릭 키가 구성된 인스턴스에 연결할 때 이 암호가 필요합니다.

   ```
   Enter passphrase (empty for no passphrase):
   ```

5. 암호를 다시 입력하여 확인하고 Enter 키를 누릅니다. 암호를 입력할 때 암호를 볼 수 없습니다.

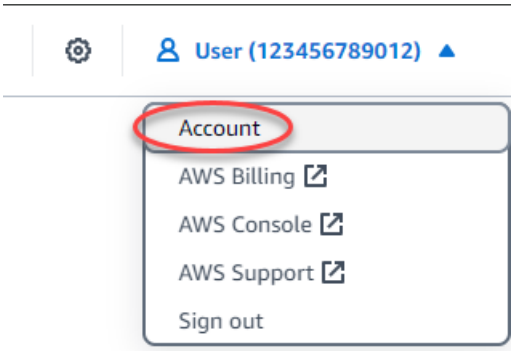
   ```
   Enter same passphrase again:
   ```

6. 프라이빗 키와 퍼블릭 키가 지정된 디렉터리에 저장되었음을 확인하는 메시지가 표시됩니다.

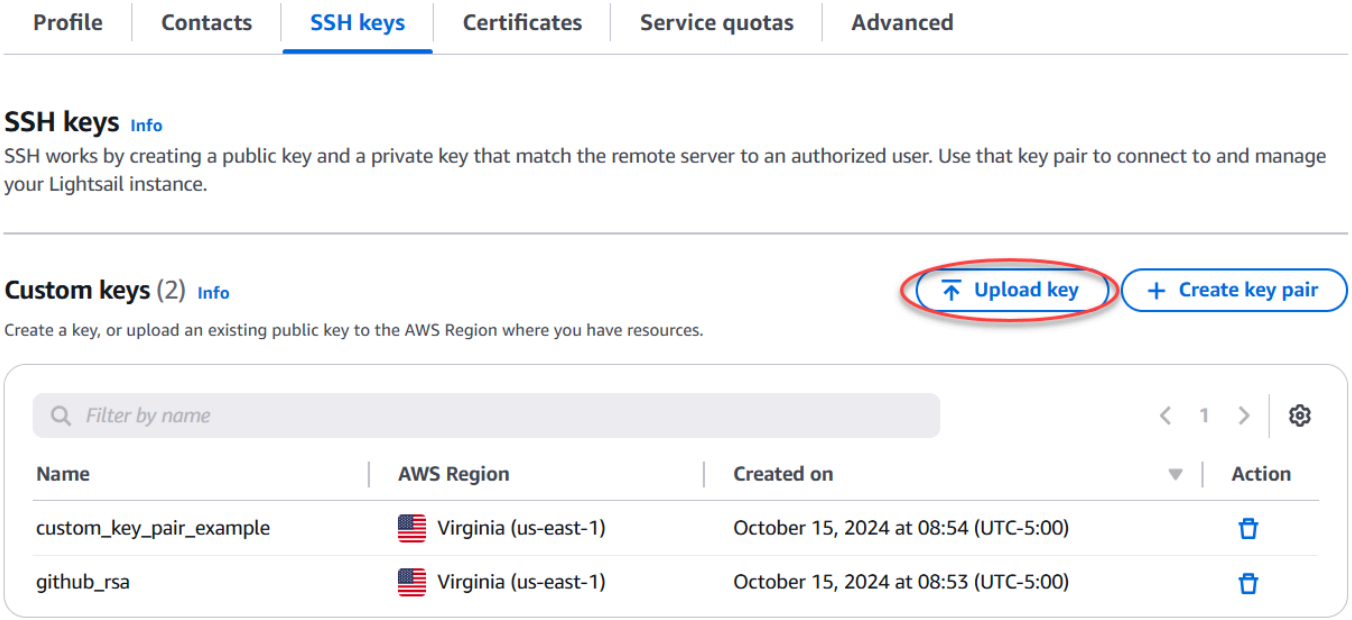
   ```
   Your identification has been saved in C:\Keys\MyNewLighstailCustomKey.
   Your public key has been saved in C:\Keys\MyNewLighstailCustomKey.pub.
   ```

   다음으로 키 페어의 퍼블릭 키를 Lightsail 콘솔에 업로드합니다.

7. Lightsail 콘솔에 로그인합니다.
8. Lightsail 홈 페이지의 상단 탐색 메뉴에서 계정을 선택합니다.
9. 드롭다운 메뉴에서 계정(Account)을 선택합니다.

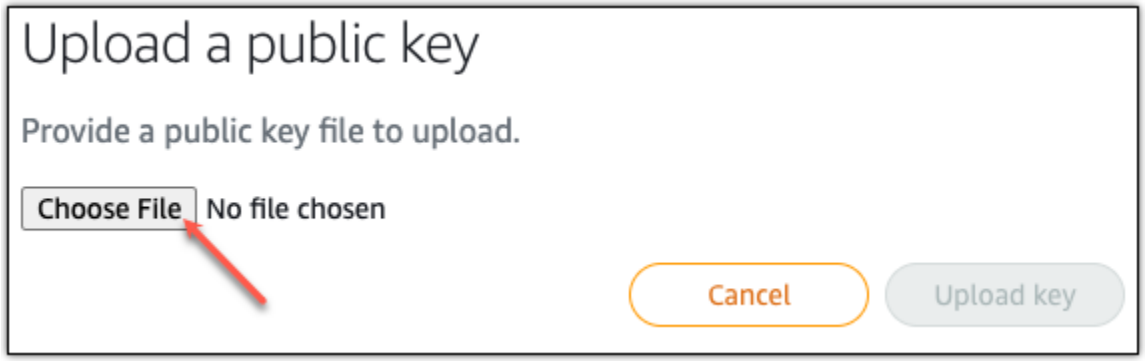


10. SSH 키(SSH keys) 탭을 선택합니다.
11. 페이지의 사용자 지정 키 섹션에서 키 업로드를 선택합니다.

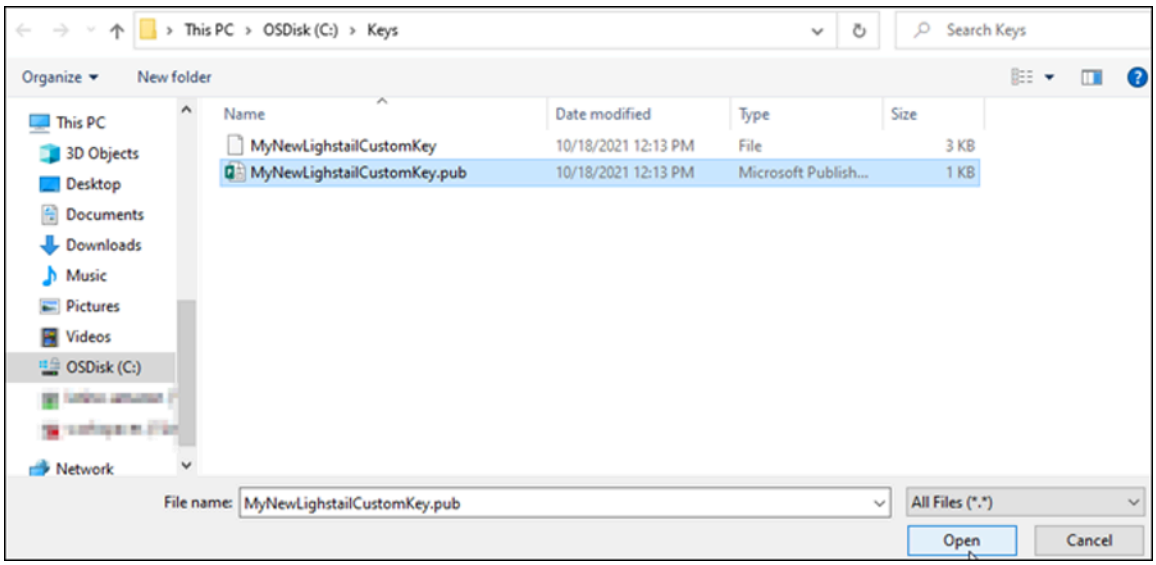


12. 표시되는 리전 선택 프롬프트 AWS 리전 에서 새 사용자 지정 키를 업로드할를 선택합니다. 동일한 AWS 리전의 새 인스턴스에서 새 사용자 지정 키를 구성할 수 있습니다.

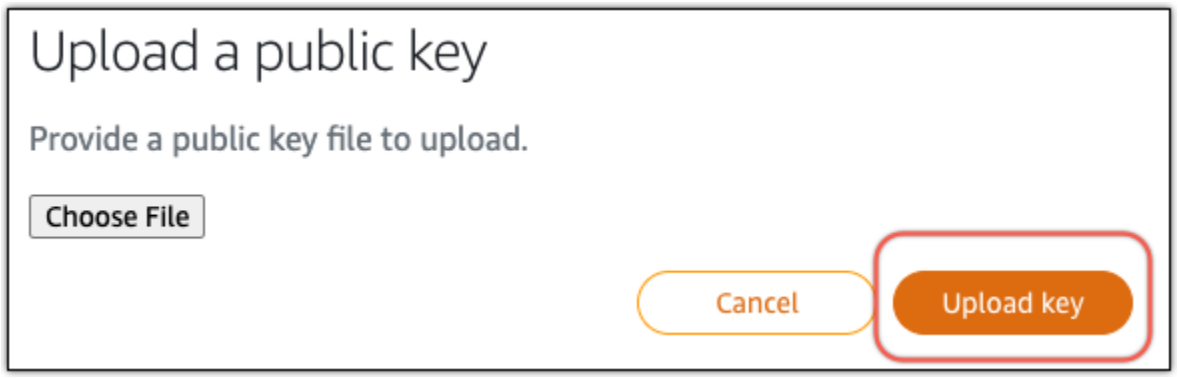
13. 업로드를 선택합니다.
14. 표시되는 퍼블릭 키 업로드(Upload a public key) 프롬프트에서 파일 선택(Choose File)을 클릭합니다.



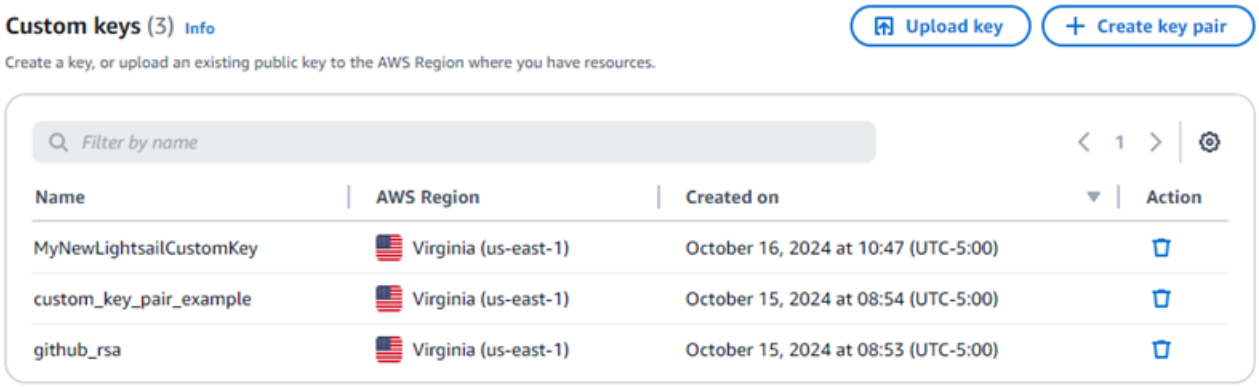
15. 로컬 컴퓨터에서 이 절차의 앞부분에서 생성한 키 페어의 퍼블릭 키를 찾은 다음 열기(Open)를 선택합니다. 키 페어의 퍼블릭 키는.PUB 파일 확장명을 가진 파일입니다.



16. 키 업로드를 선택합니다.



17. 새 사용자 지정 키는 페이지의 사용자 지정 키 섹션에 나열됩니다.



키를 업로드한 AWS 리전에서 생성한 새 인스턴스에 새 사용자 지정 키를 구성할 수 있습니다. 이전에 생성되었고 현재 실행 중인 인스턴스에 새 사용자 지정 키를 구성하려면 Amazon Lightsail의 인스턴스에 저장된 키 관리를 참조하세요.

Lightsail Linux 인스턴스에서 SSH 키 관리

키 페어를 사용하여 Amazon Lightsail에 대한 보안 연결을 설정할 수 있습니다. Lightsail은 처음 생성할 때 Linux 또는 Unix 인스턴스에서 키 페어의 퍼블릭 키를 구성합니다. SSH 연결을 설정할 때 인스턴스를 인증하기 위해 키 페어의 프라이빗 키를 사용합니다. 키에 대한 자세한 내용은 키 페어 및 인스턴스 연결을 참조하세요.

인스턴스가 가동된 이후 인스턴스에서 새 퍼블릭 키를 추가하거나 퍼블릭 키를 대체(기존 퍼블릭 키 삭제 및 새 퍼블릭 키 추가)하여 인스턴스에 연결하는 데 사용되는 키 페어를 변경할 수 있습니다. 이렇게 해야 하는 경우는 다음과 같습니다.

- 조직 내 사용자가 별도의 키 페어를 사용하여 인스턴스에 액세스해야 하는 경우 인스턴스에 퍼블릭 키 페어를 추가할 수 있습니다.
- 손상된 키를 사용하는 인스턴스의 스냅샷에서 생성된 새 인스턴스를 보호해야 하는 경우.
- 프라이빗 키의 복사본을 보유하고 있는 누군가의 인스턴스 연결을 차단하려는 경우(예: 퇴사자) 인스턴스의 퍼블릭 키를 삭제하고 새 키로 교체할 수 있습니다.

인스턴스의 키를 추가하거나 교체하려면 인스턴스에 연결할 수 있어야 합니다. 기존 프라이빗 키를 분실한 경우 Lightsail 브라우저 기반 SSH 클라이언트를 사용하여 인스턴스에 연결할 수 있습니다. 자세한 내용은 Linux 또는 Unix 인스턴스에 연결을 참조하세요.

목차

- 1단계: 프로세스 알아보기
- 2단계: 키 페어 생성
- 3단계: 인스턴스에 퍼블릭 키 추가
- 4단계: 새 키 페어를 사용하여 인스턴스에 연결
- 5단계: 인스턴스에서 기존 퍼블릭 키 삭제

1단계: 프로세스 알아보기

다음은 인스턴스에서 키를 추가 및 제거하는 일반적인 단계입니다. 새 키를 추가하지 않고 인스턴스에서 키를 제거하려면 이 가이드 후반부에 있는 5단계: 인스턴스에서 기존 퍼블릭 키 삭제 섹션을 참조하세요.

1. 키 페어 생성 - 인스턴스에 새 키를 추가하려면 먼저 새 키 페어를 생성해야 합니다. Lightsail 콘솔을 사용하여 사용자 지정 키 페어를 생성할 수 있습니다. 또는 로컬 컴퓨터에서 ssh-keygen과 같은

서드 파티 도구를 사용하여 생성할 수 있습니다. 두 방법 모두 퍼블릭 키와 프라이빗 키로 구성된 새 키 페어를 생성합니다. 자세한 내용은 이 가이드 후반부에 있는 2단계: 키 페어 생성을 참조하세요.

2. 인스턴스에 퍼블릭 키 추가 - 키 페어를 생성한 후 SSH를 사용하여 인스턴스에 연결하고 키 페어의 퍼블릭 키를 인스턴스에 추가합니다. 자세한 내용은 이 가이드의 후반부에 있는 3단계: 인스턴스에 퍼블릭 키 추가를 참조하세요.
3. 새 키 페어를 사용하여 인스턴스에 연결할 수 있는지 테스트 - 키 페어의 퍼블릭 키가 인스턴스에 저장되면 키 페어의 프라이빗 키를 사용하여 SSH를 사용하는 인스턴스에 연결할 수 있는지 테스트해야 합니다. 자세한 내용은 이 가이드의 후반부에 있는 4단계: 새 키 페어를 사용하여 인스턴스에 연결을 참조하세요.
4. 인스턴스에서 이전 퍼블릭 키 제거 - 새 키를 사용하여 인스턴스에 성공적으로 연결한 후에는 인스턴스에서 이전 퍼블릭 키를 제거할 수 있습니다. 사용자가 이전 키 페어를 사용하여 인스턴스에 연결하지 못하도록 하려면 이 단계를 완료하세요. 자세한 내용은 이 가이드의 후반부에 있는 5단계: 인스턴스에서 기존 퍼블릭 키 삭제를 참조하세요.

2단계: 키 페어 생성

ssh-keygen을 사용하여 로컬 컴퓨터에서 키 페어를 생성하려면 다음 절차를 완료하세요.

1. 로컬 컴퓨터에서 명령 프롬프트 또는 터미널을 엽니다.
2. 다음 명령을 입력하여 키 페어를 생성합니다.

```
ssh-keygen -t rsa
```

3. 컴퓨터에서 키 페어를 저장할 디렉터리 위치를 지정합니다.

   예:

   - Windows: `C:\Users\<UserName>\.ssh\<KeyPairName>`
   - macOS, Linux 또는 Unix: `/home/<UserName>/.ssh/<KeyPairName>`

   *<UserName>*을 현재 로그인한 사용자의 이름으로 바꾸고, *<KeyPairName>*을 새 키 페어의 이름으로 바꿉니다.

   다음 예제에서는 Windows 컴퓨터에서 `C:\Keys` 디렉터리를 지정하고, 새 키에 `MyNewLightsailCustomKey`라는 이름을 지정했습니다.

```
C:\Users\[redacted]>ssh-keygen -t rsa
Generating public/private rsa key pair.
Enter file in which to save the key (C:\Users\[redacted]/.ssh/id_rsa): C:\Keys\MyNewLighstailCustomKey
```

4. 키의 암호를 입력하고 Enter 키를 누릅니다. 암호를 입력할 때 암호를 볼 수 없습니다.

   나중에 SSH 클라이언트에서 프라이빗 키를 구성하여 퍼블릭 키가 구성된 인스턴스에 연결할 때 이 암호가 필요합니다.

```
Enter passphrase (empty for no passphrase):
```

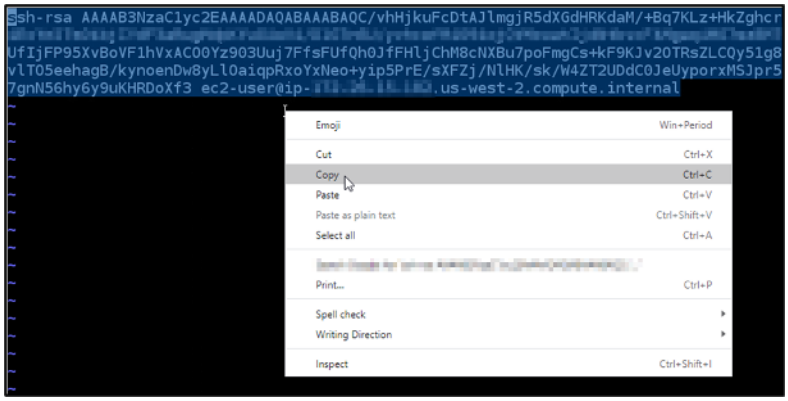
5. 암호를 다시 입력하여 확인하고 Enter 키를 누릅니다. 암호를 입력할 때 암호를 볼 수 없습니다.

```
Enter same passphrase again:
```

6. 프라이빗 키와 퍼블릭 키가 지정된 디렉터리에 저장되었음을 확인하는 메시지가 표시됩니다.

```
Your identification has been saved in C:\Keys\MyNewLighstailCustomKey.
Your public key has been saved in C:\Keys\MyNewLighstailCustomKey.pub.
```

7. 퍼블릭 키(.PUB) 파일을 열고 파일의 텍스트를 복사합니다.

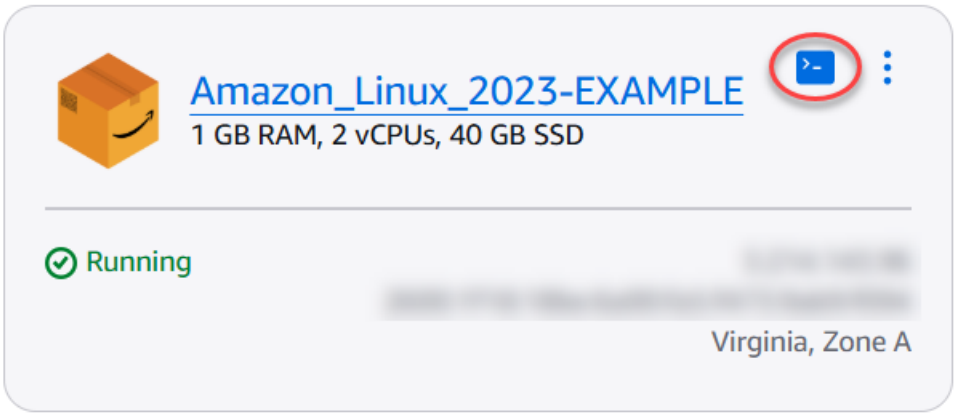


이 가이드의 다음 섹션으로 이동하여 Lightsail 인스턴스에 새 퍼블릭 키를 추가합니다.

3단계: 인스턴스에 퍼블릭 키 추가

인스턴스에 퍼블릭 키를 추가하려면 다음 절차를 완료하세요. 퍼블릭 키 콘텐츠는 Linux 및 Unix 인스턴스의 `~/.ssh/authorized_keys` 파일에 저장됩니다.

1. Lightsail 콘솔에 로그인합니다.
2. Lightsail 홈 페이지에서 인스턴스 섹션을 선택합니다.
3. 연결할 인스턴스에 대한 브라우저 기반 SSH 클라이언트 아이콘을 선택합니다.



4. 연결한 후 다음 명령을 입력하여 원하는 텍스트 편집기를 사용하여 authorized_keys 파일을 편집합니다. 다음 단계에서는 설명을 위해 Vim을 사용합니다.

```
sudo vim ~/.ssh/authorized_keys
```

인스턴스에 구성된 현재 퍼블릭 키를 보여주는 다음 예와 비슷한 결과가 나타나는 것을 볼 수 있습니다. 이 경우 인스턴스가 생성된 AWS 리전 의 Lightsail 기본 키는 인스턴스에 구성된 유일한 퍼블릭 키가 됩니다.

5. Vim 편집기에서 I 키를 눌러 삽입 모드를 시작합니다.
6. 파일의 마지막 퍼블릭 키 뒤에서 줄을 바꿉니다.
7. 새 키 페어를 생성한 후 가이드의 앞부분에서 복사한 퍼블릭 키 텍스트를 붙여 넣습니다. 다음 예와 비슷한 결과가 나타나야 합니다.

8. ESC 키를 누릅니다. 다음으로 `:wq!`를 입력하고 Enter 키를 눌러 편집한 내용을 저장하고 Vim 편집기를 종료합니다.

이제 새 퍼블릭 키가 인스턴스에 추가됩니다. 이 가이드의 다음 섹션으로 계속 진행하여 새 키 페어를 사용하여 인스턴스에 연결합니다.

4단계: 새 키 페어를 사용하여 인스턴스에 연결

새 키 페어를 테스트하려면 인스턴스와 연결을 끊었다가 가이드의 앞부분에서 생성한 프라이빗 키를 사용하여 다시 연결합니다. 자세한 내용은 Amazon Lightsail의 키 페어 및 인스턴스 연결을 참조하세요. 새 키를 사용하여 인스턴스에 성공적으로 연결한 후에는 인스턴스에서 이전 키를 제거할 수 있습니다. 인스턴스에서 퍼블릭 키를 삭제하는 방법을 알아보려면 다음 단계를 계속합니다.

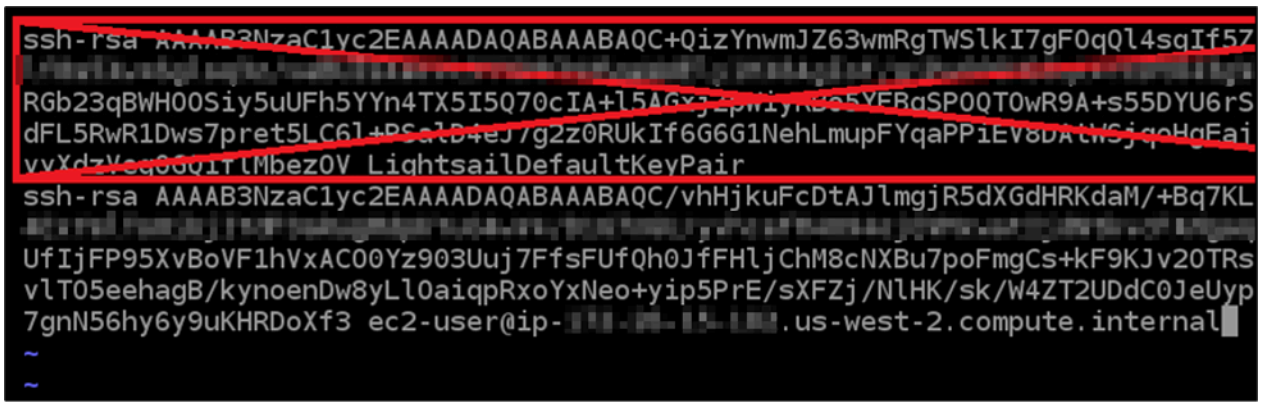
5단계: 인스턴스에서 기존 퍼블릭 키 삭제

인스턴스에서 퍼블릭 키를 제거하려면 다음 절차를 완료하세요. 이를 통해 사용자가 이전 키 페어를 사용하여 인스턴스에 연결하지 못합니다. 새 키 페어를 사용하여 성공적으로 연결한 후 이 작업을 수행합니다.

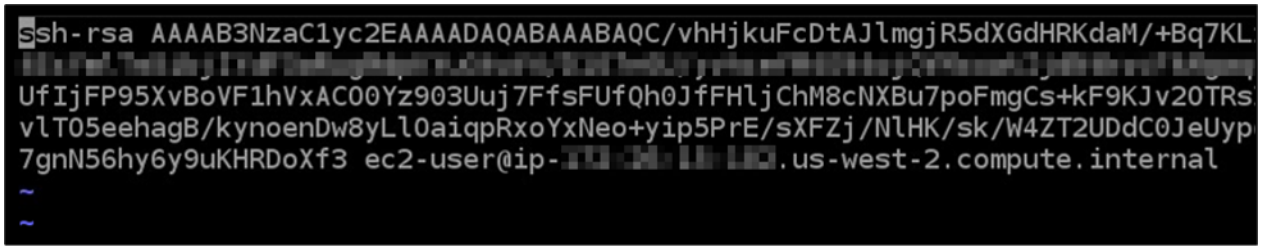
1. SSH로 인스턴스에 연결합니다.
2. 다음 명령을 입력하여 원하는 텍스트 편집기를 사용하여 authorized_keys 파일을 편집합니다. 다음 단계에서는 설명을 위해 Vim을 사용합니다.

```
sudo vim ~/.ssh/authorized_keys
```

3. Vim 편집기에서 I 키를 눌러 삽입 모드를 시작합니다.
4. 인스턴스에서 제거하려는 퍼블릭 키가 포함된 텍스트 행을 삭제합니다.



다음 예시와 같이 새 퍼블릭 키만 표시되어야 합니다.



5. ESC 키를 누릅니다. 다음으로 `:wq!`를 입력하고 Enter 키를 눌러 편집한 내용을 저장하고 Vim 편집기를 종료합니다.

이제 삭제 완료한 퍼블릭 키가 인스턴스에서 제거되었습니다. 이제 인스턴스에서 해당 키 페어의 프라이빗 키를 사용하는 연결을 거부합니다.

Lightsail에 Linux 또는 Unix 인스턴스에 연결

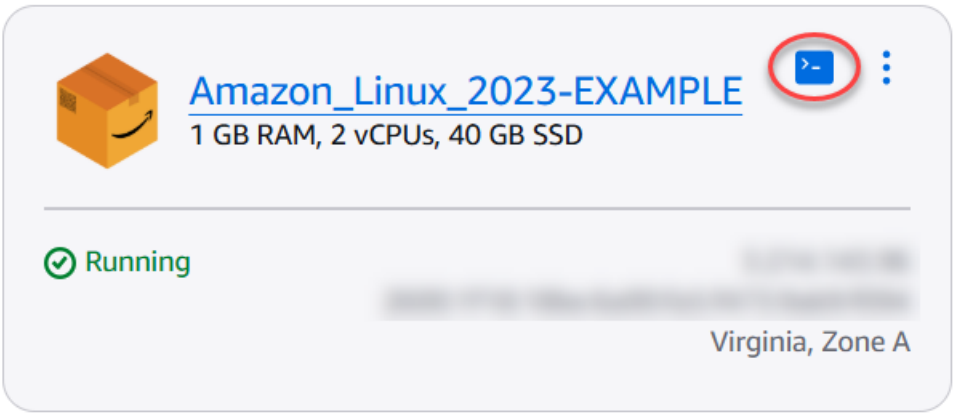
Amazon Lightsail은 브라우저 기반의 SSH 클라이언트를 제공합니다. 이 클라이언트는 Linux 또는 Unix 인스턴스에 연결하는 가장 빠른 방법입니다. 이제 자체 SSH 클라이언트를 사용하여 인스턴스에 연결할 수 있습니다. 자세한 내용은 PuTTY 다운로드 및 설정을 참조하세요.

SSH를 사용하여 인스턴스에 연결하여 소프트웨어 패키지 설치 또는 웹 애플리케이션 구성과 같은 서버에서 관리 작업을 수행합니다. 브라우저 기반 SSH 클라이언트는 소프트웨어 설치가 필요 없으며 인스턴스를 만든 직후에 사용할 수 있습니다.

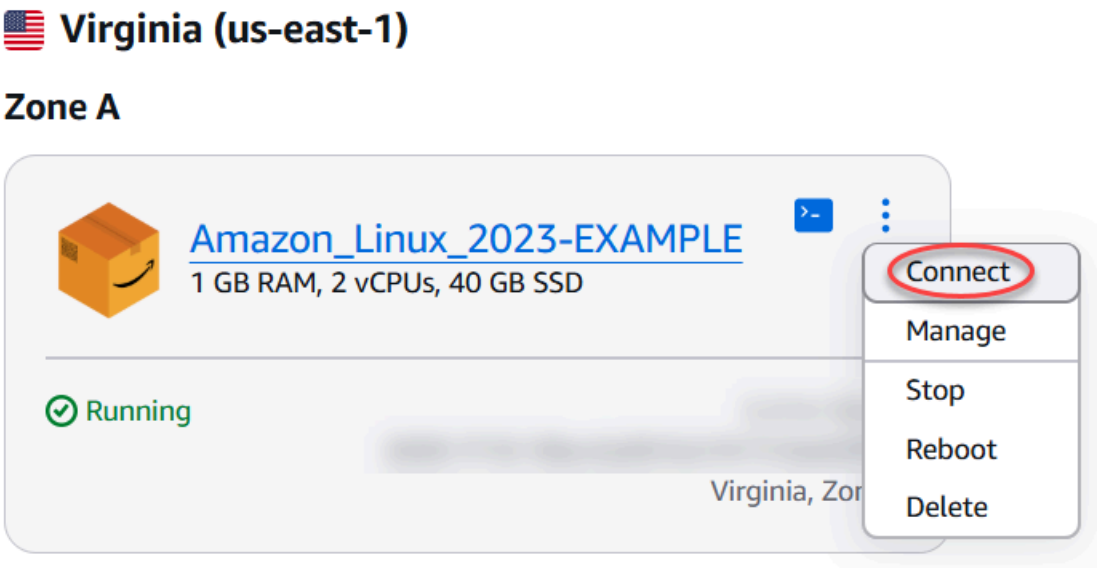
Lightsail에서 Windows Server 인스턴스에 연결하려면 Windows 기반 인스턴스에 연결을 참조하세요.

Linux 또는 Unix 인스턴스에 연결하려면

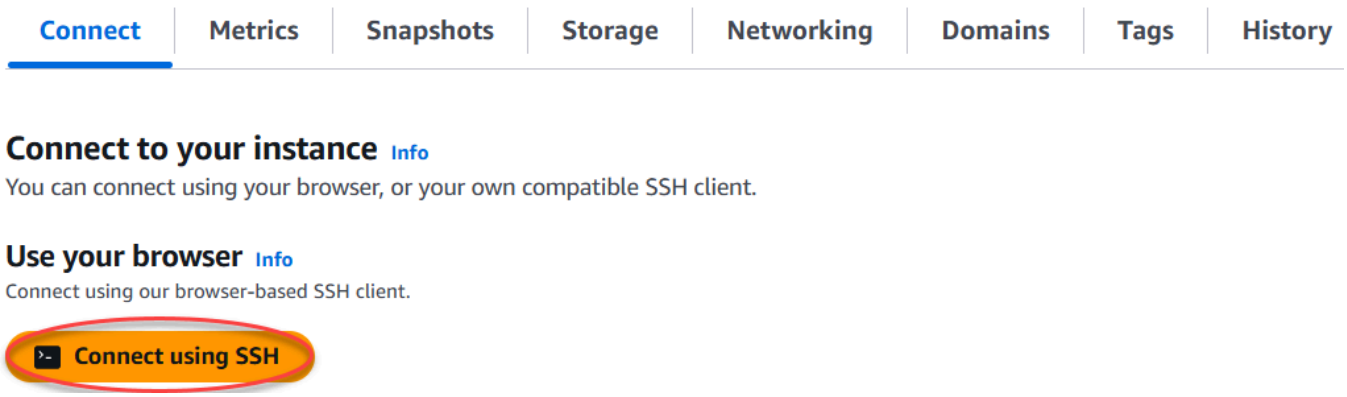
1. Lightsail 콘솔에 로그인합니다.
2. 다음 중 하나를 사용하여 연결할 인스턴스의 브라우저 기반 SSH 클라이언트에 액세스합니다.
   - 다음 예제와 같이 빠른 연결 아이콘을 선택합니다.



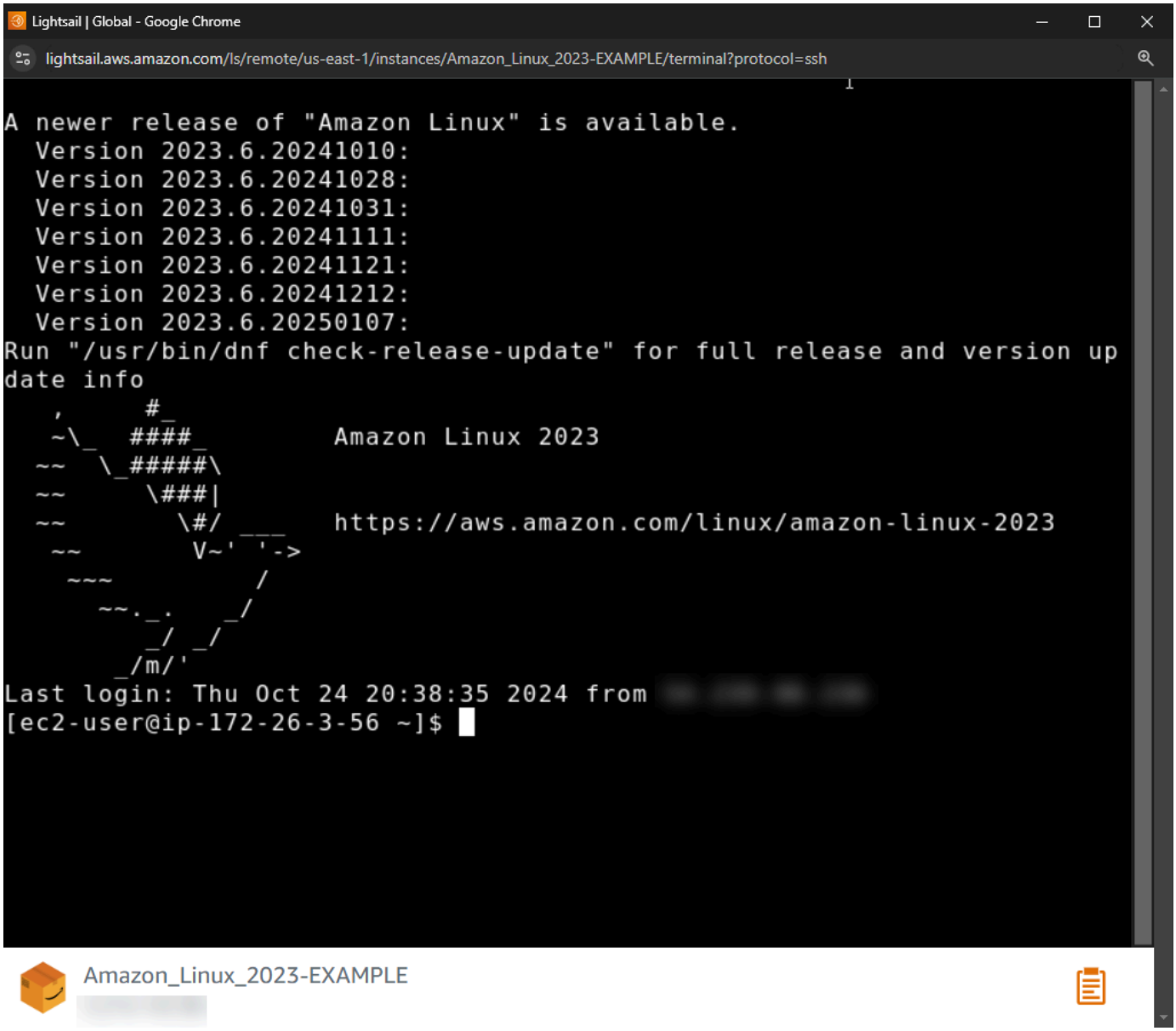
- 작업 메뉴 아이콘(⋮)을 선택한 다음 연결을 선택합니다.



- 인스턴스의 이름을 선택하고 연결 탭에서 SSH를 사용하여 연결을 선택합니다.



브라우저 기반 SSH 클라이언트가 열리면 인스턴스와 상호 작용을 시작할 수 있으며 다음 예와 같이 터미널 화면이 표시됩니다.



> **Note**
>
> 연결 탭은 또한 자신의 SSH 클라이언트를 사용하여 연결하는 데 필요한 정보를 제공합니다. 자세한 내용은 PuTTY 다운로드 및 설정을 참조하세요.

브라우저 기반 SSH 클라이언트를 사용하여 Linux 또는 Unix 인스턴스 연결

Linux 또는 Unix 명령을 터미널 화면에 직접 입력하거나, 터미널 화면에 텍스트를 붙여넣거나, 브라우저 기반 SSH 클라이언트의 터미널 화면에서 텍스트를 복사하십시오. 다음 섹션에서는 SSH의 클립보드에서 텍스트를 복사하여 붙여넣는 방법을 보여줍니다.

브라우저 기반 SSH 클라이언트에 텍스트를 붙여넣으려면

1. 로컬 데스크톱에서 텍스트를 강조 표시한 다음 Ctrl+C 또는 Cmd+C를 눌러 로컬 클립보드로 복사합니다.
2. 브라우저 기반 SSH 클라이언트의 오른쪽 하단에서 클립보드 아이콘을 선택합니다. 브라우저 기반 SSH 클라이언트 클립보드 텍스트 상자가 나타납니다.
3. 텍스트 상자를 클릭한 다음 Ctrl+V 또는 Cmd+V를 눌러 로컬 클립보드의 내용을 브라우저 기반 SSH 클라이언트 클립보드에 붙여넣습니다.
4. SSH 터미널 화면에서 임의의 영역을 마우스 오른쪽 버튼으로 클릭하여 브라우저 기반 SSH 클라이언트 클립보드의 텍스트를 터미널 화면에 붙여넣습니다.

브라우저 기반 SSH 클라이언트에서 텍스트를 복사하려면

1. 터미널 화면에서 텍스트를 강조 표시합니다.
2. 브라우저 기반 SSH 클라이언트의 오른쪽 하단에서 클립보드 아이콘을 선택합니다. 브라우저 기반 SSH 클라이언트 클립보드 텍스트 상자가 나타납니다.
3. 복사할 텍스트를 강조 표시한 다음 Ctrl+C 또는 Cmd+C를 눌러 로컬 클립보드로 텍스트를 복사합니다. 이제 로컬 데스크톱의 아무 곳에나 복사된 텍스트를 붙여넣을 수 있습니다.

SSH 명령을 사용하여 Lightsail Linux 또는 Unix 인스턴스에 연결

로컬 시스템에서 macOS를 포함한 Linux 또는 Unix 운영 체제를 사용하는 경우 Amazon Lightsail에서 터미널 창을 통해 SSH 클라이언트를 사용하여 Linux 또는 Unix 인스턴스에 연결할 수 있습니다.

이 가이드에서는 인스턴스에 연결하는 여러 가지 방법 중 하나를 안내합니다. 다른 방법에 대한 자세한 내용은 SSH 키 페어를 참조하세요.

Lightsail에서 Linux 또는 Unix 인스턴스에 연결하는 가장 쉬운 방법은 Lightsail 콘솔에서 사용할 수 있는 브라우저 기반 SSH 클라이언트를 활용하는 것입니다. 자세한 내용은 Linux 또는 Unix 인스턴스에 연결을 참조하세요.

내용

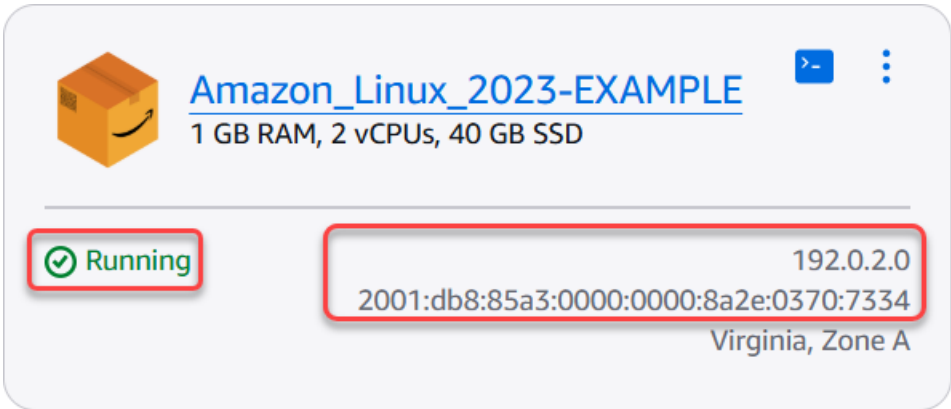
- 1단계: 인스턴스가 실행 중인지 확인하고 퍼블릭 IP 주소 가져오기
- 2단계: 인스턴스에서 사용 중인 SSH 키 페어 확인
- 3단계: 프라이빗 키의 권한을 변경하고 SSH를 사용하여 인스턴스에 연결

1단계: 인스턴스가 실행 중인지 확인하고 퍼블릭 IP 주소 가져오기

다음 절차에서 Lightsail 콘솔에 로그인하여 인스턴스가 실행 중 상태인지 확인하고 인스턴스의 퍼블릭 IP 주소를 가져옵니다. SSH 연결을 설정하려면 인스턴스가 실행 중이어야 하며, 이 가이드의 후반부에서 인스턴스에 연결하려면 인스턴스의 퍼블릭 IP 주소가 필요합니다.

1. Lightsail 콘솔에 로그인합니다.
2. Lightsail 홈 페이지의 인스턴스 섹션에서 연결할 인스턴스를 찾습니다.
3. 인스턴스가 실행 중인 상태인지 확인하고 인스턴스의 퍼블릭 IP 주소를 기록해 둡니다.

   인스턴스의 상태와 퍼블릭 IP 주소는 다음 예와 같이 인스턴스 이름 옆에 나열됩니다.

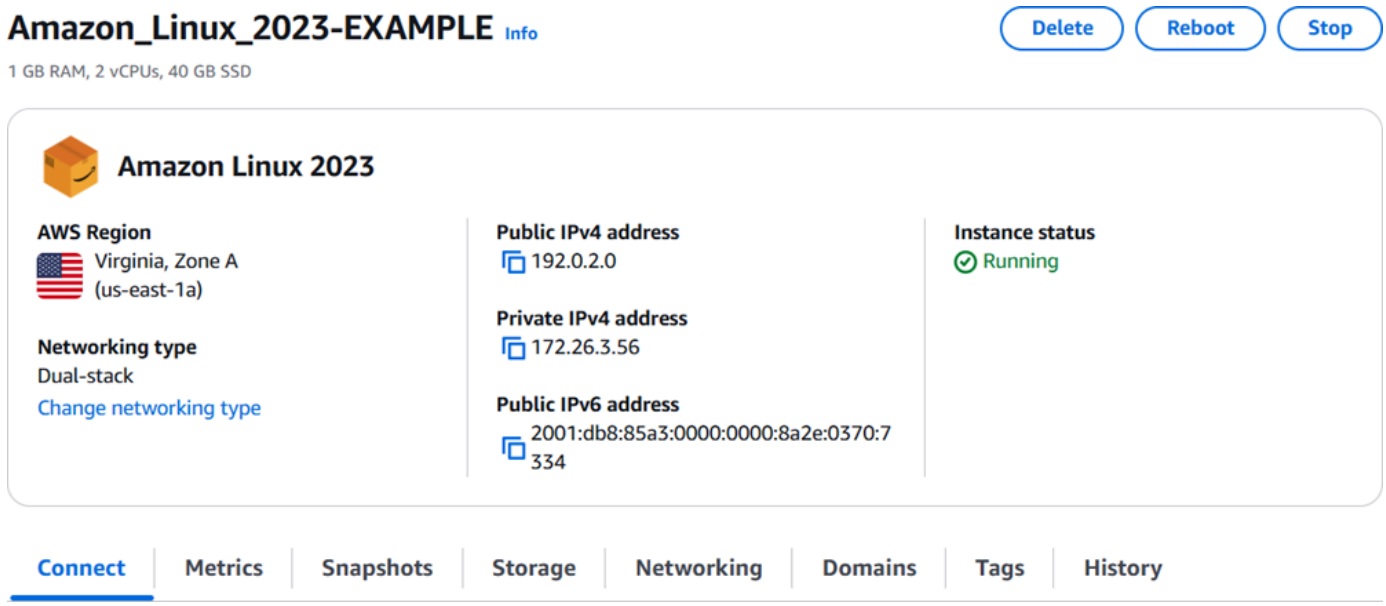


2단계: 인스턴스에서 사용 중인 SSH 키 페어 확인

다음 절차에서는 인스턴스에서 사용 중인 SSH 키 페어를 확인합니다. 인스턴스를 인증하고 SSH 연결을 설정하려면 키 페어의 프라이빗 키가 필요합니다.

1. Lightsail 홈 페이지의 인스턴스 섹션에서 연결할 인스턴스의 이름을 선택합니다.

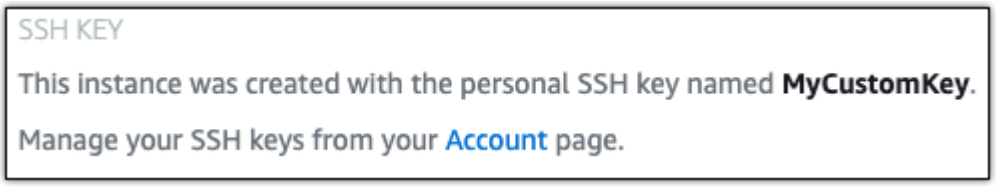
   인스턴스를 관리할 수 있는 여러 탭 옵션이 포함된 인스턴스 관리(Instance management) 페이지가 표시됩니다.



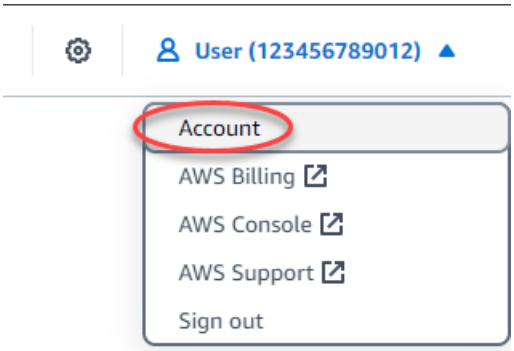
2. 연결(Connect) 탭에서 아래로 스크롤하여 인스턴스에서 사용 중인 키 페어를 확인합니다. 2가지 경우가 있습니다.

   1. 다음 예에서는 인스턴스를 생성한 AWS 리전에 대해 기본 키 페어를 사용하는 인스턴스를 보여줍니다. 인스턴스에서 기본 키 페어를 사용하는 경우 이 절차의 3단계를 계속 진행하여 키 페어의 프라이빗 키를 다운로드할 수 있습니다. Lightsail에서는 각 AWS 리전의 기본 키 페어에 대해서만 프라이빗 키를 저장합니다.

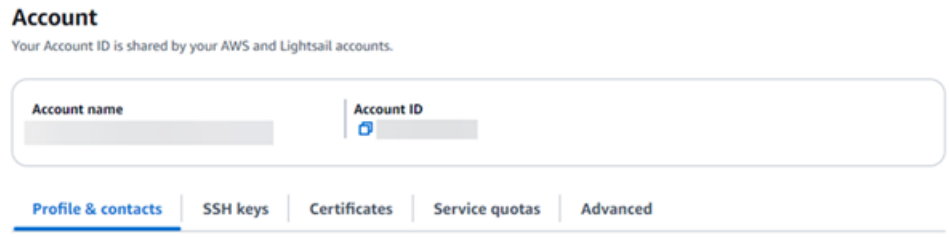
   2. 다음 예에서는 직접 업로드하거나 생성한 사용자 지정 키 페어를 사용하는 인스턴스를 보여줍니다. 인스턴스에서 사용자 지정 키 페어를 사용하는 경우 키를 저장한 사용자 지정 키 페어의 프라이빗 키를 찾아야 합니다. 사용자 지정 키 페어의 프라이빗 키를 분실한 경우 자체 클라이언트를 사용하여 인스턴스에 SSH 연결을 설정할 수 없습니다. 그러나 Lightsail 콘솔에서 지원되는 브라우저 기반 SSH 클라이언트는 계속 사용할 수 있습니다. 사용자 지정 키 페어의 프라이빗 키를 찾은 후 가이드의 3단계: 프라이빗 키의 권한을 변경하고 SSH를 사용하여 인스턴스에 연결 섹션을 이어서 진행합니다.

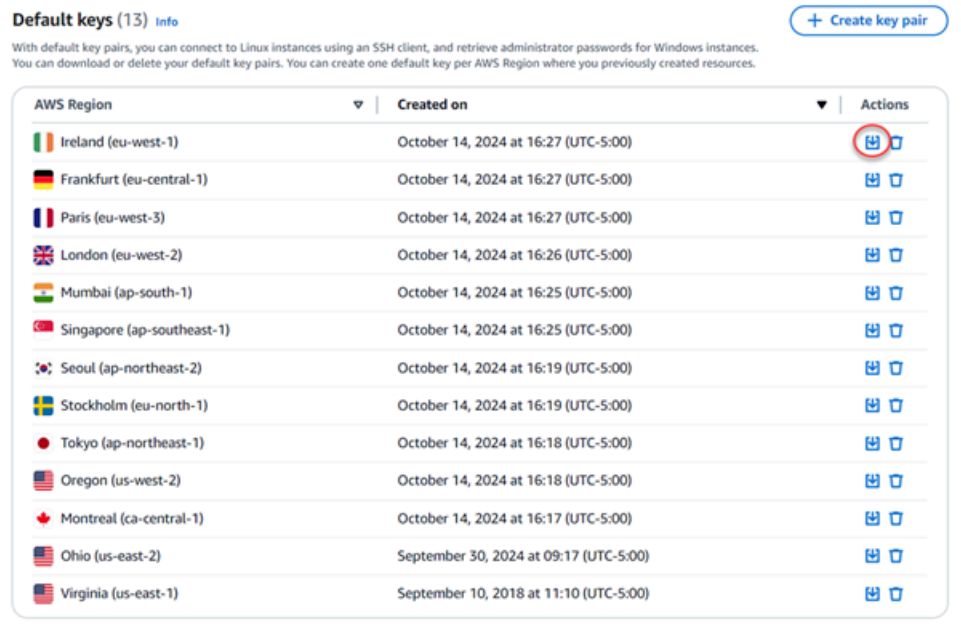
3. Lightsail 홈 페이지의 상단 탐색 메뉴에서 계정을 선택합니다.
4. 드롭다운 메뉴에서 계정(Account)을 선택합니다.



계정 설정을 관리할 수 있는 여러 탭 옵션이 포함된 계정 관리(Account management) 페이지가 표시됩니다.



5. SSH 키(SSH keys) 탭을 선택합니다.
6. 아래로 스크롤하고 연결할 인스턴스의 AWS 리전 의 기본 키 옆에 있는 다운로드 아이콘을 선택합니다.



프라이빗 키가 로컬 시스템에 다운로드됩니다. 다운로드한 키를 사용자의 홈 디렉터리에 있는 '키(Keys)' 폴더와 같이 모든 SSH 키를 저장하는 디렉터리로 옮길 수 있습니다. 이 가이드의 다음 섹션에서 프라이빗 키가 저장된 디렉터리를 참조해야 합니다. 프라이빗 키가 `.pem`이 아닌 다른 형식으로 저장하려고 할 경우 저장하기 전에 수동으로 형식을 `.pem`으로 변경해야 합니다.

> **Note**
>
> Lightsail은 `.pem` 파일 또는 기타 인증서 형식을 조작하기 위한 유틸리티를 제공하지 않습니다. 프라이빗 키 파일의 형식을 변환해야 하는 경우 OpenSSL과 같은 무료 오픈 소스 도구를 손쉽게 사용할 수 있습니다.

방금 다운로드한 프라이빗 키를 사용하고 인스턴스에 SSH 연결을 설정하려면 가이드의 다음 3단계: 프라이빗 키의 권한을 변경하고 SSH를 사용하여 인스턴스에 연결 섹션을 이어서 진행합니다.

3단계: 프라이빗 키의 권한을 변경하고 SSH를 사용하여 인스턴스에 연결

다음 절차에서는 프라이빗 키 파일의 권한을 사용자만 읽고 쓸 수 있도록 변경합니다. 그런 다음 로컬 시스템에서 터미널 창을 열고 SSH 명령을 실행하여 Lightsail에서 인스턴스와의 연결을 설정합니다.

1. 로컬 시스템에서 터미널 창을 엽니다.
2. 다음 명령을 입력하여 키 페어의 프라이빗 키를 사용자만 읽고 쓸 수 있도록 합니다. 이는 일부 운영 체제에서 요구하는 보안 모범 사례입니다.

   ```
   sudo chmod 400 /path/to/private-key.pem
   ```

   명령에서 */path/to/private-key.pem*을 인스턴스에서 사용 중인 키 페어의 프라이빗 키를 저장한 디렉터리 경로로 바꿉니다.

   예:

   ```
   sudo chmod 400 /Users/user/Keys/LightsailDefaultKey-us-west-2.pem
   ```

3. 다음 명령을 입력하여 SSH를 통해 Lightsail에서 인스턴스에 연결합니다.

   ```
   ssh -i /path/to/private-key.pem username@public-ip-address
   ```

명령에서 다음과 같이 바꿉니다.

- */path/to/private-key.pem*을 인스턴스에서 사용 중인 키 페어의 프라이빗 키를 저장한 디렉터리 경로로 바꿉니다.
- *username*을 인스턴스의 사용자 이름으로 바꿉니다. 인스턴스에서 사용하는 블루프린트에 따라 다음 사용자 이름 중 하나를 지정할 수 있습니다.
  - AlmaLinux OS 9, Amazon Linux 2 및 Amazon Linux 2023, CentOS Stream 9, FreeBSD 및 openSUSE 인스턴스의 경우: `ec2-user`
  - Debian 인스턴스: `admin`
  - Ubuntu 인스턴스: `ubuntu`
  - Bitnami 인스턴스: `bitnami`
  - Plesk 인스턴스: `ubuntu`
  - cPanel 및 WHM 인스턴스: `centos`
- *public-ip-address*를 앞서 Lightsail 콘솔에서 확인한 인스턴스의 퍼블릭 IP 주소로 바꿉니다.

절대 경로가 포함된 예:

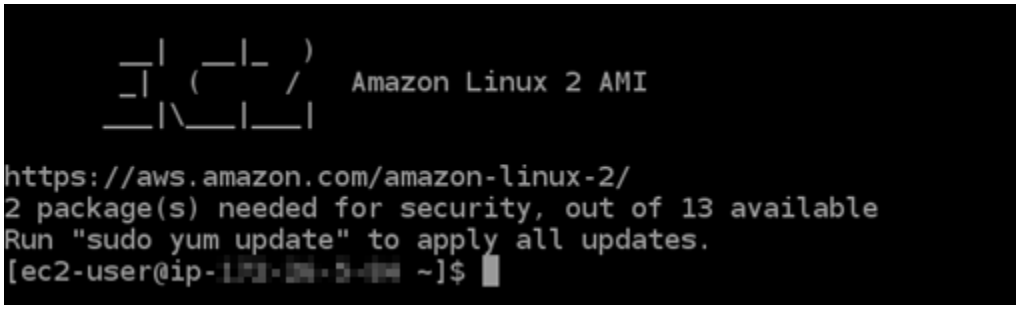
```
ssh -i /Users/user/Keys/LightsailDefaultKey-us-west-2.pem ec2-user@192.0.2.0
```

상대 경로가 포함된 예:

`.pem` 파일의 `./` 접두사에 유의해야 합니다. `./`를 생략하고 단순히 `LightsailDefaultKey-us-west-2.pem`을 쓰면 효과가 없습니다.

```
ssh -i ./LightsailDefaultKey-us-west-2.pem ec2-user@192.0.2.0
```

인스턴스의 시작 메시지가 표시되면 인스턴스에 성공적으로 연결된 것입니다. 다음 예는 Amazon Linux 2 인스턴스의 시작 메시지를 보여줍니다. 다른 인스턴스 블루프린트에도 비슷한 시작 메시지가 사용됩니다. 연결된 후에는 Lightsail의 인스턴스에서 명령을 실행할 수 있습니다. 연결을 끊으려면 `exit`를 입력하고 Enter 키를 누릅니다.



PuTTY를 사용하여 Linux/Unix Lightsail 인스턴스에 연결

Lightsail의 브라우저 기반 SSH 터미널 외에, PuTTY와 같은 SSH 클라이언트를 사용하여 Linux 기반 인스턴스에 연결할 수도 있습니다. PuTTY를 설정하는 방법을 알아보려면 Lightsail에서 SSH를 사용하여 연결할 PuTTY 다운로드 및 설정을 참조하십시오.

> **Note**
>
> RDP를 사용하여 Windows 기반 인스턴스에 연결하려면 Windows 기반 Lightsail 인스턴스에 연결을 참조하십시오.

Lightsail이 제공하는 기본 프라이빗 키, Lightsail의 새 프라이빗 키 또는 다른 서비스와 함께 사용하는 다른 프라이빗 키를 사용할 수 있습니다.

1. PuTTY를 시작합니다(예: 시작 메뉴에서 모든 프로그램, PuTTY, PuTTY 선택).
2. 로드를 선택한 다음 저장한 세션을 찾습니다.

   저장한 세션이 없으면 4단계: 프라이빗 키 및 인스턴스 정보로 PuTTY 구성 완료를 참조하십시오.
3. 인스턴스 운영 체제에 따라 다음 기본 사용자 이름 중 하나를 사용하여 로그인합니다.

   - AlmaLinux, Amazon Linux 2 및 Amazon Linux 2023, CentOS Stream 9, FreeBSD 및 openSUSE 인스턴스의 경우: `ec2-user`
   - Debian 인스턴스: `admin`
   - Ubuntu 인스턴스: `ubuntu`
   - Bitnami 인스턴스: `bitnami`
   - Plesk 인스턴스: `ubuntu`
   - cPanel & WHM 인스턴스: `centos`

인스턴스 운영 체제에 대한 자세한 내용은 Lightsail 인스턴스 이미지 선택을 참조하십시오.

SSH에 대해 자세히 알아보려면 SSH와 Amazon Lightsail 인스턴스 연결을 참조하십시오.

PuTTY를 사용하여 Lightsail Linux 인스턴스에 연결

PuTTY 같은 SSH 클라이언트를 사용하여 Amazon Lightsail 인스턴스에 연결할 수 있습니다. PuTTY에는 프라이빗 SSH 키의 복사본이 필요합니다. 이미 키를 가지고 있을 수도 있고 Lightsail에서 생성되는 키 페어를 사용할 수도 있습니다. 어떤 키를 사용하든 지원됩니다. SSH 키에 대한 자세한 내용은 SSH 키 페어를 참조하세요. 이 주제에서는 키 페어를 다운로드하고 인스턴스에 연결하도록 PuTTY를 설정하는 절차를 안내합니다.

이 가이드에서는 인스턴스에 연결하는 여러 가지 방법 중 하나를 안내합니다. 다른 방법에 대한 자세한 내용은 SSH 키 페어를 참조하세요.

Lightsail에서 Linux 또는 Unix 인스턴스에 연결하는 가장 쉬운 방법은 Lightsail 콘솔에서 사용할 수 있는 브라우저 기반 SSH 클라이언트를 활용하는 것입니다. 자세한 내용은 Amazon Lightsail에서 Linux 또는 Unix 인스턴스에 연결을 참조하세요.

사전 조건

- Lightsail에서 실행 중인 인스턴스가 필요합니다. 자세한 내용은 Amazon Lightsail에서 인스턴스 생성을 참조합니다.
- 나중에 퍼블릭 IP 주소가 변경될 경우 PuTTY를 다시 구성할 필요가 없도록 고정 IP 주소를 만들어 인스턴스에 연결하는 것이 좋습니다. 자세한 내용은 고정 IP를 생성하여 인스턴스에 연결을 참조하세요.

1단계: PuTTY 다운로드 및 설치

PuTTY는 Windows용 SSH를 무료로 구현하는 것입니다. PuTTY 웹사이트에서 암호화가 허용되지 않는 국가와 관련된 제한 사항을 비롯하여 PuTTY에 관한 모든 내용을 알아봅니다. PuTTY가 이미 있는 경우 2단계로 건너뛸 수 있습니다.

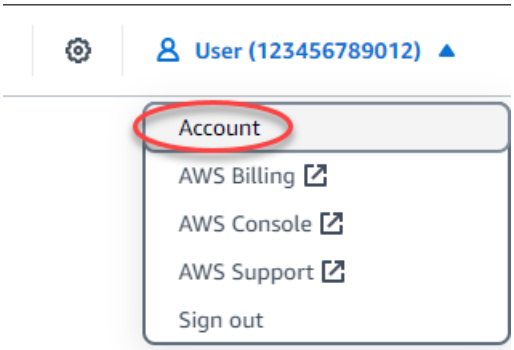
1. Download PuTTY 링크에서 PuTTY 설치 관리자 또는 실행 파일을 다운로드합니다.

   선택할 다운로드 파일을 결정하는 데 도움이 필요할 경우 PuTTY Documentation을 참조하십시오. 최신 버전을 사용하는 것이 좋습니다.
2. 2단계로 이동해 프라이빗 키를 얻은 후 PuTTY를 구성합니다.

2단계: 프라이빗 키 준비

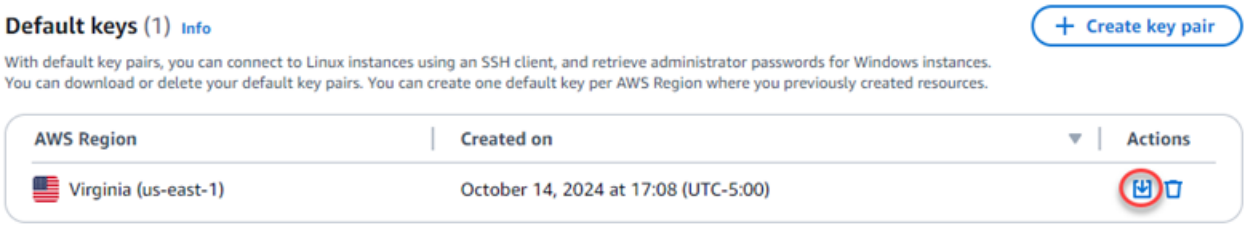
프라이빗 키를 얻는 옵션은 여러 가지가 있습니다. Lightsail에서 생성하는 기본 프라이빗 키를 사용하거나 Lightsail에서 새 프라이빗 키를 생성하게 하거나 다른 서비스에서 제공하는 프라이빗 키가 이미 있을 수 있습니다. 이 옵션마다 수행하는 단계는 다음 절차에서 간략하게 설명합니다.

1. Lightsail 콘솔에 로그인합니다.
2. Lightsail 홈 페이지의 상단 탐색 메뉴에서 계정을 선택합니다.
3. 드롭다운 메뉴에서 계정(Account)을 선택합니다.



4. SSH 키 탭을 선택합니다.
5. 사용할 프라이빗 키에 따라 다음 옵션 중 하나를 선택합니다.

   - 가 Lightsail 생성하는 기본 프라이빗 키를 사용하려면 페이지의 기본 키 섹션에서 인스턴스가 위치한의 기본 프라이빗 키 옆에 AWS 리전 있는 다운로드 아이콘을 선택합니다.



   - Lightsail에서 새 키 페어를 생성하려면 페이지의 사용자 지정 키(Custom keys) 섹션에서 키 페어 생성(Create key pair)을 선택합니다. 인스턴스가 AWS 리전 위치한를 선택하고 생성을 선택합니다. 이름을 입력하고 키 페어 생성을 선택합니다. 프라이빗 키를 다운로드하기 위한 옵션이 제공됩니다.

> **Important**
>
> 이 프라이빗 키를 한 번만 다운로드할 수 있습니다. 안전한 위치에 보관하십시오.

- 자체 키 페어를 사용하려면 새로 업로드를 선택합니다. 인스턴스가 AWS 리전 있는를 선택하고 업로드를 선택합니다. 파일 업로드를 선택한 후 로컬 드라이브에서 파일을 찾습니다. 키 파일을 Lightsail에 업로드할 준비가 되면 키 업로드(Upload key)를 선택합니다.

6. 프라이빗 키를 다운로드하거나 Lightsail에서 새 프라이빗 키를 생성한 경우 쉽게 찾을 수 있는 곳에 `.pem` 키 파일을 저장해야 합니다.

   다른 사람이 읽을 수 없도록 파일의 권한을 설정하는 것이 좋습니다.

3단계: Lightsail 프라이빗 키로 PuTTYgen 구성

`.pem` 키 파일의 복사본이 있으므로 PuTTY Key Generator(PuTTYgen)를 사용하여 PuTTY를 설정할 수 있습니다.

1. PuTTYgen을 시작합니다(예: 시작 메뉴에서 모든 프로그램, PuTTY, PuTTYgen 선택).
2. 로드(Load)를 선택합니다.

   기본적으로 PuTTYgen에는 확장명이 `.ppk`인 파일만 표시됩니다. `.pem` 파일을 찾으려면 모든 유형의 파일을 표시하는 옵션을 선택합니다.

3. `lightsailDefaultKey.pem`을 선택한 다음 열기를 누릅니다.

   PuTTYgen에서 키를 성공적으로 가져왔다고 확인해주면 확인을 선택할 수 있습니다.

4. Save private key(프라이빗 키 저장)를 선택한 다음 암호와 함께 저장하지 않을 것임을 확인합니다.

   추가적인 보안 방법으로서 암호를 만들 경우에는 PuTTY를 사용하여 인스턴스에 연결할 때마다 암호를 입력해야 합니다.

5. 프라이빗 키를 저장하기 위해 이름과 위치를 지정한 후 저장을 선택합니다.
6. PuTTYgen을 닫습니다.

4단계: 프라이빗 키 및 인스턴스 정보로 PuTTY 구성 완료

거의 다 됐습니다! 마지막으로 한 가지만 변경하는 동안 기다려 주십시오.

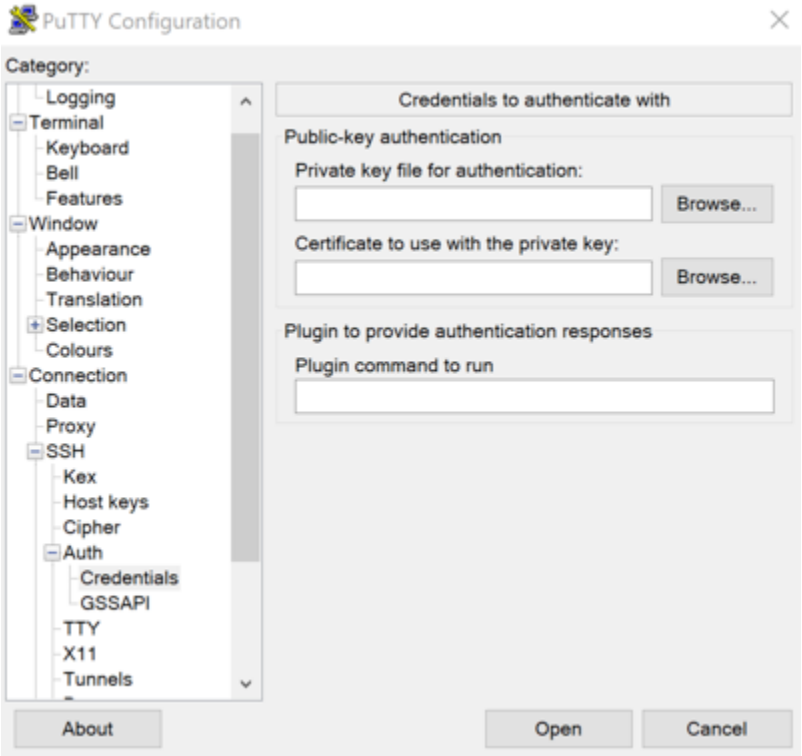
1. PuTTY를 엽니다.
2. Lightsail에서 인스턴스 관리 페이지의 퍼블릭 IP 주소를 가져옵니다(고정 IP 주소 사용 권장).

Lightsail 홈 페이지에서 퍼블릭 IP 주소를 가져오거나 인스턴스를 선택하여 그에 대한 세부 정보를 볼 수 있습니다.

3. 퍼블릭 IP 주소를 Host Name (or IP address)(호스트 이름(또는 IP 주소)) 필드에 입력하거나 붙여 넣습니다.

> **Note**
>
> 포트 22가 Lightsail 인스턴스에서 SSH에 대해 이미 열려 있으므로 기본 포트를 적용합니다.

4. 연결에서 SSH를 펼친 후 인증을 선택한 다음 보안 인증을 선택합니다.



5. 찾아보기를 선택하여 이전 단계에서 생성한 `.ppk` 파일을 탐색한 후 열기를 선택합니다.
6. 열기를 다시 선택한 후 앞으로 이 연결을 신뢰하도록 수락을 선택합니다.
7. 인스턴스 운영 체제에 따라 다음 기본 사용자 이름 중 하나를 사용하여 로그인합니다.

   - AlmaLinux, Amazon Linux 2 및 Amazon Linux 2023, CentOS Stream 9, FreeBSD 및 openSUSE 인스턴스의 경우: `ec2-user`
   - Debian 인스턴스: `admin`
   - Ubuntu 인스턴스: `ubuntu`

- Bitnami 인스턴스: `bitnami`
- Plesk 인스턴스: `ubuntu`
- cPanel 및 WHM 인스턴스: `centos`

인스턴스 운영 체제에 대한 자세한 내용은 이미지 선택을 참조하세요.

8. 앞으로 사용할 것에 대비해 연결 정보를 꼭 저장하십시오.

다음 단계

다시 연결해야 하는 경우 PuTTY를 사용하여 Linux/Unix 기반 인스턴스에 연결을 참조하세요.

SFTP를 사용하여 Lightsail Linux 인스턴스로 안전하게 파일 전송

SFTP(SSH File Transfer Protocol)를 사용하여 인스턴스에 연결함으로써 Amazon Lightsail에서 로컬 컴퓨터와 Linux/Unix 인스턴스 간에 파일을 전송할 수 있습니다. 이를 위해서는 인스턴스에 대한 프라이빗 키를 가져와서 이를 사용해 FTP 클라이언트를 구성해야 합니다. 이 자습서에서는 인스턴스에 연결하도록 FileZilla FTP 클라이언트를 구성하는 방법을 보여줍니다. 여기 나온 단계들은 다른 FTP 클라이언트에도 적용이 가능합니다.

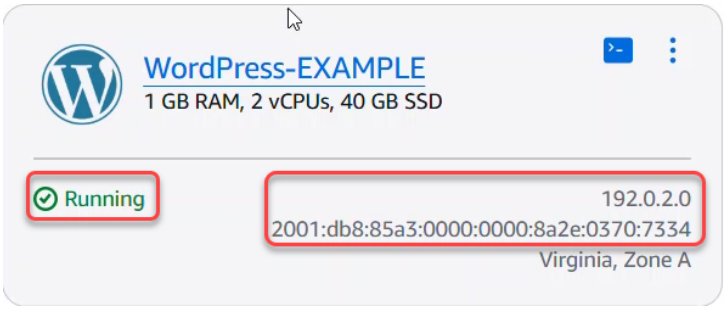
내용

- 사전 조건
- 인스턴스의 SSH 키 가져오기
- FileZilla를 구성하고 인스턴스에 연결

사전 조건

아직 수행하지 않은 경우 다음 사전 조건을 완료합니다.

- 로컬 컴퓨터에 FileZilla를 다운로드하고 설치합니다. 자세한 내용은 다음 다운로드 옵션을 참조하십시오.
  - Windows용 FileZilla 클라이언트 다운로드
  - Mac OS X용 FileZilla 클라이언트 다운로드
  - Linux용 FileZilla 클라이언트 다운로드
- 인스턴스의 퍼블릭 IP 주소를 가져옵니다. 다음 예제에서와 같이 Lightsail 콘솔에 로그인한 다음, 인스턴스 옆에 표시되는 퍼블릭 IP 주소를 복사합니다.

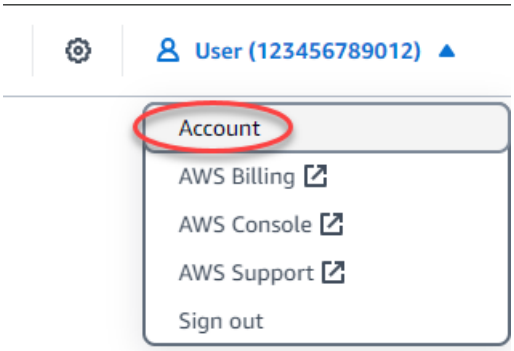


인스턴스의 SSH 키 가져오기

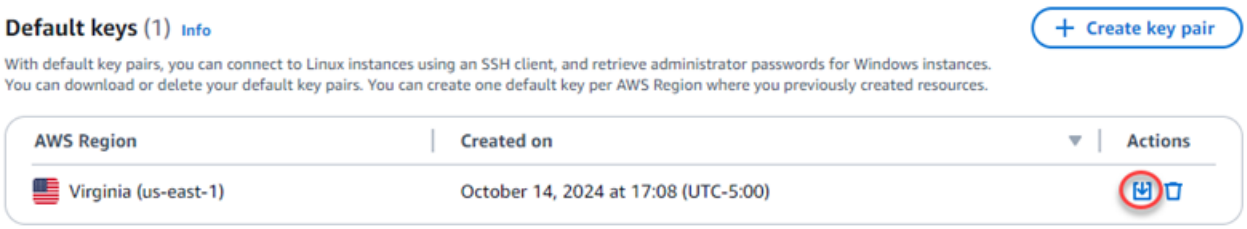
아래의 단계를 완료하여 인스턴스의 AWS 리전을 위한 기본 프라이빗 키를 가져오십시오. FileZilla를 사용하여 인스턴스에 연결하려면 이 키가 필요합니다.

> **Note**
>
> 자체 키 페어를 사용 중이거나 Lightsail 콘솔을 사용하여 키 페어를 생성한 경우에는 자체 프라이빗 키를 찾아서 이를 사용해 인스턴스에 연결합니다. 사용자가 자체 키를 업로드하거나 Lightsail 콘솔을 사용해 키 페어를 생성한 경우에는 Lightsail가 프라이빗 키를 저장하지 않습니다. 프라이빗 키가 없는 SFTP를 사용하면 인스턴스에 연결할 수 없습니다.

1. Lightsail 콘솔에 로그인합니다.
2. Lightsail 홈 페이지의 상단 탐색 메뉴에서 사용자 또는 역할을 선택합니다.
3. 드롭다운 메뉴에서 계정을 선택합니다.



4. SSH 키 탭을 선택합니다.
5. 페이지의 아래로 스크롤하여 기본 키(Default keys) 섹션으로 이동합니다.
6. 인스턴스가 위치한 리전의 기본 프라이빗 키 옆에 있는 다운로드를 선택합니다.

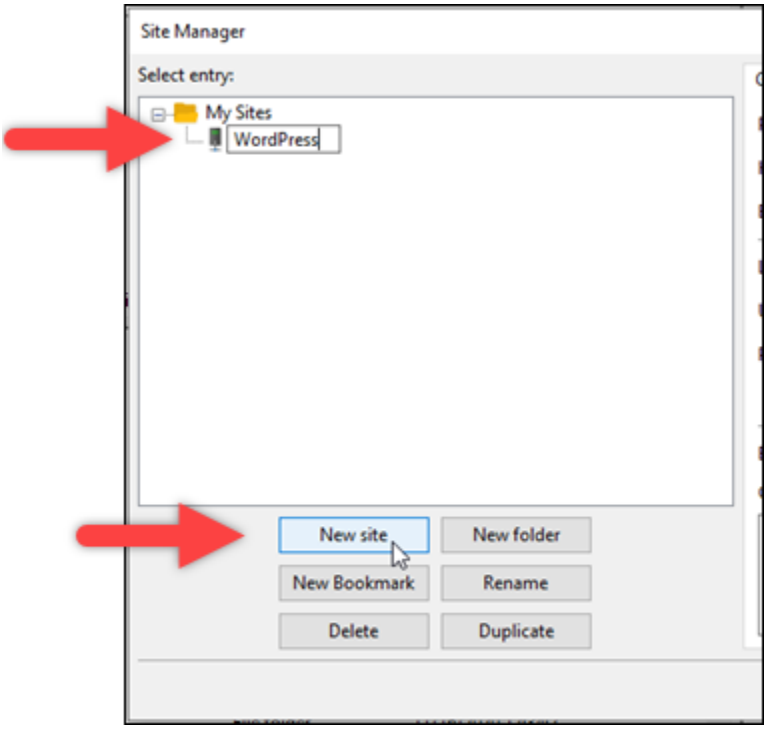


7. 로컬 드라이브의 안전한 위치에 프라이빗 키를 저장합니다.

FileZilla를 구성하고 인스턴스에 연결

아래의 단계를 완료하여 인스턴스에 연결하도록 FileZilla를 구성합니다.

1. FileZilla를 엽니다.
2. 파일과 Site Manager(사이트 관리자)를 차례로 선택합니다.
3. 새 사이트(New site)를 선택하고 사이트 이름을 지정합니다.

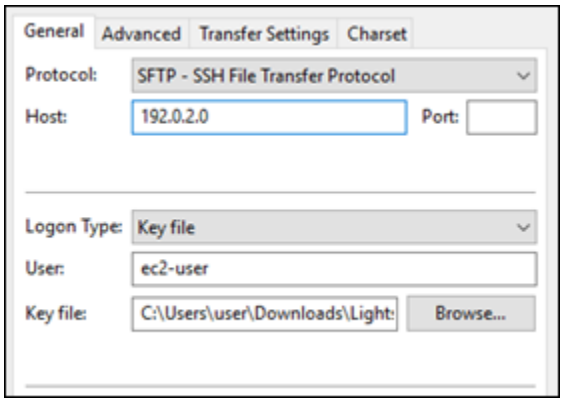


4. 프로토콜 드롭다운 메뉴에서 SFTP(SSH File Transfer Protocol)를 선택합니다.
5. 호스트 텍스트 상자에 인스턴스의 퍼블릭 IP 주소를 입력하거나 붙여 넣습니다.
6. Logon Type(로그인 유형) 드롭다운 메뉴에서 Key File(키 파일)을 선택합니다.
7. 사용자 텍스트 상자에 인스턴스 운영 체제에 따라 다음 기본 사용자 이름 중 하나를 입력합니다.

- AlmaLinux, Amazon Linux 2 및 Amazon Linux 2023, CentOS Stream 9, FreeBSD 및 openSUSE 인스턴스의 경우: `ec2-user`
- Debian 인스턴스: `admin`
- Ubuntu 인스턴스: `ubuntu`
- Bitnami 인스턴스: `bitnami`
- Plesk 인스턴스: `ubuntu`
- cPanel & WHM 인스턴스: `centos`

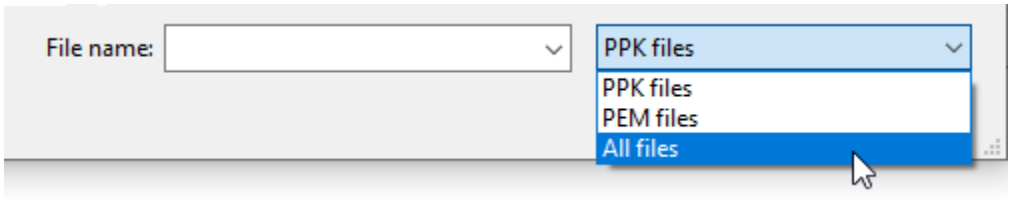
> **Important**
>
> 여기에 나열된 기본 사용자 이름과는 다른 사용자 이름을 활용하는 경우 사용자에게 인스턴스에 대한 쓰기 권한을 부여해야 할 수 있습니다.

8. Key File(키 파일) 텍스트 상자 옆의 찾아보기를 선택합니다.

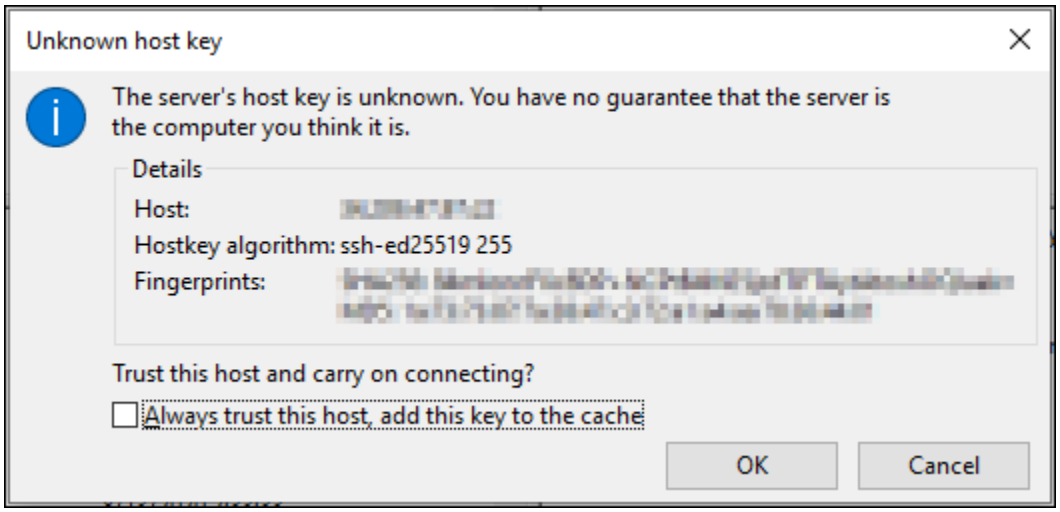


9. 이 절차 앞부분에 나온 Lightsail 콘솔에서 다운로드한 프라이빗 키 파일을 찾은 다음, 열기를 선택합니다.

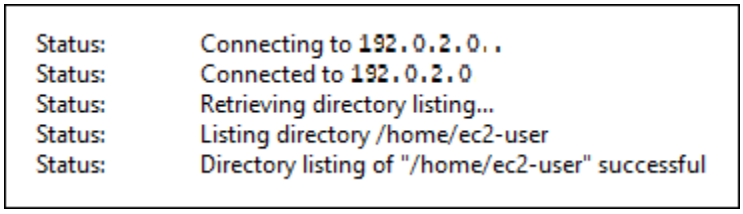
> **Note**
>
> Windows를 사용하는 경우 pem 파일을 검색할 때 기본 파일 유형을 모든 파일(All files)로 변경합니다.



10. 연결을 선택합니다.
11. 다음 예와 유사한 호스트 키를 알 수 없다는 메시지가 표시될 수 있습니다. 확인(OK)을 선택하여 프롬프트를 확인하고 인스턴스에 연결합니다.



다음 예제와 비슷한 상태 메시지가 나타나면 성공적으로 연결이 된 것입니다.



로컬 컴퓨터와 인스턴스 간의 파일 전송 방법을 포함하여 FileZilla 사용에 관한 자세한 내용은 FileZilla Wiki 페이지를 참조하십시오.

RDP를 사용하여 Lightsail Windows 인스턴스에 연결

Lightsail 콘솔에서 제공되는 브라우저 기반 RDP 클라이언트를 사용하여 Amazon Lightsail에서 Windows Server 인스턴스에 연결할 수 있습니다. 브라우저 기반 RDP 클라이언트는 소프트웨어를 설치할 필요가 없으며, 인스턴스를 생성한 직후 Windows Server 인스턴스에 연결하여 바로 사용할 수 있

습니다. 인스턴스에 연결하여 소프트웨어 설치 또는 웹 애플리케이션 구성과 같은 서버 관리 작업을 수행합니다.

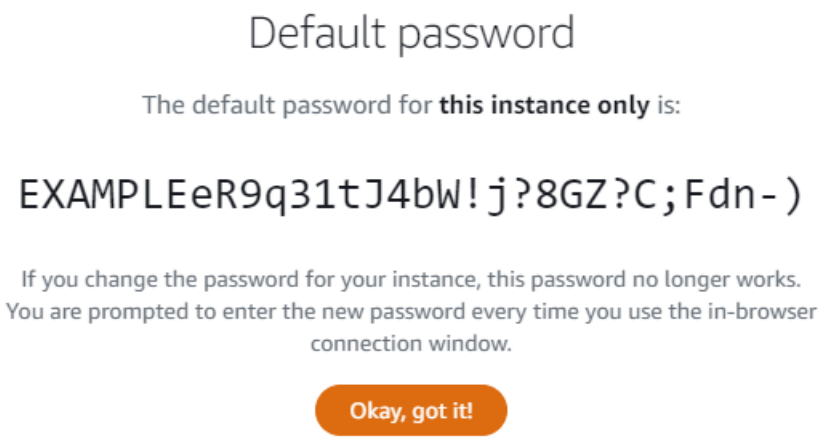
자체 RDP 클라이언트(예: Windows와 번들로 함께 제공되는 원격 데스크톱 연결)를 사용하여 인스턴스에 연결할 수도 있습니다. 자체 RDP 클라이언트를 구성하는 방법에 대한 자세한 내용은 원격 데스크톱 연결 클라이언트를 사용하여 Windows 인스턴스에 연결을 참조하세요. Lightsail에서 Linux 또는 Unix 인스턴스에 연결하려면 Linux 또는 Unix 인스턴스에 연결을 참조하세요.

Windows Server 인스턴스의 기본 관리자 암호

Windows Server 인스턴스를 생성하면 무작위로 생성된 기본 관리자 암호가 인스턴스에 할당됩니다. Lightsail 콘솔의 브라우저 기반 RDP 클라이언트는 기본 관리자 암호를 사용하여 인스턴스에 로그인합니다. 인스턴스의 관리자 암호를 변경하면 브라우저 기반 RDP 클라이언트를 사용하여 인스턴스에 연결하려고 할 때마다 새 암호를 수동으로 입력하라는 메시지가 표시됩니다. Lightsail은 새 관리자 암호를 저장하지 않으며 인스턴스에서 가져올 수 없습니다.

> **Important**
>
> 관리자 암호를 분실하면 인스턴스에 로그인할 수 없으며 암호를 재설정할 수도 없습니다. AWS 새 관리자 암호를 분실한 경우 나중에 검색할 수 있는 안전한 위치에 저장합니다. 자세한 내용은 AWS Secrets Manager 사용 설명서를 참조하세요.

브라우저 기반 RDP 클라이언트를 사용하여 인스턴스에 액세스할 때마다 암호를 묻는 메시지가 표시되지 않도록 관리자 암호를 원래 기본 관리자 암호로 다시 변경할 수 있습니다. Lightsail 홈 페이지에서 인스턴스(Instances) 탭을 선택하여 원래 기본 관리자 암호를 찾을 수 있습니다. Windows Server 인스턴스의 이름을 선택한 다음 연결 탭에서 기본 암호 표시를 선택하면 다음 예와 같이 원래 기본 관리자 암호가 표시됩니다.

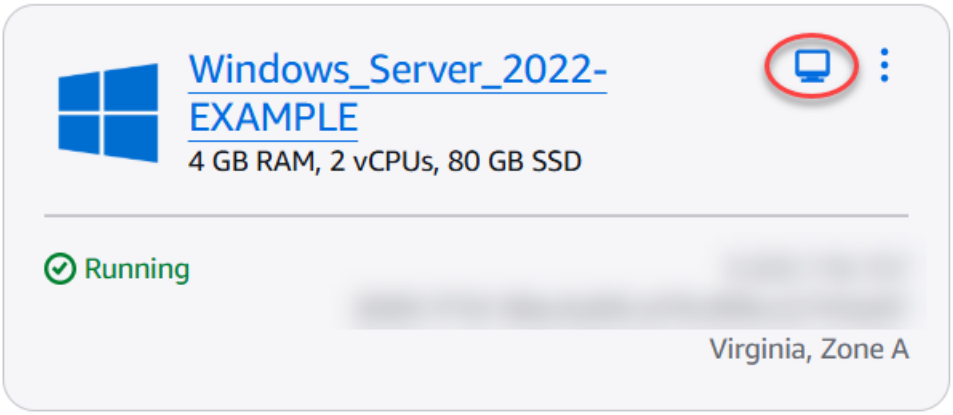


브라우저 기반 RDP 클라이언트를 사용하여 Windows Server 인스턴스에 연결

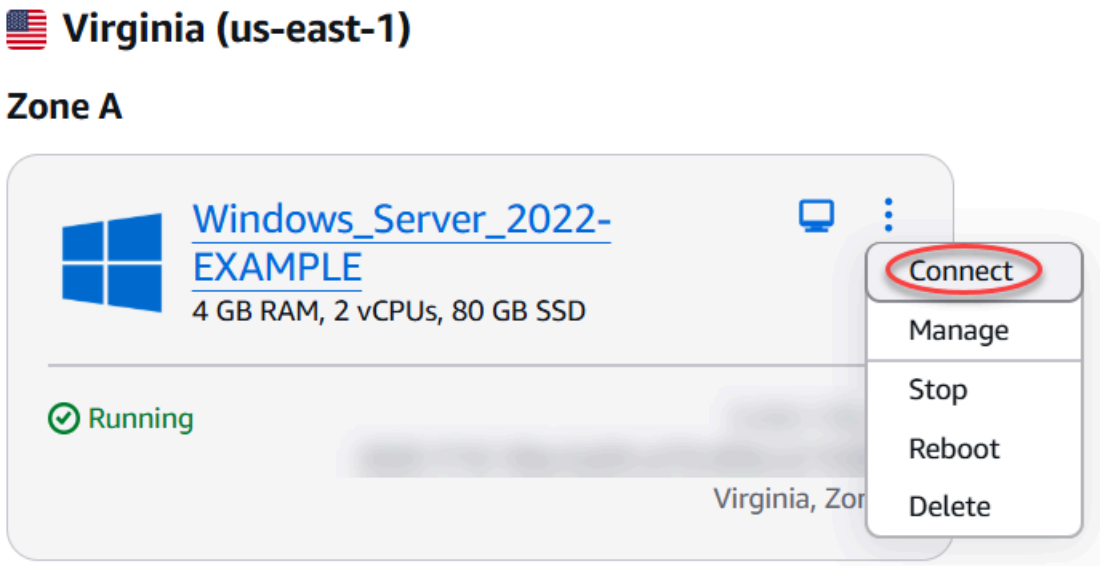
Lightsail 콘솔에서 브라우저 기반 RDP 클라이언트를 사용하여 Windows Server 인스턴스에 연결하려면 다음 절차를 따르십시오.

1. Lightsail 콘솔에 로그인합니다.
2. 다음 단계 중 하나를 사용하여 연결할 인스턴스의 브라우저 기반 RDP 클라이언트에 액세스합니다.

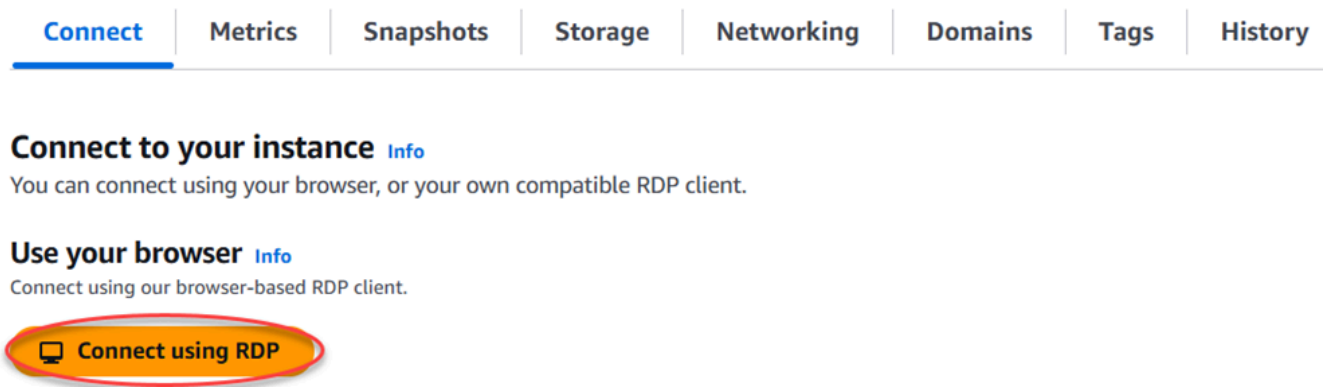
   - 다음 예와 같이 브라우저 기반 RDP 클라이언트 아이콘을 선택합니다.



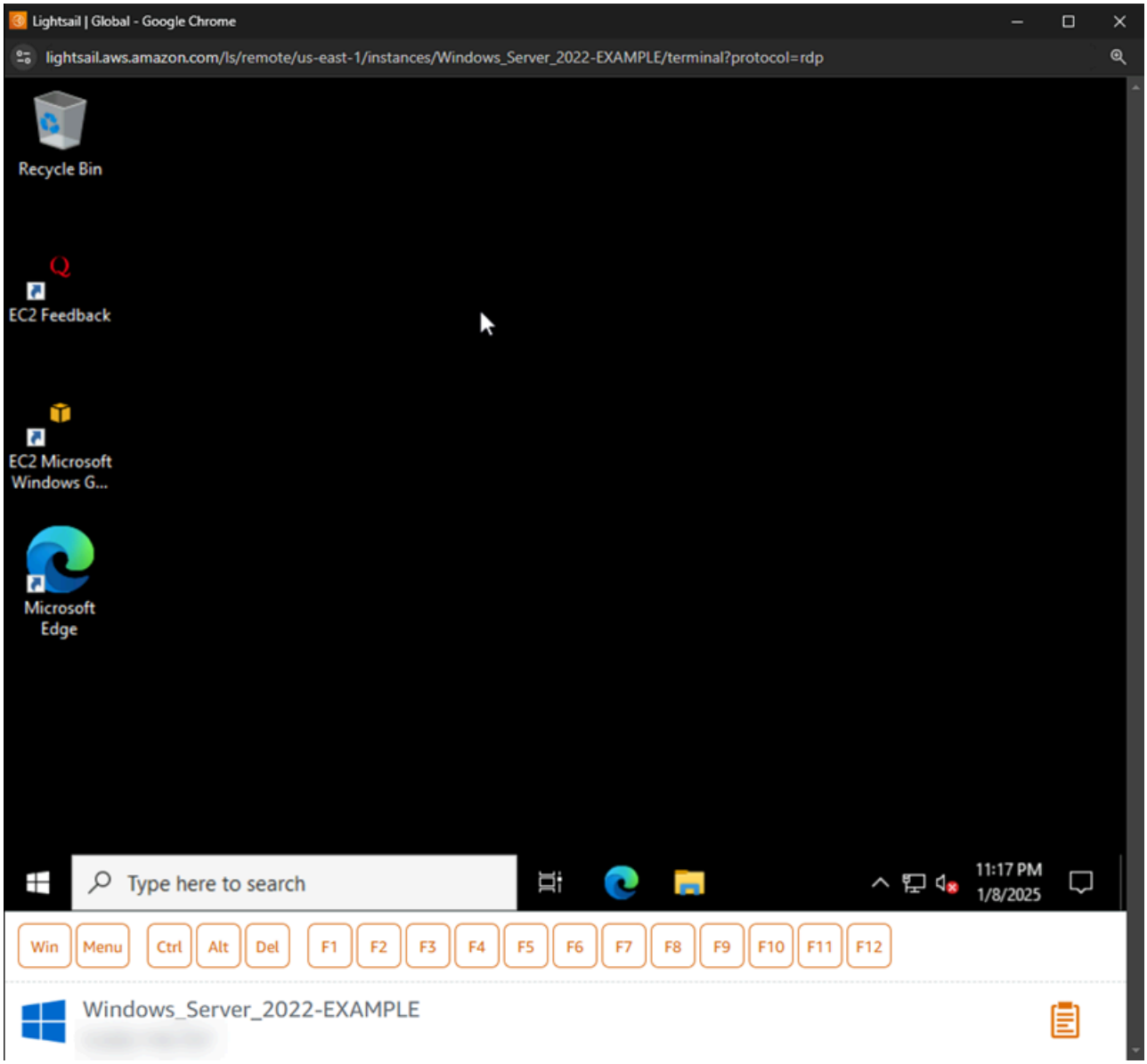
   - 작업 메뉴 아이콘(⋮)을 선택하고 다음 예와 같이 연결을 선택합니다.



   - 인스턴스의 이름을 선택하고 연결 탭에서 RDP를 사용하여 연결을 선택합니다.



브라우저 기반 RDP 클라이언트가 열리면 인스턴스와 상호 작용을 시작할 수 있으며 다음 예와 같이 Windows 바탕 화면이 표시됩니다.



> **Note**
>
> 연결 탭에는 Windows 인스턴스의 기본 사용자 이름 및 암호와 같은 자체 RDP 클라이언트를 사용하여 연결하는 데 필요한 정보도 제공됩니다. 자체 RDP 클라이언트를 구성하는 방법에 대한 자세한 내용은 원격 데스크톱 연결 클라이언트를 사용하여 Amazon Lightsail에서 Windows 인스턴스에 연결을 참조하십시오.

브라우저 기반 RDP 클라이언트를 사용하여 Windows 인스턴스 연결

로컬 Windows 데스크톱과 같은 브라우저 기반 RDP 클라이언트를 사용합니다. RDP에는 기능 키 및 인스턴스와 상호 작용하는 데 도움이 되는 Windows 관련 특정 키가 포함됩니다. 다음 섹션에서는 RDP의 클립보드에서 텍스트를 복사하여 붙여넣는 방법을 보여줍니다.

브라우저 기반 RDP 클라이언트에 텍스트를 붙여넣으려면

1. 로컬 데스크톱에서 텍스트를 강조 표시한 다음 Ctrl+C 또는 Cmd+C를 눌러 로컬 클립보드로 복사합니다.
2. 브라우저 기반 RDP 클라이언트의 오른쪽 하단에서 클립보드 아이콘을 선택합니다. 브라우저 기반 RDP 클라이언트 클립보드 텍스트 상자가 나타납니다.
3. 텍스트 상자를 클릭한 다음 Ctrl+V 또는 Cmd+V를 눌러 로컬 클립보드의 내용을 브라우저 기반 RDP 클라이언트 클립보드에 붙여넣습니다.
4. 원격 데스크톱 화면에서 임의의 영역을 마우스 오른쪽 버튼으로 클릭하여 브라우저 기반 RDP 클라이언트 클립보드의 텍스트를 원격 데스크톱 화면에 붙여넣습니다.

브라우저 기반 RDP 클라이언트에서 텍스트를 복사하려면

1. 원격 데스크톱 화면에서 텍스트를 강조 표시합니다.
2. 브라우저 기반 RDP 클라이언트의 오른쪽 하단에서 클립보드 아이콘을 선택합니다. 브라우저 기반 RDP 클라이언트 클립보드 텍스트 상자가 나타납니다.
3. 복사할 텍스트를 강조 표시한 다음 Ctrl+C 또는 Cmd+C를 눌러 로컬 클립보드로 텍스트를 복사합니다. 이제 로컬 데스크톱의 아무 곳에나 복사된 텍스트를 붙여넣을 수 있습니다.

Lightsail Windows 인스턴스에 대한 관리자 암호 변경

Windows Server 기반 Lightsail 인스턴스를 생성하면 인스턴스를 생성하는 AWS 리전 의 기본 암호가 사용됩니다. 이렇게 하면 브라우저 기반의 RDP(원격 데스크톱) 클라이언트는 물론 원격 데스크톱 연결 등의 클라이언트를 이용하여 연결하기가 더 쉬워집니다.

> **Important**
>
> 인스턴스 암호를 Lightsail에서 자동 지정하도록 하는 것이 좋습니다. 사용자 지정 암호는 저장되지 않기 때문에 관리자 암호를 변경했다가 Lightsail 인스턴스에 액세스하지 못하게 될 위험이 있습니다.

Windows Server를 사용하여 관리자 암호 변경

Windows Server 암호 변경 도구를 사용하여 관리자 암호를 변경할 수 있습니다. `Ctrl` + `Alt` + `Del`을 Windows Server 기반 Lightsail 인스턴스에 입력하고 암호 변경을 선택합니다.

를 사용하여 Lightsail 키 페어의 사이퍼텍스트 가져오기 AWS CLI

Windows Server 기반 Lightsail 인스턴스에서 암호를 변경하는 경우 AWS Command Line Interface (AWS CLI)를 사용하여 암호를 해독하는 데 도움이 되는 정보를 얻을 수 있습니다.

> **Note**
>
> Lightsail는 .pem 파일을 조작하기 위한 유틸리티를 제공하지 않습니다. 프라이빗 키 파일의 형식을 변환해야 하는 경우 OpenSSL용 Linux, Windows용 base64와 같은 무료 오픈 소스 도구를 손쉽게 사용할 수 있습니다.

사이퍼텍스트 가져오기

1. AWS CLI를 아직 설치 및 구성하지 않았다면 설치하고 구성합니다.

   자세한 내용은 AWS Command Line Interface 가 Amazon Lightsail과 함께 작동하도록 구성을 참조하세요.
2. 명령 프롬프트 또는 터미널을 엽니다.
3. 다음 명령을 입력합니다.

   ```
   aws lightsail get-instance-access-details --instance-name my-instance
   ```

   여기서 *my-instance*는 정보를 얻고자 하는 인스턴스의 이름입니다.

   출력은 다음과 비슷합니다.

```
{
    "accessDetails": {
        "username": "Administrator",
        "protocol": "rdp",
        "ipAddress": "12.345.678.910",
        "passwordData": {
            "ciphertext": "cipher",
            "keyPairName": "my-ohio-key"
        },
        "password": "",
        "instanceName": "2016-ohio-windows"
    }
}
```

4. 사용 가능한 모든 애플리케이션에 이 암호화 텍스트를 사용하여 암호를 해독할 수 있습니다.

원격 데스크톱으로 Windows에서 Lightsail Windows 인스턴스에 연결

Windows 운영 체제에 포함된 RDC(원격 데스크톱 연결) 클라이언트를 사용하여 Amazon Lightsail에서 Windows 인스턴스에 연결할 수 있습니다. RDC를 사용하려면 Windows 인스턴스의 관리자 사용자 이름과 암호를 사용해야 합니다. 이 암호는 생성 당시 인스턴스에 지정된 기본 암호이거나, 기본 암호를 변경한 경우 직접 지정한 암호입니다.

이 주제에서는 Lightsail 콘솔에서 기본 관리자 암호를 가져오고 RDC를 구성하여 Windows 인스턴스에 연결하는 단계를 안내합니다. 브라우저를 사용해 Lightsail 콘솔에서 인스턴스에 연결할 수도 있습니다. 자세한 내용은 웹 기반 RDP 클라이언트를 사용하여 Windows 인스턴스에 연결을 참조하세요.

Windows 인스턴스의 기본 관리자 암호 가져오기

아래의 단계를 완료하여 Windows 인스턴스의 기본 관리자 암호를 가져옵니다. RDC를 사용하여 인스턴스에 연결하려면 이 암호가 필요합니다.

> **Note**
>
> 기본 관리자 암호를 변경한 경우 인스턴스의 Lightsail 콘솔에 표시되는 암호를 사용할 수 없습니다. 암호를 기억해 두십시오. 관리자 암호가 없으면 RDC를 사용하여 인스턴스에 연결할 수 없습니다.

1. Lightsail 콘솔에 로그인합니다.

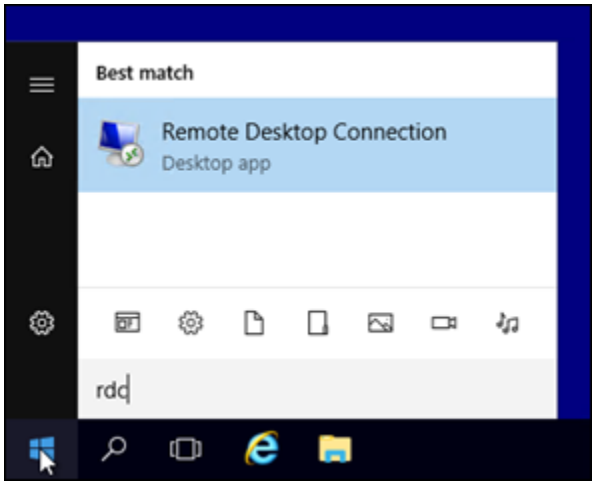
2. 연결하려는 Windows 인스턴스를 선택합니다.
3. 인스턴스 관리 페이지의 연결 탭에서 기본 암호 표시를 선택합니다.
4. 표시되는 기본 암호를 강조 표시하고 Ctrl+C 또는 Cmd+C를 눌러 이 암호를 복사합니다. 이제 암호가 클립보드에 저장됩니다.

   이 안내서의 다음 단원으로 넘어가 RDC를 구성하고 암호를 클라이언트에 붙여 넣습니다.

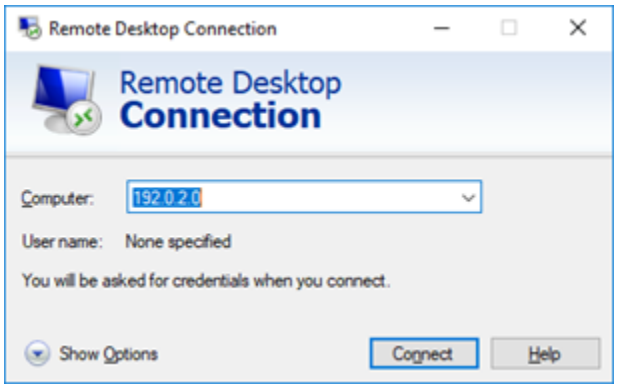
RDC를 구성하고 Windows 인스턴스에 연결

아래의 단계를 완료하여 RDC를 구성하고 Windows 인스턴스에 연결합니다.

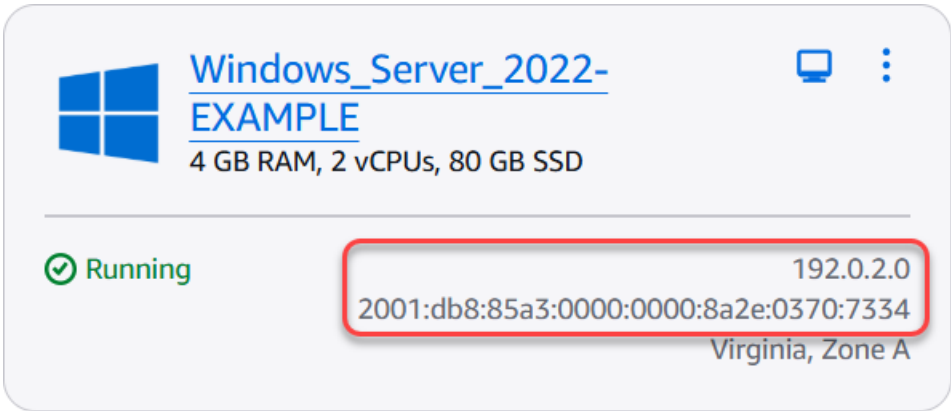
1. Windows 메뉴를 열고 `Remote Desktop Connection` 또는 RDC를 검색합니다.
2. 검색 결과에서 원격 데스크톱 연결을 선택합니다.



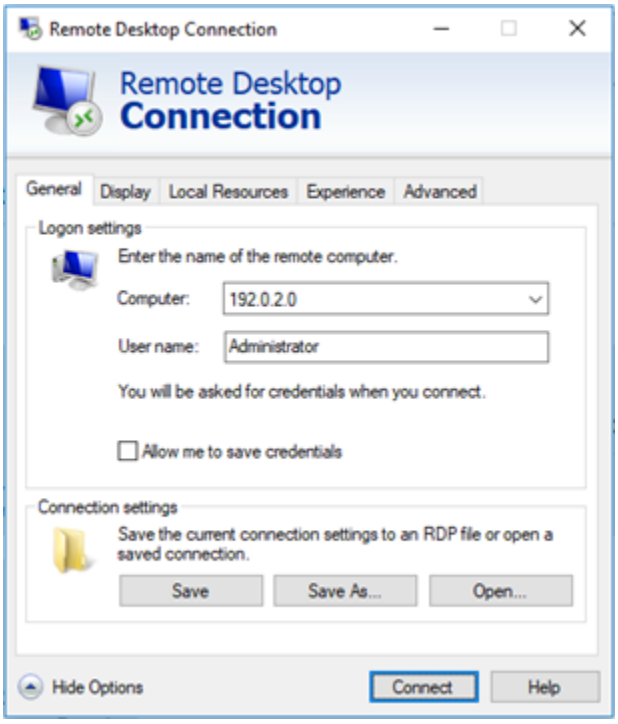
3. 컴퓨터 텍스트 상자에 Windows 인스턴스의 퍼블릭 IP 주소를 입력합니다.



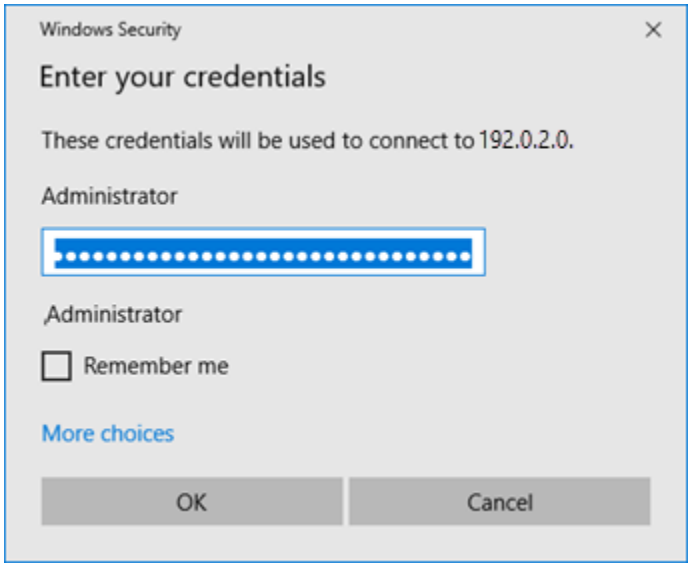
   다음 예와 같이 Lightsail 콘솔에서 인스턴스 옆에 퍼블릭 IP가 표시됩니다.



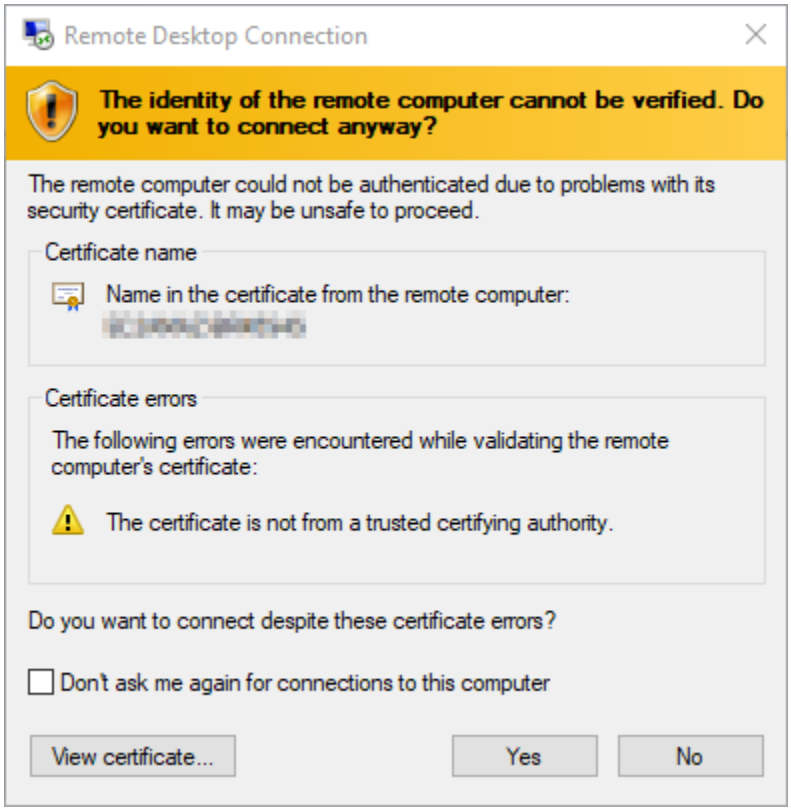
4. Show Options(옵션 표시)를 선택하여 추가 연결 옵션을 표시합니다.
5. 사용자 이름 텍스트 상자에 `Administrator`를 입력합니다. Lightsail의 모든 Windows 인스턴스에 지정되는 기본 사용자 이름입니다.



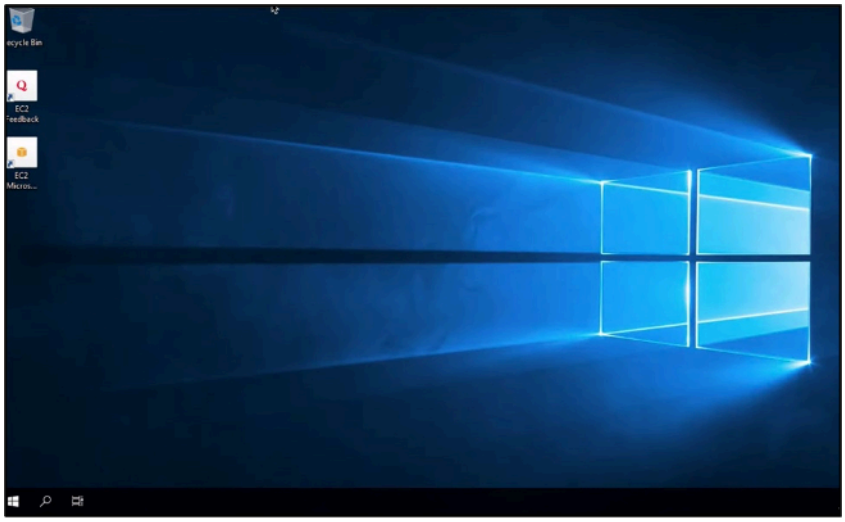
6. 연결을 선택합니다.
7. 프롬프트가 나타나면 이 절차에서 이전에 Lightsail 콘솔에서 복사한 기본 관리자 암호를 입력하거나 붙여 넣고 확인을 선택합니다.



8. 프롬프트가 나타나면 인증서 오류가 있어도 예를 선택하여 Windows 인스턴스에 연결합니다.



인스턴스에 연결되면 다음 예와 같은 화면이 보여야 합니다.



원격 데스크톱으로 macOS에서 Lightsail Windows 인스턴스에 연결

macOS 컴퓨터에서 Microsoft 원격 데스크톱을 사용하여 Windows 인스턴스에 연결할 수 있습니다. Microsoft 원격 데스크톱을 사용하려면 Lightsail Windows 인스턴스에 대한 관리자 사용자 이름과 암호를 사용해야 합니다. 이 암호는 생성할 때 인스턴스에 지정된 기본 암호입니다. 기본 암호를 변경한 경우에 이것은 사용자 자신의 암호입니다.

이 주제에서는 Lightsail 콘솔에서 기본 관리자 암호를 가져오고 Microsoft 원격 데스크톱을 구성하여 Windows 인스턴스에 연결하는 단계를 안내합니다. 브라우저를 사용해 Lightsail 콘솔에서 인스턴스에 연결할 수도 있습니다. 자세한 내용은 Microsoft 원격 데스크톱 클라이언트를 사용하여 Windows 인스턴스에 연결을 참조하세요.

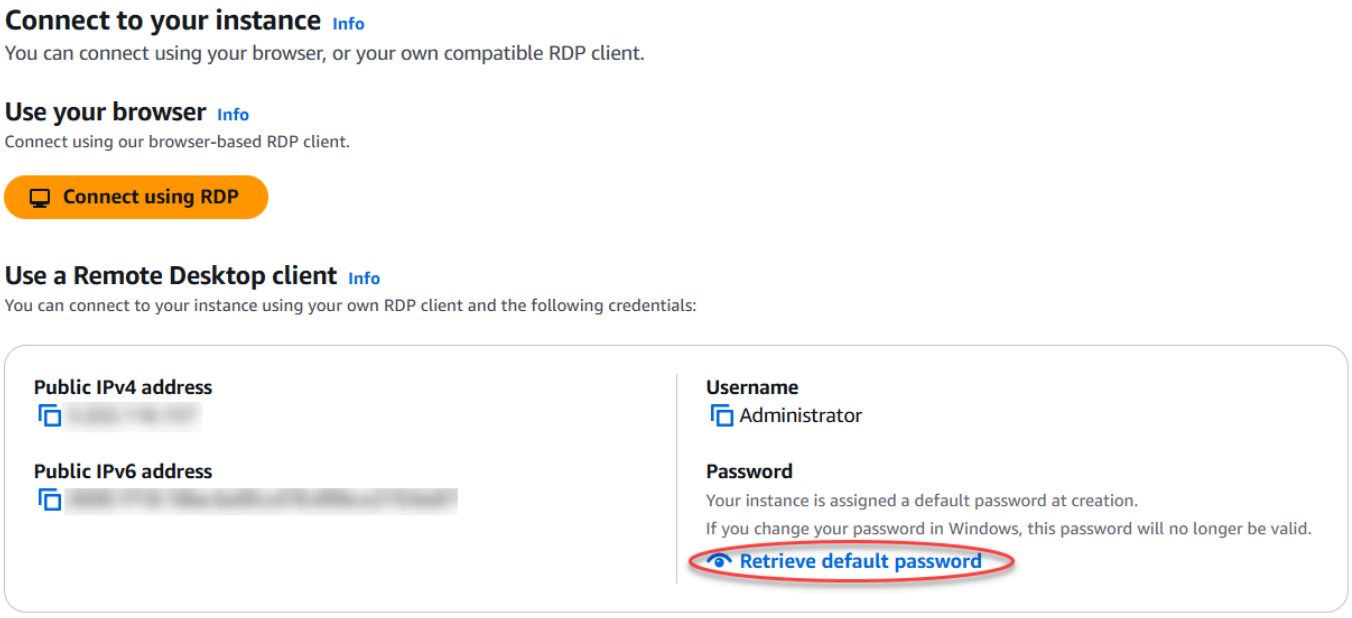
Windows 인스턴스에 대한 필수 연결 정보 가져오기

Microsoft 원격 데스크톱 클라이언트를 사용하여 Windows 인스턴스에 연결하려면 퍼블릭 IP 주소, 사용자 이름 및 관리자 암호가 필요합니다.

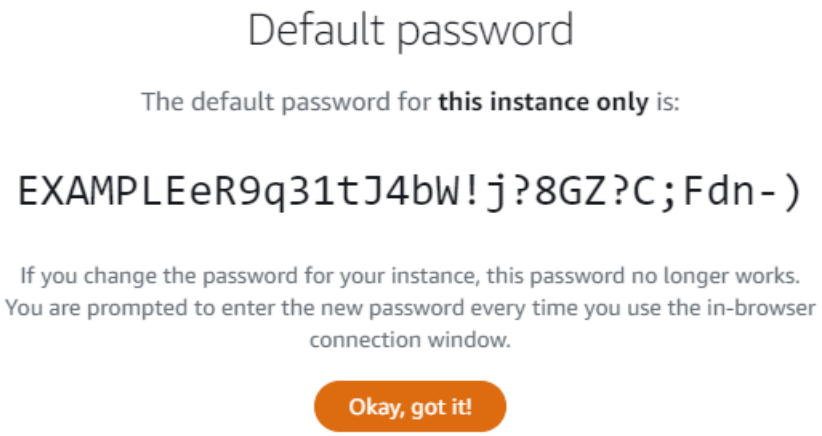
필요한 정보를 가져오려면 다음 절차를 완료합니다.

1. Lightsail 콘솔에 로그인합니다.
2. Lightsail 홈 페이지에서 인스턴스 섹션을 선택합니다.
3. 연결하려는 인스턴스의 퍼블릭 IP 주소를 기록해 둡니다.
4. 연결할 인스턴스의 이름을 선택합니다.
5. 연결 탭을 선택합니다.

6. 기본 암호 표시(Show default password)를 선택하여 인스턴스의 Windows 관리자 암호를 가져옵니다.



프롬프트에 Windows 인스턴스의 기본 관리자 암호가 표시됩니다.



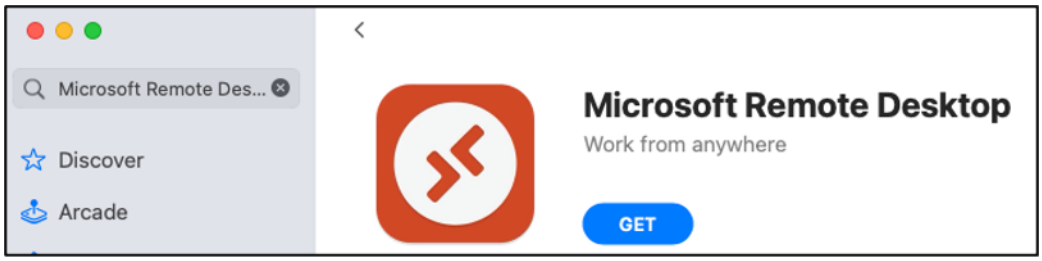
7. 관리자 암호를 복사합니다. 이후 가이드에서 Microsoft 원격 데스크톱 클라이언트를 사용하여 인스턴스에 로그인하는 데 사용합니다.

Microsoft 원격 데스크톱 구성 및 인스턴스에 연결

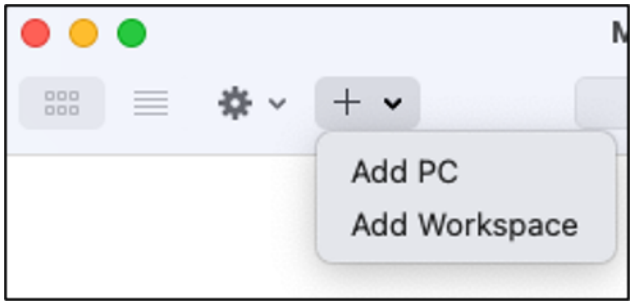
다음 절차를 완료하여 Mac에 Microsoft 원격 데스크톱 클라이언트를 설치하고 인스턴스에 연결하도록 구성합니다.

1. Mac에서 App Store를 열고 Microsoft 원격 데스크톱(Microsoft Remote Desktop)을 검색합니다.

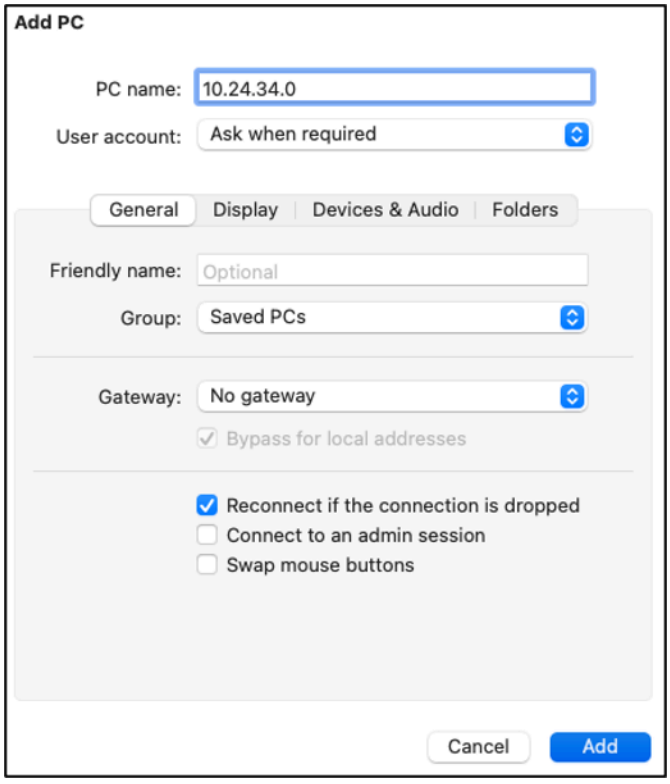
2. 검색 결과에서 Microsoft 원격 데스크톱(Microsoft Remote Desktop) 앱을 찾고 가져오기(GET)를 선택하여 애플리케이션을 설치합니다.



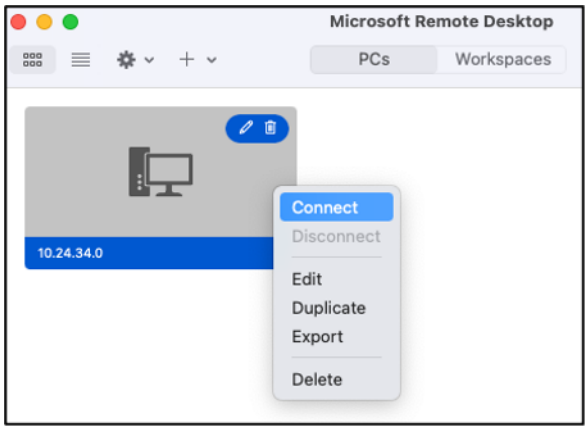
3. 설치가 완료된 후 Microsoft 원격 데스크톱(Microsoft Remote Desktop)을 엽니다.
4. 상단에서 더하기(+) 아이콘을 선택하고 PC 추가를 선택합니다.



5. PC 이름(PC name) 텍스트 상자에 인스턴스의 퍼블릭 IP 주소를 붙여 넣습니다.
6. 추가를 선택합니다.

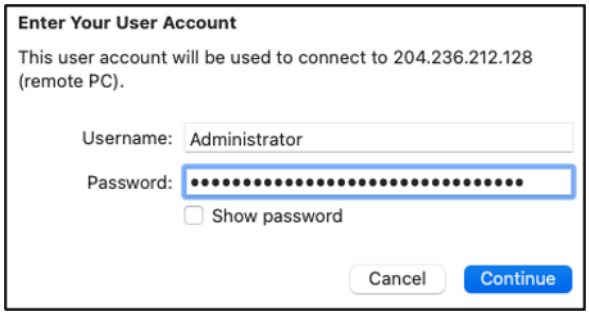


7. 인스턴스의 아이콘을 마우스 오른쪽 버튼으로 클릭하고 연결(Connect)을 선택합니다.

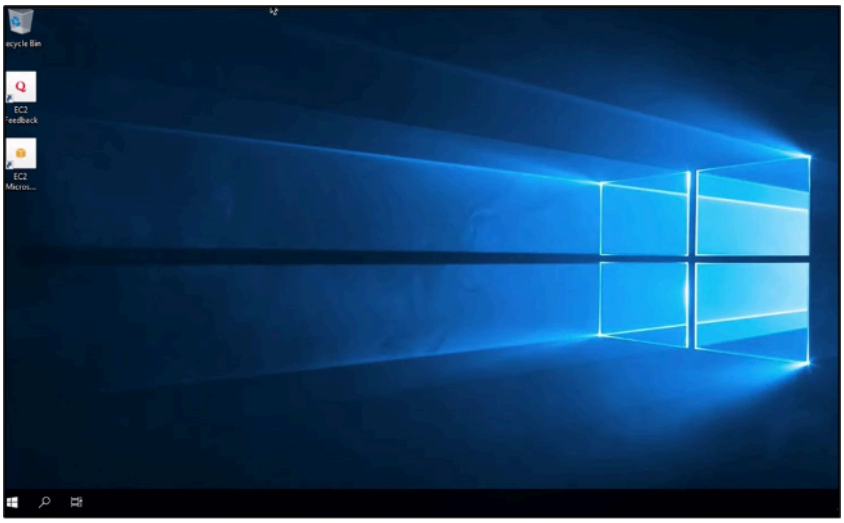


8. 사용자 이름(Username) 텍스트 상자에 Administrator를 입력하고 암호(Password) 텍스트 상자에 앞선 가이드에서 얻은 기본 관리자 암호를 입력합니다.

9. 계속을 선택하여 인스턴스에 연결합니다.



이제 Lightsail Windows 인스턴스와 연결됩니다.



Lightsail의 인스턴스 메타데이터 서비스(IMDS) 및 사용자 데이터 액세스

인스턴스 메타데이터는 실행 중인 인스턴스를 구성 또는 관리하는 데 사용될 수 있는 인스턴스 관련 데이터입니다. 인스턴스 메타데이터는 호스트 이름, 이벤트, 보안 그룹과 같은 범주로 분류됩니다. 인스턴스 메타데이터를 사용하여 인스턴스를 시작할 때 지정한 사용자 데이터에도 액세스할 수 있습니다. 예를 들어, 인스턴스를 구성하기 위한 파라미터를 지정하거나 단순 스크립트를 포함시킬 수 있습니다. 또한 인스턴스는 인스턴스가 시작되었을 때 생성되는 인스턴스 자격 증명 문서와 같은 동적 데이터를 포함할 수도 있습니다.

> **Important**
>
> 사용자는 인스턴스 자체 내에서 인스턴스 메타데이터 및 사용자 데이터에만 액세스할 수 있지만, 데이터는 인증 또는 암호화 방법으로 보호되지 않습니다. 인스턴스에 직접 액세스할 수 있는 모든 사람과 인스턴스에서 실행 중인 모든 소프트웨어는 메타데이터를 볼 수 있습니다. 따라서 암호 또는 수명이 긴 암호화 키와 같은 민감한 데이터를 사용자 데이터로 저장해서는 안 됩니다.

인스턴스 메타데이터 서비스 사용

다음 방법 중 하나를 사용하여 Lightsail의 실행 중인 인스턴스에서 인스턴스 메타데이터에 액세스할 수 있습니다.

- 인스턴스 메타데이터 서비스 버전 1(IMDSv1) – 요청/응답 방법
- 인스턴스 메타데이터 서비스 버전 2(IMDSv2) – 세션 지향 방법

> **Important**
>
> Lightsail의 모든 인스턴스 블루프린트가 IMDSv2를 지원하는 것은 아닙니다. IMDSv1을 사용하는 인스턴스 메타데이터 서비스에 대한 호출 수를 추적하려면 `MetadataNoToken` 인스턴스 지표를 사용하세요. 자세한 내용은 인스턴스 지표 보기를 참조하세요.

IMDS 사용에 대한 자세한 내용은 인스턴스 메타데이터 서비스(IMDS) 구성을 참조하세요.

추가 IMDS 문서

다음 IMDS 설명서는 Linux 인스턴스용 Amazon Elastic Compute Cloud 사용 설명서와 Windows 인스턴스용 Amazon Elastic Compute Cloud 사용 설명서에서 확인할 수 있습니다.

> **Note**
>
> Amazon EC2에서 인스턴스 블루프린트는 Amazon Machine Image(AMI)라고 합니다.

- Linux 인스턴스의 경우:
  - 인스턴스 메타데이터 옵션 구성
  - 인스턴스 메타데이터 검색
  - 인스턴스 사용자 데이터 작업
  - 동적 데이터 검색
  - 인스턴스 메타데이터 카테고리
  - 예시: AMI 시작 인덱스 값
  - 인스턴스 자격 증명 문서
- Windows 인스턴스의 경우:
  - 인스턴스 메타데이터 옵션 구성
  - 인스턴스 메타데이터 검색
  - 인스턴스 사용자 데이터 작업
  - 동적 데이터 검색
  - 인스턴스 메타데이터 카테고리
  - 예시: AMI 시작 인덱스 값
  - 인스턴스 자격 증명 문서

Lightsail의 인스턴스 메타데이터 서비스(IMDS) 액세스 및 구성

다음 방법 중 하나를 사용하여 실행 중인 인스턴스에서 인스턴스 메타데이터에 액세스할 수 있습니다.

- 인스턴스 메타데이터 서비스 버전 1(IMDSv1) – 요청/응답 방법
- 인스턴스 메타데이터 서비스 버전 2(IMDSv2) – 세션 지향 방법

> **Important**
>
> Lightsail의 모든 인스턴스 블루프린트가 IMDSv2를 지원하는 것은 아닙니다. IMDSv1을 사용하는 인스턴스 메타데이터 서비스에 대한 호출 수를 추적하려면 `MetadataNoToken` 인스턴스 지표를 사용하세요. 자세한 내용은 인스턴스 지표 보기를 참조하세요.

기본적으로 IMDSv1 또는 IMDSv2를 사용하거나 둘 다 사용할 수 있습니다. 인스턴스 메타데이터 서비스는 IMDSv2에 고유한 PUT 또는 GET 헤더가 주어진 요청에 있는지 여부에 따라 IMDSv1과 IMDSv2 요청을 구별합니다. 자세한 내용은 EC2 인스턴스 메타데이터 서비스의 향상된 기능을 통해 개방형 방화벽, 역방향 프록시 및 SSRF 취약성에 대한 심층적인 방어 기능 추가를 참조하세요.

로컬 코드 또는 사용자가 IMDSv2를 사용해야 하도록 각 인스턴스에서 인스턴스 메타데이터 서비스를 구성할 수 있습니다. IMDSv2를 사용해야 하도록 지정하면 IMDSv1는 더 이상 작동하지 않습니다. 자세한 내용은 Linux 인스턴스용 Amazon Elastic Compute Cloud 사용 설명서의 인스턴스 메타데이터 옵션 구성을 참조하세요.

인스턴스 메타데이터를 검색하려면 Linux 인스턴스용 Amazon Elastic Compute Cloud 사용 설명서의 인스턴스 메타데이터 검색을 참조하세요.

> **Note**
>
> 이 섹션의 예에서는 인스턴스 메타데이터 서비스의 IPv4 주소(`169.254.169.254`)를 사용합니다. IPv6 주소를 통해 인스턴스의 인스턴스 메타데이터를 검색하는 경우 대신 IPv6 주소(`fd00:ec2::254`)를 활성화하고 사용해야 합니다. 인스턴스 메타데이터 서비스의 IPv6 주소는 IMDSv2 명령과 호환됩니다.

인스턴스 메타데이터 서비스 버전 2 작동 방식

IMDSv2는 세션 지향 요청을 사용합니다. 세션 지향 요청을 사용하여 세션 기간을 정의하는 세션 토큰을 생성합니다. 세션 기간은 최소 1초에서 최대 6시간일 수 있습니다. 지정된 기간 중에는 후속 요청에 동일한 세션 토큰을 사용할 수 있습니다. 지정된 기간이 만료된 후에는 향후 요청에 사용할 새로운 세션 토큰을 생성할 수 있습니다.

> **Important**
>
> Amazon Linux 2023 및 Ubuntu 24 블루프린트에서 시작된 Lightsail 인스턴스에는 기본으로 IMDSv2가 구성됩니다.

다음 예제에서는 Linux, PowerShell 쉘 스크립트 및 IMDSv2를 사용하여 최상위 인스턴스 메타데이터 항목을 검색합니다. 이 예제는 다음을 수행합니다.

- PUT 요청을 사용하여 6시간(21,600초) 동안 지속되는 세션 토큰 생성
- TOKEN(Linux의 경우) 또는 token(Windows의 경우)이라는 변수에 세션 토큰 헤더 저장
- 토큰을 사용하여 최상위 메타데이터 항목 요청

다음 명령을 실행하여 시작합니다.

- Linux의 경우:
  - 먼저 다음 명령을 사용하여 토큰을 생성합니다.

    ```
    [ec2-user ~]$ TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-aws-ec2-metadata-token-ttl-seconds: 21600"`
    ```

  - 그런 다음 해당 토큰을 사용하여 다음 명령으로 최상위 메타데이터 항목을 생성합니다.

    ```
    [ec2-user ~]$ curl -H "X-aws-ec2-metadata-token: $TOKEN" -v http://169.254.169.254/latest/meta-data/
    ```

- Windows의 경우:
  - 먼저 다음 명령을 사용하여 토큰을 생성합니다.

    ```
    PS C:\> [string]$token = Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token-ttl-seconds" = "21600"} -Method PUT -Uri http://169.254.169.254/latest/api/token
    ```

  - 그런 다음 해당 토큰을 사용하여 다음 명령으로 최상위 메타데이터 항목을 생성합니다.

    ```
    PS C:\> Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token" = $token} -Method
     GET -Uri http://169.254.169.254/latest/meta-data/
    ```

토큰을 생성한 후에는 만료될 때까지 토큰을 재사용할 수 있습니다. 다음 예제에서 각 명령은 인스턴스를 시작하는 데 사용되는 블루프린트(Amazon Machine Image(AMI))의 ID를 가져옵니다. 이전 예제의 토큰이 재사용됩니다. 토큰은 $TOKEN(Linux의 경우) 또는 $token(Windows의 경우)에 저장됩니다.

- Linux의 경우:

```
[ec2-user ~]$ curl -H "X-aws-ec2-metadata-token: $TOKEN" -v http://169.254.169.254/
latest/meta-data/ami-id
```

- Windows의 경우:

```
PS C:\> Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token" = $token} `
-Method GET -uri http://169.254.169.254/latest/meta-data/ami-id
```

IMDSv2를 사용하여 인스턴스 메타데이터를 요청하는 경우 요청에는 다음이 포함되어야 합니다.

- **PUT** 요청 - PUT 요청을 사용하여 인스턴스 메타데이터 서비스의 세션을 초기화합니다. PUT 요청은 후속 GET 요청에 포함되어야 하는 토큰을 인스턴스 메타데이터 서비스에 반환합니다. 토큰은 IMDSv2 사용 시 메타데이터에 액세스하는 데 필요합니다.
- 토큰 - 인스턴스 메타데이터 서비스에 대한 모든 GET 요청에 토큰을 포함합니다. 토큰 사용이 `required`로 설정되면 유효한 토큰이 없거나 만료된 토큰이 있는 요청은 `401 - Unauthorized` HTTP 오류 코드를 수신합니다. 토큰 사용 요구 사항 변경에 대한 자세한 내용은 AWS CLI 명령 레퍼런스의 update-instance-metadata-options를 참조하세요.
  - 토큰은 인스턴스에 특정한 키입니다. 토큰은 다른 인스턴스에서 유효하지 않으며 해당 토큰이 생성된 인스턴스 외부에서 사용하려고 시도하면 거부됩니다.
  - PUT 요청에는 토큰의 TTL(Time to Live)을 초 단위로 지정하는 헤더가 포함되어야 합니다. TTL은 최대 6시간(21,600초)까지 지정할 수 있습니다. 토큰은 논리 세션을 나타냅니다. TTL은 토큰이 유효한 시간 길이를 지정하며 따라서 세션 기간을 지정합니다.
  - 토큰이 만료된 후 인스턴스 메타데이터에 계속 액세스하려면 다른 PUT 요청을 사용하여 새 세션을 생성해야 합니다.
  - 토큰을 재사용하거나 모든 요청에 새 토큰을 생성하도록 선택할 수 있습니다. 요청 수가 적은 경우 인스턴스 메타데이터 서비스에 액세스해야 할 때마다 토큰을 생성하고 즉시 사용하는 것이 더 간편할 수 있습니다. 하지만 효율성을 향상하려면 인스턴스 메타데이터를 요청해야 할 때마다 PUT 요청을 작성하는 대신 토큰에 더 긴 기간을 지정하고 토큰을 재사용할 수 있습니다. 동시 토큰 수에는 실질적인 제한이 없으며 각각은 자체 세션을 나타냅니다. 그러나 IMDSv2에는 표준 인스턴스

메타데이터 서비스 연결 및 스로틀링 제한이 여전히 적용됩니다. 자세한 내용은 Linux 인스턴스용 Amazon Elastic Compute Cloud 사용 설명서의 쿼리 스로틀링을 참조하세요.

IMDSv2 인스턴스 메타데이터 요청에서는 HTTP GET 및 HEAD 메서드가 허용됩니다. PUT 요청은 X-Forwarded-For 헤더가 포함된 경우 거부됩니다.

기본적으로 PUT 요청에 대한 응답에는 IP 프로토콜 수준에서 1의 응답 홉 제한(TTL(Time to Live))이 있습니다. 더 큰 홉 제한이 필요한 경우 update-instance-metadata-options 명령을 사용하여 홉 제한을 조정할 수 있습니다. 예를 들어, 인스턴스에서 실행 중인 컨테이너 서비스가 있는 경우 이전 버전과의 호환성을 위해 더 큰 홉 제한이 필요할 수 있습니다. 자세한 내용은 AWS CLI 명령 레퍼런스의 update-instance-metadata-options를 참조하세요.

인스턴스 메타데이터 서비스 버전 2 사용으로 전환

Instance Metadata Service 버전 2(IMDSv2) 사용은 선택 사항입니다. Instance Metadata Service 버전 1(IMDSv1)은 무기한 지원됩니다. IMDSv2 사용으로 마이그레이션하도록 선택하는 경우 다음과 같은 도구와 전환 경로를 사용하는 것이 좋습니다.

IMDSv2로 전환하는 데 도움이 되는 도구

소프트웨어가 IMDSv1을 사용하는 경우 다음 도구를 사용하면 IMDSv2를 사용하도록 소프트웨어를 재구성하는 데 도움이 됩니다.

- AWS 소프트웨어: 최신 버전의 AWS SDKs 및 AWS CLI 지원 IMDSv2. IMDSv2를 사용하려면 인스턴스에 최신 버전의 AWS SDKs 및가 있어야 합니다 AWS CLI. 업데이트에 대한 자세한 내용은 AWS Command Line Interface 사용 설명서의 설치, 업데이트 및 제거 AWS CLI를 AWS CLI참조하세요. 모든 Amazon Linux 2 소프트웨어 패키지에서는 IMDSv2가 지원됩니다.
- 인스턴스 지표: IMDSv2는 토큰 지원 세션을 사용하지만 IMDSv1은 사용하지 않습니다. MetadataNoToken 인스턴스 지표는 IMDSv1을 사용하는 인스턴스 메타데이터 서비스에 대한 호출 수를 추적합니다. 이 지표를 0까지 추적하면 모든 소프트웨어가 IMDSv2를 사용하도록 업그레이드되었는지 여부와 업그레이드된 시간을 확인할 수 있습니다. 자세한 내용은 Amazon Lightsail의 인스턴스 지표 보기를 참조하세요.
- Lightsail API 작업 및 AWS CLI 명령 업데이트: 기존 인스턴스의 경우 update-instance-metadata-options AWS CLI 명령(또는 UpdateInstanceMetadataOptions API 작업)을 사용하여 IMDSv2를 사용해야 할 수 있습니다. 다음 명령은 예제입니다. *InstanceName*을 인스턴스 이름으로 바꾸고 *RegionName*을 인스턴스가 있는 로 바꿔야 AWS 리전 합니다.

```
aws lightsail update-instance-metadata-options --region RegionName --instance-
name InstanceName --http-tokens required
```

IMDSv2에 액세스해야 하도록 설정하는 권장 경로

위의 도구를 사용하여 IMDSv2로 전환하기 위해 다음 경로를 따르는 것이 좋습니다.

1단계: 시작 시

인스턴스에서 역할 자격 증명을 사용하는 AWS SDKs AWS CLI, 및 소프트웨어를 IMDSv2-compatible 버전으로 업데이트합니다. 업데이트에 대한 자세한 내용은 AWS Command Line Interface 사용 설명서의의 최신 버전으로 업그레이드를 AWS CLI AWS CLI참조하세요.

그런 다음 IMDSv2 요청을 사용하여 인스턴스 메타데이터에 직접 액세스하는 소프트웨어(즉, AWS SDK를 사용하지 않는 소프트웨어)를 변경합니다.

2단계: 전환 도중

인스턴스 지표 `MetadataNoToken`을 사용하여 전환 진행률을 추적합니다. 이 지표는 인스턴스에서 IMDSv1을 사용하는 인스턴스 메타데이터 서비스에 대한 호출 수를 표시합니다. 자세한 내용은 인스턴스 지표 보기를 참조하세요.

3단계: 모든 인스턴스에서 모든 준비가 완료될 때

인스턴스 지표 `MetadataNoToken`이 IMDSv1 사용량 0을 기록하면 모든 인스턴스에서 준비가 모두 완료됩니다. 이 단계에서 update-instance-metadata-options 명령을 통해 IMDSv2 사용을 요구할 수 있습니다. 실행 중인 인스턴스에서 이렇게 변경할 수 있으며, 인스턴스를 다시 시작할 필요는 없습니다.

기존 인스턴스에 대한 인스턴스 메타데이터 옵션 업데이트는 Lightsail API 또는 AWS CLI를 통해서만 사용할 수 있습니다. 현재 Lightsail 콘솔에서 사용할 수 없습니다. 자세한 내용은 update-instance-metadata-options를 참조하세요.

추가 IMDS 문서

다음 IMDS 설명서는 Linux 인스턴스용 Amazon Elastic Compute Cloud 사용 설명서와 Windows 인스턴스용 Amazon Elastic Compute Cloud 사용 설명서에서 확인할 수 있습니다.

> **Note**
>
> Amazon EC2에서 인스턴스 블루프린트는 Amazon Machine Image(AMI)라고 합니다.

- Linux 인스턴스의 경우:
  - 인스턴스 메타데이터 옵션 구성
  - 인스턴스 메타데이터 검색
  - 인스턴스 사용자 데이터 작업
  - 동적 데이터 검색
  - 인스턴스 메타데이터 카테고리
  - 예시: AMI 시작 인덱스 값
  - 인스턴스 자격 증명 문서
- Windows 인스턴스의 경우:
  - 인스턴스 메타데이터 옵션 구성
  - 인스턴스 메타데이터 검색
  - 인스턴스 사용자 데이터 작업
  - 동적 데이터 검색
  - 인스턴스 메타데이터 카테고리
  - 예시: AMI 시작 인덱스 값
  - 인스턴스 자격 증명 문서

Lightsail 블록 스토리지 디스크로 스토리지 및 성능 확장

시스템 디스크는 지연 시간이 짧고 일관된 성능으로 워크로드를 실행해 줍니다. Lightsail 디스크를 사용하면 몇 분 이내에 사용량을 늘리거나 줄이면서 프로비저닝한 용량에 대해서만 적은 요금을 지불할 수 있습니다.

Linux/Unix 기반 인스턴스 또는 Windows Server 기반 인스턴스에서 최대 80GB의 시스템 디스크를 선택할 수 있습니다. Lightsail에서 Linux 기반 인스턴스 시작하기 또는 &lightsail;에서 Windows Server 기반 인스턴스 시작하기를 참조하십시오.

블록 스토리지 디스크를 추가로 생성하여 가상 프라이빗 서버에 더 많은 스토리지를 연결할 수도 있습니다. 블록 스토리지 디스크를 생성하여 Linux 기반 인스턴스에 연결 또는 블록 스토리지 디스크를 생성하여 Windows Server 인스턴스에 연결을 참조하세요.

블록 스토리지 디스크

블록 스토리지란 데이터를 "블록" 단위로 관리하는 스토리지 아키텍처를 말합니다. 각각의 스토리지 블록(Lightsail의 "디스크")은 개별 하드 디스크처럼 작동하며, 이를 서버에 연결할 수 있습니다. 일반적으로 추가 블록 스토리지는 특정한 데이터를 핵심 서비스와 분리해야 하는 애플리케이션 또는 소프트웨어에 사용할 수 있으며, 인스턴스와 부팅 스토리지 디스크에 결함 또는 다른 문제가 있을 때 애플리케이션 데이터를 보호하기 위해서도 사용합니다.

Lightsail은 블록 스토리지용으로 SSD(Solid State Drive)를 제공합니다. 이 종류의 블록 스토리지는 합리적인 가격과 우수한 성능이 균형을 이루고 있습니다. 용도는 Lightsail에서 실행되는 워크로드 중 대부분을 지원하는 것입니다. Lightsail의 추가 블록 스토리지 디스크는 저장된 데이터에 자주 액세스해야 하는 애플리케이션 또는 소프트웨어를 실행하는 데 필요한 일관된 성능과 짧은 지연 시간을 제공합니다.

> **Note**
>
> 디스크당 처리량이 많아야 하거나 일관된 IOPS 성능이 필요한 애플리케이션을 사용하는 고객 또는 MongoDB, Cassandra 등 대형 데이터베이스를 실행하는 고객의 경우, Lightsail 대신 GP2 또는 프로비저닝된 IOPS SSD 스토리지가 탑재된 Amazon EC2를 사용하는 것이 좋습니다.

Amazon EC2 사용 설명서의 Amazon EBS 볼륨에 대해 알아볼 수 있습니다.

디스크 할당량

- 리전당 20,000GB
- 디스크당 최대 16TB 또는 디스크당 최소 8GB
- 각 인스턴스에 최대 15개의 연결된 디스크와 1개의 부트 볼륨 디스크가 있을 수 있습니다.

Lightsail 블록 스토리지 디스크를 생성하고 Linux 인스턴스에 연결

Amazon Lightsail 인스턴스용으로 블록 스토리지 디스크를 추가로 생성하여 연결할 수 있습니다. 추가로 생성한 디스크를 Linux/Unix 기반의 Lightsail 인스턴스에 연결한 다음 그 디스크를 포맷하고 탑재해야 합니다.

이 주제에서는 새 디스크를 생성하고 Lightsail을 사용하여 연결하는 방법을 보여 줍니다. 또 연결된 디스크를 포맷하고 탑재하기 위해 SSH를 사용하여 Linux/Unix 기반 인스턴스에 연결하는 방법도 설명합니다.

Windows Server 기반의 인스턴스가 있는 경우 블록 스토리지 디스크 생성 Windows Server 인스턴스에 연결을 참조하세요.

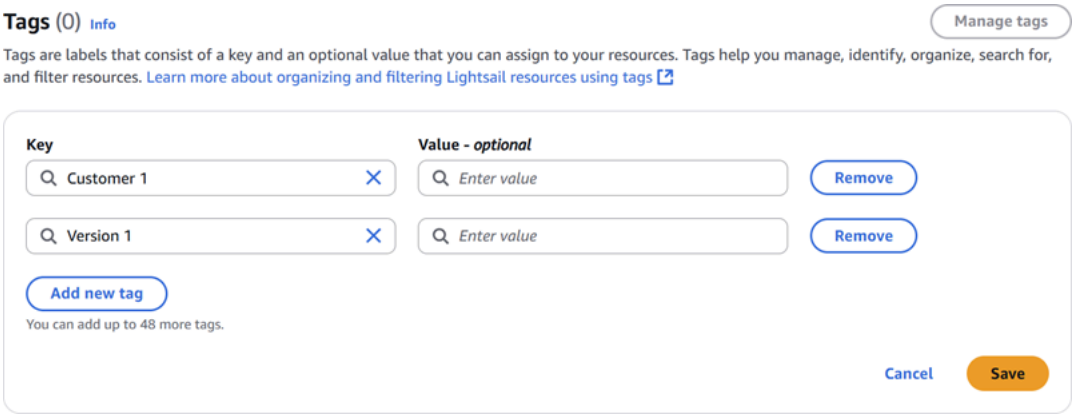
1단계: 새 디스크 생성 및 인스턴스에 연결

1. 왼쪽 탐색 창에서 스토리지를 선택합니다.
2. 디스크 생성을 선택합니다.
3. Lightsail 인스턴스가 위치한 AWS 리전 및 가용 영역을 선택합니다.
4. 크기를 선택합니다.
5. 디스크의 이름을 입력합니다.

   리소스 이름:

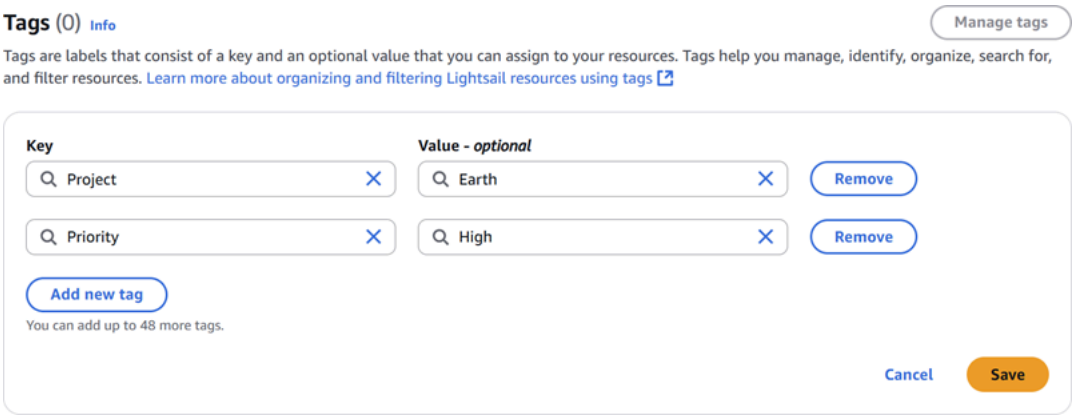
   - Lightsail 계정 AWS 리전 의 각 내에서 고유해야 합니다.
   - 2~255자의 문자로 구성되어야 합니다.
   - 영숫자 문자 또는 숫자로 시작하고 끝나야 합니다.
   - 영숫자 문자, 숫자, 마침표, 대시, 밑줄이 포함될 수 있습니다.
6. 다음 옵션 중 하나를 선택하여 디스크에 태그를 추가합니다.

- 키 전용 태그를 추가하거나 태그를 관리(이미 태그를 추가한 경우)합니다. 새 태그를 태그 키 텍스트 상자에 입력하고 Enter를 누릅니다. 태그를 추가하려면 태그 입력이 완료될 때 저장을 선택하고, 태그를 추가하지 않으려면 취소를 선택합니다.



- 키-값 태그를 생성한 다음 키를 키 텍스트 상자에 입력하고, 값을 값 텍스트 상자에 입력합니다. 태그 입력이 완료되면 저장을 선택하고, 태그를 추가하지 않으려면 취소를 선택합니다.

  키-값 태그는 저장 전에 한 번에 하나씩만 추가할 수 있습니다. 둘 이상의 키-값 태그를 추가하려면 이전 단계를 반복하세요.



> **Note**
>
> 키 전용 태그 및 키-값 태그에 대한 자세한 내용은 태그를 참조하세요.

7. 디스크 생성을 선택합니다.

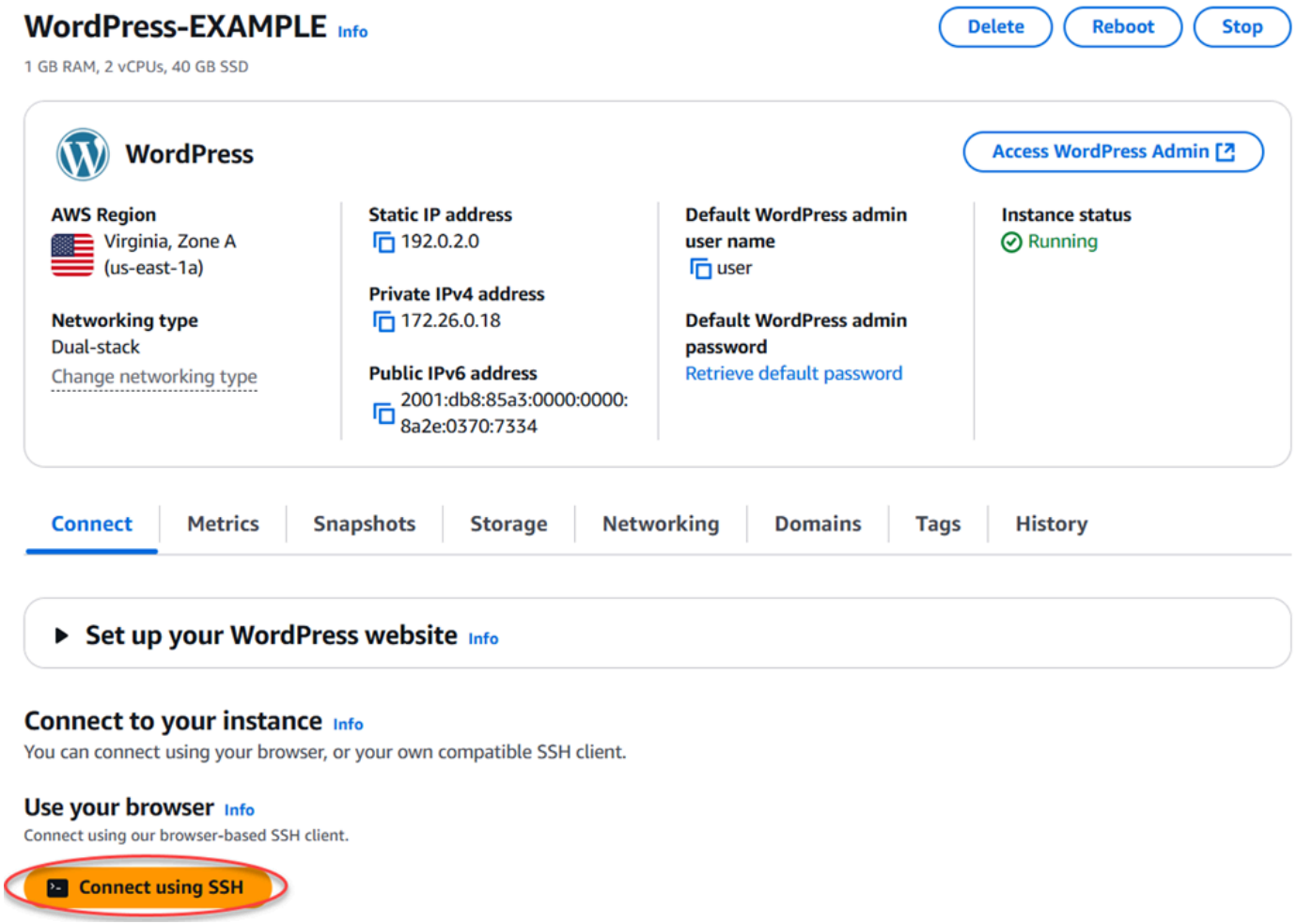
   몇 초 뒤 디스크가 생성되고, 새 디스크 관리 페이지가 열립니다.

8. 목록에서 원하는 인스턴스를 선택하고 연결을 선택하여 새 디스크를 인스턴스에 연결합니다.

2단계: 인스턴스에 연결하여 디스크 포맷 및 탑재

1. 디스크를 생성하여 연결했으면 Lightsail의 인스턴스 관리 페이지로 돌아갑니다.

   기본적으로 연결 탭이 표시됩니다.



2. SSH를 사용하여 연결을 선택하여 인스턴스에 연결합니다.
3. 터미널 창에 다음 명령을 입력합니다.

```
lsblk
```

   `lsblk` 명령의 출력에서는 디스크 경로 맨 앞의 `/dev/` 접두사가 생략됩니다.

> **Note**
>
> 2023년 6월 29일 Lightsail 인스턴스의 기본 하드웨어를 업데이트했습니다. 다음 예제에서는 이전 세대 인스턴스의 디바이스 이름이 /dev/xvda로 표시됩니다. 이 날짜 이후에 생성된 인스턴스의 디바이스 이름은 /dev/nvme0n1로 표시됩니다.

Current generation instances

다음 예제 출력에서 루트 볼륨(nvme0n1)에는 2개의 파티션(nvme0n1p1 및 nvme0n1p128)이 있는 반면 추가 볼륨(nvme1n1)에는 파티션이 없습니다.

```
[ec2-user ~]$ sudo lsblk
NAME          MAJ:MIN RM SIZE RO TYPE MOUNTPOINT
nvme1n1       259:0    0  30G  0 disk /data
nvme0n1       259:1    0  16G  0 disk
##nvme0n1p1   259:2    0   8G  0 part /
##nvme0n1p128 259:3    0   1M  0 part
```

Previous generation instances

다음 예제 출력에서 루트 볼륨(xvda)에는 1개의 파티션(xvda1)이 있는 반면 추가 볼륨(xvdf)에는 파티션이 없습니다.

```
[ec2-user ~]$ sudo lsblk
NAME    MAJ:MIN RM SIZE RO TYPE MOUNTPOINT
xvda    202:0    0  16G  0 disk
##xvda1 202:1    0   8G  0 part /
xvdf    202:80   0  24G  0 disk
```

4. 해당 디스크에 파일 시스템을 생성할지 여부를 결정합니다. 새 디스크는 원시 블록 디바이스이므로, 먼저 파일 시스템을 생성해야 이 디스크를 탑재하고 사용할 수 있습니다. 스냅샷에서 복원한 디스크는 이미 파일 시스템이 있을 가능성이 높습니다. 기존의 파일 시스템 위에 새 파일 시스템을 생성하면 그 작업으로 인해 데이터를 덮어쓰게 됩니다.

   다음을 사용하여 디스크에 파일 시스템이 있는지 확인합니다. 디스크에 파일 시스템이 없는 경우 2.5단계로 진행합니다. 디스크에 파일 시스템이 있는 경우 2.6단계로 건너뜁니다.

Current generation instances

```
sudo file -s /dev/nvme1n1
```

새로운 디스크에 대해 다음과 같은 결과가 출력됩니다.

```
/dev/nvme1n1: data
```

다음과 같이 출력되면 디스크에 이미 파일 시스템이 있는 것입니다.

```
/dev/nvme1n1: SGI XFS filesystem data (blksz 4096, inosz 512, v2 dirs)
```

Previous generation instances

```
sudo file -s /dev/xvdf
```

새로운 디스크에 대해 다음과 같은 결과가 출력됩니다.

```
/dev/xvdf: data
```

다음과 같이 출력되면 디스크에 이미 파일 시스템이 있는 것입니다.

```
/dev/xvda1: Linux rev 1.0 ext4 filesystem data, UUID=1701d228-e1bd-4094-
a14c-12345EXAMPLE (needs journal recovery) (extents) (large files) (huge files)
```

5. 다음 명령을 사용하여 해당 디스크에 새 파일 시스템을 생성하세요. 디바이스 이름(예: /dev/nvme1n1)을 *device_name*으로 바꿉니다. 애플리케이션의 요구 사항 또는 운영 체제의 제한에 따라 ext3 또는 ext4 등 다양한 파일 시스템 유형을 선택할 수 있습니다.

> **Important**
>
> 이 단계에서는 비어 있는 디스크를 탑재한다고 가정합니다. 이미 데이터가 있는 디스크를 탑재하는 경우(예: 스냅샷에서 복원한 디스크), 디스크를 탑재하기 전에 mkfs 명령을 사용하지 마세요. 그 대신 2.6단계로 이동하여 탑재 지점을 만듭니다. 아니면 디스크를 포맷하고 기존 데이터를 삭제합니다.

Current generation instances

```
sudo mkfs -t xfs device_name
```

그러면 다음과 같은 결과가 표시됩니다.

```
meta-data=/dev/nvme1n1           isize=512    agcount=16, agsize=1048576 blks
         =                       sectsz=512   attr=2, projid32bit=1
         =                       crc=1        finobt=1, sparse=1, rmapbt=0
         =                       reflink=1    bigtime=1 inobtcount=1
data     =                       bsize=4096   blocks=16777216, imaxpct=25
         =                       sunit=1      swidth=1 blks
naming   =version 2              bsize=4096   ascii-ci=0, ftype=1
log      =internal log           bsize=4096   blocks=16384, version=2
         =                       sectsz=512   sunit=1 blks, lazy-count=1
realtime =none                   extsz=4096   blocks=0, rtextents=0
```

Previous generation instances

```
sudo mkfs -t ext4 device_name
```

그러면 다음과 같은 다음 결과가 표시됩니다.

```
mke2fs 1.42.9 (4-Feb-2014)
Filesystem label=
OS type: Linux
Block size=4096 (log=2)
Fragment size=4096 (log=2)
Stride=0 blocks, Stripe width=0 blocks
4194304 inodes, 16777216 blocks
838860 blocks (5.00%) reserved for the super user
First data block=0
Maximum filesystem blocks=4294967296
512 block groups
32768 blocks per group, 32768 fragments per group
8192 inodes per group
Superblock backups stored on blocks:
32768, 98304, 163840, 229376, 294912, 819200, 884736, 1605632, 2654208,
4096000, 7962624, 11239424
```

```
Allocating group tables: done
Writing inode tables: done
Creating journal (32768 blocks): done
Writing superblocks and filesystem accounting information: done
```

6. 다음 명령으로 해당 디스크의 탑재 지점 디렉터리를 생성합니다. 탑재 지점이란 파일 시스템 트리에서 디스크가 차지하는 위치이자, 디스크를 탑재한 후 파일을 읽어 오거나 쓰는 위치입니다. *mount_point*의 위치를 /data와 같이 사용하지 않는 공간으로 변경합니다.

```
sudo mkdir mount_point
```

7. 다음 명령을 입력해서 현재 디스크에 파일 시스템이 있는지 확인할 수 있습니다.

   Current generation instances

   ```
   sudo file -s /dev/nvme1n1
   ```

   /dev/nvme1n1: data 대신, 다음과 비슷한 결과가 출력됩니다.

   ```
   /dev/nvme1n1: SGI XFS filesystem data (blksz 4096, inosz 512, v2 dirs)
   ```

   Previous generation instances

   ```
   sudo file -s /dev/xvdf
   ```

   /dev/xvdf: data 대신, 다음과 비슷한 결과가 출력됩니다.

   ```
   /dev/xvdf: Linux rev 1.0 ext4 filesystem data, UUID=0ee83fdf-e370-442e-
   ae38-12345EXAMPLE (extents) (large files) (huge files)
   ```

8. 마지막으로 다음 명령을 입력하여 디스크를 탑재합니다.

```
sudo mount device_name mount_point
```

새 디스크 탑재 지점에 대한 파일 권한을 검토하여 사용자 및 애플리케이션이 그 디스크에 쓸 수 있는지 확인합니다. 파일 권한에 대한 자세한 내용은 Amazon EC2 사용 설명서의 Amazon EBS 볼륨을 사용할 수 있도록 만들기를 참조하세요.

3단계: 인스턴스를 재부팅할 때마다 디스크 탑재

Lightsail 인스턴스를 재부팅할 때마다 이 디스크를 탑재해야 하는 경우도 있습니다. 그렇지 않다면 이 단계는 건너뛰어도 됩니다.

1. 시스템을 재부팅할 때마다 이 디스크를 탑재하려면 해당 디바이스 항목을 `/etc/fstab` 파일에 추가합니다.

   수정 도중 실수로 이 파일이 손상되거나 삭제되는 경우에 대비하여 `/etc/fstab` 파일의 백업을 만들어 둡니다.

   ```
   sudo cp /etc/fstab /etc/fstab.orig
   ```

2. vim과 같은 텍스트 편집기를 사용하여 `/etc/fstab` 파일을 엽니다.

   파일을 열기 전에 `sudo`를 입력해야만 변경 사항을 저장할 수 있습니다.

3. 다음 형식으로 디스크 파일의 마지막에 새 줄을 추가합니다.

   ```
   device_name  mount_point  file_system_type  fs_mntops  fs_freq  fs_passno
   ```

   새로 만든 줄은 예를 들면 다음과 같은 모습으로 보입니다.

   Current generation instances

   ```
   /dev/nvme1n1 /data xfs defaults,nofail 0 2
   ```

   Previous generation instances

   ```
   /dev/xvdf /data ext4 defaults,nofail 0 2
   ```

4. 파일을 저장하고 텍스트 편집기를 종료합니다.

Lightsail 블록 스토리지 디스크를 생성하고 Windows Server 인스턴스에 연결

추가 스토리지 공간이 필요한 경우 Amazon Lightsail에서 Windows Server 인스턴스에서 블록 스토리지 디스크를 생성하고 연결할 수 있습니다. 블록 스토리지 디스크에 대한 자세한 내용은 블록 스토리지 디스크를 참조하세요.

이 안내서는 Lightsail 콘솔을 사용하여 새 블록 스토리지 디스크를 생성하고 Windows Server 인스턴스에 연결하는 방법을 보여줍니다. 또한 RDP를 사용하여 Windows Server 인스턴스에 연결하여 디스크를 온라인 상태로 만들고 초기화할 수 있는 방법에 대해서도 설명합니다.

> **Note**
>
> Linux/Unix 기반의 인스턴스가 있는 경우 디스크를 생성하여 Linux 또는 Unix 인스턴스에 연결을 참조하세요.

1단계: 새 블록 스토리지 디스크 생성 및 인스턴스에 연결

Amazon Lightsail 콘솔을 사용하여 새 블록 스토리지 디스크 생성 및 인스턴스에 연결

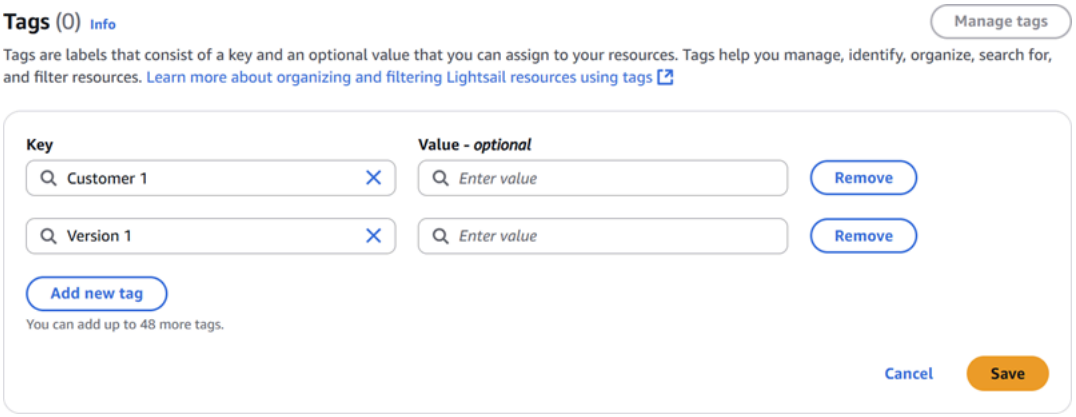
새 블록 스토리지 디스크 생성 및 인스턴스에 연결하려면

1. Lightsail 콘솔에 로그인합니다.
2. 스토리지 탭을 선택하고 디스크 생성을 선택합니다.
3. Lightsail 인스턴스가 위치한 AWS 리전 및 가용 영역을 선택합니다.
4. 디스크 크기를 선택합니다.
5. 스토리지 디스크의 이름을 입력합니다.

   리소스 이름:

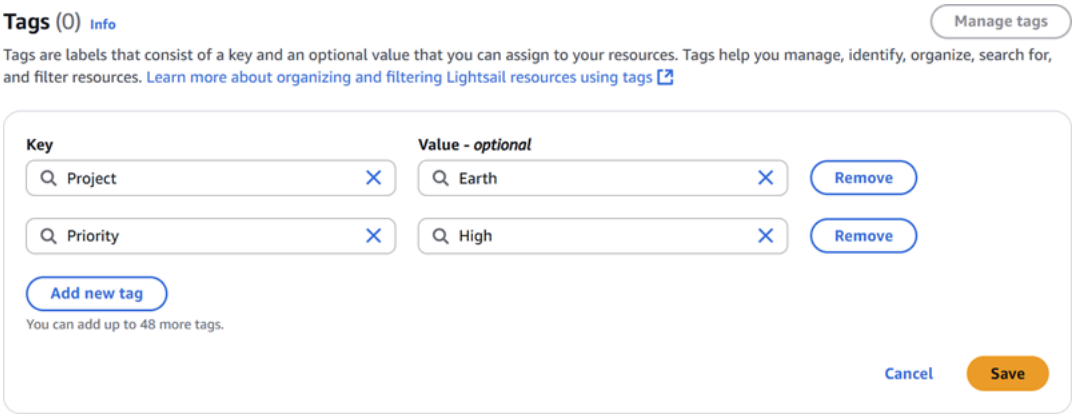
   - Lightsail 계정 AWS 리전 의 각 내에서 고유해야 합니다.
   - 2~255자의 문자로 구성되어야 합니다.
   - 영숫자 문자 또는 숫자로 시작하고 끝나야 합니다.
   - 영숫자 문자, 숫자, 마침표, 대시, 밑줄이 포함될 수 있습니다.
6. 다음 옵션 중 하나를 선택하여 디스크에 태그를 추가합니다.

   - 키 전용 태그를 추가하거나 태그를 관리(이미 태그를 추가한 경우)합니다. 새 태그를 태그 키 텍스트 상자에 입력하고 Enter를 누릅니다. 태그를 추가하려면 태그 입력이 완료될 때 저장을 선택하고, 태그를 추가하지 않으려면 취소를 선택합니다.



- 키-값 태그를 생성한 다음 키를 키 텍스트 상자에 입력하고, 값을 값 텍스트 상자에 입력합니다. 태그 입력이 완료되면 저장을 선택하고, 태그를 추가하지 않으려면 취소를 선택합니다.

  키-값 태그는 저장 전에 한 번에 하나씩만 추가할 수 있습니다. 둘 이상의 키-값 태그를 추가하려면 이전 단계를 반복하세요.



> **Note**
>
> 키 전용 태그 및 키-값 태그에 대한 자세한 내용은 태그를 참조하세요.

7. 디스크 생성을 선택합니다.

   몇 초 뒤 디스크가 생성되고, 디스크 관리 페이지에서 관련 정보를 볼 수 있습니다.

8. 목록에서 원하는 인스턴스를 선택하고 연결을 선택하여 새 디스크를 인스턴스에 연결합니다.

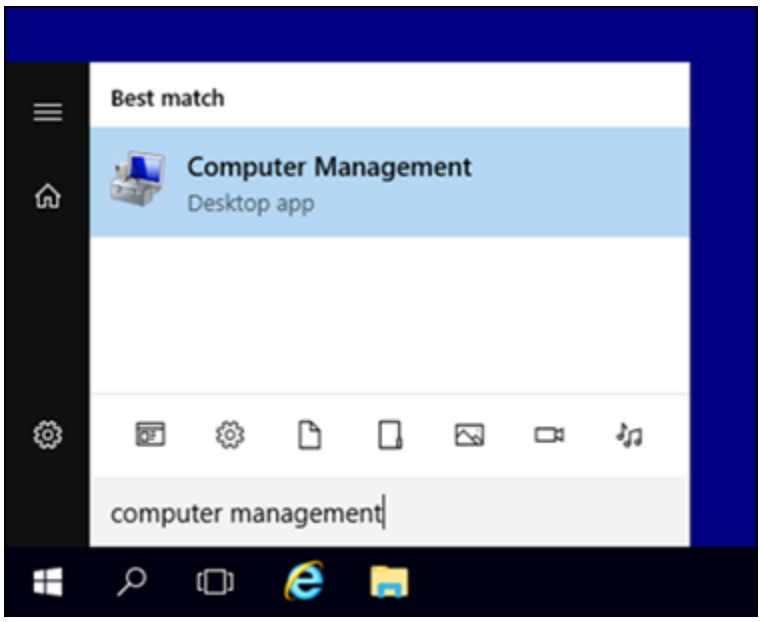
이 안내서의 2단계: 인스턴스에 연결하고 블록 스토리지 디스크를 온라인 상태로 만들기 단원으로 계속 진행하여 블록 스토리지 디스크를 온라인 상태로 만듭니다.

2단계: 인스턴스에 연결하고 블록 스토리지 디스크를 온라인 상태로 만들기

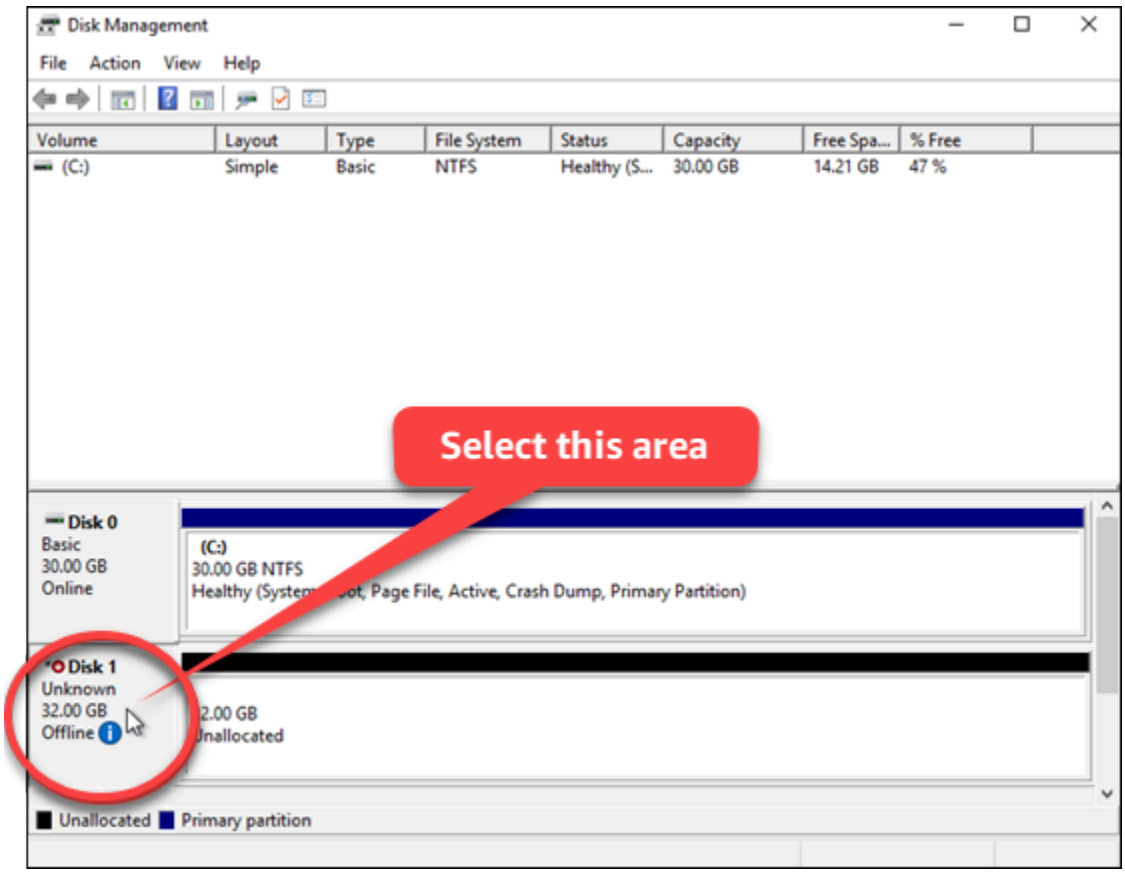
Windows Server 인스턴스에 연결하고 디스크 관리 유틸리티를 사용하여 최근에 연결된 블록 스토리지 디스크를 온라인 상태로 만들 수 있습니다.

인스턴스에 연결하고 블록 스토리지 디스크를 온라인 상태로 만들려면

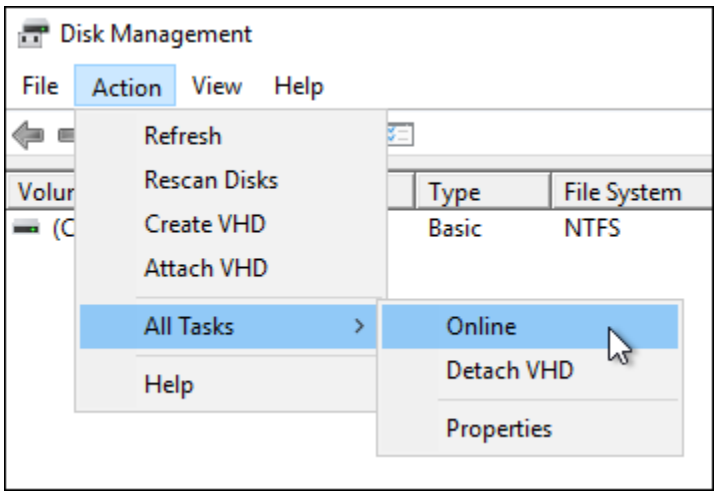
1. Lightsail 콘솔 홈 페이지로 이동합니다.
2. 이 안내서의 앞부분에서 추가 스토리지 디스크를 연결한 인스턴스의 이름을 선택합니다.
3. 연결 탭에서 RDP를 사용하여 연결을 선택합니다.
4. Windows 시작 메뉴에서 컴퓨터 관리를 검색한 다음 검색 결과에서 컴퓨터 관리를 선택합니다.



5. 컴퓨터 관리의 왼쪽 창에서 디스크 관리를 선택합니다.
6. 디스크 관리 유틸리티의 아래쪽 창에서 알 수 없음/오프라인이라는 레이블이 붙은 디스크를 선택합니다. 이 디스크는 이 안내서의 앞부분에서 인스턴스에 연결한 블록 스토리지 디스크입니다.



7. 디스크를 선택하고 작업 메뉴에서 모든 작업을 선택한 다음 온라인을 선택합니다.



블록 스토리지 디스크의 상태가 초기화 안 됨으로 업데이트됩니다. 블록 스토리지 디스크가 아직 온라인 상태가 아닙니다. 이 안내서의 3단계: 블록 스토리지 디스크 초기화 단원으로 계속 진행하여 블록 스토리지 디스크를 초기화합니다.

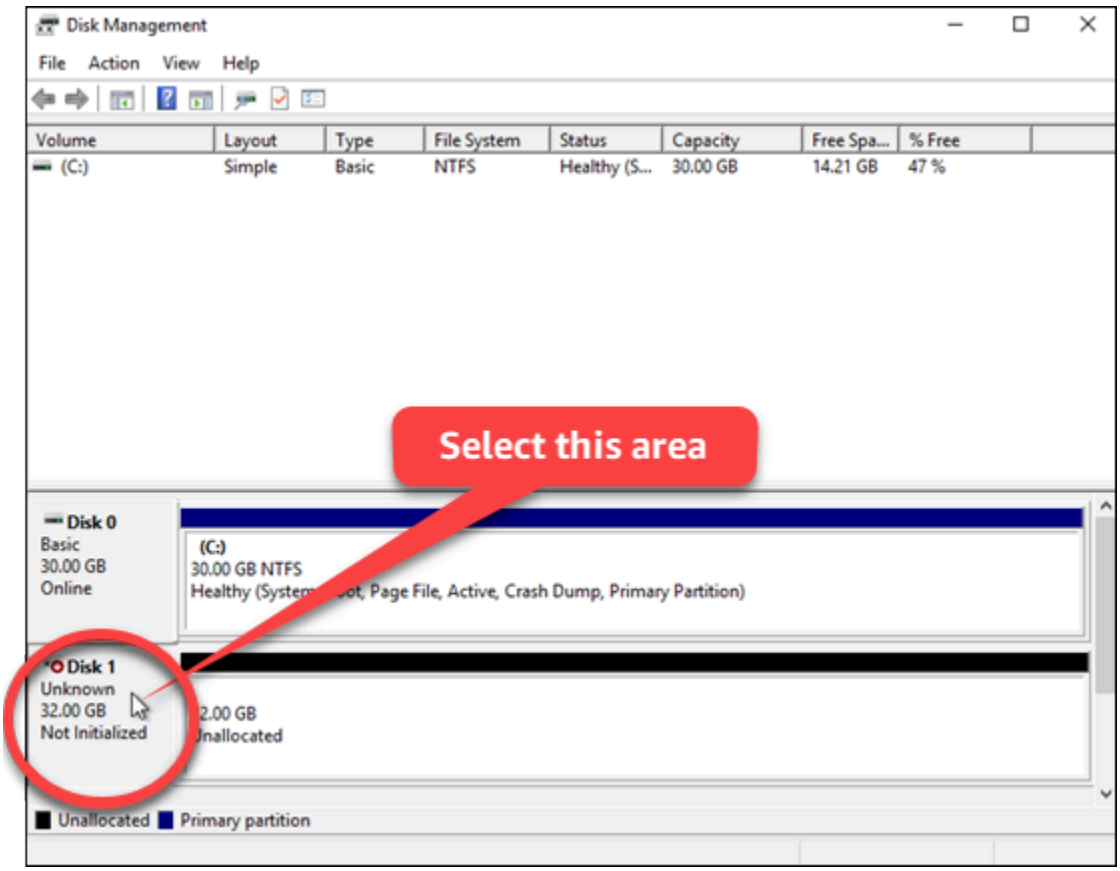
3단계: 블록 스토리지 디스크 초기화

블록 스토리지 디스크를 포맷할 수 있도록 초기화합니다.

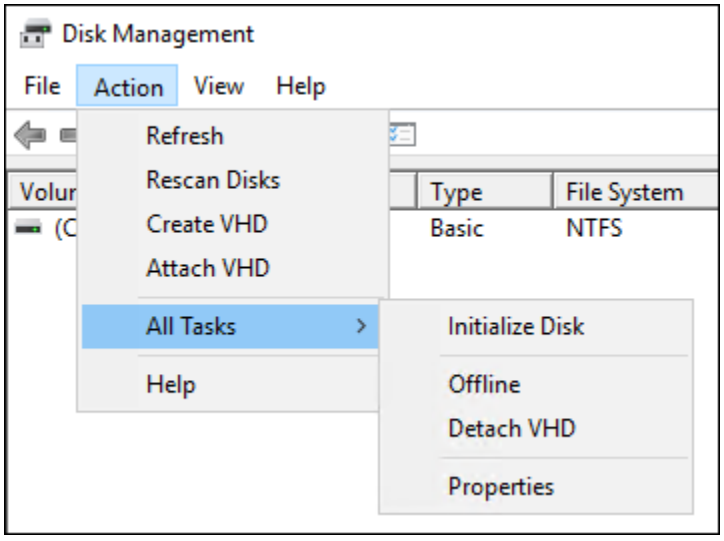
> **Important**
>
> 이미 데이터가 있는 디스크를 탑재하는 경우(예: 스냅샷으로 생성한 디스크), 디스크를 다시 포맷하지 말고 기존 데이터를 삭제하세요.

블록 스토리지 디스크를 초기화하려면

1. 디스크 관리 유틸리티의 아래쪽 창에서 알 수 없음/초기화 안 됨이라는 레이블의 디스크를 선택합니다.



2. 디스크를 선택하고 작업 메뉴에서 모든 작업을 선택한 다음 디스크 초기화를 선택합니다.



3. 새 디스크의 파티션 스타일을 선택하고 확인을 선택합니다.

> **Note**
>
> 파티션 스타일에 대한 자세한 내용은 Microsoft의 About partition styles - GPT and MBR(파티션 스타일 정보 - GPT 및 MBR) 문서를 참조하세요.

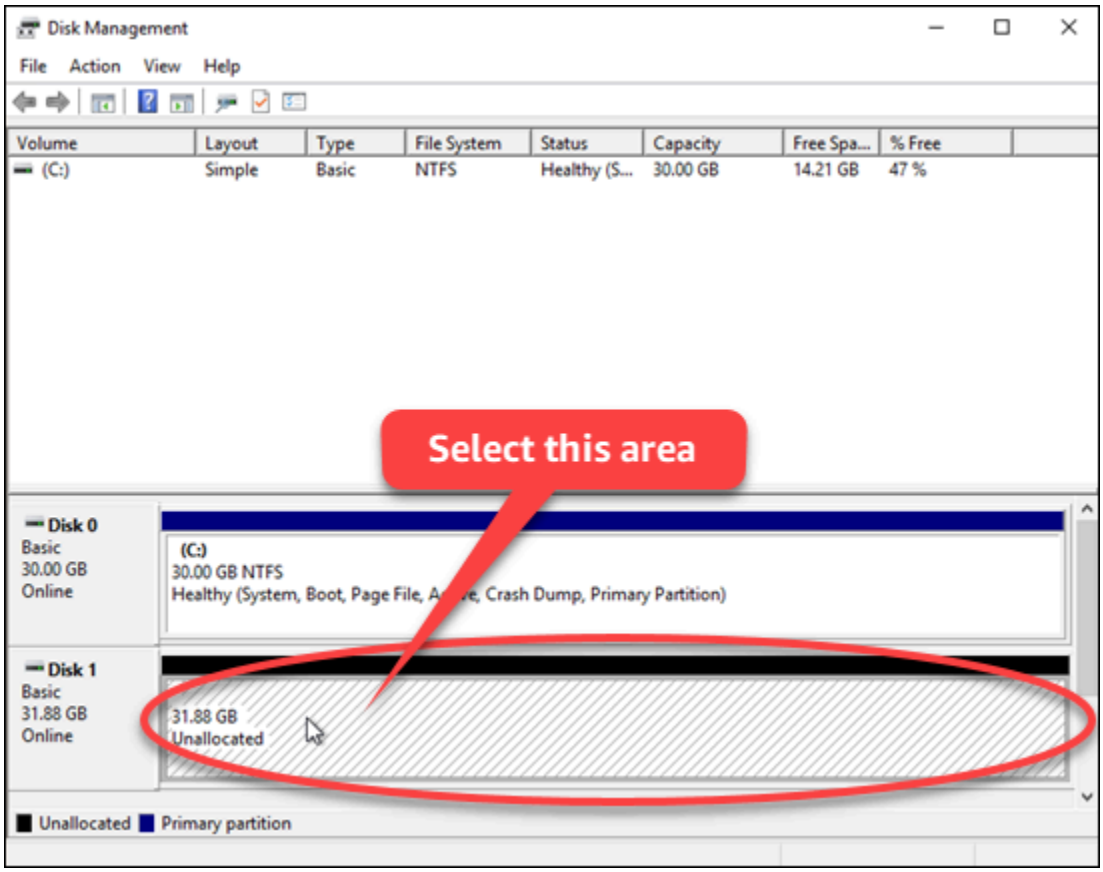
블록 스토리지 디스크의 상태가 온라인으로 업데이트됩니다. 이 안내서의 4단계: 파일 시스템으로 디스크 포맷 단원으로 계속 진행하여 블록 스토리지 디스크를 파일 시스템으로 포맷합니다.

4단계: 파일 시스템으로 디스크 포맷

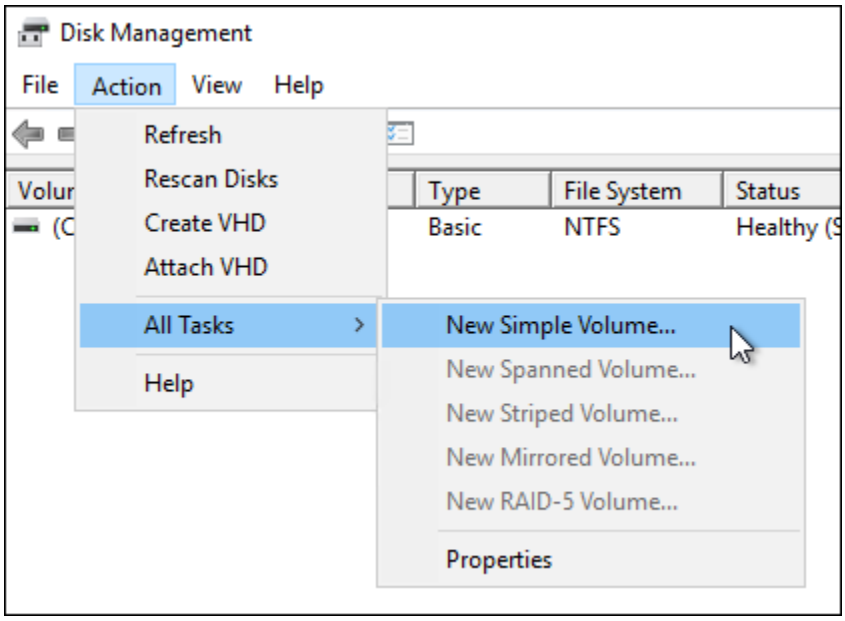
Windows Server의 단순 볼륨 만들기 마법사를 사용하여 드라이브 문자를 지정하고 디스크를 파일 시스템으로 포맷합니다.

파일 시스템으로 디스크를 포맷하려면

1. 디스크 관리 유틸리티의 아래쪽 창에서 할당되지 않음이라는 레이블이 붙은 블록 스토리지 디스크의 파티션을 선택합니다.



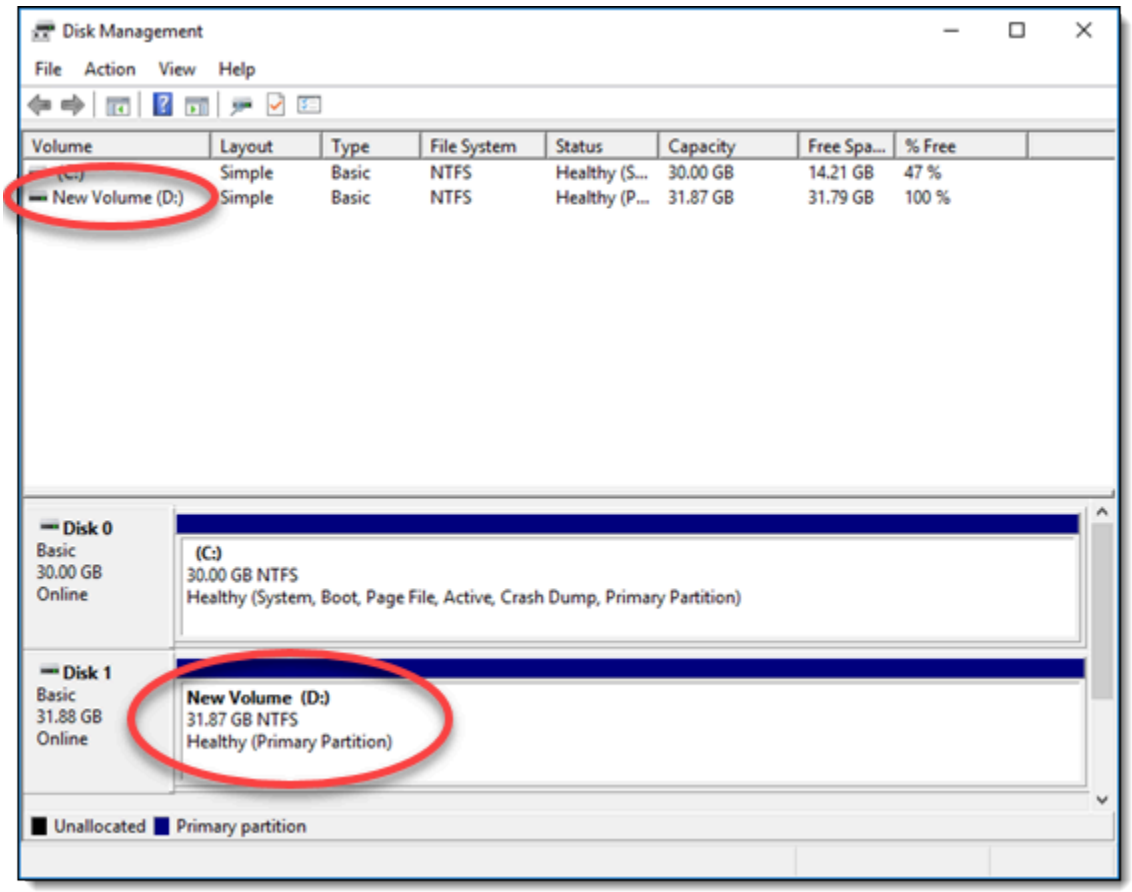
2. 파티션을 선택하고 작업 메뉴에서 모든 작업을 선택한 다음 단순 볼륨 만들기를 선택합니다.



3. 단순 볼륨 만들기 마법사의 지침에 따라 파일 시스템 유형(NTFS, FAT32 또는 ReFS)을 선택하고 디스크를 포맷합니다.

> **Note**
>
> 각 파일 시스템에 대한 자세한 내용은 Microsoft의 NTFS 개요, ReFS(복원 파일 시스템) 개요 및 FAT32 파일 시스템에 대한 설명 문서를 참조하세요.

완료되면 드라이브 문자와 함께 컴퓨터 관리 또는 디스크 관리 유틸리티에 다음 메시지가 나타납니다.



Lightsail 블록 스토리지 디스크 분리 및 삭제

더 이상 필요 없는 블록 스토리지 디스크를 중지된 Amazon Lightsail 인스턴스에서 분리한 다음 삭제할 수 있습니다. 이 주제에서는 데이터를 백업하고 디스크를 안전하게 삭제하는 방법을 보여 줍니다.

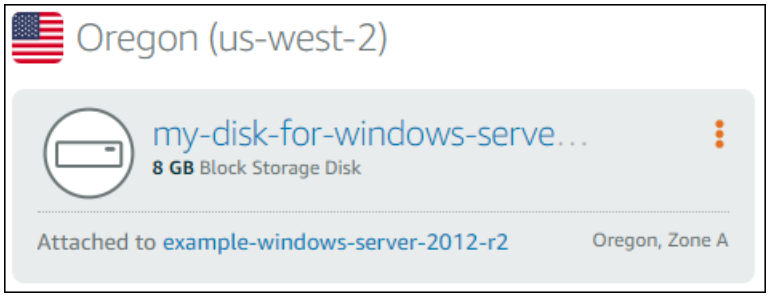
사전 조건

- 실행 중인 인스턴스를 중지합니다. 디스크를 분리하고 삭제하려면 먼저 중지해야 합니다. 인스턴스 중지 방법에 대해 자세히 알아보기
- (선택) 디스크의 스냅샷을 생성하는 것이 좋습니다. 그러면 마음이 바뀌었을 때를 대비하여 백업을 만들 수 있습니다. 자세한 내용은 데이터베이스의 스냅샷 생성을 참조하세요.

디스크 분리 및 삭제

Lightsail 인스턴스가 중지되면 디스크를 안전하게 분리하고 삭제할 수 있습니다.

1. 홈 페이지에서 스토리지를 선택합니다.
2. 연결된 디스크의 이름을 선택하여 관리합니다.



3. 디스크 관리 페이지에서 분리를 선택합니다.

   몇 초가 지나면 디스크가 분리되어 삭제 또는 다시 연결할 수 있는 준비 상태가 됩니다.
4. 삭제 탭을 선택합니다.
5. 디스크 삭제를 선택하고, 예, 삭제를 선택하여 확인합니다.

> **Important**
>
> 이것은 영구적인 작업이며 실행 취소할 수 없습니다. 디스크를 삭제하면 디스크에 저장된 모든 데이터가 손실됩니다.

Amazon Lightsail의 스냅샷

Amazon Lightsail에서 인스턴스, 데이터베이스, 블록 스토리지 디스크의 특정 시점 스냅샷을 생성하고 이를 기준으로 사용하여 새 리소스를 생성하거나 데이터를 백업할 수 있습니다. 스냅샷은 스냅샷을 생성한 시점부터 리소스를 복원하는 데 필요한 모든 데이터를 포함합니다. 스냅샷에서 리소스를 생성하여 리소스를 복원하는 경우 새 리소스는 스냅샷을 생성하는 데 사용된 원래 리소스와 정확히 동일한 복제본으로 시작됩니다. 수동 스냅샷, 자동 스냅샷, 복사된 스냅샷 또는 시스템 디스크 스냅샷 등 Lightsail 계정의 스냅샷에 대한 스냅샷 스토리지 요금이 청구됩니다. 데이터 손상이나 디스크 장애가 발생하는 경우, 생성한 스냅샷으로 디스크를 생성하고 이전 디스크를 교체할 수 있습니다. 스냅샷을 사용하여 새 디스크를 프로비저닝하고 새 인스턴스 시작 중에 연결할 수도 있습니다.

목차

- 수동 스냅샷 수
- 자동 스냅샷
- 시스템 디스크 스냅샷
- 스냅샷에서 새 리소스 생성
- 스냅샷 복사
- Amazon EC2로 스냅샷 내보내기
- 스냅샷 삭제

수동 스냅샷 수

인스턴스, 관리형 데이터베이스 및 블록 스토리지 디스크의 수동 스냅샷을 생성합니다. 수동 스냅샷은 사용자가 삭제할 때까지 무기한 저장됩니다.

수동 스냅샷 생성에 대한 자세한 내용은 다음 가이드를 참조하세요.

- Linux 또는 Unix 인스턴스의 스냅샷 생성
- Windows Server 인스턴스의 스냅샷 생성
- 데이터베이스의 스냅샷 생성
- 블록 스토리지 디스크 스냅샷 생성

자동 스냅샷

Lightsail 인스턴스 또는 블록 스토리지 디스크에서 중요한 정보를 호스팅하는 경우 수동 스냅샷을 생성하여 자주 백업해야 합니다. 그러나 빈번한 관리 작업에 시간을 할애하기가 항상 쉽지는 않습니다. 이 경우 자동 스냅샷을 사용하여 수동 개입 없이 Lightsail에서 자동으로 인스턴스나 블록 스토리지 디스크를 매일 백업하도록 설정할 수 있습니다. 7개의 최신 자동 스냅샷이 매일 단위로 저장되며, 가장 오래된 스냅샷은 가장 최근의 스냅샷으로 교체됩니다.

자동 스냅샷에 대한 자세한 내용은 다음 가이드를 참조하세요.

- 자동 인스턴스 스냅샷 활성화 또는 비활성화
- 인스턴스 또는 디스크의 자동 스냅샷 시간 변경
- 자동 스냅샷 삭제

> **Important**
>
> 소스 리소스를 삭제하면 리소스와 연결된 모든 자동 스냅샷이 삭제됩니다. 이 동작은 수동 스냅샷과는 다른데, 수동 스냅샷의 경우 소스 리소스를 삭제한 후에도 Lightsail 계정에 스냅샷이 유지됩니다. 소스 리소스를 삭제할 때 자동 스냅샷을 유지하려면 자동 스냅샷 유지를 참조하세요.

시스템 디스크 스냅샷

인스턴스가 응답하지 않는 상태에서 시스템 디스크의 파일에 액세스해야 하는 경우, 스냅샷을 생성하여 인스턴스 루트 볼륨을 백업하면 됩니다. 그런 다음, 스냅샷에서 새로운 블록 스토리지 디스크를 생성하고 이를 또 다른 인스턴스에 연결하여 시스템 디스크의 파일에 액세스할 수 있습니다. 자세한 내용은 인스턴스 루트 볼륨의 스냅샷 생성을 참조하세요.

스냅샷에서 새 리소스 생성

스냅샷은 원래 리소스보다 동일한 플랜 또는 더 큰 플랜을 사용하여 새 Lightsail 리소스를 생성하는 데 사용됩니다. 스냅샷은 더 작은 Lightsail 플랜을 사용하여 새 리소스를 생성하는 데 사용할 수 없습니다. 스냅샷을 기반으로 리소스를 생성하는 경우, 새 리소스는 해당 스냅샷을 생성하는 데 사용된 원래 리소스와 일치합니다.

자세한 내용은 다음 안내서를 참조하세요.

- 스냅샷에서 인스턴스 생성
- 스냅샷에서 데이터베이스 생성
- 스냅샷에서 블록 스토리지 디스크 생성
- 스냅샷으로 더 큰 인스턴스, 블록 스토리지 디스크 또는 데이터베이스 생성

스냅샷 복사

동일한 Lightsail 계정 내에서 한 Amazon Web Services(AWS) 리전에서 다른 리전으로 인스턴스 스냅샷 및 블록 스토리지 디스크 스냅샷을 복사할 수 있습니다. 리전 간 데이터베이스 스냅샷은 복사할 수 없습니다. 자세한 내용은 스냅샷을 한 개에서 다른 AWS 리전 개로 복사를 참조하세요.

Amazon EC2로 스냅샷 내보내기

Lightsail를 시작하는 가장 쉬운 방법입니다 AWS. 그러나 Amazon EC2 또는 기타 AWS 서비스에 없는 에는 제한이 Lightsail 있습니다. Lightsail 인스턴스 및 블록 스토리지 디스크 스냅샷을 Amazon EC2로 내보내서 제공되는 광범위한 인스턴스 유형을 활용하고 AWS에서 전체 서비스 범위를 사용하세요. 자세한 내용은 스냅샷을 Amazon EC2로 내보내기를 참조하세요.

> **Note**
>
> cPanel 및 WHM(CentOS 7) 인스턴스의 스냅샷은 Amazon EC2로 내보낼 수 없습니다.

스냅샷 삭제

더 이상 필요하지 않은 Lightsail 스냅샷은 삭제하여 월별 스냅샷 스토리지 요금이 발생하지 않도록 합니다. 자세한 내용은 스냅샷 삭제를 참조하세요.

Lightsail 인스턴스 및 디스크의 자동 스냅샷 구성

인스턴스 또는 블록 스토리지 디스크의 자동 스냅샷 기능을 사용하도록 설정하면 Amazon Lightsail이 기본 자동 스냅샷 시간 또는 사용자가 지정한 시간 동안 리소스의 일별 스냅샷을 생성합니다. 수동 스냅샷과 마찬가지로 자동 스냅샷을 기준으로 사용하여 새 리소스를 생성하거나 데이터를 백업할 수 있습니다.

자동 스냅샷이 생성되면 Lightsail 계정에 저장된 자동 스냅샷에 대한 스냅샷 스토리지 요금이 청구됩니다.

목차

- 자동 스냅샷 제한
- 자동 스냅샷 보존
- Lightsail 콘솔을 사용하여 자동 인스턴스 스냅샷 활성화 또는 비활성화
- AWS CLI를 사용하여 인스턴스 또는 블록 스토리지 디스크의 자동 스냅샷 활성화 또는 비활성화

자동 스냅샷 제한

자동 스냅샷에는 다음 제한이 적용됩니다.

- Lightsail 콘솔을 사용하여 블록 스토리지 디스크의 자동 스냅샷을 활성화하거나 비활성화할 수 없습니다. 블록 스토리지 디스크에 대한 자동 스냅샷을 활성화하거나 비활성화하려면 Lightsail API, AWS Command Line Interface (AWS CLI) 또는 SDKs를 사용해야 합니다. 자세한 내용은 AWS CLI를 사용하여 자동 스냅샷 활성화 또는 비활성화를 참조하세요.
- 현재 자동 스냅샷은 Windows 인스턴스 또는 관리형 데이터베이스에 지원되지 않습니다. 대신, Windows 인스턴스 또는 관리형 데이터베이스의 수동 스냅샷을 생성하여 백업해야 합니다. 자세한 내용은 Windows Server 인스턴스의 스냅샷 생성 또는 데이터베이스의 스냅샷 생성을 참조하세요. 관리형 데이터베이스에는 특정 시점 백업 기능도 기본적으로 지원되므로 데이터를 새 데이터베이스로 복원하는 데 사용할 수 있습니다. 자세한 내용은 특정 시점 백업에서 데이터베이스 생성을 참조하세요.
- 자동 스냅샷은 소스 리소스의 태그를 유지하지 않습니다. 자동 스냅샷에서 만든 새 리소스에 소스 리소스의 태그를 유지하려면 자동 스냅샷에서 새 리소스를 만들 때 태그를 수동으로 추가해야 합니다. 자세한 내용은 리소스에 태그 추가를 참조하세요.

자동 스냅샷 보존

7개의 최신 자동 스냅샷이 매일 저장되며, 가장 오래된 스냅샷은 가장 최근의 스냅샷으로 교체됩니다. 또한, 소스 리소스를 삭제하면 리소스와 연결된 모든 자동 스냅샷이 삭제됩니다. 이 동작은 수동 스냅샷과는 다른데, 수동 스냅샷의 경우 소스 리소스를 삭제한 후에도 Lightsail 계정에 스냅샷이 유지됩니다. 소스 리소스를 삭제할 때 자동 스냅샷이 교체되지 않거나 삭제되지 않도록 하려면 자동 스냅샷을 수동 스냅샷으로 복사하면 됩니다.

리소스에 대한 자동 스냅샷 기능을 사용 중지하면 다음 작업 중 하나를 수행할 때까지 리소스의 기존 자동 스냅샷이 소스 리소스와 함께 유지됩니다.

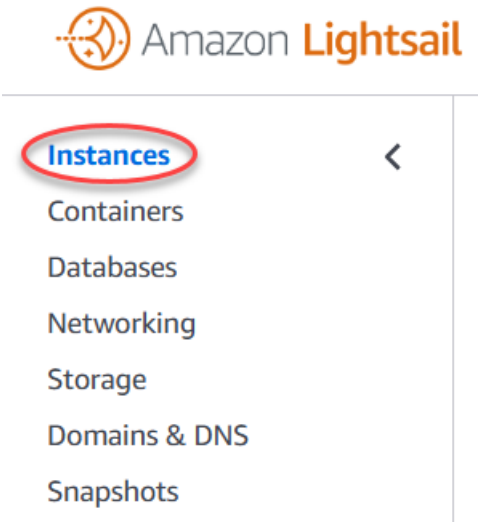
- 자동 스냅샷을 다시 사용하도록 설정하고 기존 자동 스냅샷을 새 스냅샷으로 변경합니다.

- 기존 자동 스냅샷을 수동으로 삭제합니다.
- 소스 리소스를 삭제합니다. 그러면 연결된 자동 스냅샷이 삭제됩니다.

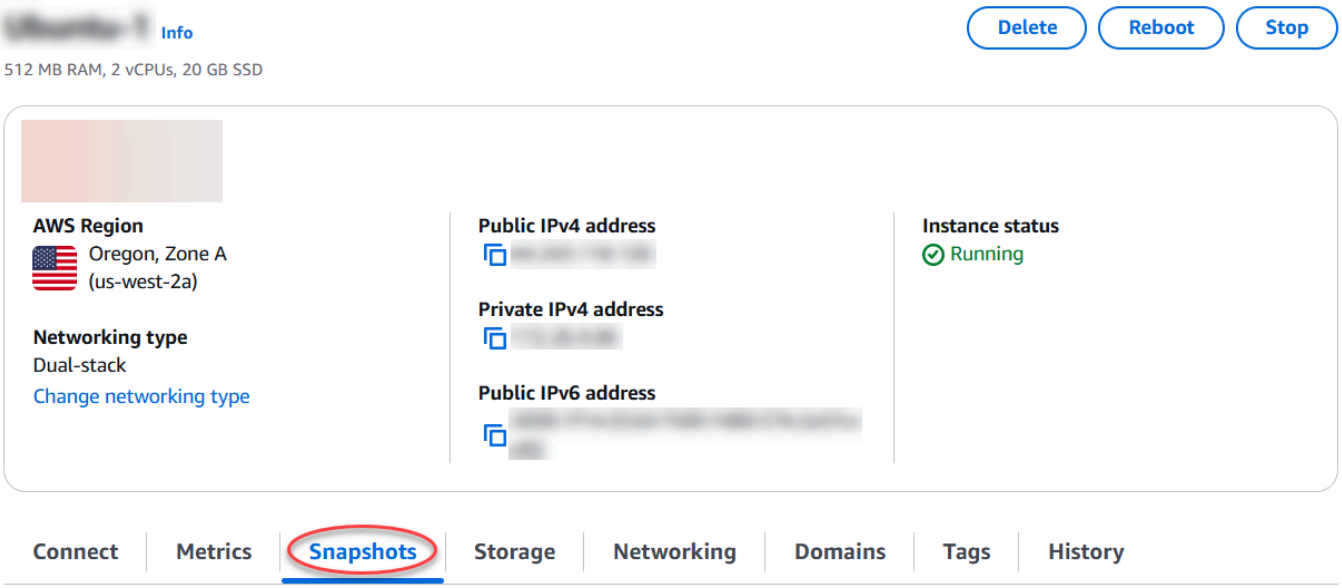
Lightsail 콘솔을 사용하여 자동 인스턴스 스냅샷 활성화 또는 비활성화

Lightsail 콘솔을 사용하여 인스턴스의 자동 스냅샷을 활성화하거나 비활성화하려면 다음 단계를 완료하세요.

1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 인스턴스를 선택합니다.



3. 자동 스냅샷을 활성화하거나 비활성화할 인스턴스의 이름을 선택하세요.
4. 인스턴스 관리 페이지에서 스냅샷 탭을 선택하세요.



5. 자동 스냅샷 섹션에서 토글을 선택하여 활성화하세요. 마찬가지로, 활성화된 경우 비활성화하려면 토글을 선택하세요.
6. 프롬프트에서 예, 활성화를 선택하여 자동 스냅샷을 활성화하거나 예, 비활성화를 선택하여 기능을 비활성화하세요.

   잠시 후 자동 스냅샷이 활성화 또는 비활성화됩니다.

   - 자동 스냅샷 기능을 활성화한 경우 자동 스냅샷 시간도 변경할 수 있습니다. 자세한 내용은 인스턴스 또는 블록 스토리지 디스크의 자동 스냅샷 시간 변경을 참조하세요.
   - 자동 스냅샷 기능을 비활성화한 경우, 사용자가 기능을 다시 활성화하고 새로운 스냅샷으로 교체되거나 사용자가 스냅샷을 삭제할 때까지 리소스의 기존 자동 스냅샷이 유지됩니다. Lightsail 계정에 저장된 자동 스냅샷에 대해서는 스냅샷 스토리지 요금이 청구됩니다. 자동 스냅샷 삭제에 대한 자세한 내용은 자동 인스턴스 스냅샷 삭제를 참조하세요.

를 사용하여 인스턴스 또는 블록 스토리지 디스크에 대한 자동 스냅샷 활성화 또는 비활성화 AWS CLI

AWS CLI를 사용하여 인스턴스 또는 블록 스토리지 디스크의 자동 스냅샷을 활성화하거나 비활성화하려면 다음 단계를 완료하세요.

1. 터미널 또는 명령 프롬프트 창을 엽니다.

   아직 설치하지 않았다면 를 설치하고에서 작동하도록 AWS CLI 구성합니다Lightsail.

2. 자동 스냅샷을 활성화할지 여부에 따라 이 단계에 설명된 명령 중 하나를 입력하세요.

> **Note**
>
> 이 명령에서 autoSnapshotAddOnRequest={snapshotTimeOfDay=HH:00} 파라미터는 선택 사항입니다. 자동 스냅샷을 활성화할 때 일별 자동 스냅샷 시간을 지정하지 않으면 Lightsail이 리소스의 기본 스냅샷 시간을 할당합니다. 자세한 내용은 인스턴스 또는 블록 스토리지 디스크의 자동 스냅샷 시간 변경을 참조하세요.

- 기존 리소스의 자동 스냅샷을 활성화하려면 다음 명령을 입력하세요.

```
aws lightsail enable-add-on --region Region --resource-name ResourceName --add-
on-request
 addOnType=AutoSnapshot,autoSnapshotAddOnRequest={snapshotTimeOfDay=HH:00}
```

명령에서 다음과 같이 바꿉니다.

- 리소스 AWS 리전 가 위치한 *##*입니다.
- *ResourceName*을 리소스의 이름으로 바꿉니다.
- *HH:00*을 시간당 증분으로 협정 세계시(UTC)의 일일 자동 스냅샷 시간으로 바꿉니다.

예:

```
aws lightsail enable-add-on --region us-west-2 --resource-name WordPress-1 --add-
on-request
 addOnType=AutoSnapshot,autoSnapshotAddOnRequest={snapshotTimeOfDay=18:00}
```

- 새 디스크를 생성할 때 자동 스냅샷을 활성화하려면 다음 명령을 입력하세요.

```
aws lightsail create-instances --region Region --availability-
zone AvailabilityZone --blueprint-id BlueprintID --
bundle-id BundleID --instance-name InstanceName --add-ons
 addOnType=AutoSnapshot,autoSnapshotAddOnRequest={snapshotTimeOfDay=HH:00}
```

명령에서 다음과 같이 바꿉니다.

- 인스턴스를 생성해야 AWS 리전 하는이 있는 *##*입니다.
- *AvailabilityZone*을 인스턴스를 생성해야 하는 가용성 영역으로 바꿉니다.
- *BlueprintID*를 인스턴스에 사용할 블루프린트 ID로 바꿉니다.

- *BundleID*를 인스턴스에 사용할 번들 ID로 바꿉니다.
- *InstanceName*을 인스턴스에 사용할 이름으로 바꿉니다.
- *HH:00*을 시간당 증분으로 협정 세계시(UTC)의 일일 자동 스냅샷 시간으로 바꿉니다.

예:

```
aws lightsail create-instances --region us-west-2 --availability-
zone us-west-2a --blueprint-id wordpress_5_1_1_2 --bundle-
id medium_2_0 --instance-name WordPressInstance --add-ons
 addOnType=AutoSnapshot,autoSnapshotAddOnRequest={snapshotTimeOfDay=20:00}
```

- 새 디스크를 생성할 때 자동 스냅샷을 활성화하려면 다음 명령을 입력하세요.

```
aws lightsail create-disk --region Region --availability-
zone AvailabilityZone --size-in-gb Size --disk-name DiskName --add-ons
 addOnType=AutoSnapshot,autoSnapshotAddOnRequest={snapshotTimeOfDay=HH:00}
```

명령에서 다음과 같이 바꿉니다.

- 디스크를 생성해야 하는 AWS 리전 가 있는 *##*입니다.
- *AvailabilityZone*을 디스크를 생성해야 하는 가용성 영역으로 바꿉니다.
- *##*를 원하는 디스크 크기(GB)로 바꿉니다.
- *DiskName*을 디스크에 사용할 이름으로 바꿉니다.
- *HH:00*을 시간당 증분으로 협정 세계시(UTC)의 일일 자동 스냅샷 시간으로 바꿉니다.

예:

```
aws lightsail create-disk --region us-west-2 --availability-
zone us-west-2a --size-in-gb 32 --disk-name Disk01 --add-ons
 addOnType=AutoSnapshot,autoSnapshotAddOnRequest={snapshotTimeOfDay=18:59}
```

- 리소스의 자동 스냅샷을 비활성화하려면 다음 명령을 입력하세요.

```
aws lightsail disable-add-on --region Region --resource-name ResourceName --add-
on-type AutoSnapshot
```

명령에서 다음과 같이 바꿉니다.

- AWS 리전 리소스가 있는 *##*입니다.

- *ResourceName*을 리소스의 이름으로 바꿉니다.

예:

```
aws lightsail disable-add-on --region us-west-1 --resource-
name MyFirstWordPressWebsite01 --add-on-type AutoSnapshot
```

다음 예와 비슷한 결과가 나타나야 합니다.

잠시 후 자동 스냅샷이 활성화 또는 비활성화됩니다.

- 자동 스냅샷을 활성화한 경우 자동 스냅샷 시간도 변경할 수 있습니다. 자세한 내용은 인스턴스 또는 블록 스토리지 디스크의 자동 스냅샷 시간 변경을 참조하세요.
- 자동 스냅샷을 비활성화한 경우, 사용자가 기능을 다시 활성화하고 새로운 스냅샷으로 교체되거나 사용자가 스냅샷을 삭제할 때까지 기존 자동 스냅샷이 유지됩니다. Lightsail 계정에 저장된 자동 스냅샷에 대해서는 스냅샷 스토리지 요금이 청구됩니다. 자동 스냅샷 삭제에 대한 자세한 내용은 자동 인스턴스 스냅샷 삭제를 참조하세요.

> **Note**
>
> 이러한 명령의 EnableAddOn 및 DisableAddOn API 작업에 대한 자세한 내용은 Lightsail API 문서의 EnableAddOn 및 DisableAddOn을 참조하세요.

Lightsail 인스턴스 및 디스크의 자동 스냅샷 일정 조정

인스턴스 또는 블록 스토리지 디스크의 자동 스냅샷 기능을 활성화하면 Lightsail이 기본 자동 스냅샷 시간 또는 사용자가 지정한 시간 동안 리소스의 일별 스냅샷을 생성합니다. 이 가이드의 단계에 따라 리소스의 자동 스냅샷 시간을 변경하세요.

목차

- 자동 스냅샷 시간 제한
- 의 기본 자동 스냅샷 시간 AWS 리전
- Lightsail 콘솔을 사용하여 자동 스냅샷 시간 변경
- 를 사용하여 자동 스냅샷 시간 및 블록 스토리지 디스크 변경 AWS CLI

자동 스냅샷 시간 제한

자동 스냅샷 시간에는 다음 제한이 적용됩니다.

- Lightsail 콘솔을 사용하여 블록 스토리지 디스크의 자동 스냅샷 시간을 변경할 수 없습니다. 블록 스토리지 디스크의 자동 스냅샷 시간을 변경하려면 Lightsail API, AWS Command Line Interface (AWS CLI) 또는 SDKs를 사용해야 합니다. 자세한 내용은 AWS CLI를 사용하여 자동 스냅샷 시간 변경을 참조하세요.
- 자동 스냅샷 시간은 시간당 증분으로 지정할 수 있습니다. 또한 현재 시간에서 30분 이상 경과한 시간이어야 합니다. Lightsail은 사용자가 지정한 시간으로부터 최대 45분 동안 자동 스냅샷을 생성합니다.

> **Important**
>
> 자동 스냅샷이 생성되는 동안에는 수동 스냅샷을 생성할 수 없습니다.

- 리소스의 자동 스냅샷 시간을 변경하면 다음 조건을 제외한 일반적인 경우 즉시 적용됩니다.

- 현재 날짜의 자동 스냅샷이 이미 생성된 상태에서 스냅샷 시간을 더 늦은 시간으로 변경하면 새 스냅샷 시간은 다음 날 적용됩니다. 그러므로 현재 날짜에 두 개의 스냅샷이 생성되지 않습니다.
- 현재 날짜의 자동 스냅샷이 아직 생성되지 않은 상태에서 스냅샷 시간을 더 이른 시간으로 변경하면 새 스냅샷 시간은 다음 날 적용됩니다. 또한 현재 날짜의 이전에 설정된 시간에 스냅샷이 자동으로 생성됩니다. 그러므로 현재 날짜에 하나의 스냅샷이 생성됩니다.
- 현재 날짜의 자동 스냅샷이 아직 생성되지 않은 상태에서 스냅샷 시간을 현재 시간으로부터 30분 이내의 시간으로 변경하면 새 스냅샷 시간은 다음 날 적용됩니다. 또한 현재 날짜의 이전에 설정된 시간에 스냅샷이 자동으로 생성됩니다. 그러므로 현재 날짜에 하나의 스냅샷이 생성됩니다. 현재 시간과 새로 지정하는 스냅샷 시간 사이에 30분이 필요하기 때문입니다.
- 현재 시간으로부터 30분 이내에 자동 스냅샷이 생성되도록 예약된 상태에서 스냅샷 시간을 변경하면 새 스냅샷 시간은 다음 날 적용됩니다. 또한 현재 날짜의 이전에 설정된 시간에 스냅샷이 자동으로 생성됩니다. 그러므로 현재 날짜에 하나의 스냅샷이 생성됩니다. 현재 시간과 새로 지정하는 스냅샷 시간 사이에 30분이 필요하기 때문입니다.

위 조건에 해당하는 경우, 새 스냅샷 시간이 적용되려면 최대 24시간이 걸릴 수 있음을 알리는 메시지가 Lightsail 콘솔에 표시됩니다.

의 기본 자동 스냅샷 시간 AWS 리전

자동 스냅샷을 활성화할 때 자동 스냅샷 시간을 지정하지 않으면 Lightsail이 다음 기본 자동 스냅샷 시간 중 하나를 지정합니다. 시간은 인스턴스 또는 블록 스토리지 디스크가 AWS 리전 있는에 따라 달라집니다.

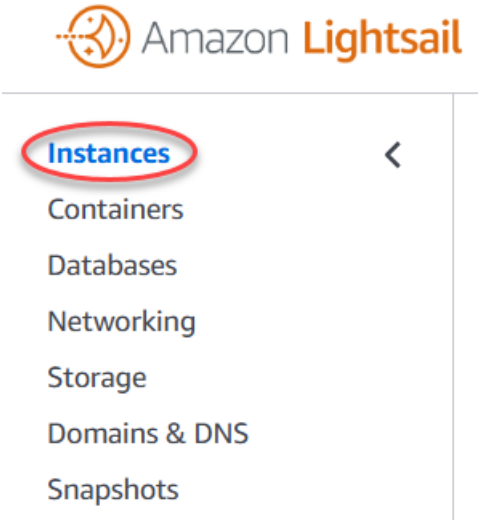
- 미국 동부(버지니아 북부)(us-east-1): 06:00 UTC
- 미국 동부(오하이오)(us-east-2): 03:00 UTC
- 미국 서부(오레곤)(us-west-2): 06:00 UTC
- 아시아 태평양(뭄바이)(ap-south-1): 17:00 UTC
- 아시아 태평양(서울)(ap-northeast-2): 13:00 UTC
- 아시아 태평양(싱가포르)(ap-southeast-1): 14:00 UTC
- 아시아 태평양(시드니)(ap-southeast-2): 12:00 UTC
- 아시아 태평양(자카르타)(ap-southeast-3): 08:00 UTC
- 아시아 태평양(도쿄)(ap-northeast-1): 13:00 UTC
- 캐나다(중부)(ca-central-1): 06:00 UTC
- EU(프랑크푸르트)(eu-central-1): 20:00 UTC

- EU(아일랜드)(eu-west-1): 22:00 UTC
- EU(런던)(eu-west-2): 06:00 UTC
- EU(파리)(eu-west-3): 07:00 UTC
- EU(스톡홀름)(eu-north-1): 08:00 UTC

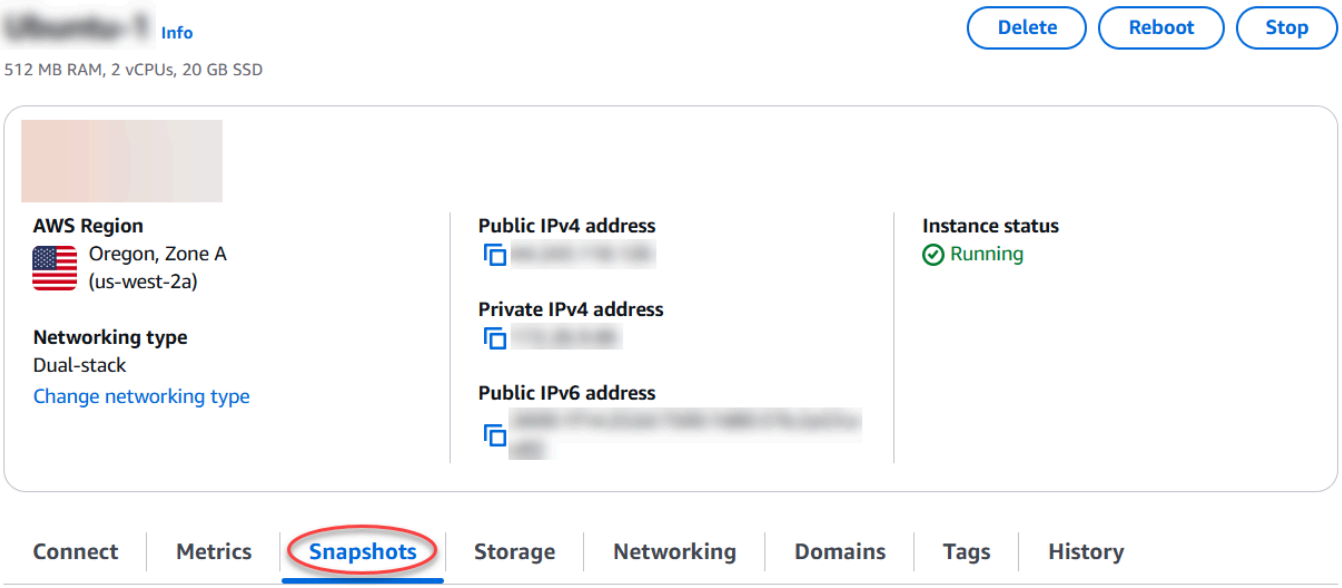
Lightsail 콘솔을 사용하여 자동 스냅샷 시간 변경

Lightsail 콘솔을 사용하여 인스턴스의 자동 스냅샷 시간을 변경하려면 다음 단계를 완료하세요.

1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 인스턴스를 선택합니다.



3. 자동 스냅샷 시간을 변경할 인스턴스의 이름을 선택하세요.
4. 인스턴스 관리 페이지에서 스냅샷 탭을 선택하세요.



5. 자동 스냅샷 섹션에서 스냅샷 시간 변경을 선택하세요.
6. Lightsail이 자동 스냅샷을 생성하기를 원하는 시간을 선택하세요. 선택하는 시간은 협정 세계시(UTC)여야 합니다.
7. 새 스냅샷 시간을 저장하려면 변경을 선택하세요.

   잠시 후에 자동 스냅샷 시간이 업데이트됩니다. 새로운 자동 스냅샷 시간의 유효 날짜에 제한이 있을 수 있습니다. 자세한 내용은 자동 스냅샷 시간 제한을 참조하세요.

를 사용하여 인스턴스 및 블록 스토리지 디스크의 자동 스냅샷 시간 변경 AWS CLI

AWS CLI를 사용하여 인스턴스 또는 블록 스토리지 디스크의 자동 스냅샷 시간을 변경하려면 다음 단계를 완료하세요.

1. 터미널 또는 명령 프롬프트 창을 엽니다.

   아직 설치하지 않았다면 를 AWS CLI 설치하고에서 작동하도록 구성합니다Lightsail.
2. 리소스의 자동 스냅샷 시간을 변경하려면 다음 명령을 입력하세요.

```
aws lightsail enable-add-on --region Region --resource-name ResourceName --add-on-
request addOnType=AutoSnapshot,autoSnapshotAddOnRequest={snapshotTimeOfDay=HH:00}
```

   명령에서 다음과 같이 바꿉니다.

   - AWS 리전 리소스가 위치한 ##입니다.

- *ResourceName*을 리소스의 이름으로 바꿉니다.
- *HH:00*을 시간당 증분으로 협정 세계시(UTC)의 일일 자동 스냅샷 시간으로 바꿉니다.

예:

```
aws lightsail enable-add-on --region us-west-1 --resource-
name MyFirstWordPressWebsite01 --add-on-request
 addOnType=AutoSnapshot,autoSnapshotAddOnRequest={snapshotTimeOfDay=12:00}
```

다음 예와 비슷한 결과가 나타나야 합니다.

잠시 후에 자동 스냅샷 시간이 업데이트됩니다. 새로운 자동 스냅샷 시간의 유효 날짜에 제한이 있을 수 있습니다. 자세한 내용은 자동 스냅샷 시간 제한을 참조하세요.

> **Note**
>
> 이 명령의 EnableAddOn API 작업에 대한 자세한 내용은 Lightsail API 문서의 EnableAddOn을 참조하세요.

미사용 Lightsail 인스턴스 및 디스크 스냅샷 삭제

기능이 활성화되어 있든 활성화된 후에 비활성화되어 있든 상관 없이, Amazon Lightsail에서 언제든 인스턴스 또는 블록 스토리지 디스크의 자동 스냅샷을 삭제할 수 있습니다. Lightsail 계정에 저장된 자동 스냅샷에 대해서는 스냅샷 스토리지 요금이 청구됩니다. 더 이상 필요하지 않은 자동 스냅샷은 이 가이

드의 단계에 따라 삭제하세요. 자동 스냅샷을 수동 스냅샷에 복사하여 더 이상 원본이 필요하지 않거나, 리소스의 자동 스냅샷 기능을 비활성화하여 보관된 기존 자동 스냅샷이 필요하지 않은 경우 등을 예로 들 수 있습니다.

목차

- 자동 스냅샷 삭제 제한
- Lightsail 콘솔을 사용하여 인스턴스의 자동 스냅샷 삭제
- 를 사용하여 인스턴스 또는 블록 스토리지 디스크의 자동 스냅샷 삭제 AWS CLI

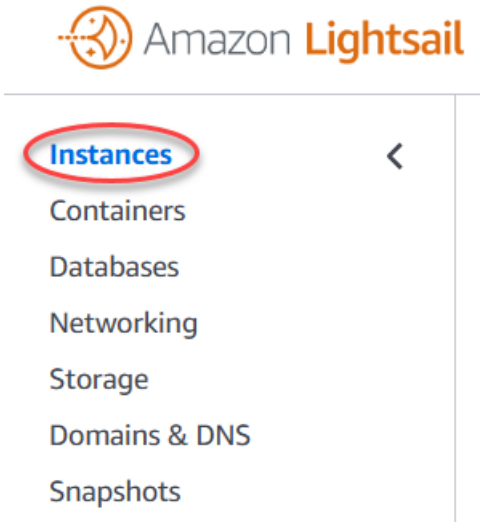
자동 스냅샷 삭제 제한

Lightsail 콘솔을 사용하여 블록 스토리지 디스크의 자동 스냅샷을 삭제할 수 없습니다. 블록 스토리지 디스크의 자동 스냅샷을 삭제하려면 Lightsail API, AWS Command Line Interface (AWS CLI) 또는 SDKs를 사용해야 합니다. 자세한 내용은 AWS CLI를 사용하여 인스턴스 또는 블록 스토리지 디스크의 자동 스냅샷 삭제를 참조하세요.

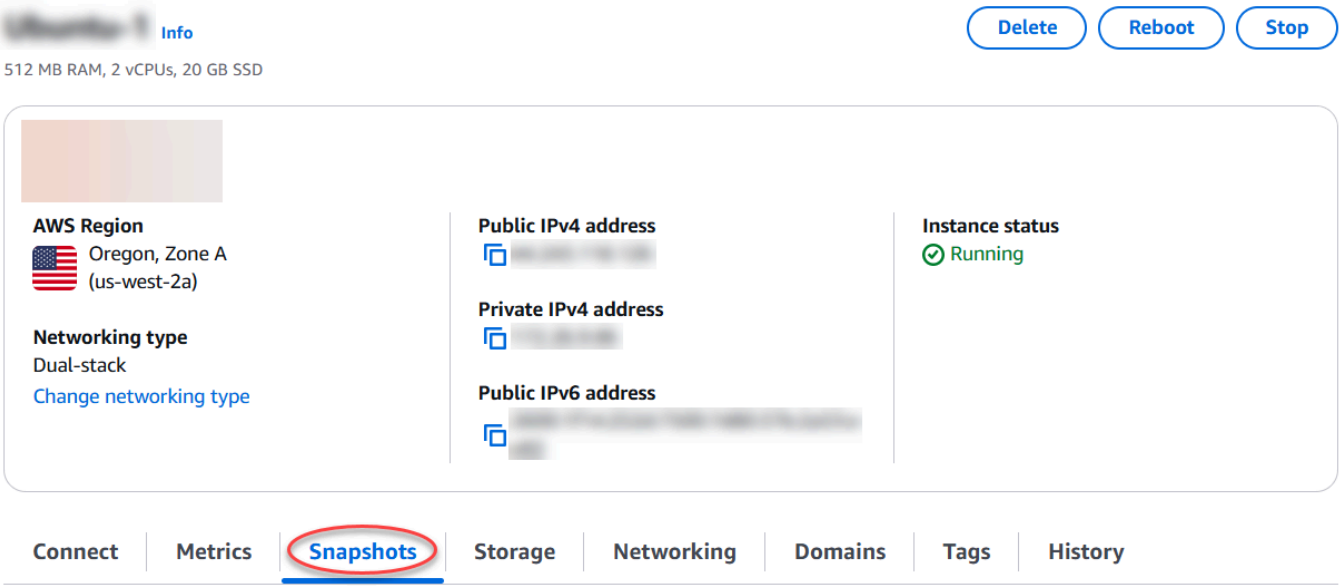
Lightsail 콘솔을 사용하여 인스턴스의 자동 스냅샷 삭제

Lightsail 콘솔을 사용하여 인스턴스의 자동 스냅샷을 삭제하려면 다음 단계를 완료하세요.

1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 인스턴스를 선택합니다.



3. 자동 스냅샷을 삭제할 인스턴스의 이름을 선택하세요.
4. 인스턴스 관리 페이지에서 스냅샷 탭을 선택하세요.



5. 자동 스냅샷 섹션에서 삭제할 자동 스냅샷 옆에 있는 줄임표 아이콘을 선택한 다음 스냅샷 삭제를 선택하세요.
6. 프롬프트에서 예를 선택하여 스냅샷 삭제를 확인합니다.

   잠시 후에 자동 스냅샷이 삭제됩니다.

를 사용하여 인스턴스 또는 블록 스토리지 디스크의 자동 스냅샷 삭제 AWS CLI

AWS CLI를 사용하여 인스턴스 또는 블록 스토리지 디스크의 자동 스냅샷을 삭제하려면 다음 단계를 완료하세요.

1. 터미널 또는 명령 프롬프트 창을 엽니다.

   아직 설치하지 않았다면 를 AWS CLI 설치하고에서 작동하도록 구성합니다Lightsail.
2. 특정 리소스에 사용 가능한 자동 스냅샷 날짜를 가져오려면 다음 명령을 입력하세요. 후속 명령에 `date` 파라미터로 지정하려면 자동 스냅샷 날짜가 필요합니다.

   ```
   aws lightsail --region Region get-auto-snapshots --resource-name ResourceName
   ```

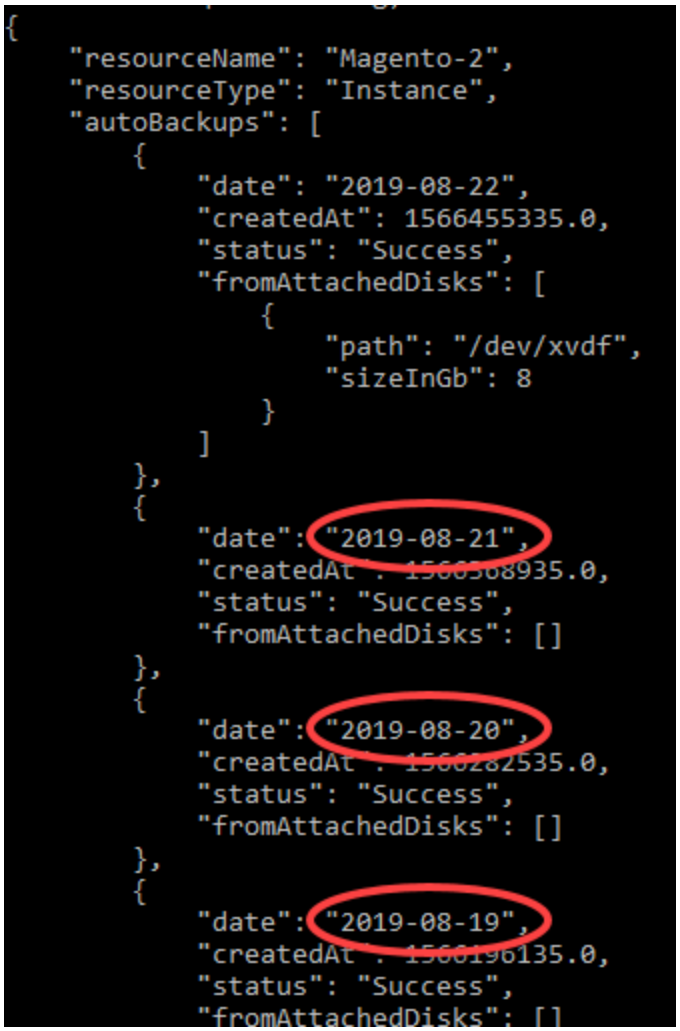
   명령에서 다음과 같이 바꿉니다.

   - AWS 리전 리소스가 위치한 *##*입니다.
   - *ResourceName*을 리소스의 이름으로 바꿉니다.

예:

```
aws lightsail --region us-west-2 get-auto-snapshots --resource-
name MyFirstWordPressWebsite01
```

다음과 같이 사용 가능한 자동 스냅샷이 나열된 결과가 나타납니다.



3. 다음 명령을 입력하여 자동 스냅샷을 삭제합니다.

```
aws lightsail --region Region delete-auto-snapshot --resource-name ResourceName --
date YYYY-MM-DD
```

명령에서 다음과 같이 바꿉니다.

- AWS 리전 리소스가 위치한 ##입니다.

- *ResourceName*을 리소스의 이름으로 바꿉니다.
- *YYYY-MM-DD*를 앞의 명령을 사용하여 얻은 사용 가능한 자동 스냅샷의 날짜로 바꿉니다.

예:

```
aws lightsail --region us-west-2 delete-auto-snapshot --resource-
name MyFirstWordPressWebsite01 --date 2019-09-16
```

다음 예와 비슷한 결과가 나타나야 합니다.


```
{
    "operation": {
        "id": "8f253c00-c34f-4073-9b0e-e5507ce264d9",
        "resourceName": "Magento-2",
        "resourceType": "Instance",
        "createdAt": 1566507472.323,
        "location": {
            "availabilityZone": "us-west-2",
            "regionName": "us-west-2"
        },
        "isTerminal": true,
        "operationDetails": "DeleteAutoBackup-2019-08-16",
        "operationType": "DeleteAutoBackup",
        "status": "Succeeded"
    }
}
```


잠시 후에 자동 스냅샷이 삭제됩니다.

> **Note**
>
> 이러한 명령의 GetAutoSnapshots 및 DeleteAutoSnapshot API 작업에 대한 자세한 내용은 Lightsail API 문서의 GetAutoSnapshots 및 DeleteAutoSnapshot을 참조하세요.

Lightsail에서 자동 스냅샷이 교체되는 것을 방지

Amazon Lightsail에서 인스턴스 또는 블록 스토리지 디스크의 자동 스냅샷 기능을 활성화하면 리소스의 최신 자동 스냅샷 7개만 매일 저장됩니다. 그런 다음 가장 오래된 스냅샷이 최신 스냅샷으로 바뀝니다. 또한, 소스 리소스를 삭제하면 리소스와 연결된 모든 자동 스냅샷이 삭제됩니다.

소스 리소스를 삭제할 때 특정 자동 스냅샷이 교체되거나 삭제되지 않도록 하려면 해당 스냅샷을 수동 스냅샷으로 복사하면 됩니다. 수동 스냅샷은 사용자가 직접 삭제할 때까지 유지됩니다.

자동 스냅샷을 수동 스냅샷으로 복사하여 유지하려면 이 가이드의 단계를 따르세요. Lightsail 계정에 저장된 자동 스냅샷에 대해서는 스냅샷 스토리지 요금이 청구됩니다.

> **Note**
>
> 리소스의 자동 스냅샷 기능을 비활성화하면, 사용자가 기능을 다시 활성화하고 새로운 스냅샷으로 교체되거나 자동 스냅샷을 삭제할 때까지 리소스의 기존 자동 스냅샷이 유지됩니다.

목차

- 자동 스냅샷 유지 제한
- Lightsail 콘솔을 사용하여 인스턴스의 자동 스냅샷 유지
- 를 사용하여 인스턴스 및 블록 스토리지 디스크의 자동 스냅샷 유지 AWS CLI

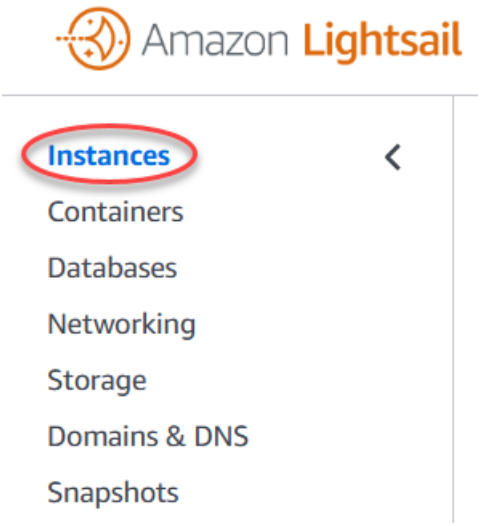
자동 스냅샷 유지 제한

Lightsail 콘솔을 사용하여 블록 스토리지 디스크의 자동 스냅샷을 수동 스냅샷으로 복사할 수 없습니다. 블록 스토리지 디스크의 자동 스냅샷을 복사하려면 Lightsail API, AWS Command Line Interface (AWS CLI) 또는 SDKs를 사용해야 합니다. 자세한 내용은 AWS CLI를 사용하여 인스턴스 및 블록 스토리지 디스크의 자동 스냅샷 유지를 참조하세요.

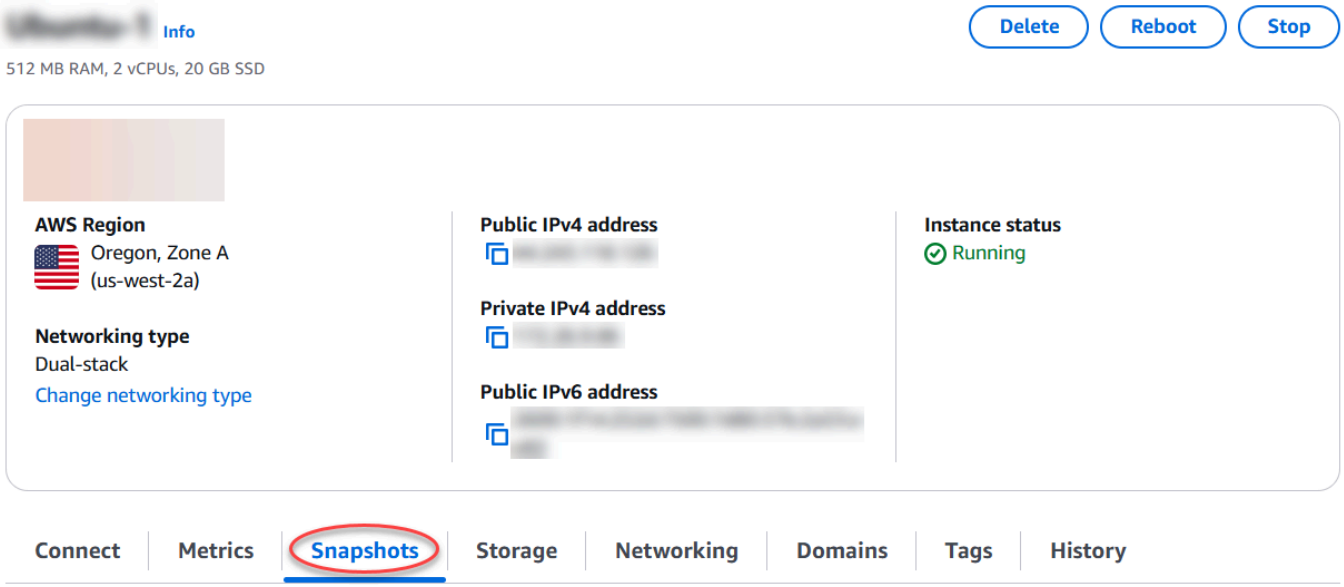
Lightsail 콘솔을 사용하여 인스턴스의 자동 스냅샷 유지

Lightsail 콘솔을 사용하여 인스턴스의 자동 스냅샷을 유지하려면 다음 단계를 완료하세요.

1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 인스턴스를 선택합니다.



3. 자동 스냅샷을 유지할 인스턴스의 이름을 선택하세요.
4. 인스턴스 관리 페이지에서 스냅샷 탭을 선택하세요.



5. 자동 스냅샷 섹션에서 유지할 자동 스냅샷 옆에 있는 줄임표 아이콘을 선택한 다음 스냅샷 유지를 선택하세요.
6. 프롬프트에서 Yes, save(예, 저장)를 선택하여 자동 스냅샷 유지를 확인합니다.

   잠시 후에 자동 스냅샷이 수동 스냅샷으로 복사됩니다. 수동 스냅샷은 사용자가 삭제할 때까지 유지됩니다.

> **Important**
>
> 자동 스냅샷이 더 이상 필요하지 않으면 삭제하는 것이 좋습니다. 그렇지 않으면 Lightsail 계정에 저장된 자동 스냅샷 및 복제 수동 스냅샷에 대해 스냅샷 스토리지 비용이 청구됩니다. 자세한 내용은 자동 인스턴스 스냅샷 삭제를 참조하세요.

를 사용하여 인스턴스 및 블록 스토리지 디스크의 자동 스냅샷 유지 AWS CLI

AWS CLI를 사용하여 인스턴스 또는 블록 스토리지 디스크의 자동 스냅샷을 유지하려면 다음 단계를 완료하세요.

1. 터미널 또는 명령 프롬프트 창을 엽니다.

   아직 설치하지 않았다면 를 AWS CLI 설치하고에서 작동하도록 구성합니다Lightsail.

2. 특정 리소스에 사용 가능한 자동 스냅샷 날짜를 가져오려면 다음 명령을 입력하세요. 후속 명령에 `restore date` 파라미터로 지정하려면 자동 스냅샷 날짜가 필요합니다.

   ```
   aws lightsail get-auto-snapshots --region Region --resource-name ResourceName
   ```

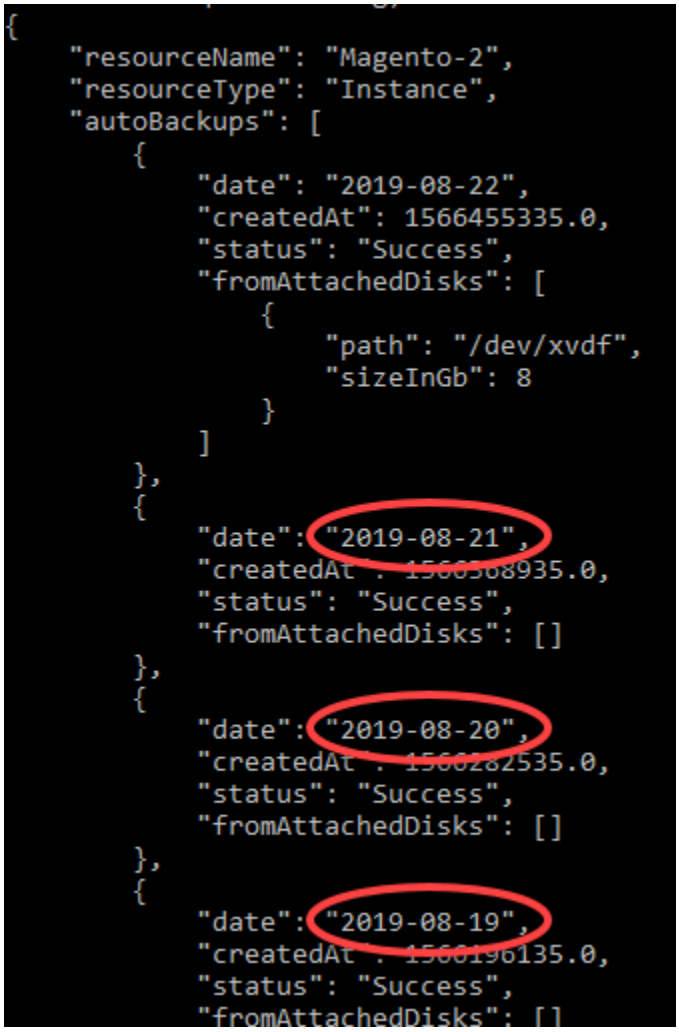
   명령에서 다음과 같이 바꿉니다.

   - AWS 리전 리소스가 위치한 *##*입니다.
   - *ResourceName*을 리소스의 이름으로 바꿉니다.

   예:

   ```
   aws lightsail get-auto-snapshots --region us-west-2 --resource-
   name MyFirstWordPressWebsite01
   ```

   다음과 같이 사용 가능한 자동 스냅샷이 나열된 결과가 나타납니다.



3. 특정 리소스의 자동 스냅샷을 유지하려면 다음 명령을 입력하세요.

```
aws lightsail copy-snapshot --region TargetRegion --source-resource-
name ResourceName --restore-date YYYY-MM-DD --source-region SourceRegion --target-
snapshot-name SnapshotName
```

명령에서 다음과 같이 바꿉니다.

- 스냅샷을 복사할 AWS 리전 가 있는 *TargetRegion*.
- *ResourceName*을 리소스의 이름으로 바꿉니다.
- *YYYY-MM-DD*를 앞의 명령을 사용하여 얻은 사용 가능한 자동 스냅샷의 날짜로 바꿉니다.
- 자동 스냅샷이 현재 있는 AWS 리전 이 있는 *SourceRegion*.
- *DiskSnapshotName*을 생성할 새 스냅샷의 이름으로 바꿉니다.

예:

```
aws lightsail copy-snapshot --region us-west-2 --source-resource-
name MyFirstWordPressWebsite01 --restore-date 2019-09-16 --source-region us-west-2
 --target-snapshot-name Snapshot-Copied-From-Auto-Snapshot
```

다음 예와 비슷한 결과가 나타나야 합니다.


```
{
    "operations": [
        {
            "id": "6f2607ca-c3d3-4e92-9795-8d7c8d72b038",
            "resourceName": "Snapshot-Copied-From-Auto-Backup",
            "resourceType": "InstanceSnapshot",
            "createdAt": 1566504306.107,
            "location": {
                "availabilityZone": "all",
                "regionName": "us-west-2"
            },
            "isTerminal": false,
            "operationDetails": "us-west-2:Magento-2",
            "operationType": "CopySnapshot",
            "status": "Started",
            "statusChangedAt": 1566504306.107
        }
    ]
}
```


잠시 후에 자동 스냅샷이 수동 스냅샷으로 복사됩니다. 수동 스냅샷은 사용자가 삭제할 때까지 유지됩니다.

> **Important**
>
> 자동 스냅샷이 더 이상 필요하지 않으면 삭제하는 것이 좋습니다. 그렇지 않으면 Lightsail 계정에 저장된 자동 스냅샷 및 복제 수동 스냅샷에 대해 스냅샷 스토리지 비용이 청구됩니다. 자세한 내용은 자동 인스턴스 스냅샷 삭제를 참조하세요.

> **Note**
>
> 이러한 명령의 GetAutoSnapshots 및 CopySnapshot API 작업에 대한 자세한 내용은 Lightsail API 문서의 GetAutoSnapshots 및 CopySnapshot을 참조하세요.

스냅샷으로 Linux/Unix Lightsail 인스턴스 백업

Linux/Unix 기반 Amazon Lightsail 인스턴스의 스냅샷을 생성할 수 있습니다. 인스턴스 스냅샷이란 시스템 디스크의 복제본으로, 원래 시스템 구성(메모리, CPU, 디스크 크기, 데이터 전송 속도)과 일치합니다. 인스턴스에 블록 스토리지 디스크를 연결한 경우 Lightsail은 그러한 추가 디스크를 스냅샷에 포함하여 복사합니다. 자세한 내용은 스냅샷을 참조하세요.

> **Note**
>
> Windows Server 기반 Lightsail 인스턴스의 스냅샷을 만드는 단계는 다릅니다. 자세한 내용은 Windows Server 인스턴스의 스냅샷 생성을 참조하세요.

Lightsail에 스냅샷을 생성할 인스턴스가 이미 있는 상태여야 합니다. 인스턴스가 있는 경우 다음 단계에 따라 스냅샷을 생성합니다.

1. Lightsail 홈 페이지에서 스냅샷을 생성할 인스턴스의 이름을 선택합니다.
2. 스냅샷 탭을 선택합니다.
3. 페이지의 수동 스냅샷 섹션에서 스냅샷 생성을 선택한 다음 스냅샷 이름을 입력합니다.

   리소스 이름:

   - Lightsail 계정 AWS 리전 의 각 내에서 고유해야 합니다.
   - 2~255자의 문자로 구성되어야 합니다.
   - 영숫자 문자 또는 숫자로 시작하고 끝나야 합니다.
   - 영숫자 문자, 숫자, 마침표, 대시, 밑줄이 포함될 수 있습니다.
4. 생성을 선택합니다.

   Snapshotting... 상태로 방금 만든 스냅샷을 볼 수 있습니다.

   스냅샷이 완료된 후 이 스냅샷에서 다른 인스턴스를 생성할 수 있습니다. 예를 들어, 이전보다 더 큰 크기의 번들을 선택하고 싶을 수도 있습니다.

> **Important**
>
> 스냅샷에서 새 인스턴스를 생성하는 경우 Lightsail은 사용자가 같은 크기 또는 더 큰 크기의 인스턴스 번들만 생성하도록 합니다. 스냅샷에서 더 작은 크기의 인스턴스를 생성하는 것은 현재

지원되지 않습니다. 스냅샷에서 새 인스턴스를 생성할 때 더 작은 크기 옵션은 회색 음영으로 표시됩니다.

스냅샷에서 더 큰 인스턴스를 생성하려면 Lightsail 콘솔, create-instances-from-snapshot CLI 명령 또는 CreateInstancesFromSnapshot API 작업을 사용하면 됩니다. 자세한 내용은 스냅샷에서 인스턴스 생성을 참조하세요. Lightsail 번들에 대한 자세한 내용은 Lightsail 요금을 참조하세요.

Lightsail Windows Server 인스턴스의 스냅샷 생성

스냅샷은 시스템 디스크의 복사본이며, 인스턴스의 원본 구성입니다. 스냅샷에는 메모리, CPU, 디스크 크기, 데이터 전송 속도 등의 정보가 포함되어 있습니다. 자세한 내용은 스냅샷을 참조하세요.

Lightsail에서 Windows Server 인스턴스의 스냅샷을 생성하려면 먼저 백업 스냅샷을 만듭니다. 그런 다음 System Preparation(Sysprep)이라는 특수 유틸리티를 사용하여 두 번째 스냅샷을 만듭니다. Sysprep은 인스턴스가 스냅샷으로 백업될 수 있도록 Windows Server 설치를 생성합니다. 그러면 이 스냅샷으로 인스턴스를 생성할 때 처음으로 해당 Windows 인스턴스를 실행하는 것처럼 Out-of-Box Experience가 생깁니다.

Linux 또는 Unix 인스턴스의 스냅샷을 생성하려면 Linux 또는 Unix 인스턴스의 스냅샷 생성을 참조하세요.

목차

- 1단계: Sysprep을 실행하기 전에 백업 스냅샷 생성
- 2단계: Sysprep을 사용하여 인스턴스에 연결 및 종료
- 3단계: Sysprep 실행 후 스냅샷 생성

1단계: Sysprep을 실행하기 전에 백업 스냅샷 생성

Sysprep을 실행하여 스냅샷을 생성할 때는 시스템 관련 정보가 인스턴스에서 제거됩니다. 그래서 인스턴스에서 실행되는 애플리케이션에 의도하지 않은 결과가 생길 수 있습니다. 따라서 Sysprep을 사용하기 전에 먼저 백업 스냅샷을 생성하여 문제가 생길 경우를 위해 대체 스냅샷을 마련해야 합니다.

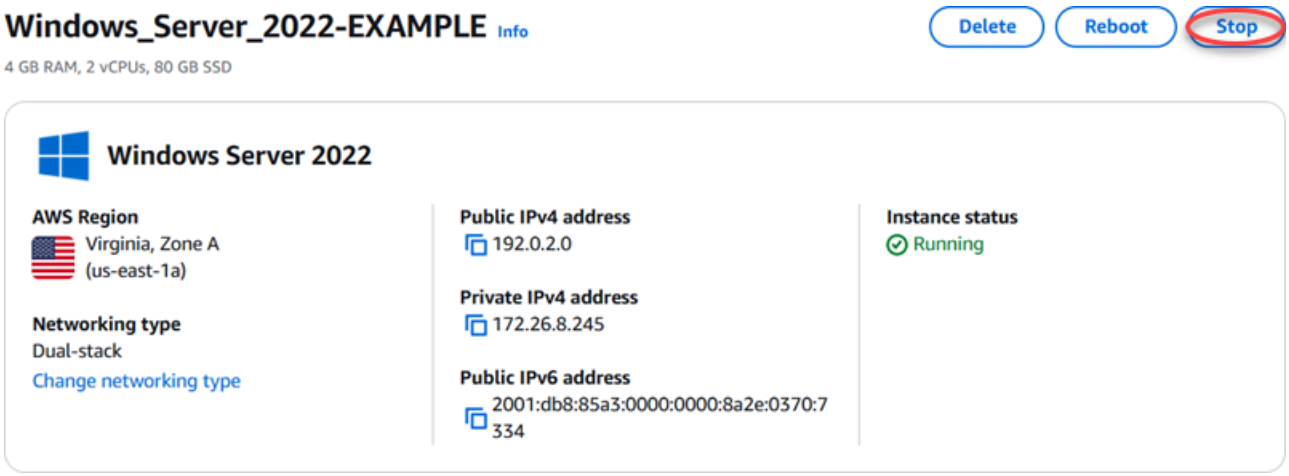
Sysprep을 실행하기 전 스냅샷을 생성할 때 백업 스냅샷을 사용하여 생성하는 인스턴스의 관리자 암호는 원래 인스턴스의 관리자 암호와 동일합니다. Lightsail 콘솔에서 브라우저 기반 RDP 클라이언트를 사용하여 해당 인스턴스에 연결할 수 없습니다. 하지만 자체 RDP 클라이언트 및 원본 인스턴스와 동일

한 관리자 암호를 사용하여 연결할 수는 있습니다. 자세한 내용은 Windows 컴퓨터에서 원격 데스크톱 연결 클라이언트를 사용하여 Amazon Lightsail에서 Windows 인스턴스에 연결을 참조하세요.

> **Important**
>
> 원본 Windows 인스턴스의 관리자 암호를 저장하고 안전한 장소에 보관하세요. 나중에 문제가 발생하는 경우 관리자 암호가 필요하며 Sysprep을 실행하기 전에 생성한 스냅샷에서 인스턴스를 생성합니다.

Sysprep을 실행하기 전에 백업 스냅샷을 생성하려면

1. Lightsail 콘솔에 로그인합니다.
2. Lightsail 홈 페이지에서 스냅샷을 생성할 Windows Server 인스턴스의 이름을 선택합니다.
3. 인스턴스 관리 페이지 위쪽의 중지를 선택하여 인스턴스를 중지합니다.



> **Note**
>
> 인스턴스를 중지하면 다시 시작할 때까지 모든 웹사이트 또는 서비스를 인스턴스에서 사용할 수 없습니다.

4. 스냅샷 탭을 선택합니다.
5. 페이지의 수동 스냅샷 섹션에서 스냅샷 생성을 선택한 다음 스냅샷 이름을 입력합니다.

   리소스 이름:

   - Lightsail 계정 AWS 리전 의 각 내에서 고유해야 합니다.
   - 2~255자의 문자로 구성되어야 합니다.

- 영숫자 문자 또는 숫자로 시작하고 끝나야 합니다.
- 영숫자 문자, 숫자, 마침표, 대시, 밑줄이 포함될 수 있습니다.

6. 생성(Create)을 선택합니다.
7. 프롬프트에서 스냅샷 생성을 다시 선택하여 확인합니다.

   스냅샷 프로세스는 완료할 때까지 몇 분 걸립니다.
8. 스냅샷이 생성된 후 인스턴스 관리 페이지 위쪽의 시작을 선택하여 인스턴스를 다시 시작합니다.

2단계: Sysprep을 사용하여 인스턴스에 연결 및 종료

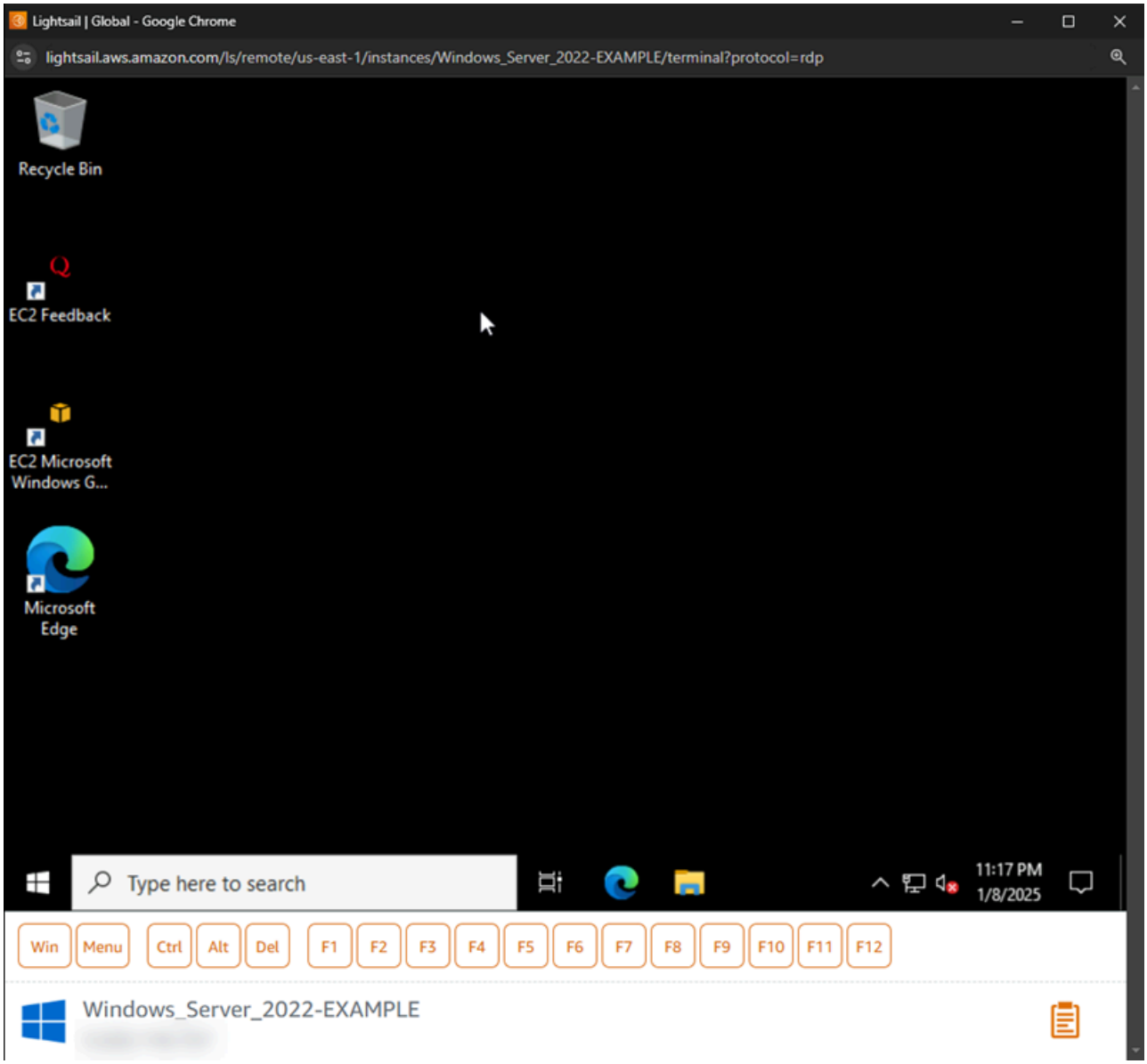
백업 스냅샷이 있으므로 이제 Windows Server 인스턴스에서 Sysprep을 실행합니다. 그러면 스냅샷을 만들 수 있도록 인스턴스가 종료됩니다. Sysprep에 대한 자세한 내용은 Microsoft 설명서의 Sysprep Overview를 참조하십시오.

이 단계에서는 미리 설치된 애플리케이션을 통해 인스턴스에 연결하고 Sysprep을 실행합니다. 애플리케이션이 Windows Server 2019 및 Windows Server 2016 인스턴스에서는 EC2LaunchSettings, Windows Server 2012에서는 Ec2ConfigService Settings라고 부릅니다.

인스턴스에 연결하고 Sysprep을 실행하려면

1. 인스턴스 관리 페이지에서 연결 탭을 선택한 후 RDP를 사용하여 연결을 선택합니다.

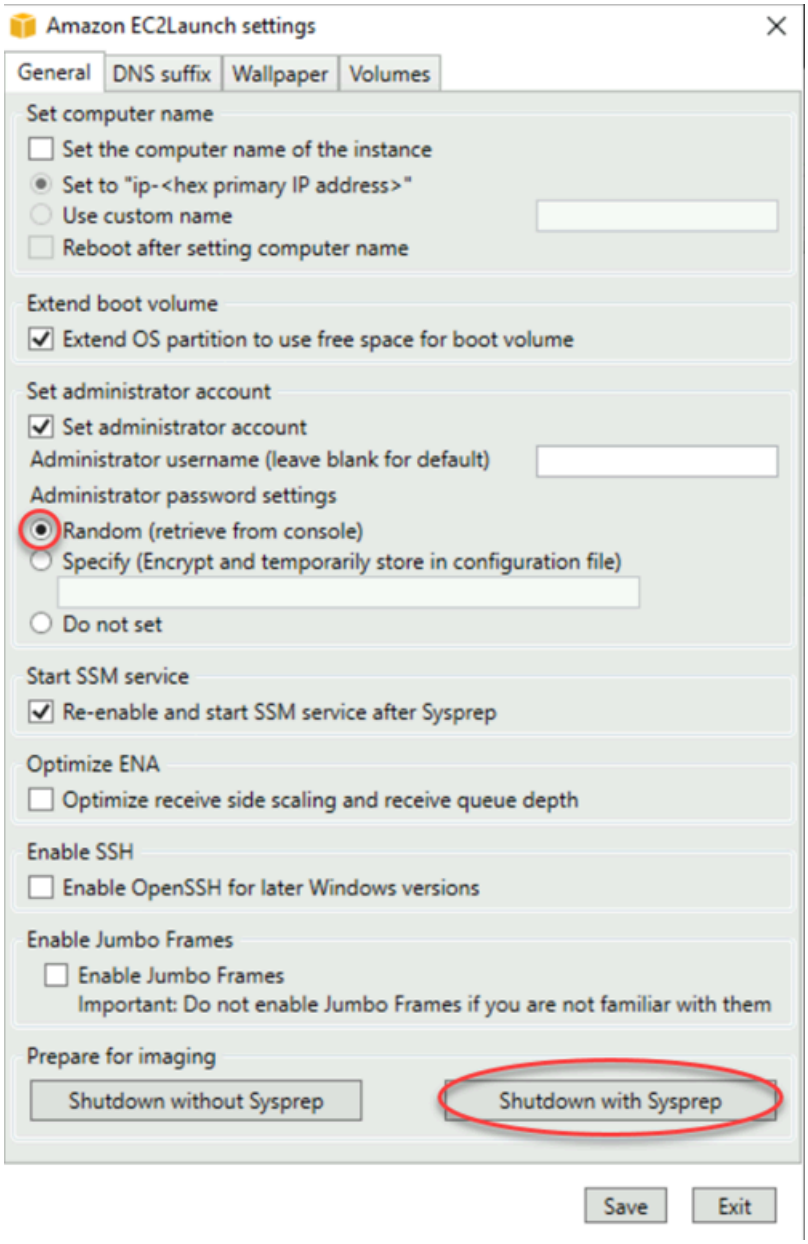
   브라우저 기반 RDP 창이 열리고 다음 예처럼 표시됩니다.



2. 작업 표시줄에서 Windows 아이콘을 선택하거나 Win을 선택하여 시작 메뉴를 엽니다.
3. 다음 옵션 중 하나를 선택합니다.

   - Windows Server 2022, Windows Server 2019 및 Windows Server 2016 인스턴스에서는 시작을 선택하고 Ec2LaunchSettings를 선택합니다.

4. 관리자 암호 섹션에서 Random (Retrieve from console)(임의(콘솔에서 검색))을 선택한 후 Shutdown with Sysprep(Sysprep을 이용해 종료)을 선택합니다.



5. 예를 실행하여 Sysprep을 실행하고 인스턴스를 종료하도록 확인합니다.

   인스턴스에서 Sysprep 실행을 시작하고 RDP 연결이 종료되며 몇 분 후 Lightsail 인스턴스 실행이 중지됩니다.

3단계: Sysprep 실행 후 스냅샷 생성

인스턴스가 중지 상태가 된 후 Lightsail 콘솔에서 스냅샷을 생성합니다. Sysprep 실행 후 Windows Server 인스턴스의 스냅샷을 생성할 때 스냅샷을 기반으로 만드는 모든 인스턴스는 고유한 관리자 암

호를 갖게 됩니다. Lightsail 콘솔에서 브라우저 기반 RDP 클라이언트를 사용하여 해당 인스턴스에 연결할 수 있습니다.

Lightsail 콘솔에서 스냅샷을 생성하려면

1. Lightsail 콘솔로 돌아갑니다.
2. Windows Server 인스턴스의 인스턴스 관리 페이지에서 스냅샷 탭을 선택합니다.
3. 페이지의 수동 스냅샷 섹션에서 스냅샷 생성을 선택한 다음 스냅샷 이름을 입력합니다.

   리소스 이름:

   - Lightsail 계정 AWS 리전 의 각 내에서 고유해야 합니다.
   - 2~255자의 문자로 구성되어야 합니다.
   - 영숫자 문자 또는 숫자로 시작하고 끝나야 합니다.
   - 영숫자 문자, 숫자, 마침표, 대시, 밑줄이 포함될 수 있습니다.
4. 생성을 선택합니다.
5. 프롬프트에서 스냅샷 생성을 선택하여 스냅샷의 인스턴스가 준비되었음을 확인합니다.

   스냅샷 프로세스는 완료할 때까지 몇 분 걸립니다.
6. 스냅샷이 생성된 후 인스턴스 관리 페이지 위쪽의 시작을 선택하여 인스턴스를 다시 시작합니다.

   이때 다음 예와 같이 Window Server 인스턴스의 스냅샷이 2개 있어야 합니다.

   Sysprep 스냅샷을 사용하여 새 인스턴스를 생성합니다. Sysprep 실행 후 원래 인스턴스가 예상대로 작동하지 않을 경우에만 백업 스냅샷을 사용합니다.

다음 단계

Sysprep 및 백업 스냅샷이 있으므로 몇 가지 다음 단계를 완료해야 합니다.

- 원래 인스턴스에 연결하고 Sysprep 실행 후 인스턴스의 애플리케이션이 예상대로 작동하는 것을 확인합니다. 자세한 내용은 Amazon Lightsail을 사용해 Windows Server 인스턴스에 연결을 참조하십시오.

- Sysprep 스냅샷을 사용하여 새 인스턴스를 생성하고 이 인스턴스에 연결한 후 새 인스턴스의 애플리케이션이 예상대로 작동하는 것을 확인합니다. 자세한 내용은 스냅샷에서 인스턴스 생성을 참조하세요.
- Sysprep을 실행한 후 원본 인스턴스가 예상대로 작동하는지 확인하고 백업 스냅샷을 삭제합니다. 자세한 내용은 스냅샷 삭제를 참조하세요.
- Sysprep을 실행한 후 인스턴스가 예상대로 작동하지 않으면 스냅샷에서 인스턴스 생성의 단계를 따라 백업 스냅샷에서 새 인스턴스를 생성합니다.

백업 또는 기준에 대한 Lightsail 블록 스토리지 디스크 스냅샷 생성

추가로 연결한 블록 스토리지 디스크의 백업을 위해 Amazon Lightsail에서 디스크 스냅샷을 생성할 수 있습니다.

이러한 디스크 스냅샷을 새 디스크 또는 데이터 백업의 기준으로 삼을 수 있습니다. 정기적으로 디스크의 스냅샷을 만드는 경우, 스냅샷은 증분식으로 늘어납니다. 새 스냅샷에는 마지막 스냅샷 이후로 변경된 디바이스 블록만 저장됩니다. 스냅샷은 증분식으로 저장되며, 스냅샷 삭제 프로세스는 전체 디스크 복원을 위해 가장 최근의 스냅샷만 남기도록 설계되어 있습니다.

자세한 내용은 스냅샷을 참조하세요.

1. 왼쪽 탐색 창에서 스토리지를 선택합니다.
2. 스냅샷을 생성하려는 블록 스토리지 디스크의 이름을 선택합니다.
3. 스냅샷 탭을 선택합니다.
4. 페이지의 수동 스냅샷 섹션에서 스냅샷 생성을 선택한 다음 스냅샷 이름을 입력합니다.

   리소스 이름:

   - Lightsail 계정 AWS 리전 의 각 내에서 고유해야 합니다.
   - 2~255자의 문자로 구성되어야 합니다.
   - 영숫자 문자 또는 숫자로 시작하고 끝나야 합니다.
   - 영숫자 문자, 숫자, 마침표, 대시, 밑줄이 포함될 수 있습니다.
5. 생성을 선택합니다.

   Snapshotting... 상태로 방금 만든 스냅샷을 볼 수 있습니다.

   스냅샷이 완료된 후 이 스냅샷에서 다른 인스턴스를 생성할 수 있습니다.

Lightsail의 스냅샷에서 블록 스토리지 디스크 생성

디스크 스냅샷에서 새로운 블록 스토리지 디스크를 만들 수 있습니다. 완전히 새로운 디스크를 만들려면 추가 블록 스토리지 디스크 만들기(Linux/Unix) 또는 블록 스토리지 디스크 생성 및 Windows Server 인스턴스에 연결을 참조하세요.

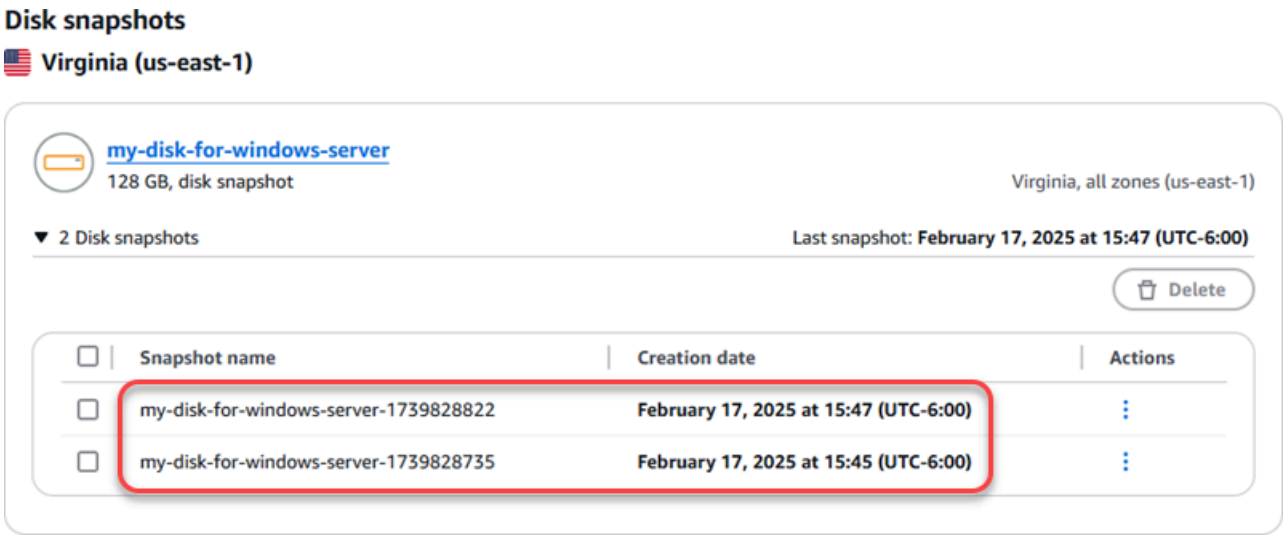
블록 스토리지 디스크의 스냅샷을 새 디스크 또는 데이터 백업의 기준으로 삼을 수 있습니다. 정기적으로 디스크의 스냅샷을 만드는 경우, 스냅샷은 증분식으로 늘어납니다. 새 스냅샷에는 마지막 스냅샷 이후로 변경된 디스크의 블록만 저장됩니다. 스냅샷은 증분식으로 저장되며, 스냅샷 삭제 프로세스는 전체 디스크 복원을 위해 가장 최근의 스냅샷만 남기도록 설계되어 있습니다. 블록 스토리지 디스크의 스냅샷을 만들려면 블록 스토리지 디스크 스냅샷 생성을 참조하세요.

1단계: 디스크 스냅샷을 찾아서 새 디스크 만들기

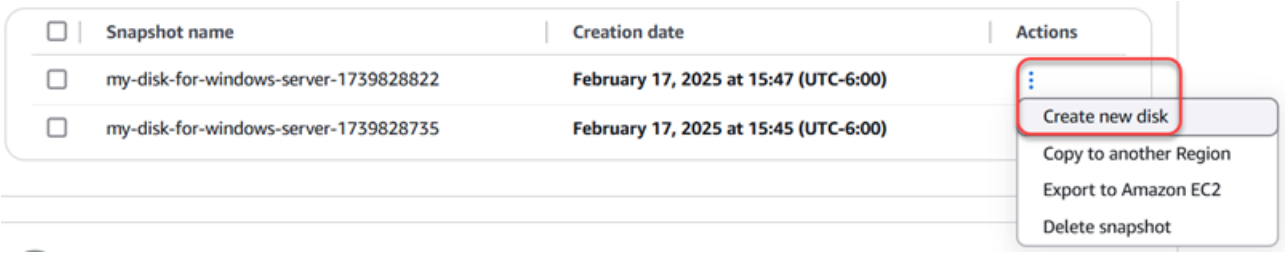
Lightsail의 두 위치(Lightsail 홈 페이지의 스냅샷(Snapshots) 탭 또는 디스크 관리 페이지의 스냅샷(Snapshots) 탭) 중 하나에서 디스크 스냅샷을 사용하여 새 인스턴스를 생성할 수 있습니다.

Lightsail 홈 페이지에서

1. 왼쪽 탐색 창의 왼쪽 탐색 모음에서 스냅샷을 선택합니다.
2. 디스크 이름을 찾은 다음 아래 노드를 확장하여 해당 디스크의 모든 사용 가능한 스냅샷을 볼 수 있습니다.

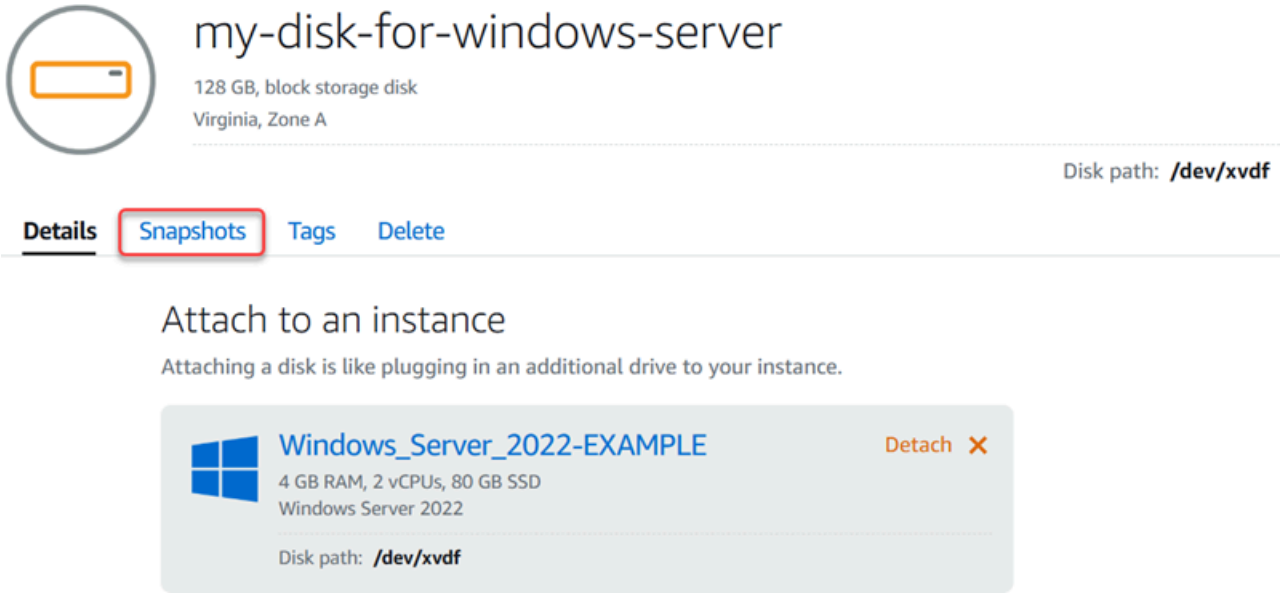


3. 새 디스크를 만들려는 스냅샷 옆에 있는 작업 메뉴 아이콘(⋮)을 선택한 후 새 디스크 생성을 선택합니다.

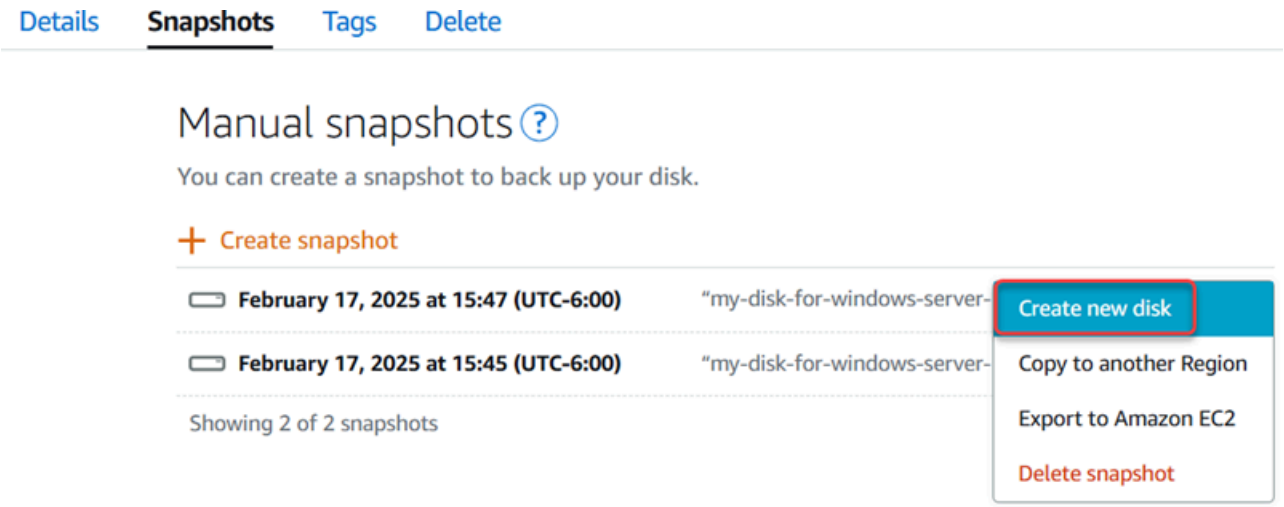


Lightsail의 디스크 관리 페이지에서

1. 왼쪽 탐색 창의 왼쪽 탐색 모음에서 스토리지 탭을 선택합니다.
2. 스냅샷을 보려는 디스크의 이름을 선택합니다.
3. 스냅샷 탭을 선택합니다.



4. 페이지의 수동 스냅샷 섹션에서 새 디스크를 생성할 스냅샷 옆에 있는 작업 메뉴 아이콘(⋮)을 선택한 후 새 디스크 생성을 선택합니다.



2단계: 디스크 스냅샷에서 새 디스크 만들기

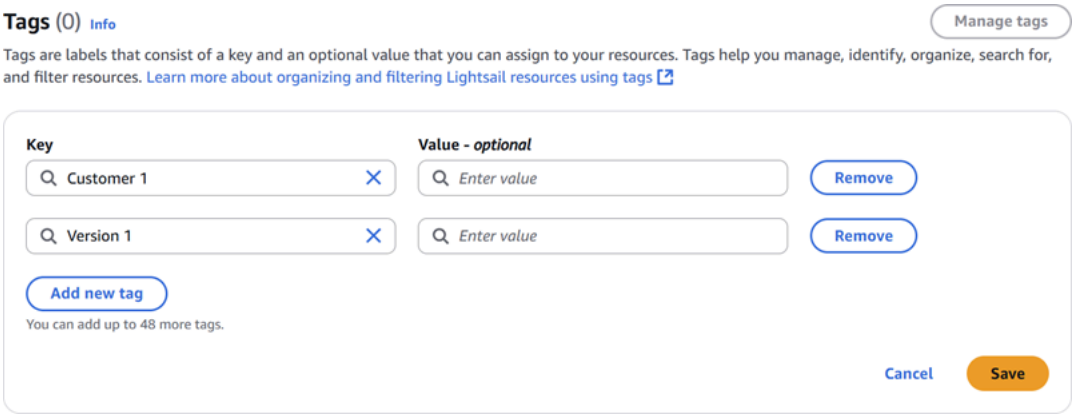
1. 새 디스크의 가용 영역을 선택하거나 기본값(us-east-2a)을 사용합니다.

   소스 디스크 AWS 리전 와 동일한에서 새 디스크를 생성해야 합니다.
2. 새 디스크의 크기는 소스 스냅샷과 같거나 커야 합니다.
3. 디스크의 이름을 입력합니다.

   리소스 이름:

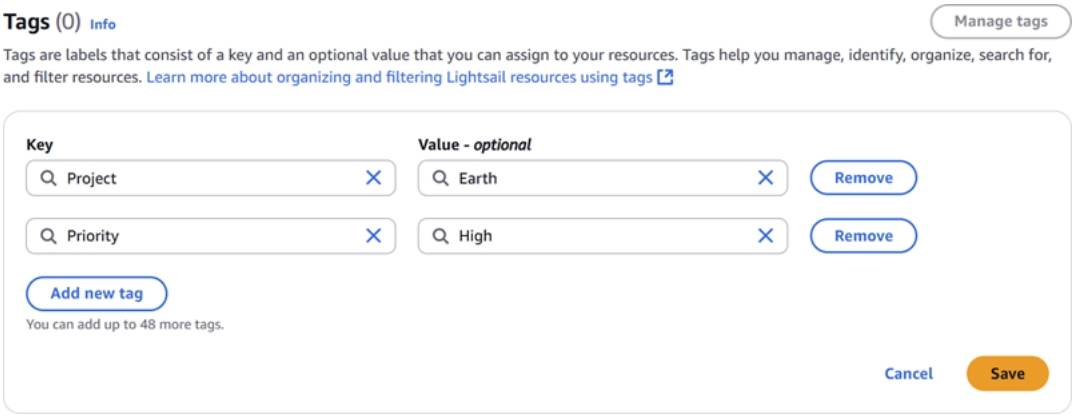
   - Lightsail 계정 AWS 리전 의 각 내에서 고유해야 합니다.
   - 2~255자의 문자로 구성되어야 합니다.
   - 영숫자 문자 또는 숫자로 시작하고 끝나야 합니다.
   - 영숫자 문자, 숫자, 마침표, 대시, 밑줄이 포함될 수 있습니다.
4. 다음 옵션 중 하나를 선택하여 디스크에 태그를 추가합니다.

   - 키 전용 태그를 추가하거나 태그를 관리(이미 태그를 추가한 경우)합니다. 새 태그를 태그 키 텍스트 상자에 입력하고 Enter를 누릅니다. 태그를 추가하려면 태그 입력이 완료될 때 저장을 선택하고, 태그를 추가하지 않으려면 취소를 선택합니다.



- 키-값 태그를 생성한 다음 키를 키 텍스트 상자에 입력하고, 값을 값 텍스트 상자에 입력합니다. 태그 입력이 완료되면 저장을 선택하고, 태그를 추가하지 않으려면 취소를 선택합니다.

  키-값 태그는 저장 전에 한 번에 하나씩만 추가할 수 있습니다. 둘 이상의 키-값 태그를 추가하려면 이전 단계를 반복하세요.



> **Note**
>
> 키 전용 태그 및 키-값 태그에 대한 자세한 내용은 태그를 참조하세요.

5. 디스크 생성을 선택합니다.

Lightsail 인스턴스의 루트 볼륨 스냅샷 생성

시스템 디스크의 스냅샷을 생성하여 Amazon Lightsail에서 인스턴스 루트 볼륨을 백업합니다. 그런 다음, 스냅샷에서 새로운 블록 스토리지 디스크를 생성하고 이를 또 다른 인스턴스에 연결하여 백업의 파일에 액세스합니다. 다음과 같은 작업이 필요한 경우에 이렇게 합니다.

- 문제가 발생한 인스턴스의 루트 볼륨에서 데이터를 복구합니다.
- 블록 스토리지 디스크에서와 마찬가지로 인스턴스의 루트 볼륨의 백업을 생성합니다.

AWS Command Line Interface (AWS CLI) 또는를 사용하여 인스턴스 루트 볼륨 스냅샷을 생성합니다 AWS CloudShell. 스냅샷을 생성한 후에는 Lightsail 콘솔을 사용해 스냅샷에서 블록 스토리지 디스크를 생성합니다. 그런 다음, 이를 실행 중인 인스턴스에 연결하고 해당 인스턴스에서 이를 액세스합니다.

목차

- 1단계: 필수 구성 요소 완성
- 2단계: 인스턴스 루트 볼륨 스냅샷 생성
- 3단계: 스냅샷에서 블록 스토리지 디스크를 생성하고 이를 인스턴스에 연결
- 4단계: 인스턴스에서 블록 스토리지 디스크 액세스

1단계: 필수 구성 요소 완성

AWS Command Line Interface (AWS CLI) 또는 AWS CloudShell 를 사용하여 인스턴스 루트 볼륨 스냅샷을 생성합니다. CloudShell은 브라우저 기반의 사전 인증된 셸로, Lightsail 콘솔에서 직접 실행할 수 있습니다. 자세한 내용은 Lightsail 작업을 AWS CLI 위한 설정 및 구성 및 를 사용하여 Lightsail 리소스 관리 AWS CloudShell 섹션을 참조하세요.

2단계: 인스턴스 루트 볼륨 스냅샷 생성

터미널, CloudShell 또는 명령 프롬프트 창을 열고 다음 명령을 입력하여 인스턴스 루트 볼륨 스냅샷을 생성합니다.

```
aws lightsail create-disk-snapshot --region AWSRegion --instance-name InstanceName --
disk-snapshot-name DiskSnapshotName
```

명령에서 다음과 같이 바꿉니다.

- 인스턴스 AWS 리전 의이 있는 *AWSRegion*.
- *InstanceName*을 인스턴스 루트 볼륨을 백업하려는 인스턴스의 이름으로 바꿉니다.
- *DiskSnapshotName*을 생성할 새 디스크 스냅샷의 이름으로 바꿉니다.

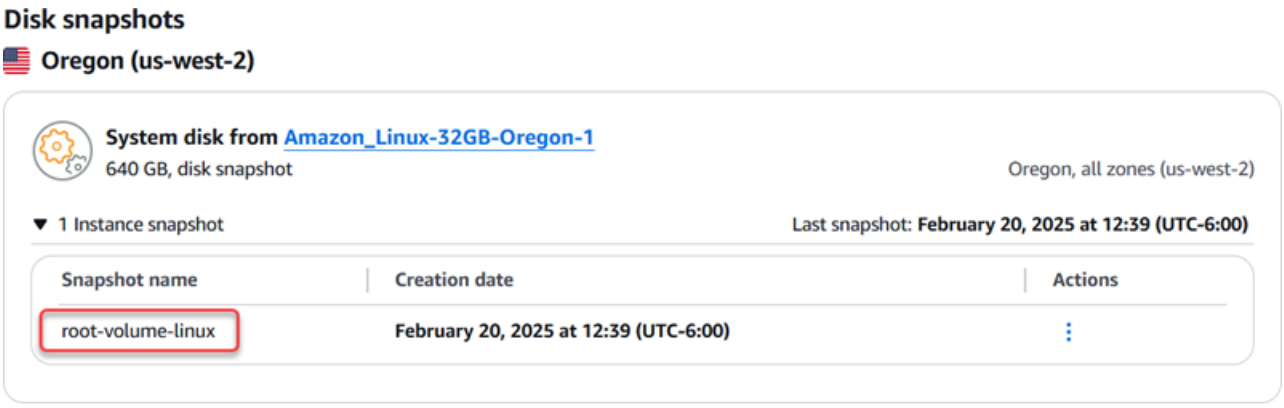
예:

```
aws lightsail create-disk-snapshot --region us-west-2 --instance-
name Amazon_Linux-32GB-Oregon-1 --disk-snapshot-name root-volume-linux
```

성공하면 다음과 비슷한 결과가 표시됩니다.

```
H:\>aws lightsail create-disk-snapshot --region us-west-2 --instance-name Amazon_Linux-32GB-Oregon-1
--disk-snapshot-name root-volume-linux

{
    "operations": [
        {
            "status": "Started",
            "resourceType": "DiskSnapshot",
            "isTerminal": false,
            "operationDetails": "Amazon_Linux-32GB-Oregon-1",
            "statusChangedAt": 1548799955.599,
            "location": {
                "availabilityZone": "all",
                "regionName": "us-west-2"
            },
            "operationType": "CreateDiskSnapshot",
            "resourceName": "root-volume-linux",
            "id": "[redacted]",
            "createdAt": 1548799955.599
        },
        {
            "status": "Started",
            "resourceType": "Instance",
            "isTerminal": false,
            "operationDetails": "root-volume-linux",
            "statusChangedAt": 1548799955.599,
            "location": {
                "availabilityZone": "us-west-2a",
                "regionName": "us-west-2"
            },
            "operationType": "CreateDiskSnapshot",
            "resourceName": "Amazon_Linux-32GB-Oregon-1",
            "id": "[redacted]",
            "createdAt": 1548799955.599
        }
    ]
}
```

스냅샷을 생성하는 데 몇 분 정도 걸릴 수 있습니다. 생성 후에는 다음 예와 같이 Lightsail 홈 페이지의 왼쪽 탐색 창에서 스냅샷 탭을 선택하고 디스크 스냅샷 섹션으로 스크롤하여 확인할 수 있습니다.



3단계: 스냅샷에서 블록 스토리지 디스크를 생성하고 이를 인스턴스에 연결

콘텐츠를 액세스해야 하는 경우에는 인스턴스 루트 볼륨 스냅샷에서 새로운 블록 스토리지 디스크를 생성하고 이를 또 다른 인스턴스에 연결합니다. 문제가 발생한 인스턴스의 루트 볼륨에서 데이터를 복구해야 하는 경우에는 이렇게 합니다.

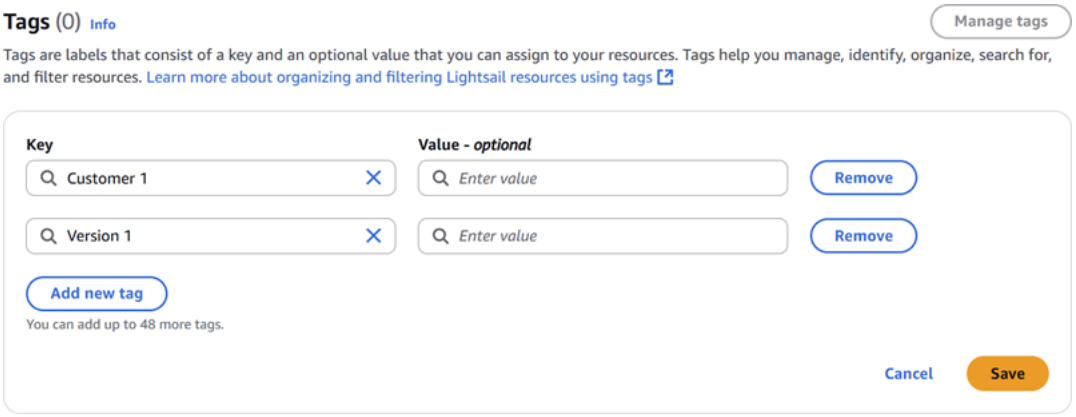
> **Note**
>
> 새 블록 스토리지 디스크는 소스 스냅샷 AWS 리전 과 동일한에 생성됩니다. 다른 리전에서 블록 스토리지 디스크를 생성하려면 스냅샷을 원하는 리전에 복사한 다음, 복사한 스냅샷에서 새 디스크를 생성합니다. 자세한 내용은 한에서 다른 AWS 리전 로 스냅샷 복사를 참조하세요.

1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 Snapshots을 선택합니다.
3. 사용하려는 루트 볼륨 디스크 스냅샷 옆에 표시되는 작업 메뉴 아이콘(:)을 선택한 다음, 새 디스크 생성을 선택합니다.
4. 해당 디스크의 가용 영역을 선택하거나 기본값을 사용합니다.
5. 디스크의 크기를 소스 스냅샷과 같거나 크게 선택합니다.
6. 디스크의 이름을 입력합니다.

   리소스 이름:

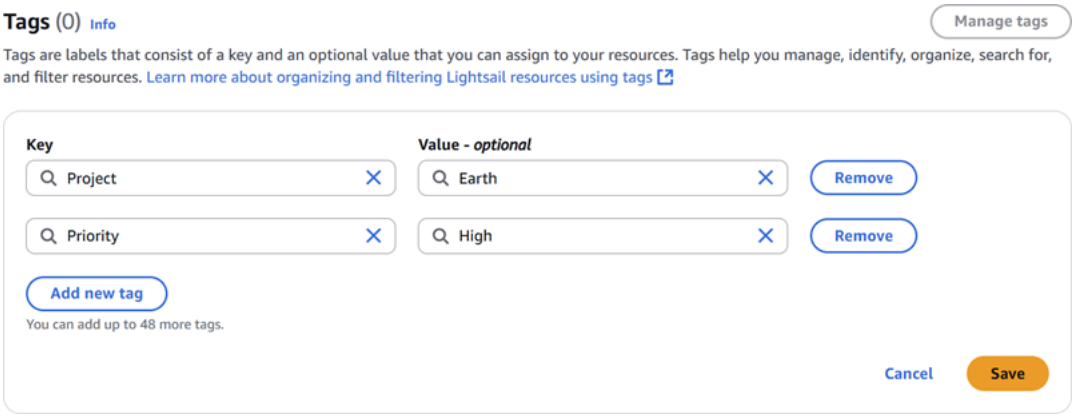
   - Lightsail 계정 AWS 리전 의 각 내에서 고유해야 합니다.
   - 2~255자의 문자로 구성되어야 합니다.
   - 영숫자 문자 또는 숫자로 시작하고 끝나야 합니다.
   - 영숫자 문자, 숫자, 마침표, 대시, 밑줄이 포함될 수 있습니다.
7. 다음 옵션 중 하나를 선택하여 디스크에 태그를 추가합니다.

   - 키 전용 태그를 추가하거나 태그를 관리(이미 태그를 추가한 경우)합니다. 새 태그를 태그 키 텍스트 상자에 입력하고 Enter를 누릅니다. 태그를 추가하려면 태그 입력이 완료될 때 저장을 선택하고, 태그를 추가하지 않으려면 취소를 선택합니다.



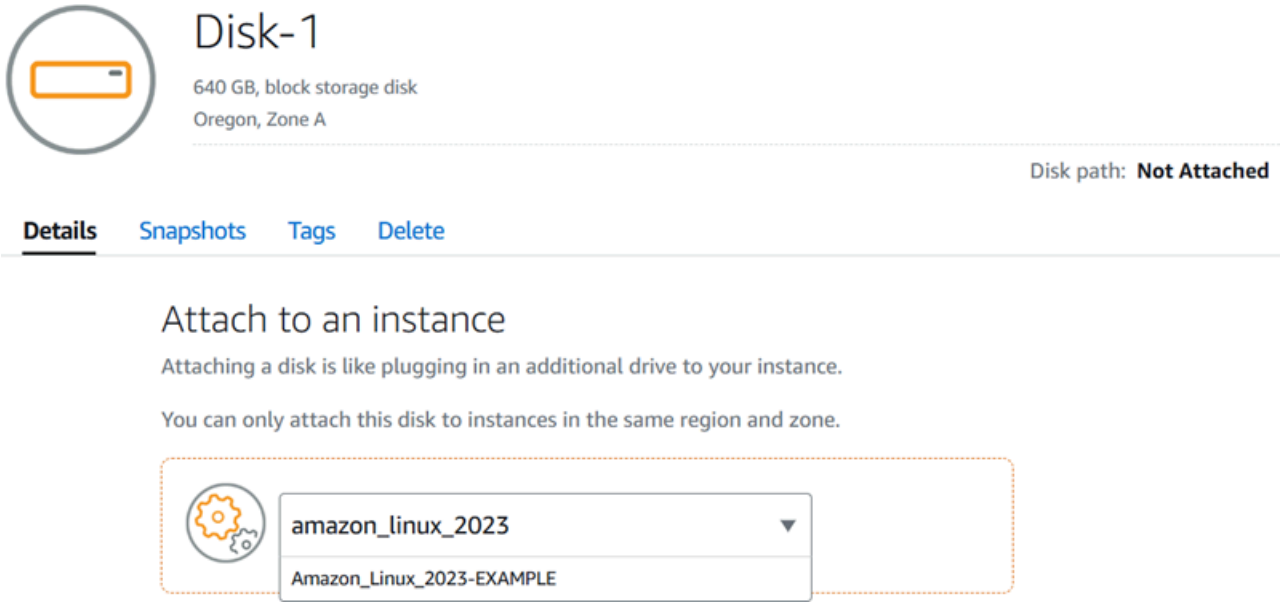
- 키-값 태그를 생성한 다음 키를 키 텍스트 상자에 입력하고, 값을 값 텍스트 상자에 입력합니다. 태그 입력이 완료되면 저장을 선택하고, 태그를 추가하지 않으려면 취소를 선택합니다.

  키-값 태그는 저장 전에 한 번에 하나씩만 추가할 수 있습니다. 둘 이상의 키-값 태그를 추가하려면 이전 단계를 반복하세요.



> **Note**
>
> 키 전용 태그 및 키-값 태그에 대한 자세한 내용은 태그를 참조하세요.

8. 디스크 생성을 선택합니다.
9. 디스크가 생성되고 나면 인스턴스 선택 드롭다운 메뉴에서 디스크를 연결할 인스턴스를 선택합니다. 방법은 다음 예제와 같습니다.



10. 연결을 선택하여 선택된 인스턴스에 디스크를 연결합니다.

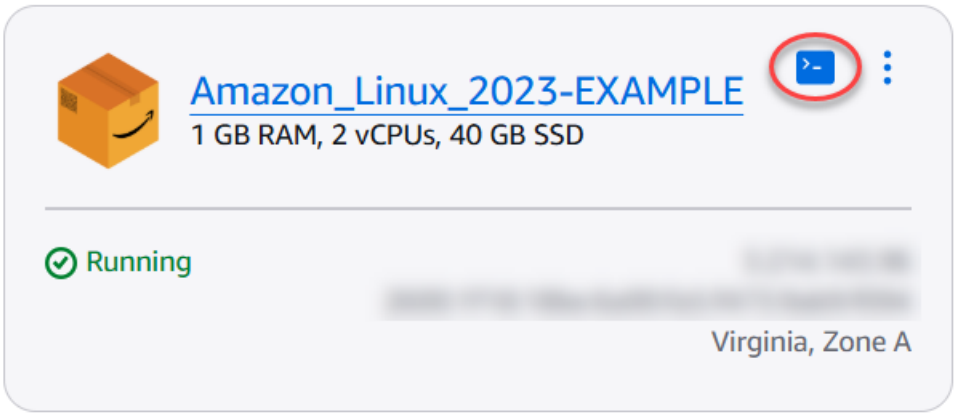
    이제 디스크가 인스턴스에 연결됩니다. 그러면 Linux에 이를 탑재하거나 Windows에서 온라인 상태로 만들어서 해당 운영 체제를 액세스가 가능하게 만들어야 합니다. 자세한 내용은 이 안내서의 인스턴스에서 블록 스토리지 액세스 단원을 참조하세요.

4단계: 인스턴스에서 블록 스토리지 디스크 액세스

블록 스토리지 디스크를 인스턴스에 연결한 이후 액세스하려면 이를 Linux 또는 Unix에 탑재하거나 Windows에서 온라인 상태로 만들어야 합니다.

Linux 또는 Unix 인스턴스에서 블록 스토리지 디스크 탑재 및 액세스

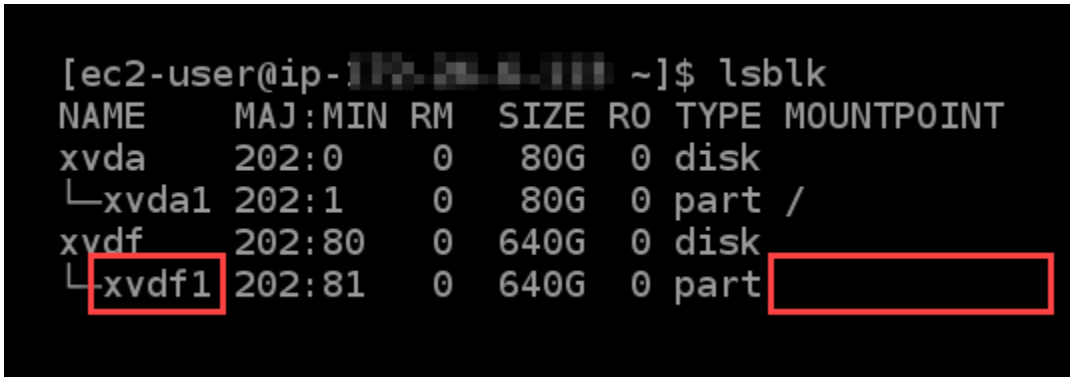
1. Lightsail 홈페이지에서 블록 스토리지 디스크를 연결한 Linux 또는 Unix 인스턴스의 브라우저 기반 SSH 클라이언트 아이콘을 선택합니다.



2. 브라우저 기반 SSH 클라이언트가 연결되고 나면 다음 명령을 입력하여 인스턴스에 연결된 블록 스토리지 디스크 장치를 확인합니다.

```
lsblk
```

다음 예제와 비슷한 결과가 나타나야 합니다. 이 예제에서 xvdf1은 탑재 지점이 없다는 점에서 아직 탑재되지 않은 인스턴스에 연결되는 블록 스토리지 디스크입니다. 또한 결과의 장치 이름에서 /dev/가 생략되므로 실제 장치 이름은 /dev/xvdf1입니다.



3. 다음 명령을 입력하여 블록 스토리지 디스크의 탑재 지점을 생성합니다.

```
sudo mkdir MountPoint
```

명령에서 *MountPoint*를 블록 스토리지 디스크가 탑재되어 액세스가 가능한 디렉터리의 이름으로 바꿉니다.

예:

```
sudo mkdir xvdf
```

4. 다음 명령을 입력하여 이전 단계에서 생성한 탑재 지점으로 블록 스토리지 디스크를 탑재합니다.

```
sudo mount /dev/DeviceName MountPoint
```

명령에서 다음과 같이 바꿉니다.

- *DeviceName*을 블록 스토리지 디스크 장치의 이름으로 바꿉니다.
- *MountPoint*를 이전 단계에서 생성한 탑재 지점 디렉터리로 바꿉니다.

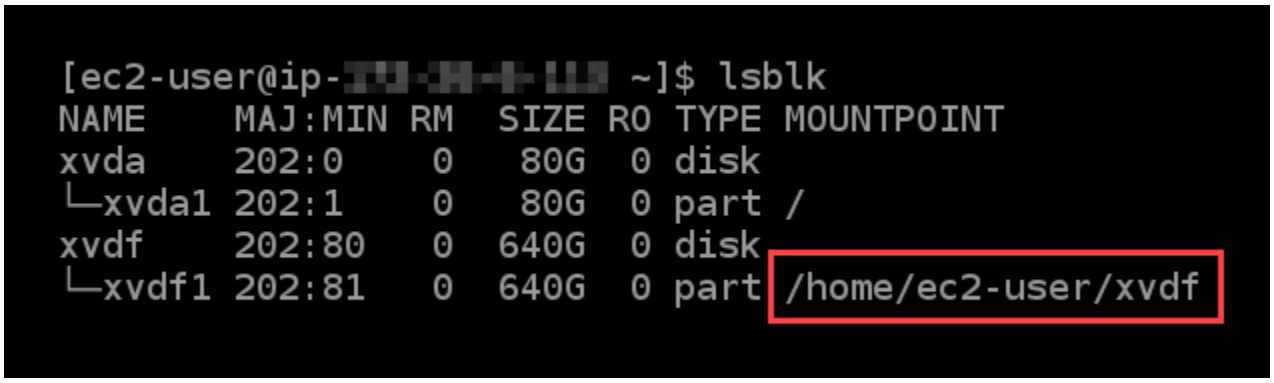
예:

```
sudo mount /dev/xvdf1 xvdf
```

5. 다음 명령을 입력하여 인스턴스에 연결된 블록 스토리지 디스크 장치를 확인합니다.

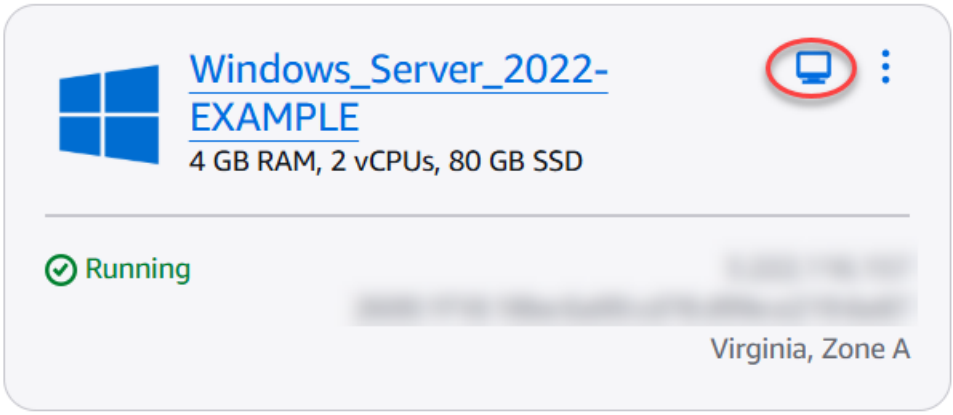
```
lsblk
```

다음 예제와 비슷한 결과가 나타나야 합니다. 이 예에서는 */home/ec2-user/xvdf* 디렉터리에 *xvdf1* 디바이스가 마운트되어 액세스할 수 있습니다. 이제 탑재 지점 디렉터리로 이동하여 블록 스토리지 디스크와 그 콘텐츠를 액세스할 수 있습니다.

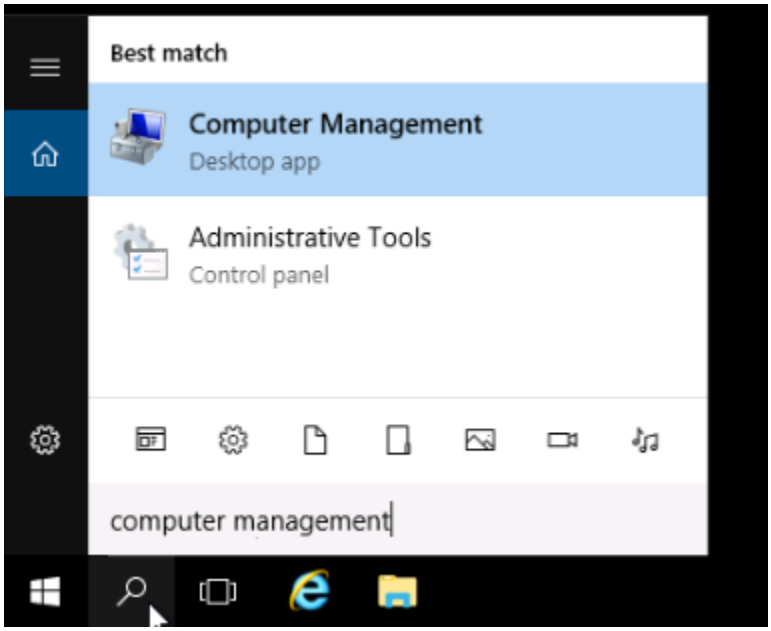


블록 스토리지 디스크를 온라인 상태로 만들고 Windows 인스턴스에서 이를 액세스합니다.

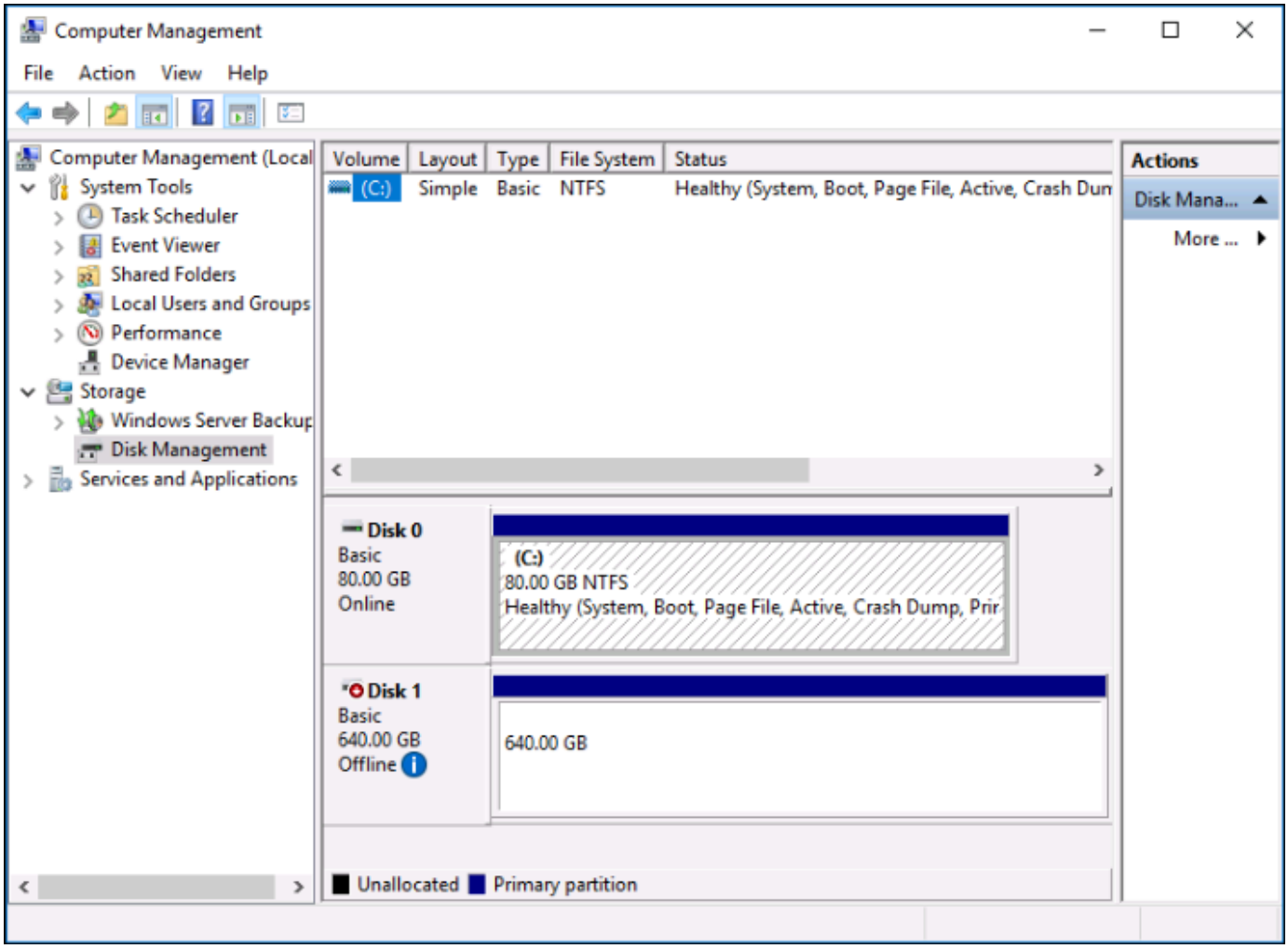
1. Lightsail 홈페이지에서 블록 스토리지 디스크를 연결한 Windows 인스턴스의 브라우저 기반 RDP 클라이언트 아이콘을 선택합니다.



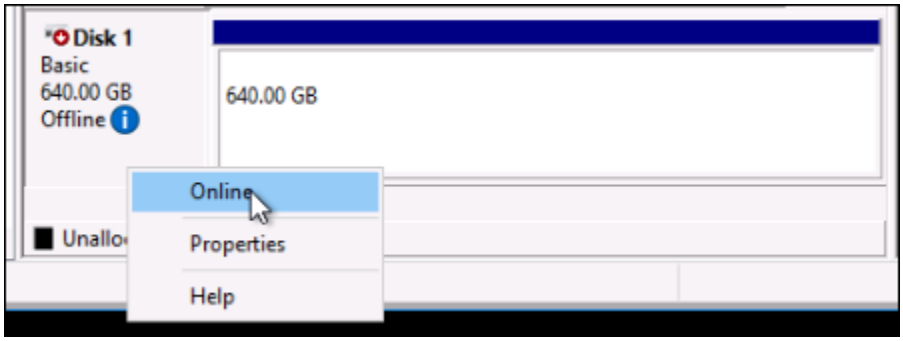
2. 브라우저 기반 SSH 클라이언트가 연결된 후에는 Windows 작업 표시줄에서 Computer Management(컴퓨터 관리)를 검색한 다음, 결과에서 Computer Management(컴퓨터 관리)를 선택합니다.



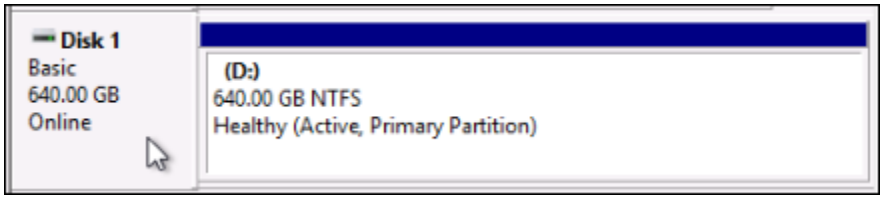
3. Computer Management(컴퓨터 관리) 콘솔의 왼쪽 탐색 창에서 다음 예제와 같이 Disk Management(디스크 관리)를 선택합니다.



4. 최근 인스턴스에 연결한 디스크를 찾습니다. 오프라인으로 레이블이 지정되어 있어야 합니다.
5. 오프라인 레이블을 마우스 오른쪽 버튼으로 클릭하고 온라인을 선택합니다.



디스크가 온라인으로 레이블이 지정되어 있어야 하며, 드라이브 문자가 여기에 연결되어 있어야 합니다. 이제 File Explorer를 열고 지정된 드라이브 문자를 검색하여 블록 스토리지 디스크와 그 콘텐츠에 액세스할 수 있습니다.

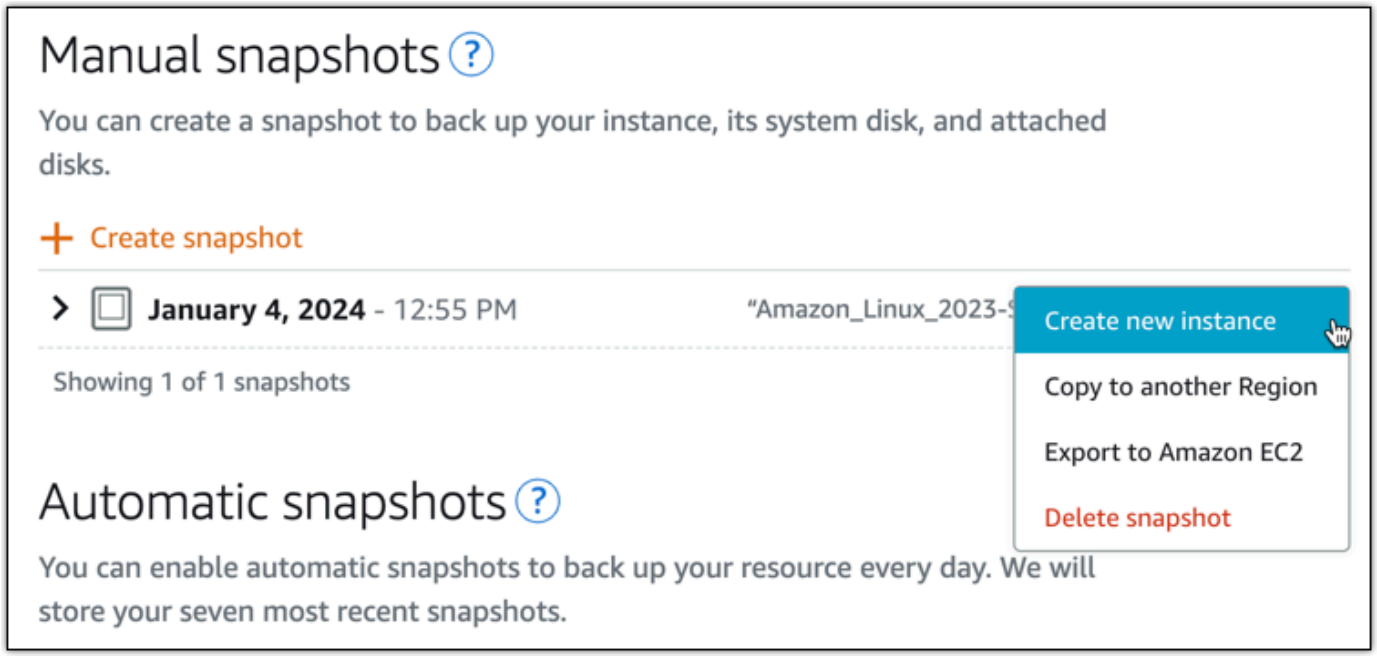


스냅샷에서 Lightsail 인스턴스 생성

Lightsail에서 스냅샷을 생성한 후 스냅샷으로 새 인스턴스를 생성할 수 있습니다. 인스턴스 크기 및 네트워킹 유형과 같은 새 인스턴스의 속성을 이중 스택 또는 IPv6 전용으로 변경할 수 있습니다. 새 인스턴스에는 시스템 디스크 및 사용자가 추가로 연결한 블록 스토리지 디스크를 포함합니다.

스냅샷으로 다른 인스턴스를 만들려면 먼저 인스턴스의 스냅샷이 있어야 합니다. 자세한 내용은 스냅샷으로 Linux/Unix Lightsail 인스턴스 백업 또는 Lightsail Windows Server 인스턴스의 스냅샷 생성을 참조하세요.

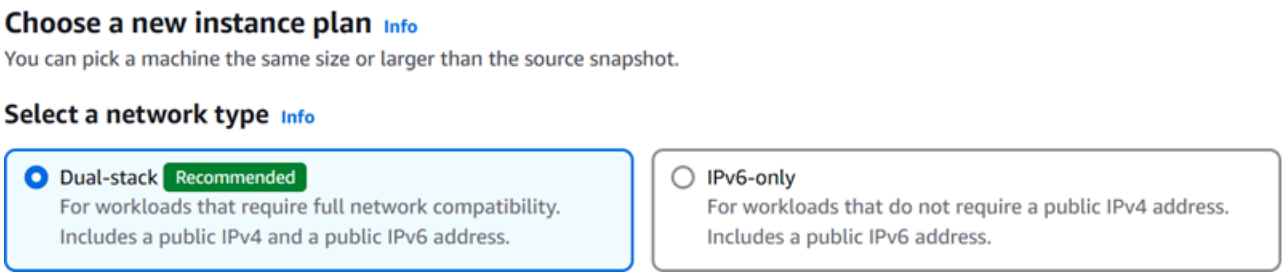
1. Lightsail 콘솔에서 스냅샷을 생성하려는 인스턴스를 선택하여 새 인스턴스를 생성합니다.
2. 스냅샷 탭을 선택합니다.
3. 수동 스냅샷 섹션에서 스냅샷 옆에 있는 작업 메뉴 아이콘(⋮)을 선택한 후 새 인스턴스 생성을 선택합니다.



4. 스냅샷에서 인스턴스 생성 페이지가 열립니다. 사용하려는 선택 설정을 선택합니다. 예를 들어, 가용 영역을 변경하거나, 시작 스크립트를 추가하거나 인스턴스에 연결하는 방법을 변경할 수 있습니다.

5. 새 인스턴스의 요금제(또는 번들)를 선택합니다. 이중 스택(IPv4 및 IPv6) 인스턴스 플랜 또는 IPv6 전용 플랜을 사용하는 인스턴스를 생성하도록 선택할 수 있습니다. 원래 인스턴스보다 더 큰 번들 크기를 선택할 수도 있습니다. IPv6 전용 인스턴스 플랜에 대한 자세한 내용은 Lightsail 인스턴스 IPv6 전용 네트워킹 구성 섹션을 참조하세요.

   > **Note**
   >
   > 원래 인스턴스보다 작은 번들 크기를 사용하는 인스턴스는 만들 수 없습니다.



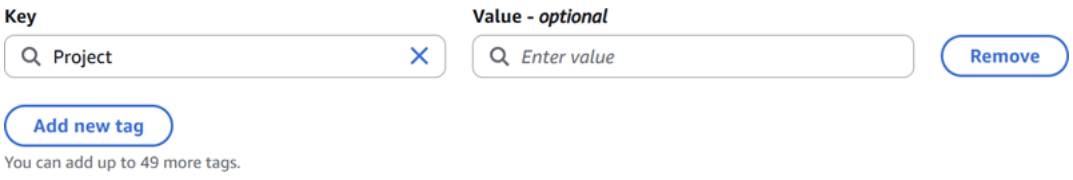
6. 인스턴스 이름을 입력합니다.

   리소스 이름:

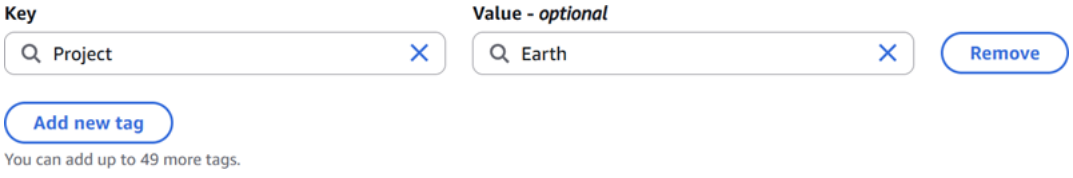
   - 각 AWS 리전 Lightsail 계정 내에서 고유해야 합니다.
   - 2~255자로 구성되어야 합니다.
   - 영숫자 문자로 시작하고 끝나야 합니다.
   - 영숫자 문자, 마침표, 대시, 밑줄이 포함될 수 있습니다.

7. (선택 사항) 인스턴스에 태그를 추가하려면 새 태그 추가를 선택합니다. 필요에 따라 이 단계를 반복하여 모든 태그를 추가합니다. 태그 사용에 대한 자세한 내용은 태그를 참조하세요.

   a. 키에는 태그 키를 입력합니다.



   b. (선택 사항) 값에 태그 값을 입력합니다.



8. 인스턴스 생성을 선택합니다.

   Lightsail이 새 인스턴스를 관리할 수 있는 관리 페이지를 엽니다.

   > **Important**
   >
   > 원래 인스턴스의 사용자 지정 방화벽 규칙은 스냅샷에서 생성한 새 인스턴스로 복사되지 않습니다. 기본 규칙만 새 인스턴스로 복사됩니다. 자세한 내용은 기본 인스턴스 방화벽 규칙을 참조하세요.

스냅샷에서 Lightsail 인스턴스, 스토리지 또는 데이터베이스 크기 조정

흔한 일이니 놀라지 마세요. 클라우드 프로젝트의 규모가 커지고 있어 지금 당장 더 많은 컴퓨팅 파워가 필요합니다. 저희가 이런 과제 수행을 도와드릴 수 있습니다. Lightsail 인스턴스, 블록 스토리지 디스크 또는 데이터베이스 크기를 늘리려면 리소스의 스냅샷을 생성한 후 스냅샷을 사용하여 리소스의 더 큰 버전을 새로 만듭니다.

> **Note**
>
> 원래 리소스보다 작은 플랜 크기를 사용하여 스냅샷으로 리소스를 만들 수는 없습니다. 예를 들어, 8GB 인스턴스에서 2GB 인스턴스로 갈 수 없습니다.
> 인스턴스를 생성할 때 인스턴스에 할당된 기본 퍼블릭 IPv4 주소는 인스턴스를 중지하고 시작할 때 변경됩니다. 고정 IPv4 주소를 생성하여 인스턴스에 연결할 수도 있습니다. 고정 IP 주소를 사용하면 주소를 계정의 다른 인스턴스에 신속하게 다시 매핑하여 인스턴스나 소프트웨어의 오류를 마스킹할 수 있습니다. 또는 도메인이 인스턴스를 가리키도록 도메인에 대한 DNS 레코드에 고정 IP 주소를 지정할 수 있습니다. 자세한 내용은 IP 주소를 참조하세요.

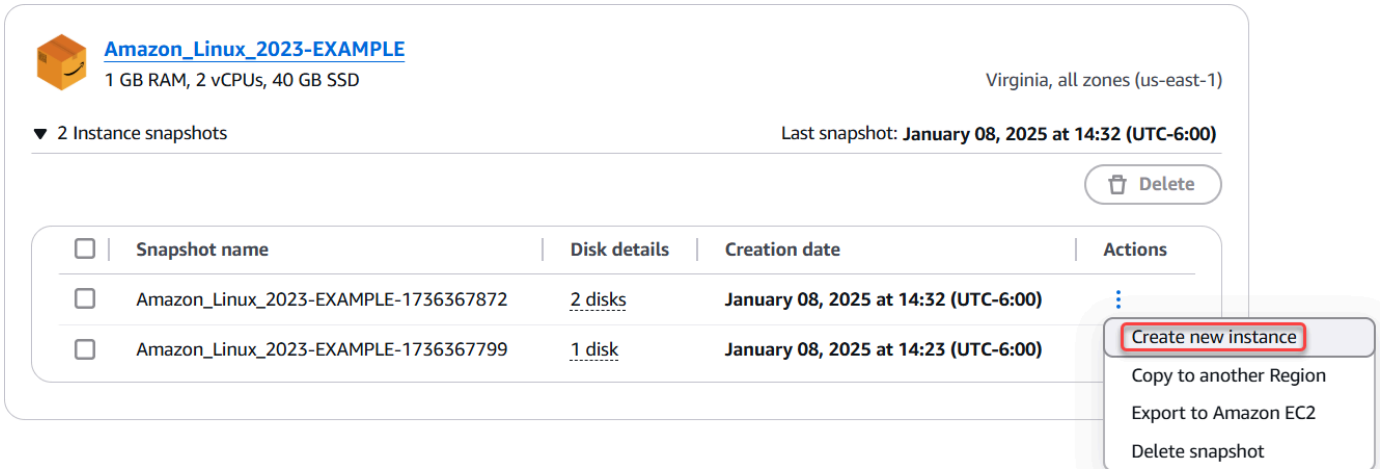
사전 조건

Lightsail 인스턴스, 블록 스토리지 디스크 또는 데이터베이스의 스냅샷이 필요합니다. 자세한 내용은 스냅샷을 참조하세요.

리소스 생성

1. Lightsail 콘솔에 로그인합니다.

2. 스냅샷 탭을 선택합니다.
3. 더 큰 리소스를 새로 만드는 데 사용할 스냅샷이 있는 Lightsail 리소스를 찾고, 오른쪽 화살표를 선택하여 스냅샷 목록을 확장합니다.
4. 사용할 스냅샷 옆의 줄임표 아이콘을 선택하고 새 인스턴스 생성을 선택합니다.



5. 생성 페이지에서 몇 가지 옵션 설정을 선택할 수 있습니다. 예를 들어, 가용 영역을 변경할 수 있습니다. 가령 시작 스크립트를 추가하거나 사용하는 SSH 키를 변경하여 연결할 수 있습니다.

   모든 기본값을 수락한 후 다음 단계로 이동할 수 있습니다.
6. 새 리소스의 플랜(또는 번들)을 선택합니다. 이때 원한다면 원래 리소스보다 더 큰 번들 크기를 선택할 수 있습니다.

> **Note**
>
> 원래 리소스보다 작은 플랜 크기를 사용하여 리소스를 만들 수는 없습니다. 원래 리소스보다 작은 번들 옵션은 사용할 수 없게 됩니다.

7. 인스턴스 이름을 입력합니다.

   리소스 이름:

   - Lightsail 계정 AWS 리전 의 각 내에서 고유해야 합니다.
   - 2~255자의 문자로 구성되어야 합니다.
   - 영숫자 문자 또는 숫자로 시작하고 끝나야 합니다.
   - 영숫자 문자, 숫자, 마침표, 대시, 밑줄이 포함될 수 있습니다.
8. 생성(Create)을 선택합니다.

Lightsail의 새 리소스 관리 페이지로 이동하여 관리를 시작할 수 있습니다.

를 사용하여 Lightsail 스냅샷에서 더 큰 인스턴스, 블록 스토리지 디스크 또는 데이터베이스 생성 AWS CLI

흔한 일이니 놀라지는 마세요. 클라우드 프로젝트의 규모가 커지고 있어 지금 당장 더 많은 컴퓨팅 파워가 필요합니다. 저희가 이런 과제 수행을 도와드릴 수 있습니다. Lightsail 콘솔 내에서 모든 작업을 수행하거나 AWS Command Line Interface (AWS CLI)를 사용하여 수행할 수 있습니다.

현재 Lightsail 인스턴스의 스냅샷을 생성하고 해당 스냅샷을 기반으로 필요한 컴퓨팅 성능을 사용하여 더 큰 새 인스턴스를 생성하는 방법을 보여드리겠습니다.

> **Note**
>
> 현재로서는 스냅샷에서 더 작은 크기의 인스턴스(또는 번들) 생성을 지원하지 않습니다. 같은 크기 또는 더 큰 인스턴스만 생성할 수 있습니다.

사전 조건

1. 먼저 아직 설치하지 않았다면를 설치해야 합니다 AWS CLI. 자세한 내용은 AWS Command Line Interface설치 섹션을 참조하세요. AWS CLI를 구성해야 합니다.
2. 작업할 인스턴스의 스냅샷도 필요합니다. 자세한 내용은 Linux 또는 Unix 인스턴스의 스냅샷 생성을 참조하세요.

1단계: 스냅샷 이름 가져오기

당연한 것처럼 보일 수도 있겠지만 더 큰 인스턴스를 생성하려면 이 AWS CLI 명령을 실행하기 전에 스냅샷 이름을 확보하고 있어야 합니다. 이름을 쉽게 확인할 수 있으므로 걱정하지 마세요.

1. 에 다음을 AWS CLI입력합니다.

   ```
   aws lightsail get-instance-snapshots
   ```

   다음과 유사한 출력 화면이 표시되어야 합니다.

```
{
    "instanceSnapshots": [
        {
            "fromInstanceName": "WordPress-512MB-EXAMPLE",
            "name": "WordPress-512MB-EXAMPLE-system-1234567891011",
            "sizeInGb": 20,
            "resourceType": "InstanceSnapshot",
            "fromInstanceArn":
            "arn:aws:lightsail:us-
east-1:123456789101:Instance/86f49ee4-26cc-4802-9b0d-12345EXAMPLE",
            "state": "available",
            "arn": "arn:aws:lightsail:us-east-1:123456789101:InstanceSnapshot/
c87acb5f-851e-4fbc-94f1-12345EXAMPLE",
            "fromBundleId": "nano_1_0",
            "fromBlueprintId": "wordpress_4_6_1",
            "createdAt": 1480898073.653,
            "location": {
                "availabilityZone": "all",
                "regionName": "us-east-2"
            }
        }
    ]
}
```

2. 이름 값을 나중에 가져올 수 있는 위치로 복사합니다. 이 값은 AWS CLI 명령에 사용할 `--instance-snapshot-name` 값입니다.

2단계: 번들 선택

번들은 실제로는 인스턴스에 대한 구성과 요금제일 뿐입니다. 예를 들어 중간 Linux 기반 번들의 요금은 매달 $24 USD이고 4.0GB RAM, 80GB SSD 스토리지 등의 조건이 있습니다.

더 작은 번들로 시작하고 컴퓨팅 파워가 더 많이 필요할 경우 더 큰 번들로 업그레이드할 수 있습니다. 자세한 내용은 스냅샷으로 더 큰 인스턴스, 블록 스토리지 디스크 또는 데이터베이스 생성을 참조하세요.

> **Important**
>
> 스냅샷에서 더 작은 번들로 크기를 조정할 수 없습니다. 더 작은 번들을 생성하려면 다시 시작해야 합니다.

1. 다음 AWS CLI 명령을 입력합니다.

```
aws lightsail get-bundles
```

다음과 유사하게 출력되어야 합니다.


```
{
    "bundles": [
        {
            "price": 5.0,
            "cpuCount": 2,
            "diskSizeInGb": 20,
            "bundleId": "nano_3_0",
            "instanceType": "nano",
            "isActive": true,
            "name": "Nano",
            "power": 298,
            "ramSizeInGb": 0.5,
            "transferPerMonthInGb": 1024,
            "supportedPlatforms": [
                "LINUX_UNIX"
            ],
            },
        {
            "price": 7.0,
            "cpuCount": 2,
            "diskSizeInGb": 40,
            "bundleId": "micro_3_0",
            "instanceType": "micro",
            "isActive": true,
            "name": "Micro",
            "power": 500,
            "ramSizeInGb": 1.0,
            "transferPerMonthInGb": 2048,
            "supportedPlatforms": [
                "LINUX_UNIX"
            ],
            },
        {
            "price": 12.0,
            "cpuCount": 2,
            "diskSizeInGb": 60,
            "bundleId": "small_3_0",
```

```
            "instanceType": "small",
            "isActive": true,
            "name": "Small",
            "power": 1000,
            "ramSizeInGb": 2.0,
            "transferPerMonthInGb": 3072,
            "supportedPlatforms": [
                "LINUX_UNIX"
            ],
            },
        {
            "price": 24.0,
            "cpuCount": 2,
            "diskSizeInGb": 80,
            "bundleId": "medium_3_0",
            "instanceType": "medium",
            "isActive": true,
            "name": "Medium",
            "power": 2000,
            "ramSizeInGb": 4.0,
            "transferPerMonthInGb": 4096,
            "supportedPlatforms": [
                "LINUX_UNIX"
            ],
            },
        {
            "price": 44.0,
            "cpuCount": 2,
            "diskSizeInGb": 160,
            "bundleId": "large_3_0",
            "instanceType": "large",
            "isActive": true,
            "name": "Large",
            "power": 3000,
            "ramSizeInGb": 8.0,
            "transferPerMonthInGb": 5120,
            "supportedPlatforms": [
                "LINUX_UNIX"
            ],
            },
    ]
}
```


2. 원하는 번들의 bundleId 값을 찾습니다. 자세한 내용은 Lightsail요금을 참조하세요.

3단계: AWS CLI 명령 작성 및 새 인스턴스 생성

이제 파라미터 값을 알고 있으므로 명령을 작성하고 실행하여 인스턴스를 생성할 준비가 완료되었습니다.

1. 다음을 입력합니다.

```
aws lightsail create-instances-from-snapshot --instance-names
 MyNewInstanceFromSnapshot --availability-zone us-east-1a --instance-snapshot-name
 WordPress-512MB-EXAMPLE-system-1234567891011 --bundle-id medium_1_0
```

   다음과 유사하게 출력되어야 합니다.

```
{
    "operations": [
        {
            "status": "Started",
            "resourceType": "Instance",
            "isTerminal": false,
            "statusChangedAt": 1486863990.961,
            "location": {
                "availabilityZone": "us-east-2a",
                "regionName": "us-east-2"
            },
            "operationType": "CreateInstance",
            "resourceName": "MyNewInstanceFromSnapshot",
            "id": "30fec45e-e7d7-4e18-96c8-12345EXAMPLE",
            "createdAt": 1486863989.784
        }
    ]
}
```

> **Note**
>
> 를 사용하여 리전 및 가용 영역 목록을 반환할 수도 있습니다 AWS CLI. `aws lightsail get-regions --include-availability-zones`를 입력하여 `get-regions` 요청으로 가용 영역의 목록을 반환합니다.

2. Lightsail 콘솔에서 새 인스턴스를 열어 인스턴스 수정을 시작합니다.

다음 단계

스냅샷에서 새 인스턴스를 생성한 후 다음으로 수행할 수 있는 작업은 다음과 같습니다.

- 이전 인스턴스를 끝낸 경우 삭제할 수도 있습니다. Lightsail 콘솔 또는 delete-instance CLI 명령을 사용하여 삭제할 수 있습니다.
- 이전 스냅샷이 필요하지 않으면 삭제할 수도 있습니다. Lightsail 콘솔 또는 delete-instance-snapshot CLI 명령을 사용하여 삭제할 수 있습니다.
- 이전 인스턴스에 고정 IP 주소가 연결되어 있다면 이를 유지하고 새 인스턴스에 연결하고 싶을 것입니다. 이를 위해 콘솔을 사용할 수 있습니다. 고정 IP 생성 및 인스턴스에 연결을 참조하세요.

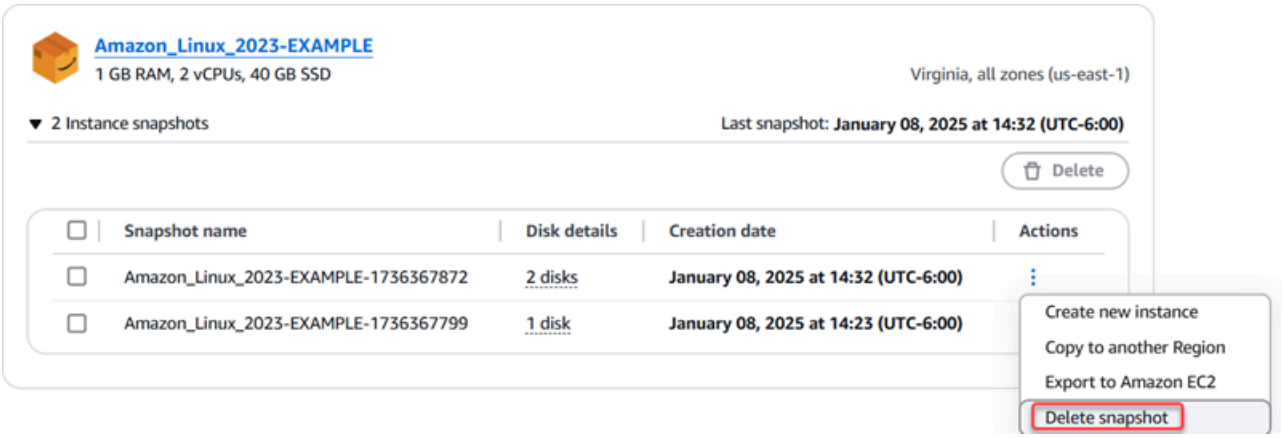
미사용 Lightsail 스냅샷을 삭제하여 월별 요금 방지

더 이상 필요가 없는 경우 인스턴스, 데이터베이스 및 디스크 스냅샷을 Amazon Lightsail에서 삭제하여 월 사용료가 부과되지 않게 합니다.

개별 스냅샷 삭제

> **Important**
>
> 이것은 영구적인 작업이며 실행 취소할 수 없습니다. 스냅샷을 삭제하면 스냅샷에 저장된 모든 데이터가 손실됩니다.

1. Lightsail 콘솔에서 스냅샷 탭을 선택합니다.
2. 삭제할 스냅샷이 있는 Lightsail 리소스를 찾고, 오른쪽 화살표를 선택하여 해당 리소스에 사용 가능한 스냅샷 목록을 확장합니다.
3. 삭제할 스냅샷 옆의 작업 메뉴 아이콘(⋮)을 선택하고 스냅샷 삭제를 선택합니다.

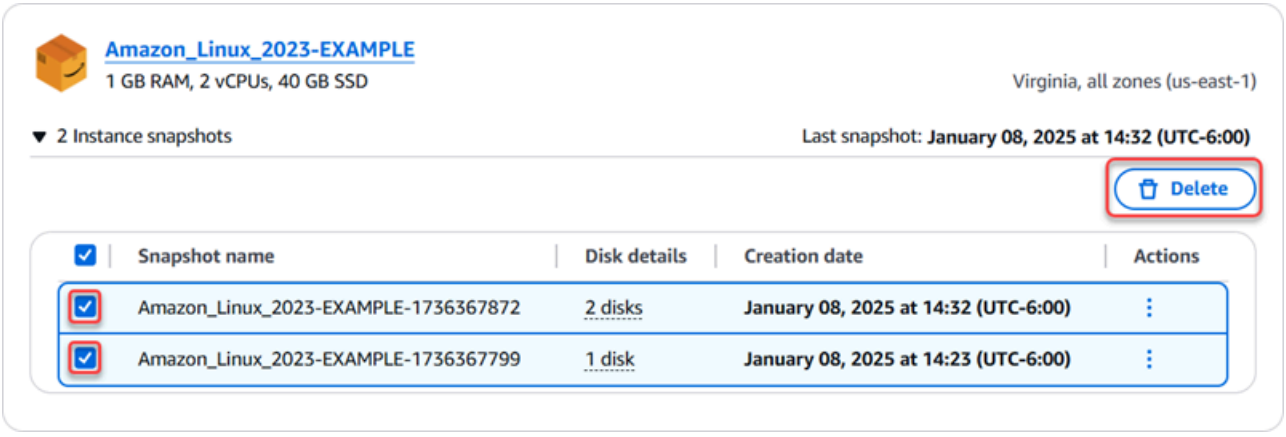


4. 예를 선택하여 스냅샷 삭제를 확인합니다.

스냅샷 여러 개 삭제

> **Important**
>
> 이것은 영구적인 작업이며 실행 취소할 수 없습니다. 스냅샷을 삭제하면 스냅샷에 저장된 모든 데이터가 손실됩니다.

1. Lightsail 홈 페이지에서 스냅샷을 선택합니다.
2. 삭제할 스냅샷이 있는 Lightsail 리소스를 찾고, 해당 리소스에 대한 스냅샷 섹션을 확장합니다.
3. 삭제할 리소스의 스냅샷을 선택한 다음 삭제를 선택합니다.



4. 예를 선택하여 스냅샷 삭제를 확인합니다.

에서 Lightsail 스냅샷 복사 AWS 리전

에서는 인스턴스 스냅샷을 복사하고 스토리지 디스크 스냅샷을 한 리전에서 다른 리전 AWS 리전 으로 또는 동일한 리전 내에서 차단할 Amazon Lightsail수 있습니다. 예를 들어, 한 리전에서 리소스를 생성 및 구성했지만, 나중에 다른 리전이 더 적합하다고 결정하는 경우 리전 간 스냅샷을 복사할 수 있습니다. 또는 여러 리전 간에 리소스를 복제하려고 할 수도 있습니다.

사전 조건

복사할 Lightsail 인스턴스 또는 블록 스토리지 디스크의 스냅샷을 생성합니다. 자세한 내용은 다음 안내서 중 하나를 참조하세요.

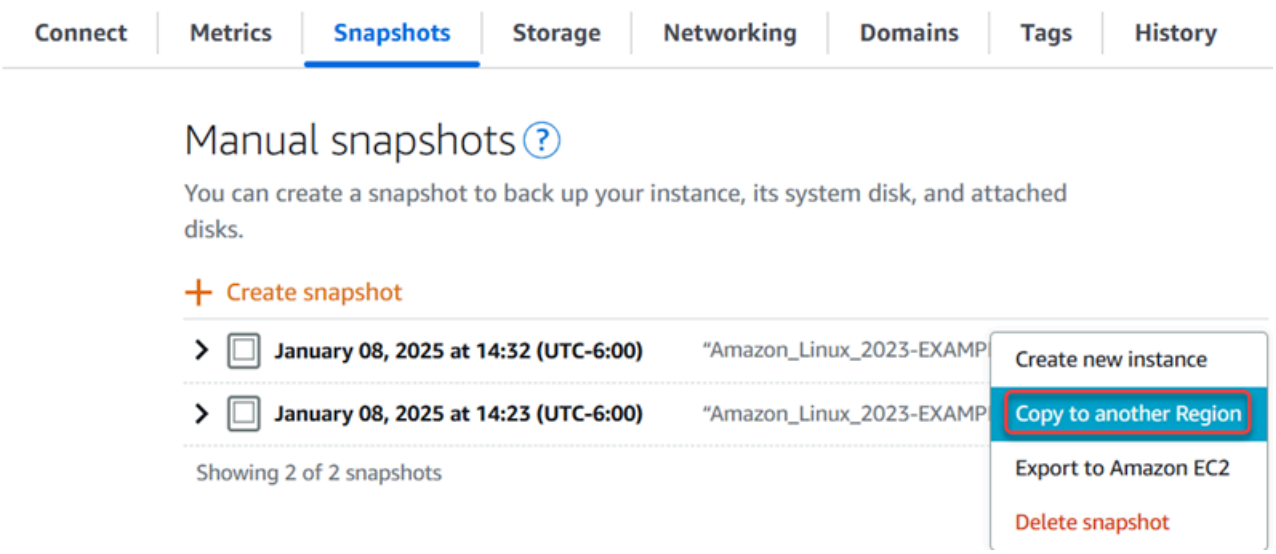
- Linux 또는 Unix 인스턴스의 스냅샷 생성
- Windows Server 인스턴스의 스냅샷 생성
- 블록 스토리지 디스크 스냅샷 생성

스냅샷 복사

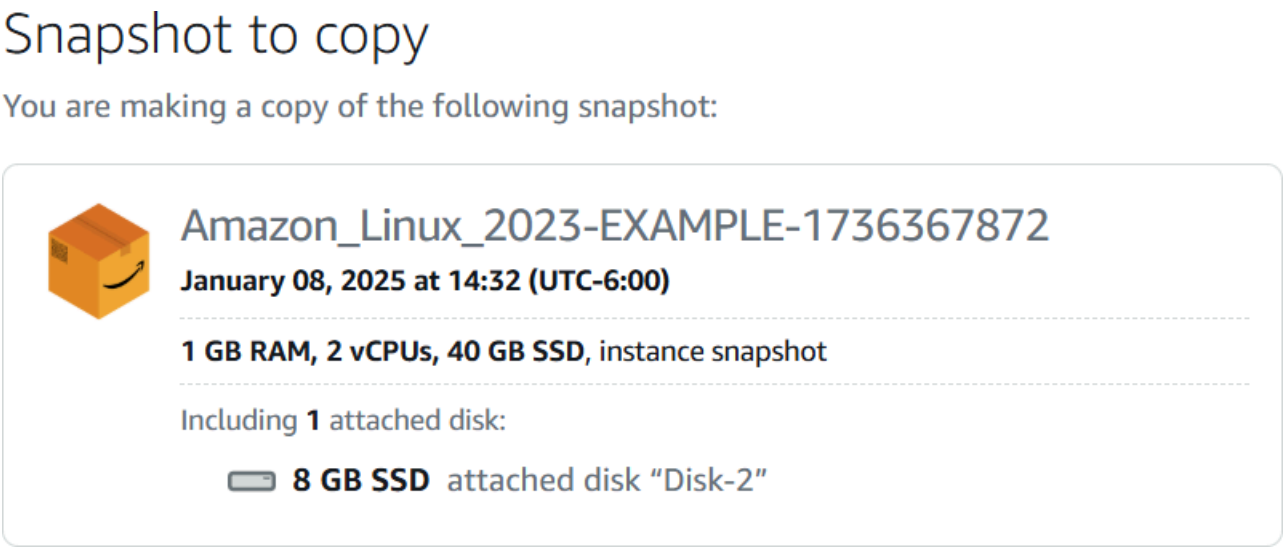
AWS 리전 리전 간 또는 동일한 리전 내에서 Lightsail 인스턴스 스냅샷 및 블록 스토리지 디스크 스냅샷을 복사할 수 있습니다.

Lightsail 스냅샷을 복사하려면

1. Lightsail 콘솔에 로그인합니다.
2. Lightsail 홈 페이지에서 스냅샷 탭을 선택합니다.
3. 복사할 인스턴스 또는 블록 스토리지 디스크를 찾고 노드를 확장하여 해당 리소스에 대해 사용 가능한 스냅샷을 확인합니다.
4. 원하는 스냅샷에 대해 작업 메뉴 아이콘(:)을 선택한 다음 Copy to another Region(다른 리전으로 복사)을 선택합니다.



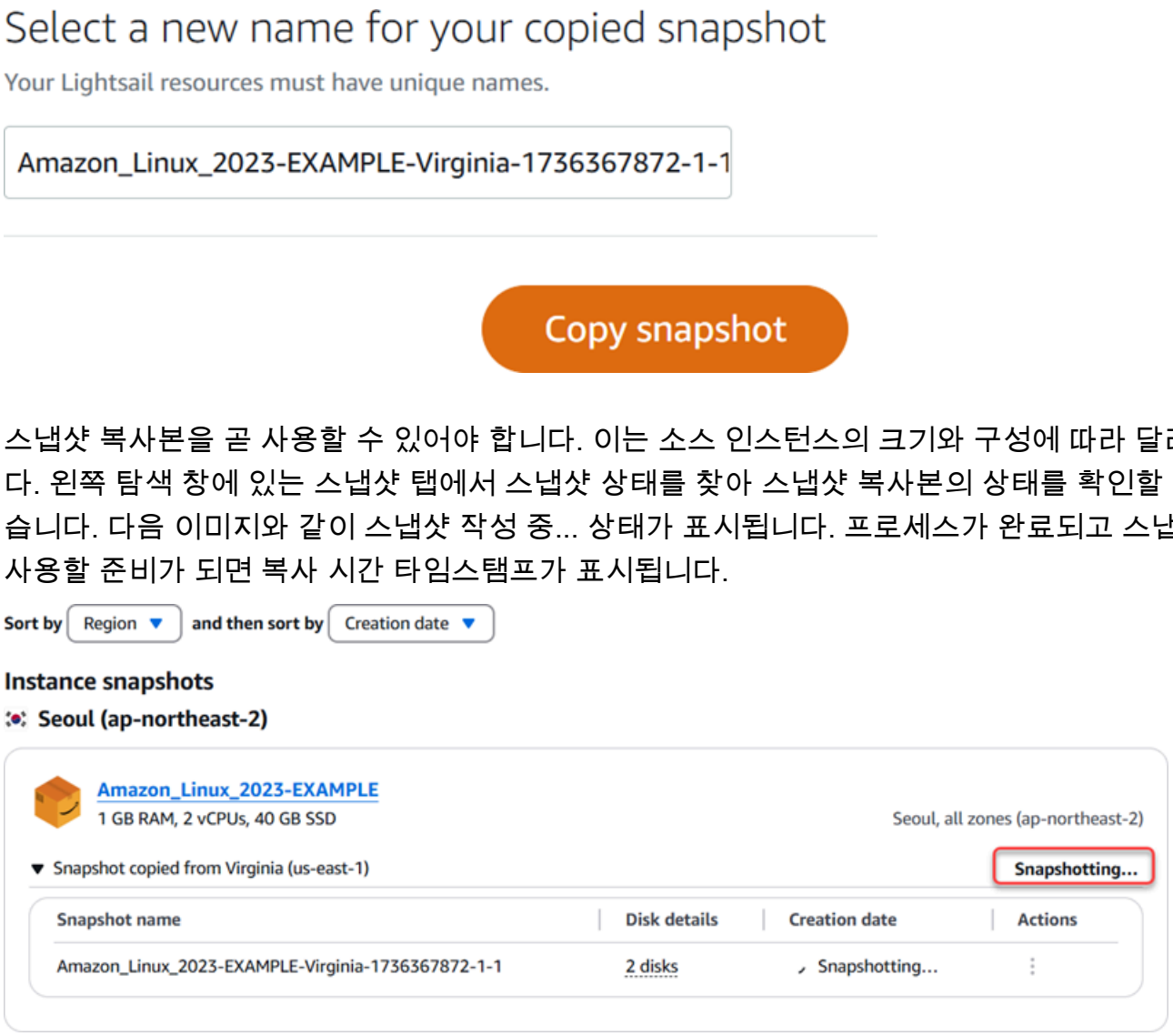
5. Copy a snapshot(스냅샷 복사) 페이지의 Snapshot to copy(복사할 스냅샷) 섹션에서 표시된 스냅샷 세부 정보가 소스 인스턴스 또는 블록 스토리지 디스크의 사양과 일치하는지 확인합니다.



6. 페이지의 Select a Region(리전 선택) 섹션에서 스냅샷 복사본에 대한 리전을 선택합니다.
7. 스냅샷 복사본 이름을 입력합니다.

   리소스 이름:

   - Lightsail 계정 AWS 리전 의 각 내에서 고유해야 합니다.
   - 2~255자의 문자로 구성되어야 합니다.
   - 영숫자 문자 또는 숫자로 시작하고 끝나야 합니다.
   - 영숫자 문자, 숫자, 마침표, 대시, 밑줄이 포함될 수 있습니다.
8. 스냅샷 복사를 선택합니다.



스냅샷 복사본을 곧 사용할 수 있어야 합니다. 이는 소스 인스턴스의 크기와 구성에 따라 달라집니다. 왼쪽 탐색 창에 있는 스냅샷 탭에서 스냅샷 상태를 찾아 스냅샷 복사본의 상태를 확인할 수 있습니다. 다음 이미지와 같이 스냅샷 작성 중... 상태가 표시됩니다. 프로세스가 완료되고 스냅샷이 사용할 준비가 되면 복사 시간 타임스탬프가 표시됩니다.

다음 단계

다음은 Lightsail에서 다른 리전으로 스냅샷을 복사한 후 수행할 수 있는 몇 가지 추가 단계입니다.

- 사용 가능한 상태가 된 후 복사된 스냅샷에서 새 인스턴스를 생성합니다. 자세한 내용은 스냅샷에서 인스턴스 생성을 참조하세요.
- 더 이상 필요하지 않는 경우 소스 스냅샷을 삭제하세요. 그렇지 않으면 스냅샷 저장에 대해 비용이 청구됩니다.

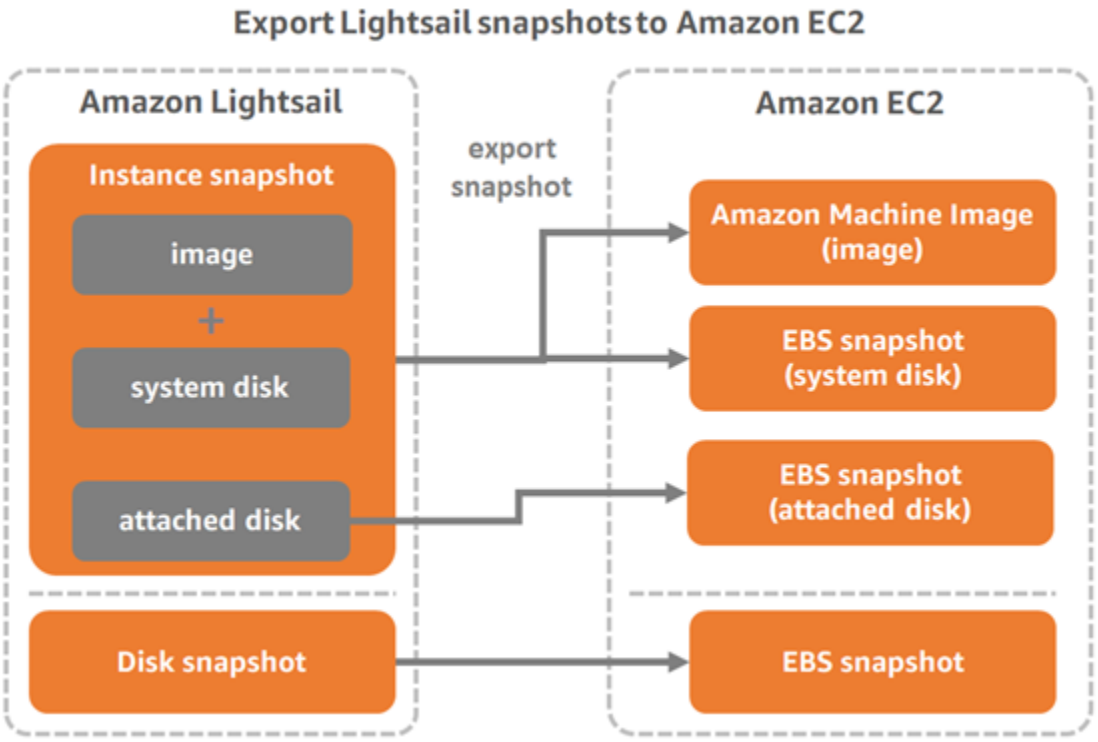
Amazon EC2로 Lightsail 스냅샷을 내보내는 방법

Lightsail 스냅샷을 Amazon EC2로 내보내고, 내보낸 스냅샷에서 EC2 리소스를 생성하고, 호환되는 EC2 인스턴스 유형을 선택하고, EC2 인스턴스에 연결하고, Lightsail 스냅샷에서 생성된 EC2 인스턴스를 보호할 수 있습니다. 다음 방법 중 하나를 사용하여 Amazon Elastic Compute Cloud(Amazon EC2)로 Amazon Lightsail 인스턴스 및 블록 스토리지 디스크 스냅샷을 내보낼 수 있습니다.

- Lightsail 콘솔. 자세한 내용은 스냅샷을 Amazon EC2로 내보내기를 참조하세요.
- Lightsail API, AWS Command Line Interface (AWS CLI) 또는 SDKs. 자세한 내용은 Lightsail API 설명서의 ExportSnapshot 작업 또는 AWS CLI 설명서의 export-snapshot 명령을 참조하세요.

인스턴스 스냅샷 및 블록 스토리지 디스크 스냅샷을 내보낼 수 있습니다. 하지만 cPanel 및 WHM(CentOS 7) 인스턴스의 스냅샷은 Amazon EC2로 내보낼 수 없습니다. 스냅샷은 Lightsail의 AWS 리전 에서 Amazon EC2의 동일한 리전으로 내보내집니다. 다른 리전으로 스냅샷을 내보내려면 먼저 Lightsail의 다른 리전으로 스냅샷을 복사한 다음 내보내기를 수행합니다. 자세한 내용은 스냅샷을 한에서 다른 AWS 리전 로 복사를 참조하세요.

Lightsail 인스턴스 스냅샷을 내보내면 Amazon EC2에 Amazon Machine Image(AMI) 및 Amazon Elastic Block Store(Amazon EBS) 스냅샷이 생성됩니다. 이는 Lightsail 인스턴스가 이미지와 시스템 디스크로 구성되지만, 더 효율적으로 관리하기 위해 Lightsail 콘솔에서 이미지와 시스템 디스크가 단일 인스턴스 개체로 그룹화되기 때문입니다. 스냅샷이 생성될 때 소스 Lightsail 인스턴스에 하나 이상의 블록 스토리지 디스크가 연결되어 있는 경우 연결된 각 디스크에 대한 추가 EBS 스냅샷이 Amazon EC2에 생성됩니다. Lightsail 블록 스토리지 디스크 스냅샷을 내보내면 Amazon EC2에 단일 EBS 스냅샷이 생성됩니다. Amazon EC2의 내보낸 모든 리소스에는 Lightsail의 식별자와 다른 자체의 고유 식별자가 있습니다.



> **Note**
>
> Lightsail는 AWS Identity and Access Management (IAM) 서비스 연결 역할(SLR)을 사용하여 스냅샷을 Amazon EC2로 내보냅니다. SLR에 대한 자세한 내용은 서비스 연결 역할을 참조하세요.

내보내기 프로세스는 시간이 걸릴 수 있습니다. 이는 소스 인스턴스 또는 블록 스토리지 디스크의 크기와 구성에 따라 달라집니다. Lightsail 콘솔의 내보내기를 사용하여 내보내기 상태를 추적합니다. 자세한 내용은 Lightsail에서 스냅샷 내보내기 상태 추적 단원을 참조하십시오.

내보낸 Lightsail 스냅샷에서 Amazon EC2 리소스 생성

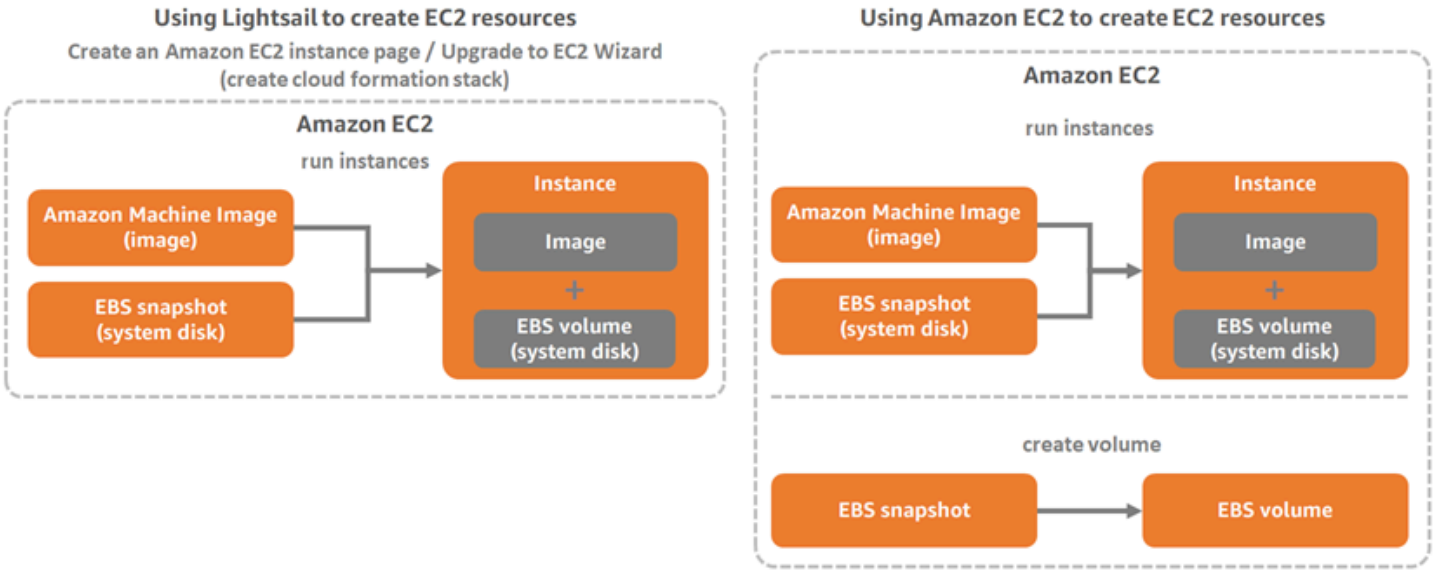
Lightsail 스냅샷이 내보내지고 Amazon EC2에서 사용할 수 있게 되면(AMI, EBS 스냅샷, 또는 둘 다로) 다음 방법 중 하나를 사용하여 스냅샷에서 Amazon EC2 리소스를 생성할 수 있습니다.

- Lightsail 콘솔의 Amazon EC2 인스턴스 생성 페이지(Amazon EC2로 업그레이드 마법사라고도 함). 자세한 내용은 내보낸 스냅샷에서 Amazon EC2 인스턴스 생성을 참조하세요.
- Lightsail API AWS CLI또는 SDKs. 자세한 내용은 Lightsail API 설명서의 CreateCloudFormationStack 작업 또는 AWS CLI 설명서의 create-cloud-formation-stack 명령을 참조하세요.

> **Note**
>
> Lightsail을 사용하여 내보낸 인스턴스 스냅샷에서 Amazon EC2 i인스턴스를 생성할 수 있지만, 이를 사용하여 내보낸 블록 스토리지 디스크 스냅샷에서 EBS 볼륨을 생성할 수는 없습니다. 따라서 Amazon EC2 콘솔, API 또는 AWS CLI를 사용해야 합니다. 자세한 내용은 내보낸 디스크 스냅샷에서 Amazon EBS 볼륨 생성을 참조하세요.

- Amazon EC2 콘솔, Amazon EC2 API AWS CLI또는 SDKs 자세한 내용은 Amazon EC2 설명서의 인스턴스 시작 마법사를 사용하여 인스턴스 시작 또는 스냅샷에서 Amazon EBS 볼륨 복원을 참조하세요.

내보낸 인스턴스 스냅샷(AMI 및 EBS 스냅샷)에서 Amazon EC2 인스턴스를 생성하면 단일 EC2 인스턴스가 시작됩니다. Lightsail 인스턴스 스냅샷 내보내기를 통해 생성된 AMI 및 EBS 스냅샷은 자동으로 연결되어 EC2 인스턴스를 형성합니다. 내보낸 Lightsail 블록 스토리지 디스크 스냅샷(EBS 스냅샷)을 사용하여 Amazon EC2에서 EBS 볼륨을 생성할 수 있습니다.



> **Note**
>
> Lightsail에서는 CloudFormation 스택을 사용하여 EC2에 인스턴스 및 이와 관련된 리소스를 생성합니다. 자세한 내용은 Lightsail용CloudFormation 스택을 참조하세요.

내보낸 스냅샷에서 Amazon EC2 리소스를 생성하는 프로세스는 시간이 걸릴 수 있습니다. 이는 소스 인스턴스의 크기와 구성에 따라 달라집니다. Lightsail 콘솔의 내보내기를 사용하여 내보내기 상태를 추적합니다. 자세한 내용은 단원을 참조하십시오Lightsail에서 스냅샷 내보내기 상태 추적.

Amazon EC2 인스턴스 유형 선택

Amazon EC2는 Lightsail에서 사용할 수 있는 광범위한 인스턴스 옵션을 제공합니다. Amazon EC2에서는 컴퓨팅(C5), 메모리(R5) 또는 2가지 모두의 균형(T3 및 M5)에 최적화된 인스턴스 유형을 선택할 수 있습니다. Lightsail은 Amazon EC2 인스턴스 생성 페이지에서 이러한 옵션을 제공합니다. 그러나 Amazon EC2를 사용하여 내보낸 스냅샷에서 새 인스턴스를 생성하는 경우, 더 많은 인스턴스 유형 옵션을 사용할 수 있습니다. EC2 인스턴스 유형에 대한 자세한 내용은 Amazon EC2 설명서의 인스턴스 유형을 참조하세요.

내보낸 스냅샷에서 EC2 인스턴스를 생성하기 전에 Lightsail과 Amazon EC2 간의 인스턴스 요금 차이를 이해하는 것이 중요합니다. 인스턴스 요금에 대한 자세한 내용은 Lightsail 요금 및 Amazon EC2 요금 페이지를 참조하세요.

Lightsail 및 Amazon EC2 인스턴스 유형 호환성

일부 Lightsail 인스턴스는 향상된 네트워킹에 대해 활성화되지 않았으므로 현재 세대 EC2 인스턴스 유형(T3, M5, C5 또는 R5)과 호환되지 않습니다. 소스 Lightsail 인스턴스가 호환되지 않는 경우 내보낸 스냅샷에서 EC2 인스턴스를 생성할 때 이전 세대 인스턴스 유형(T2, M4, C4 또는 R4)을 선택해야 합니다. 이러한 옵션은 Lightsail 콘솔에서 Amazon EC2 인스턴스 생성 페이지를 사용하여 EC2 인스턴스를 생성할 때 표시됩니다.

소스 Lightsail 인스턴스가 호환되지 않는 경우 최신 세대 EC2 인스턴스 유형을 사용하려면 이전 세대 인스턴스 유형(T2, M4, C4 또는 R4)을 사용하여 새 EC2 인스턴스를 생성하고, 네트워킹 드라이버를 업데이트하고, 인스턴스를 원하는 현재 세대 인스턴스 유형으로 업그레이드해야 합니다. 자세한 내용은 Amazon EC2 인스턴스를 위한 향상된 네트워킹을 참조하세요.

Amazon EC2 인스턴스에 연결

Lightsail 인스턴스에 연결하는 방법과 비슷하게 Amazon EC2 인스턴스에 연결할 수 있습니다. 즉, Linux 및 Unix 인스턴스의 경우 SSH를 사용하고, Windows Server 인스턴스의 경우 RDP를 사용합니다. 그러나 Lightsail 콘솔에서 사용했을 수도 있는 브라우저 기반 SSH/RDP 클라이언트는 사용하고 있는 브라우저 버전에 따라 Amazon EC2에서 사용하지 못할 수도 있습니다. 따라서 자체 SSH/RDP 클라이언트를 구성하여 EC2 인스턴스에 연결해야 할 수도 있습니다. 자세한 내용은 다음 안내서를 참조하세요.

- Lightsail 스냅샷에서 생성된 Amazon EC2 Linux 또는 Unix 인스턴스에 연결

- Lightsail 스냅샷에서 생성된 Amazon EC2 Windows Server 인스턴스에 연결

Amazon EC2 인스턴스 보호

내보낸 Lightsail 스냅샷에서 EC2 인스턴스를 생성한 후 새 인스턴스의 보안을 강화하기 위해 몇 가지 작업을 수행해야 할 수도 있습니다. 이 작업은 EC2 인스턴스의 운영 체제에 따라 달라집니다.

Amazon EC2의 Linux 및 Unix 인스턴스 보호

EC2(EC2 콘솔, EC2 API, EC2용 또는 EC2용 SDK)를 사용하여 내보낸 스냅샷에서 Amazon EC2에 Linux 또는 Unix 인스턴스를 생성하는 경우 새 ECEC2 인스턴스에 Lightsail 서비스의 남은 SSH 키가 포함될 수 있습니다. EC2 AWS CLI SDKs 새 인스턴스의 보안을 강화하기 위해 이러한 키를 제거하는 것이 좋습니다.

자세한 내용은 Lightsail 스냅샷에서 생성된 Amazon EC2 Linux 또는 Unix 인스턴스 보호를 참조하세요.

Amazon EC2의 Windows Server 인스턴스 보호

내보낸 스냅샷에서 Amazon EC2에 Windows Server 인스턴스를 생성한 후 Lightsail 및 EC2에 액세스할 수 있는 AWS 계정의 모든 사용자는 소스 인스턴스에 처음 할당된 기본 관리자 암호인 새 EC2 인스턴스의 암호도 검색할 수 있습니다. 보안을 강화하기 위해 Amazon EC2 인스턴스에 대한 기본 관리자 암호를 아직 변경하지 않았다면 변경하는 것이 좋습니다.

자세한 내용은 Lightsail 스냅샷에서 생성된 Amazon EC2 Windows Server 인스턴스 보호를 참조하세요.

Amazon EC2로 Lightsail 스냅샷 내보내기

Amazon Lightsail 인스턴스 및 블록 스토리지 디스크 스냅샷을 Amazon Elastic Compute Cloud(Amazon EC2)로 내보낼 수 있습니다. Lightsail 인스턴스 스냅샷을 내보내면 Amazon EC2에 Amazon Machine Image(AMI) 및 Amazon Elastic Block Store(Amazon EBS) 스냅샷이 생성됩니다. 이는 Lightsail 인스턴스가 이미지와 시스템 디스크로 구성되어 있으며 더 효율적인 관리를 위해 Lightsail 콘솔에서 이들이 단일 인스턴스 엔터티로 그룹화되기 때문입니다. 스냅샷이 생성될 때 소스 Lightsail 인스턴스에 하나 이상의 블록 스토리지 디스크가 연결되어 있는 경우 연결된 각 디스크에 대한 추가 EBS 스냅샷이 Amazon EC2에 생성됩니다.

Lightsail 블록 스토리지 디스크 스냅샷을 내보내면 Amazon EC2에 단일 EBS 스냅샷이 생성됩니다. Amazon EC2의 내보낸 모든 리소스에는 Lightsail의 식별자와 다른 자체의 고유 식별자가 있습니다.

이 설명서에서는 Lightsail 스냅샷을 내보내는 방법, 내보내기 상태를 추적하는 방법 및 내보낸 스냅샷을 Amazon EC2에서 사용할 수 있게 된 이후(AMI, EBS 스냅샷 또는 둘 다) 다음 단계를 설명합니다.

> **Important**
>
> 이 안내서의 단계를 완료하기 전에 Lightsail 내보내기 프로세스에 익숙해지는 것이 좋습니다. 자세한 내용은 스냅샷을 Amazon EC2로 내보내기를 참조하세요.

목차

- Lightsail 스냅샷을 내보내기 위한 서비스 연결 역할 및 필수 IAM 권한
- 사전 조건
- Amazon EC2로 Lightsail 스냅샷 내보내기
- 내보내기 상태 추적

Lightsail 스냅샷을 내보내기 위한 서비스 연결 역할 및 필수 IAM 권한

Lightsail는 AWS Identity and Access Management (IAM) 서비스 연결 역할(SLR)을 사용하여 스냅샷을 Amazon EC2로 내보냅니다. SLR에 대한 자세한 내용은 서비스 연결 역할을 참조하세요.

스냅샷 내보내기를 수행할 사용자에 따라 IAM에서 다음 추가 권한을 구성해야 할 수도 있습니다.

- Amazon 계정 루트 사용자가 내보내기를 수행할 경우 이 안내서의 사전 조건 단원으로 계속 진행합니다. 계정 루트 사용자에게는 스냅샷 내보내기를 수행하는 데 필요한 권한이 이미 있습니다.
- IAM 사용자가 내보내기를 수행할 경우 AWS 계정 관리자는 사용자에게 다음 정책을 추가해야 합니다. 사용자의 권한을 변경하는 방법에 대한 자세한 내용은 IAM 설명서의 IAM 사용자용 권한 변경을 참조하세요.

  JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "iam:CreateServiceLinkedRole",
            "Resource": "arn:aws:iam::*:role/aws-service-role/
lightsail.amazonaws.com/AWSServiceRoleForLightsail*",
```

```
            "Condition": {"StringLike": {"iam:AWSServiceName":
 "lightsail.amazonaws.com"}}
        },
        {
            "Effect": "Allow",
            "Action": "iam:PutRolePolicy",
            "Resource": "arn:aws:iam::*:role/aws-service-role/
lightsail.amazonaws.com/AWSServiceRoleForLightsail*"
        }
    ]
}
```

사전 조건

Amazon EC2로 내보낼 Lightsail 인스턴스 또는 블록 스토리지 디스크의 스냅샷을 생성합니다. 자세한 내용은 다음 안내서 중 하나를 참조하세요.

- Linux 또는 Unix 인스턴스의 스냅샷 생성
- Windows Server 인스턴스의 스냅샷 생성
- 블록 스토리지 디스크 스냅샷 생성

Amazon EC2로 Lightsail 스냅샷 내보내기

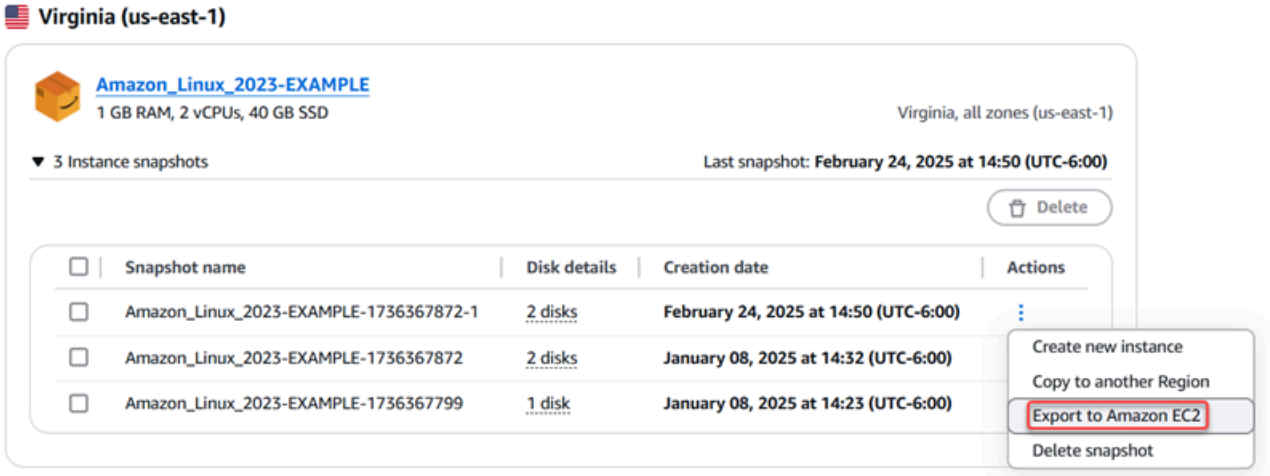
스냅샷을 Amazon EC2로 내보내는 가장 효율적인 방법은 Lightsail 콘솔을 사용하는 것입니다. Lightsail API, AWS Command Line Interface (AWS CLI) 또는 SDKs. 자세한 내용은 Lightsail API 설명서의 ExportSnapshot 작업 또는 AWS CLI 설명서의 export-snapshot 명령을 참조하세요.

> **Note**
>
> 스냅샷은 AWS 리전 에서 Amazon EC2로 동일한 Lightsail로 내보내집니다. 다른 리전으로 스냅샷을 내보내려면 먼저 Lightsail의 다른 리전으로 스냅샷을 복사한 다음 내보내기를 수행합니다. 자세한 내용은 스냅샷을 한에서 다른 AWS 리전 로 복사를 참조하세요.

Amazon EC2로 Lightsail 스냅샷 내보내기

1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 Snapshots을 선택합니다.

3. 내보낼 인스턴스 또는 블록 스토리지 디스크를 찾고 노드를 확장하여 해당 리소스에 대해 사용 가능한 스냅샷을 확인합니다.
4. 원하는 스냅샷에 대해 작업 메뉴를 선택한 다음 Amazon EC2로 내보내기를 선택합니다.



> **Note**
>
> cPanel 및 WHM(CentOS 7) 인스턴스의 스냅샷은 Amazon EC2로 내보낼 수 없습니다.

5. 프롬프트에 표시된 중요 세부 정보를 검토합니다.
6. Amazon EC2로 내보내기에 동의할 경우 예, 계속을 선택하여 프로세스를 시작합니다.

   내보내기 프로세스는 시간이 걸릴 수 있습니다. 이는 소스 인스턴스 또는 블록 스토리지 디스크의 크기와 구성에 따라 달라집니다. Lightsail 콘솔의 내보내기를 사용하여 내보내기 상태를 추적합니다. 자세한 내용은 Lightsail에서 스냅샷 내보내기 상태 추적 단원을 참조하십시오.

내보내기 상태 추적

Lightsail 콘솔의 내보내기 섹션에서 내보내기 상태를 추적합니다. Lightsail 콘솔의 모든 페이지에 있는 왼쪽 탐색 창에서 액세스할 수 있습니다. 자세한 내용은 Lightsail에서 스냅샷 내보내기 상태 추적 단원을 참조하십시오.

내보내기에 다음 정보가 표시됩니다.

- Snapshot name(스냅샷 이름) - 소스 Lightsail 스냅샷의 이름입니다.
- 상태 - 내보내기 작업 상태를 가져옵니다. `In progress`, `Successful` 또는 `Failed` 유형을 지정할 수 있습니다.
- Export started(내보내기 시작됨) - 스냅샷 내보내기가 시작된 날짜와 시간입니다.

- 소스 상세 정보 - 메모리, 처리, 스토리지 등 소스 Lightsail 인스턴스의 사양입니다.
- 소스 인스턴스 이름 - 스냅샷의 소스 인스턴스 이름입니다.
- 스냅샷 유형 - Lightsail 스냅샷의 유형입니다. 이는 인스턴스 스냅샷 또는 디스크 스냅샷입니다.
- Snapshot created(스냅샷 생성됨) - 소스 Lightsail 스냅샷이 생성된 날짜와 시간입니다.

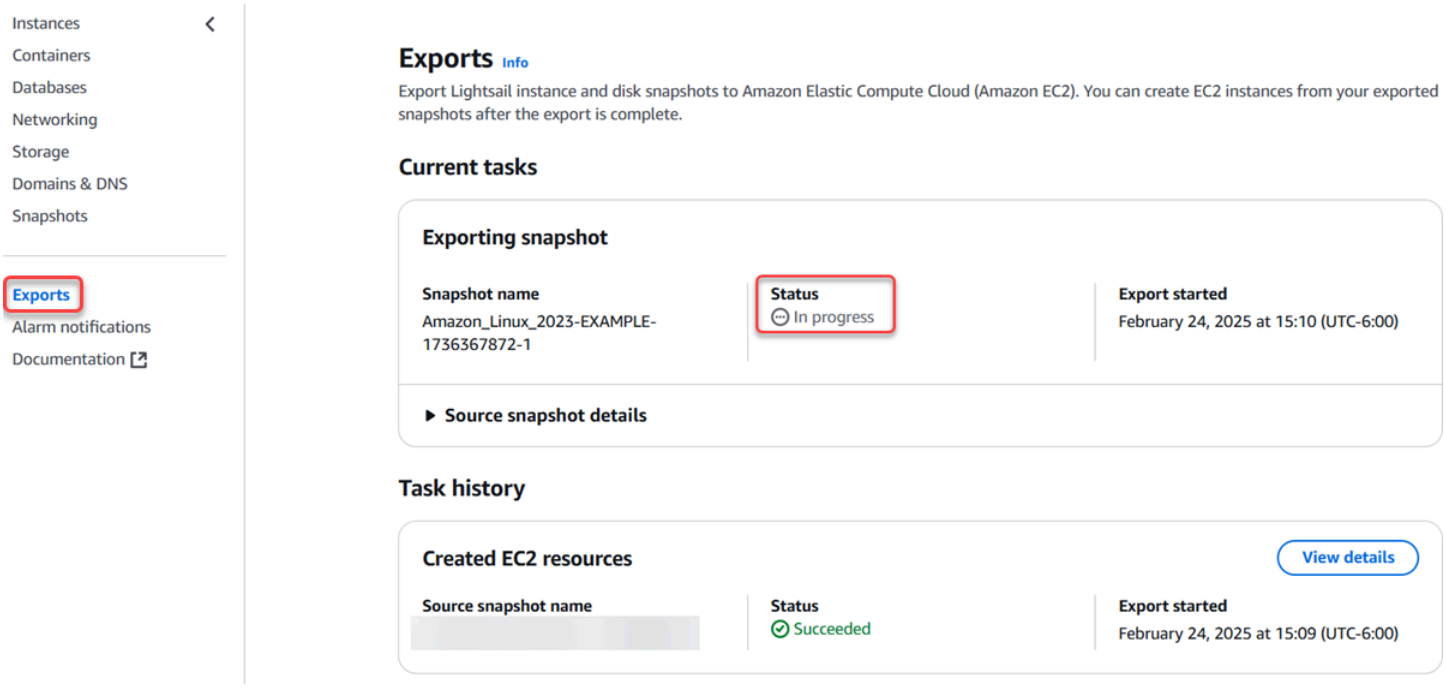
완료된 내보내기에 대해 작업 기록 섹션에 다음 정보가 표시됩니다.

- EC2에 인스턴스 생성 - Lightsail 콘솔을 사용하여 Amazon EC2에 새 인스턴스를 생성하려면 이 옵션을 선택합니다. 자세한 내용은 내보낸 스냅샷에서 Amazon EC2 인스턴스 생성을 참조하세요.
- EC2 열기 - Amazon EC2 콘솔을 사용하여 내보낸 스냅샷에서 새 EC2 리소스를 생성하려면 이 옵션을 선택합니다. Lightsail 블록 스토리지 디스크 스냅샷을 내보낸 경우 Amazon EC2를 사용하여 스냅샷(EBS 스냅샷)에서 EBS 볼륨을 생성해야 합니다. 자세한 내용은 Amazon EC2 설명서의 인스턴스 시작 마법사를 사용하여 인스턴스 시작 또는 스냅샷에서 Amazon EBS 볼륨 복원을 참조하세요.

> **Note**
>
> 더 이상 필요하지 않는 경우 소스 Lightsail 스냅샷을 삭제하세요. 그렇지 않으면 스냅샷 저장에 대해 비용이 청구됩니다.

Lightsail에서 스냅샷 내보내기 상태 추적

Amazon Lightsail 콘솔의 내보내기 섹션은 Lightsail 스냅샷을 Amazon EC2로 내보내는 작업의 상태를 추적하거나 내보낸 인스턴스 스냅샷에서 새로운 EC2 인스턴스를 생성하는 작업의 상태를 추적할 수 있습니다. 내보내기 작업은 소스 인스턴스 또는 블록 스토리지 디스크의 크기와 구성에 따라 다소 시간이 걸릴 수 있습니다. 내보내기는 Lightsail 콘솔의 모든 페이지에 있는 왼쪽 탐색 창에서 액세스할 수 있습니다.



Amazon EC2로 Lightsail 스냅샷을 내보내거나 내보낸 스냅샷에서 EC2 인스턴스를 생성하는 방법에 대한 자세한 내용은 다음 설명서를 참조하세요.

- Amazon EC2로 스냅샷 내보내기
- 내보낸 스냅샷에서 Amazon EC2 인스턴스 생성

내보낸 Lightsail 스냅샷에서 Amazon EC2 인스턴스 생성

Lightsail 인스턴스 스냅샷을 내보내고 Amazon EC2에서 AMI 및 EBS 스냅샷으로 사용할 수 있게 되면 Amazon Lightsail 콘솔의 Amazon EC2 인스턴스 생성 페이지(Amazon EC2로 업그레이드 마법사라고도 함)를 사용하여 스냅샷에서 Amazon EC2 인스턴스를 생성할 수 있습니다. 여기에는 요구 사항과 일치하는 EC2 인스턴스 유형 선택, 보안 그룹 포트 구성, 시작 스크립트 추가 등 EC2 인스턴스 구성 옵션이 잘 안내되어 있습니다. Lightsail 콘솔의 마법사는 새 EC2 인스턴스 및 이와 관련된 리소스의 생성 프로세스를 간소화합니다.

> **Note**
>
> 내보낸 블록 스토리지 디스크 스냅샷에서 Amazon Elastic Block Store(Amazon EBS) 볼륨을 생성하려면 내보낸 디스크 스냅샷에서 Amazon EBS 볼륨 생성을 참조하세요.

Lightsail API AWS CLI또는 SDK를 사용하여 새 EC2 인스턴스를 생성할 수도 있습니다. SDKs 자세한 내용은 Lightsail API 설명서의 CreateCloudFormationStack 작업 또는 AWS CLI 설명서의 create-cloud-formation-stack 명령을 참조하세요. 또는 Amazon EC2에 익숙하다면 EC2 콘솔, Amazon EC2 API AWS CLI또는 SDKs. 자세한 내용은 Amazon EC2 설명서의 인스턴스 시작 마법사를 사용하여 인스턴스 시작 또는 스냅샷에서 Amazon EBS 볼륨 복원을 참조하세요.

> **Important**
>
> 이 안내서의 단계를 완료하기 전에 Lightsail 내보내기 프로세스에 익숙해지는 것이 좋습니다. 자세한 내용은 스냅샷을 Amazon EC2로 내보내기를 참조하세요.

목차

- CloudFormation 용 스택 Lightsail
- 사전 조건
- Lightsail 콘솔의 Amazon EC2 인스턴스 생성 페이지에 액세스
- Amazon EC2 인스턴스 생성
- 새 Amazon EC2 인스턴스의 상태 추적

CloudFormation 용 스택 Lightsail

Lightsail는 CloudFormation 스택을 사용하여 EC2 인스턴스 및 관련 리소스를 생성합니다. Lightsail용 CloudFormation 스택에 대한 자세한 내용은 Lightsail용CloudFormation 스택을 참조하세요.

Amazon EC2 인스턴스 생성 페이지를 통해 EC2 인스턴스를 생성할 사용자에 따라 다음과 같은 추가 권한을 IAM에 구성해야 할 수 있습니다.

- Amazon 계정 루트 사용자가 EC2 인스턴스를 생성할 경우 이 안내서의 사전 조건 단원으로 이동합니다. 루트 사용자에게는 Lightsail을 사용하여 EC2 인스턴스를 생성하는 데 필요한 권한이 이미 있습니다.
- IAM 사용자가 EC2 인스턴스를 생성하는 경우 AWS 계정 관리자는 사용자에게 다음 권한을 추가해야 합니다. 사용자의 권한을 변경하는 방법에 대한 자세한 내용은 IAM 설명서의 IAM 사용자용 권한 변경을 참조하세요.
  - 사용자가 Lightsail을 사용하여 Amazon EC2 인스턴스를 생성할 경우, 다음 권한이 필요합니다.

> **Note**
>
> 이러한 권한을 통해 CloudFormation 스택을 생성할 수 있습니다. 그러나 생성에 실패한 경우 롤백 프로세스에는 추가 권한이 필요할 수 있습니다. 권한이 부족하면 나머지 리소스가 Amazon EC2에 롤백되지 않을 수도 있습니다. 이 경우 CloudFormation 콘솔로 이동하여 EC2 리소스를 수동으로 삭제할 수 있습니다. 자세한 내용은 Lightsail용CloudFormation 스택을 참조하세요.

  - ec2:DescribeAvailabilityZones
  - ec2:DescribeSubnets
  - ec2:DescribeRouteTables
  - ec2:DescribeInternetGateways
  - ec2:DescribeVpcs
  - cloudformation:CreateStack
  - cloudformation:ValidateTemplate
  - iam:CreateServiceLinkedRole
  - iam:PutRolePolicy
- 사용자가 EC2 인스턴스에 대한 보안 그룹에서 포트를 구성할 경우 다음 권한이 필요합니다.
  - ec2:DescribeSecurityGroups
  - ec2:CreateSecurityGroup
  - ec2:AuthorizeSecurityGroupIngress
- 사용자가 Amazon EC2에서 Windows Server 인스턴스를 생성할 경우 다음 권한이 필요합니다.
  - ec2:DescribeKeyPairs
  - ec2:ImportKeyPair
- 사용자가 처음으로 Amazon EC2 인스턴스를 생성하거나 Virtual Private Cloud(VPC)가 올바르게 구성되지 않은 경우 다음 권한이 필요합니다.
  - ec2:AssociateRouteTable
  - ec2:AttachInternetGateway
  - ec2:CreateInternetGateway
  - ec2:CreateRoute
  - ec2:CreateRouteTable

- ec2:CreateSubnet
- ec2:CreateVpc
- ec2:ModifySubnetAttribute
- ec2:ModifyVpcAttribute

사전 조건

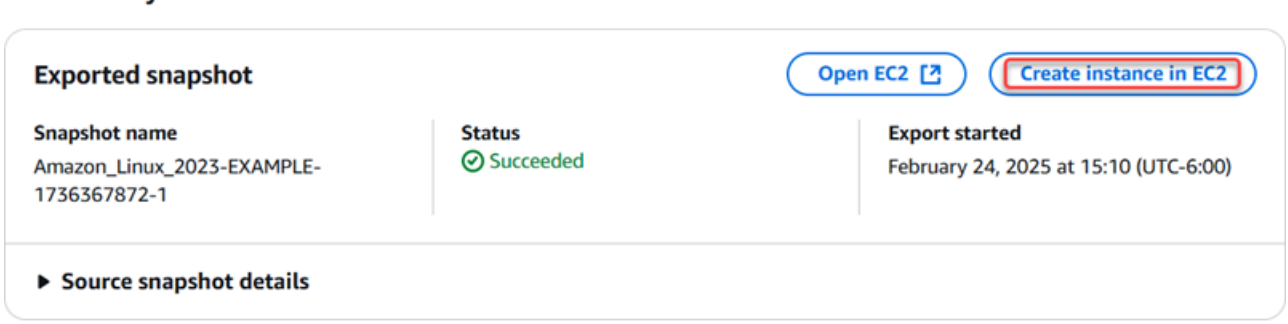
Amazon EC2로 Lightsail 인스턴스 스냅샷 내보내기 자세한 내용은 스냅샷을 Amazon EC2로 내보내기를 참조하세요.

Lightsail 콘솔의 Amazon EC2 인스턴스 생성 페이지에 액세스

인스턴스 스냅샷을 EC2로 내보낸 후에만 작업 모니터에서 Lightsail 콘솔의 Amazon EC2 인스턴스 생성 페이지에 액세스할 수 있습니다.

Lightsail 콘솔의 Amazon EC2 인스턴스 생성 페이지에 액세스

1. Lightsail 콘솔에 로그인합니다.
2. 상단 탐색 창에서 Task monitor(작업 모니터) 아이콘을 선택합니다.
3. 작업 기록 섹션에서 완료된 인스턴스 스냅샷 내보내기를 찾은 다음 EC2에서 인스턴스 생성을 선택합니다.



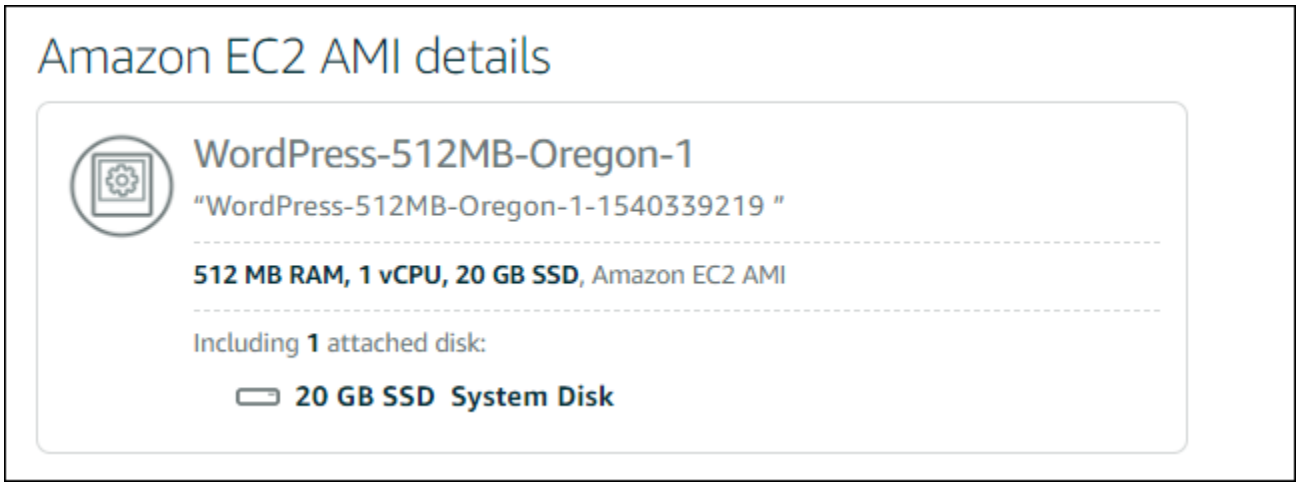
Amazon EC2 인스턴스 생성 페이지가 나타납니다. 이 설명서의 다음 Amazon EC2 인스턴스 생성 섹션으로 계속 진행하여 이 페이지에서 EC2 인스턴스를 구성하고 생성하는 방법에 대해 알아봅니다.

Amazon EC2 인스턴스 생성

Amazon EC2 인스턴스 생성 페이지를 통해 EC2 인스턴스를 생성합니다. 내보낸 Lightsail 스냅샷에서 둘 이상의 EC2 인스턴스를 생성하려면 다음 단계를 여러 번 반복하세요. 단, 각 인스턴스가 생성될 때까지 기다린 후 다음 인스턴스를 생성하세요.

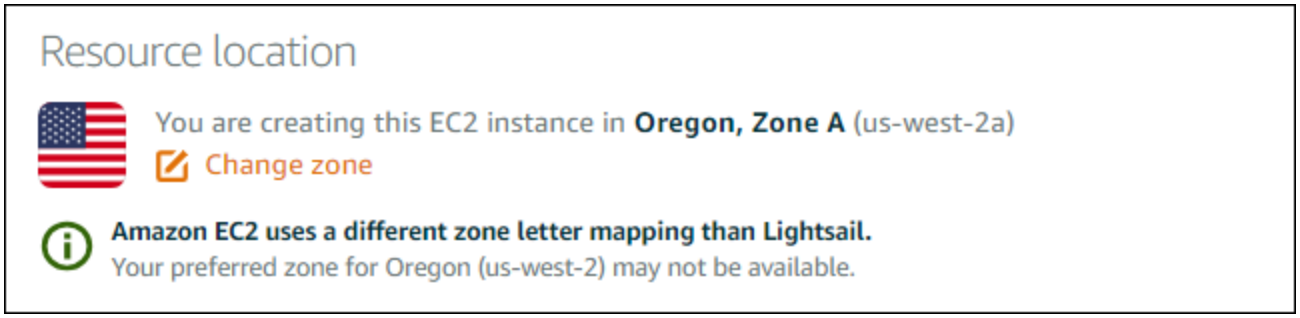
Amazon EC2 인스턴스 생성

1. 페이지의 Amazon EC2 AMI 세부 정보 섹션에서 표시된 Amazon Machine Image(AMI) 세부 정보가 소스 Lightsail 인스턴스의 사양과 일치하는지 확인합니다.

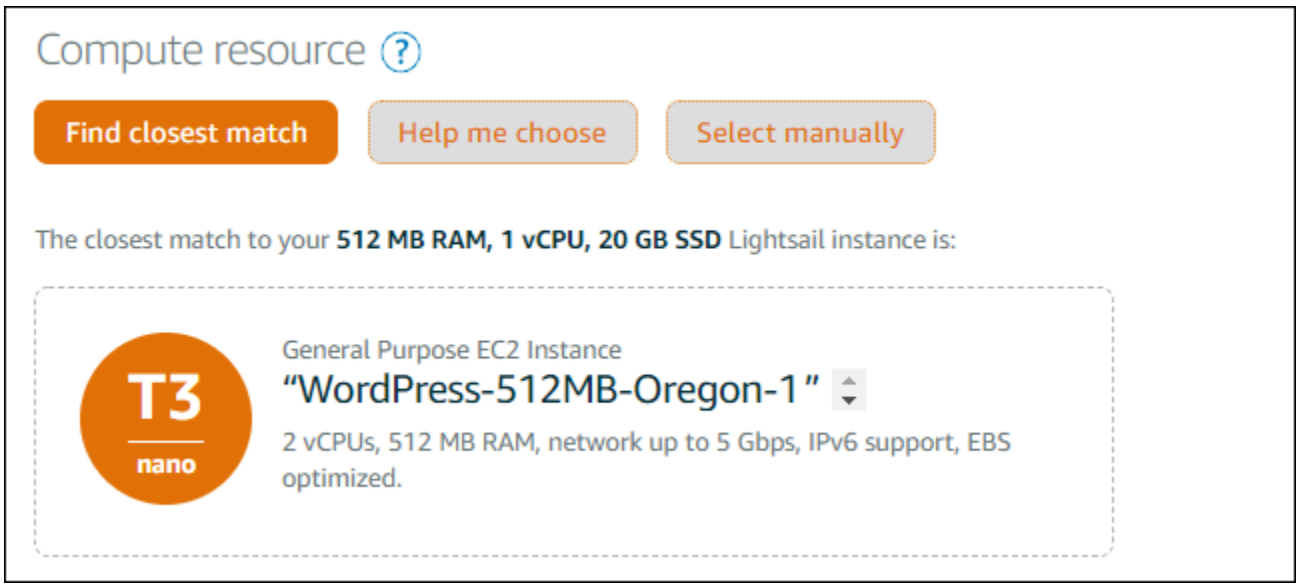


2. 필요한 경우 페이지의 Resource location(리소스 위치) 섹션에서 인스턴스의 가용 영역을 변경합니다. Amazon EC2 리소스는 소스 Lightsail 스냅샷 AWS 리전 과 동일한에 생성됩니다.

> **Note**
>
> 일부 사용자는 일부 가용 영역을 사용할 수 없습니다. 사용할 수 없는 가용 영역을 선택하면 EC2 인스턴스를 생성할 때 오류가 발생합니다.



3. 페이지의 Compute resource(컴퓨팅 리소스) 섹션에서 다음 옵션 중 하나를 선택합니다.



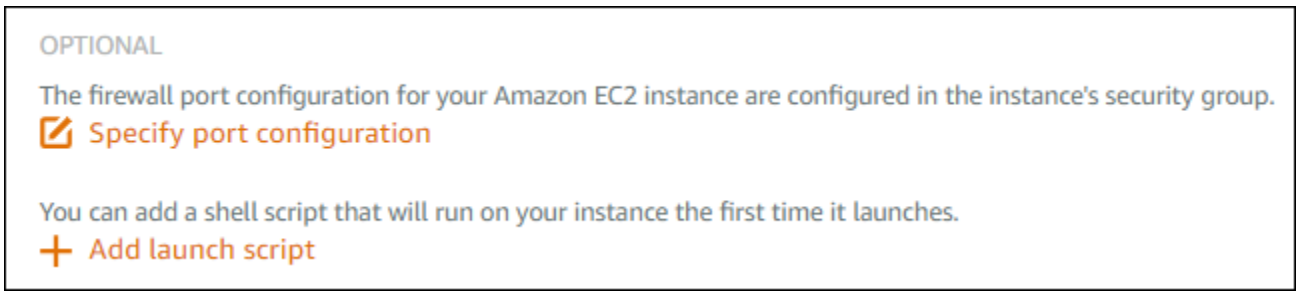
a. 가장 일치하는 항목 찾기는 소스 Lightsail 인스턴스의 사양과 가장 일치하는 Amazon EC2 인스턴스 유형을 자동으로 선택합니다.

b. 선택에 대한 도움말은 새 Amazon EC2 인스턴스의 사양에 대한 빠른 질문서에 답변합니다. 인스턴스 유형 중 컴퓨팅 최적화, 메모리 최적화 또는 두 개가 최적화된 유형을 선택할 수 있습니다.

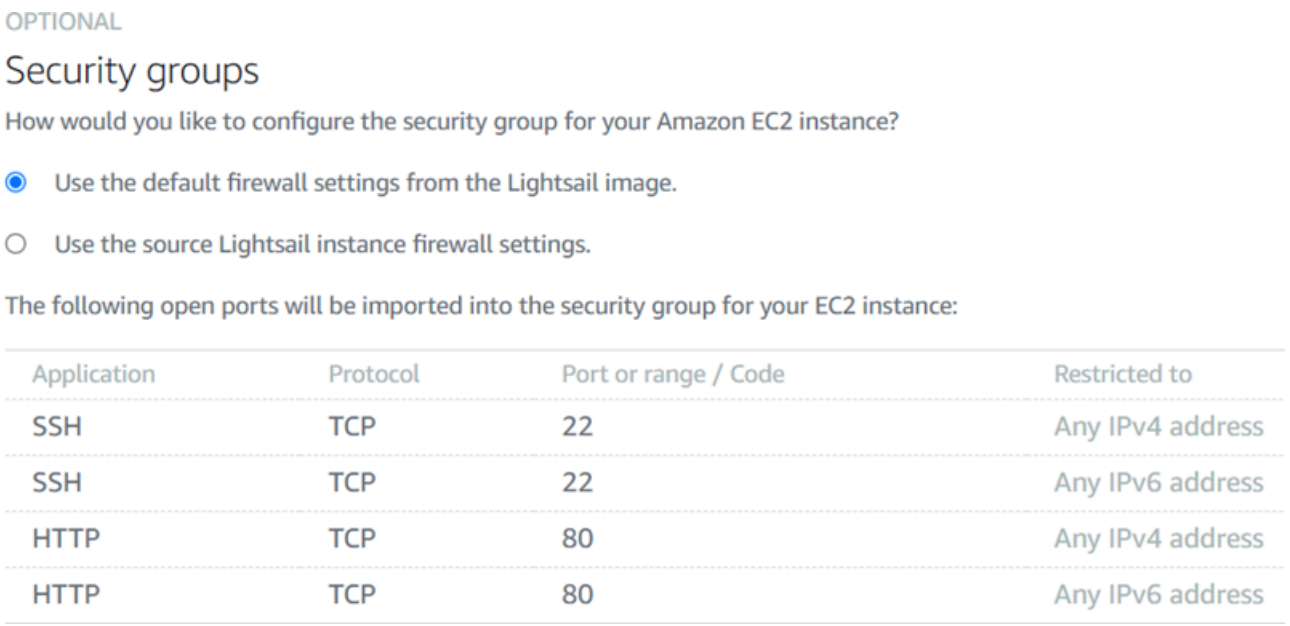
c. 수동으로 선택은 Amazon EC2 인스턴스 생성 페이지를 통해 사용할 수 있는 인스턴스 유형의 목록을 표시합니다.

> **Note**
>
> 일부 Lightsail 인스턴스는 향상된 네트워킹에 대해 활성화되지 않았으므로 현재 세대 EC2 인스턴스 유형(T3, M5, C5 또는 R5)과 호환되지 않습니다. 소스 Lightsail 인스턴스가 호환되지 않는 경우 내보낸 스냅샷에서 EC2 인스턴스를 생성할 때 이전 세대 인스턴스 유형(T2, M4, C4 또는 R4)을 선택해야 합니다. 이러한 인스턴스 유형 옵션은 Lightsail 콘솔의 Amazon EC2 인스턴스 생성 페이지에 표시됩니다.
> 소스 Lightsail 인스턴스가 호환되지 않는 경우 최신 세대 EC2 인스턴스 유형을 사용하려면 이전 세대 인스턴스 유형(T2, M4, C4 또는 R4)을 사용하여 새 EC2 인스턴스를 생성하고, 네트워킹 드라이버를 업데이트하고, 인스턴스를 원하는 현재 세대 인스턴스 유형으로 업그레이드해야 합니다. 자세한 내용은 향상된 네트워킹을 위해 Amazon EC2 인스턴스 업데이트를 참조하세요.

4. 페이지의 선택 사항 섹션에서 다음 작업을 수행합니다.



a. 포트 구성 지정을 선택하여 Amazon EC2 인스턴스에 대한 방화벽 설정을 선택하고 다음 옵션 중 하나를 선택합니다.



   i. Lightsail 이미지의 기본 방화벽 설정 사용은 새 EC2 인스턴스의 소스 Lightsail 블루프린트에서 기본 포트를 구성합니다. Lightsail 블루프린트의 기본 포트에 대한 자세한 내용은 방화벽 및 포트를 참조하세요.

   ii. 소스 Lightsail 인스턴스 방화벽 설정 사용은 새 EC2 인스턴스의 소스 Lightsail 인스턴스에서 포트를 구성합니다. 이 옵션은 소스 Lightsail 인스턴스가 실행 중인 경우에만 사용할 수 있습니다.

b. 시작할 때 EC2 인스턴스를 구성하는 스크립트를 추가하려는 경우 페이지의 시작 스크립트 섹션에서 시작 스크립트 추가를 선택합니다.

5. 페이지의 Connection security(연결 보안) 섹션에서 소스 Lightsail 인스턴스에 연결되는 방법을 결정합니다. 이렇게 하면 새 EC2 인스턴스에 연결하기 위한 올바른 SSH 키를 가져올 수 있습니다. 다음 중 하나의 방법을 사용하여 소스 Lightsail 인스턴스에 연결할 수 있습니다.

a. 소스 인스턴스 리전의 기본 Lightsail 키 페어 사용 - 해당의 고유한 기본 Lightsail 키를 다운로드하고 사용하여 EC2 인스턴스 AWS 리전 에 연결합니다.

> **Note**
>
> 기본 Lightsail 키 페어는 항상 Lightsail의 Windows Server 인스턴스에 사용됩니다.

b. 자체 키 페어 사용 - 프라이빗 키를 찾고 해당 키를 사용하여 EC2 인스턴스에 연결합니다.

> **Note**
>
> Lightsail에서는 개인용 프라이빗 키를 저장하지 않습니다. 따라서 프라이빗 키 다운로드 옵션은 제공되지 않습니다. 프라이빗 키를 찾을 수 없는 경우 EC2 인스턴스에 연결할 수 없습니다.

6. 페이지의 Storage resources(스토리지 리소스) 섹션에서 생성되는 EBS 볼륨이 시스템 디스크와 소스 Lightsail 인스턴스에 대한 연결된 블록 스토리지 디스크와 일치하는지 확인합니다.

7. Lightsail 외부에서 리소스 생성에 대한 중요 세부 정보를 검토합니다.
8. Amazon EC2에서 인스턴스 생성에 동의할 경우 EC2에서 리소스 생성을 선택합니다.

Lightsail에서는 인스턴스가 생성되는지 및 CloudFormation 스택에 대한 정보가 표시되는지를 확인합니다. Lightsail은 CloudFormation 스택을 사용하여 EC2 인스턴스 및 이와 관련된 리소스를 생성합니다. 자세한 내용은 Lightsail용CloudFormation 스택을 참조하세요.

이 설명서의 새 Amazon EC2 인스턴스의 상태 추적 섹션으로 계속 진행하여 새 EC2 인스턴스의 상태를 추적합니다.

> **Important**
>
> 새 EC2 인스턴스가 생성된 후 동일한 내보낸 스냅샷에서 다른 EC2 인스턴스를 생성할 때까지 기다립니다.

새 Amazon EC2 인스턴스의 상태 추적

Lightsail 콘솔의 내보내기를 사용하여 EC2 인스턴스 상태를 추적합니다. 자세한 내용은 Lightsail에서 스냅샷 내보내기 상태 추적 단원을 참조하십시오.

생성하는 중인 EC2 인스턴스에 대해서는 다음 정보가 표시됩니다.

- Source name(소스 이름) - 소스 Lightsail 스냅샷의 이름입니다.
- 시작됨 - 생성 요청이 시작된 날짜와 시간입니다.

생성된 EC2 인스턴스에 대해서는 작업 모니터에 다음 정보가 표시됩니다.

- Amazon EC2 리소스가 성공적으로 생성된 경우 생성 완료가 표시됩니다.
- EC2 인스턴스 생성 시 문제가 있는 경우 실패가 표시됩니다.

내보낸 Lightsail 디스크 스냅샷에서 Amazon Elastic Block Store 볼륨 생성

Lightsail 블록 스토리지 디스크 스냅샷이 내보내지고 Amazon EC2에서 사용할 수 있게 되면(EBS 스냅샷으로) Amazon EC2 콘솔을 사용하여 스냅샷에서 EBS 볼륨을 생성할 수 있습니다.

> **Note**
>
> 내보낸 인스턴스 스냅샷에서 EC2 인스턴스를 생성하려면 Lightsail의 내보낸 스냅샷에서 Amazon EC2 인스턴스 생성을 참조하세요.

Amazon EC2 API AWS CLI또는 SDKs. 자세한 내용은 Amazon EC2 설명서의 인스턴스 시작 마법사를 사용하여 인스턴스 시작 또는 스냅샷에서 Amazon EBS 볼륨 복원을 참조하세요.

> **Important**
>
> 이 안내서의 단계를 완료하기 전에 Lightsail 내보내기 프로세스에 익숙해지는 것이 좋습니다. 자세한 내용은 스냅샷을 Amazon EC2로 내보내기를 참조하세요.

사전 조건

Lightsail 블록 스토리지 디스크 스냅샷을 Amazon EC2로 내보냅니다. 자세한 내용은 스냅샷을 Amazon EC2로 내보내기를 참조하세요.

내보낸 Lightsail 블록 스토리지 디스크 스냅샷에서 EBS 볼륨 생성

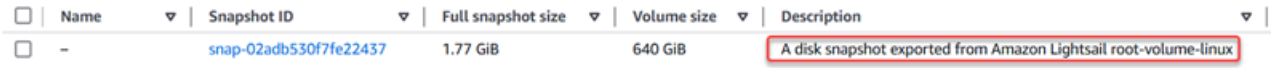
Amazon EC2 콘솔을 사용하여 내보낸 Lightsail 블록 스토리지 디스크 스냅샷에서 새 EBS 볼륨을 생성합니다.

> **Note**
>
> 이 단계는 Amazon EC2 설명서에도 나와 있습니다. 자세한 내용은 Amazon EC2 설명서의 스냅샷에서 Amazon EBS 볼륨 복원을 참조하십시오.

내보낸 Lightsail 블록 스토리지 디스크 스냅샷에서 EBS 볼륨을 생성하려면

1. Amazon EC2 콘솔에 로그인합니다.
2. 탐색 모음에서 스냅샷이 있는 리전을 선택합니다.
3. 왼쪽 탐색 창의 Elastic Block Store에서 스냅샷을 선택합니다.
4. 내보낸 Lightsail 블록 스토리지 디스크 스냅샷을 찾아 선택합니다.

   내보낸 디스크 스냅샷은 다음 스크린샷에 표시된 대로 EBS 스냅샷의 Amazon Lightsail에서 내보낸 디스크 스냅샷 설명으로 식별할 수 있습니다.



5. Actions(작업), Create Volume(볼륨 생성)을 차례대로 선택합니다.

6. Volume Type(볼륨 유형) 드롭다운 메뉴에서 볼륨 유형을 선택합니다. 자세한 내용은 Amazon EC2 설명서의 Amazon EBS 볼륨 유형을 참조하십시오.
7. 크기(GiB)에서 볼륨의 크기를 입력하거나 스냅샷의 기본 크기가 적절한지 확인합니다.
8. 프로비저닝된 IOPS SSD 볼륨의 경우(IOPS) 볼륨이 지원해야 하는 최대 IOPS(초당 입/출력 작업) 수를 입력합니다.
9. 가용 영역에서 볼륨을 생성할 가용 영역을 선택합니다. 동일한 가용 영역의 EC2 인스턴스에만 EBS 볼륨을 연결할 수 있습니다.
10. (선택 사항) 추가 태그 생성(Create additional tags)을 선택하여 볼륨에 태그를 추가합니다. 각 태그에 대해 태그 키와 태그 값을 제공합니다.
11. 볼륨 생성을 선택합니다. 볼륨이 생성되면 해당 볼륨이 Amazon EC2 콘솔의 Elastic Block Store > 볼륨 섹션에 나열됩니다.

Lightsail 스냅샷에서 생성된 Amazon EC2 인스턴스의 Linux에 연결

Amazon Lightsail 스냅샷으로부터 Linux 또는 Unix 인스턴스가 Amazon Elastic Compute Cloud(Amazon EC2)에 생성되면 소스 Lightsail 인스턴스에 연결한 방법과 비슷하게 SSH를 통해 인스턴스에 연결할 수 있습니다. 인스턴스에 인증하려면 소스 인스턴스의 기본 Lightsail 키 페어 AWS 리전 또는 자체 키 페어를 사용합니다. 이 안내서에서는 PuTTY를 사용하여 EC2의 Linux 또는 Unix 인스턴스에 연결하는 방법을 보여줍니다.

> **Note**
>
> Windows Server 인스턴스에 연결하는 방법에 대한 자세한 내용은 Lightsail 스냅샷에서 생성된 Amazon EC2 Windows Server 인스턴스에 연결을 참조하세요.

목차

- 인스턴스의 키 가져오기
- 인스턴스에 대한 퍼블릭 DNS 주소 가져오기
- PuTTY 다운로드 및 설치
- PuTTYgen을 사용하여 키 구성
- PuTTY를 구성하여 인스턴스에 연결
- 다음 단계

인스턴스의 키 가져오기

새 Amazon EC2 인스턴스에 연결하는 데 필요한 올바른 키를 가져옵니다. 필요한 키는 소스 Lightsail 인스턴스에 연결하는 방법에 따라 다릅니다. 다음 중 하나의 방법을 사용하여 소스 Lightsail 인스턴스에 연결할 수 있습니다.

- 소스 인스턴스의 영역에 대한 기본 Lightsail 키 페어 사용 - Lightsail 계정 페이지의 SSH 키(SSH keys) 탭에서 기본 프라이빗 키를 다운로드합니다. 기본 Lightsail 키 사용에 대한 자세한 내용은 SSH 키 페어를 참조하세요.

  > **Note**
  >
  > EC2 인스턴스에 연결한 후 인스턴스에서 기본 Lightsail 키를 제거하고 자체 키 페어로 바꾸는 것이 좋습니다. 자세한 내용은 Lightsail 스냅샷에서 생성된 Amazon EC2의 Linux 또는 Unix 인스턴스 보호를 참조하세요.

- 자체 키 페어 사용 - 프라이빗 키를 찾고 이 키를 사용하여 Amazon EC2 인스턴스에 연결합니다. 자체 키 페어를 사용할 때에는 Lightsail에서 프라이빗 키를 저장하지 않습니다. 프라이빗 키를 분실한 경우 Amazon EC2 인스턴스에 연결할 수 없습니다.

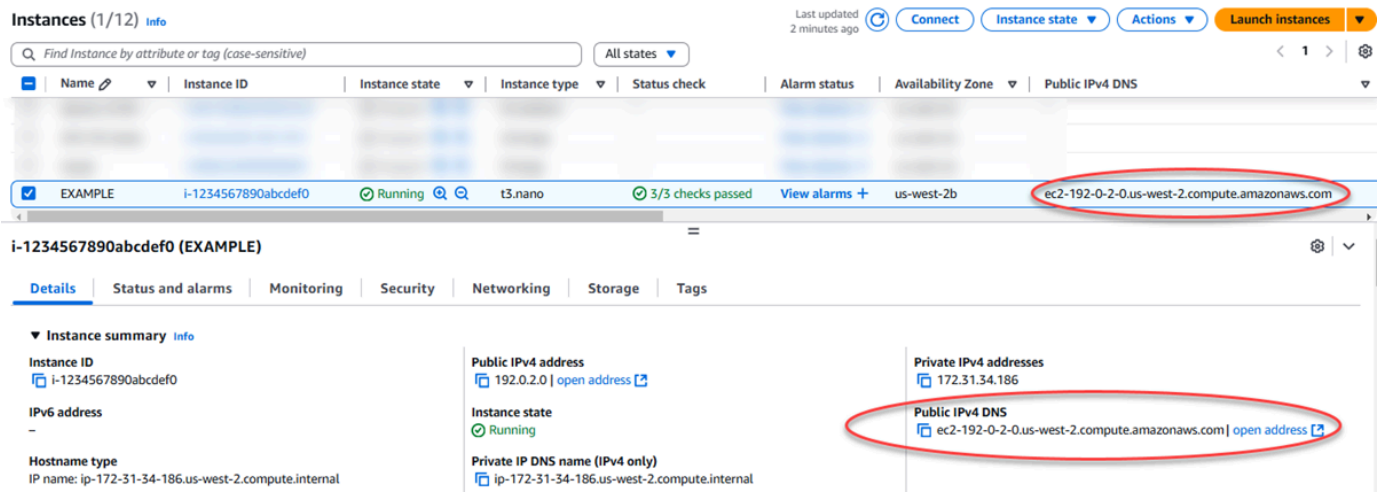
인스턴스에 대한 퍼블릭 DNS 주소 가져오기

PuTTY와 같은 SSH 클라이언트를 구성하여 인스턴스에 연결할 때 Amazon EC2 인스턴스에 대한 퍼블릭 DNS 주소를 사용할 수 있도록 해당 주소를 가져옵니다.

인스턴스에 대한 퍼블릭 DNS 주소를 가져오려면

1. Amazon EC2 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 인스턴스를 선택합니다.
3. 연결하려는 실행 중인 Linux 또는 Unix 인스턴스를 선택합니다.
4. 하단 창에서 인스턴스에 대한 퍼블릭 DNS 주소를 찾습니다.

   이는 SSH 클라이언트를 구성하여 인스턴스에 연결할 때 사용할 주소입니다. 이 안내서의 PuTTY 다운로드 및 설치로 계속 진행하여 PuTTY SSH 클라이언트의 다운로드 및 설치 방법을 알아봅니다.



PuTTY 다운로드 및 설치

PuTTY는 Windows용 무료 SSH 클라이언트입니다. PuTTY에 대한 자세한 내용은 PuTTY: 무료 SSH 및 Telnet 클라이언트를 참조하세요. 이 웹 사이트에는 암호화가 허용되지 않는 국가에서의 제한 사항도 설명되어 있습니다. PuTTY가 이미 있는 경우, 다음과 같이 이 안내서의 PuTTYgen을 사용하여 키 구성 단원으로 건너뛸 수 있습니다.

PuTTY 설치 관리자 또는 실행 파일을 다운로드합니다. 최신 버전을 사용하는 것이 좋습니다. 그러나 선택할 다운로드 파일에 대한 정보는 PuTTY 설명서를 참조하세요.

이 안내서의 PuTTYgen을 사용하여 키 구성으로 계속 진행하여 PuTTYgen을 사용하여 키를 구성합니다.

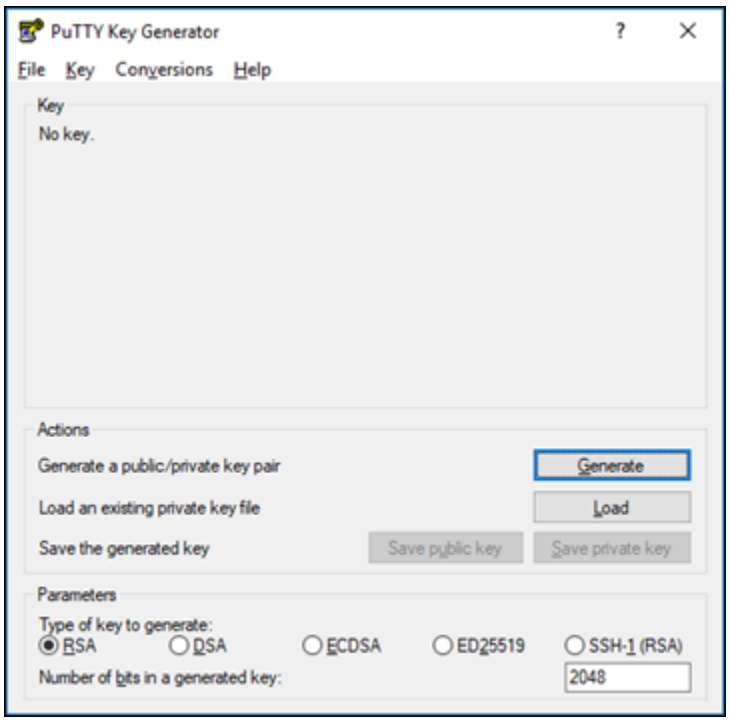
PuTTYgen을 사용하여 키 구성

PuTTYgen은 PuTTY와 함께 사용할 퍼블릭 및 프라이빗 키 페어를 생성합니다. 이 단계에서는 PuTTY에서 허용하는 키 파일 유형(.PPK)을 사용해야 합니다.

PuTTYgen을 사용하여 키를 구성하려면

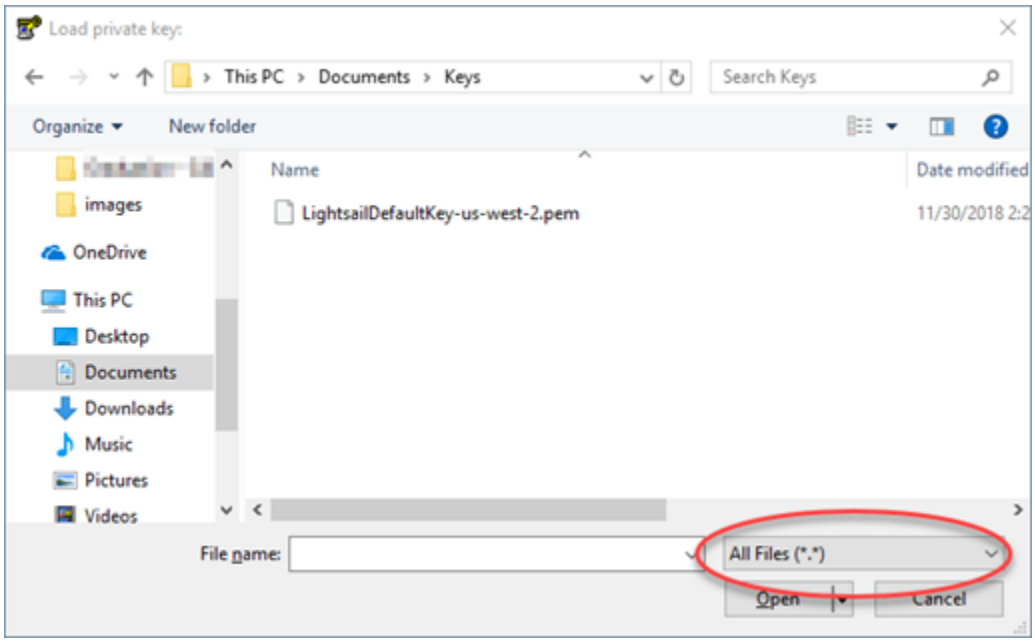
1. PuTTYgen을 시작합니다.

   예를 들어, Windows 시작 메뉴, 모든 프로그램, PuTTY 및 PuTTYgen을 차례대로 선택합니다.



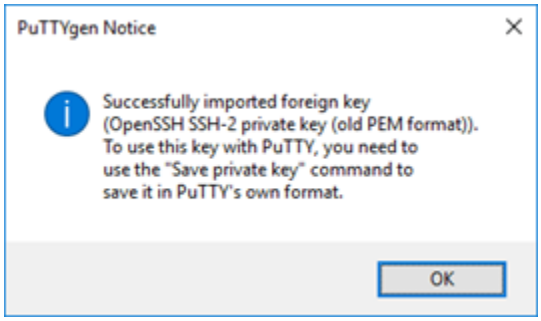
2. 로드(Load)를 선택합니다.

   기본적으로 PuTTYgen에는 확장명이 .PPK인 파일만 표시됩니다. .PEM 파일을 찾으려면 모든 유형의 파일을 표시하는 옵션을 선택합니다.



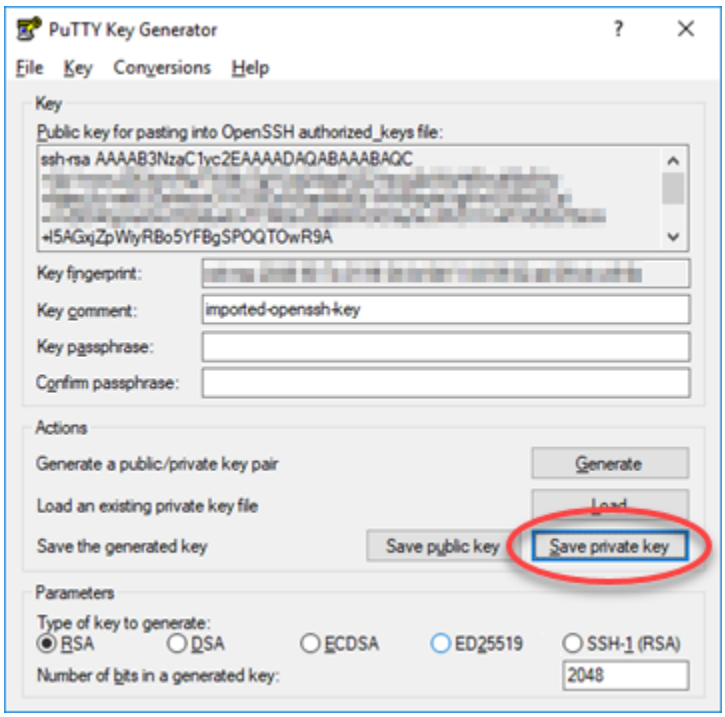
3. 이 안내서의 초반부에서 다운로드한 기본 Lightsail 키 파일(.PEM)을 선택하고 Open(열기)을 선택합니다.

4. PuTTYgen에서 키를 성공적으로 가져왔다고 확인하면 OK(확인)를 선택합니다.



5. Save private key(프라이빗 키 저장)를 선택한 다음 암호와 함께 저장하지 않을 것임을 확인합니다.

   추가적인 보안 방법으로서 암호를 생성할 경우 PuTTY를 사용하여 인스턴스에 연결할 때마다 암호를 입력해야 합니다.



6. 프라이빗 키를 저장하기 위해 이름과 위치를 지정한 후 저장을 선택합니다.

   PuTTYgen에서 새 키 파일을 .PPK 파일 유형으로 저장합니다.

7. PuTTYgen을 닫습니다.

   이 설명서의 PuTTY를 구성하여 인스턴스에 연결 섹션으로 계속 진행합니다. 이 섹션에서는 생성된 새 .PPK 파일을 사용하여 PuTTY를 구성하고 Amazon EC2의 Linux 또는 Unix 인스턴스에 연결합니다.

PuTTY를 구성하여 인스턴스에 연결

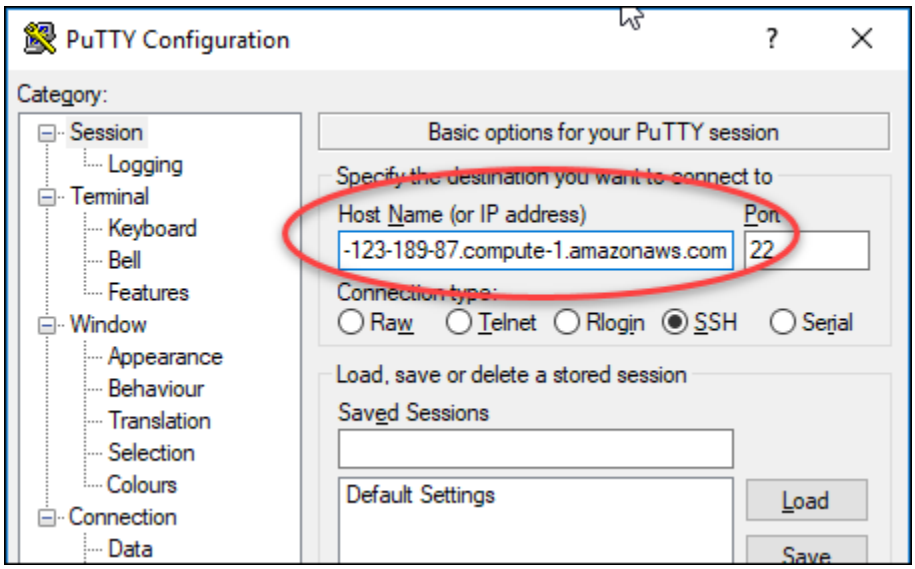
이제 SSH를 사용하여 Linux 또는 Unix 인스턴스에 연결하기 위한 모든 사전 조건이 충족되었으므로 PuTTY를 구성합니다.

PuTTY를 구성하여 Linux 또는 Unix 인스턴스에 연결하려면

1. PuTTY를 엽니다.

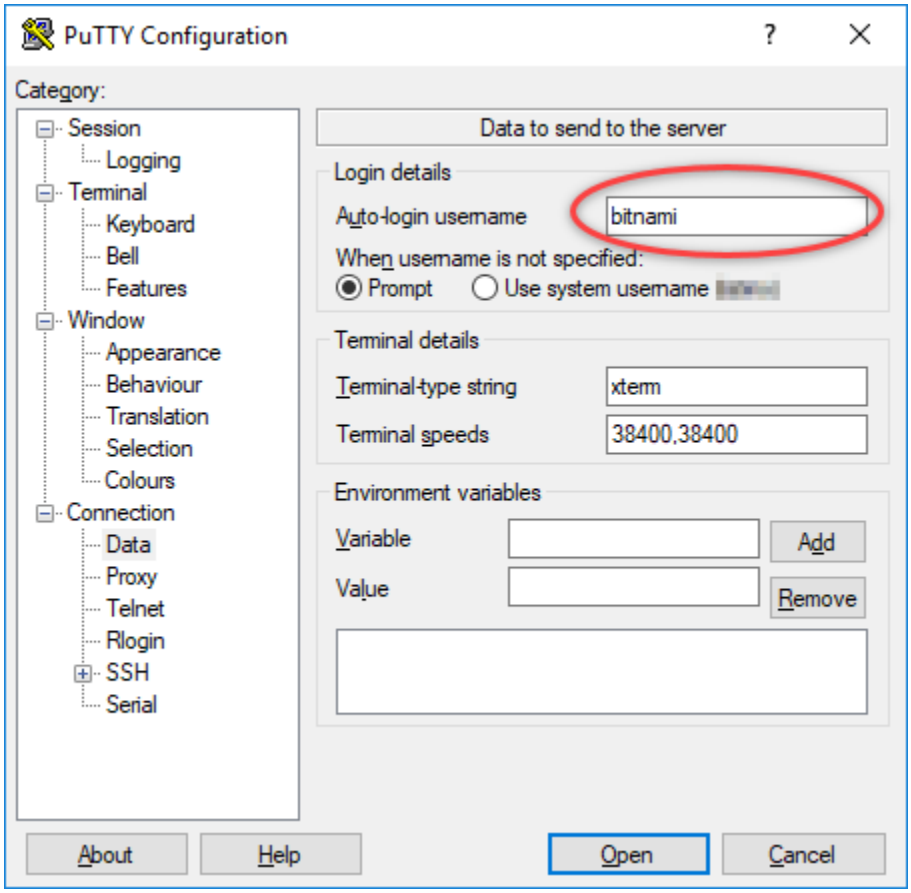
   예를 들어, Windows 시작 메뉴, 모든 프로그램, PuTTY 및 PuTTY를 차례대로 선택합니다.

2. 호스트 이름 텍스트 상자에 이 설명서의 초반부에서 Amazon EC2 콘솔로부터 가져온 인스턴스에 대한 퍼블릭 DNS 주소를 입력합니다.



3. 왼쪽 탐색 창의 Connection(연결) 섹션에서 Data(데이터)를 선택합니다.

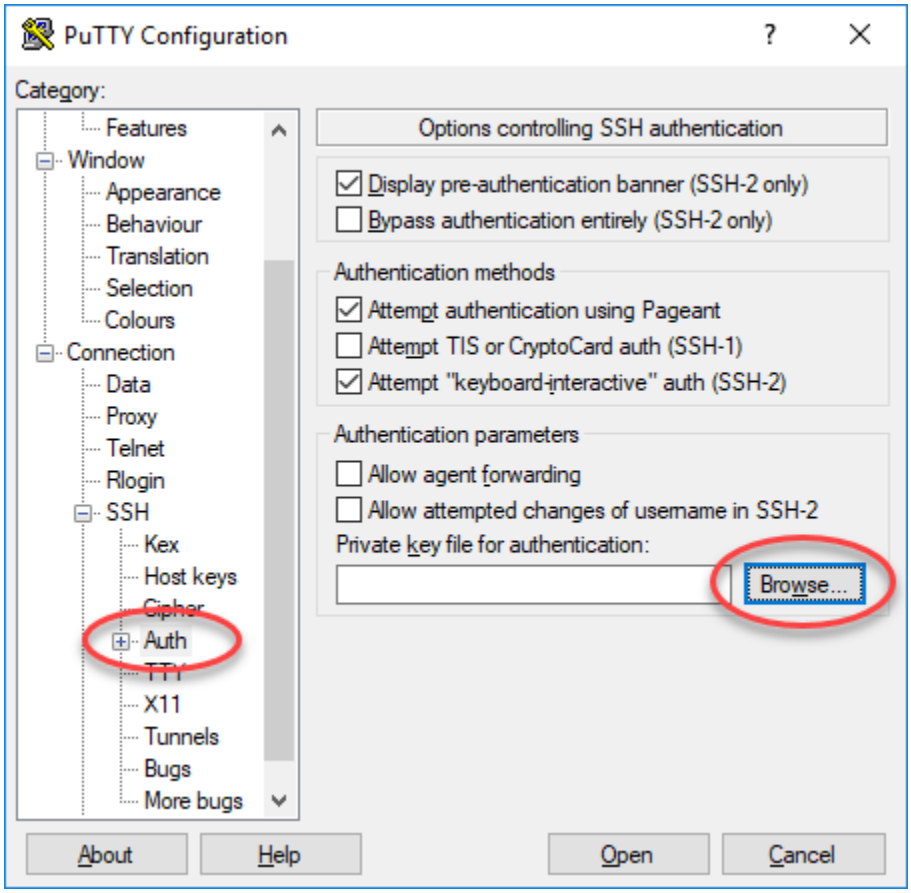
4. Auto-login username(자동 로그인 사용자 이름) 텍스트 상자에 인스턴스에 로그인할 때 사용할 사용자 이름을 입력합니다.



소스 Lightsail 인스턴스의 블루프린트에 따라 다음 기본 사용자 이름 중 하나를 입력합니다.

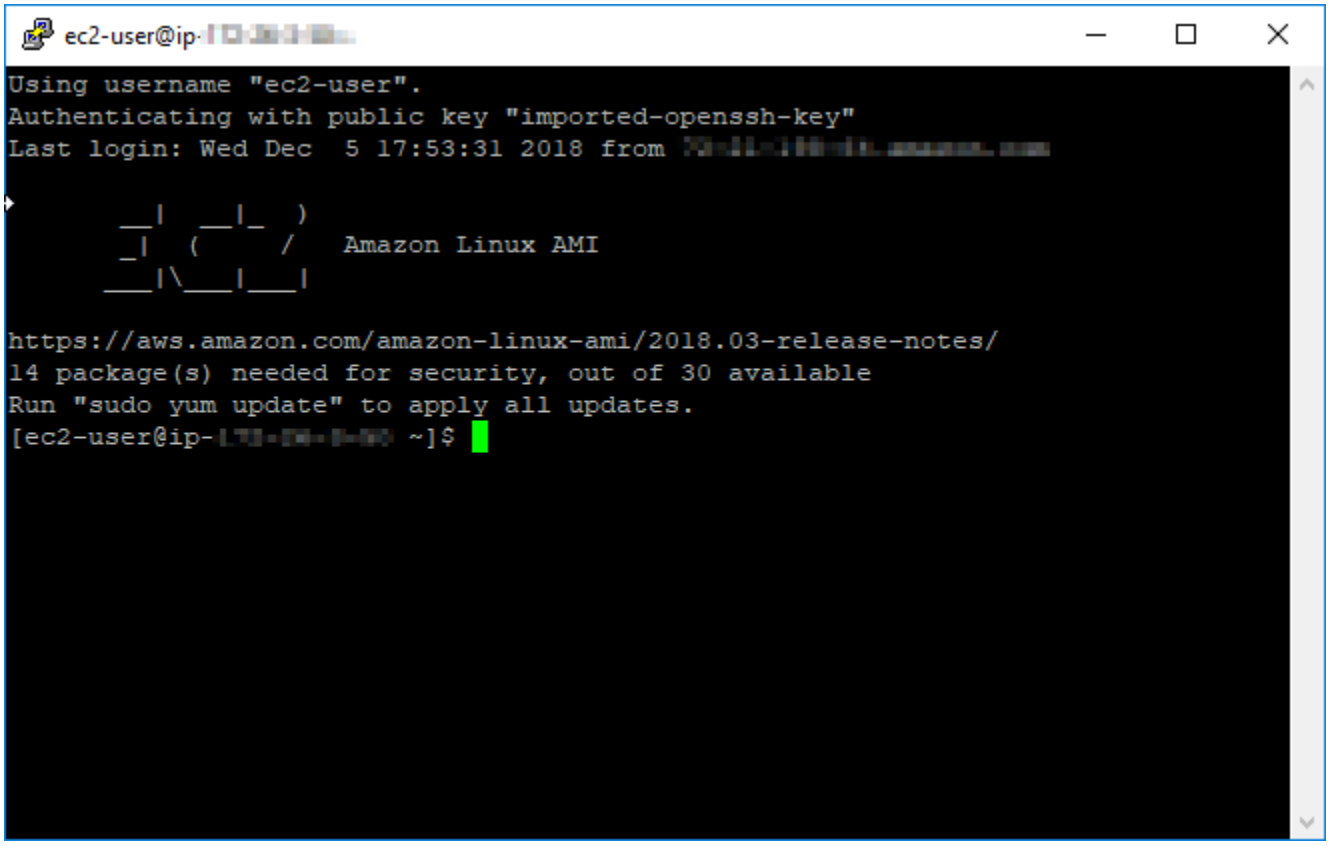
- AlmaLinux, Amazon Linux 2 및 Amazon Linux 2023, CentOS Stream 9, FreeBSD 및 openSUSE 인스턴스의 경우: `ec2-user`
- Debian 인스턴스: `admin`
- Ubuntu 인스턴스: `ubuntu`
- Bitnami 인스턴스: `bitnami`
- Plesk 인스턴스: `ubuntu`
- cPanel & WHM 인스턴스: `centos`

5. 왼쪽 탐색 창의 연결 섹션에서 SSH를 확장한 다음 Auth(인증)를 선택합니다.
6. Browse(찾아보기)를 선택하여 이 안내서의 이전 단원에서 생성한 .PPK 파일을 찾은 다음 Open(열기)을 선택합니다.



7. Open(열기)을 선택하여 인스턴스에 연결한 후 예를 선택하여 앞으로는 이 연결을 신뢰한다는 점을 확인합니다.

   인스턴스에 성공적으로 연결한 경우 다음과 비슷한 화면이 표시됩니다.



다음 단계

Amazon EC2를 사용하여 내보낸 스냅샷에서 새 인스턴스를 생성할 경우 Amazon EC2의 새 Linux 또는 Unix 인스턴스에는 Lightsail 서비스의 나머지 키가 포함됩니다. 새 Amazon EC2 인스턴스에 대한 보안을 강화하기 위해서는 이러한 키를 제거하는 것이 좋습니다. 자세한 내용은 Lightsail 스냅샷에서 생성된 Amazon EC2의 Linux 또는 Unix 인스턴스 보호를 참조하세요.

Lightsail 스냅샷에서 시작된 Amazon EC2 인스턴스 보호

Amazon Lightsail과 Amazon Elastic Compute Cloud(Amazon EC2)는 퍼블릭 키 암호화 기법을 사용하여 로그인 정보를 암호화하고 복호화합니다. 공개 키 암호화 기법은 공개 키를 사용하여 암호 등의 데이터를 암호화하고, 수신자가 개인 키를 사용하여 해당 데이터를 해독하는 방식입니다. 퍼블릭 키와 프라이빗 키를 키 페어라고 합니다.

Linux 또는 Unix Lightsail 인스턴스를 EC2로 내보낼 때 새 EC2 인스턴스에는 Lightsail 서비스의 나머지 키가 포함됩니다. 보안을 위해 인스턴스에서 사용하지 않은 키를 제거해야 합니다.

Lightsail 스냅샷에서 생성된 EC2의 Linux 또는 Unix 인스턴스 보안을 강화하기 위해 인스턴스를 생성한 후 다음 작업을 수행하는 것이 좋습니다.

- Lightsail 기본 키를 사용하여 Lightsail의 소스 인스턴스에 연결한 경우 해당 키를 제거하고 교체합니다. 자체 키를 사용하여 인스턴스에 연결한 경우 또는 Lightsail 콘솔에서 인스턴스에 대한 키를 생성한 경우에는 Lightsail 기본 키가 Amazon EC2 인스턴스에 제공되지 않습니다.
- `lightsail_instance_ca.pub` 키라고도 하는 Lightsail 시스템 키를 제거합니다. Linux 및 Unix 인스턴스에서 이 키를 사용하면 Lightsail 브라우저 기반 SSH 클라이언트를 연결할 수 있습니다. Lightsail 콘솔의 Amazon EC2 인스턴스 생성 페이지 또는 Lightsail API를 사용하여 EC2 인스턴스를 생성한 경우에는 `lightsail_instance_ca.pub` 키가 자동으로 제거됩니다.

목차

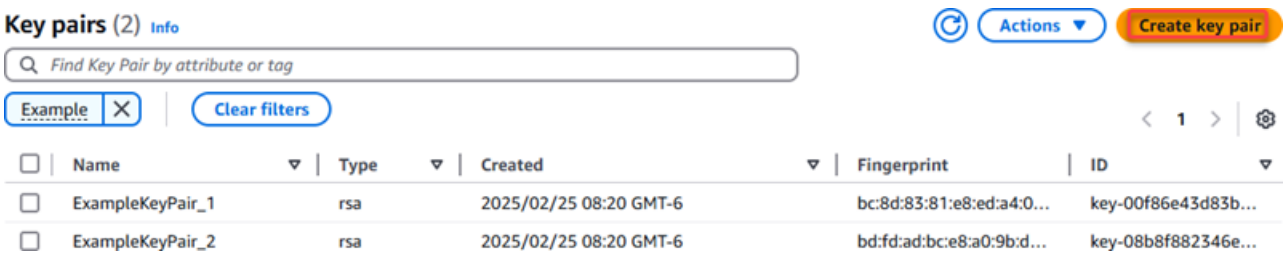
- Amazon EC2를 사용하여 프라이빗 키 생성
- PuTTYgen을 사용하여 퍼블릭 키 생성
- Amazon EC2의 Linux 또는 Unix 인스턴스에 연결
- 인스턴스에 퍼블릭 키 추가 및 연결 테스트
- Lightsail 기본 키 제거
- Lightsail 시스템 키 제거

Amazon EC2를 사용하여 프라이빗 키 생성

Amazon EC2 콘솔을 사용하여 Lightsail 기본 키 페어를 교체하는 데 사용할 수 있는 새 키 페어를 생성합니다.

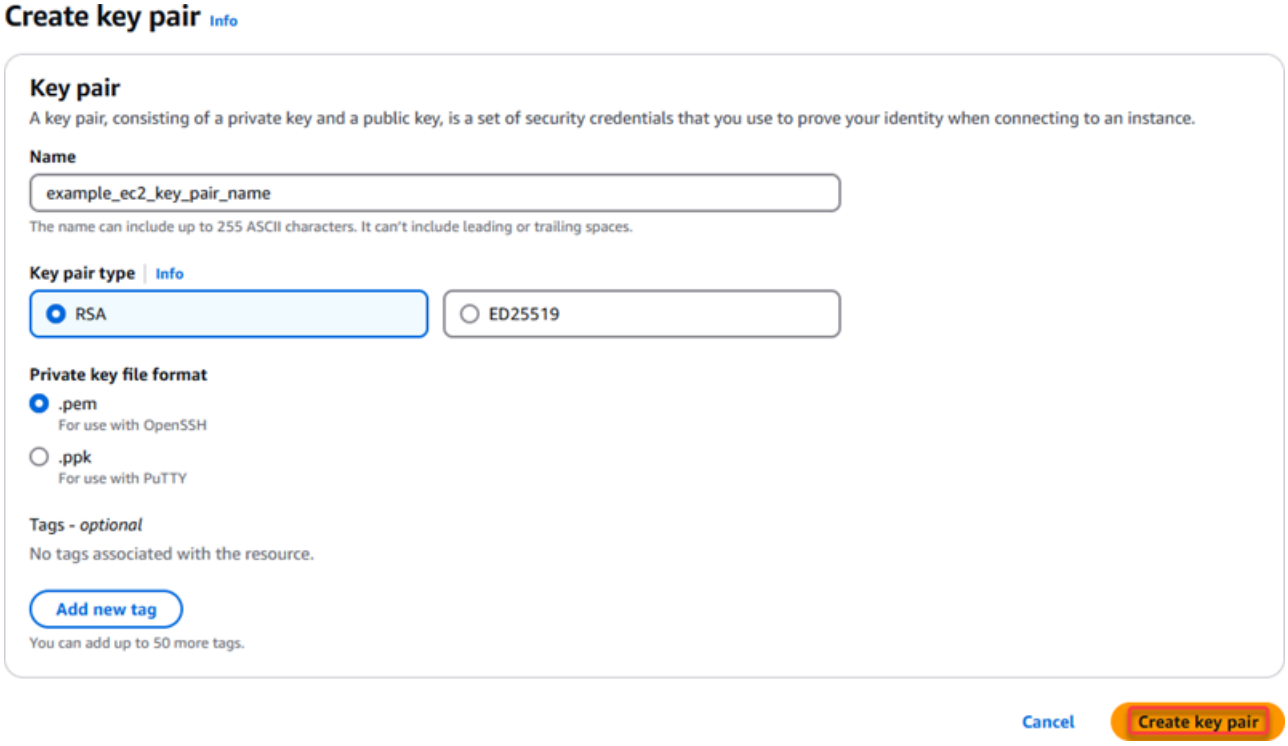
Amazon EC2를 사용하여 프라이빗 키 생성

1. Amazon EC2 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 Key Pairs(키 페어)를 선택합니다.
3. 키 페어 생성(Create key pair)를 선택합니다.



4. 키 페어 이름 텍스트 상자에 키 이름을 입력한 다음 키 페어 생성을 선택합니다. Amazon EC2에서의 키 페어 생성에 대한 자세한 내용은 Amazon Elastic Compute Cloud 사용 설명서의 Amazon EC2 인스턴스에 대한 키 페어 생성을 참조하세요.

새 프라이빗 키가 자동으로 다운로드됩니다. 프라이빗 키를 저장한 위치를 기록해 둡니다. 이 안내서의 다음 PuTTYgen을 사용하여 퍼블릭 키 생성 단원에서 퍼블릭 키를 생성할 때 필요합니다.



PuTTYgen을 사용하여 퍼블릭 키 생성

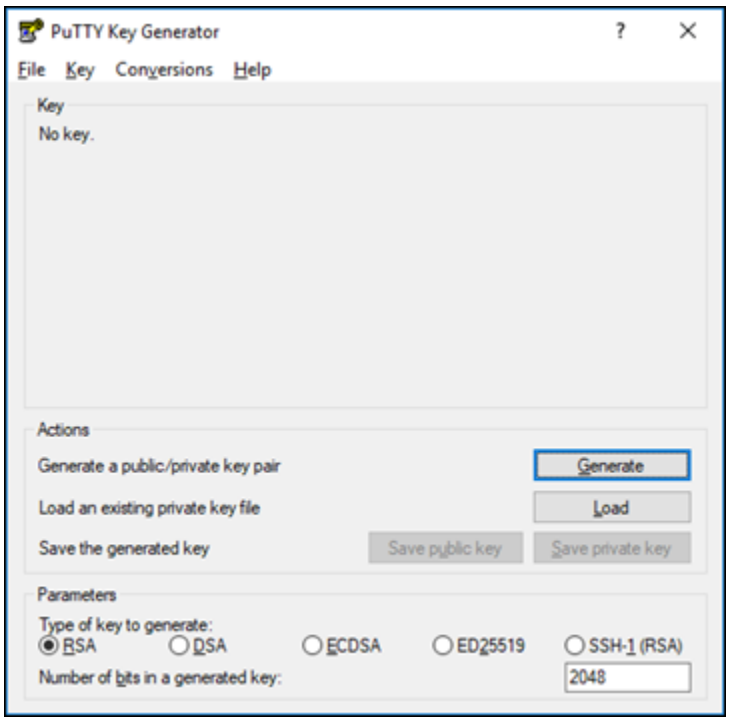
PuTTYgen은 PuTTY에 포함된 도구입니다. PuTTYgen을 사용하여 이 안내서의 뒷부분에서 인스턴스에 추가할 퍼블릭 키 텍스트를 생성합니다.

> **Note**
>
> PuTTY을 구성하여 Linux 또는 Unix 인스턴스에 연결하는 방법에 대한 자세한 내용은 Lightsail 스냅샷에서 생성된 Amazon EC2 Linux 또는 Unix 인스턴스에 연결을 참조하세요.

PuTTYgen을 사용하여 퍼블릭 키를 생성하려면

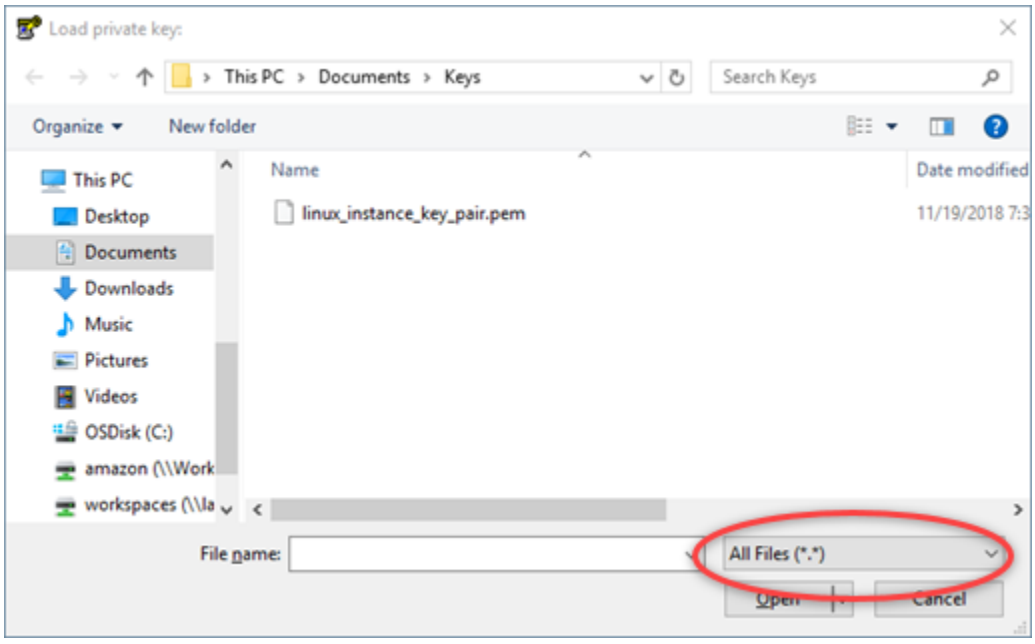
1. PuTTYgen을 시작합니다.

   예를 들어, Windows 시작 메뉴, 모든 프로그램, PuTTY 및 PuTTYgen을 차례대로 선택합니다.



2. 로드(Load)를 선택합니다.

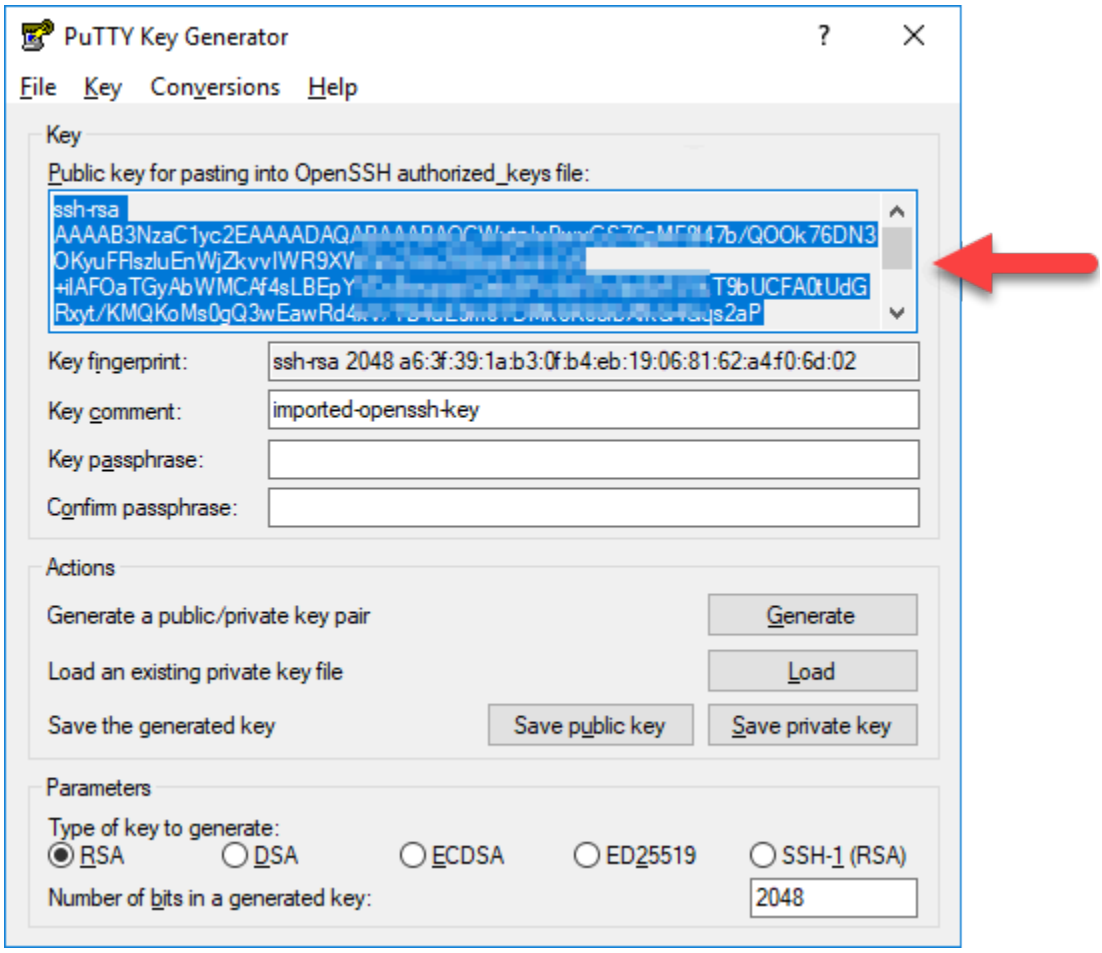
   기본적으로 PuTTYgen에는 확장명이 .PPK인 파일만 표시됩니다. .PEM 파일을 찾으려면 모든 유형의 파일을 표시하는 옵션을 선택합니다.



3. 이 안내서의 초반부에 생성한 프라이빗 키 위치로 이동합니다. 프라이빗 키를 선택한 다음 Open(열기)을 선택합니다.
4. PuTTYgen에서 키를 성공적으로 가져왔다고 확인하면 OK(확인)를 선택합니다.

5. Public key(퍼블릭 키) 텍스트 상자의 콘텐츠를 강조 표시하고 Windows를 사용할 경우 Ctrl+C를, macOS를 사용할 경우 Cmd+C를 눌러 클립보드에 복사합니다.

   Notepad 또는 TextEdit과 같은 텍스트 편집기를 열고 Windows를 사용할 경우 Ctrl+V를, macOS를 사용할 경우 Cmd+V를 눌러 퍼블릭 키 텍스트를 붙여넣습니다. 퍼블릭 키 텍스트가 있는 파일을 저장합니다. 이 안내서의 뒷부분에서 이 파일이 필요합니다.



6. 이 설명서의 Amazon EC2의 Linux 또는 Unix 인스턴스에 연결 섹션으로 계속 진행하여 EC2 인스턴스에 연결하고 퍼블릭 키를 추가합니다.

Amazon EC2의 Linux 또는 Unix 인스턴스에 연결

SSH를 통해 Amazon EC2의 Linux 또는 Unix 인스턴스에 연결하여 Lightsail 기본 키 및 시스템 키를 제거합니다. 자세한 내용은 Amazon Lightsail 스냅샷에서 생성된 Amazon EC2의 Linux 또는 Unix 인스턴스에 연결을 참조하세요.

Amazon EC2에서 인스턴스에 연결한 후에 이 설명서의 인스턴스에 퍼블릭 키 추가 및 연결 테스트 섹션을 이어서 진행합니다.

인스턴스에 퍼블릭 키 추가 및 연결 테스트

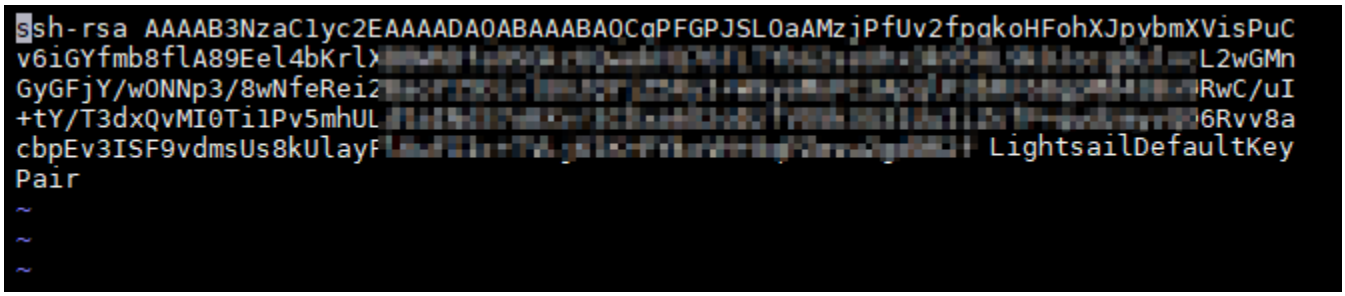
퍼블릭 키 콘텐츠는 Linux 및 Unix 인스턴스의 `~/.ssh/authorized_keys` 파일에 저장됩니다. 파일을 편집하여 Amazon EC2의 Linux 또는 Unix 인스턴스에서 Lightsail 기본 키를 제거하고 교체합니다.

인스턴스에 퍼블릭 키를 추가하고 연결을 테스트하려면

1. 인스턴스에 대한 SSH 연결을 설정한 후 Vim 텍스트 편집기에 다음 명령을 입력하여 `authorized_keys`을 편집합니다.

```
sudo vim ~/.ssh/authorized_keys
```

> **Note**
>
> 이 단계에서는 설명을 위해 Vim을 사용합니다. 그러나 이들 단계에서는 어떤 텍스트 편집기든 사용할 수 있습니다.



2. Vim 편집기에서 I 키를 눌러 삽입 모드를 시작합니다.
3. Lightsail 기본 키 뒤에 추가 행을 입력합니다.
4. 이 안내서의 초반부에 저장한 퍼블릭 키 텍스트를 복사하여 붙여넣습니다.

   결과는 다음과 같아야 합니다.

5. ESC 키를 누른 다음 `:wq!`를 입력하여 편집 내용을 저장하고 Vim을 종료합니다.
6. 다음 명령을 입력하여 개방형 SSH 서버를 다시 시작합니다.

```
sudo /etc/init.d/sshd restart
```

다음과 유사한 결과가 출력되어야 합니다.

```
[ec2-user@ip-172-26-11-173 ~]$ sudo /etc/init.d/sshd restart
Stopping sshd:                                             [  OK  ]
Starting sshd:                                             [  OK  ]
[ec2-user@ip-172-26-11-173 ~]$
```

이제 새 퍼블릭 키가 인스턴스에 추가됩니다. 새 키 페어를 테스트하려면 인스턴스에서 연결을 해제합니다. Lightsail 기본 키 대신 새 프라이빗 키를 사용하도록 PuTTY를 구성합니다. 새 키 페어를 사용하여 성공적으로 인스턴스에 연결할 수 있는 경우 이 안내서의 Lightsail 기본 키 제거 단원으로 계속 진행하여 Lightsail 기본 키를 제거합니다.

Lightsail 기본 키 제거

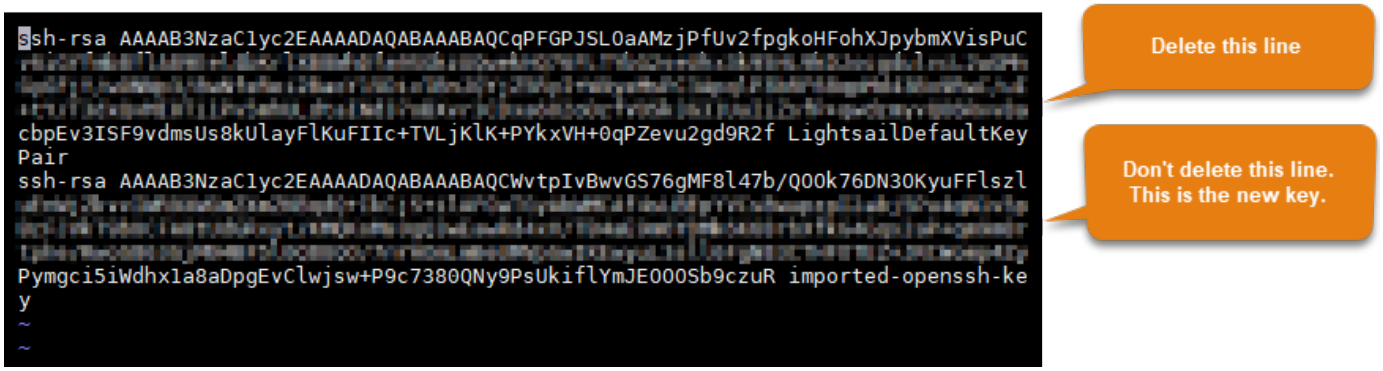
인스턴스에 새 퍼블릭 키를 추가하고 새 키 페어를 사용하여 성공적으로 연결한 후에는 Lightsail 기본 키를 제거합니다.

Lightsail 기본 키를 제거하려면

1. 인스턴스에 대한 SSH 연결을 설정한 후 Vim 텍스트 편집기에 다음 명령을 입력하여 `authorized_keys file`을 편집합니다.

```
sudo vim ~/.ssh/authorized_keys
```

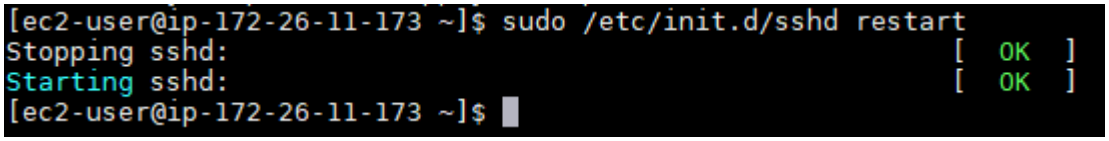
2. Vim 편집기에서 `I` 키를 눌러 삽입 모드를 시작합니다.
3. `LightsailDefaultKeyPair`로 끝나는 행을 삭제합니다. 이것이 Lightsail 기본 키입니다.



4. ESC 키를 누른 다음 `:wq!`를 입력하여 편집 내용을 저장하고 Vim을 종료합니다.
5. 다음 명령을 입력하여 개방형 SSH 서버를 다시 시작합니다.

```
sudo /etc/init.d/sshd restart
```

다음과 유사한 결과가 출력되어야 합니다.



이제 Lightsail 기본 키가 인스턴스에서 제거되었습니다. 이제 인스턴스에서 Lightsail 기본 키를 사용하는 연결을 거부합니다. 이 안내서의 Lightsail 시스템 키 제거 단원으로 계속 진행하여 Lightsail 시스템 키를 제거합니다.

Lightsail 시스템 키 제거

Linux 및 Unix 인스턴스에서 `lightsail_instance_ca.pub` 키라고도 하는 Lightsail 시스템 키를 사용하면 Lightsail 브라우저 기반 SSH 클라이언트를 연결할 수 있습니다. 다음 단계를 수행하여 Amazon EC2의 Linux 또는 Unix 인스턴스에서 `lightsail_instance_ca.pub` 키를 제거하고 `/etc/ssh/sshd_config` 파일을 편집합니다. `/etc/ssh/sshd_config` 파일은 인스턴스에 대한 SSH 연결 파라미터를 정의합니다.

Lightsail 시스템 키를 제거하려면

1. 인스턴스에 연결된 SSH 터미널 창에 다음 명령을 입력하여 `lightsail_instance_ca.pub` 키를 제거합니다.

```
sudo rm -r /etc/ssh/lightsail_instance_ca.pub
```

2. Vim 텍스트 편집기에 다음 명령을 입력하여 `sshd_config` 파일을 편집합니다.

```
sudo vim /etc/ssh/sshd_config
```

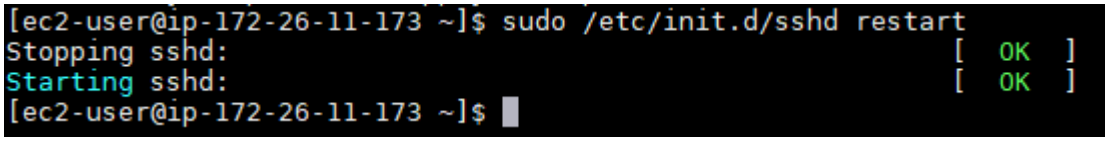
3. Vim 편집기에서 I 키를 눌러 삽입 모드를 시작합니다.
4. 파일에 다음 텍스트가 있는 경우 삭제합니다.

```
TrustedUserCAKeys /etc/ssh/lightsail_instance_ca.pub
```

5. ESC 키를 누른 다음 `:wq!`를 입력하여 편집 내용을 저장하고 Vim을 종료합니다.
6. 다음 명령을 입력하여 개방형 SSH 서버를 다시 시작합니다.

```
sudo /etc/init.d/sshd restart
```

다음과 유사한 결과가 출력되어야 합니다.



이제 `lightsail_instance_ca.pub` 키가 인스턴스에서 제거되었습니다. 해당 키를 제외하도록 연결된 `sshd_config` 파일이 업데이트됩니다.

Lightsail 스냅샷에서 생성된 Amazon EC2의 Windows Server에 연결

Amazon Elastic Compute Cloud(Amazon EC2)에 새 Windows Server 인스턴스가 생성되면 RDP(원격 데스크톱 프로토콜)를 사용하여 해당 인스턴스에 연결할 수 있습니다. 이는 소스 Amazon Lightsail 인스턴스에 연결하는 방법과 비슷합니다. 소스 인스턴스의 기본 Lightsail 키 페어를 사용하여 EC2 인스턴스에 연결합니다 AWS 리전. 이 안내서에서는 Microsoft 원격 데스크톱 연결을 사용하여 Windows Server 인스턴스에 연결하는 방법을 보여줍니다.

> **Note**
>
> Linux 또는 Unix 인스턴스에 연결하는 방법에 대한 자세한 내용은 Lightsail 스냅샷에서 생성된 Amazon EC2의 Linux 또는 Unix 인스턴스에 연결을 참조하세요.

목차

- 인스턴스의 키 가져오기
- 인스턴스에 대한 퍼블릭 DNS 주소 가져오기
- Windows Server 인스턴스에 대한 암호 가져오기
- 원격 데스크톱 연결을 구성하여 Windows Server 인스턴스에 연결
- 다음 단계

인스턴스의 키 가져오기

Amazon EC2의 Windows Server 인스턴스는 소스 인스턴스의 리전에 대한 기본 Lightsail 키 페어를 사용하여 기본 관리자 암호를 검색합니다.

Lightsail 계정 페이지의 SSH 키(SSH keys) 탭에서 기본 프라이빗 키를 다운로드합니다. 기본 Lightsail SSH 키에 대한 자세한 내용은 SSH 키 페어를 참조하세요.

> **Note**
>
> EC2 인스턴스에 연결한 후 Amazon EC2의 Windows Server 인스턴스에 대한 관리자 암호를 변경하는 것이 좋습니다. 이는 기본 Lightsail 키 페어와 Amazon EC2의 Windows Server 인스턴스 간 연결을 제거합니다. 자세한 내용은 Lightsail 스냅샷에서 생성된 Amazon EC2 Windows Server 인스턴스 보호를 참조하세요.

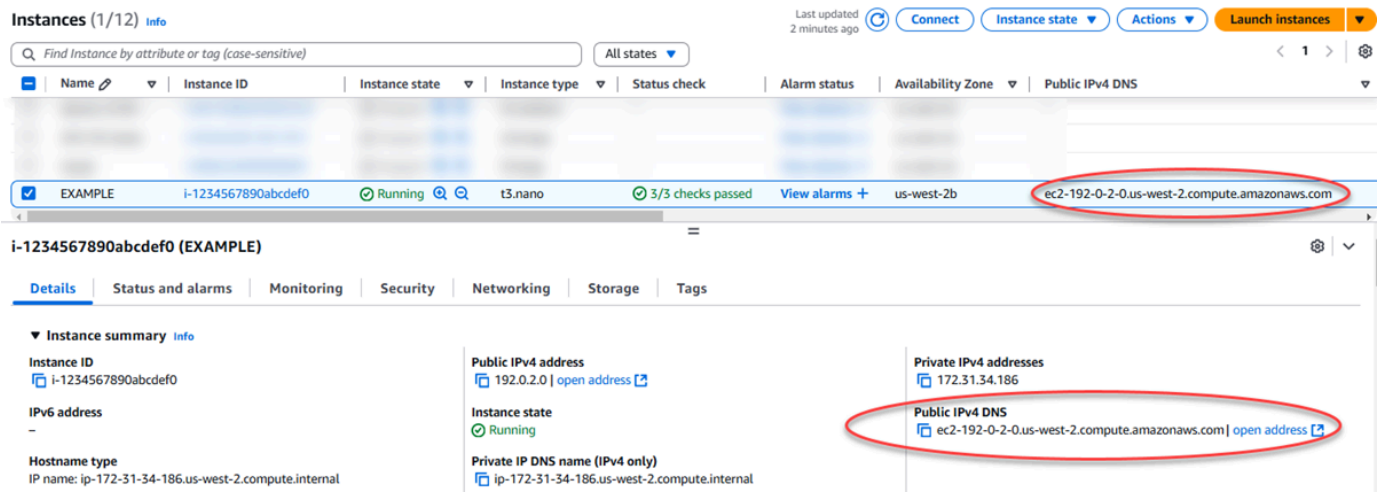
인스턴스에 대한 퍼블릭 DNS 주소 가져오기

Microsoft 원격 데스크톱 연결과 같은 RDP 클라이언트를 구성하여 인스턴스에 연결할 때 Amazon EC2 인스턴스에 대한 퍼블릭 DNS 주소를 사용할 수 있도록 해당 주소를 가져옵니다.

인스턴스에 대한 퍼블릭 DNS 주소를 가져오려면

1. Amazon EC2 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 인스턴스를 선택합니다.
3. 연결하려는 실행 중인 Windows Server 인스턴스를 선택합니다.
4. 하단 창에서 인스턴스에 대한 퍼블릭 DNS 주소를 찾습니다.

   이는 RDP 클라이언트를 구성하여 인스턴스에 연결할 때 사용할 주소입니다. 이 설명서의 Windows Server 인스턴스에 대한 암호 가져오기로 계속 진행하여 Amazon EC2의 Windows Server 인스턴스에 대한 기본 관리자 암호를 가져오는 방법에 대해 알아봅니다.

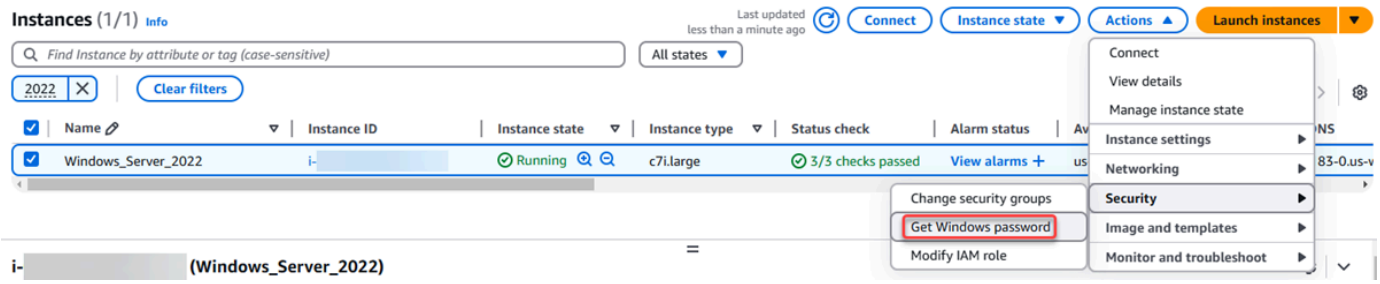


Windows Server 인스턴스에 대한 암호 가져오기

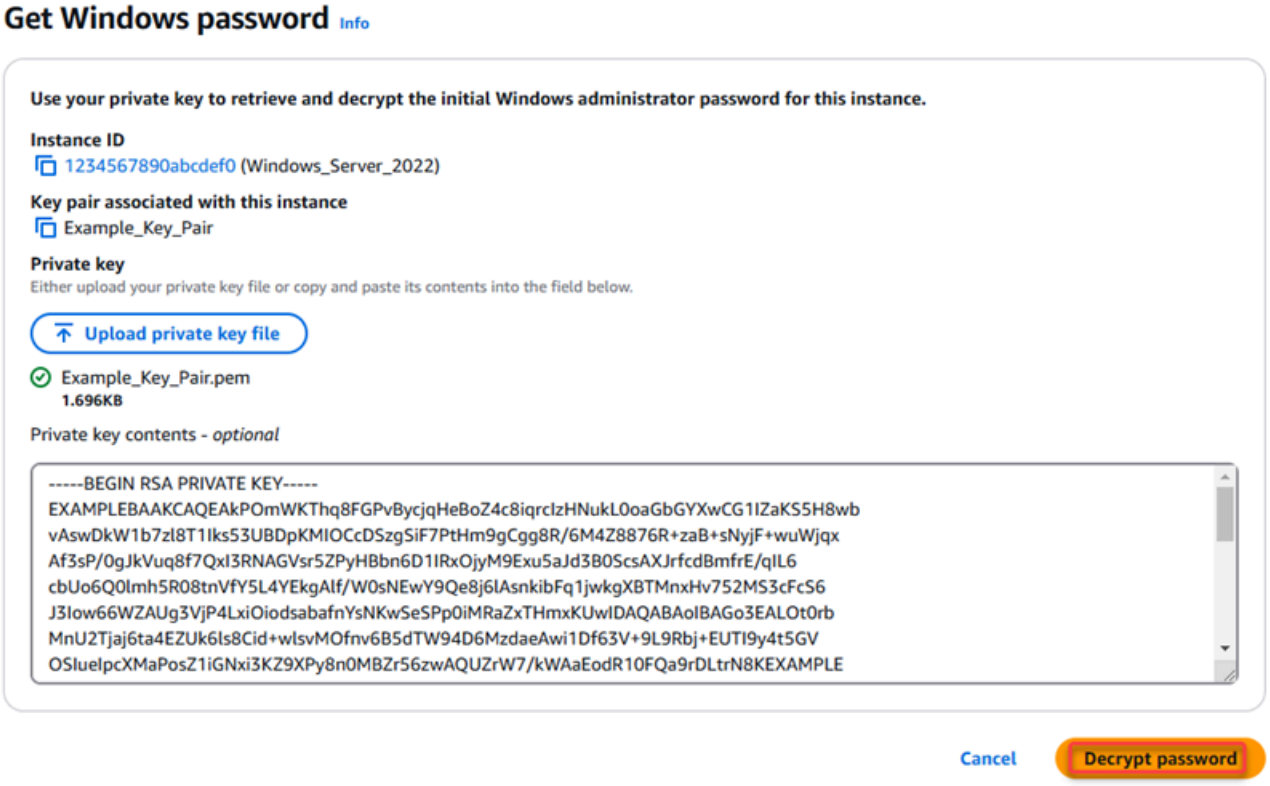
Amazon EC2 콘솔에서 Windows Server 인스턴스에 대한 암호를 가져옵니다. RDP를 통해 Windows Server 인스턴스에 연결할 때 해당 인스턴스에 로그인하려면 이 암호가 필요합니다.

Windows Server 인스턴스에 대한 암호를 가져오려면

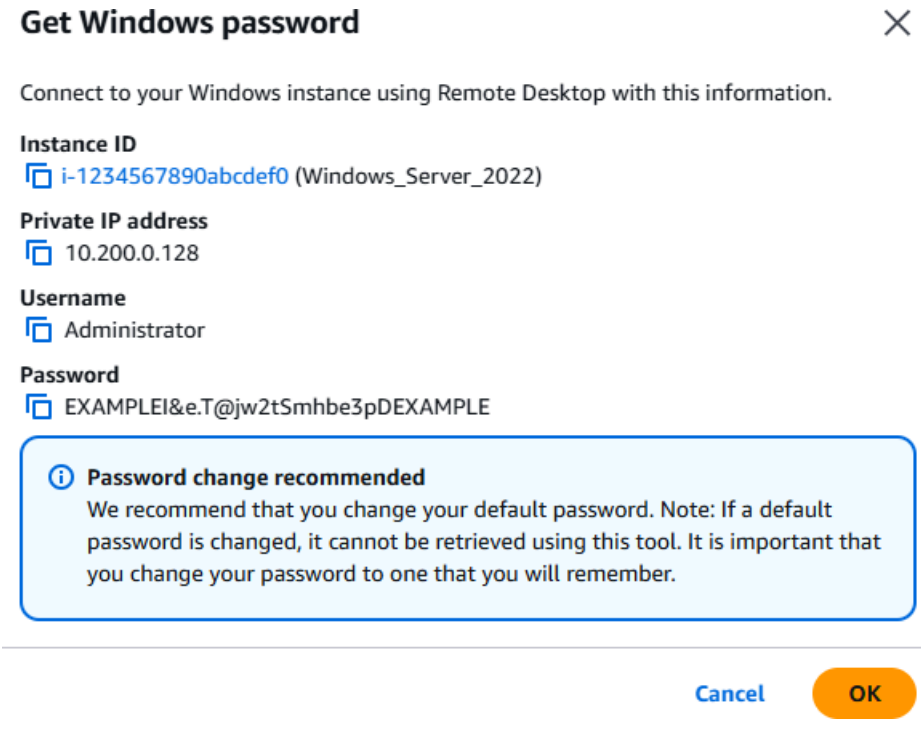
1. Amazon EC2 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 인스턴스를 선택합니다.
3. 연결하려는 Windows Server 인스턴스를 선택합니다.
4. 작업에서 보안, Windows 암호 가져오기를 선택합니다.



5. 프롬프트에서 Browse(찾아보기)를 선택하고 이 안내서의 초반부에서 Lightsail로부터 다운로드한 기본 프라이빗 키 파일을 엽니다.
6. 암호 해독을 선택합니다.



암호, 사용자 이름 및 프라이빗 IP 주소가 표시됩니다. 이 안내서의 다음 원격 데스크톱 연결을 구성하여 Windows Server 인스턴스에 연결 단원에서 암호를 사용할 수 있도록 암호를 클립보드에 복사합니다. 암호를 강조 표시하고 Windows를 사용하고 있는 경우 Ctrl+C를, macOS를 사용하고 있는 경우 Cmd+C를 누릅니다.



이 설명서의 원격 데스크톱 연결을 구성하여 Windows Server 인스턴스에 연결 섹션으로 계속 진행합니다. 이 섹션에서는 원격 데스크톱 연결을 구성하여 Amazon EC2의 Windows Server 인스턴스에 연결하는 방법에 대해 알아봅니다.

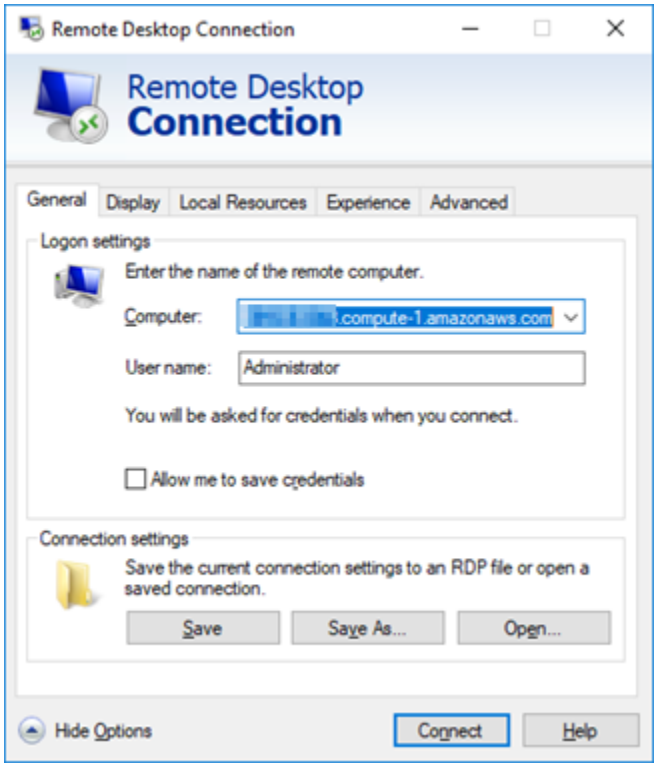
원격 데스크톱 연결을 구성하여 Windows Server 인스턴스에 연결

원격 데스크톱 연결은 대부분의 Windows 운영 체제에 사전 설치되어 있는 RDP 클라이언트입니다. 이를 사용하여 Amazon EC2의 Windows Server 인스턴스에 그래픽 방식으로 연결합니다.

원격 데스크톱 연결을 구성하여 Windows Server 인스턴스에 연결하려면

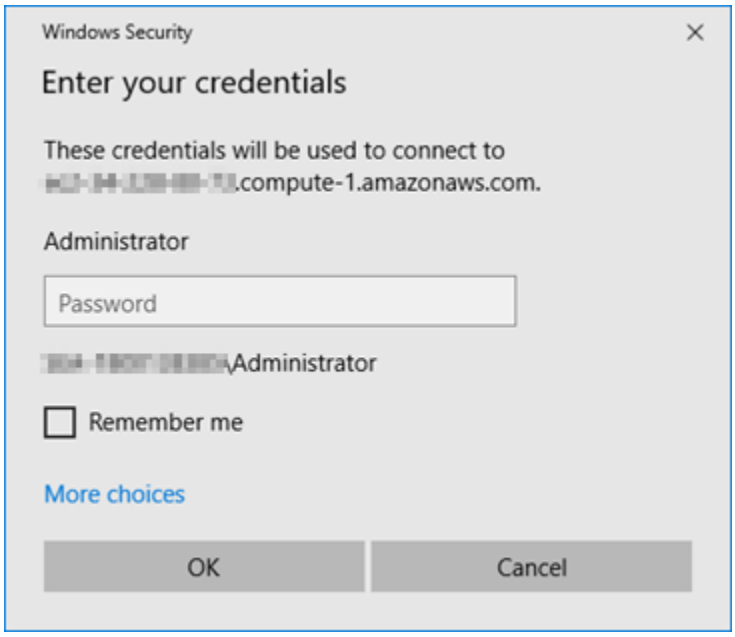
1. 원격 데스크톱 연결을 엽니다.

   예를 들어, Windows 시작 메뉴를 선택한 다음 Remote Desktop Connection(원격 데스크톱 연결)을 검색합니다.
2. 컴퓨터 텍스트 상자에 이 설명서의 초반부에서 가져온 Amazon EC2의 Windows Server 인스턴스에 대한 퍼블릭 DNS 주소를 입력합니다.
3. Show Options(옵션 표시)를 선택하여 추가 옵션을 확인합니다.
4. 사용자 이름(User name) 텍스트 상자에 `Administrator`를 입력합니다.

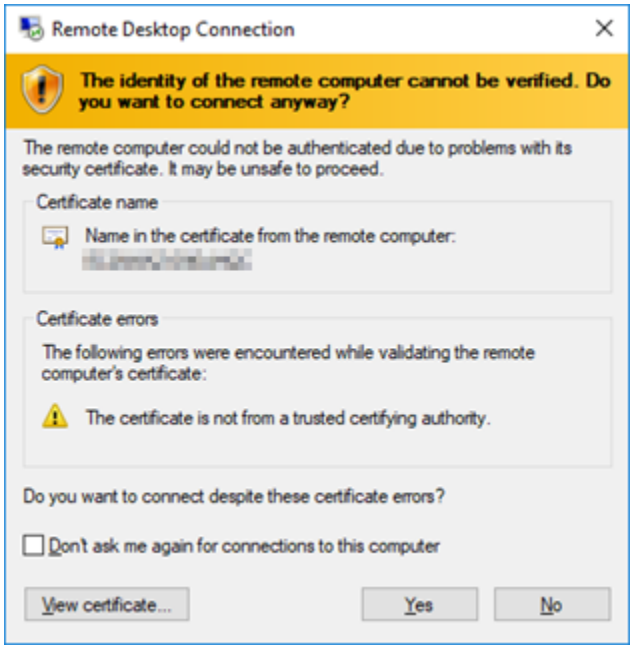


5. 연결을 선택하여 Windows Server 인스턴스에 연결합니다.

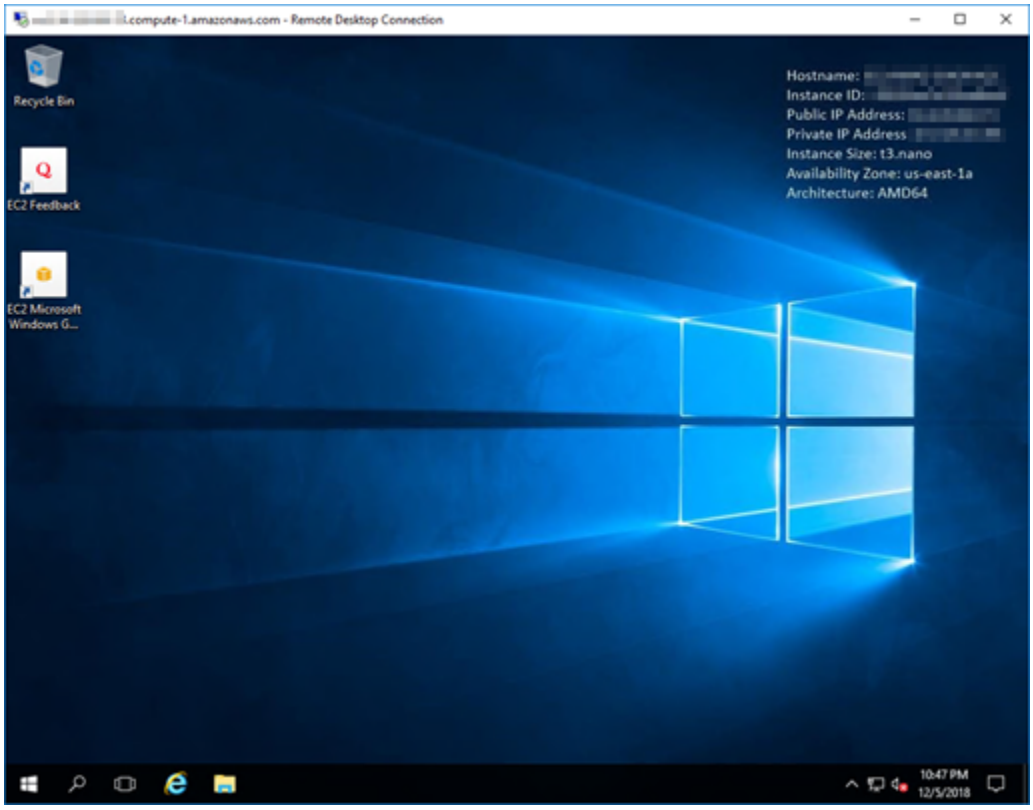
6. Windows Security 프롬프트의 Password(암호) 텍스트 상자에 Windows Server 인스턴스에 대한 암호를 입력한 다음 OK(확인)를 선택합니다.



7. 원격 데스크톱 연결 프롬프트에서 예를 선택하여 연결합니다.



인스턴스에 성공적으로 연결한 경우 다음과 비슷한 화면이 표시됩니다.



다음 단계

Amazon EC2의 Windows Server 인스턴스에 대한 관리자 암호를 변경하는 것이 좋습니다. 이는 기본 Lightsail 키 페어와 Amazon EC2의 Windows Server 인스턴스 간 연결을 제거합니다. 자세한 내용은 Lightsail 스냅샷에서 생성된 Amazon EC2의 Windows Server 인스턴스 보호를 참조하세요.

Lightsail 스냅샷에서 시작된 보안 Windows Server Amazon EC2 인스턴스

Amazon Lightsail 스냅샷에서 생성된 Amazon Elastic Compute Cloud(Amazon EC2)에서 Windows Server 인스턴스의 보안을 강화하려면 기본 관리자 암호를 변경하는 것이 좋습니다. 이렇게 하면 Lightsail 키 페어와 Amazon EC2의 새 Windows Server 인스턴스 간 연결이 제거됩니다.

> **Note**
>
> Lightsail 스냅샷에서 Amazon EC2에 Linux 또는 Unix 인스턴스를 생성한 경우 이러한 인스턴스를 보호하기 위해 몇 가지 단계를 수행해야 합니다. 자세한 내용은 Lightsail 스냅샷에서 생성된 Amazon EC2 Linux 또는 Unix 인스턴스 보호를 참조하세요.

목차

- Amazon EC2의 Windows Server 인스턴스에 연결
- Amazon EC2의 Windows Server 인스턴스에 대한 기본 관리자 암호 변경

Amazon EC2의 Windows Server 인스턴스에 연결

Windows Server 관리자 암호를 변경하려면 RDP(Remote Desktop Protocol)를 사용하여 Amazon EC2의 Windows Service 인스턴스에 연결합니다. 인스턴스에 연결하는 방법은 Lightsail 스냅샷에서 생성된 Amazon EC2의 Windows Server 인스턴스에 연결을 참조하세요.

Amazon EC2에서 인스턴스에 연결한 후에 이 설명서의 Amazon EC2의 Windows Server 인스턴스에 대한 기본 관리자 암호 변경 섹션을 이어서 진행합니다.

Amazon EC2의 Windows Server 인스턴스에 대한 기본 관리자 암호 변경

Windows Server 인스턴스의 기본 암호를 변경하여 Amazon EC2의 Lightsail 키 페어와 새 Windows Server 인스턴스 간의 연결을 제거하세요.

Amazon EC2의 Windows Server 인스턴스에 대한 기본 관리자 암호 변경

1. 인스턴스에 대한 RDP 연결을 설정한 후 명령 프롬프트를 열고 다음 명령을 입력합니다.

   ```
   net user Administrator "Password"
   ```

   명령에서 *Password*를 새 암호로 바꿉니다.

   예:

   ```
   net user Administrator "EXAMPLE%4=Bwk^GEAg8$u@5"
   ```

   다음과 유사한 결과가 출력되어야 합니다.

   ```
   C:\users\Administrator>net user Administrator "EXAMPLE%4=Bwk^GEAg8$u@5"
   The command completed successfully.


   C:\users\Administrator>
   ```

2. 새 암호를 안전한 위치에 저장합니다. Amazon EC2 콘솔을 사용하여 새 암호를 검색할 수 없습니다. 콘솔에서는 기본 암호만 검색할 수 있습니다. 암호를 변경한 이후에 기본 암호를 사용하여 인스턴스에 연결할 경우 자격 증명이 작동하지 않는다는 오류 메시지가 나타납니다.

암호가 기억나지 않거나 만료된 경우 새 암호를 생성할 수 있습니다. 암호 재설정 절차는 Amazon EC2 설명서의 기억나지 않거나 만료된 Windows 관리자 암호 재설정을 참조하세요.

Lightsail 인스턴스에 대한 CloudFormation 스택 보기

Amazon Lightsail는 CloudFormation 를 사용하여 내보낸 스냅샷에서 Amazon Elastic Compute Cloud(Amazon EC2) 인스턴스를 생성합니다. Lightsail 콘솔 또는 Lightsail API를 사용하여 Amazon EC2 인스턴스를 생성하도록 요청할 때 CloudFormation 스택이 생성됩니다. 이 스택은 Amazon Web Services(AWS) 계정에서 다양한 작업을 수행하여 Amazon Machine Image(AMI)의 Amazon EC2 인스턴스, Elastic Block Store(EBS) 스냅샷의 EBS 시스템 볼륨, 인스턴스에 대한 보안 그룹 등 인스턴스와 관련된 모든 리소스를 생성합니다. CloudFormation 스택에 대한 자세한 내용은 CloudFormation 설명서의 스택 작업을 참조하세요.

콘솔 또는 Lightsail 콘솔을 통해 CloudFormation 스택에 액세스할 수 있습니다 CloudFormation . 이 안내서는 두 액세스 방법을 모두 보여줍니다.

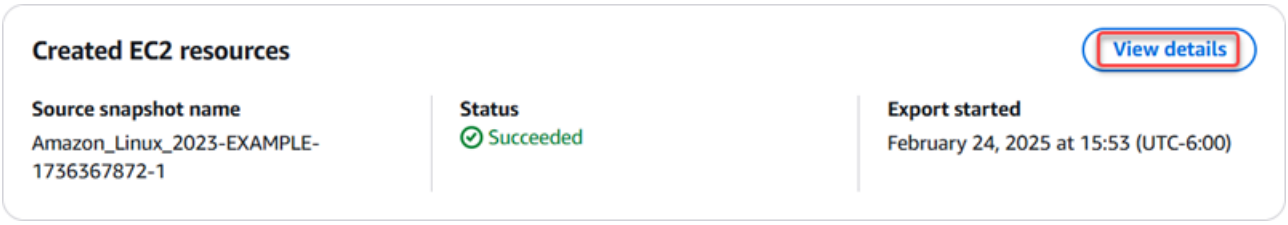
> **Note**
>
> Amazon EC2 리소스를 생성하는 데 사용되는 CloudFormation 스택은 Amazon EC2 리소스에 영구적으로 연결됩니다. 스택을 삭제하면 관련된 모든 리소스도 자동으로 삭제됩니다. 따라서 에서 생성한 CloudFormation 스택을 삭제해서는 안 되며Lightsail, 대신 Amazon EC2 EC2 리소스를 삭제해야 합니다.

Lightsail 콘솔을 통해 CloudFormation 스택에 액세스

Lightsail 콘솔 또는 Lightsail API를 사용하여 Amazon EC2에서 인스턴스를 생성하도록 선택하면 CloudFormation 스택이 생성되고 Lightsail 콘솔의 내보내기 섹션에서 해당 상태가 추적됩니다. 내보내기에 대한 자세한 내용은 Lightsail에서 스냅샷 내보내기 상태 추적 섹션을 참조하세요.

Lightsail 콘솔에서 CloudFormation 스택을 보려면

1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 내보내기를 선택합니다.
3. 이전에 생성된 Amazon EC2 인스턴스의 CloudFormation 스택에 액세스하려면 생성된 EC2 리소스로 레이블이 지정된 작업의 세부 정보 보기를 선택합니다.

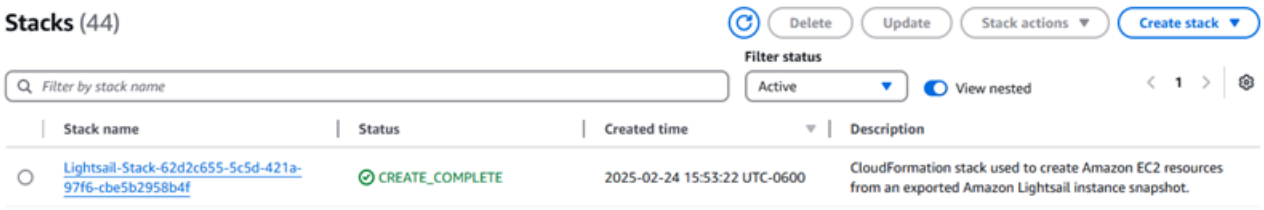


4. 나타나는 확인 페이지에는 작업에 대한 CloudFormation 스택이 나열됩니다. 스택 이름을 선택하여 CloudFormation 콘솔에서 스택 세부 정보를 엽니다.

CloudFormation 콘솔에서 스택에 액세스

또한 CloudFormation 콘솔을 통해 스택 세부 정보에 액세스할 수도 있습니다. 다음 스크린샷에 표시된 대로 Lightsail에서 생성된 스택은 'Lightsail-stack'으로 시작하며, 'Amazon EC2 리소스를 생성하는 데 사용된 CloudFormation 스택'이라는 설명이 있습니다.

CREATE_IN_PROGRESS 상태의 스택은 내보낸 Lightsail 스냅샷에서 Amazon EC2 리소스를 생성하고 있는 중임을 나타냅니다. CREATE_COMPLETED 상태의 스택은 Amazon EC2 리소스의 생성을 완료했음을 나타냅니다. 스택에서 생성된 리소스를 보려면 스택 이름 옆에 있는 확인란을 선택한 다음 리소스 탭을 선택합니다.



Lightsail에서 웹 사이트의 도메인 등록 및 관리

웹 사이트의 이름(예: `example.com`)이 필요합니다. Amazon Lightsail을 통해 웹사이트의 이름, 즉 도메인 이름을 등록할 수 있습니다. 웹 사이트에 액세스하려면 사용자가 웹 브라우저에 도메인 이름을 입력합니다.

Amazon Lightsail 콘솔의 도메인 및 DNS 탭을 사용하여 도메인 이름을 등록하고 관리합니다. Lightsail에서는 가용성과 확장성이 우수한 도메인 이름 시스템(DNS) 웹 서비스인 Amazon Route 53을 사용하여 도메인을 등록합니다. 도메인이 등록되면 도메인을 Lightsail 리소스에 할당하거나 DNS 레코드를 관리할 수 있습니다. DNS에 대한 자세한 내용은 DNS를 참조하세요.

Amazon Lightsail에서의 도메인 등록에 대한 자세한 내용은 계속 읽어서 확인하세요.

내용

- 도메인 등록 방식
- 에 등록할 수 있는 도메인Lightsail
- 도메인 등록 요금

도메인 등록 방식

다음 개요에서는 Amazon Lightsail에서 도메인 이름을 등록하는 방법을 보여줍니다.

1. 원하는 도메인 이름이 인터넷에서 사용 가능한지 확인합니다. 원하는 도메인 이름을 사용할 수 없는 경우 다른 이름을 사용하거나 .com과 같은 최상위 도메인을 .org 또는 .net과 같은 다른 최상위 도메인으로 변경할 수 있습니다. Lightsail에서 지원하는 최상위 도메인(TLD) 목록은 Amazon Lightsail에 등록할 수 있는 도메인을 참조하세요.
2. Lightsail에 도메인 이름을 등록합니다. 도메인을 등록할 때는 도메인 소유자 및 다른 연락처의 이름과 연락처 정보를 제공합니다.

등록 프로세스가 끝나면 도메인 등록 대행자에 제공한 정보를 전송합니다. 도메인 등록 대행자는 국제 인터넷주소관리기구(ICANN)가 특정 TLD에 대한 도메인 등록 처리를 인증하는 회사입니다. 도메인 등록 대행자는 Amazon Registrar 또는 등록 대행 협력사 Gandi입니다.

Amazon Registrar와 Gandi는 기본적으로 다른 정보를 숨깁니다. Amazon Registrar, Inc.는 모든 연락처 정보를 숨기고, Gandi는 조직 이름을 제외한 모든 연락처 정보를 숨깁니다.

- 도메인의 등록 대행자를 확인하려면 Amazon Lightsail에 등록할 수 있는 도메인을 참조하세요.
- 등록 대행자는 사용자 정보를 도메인의 등록 기관으로 전송합니다. 등록 기관은 .com과 같은 하나 이상의 최상위 도메인의 도메인 등록을 판매하는 회사입니다.
- 등록 기관은 자체 데이터베이스에 사용자의 도메인에 관한 정보를 저장하고 일부 정보는 퍼블릭 WHOIS 데이터베이스에도 저장합니다.

도메인 이름을 등록하는 방법에 대한 자세한 내용은 새 도메인 등록을 참조하세요.

Lightsail을 사용하여 도메인을 등록한 후 Route 53은 도메인에 이름 서버 세트를 할당하여 도메인에 대한 DNS 서비스가 됩니다. 이름 서버는 도메인 이름을 IP 주소로 변환하는 데 도움을 주는 서버입니다.

Lightsail에서는 다음을 통해 도메인에 대한 DNS 서비스가 됩니다.

- 도메인과 이름이 같은 Lightsail DNS 영역을 생성합니다.
- Lightsail DNS 영역에 4개의 이름 서버 세트를 할당합니다.
- 도메인의 Route 53 이름 서버를 Lightsail DNS 영역의 이름 서버로 대체합니다.

다른 등록 대행자에 이미 도메인 이름을 등록한 경우 도메인 DNS의 관리를 Lightsail로 이전할 수 있습니다. 다른 Lightsail 기능을 사용하는 데는 필요하지 않습니다. 자세한 내용은 DNS 영역을 생성하여 도메인의 DNS 레코드 관리를 참조하세요.

에 등록할 수 있는 도메인Lightsail

Lightsail에서는 Route 53와 동일한 일반적인 최상위 도메인(TLD)을 사용합니다. Lightsail에 도메인을 등록하는 데 사용할 수 있는 일반 TLD 목록은 Amazon Route 53 개발자 안내서의 Amazon Route 53에 등록할 수 있는 도메인을 참조하세요.

TLD가 목록에 포함되어 있지 않거나 지리적 도메인을 등록하려면 Route 53 콘솔을 사용하는 것이 좋습니다. Route 53를 사용하여 등록된 후 Lightsail 콘솔에서 지리적 도메인을 사용할 수 있습니다. 자세한 내용은 Amazon Route 53 개발자 안내서의 지리적 최상위 도메인을 참조하세요.

도메인 등록 요금

Lightsail은 Route 53를 도메인 등록에 사용합니다. 따라서 Route 53 요금은 Lightsail 등록에도 적용됩니다.

도메인 등록 비용에 대한 자세한 내용은 Amazon Route 53 개발자 안내서의 Amazon Route 53에 등록할 수 있는 도메인을 참조하세요.

도메인에 대한 추가 정보

다음 문서는 Lightsail에서 도메인을 관리하는 데 도움이 될 수 있습니다.

- DNS
- 도메인 이름 형식 지정
- Amazon Route 53에서 Lightsail 도메인 관리
- DNS 영역을 생성하여 도메인의 DNS 레코드 관리
- 도메인 등록 갱신
- DNS 영역 편집 또는 삭제
- 로드 밸런서로 도메인 연결
- 배포로 도메인 연결
- 인스턴스로 도메인 연결
- 도메인의 트래픽을 컨테이너 서비스로 라우팅

Lightsail의 DNS 이해

다른 사용자가 IPv4 또는 IPv6 주소일 수 있는 인스턴스의 퍼블릭 인터넷 프로토콜(IP) 주소를 방문하여 Lightsail 인스턴스의 웹 애플리케이션에 액세스할 수 있습니다. 하지만 IP 주소는 복잡하고 기억하기 어렵기 때문에 사람들이 기억하기 쉬운 도메인 이름(예: `example.com`)으로 인스턴스의 웹 애플리케이션에 액세스할 수 있어야 합니다. 등록된 도메인 이름을 IP 주소로 매핑하는 디렉터리 역할을 하는 도메인 이름 시스템(DNS)을 이용하면 가능합니다.

사용자의 도메인 이름에 대한 트래픽을 Lightsail 인스턴스로 라우팅하려면, 사용자의 도메인 이름을 인스턴스의 고정 IPv4 주소로 연결하는 주소(A) 레코드 또는 인스턴스의 IPv6 주소로 연결하는 AAAA 레코드를 추가합니다. Lightsail을 사용하여 도메인 이름을 등록한 경우 도메인 이름을 등록할 때 생성된 DNS 영역에서 DNS 레코드를 관리할 수 있습니다. 도메인이 다른 등록 대행자를 통해 등록된 경우 등록 대행자에서 DNS 레코드를 관리하거나 도메인 DNS 관리를 Lightsail로 이전할 수 있습니다.

도메인 이름을 Lightsail 인스턴스에 쉽게 매핑할 수 있게 하려면 DNS 영역을 만들어 도메인 DNS 레코드의 관리를 Lightsail로 이전하는 것이 좋습니다. 자세한 내용은 DNS 영역을 생성하여 도메인의 DNS 레코드 관리를 참조하세요. Lightsail에서 최대 6개의 DNS 영역을 생성할 수 있습니다. 6개 이상의 DNS 영역이 필요한 경우 Route 53를 사용하여 모든 도메인의 DNS를 관리하는 것이 좋습니다.

Route 53를 사용하여 도메인 이름을 Lightsail 인스턴스로 지정할 수 있습니다. Route 53를 사용하여 DNS를 관리하는 방법에 대한 자세한 내용은 Amazon Route 53를 사용하여 인스턴스로 도메인을 지정을 참조하세요.

DNS 용어

도메인의 DNS를 관리하려면 몇 가지 용어를 숙지해야 합니다.

Apex 도메인/루트 도메인

루트 도메인이라고도 하는 apex 도메인은 하위 도메인 부분이 없는 도메인입니다. apex 도메인의 예는 `example.com`이고 하위 도메인의 예는 `www.example.com` 및 `blog.example.com`입니다. `www` 및 `blog` 하위 도메인 부분을 각각 포함하기 때문에 이들은 하위 도메인입니다.

도메인 이름 시스템(DNS)

DNS는 `example.com`과 같이 기억하기 쉬운 도메인 이름을 웹 서버의 IP 주소로 라우팅합니다.

자세한 내용은 Wikipedia에서 Domain Name System을 참조하세요.

DNS 레코드

DNS 레코드는 매핑 파라미터이며 도메인이나 하위 도메인이 연결된 IP 주소나 호스트 이름을 DNS 서버에 알립니다.

자세한 내용은 Wikipedia에서 List of DNS record types를 참조하세요.

DNS 영역

DNS 영역은 특정 도메인(예: `example.com`)과 하위 도메인(예: `blog.example.com`)의 인터넷 트래픽을 라우팅하는 방법에 대한 정보가 담긴 컨테이너입니다.

자세한 내용은 Wikipedia에서 DNS zone을 참조하세요.

도메인 이름 등록 기관

도메인 이름 공급자라고도 하는 도메인 이름 등록 기관은 도메인 이름 할당을 관리하는 회사나 조직입니다. 도메인을 구입하거나 Lightsail, Amazon Route 53 또는 다른 도메인 이름 등록 기관을 사용하여 기존의 도메인을 관리할 수 있습니다.

자세한 내용은 Wikipedia에서 Domain name registrar를 참조하세요.

이름 서버

이름 서버는 도메인으로 트래픽을 라우팅합니다. 에서 Lightsail이름 서버는 easy-to-remember 도메인 이름을 IP 주소로 변환하는 데 도움이 되는 네트워크 서비스를 실행하는 AWS 인스턴스

입니다. Lightsail는 트래픽을 도메인으로 라우팅하기 위한 여러 AWS 이름 서버 옵션(예: ns-NN.awsdns-NN.com)을 제공합니다. 도메인 등록 대행자를 사용하여 도메인을 변경할 때 이러한 AWS 이름 서버 중에서 선택할 수 있습니다.

자세한 내용은 Wikipedia에서 이름 서버를 참조하세요.

하위 도메인

하위 도메인은 더 큰 도메인의 일부분으로서 루트 도메인이 아닌 도메인 계층 구조에 있는 도메인입니다. 예를 들어 blog는 blog.example.com 하위 도메인의 하위 도메인 부분입니다.

자세한 내용은 Wikipedia에서 Subdomain을 참조하세요.

TTL(Time to Live)

TTL은 로컬에서 이름 서버를 확인하는 DNS 레코드의 수명을 나타냅니다. 예를 들어 시간이 짧으면 변경 사항이 적용될 때까지 기다리는 시간이 적어집니다. Lightsail DNS 영역에서는 TTL을 구성할 수 없습니다. 대신 모든 Lightsail DNS 레코드의 TTL이 기본적으로 60초로 설정됩니다.

자세한 내용은 Wikipedia에서 Time to live를 참조하세요.

와일드카드 DNS 레코드

와일드카드 DNS 레코드는 존재하지 않는 도메인 이름의 요청과 일치합니다. 별표 기호(*)를 도메인 이름의 제일 왼쪽에 사용하여(예: *.example.com 또는 *example.com) 와일드카드 DNS 레코드를 지정합니다.

> **Note**
>
> Lightsail DNS 영역은 NS(이름 서버) 레코드에 정의된 이름 서버 도메인(예: *awsdns.com)에 대해 와일드카드 레코드를 지원합니다.

Lightsail DNS 영역에서 지원되는 DNS 레코드 유형

주소(A) 레코드

A 레코드는 도메인(예: example.com)이나 하위 도메인(예: blog.example.com)을 웹 서버의 IP 주소로 매핑합니다.

예를 들어 Lightsail DNS 영역에서 example.com(도메인의 apex)의 웹 트래픽을 인스턴스로 보내려고 합니다. A 레코드를 만들고 @ 기호를 하위 도메인 텍스트 상자에 입력한 후 웹 서버의 IP 주소를 Resolves to address(주소로 확인) 텍스트 상자에 입력합니다.

A 레코드에 대한 자세한 내용은 Wikipedia의 List of DNS record types를 참조하세요.

AAAA 레코드

AAAA 레코드는 도메인(예: `example.com`)이나 하위 도메인(예: `blog.example.com`)을 웹 서버의 IPv6 주소로 매핑합니다.

예를 들어, Lightsail DNS 영역에서 IPv6 프로토콜을 통해 `example.com`(도메인의 정점)에 대한 웹 트래픽을 인스턴스로 연결하려고 합니다. AAAA 레코드를 만들고 @ 기호를 하위 도메인 텍스트 상자에 입력한 후 웹 서버의 IP 주소를 주소로 확인(Resolves to address) 텍스트 상자에 입력합니다.

AAAA 레코드에 대한 자세한 내용은 Wikipedia의 Domain Name System for IPv6를 참조하세요.

> **Note**
>
> Lightsail은 고정 IPv6 주소를 지원하지 않습니다. 리소스를 삭제하고 새 Lightsail 리소스를 생성하거나 동일한 리소스에서 IPv6를 비활성화했다가 다시 활성화하는 경우, 리소스의 최신 IPv6 주소를 반영하도록 AAAA 레코드를 업데이트해야 할 수 있습니다.

정식 이름(CNAME) 레코드

CNAME 레코드는 별칭이나 하위 도메인(예: `blog.example.com`)을 다른 도메인이나 하위 도메인에 매핑합니다.

예를 들어 Lightsail DNS 영역에서 `www.example.com`의 웹 트래픽을 `example.com`으로 보내려고 합니다. "확인" 주소가 `example.com`인 `www`의 별칭 CNAME 레코드를 만듭니다.

자세한 내용은 Wikipedia의 CNAME Record를 참조하세요.

메일 교환기(MX) 레코드

MX 레코드는 여러 서버가 정의된 경우 우선 순위 값을 사용하여 하위 도메인(예: `mail.example.com`)을 이메일 서버 주소로 매핑합니다.

예를 들어 Lightsail DNS 영역에서 `mail.example.com`의 메일을 `10 inbound-smtp.us-west-2.amazonaws.com` Amazon WorkMail 서버로 보내려고 합니다. `example.com`의 하위 도메인, 우선 순위 `10`, "확인" 주소 `inbound-smtp.us-west-2.amazonaws.com`으로 MX 레코드를 만듭니다.

자세한 내용은 Wikipedia의 MX Record를 참조하세요.

NS(이름 서버) 레코드

NS 레코드는 `test.example.com`과 같은 하위 도메인을 `ns-NN.awsdns-NN.com`과 같은 이름 서버로 위임합니다.

자세한 내용은 Wikipedia에서 Name server를 참조하세요.

서비스 로케이터(SRV) 레코드

SRV 레코드는 우선 순위, 가중치 및 포트 번호의 값을 사용하여 하위 도메인(예: `service.example.com`)을 서비스 주소로 매핑합니다. 텔레포니 또는 인스턴트 메시징은 일반적으로 SRV 레코드에 연결되는 몇 가지 서비스입니다.

예를 들어 Lightsail DNS 영역에서 `service.example.com`의 트래픽을 `1 10 5269 xmpp-server.example.com`으로 보내려고 합니다. 우선 순위 `1`, 가중치 `10`, 포트 번호 `5269`, "매핑" 주소 `xmpp-server.example.com`으로 SRV 레코드를 만듭니다.

자세한 내용은 Wikipedia의 SRV Record를 참조하세요.

텍스트(TXT) 레코드

TXT 레코드는 하위 도메인을 일반 텍스트로 매핑합니다. 도메인 소유권을 서비스 공급자에게 확인시키기 위해 TXT 레코드를 만듭니다.

예를 들어 Lightsail DNS 영역에서 `_amazonchime.example.com` 호스트 이름이 쿼리되면 `23223a30-7f1d-4sx7-84fb-31bdes7csdbb`로 응답하려고 합니다. 하위 도메인 값 `_amazonchime`, "응답" 값 `23223a30-7f1d-4sx7-84fb-31bdes7csdbb`.로 TXT 레코드를 만듭니다.

자세한 내용은 Wikipedia의 TXT Record를 참조하세요.

DNS 영역을 생성하여 Lightsail 인스턴스의 도메인 레코드 관리

도메인 이름(예: `example.com`)에 대한 트래픽을 Amazon Lightsail 인스턴스로 라우팅하려면 도메인의 도메인 이름 시스템(DNS)에 레코드를 추가합니다. 도메인을 등록한 등록 대행자를 사용하여 도메인의 DNS 레코드를 관리하거나 Lightsail을 사용하여 관리할 수 있습니다.

도메인의 DNS 레코드 관리를 Lightsail에 이전하는 것이 좋습니다. 이를 통해 도메인과 컴퓨팅 리소스를 Lightsail 한 곳에서 효율적으로 관리할 수 있습니다. Lightsail DNS 영역을 생성하여 Lightsail로 도메인의 DNS 레코드를 관리할 수 있습니다. 최대 6개의 Lightsail DNS 영역을 생성할 수 있습니다. 6개

이상의 DNS 영역이 필요한 경우 6개 이상의 도메인 이름을 관리하므로 Amazon Route 53를 사용하여 모든 도메인의 DNS를 관리하는 것이 좋습니다. Route 53를 사용하여 도메인의 트래픽을 Lightsail 리소스로 라우팅할 수 있습니다. Route 53를 사용하여 DNS를 관리하는 방법에 대한 자세한 내용은 Amazon Route 53를 사용하여 인스턴스로 도메인을 지정을 참조하세요.

이 가이드에서는 도메인에 대한 Lightsail DNS 영역을 생성하는 방법과 도메인의 DNS 레코드 관리를 Lightsail에 전송하는 방법을 안내합니다. 도메인의 DNS 레코드 관리를 Lightsail에 이전한 후 도메인의 등록 대행자에서 도메인에 대한 갱신 및 청구 작업을 계속 관리합니다.

> **Important**
>
> 도메인의 DNS를 변경하면 인터넷의 DNS를 통해 전파하는 데 몇 시간이 걸릴 수 있습니다. 따라서 관리를 Lightsail에 전파하는 동안 도메인의 현재 DNS 호스팅 공급자에 도메인의 DNS 레코드를 보관해야 합니다. 이렇게 하면 전송이 진행되는 동안 도메인의 트래픽이 중단되지 않고 리소스로 계속 라우팅됩니다.

1단계: 필수 구성 요소 완성

아직 수행하지 않은 경우 다음 사전 조건을 완료하세요.

1. 도메인 이름을 등록합니다. 그런 다음 도메인 이름 서버를 편집할 수 있는 관리 액세스 권한이 있음을 확인합니다.

   등록된 도메인 이름이 필요한 경우 Lightsail을 사용하여 도메인을 등록할 수 있습니다. 자세한 내용은 도메인 등록을 참조하세요.
2. 도메인에 필요한 DNS 레코드 유형이 Lightsail DNS 영역에서 지원되는지 확인합니다. 현재 Lightsail DNS 영역에서는 주소(A 및 AAAA), 정식 이름(CNAME), 메일 교환기(MX), 이름 서버(NS), 서비스 로케이터(SRV) 및 텍스트(TXT) 레코드 유형이 지원됩니다. NS 레코드에는 와일드카드 DNS 레코드 항목을 사용할 수 있습니다.

   도메인에 필요한 DNS 레코드 유형이 Lightsail DNS 영역에서 지원되지 않으면 더 많은 수의 레코드 유형을 지원하므로 Route 53를 도메인의 DNS 호스팅 공급자로 사용할 수 있습니다. 자세한 내용은 Amazon Route 53 개발자 안내서의 지원되는 DNS 레코드 유형 및 Amazon Route 53를 기존 도메인에 대한 DNS 서비스로 설정을 참조하세요.
3. 도메인이 가리킬 Lightsail 인스턴스를 생성합니다. 자세한 내용은 인스턴스 생성을 참조하세요.
4. 고정 IP를 만들어 Lightsail 인스턴스에 연결합니다. 자세한 내용은 고정 IP를 생성하여 인스턴스에 연결을 참조하세요.

2단계: Lightsail 콘솔에서 DNS 영역 생성

Lightsail에서 DNS 영역을 생성하려면 다음 단계를 완료합니다. DNS 영역을 생성할 때 DNS 영역이 적용되는 도메인 이름을 지정합니다.

1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 도메인 및 DNS를 선택합니다. 그런 다음 DNS 영역 생성을 선택합니다.
3. 다음 옵션 중 하나를 선택하세요.
   - Amazon Route 53에 등록된 도메인을 사용하여 Amazon Route 53에 등록된 도메인 지정
   - 다른 등록 대행자의 도메인을 사용하여 다른 등록 대행자를 통해 등록된 도메인을 지정합니다.
4. 등록된 도메인 이름(예: `example.com`)을 선택하거나 입력합니다.

   도메인 이름을 입력할 때 `www`를 포함하지 않아도 됩니다. 이 가이드 후반부의 3단계: DNS 영역에 레코드 추가 섹션에서 주소(A) 레코드를 사용하여 `www`를 추가할 수 있습니다.

   > **Note**
   >
   > Lightsail DNS 영역은 버지니아(`us-east-1`) AWS 리전에서 생성됩니다. 생성하려는 Lightsail DNS 영역(`example.com`)과 동일하게 해당 리전에 있는 리소스 이름을 지정한 경우 리소스 이름 충돌 오류("일부 이름이 이미 사용 중입니다.")가 표시됩니다.
   > 오류를 해결하려면 리소스의 스냅샷을 생성합니다. 스냅샷에서 새 리소스를 생성하고 고유의 새 이름을 지정합니다. 그런 다음 Lightsail DNS 영역을 생성하려는 도메인과 동일한 이름의 원래 리소스를 삭제합니다.

5. DNS 영역 생성을 선택합니다.

   도메인 리소스 할당을 관리할 수 있는 DNS 영역 Assignments(할당) 페이지로 리디렉션됩니다. 할당을 사용하여 도메인을 로드 밸런서 및 인스턴스와 같은 Lightsail 리소스로 연결합니다.

3단계: DNS 영역에 레코드 추가

다음 단계를 완료하여 도메인의 DNS 영역에 레코드를 추가합니다. DNS 레코드는 인터넷 트래픽이 도메인에 라우팅되는 방법을 지정합니다. 예를 들어 도메인(예: `example.com`)의 apex에 대한 트래픽을 인스턴스 하나에 라우팅하고 하위 도메인(예: `blog.example.com`)의 트래픽을 다른 인스턴스로 라우팅할 수 있습니다.

1. DNS 영역 할당 페이지에서 DNS records(DNS 레코드) 탭을 선택합니다.

DNS 영역은 Lightsail 콘솔의 Domains & DNS(도메인 및 DNS) 탭에 나열되어 있습니다.

> **Note**
>
> DNS 영역 할당 페이지에서 도메인이 가리키는 Lightsail 리소스를 추가, 제거 또는 변경할 수 있습니다. Lightsail 인스턴스, 배포, 컨테이너 서비스, 로드 밸런서, 고정 IP 주소 등에서 도메인을 가리킬 수 있습니다. DNS 레코드에서 도메인의 DNS 레코드를 추가, 편집 또는 삭제할 수 있습니다.

2. 다음 레코드 유형 중 하나를 선택합니다.

   주소(A) 레코드

   A 레코드는 도메인(예: `example.com`)이나 하위 도메인(예: `blog.example.com`)을 웹 서버 또는 인스턴스의 IPv4 주소로 매핑합니다(예: `192.0.2.255`).

   1. Record name(레코드 이름) 텍스트 상자에 레코드의 대상 하위 도메인을 입력하거나 @ 기호를 입력하여 도메인의 Apex를 정의합니다.
   2. Resolves to(확인) 텍스트 상자에 레코드의 대상 IP 주소를 입력하고 실행 중인 인스턴스나 구성된 로드 밸런서를 선택합니다. 실행 중인 인스턴스를 선택할 때 인스턴스의 퍼블릭 IP 주소가 자동으로 추가됩니다.
   3. 배포 또는 컨테이너 서비스와 같은 Lightsail 및 AWS 리소스로 트래픽을 라우팅하려면 리소스 별칭인을 선택합니다. AWS DNS 영역의 한 레코드에서 다른 레코드로 트래픽을 라우팅할 수도 있습니다.

      > **Note**
      >
      > 고정 IP를 Lightsail 인스턴스에 연결한 다음 레코드가 확인할 값으로 고정 IP를 선택하는 것이 좋습니다. 자세한 내용은 고정 IP 생성을 참조하세요.

   AAAA 레코드

   AAAA 레코드는 도메인(예: `example.com`)이나 하위 도메인(예: `blog.example.com`)을 웹 서버 또는 인스턴스의 IPv6 주소로 매핑합니다(예: `2001:0db8:85a3:0000:0000:8a2e:0370:7334`).

> **Note**
>
> Lightsail은 고정 IPv6 주소를 지원하지 않습니다. 리소스를 삭제하고 새 Lightsail 리소스를 생성하거나 동일한 리소스에서 IPv6를 비활성화했다가 다시 활성화하는 경우, 리소스의 최신 IPv6 주소를 반영하도록 AAAA 레코드를 업데이트해야 할 수 있습니다.

1. Record name(레코드 이름) 텍스트 상자에 레코드의 대상 하위 도메인을 입력하거나 @ 기호를 입력하여 도메인의 Apex를 정의합니다.
2. 확인(Resolves to) 텍스트 상자에 레코드의 대상 IPv6 주소를 입력하고 실행 중인 인스턴스나 구성된 로드 밸런서를 선택합니다. 실행 중인 인스턴스를 선택할 때 인스턴스의 퍼블릭 IPv6 주소가 자동으로 추가됩니다.
3. 배포 또는 컨테이너 서비스와 같은 Lightsail 및 AWS 리소스로 트래픽을 라우팅하려면 리소스 별칭인을 선택합니다. AWS DNS 영역의 한 레코드에서 다른 레코드로 트래픽을 라우팅할 수도 있습니다.

정식 이름(CNAME) 레코드

CNAME 레코드는 별칭 또는 하위 도메인(예: `www.example.com`)을 다른 도메인(예: `example.com`)이나 다른 하위 도메인(예: `blog.example.com`)에 매핑합니다.

1. Record name(레코드 이름) 텍스트 상자에 레코드의 하위 도메인을 입력합니다.
2. Route traffic to(트래픽 라우팅 대상) 텍스트 상자에 레코드의 대상 도메인이나 하위 도메인을 입력합니다.

메일 교환기(MX) 레코드

MX 레코드는 여러 서버가 정의된 경우 우선 순위 값을 사용하여 하위 도메인(예: `mail.example.com`)을 이메일 서버 주소로 매핑합니다.

1. Record name(레코드 이름) 텍스트 상자에 레코드의 하위 도메인을 입력합니다.
2. 우선순위 텍스트 상자에 레코드의 우선순위를 입력합니다. 여러 서버의 레코드를 추가할 때 이 값이 중요합니다.
3. Route traffic to(트래픽 라우팅 대상) 텍스트 상자에 레코드의 대상 도메인이나 하위 도메인을 입력합니다.

서비스 로케이터(SRV) 레코드

SRV 레코드는 우선 순위, 가중치 및 포트 번호의 값을 사용하여 하위 도메인(예: `service.example.com`)을 서비스 주소로 매핑합니다. 텔레포니 또는 인스턴트 메시징은 일반적으로 SRV 레코드에 연결되는 몇 가지 서비스입니다.

1. Record name(레코드 이름) 텍스트 상자에 레코드의 하위 도메인을 입력합니다.
2. 우선순위 텍스트 상자에 레코드의 우선순위를 입력합니다.
3. Weight(가중치) 텍스트 상자에 같은 우선 순위를 가진 SRV 레코드의 상대 가중치를 입력합니다.
4. Route traffic to(트래픽 라우팅 대상) 텍스트 상자에 레코드의 대상 도메인이나 하위 도메인을 입력합니다.
5. Port(포트) 텍스트 상자에 서비스와 연결할 수 있는 포트 번호를 입력합니다.

텍스트(TXT) 레코드

TXT 레코드는 하위 도메인을 일반 텍스트로 매핑합니다. 도메인 소유권을 서비스 공급자에게 확인시키기 위해 TXT 레코드를 만듭니다.

1. Record name(레코드 이름) 텍스트 상자에 레코드의 하위 도메인을 입력합니다.
2. 하위 도메인이 쿼리될 때 제공된 텍스트 응답을 Responds with(응답) 텍스트 상자에 입력합니다.

> **Note**
>
> 입력 텍스트는 따옴표로 묶을 필요가 없습니다.

3. 레코드 추가를 마치면 저장 아이콘을 선택하여 변경 사항을 저장합니다.

레코드가 DNS 영역에 추가됩니다. 도메인의 DNS 영역에 여러 레코드를 추가하려면 위의 단계를 반복합니다.

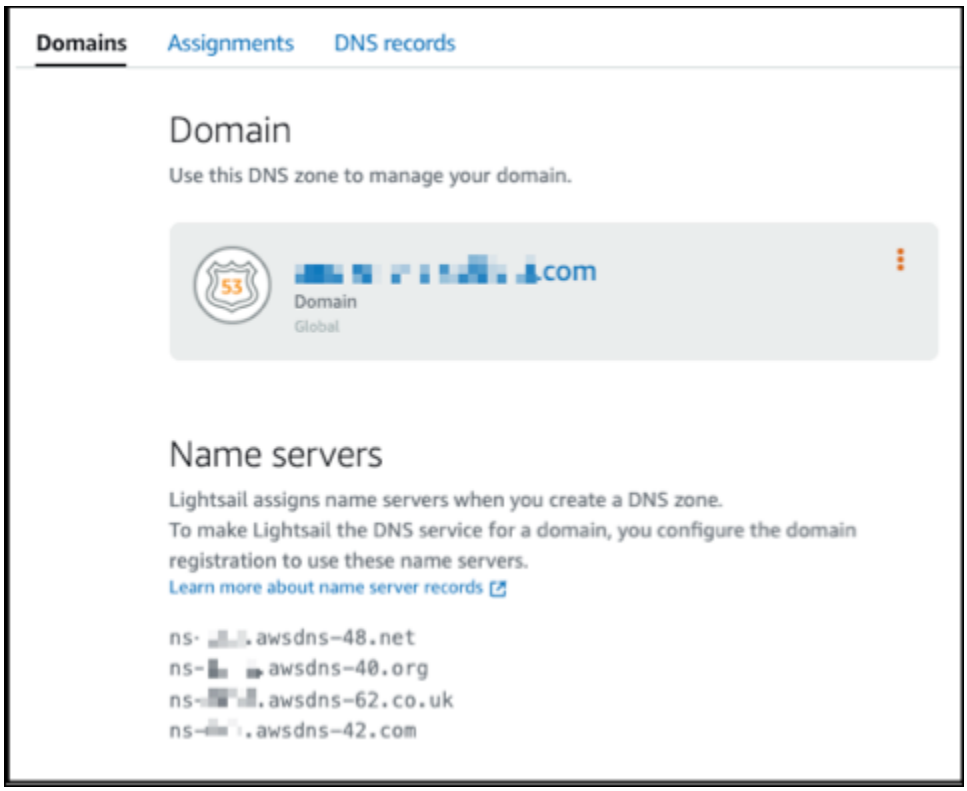
> **Note**
>
> DNS 레코드의 TTL(Time to live)은 Lightsail DNS 영역에서 구성할 수 없습니다. 대신 모든 Lightsail DNS 레코드의 TTL이 기본적으로 60초로 설정됩니다. 자세한 내용은 Wikipedia의 Time to live를 참조하십시오.

4단계: 도메인의 현재 DNS 호스팅 공급자에서 이름 서버 변경

다음 단계를 완료하여 도메인의 DNS 레코드 관리를 Lightsail에 이전합니다. 이렇게 하려면 도메인의 현재 DNS 호스팅 공급자 웹사이트에 로그인하고 도메인 이름 서버를 Lightsail 이름 서버로 변경합니다.

> **Important**
>
> 웹 트래픽이 현재 도메인으로 라우팅되는 경우 도메인의 현재 DNS 호스팅 공급자에서 이름 서버가 변경되기 전에 기존의 모든 DNS 레코드가 Lightsail DNS 영역에 있어야 합니다. 이런 식으로 Lightsail DNS 영역으로 이전된 후 트래픽이 중단되지 않고 계속 흐릅니다.

1. 도메인의 DNS 영역 관리 페이지에 있는 Lightsail 이름 서버를 적어 둡니다. 이름 서버는 Lightsail DNS 영역의 도메인(Domains) 탭에 있습니다.



2. 도메인의 현재 DNS 호스팅 공급자 웹 사이트에 로그인합니다.
3. 도메인의 이름 서버를 편집할 수 있는 페이지를 찾습니다.

   이 페이지를 찾는 데 대한 자세한 내용은 도메인의 현재 DNS 호스팅 공급자가 제공한 설명서를 참조하십시오.

4. Lightsail 이름 서버를 입력하고 나열된 다른 이름 서버를 제거합니다.
5. 변경 내용을 저장합니다.

   인터넷의 DNS를 통해 이름 서버 변경 사항이 전파될 때까지 기다립니다. 몇 시간 정도 걸릴 수 있습니다. 완료되면 도메인의 인터넷 트래픽이 Lightsail DNS 영역을 통해 라우팅되기 시작해야 합니다.

다음 단계

- DNS 영역 편집
- 로드 밸런서 생성 및 인스턴스 연결

Lightsail DNS 영역 편집

도메인의 DNS 영역에서 DNS 레코드를 편집합니다. 도메인의 DNS 레코드 관리를 다른 DNS 호스팅 공급자에게 이전하거나 도메인을 등록한 등록 대행자에게 되돌리려는 경우 Amazon Lightsail에서 도메인의 DNS 영역을 삭제할 수도 있습니다. 자세한 내용은 ??? 섹션을 참조하세요.

> **Note**
>
> Lightsail 콘솔에서 DNS 편집기를 사용하여 레코드를 편집할 수 있으므로 도메인의 DNS 레코드 관리를 Lightsail로 이전해야 합니다. 자세한 내용은 DNS 영역을 생성하여 도메인의 DNS 레코드 관리를 참조하세요.

DNS 레코드 편집

Lightsail 콘솔을 사용하여 언제든지 도메인의 DNS 영역에서 DNS 레코드를 편집할 수 있습니다.

DNS 영역을 편집하려면

1. Lightsail 콘솔에 로그인합니다.
2. Lightsail 콘솔 홈 페이지의 왼쪽 탐색 창에서 도메인 및 DNS를 선택합니다.
3. 편집할 DNS 영역의 이름을 선택합니다.
4. DNS 영역 DNS 레코드 페이지에서 삭제하려는 레코드 옆에 있는 삭제 아이콘을 선택합니다.
5. 완료했으면 저장 아이콘을 선택하여 변경 사항을 저장합니다.

> **Note**
>
> 인터넷의 DNS를 통해 DNS 레코드 변경 사항이 전파될 때까지 기다립니다. 몇 시간 정도 걸릴 수 있습니다.

Lightsail에서 DNS 영역 삭제

경우에 따라 도메인의 DNS 레코드를 관리하기 위해 Amazon Lightsail에 설정한 DNS 영역을 완전히 제거하려고 할 수 있습니다. DNS 관리를 다른 공급자로 전송하거나 도메인 등록 기관으로 다시 전송하려고 할 수 있습니다. DNS 영역을 삭제하는 것은 간단하지만 도메인의 트래픽이 계속 올바르게 라우팅되도록 미리 계획하는 것이 중요합니다. Lightsail에서 DNS 영역을 삭제하는 단계를 살펴보겠습니다.

> **Important**
>
> 도메인을 통해 계속 트래픽을 라우팅하도록 계획할 경우 Lightsail에서 도메인의 DNS 영역을 삭제하기 전에 다른 DNS 호스팅 공급자를 준비합니다. 그렇지 않으면, Lightsail DNS 영역을 삭제할 때 웹 사이트에 대한 모든 트래픽이 중지됩니다.

DNS 영역을 삭제하려면

1. Lightsail 콘솔 홈 페이지의 왼쪽 탐색 창에서 도메인 및 DNS를 선택합니다.
2. 삭제할 DNS 영역의 이름을 선택합니다.
3. 세로 줄임표 메뉴(:)를 선택합니다. 그런 다음 Delete(삭제) 옵션을 선택합니다.
4. Delete DNS zone(DNS 영역 삭제)을 선택하여 삭제를 확인합니다.

   DNS 영역이 Lightsail에서 삭제됩니다.

Lightsail에서 웹 사이트로 인터넷 트래픽을 라우팅하는 방식을 알아봅니다.

스마트폰, 노트북, 웹 사이트 서버를 비롯한 인터넷상의 모든 컴퓨터는 고유한 문자열을 사용하여 서로 통신합니다. IP 주소라고 하는 이 문자열은 다음 형식 중 하나로 되어 있습니다.

- 192.0.2.44와 같은 인터넷 프로토콜 버전 4(IPv4)
- 2001:DB8::/32와 같은 인터넷 프로토콜 버전 6(IPv6)

브라우저를 열고 웹 사이트로 이동할 때는 이런 긴 문자열을 기억해 입력할 필요가 없습니다. 그 대신 example.com과 같은 도메인 이름을 입력해도 원하는 웹 사이트로 갈 수 있습니다. 등록된 도메인 이름을 IP 주소로 매핑하는 디렉터리 역할을 하는 도메인 이름 시스템(DNS)을 이용하면 가능합니다.

목차

- 인터넷 트래픽을 도메인으로 라우팅하도록 Lightsail을 구성하는 방법 개요
- 도메인의 트래픽을 라우팅하는 방법
- 다음 단계

인터넷 트래픽을 도메인으로 라우팅하도록 Lightsail을 구성하는 방법 개요

다음 개요에서는 Lightsail을 사용하여 인터넷 트래픽을 웹 사이트 또는 웹 애플리케이션으로 라우팅하는 도메인을 등록 및 구성하는 방법을 설명합니다.

1. 도메인 이름을 등록합니다. 이에 대한 개요는 도메인 등록을 참조하세요.
2. 도메인 이름을 등록한 후 Lightsail은 도메인과 동일한 이름의 DNS 영역을 자동으로 생성합니다.
3. Lightsail 콘솔을 사용하면 인스턴스나 로드 밸런서와 같은 Lightsail 리소스에 도메인을 쉽게 할당할 수 있습니다. DNS 영역에 DNS 레코드를 생성하여 트래픽을 리소스로 라우팅할 수도 있습니다. 각각의 레코드에는 도메인의 트래픽을 라우팅할 방법에 관한 다음과 같은 정보가 포함되어 있습니다.

   이름

   레코드의 이름은 도메인 이름(예: example.com) 또는 하위 도메인 이름(예: www.example.com, retail.example.com)과 일치합니다. DNS 영역에 있는 모든 레코드의 이름은 반드시 DNS 영역의 이름으로 끝나야 합니다. 예를 들어 DNS 영역의 이름이 example.com이라면 모든 레코드 이름이 example.com으로 끝나야 합니다.

   유형

   레코드 유형은 일반적으로 트래픽을 라우팅할 리소스 유형에 따라 다릅니다. 예를 들어 트래픽을 이메일 서버로 라우팅하려면 Type(유형)을 MX로 지정합니다. 사용자의 도메인 이름에 대한 트래픽을 Lightsail 인스턴스로 라우팅하려면, 사용자의 도메인 이름을 인스턴스의 고정 IPv4 주소로 연결하는 A 레코드 또는 인스턴스의 IPv6 주소로 연결하는 AAAA 레코드를 추가합니다.
4. 대상

대상은 트래픽을 라우팅하려는 위치입니다. 트래픽을 Lightsail 인스턴스,Lightsail 컨테이너 서비스 및 기타 Lightsail 리소스로 라우팅하는 별칭 레코드를 생성할 수 있습니다. 자세한 내용은 DNS를 참조하십시오.

도메인의 트래픽을 라우팅하는 방법

인터넷 트래픽을 인스턴스, 로드 밸런서, 배포 또는 컨테이너 서비스와 같은 리소스로 라우팅하도록 Lightsail을 구성한 후 누군가 www.example.com에 대한 콘텐츠를 요청하면 이렇게 됩니다.

1. 사용자가 웹 브라우저를 열어 주소 표시줄에 www.example.com을 입력하고 Enter 키를 누릅니다.
2. www.example.com에 대한 요청은 일반적으로 인터넷 서비스 제공업체(ISP)가 관리하는 DNS 해석기로 라우팅됩니다. ISP는 케이블 인터넷 공급업체, DSL 광대역 공급업체 또는 기업 네트워크가 될 수 있습니다.
3. ISP의 DNS 해석기는 www.example.com에 대한 요청을 DNS 루트 이름 서버에 전달합니다.
4. DNS 해석기는 www.example.com에 대한 요청을 이번에는 .com 도메인의 TLD 이름 서버 중 하나에 다시 전달합니다. .com 도메인의 이름 서버는 example.com 도메인과 연관된 4개의 이름 서버의 이름을 사용하여 요청에 응답합니다.

   DNS 해석기는 4개의 이름 서버를 캐시에 저장합니다. 다음에 누군가 example.com을 찾아볼 때 example.com의 이름 서버가 이미 있으므로 해석기는 3단계와 4단계를 건너뜁니다. 이름 서버는 일반적으로 2일 동안 캐시에 저장됩니다.
5. DNS 해석기는 이름 서버 하나를 선택하여 www.example.com에 대한 요청을 해당 이름 서버에 전달합니다.
6. 이름 서버는 example.com DNS 영역에서 www.example.com 레코드를 찾아 웹 서버의 IP 주소 192.0.2.44 등 연관된 값을 가져옵니다. 이후 이름 서버는 IP 주소를 DNS 해석기에 반환합니다.
7. DNS 해석기가 마침내 사용자에게 필요한 IP 주소를 해석해 냅니다. 해석기는 이 값을 웹 브라우저로 반환합니다.
8. 웹 브라우저는 DNS 해석기로부터 얻은 IP 주소로 www.example.com에 대한 요청을 전송합니다. 여기가 콘텐츠가 있는 곳, 예컨대 웹 사이트 엔드포인트로 구성된 Lightsail 인스턴스 또는 컨테이너너 서비스에서 실행되는 웹 서버입니다.
9. 192.0.2.44에 있는 웹 서버 또는 그 밖의 리소스는 www.example.com의 웹 페이지를 웹 브라우저에게 반환하고, 웹 브라우저는 이 페이지를 표시합니다.

다음 단계

- DNS
- 인스턴스로 도메인 연결
- 로드 밸런서로 도메인 연결
- 배포로 도메인 연결

도메인 트래픽을 Lightsail 인스턴스로 라우팅

Amazon Lightsail의 DNS 영역을 사용하여 example.com과 같은 등록된 도메인 이름을 가상 프라이빗 서버(VPS)라고도 하는 Lightsail 인스턴스에서 실행되는 웹 사이트로 연결할 수 있습니다. Lightsail 계정에 최대 6개의 DNS 영역을 생성할 수 있습니다. 모든 DNS 레코드 유형이 지원되는 것은 아닙니다. Lightsail DNS 영역에 대한 자세한 내용은 DNS를 참조하세요.

DNS 영역을 6개 이상 생성하거나 Lightsail에서 지원되지 않는 DNS 레코드 유형을 사용하려는 경우 Amazon Route 53 호스팅 영역을 사용하는 것이 좋습니다. Route 53를 사용하면 최대 500개 도메인의 DNS를 관리할 수 있습니다. 또한 매우 다양한 DNS 레코드 유형을 지원합니다. 자세한 내용은 Amazon Route 53 개발자 안내서의 호스팅 영역 작업 섹션을 참조하세요.

이 안내서는 Lightsail에서 관리되는 도메인의 DNS 레코드를 편집하여 Lightsail 인스턴스에 연결하는 방법을 보여줍니다. 인터넷의 DNS를 통해 DNS 영역 변경 사항이 전파될 때까지 최대 48시간 동안 기다립니다.

사전 조건

아직 수행하지 않은 경우 다음 사전 조건을 완료하세요.

- Lightsail을 사용하여 도메인 이름 등록 자세한 내용은 새 도메인 등록을 참조하세요.
- 이미 도메인을 등록했지만 Lightsail을 사용하여 레코드를 관리하지 않는 경우 도메인의 DNS 레코드 관리를 Lightsail에 전송해야 합니다. 자세한 내용은 DNS 영역을 생성하여 도메인의 DNS 레코드 관리를 참조하세요.
- Lightsail 인스턴스에 연결된 기본 동적 퍼블릭 IP 주소는 인스턴스를 중지하고 다시 시작할 때마다 변경됩니다. 퍼블릭 IP 주소가 변경되지 않도록 고정 IP를 생성하고 인스턴스에 연결합니다. 이 안내서에서는 인스턴스를 중지 및 재시작할 때마다 도메인의 DNS 레코드를 업데이트할 필요가 없도록 도메인의 DNS 영역에서 고정 IP 주소를 확인하는 DNS 레코드를 생성합니다. 자세한 내용은 고정 IP를 생성하여 인스턴스에 연결을 참조하세요.

선택 사항 - Lightsail 인스턴스에 IPv6를 활성화된 상태로 둘 수 있습니다. 인스턴스를 중지했다가 시작해도 IPv6 주소는 유지됩니다. 자세한 내용은 IPv6 사용 및 사용 중지를 참조하세요.

Lightsail 인스턴스에 도메인 할당

다음 방법 중 하나를 사용하여 Lightsail의 인스턴스에 도메인 할당:

- 인스턴스 도메인 탭
- 고정 IP 도메인 탭
- DNS 영역 할당 탭

인스턴스 도메인 탭

다음 절차를 완료하여 Lightsail 콘솔의 인스턴스 도메인 및 DNS 섹션에 있는 Lightsail 인스턴스에 도메인을 할당합니다.

인스턴스 Domains(도메인) 탭을 사용하여 도메인 할당

1. Lightsail 콘솔에 로그인합니다.
2. 도메인을 할당할 인스턴스 이름을 선택합니다.
3. Domains(도메인) 탭에서 Assign domain(도메인 할당)을 선택합니다.
4. Lightsail 인스턴스에 할당할 도메인을 선택합니다.
5. 라우팅 정보가 올바른지 확인한 다음 Assign(할당)을 선택합니다.

선택 사항

인스턴스에서 도메인 할당을 편집하거나 제거하려면 도메인 이름 옆에 있는 편집 아이콘 또는 휴지통 아이콘을 선택합니다.

고정 IP 도메인 탭

다음 절차를 완료하여 Lightsail 콘솔의 고정 IP 도메인 및 DNS 탭에 있는 Lightsail 인스턴스에 도메인을 할당합니다.

고정 IP Domains(도메인) 탭을 사용하여 도메인 할당

1. Lightsail 콘솔에 로그인합니다.

2. 네트워킹 탭을 선택합니다.
3. 도메인을 할당할 고정 IP를 선택합니다.
4. Domains(도메인) 탭에서 Assign domain(도메인 할당)을 선택합니다.
5. 고정 IP에 할당할 도메인을 선택합니다.
6. 라우팅 정보가 올바른지 확인한 다음 Assign(할당)을 선택합니다.

선택 사항

고정 IP에서 도메인 할당을 편집하거나 제거하려면 도메인 이름 옆에 있는 편집 아이콘 또는 휴지통 아이콘을 선택합니다.

DNS 영역 할당 탭

다음 절차를 완료하여 도메인을 DNS 영역의 Assignments(할당) 탭에 있는 Lightsail 인스턴스에 할당합니다.

Assignments(할당) 탭을 사용하여 도메인 할당

1. Lightsail 콘솔에 로그인합니다.
2. Domains & DNS(도메인 및 DNS) 탭을 선택합니다.
3. 사용할 도메인 이름의 DNS 영역을 선택합니다.
4. Assignments(할당) 탭에서 Add assignment(할당 추가)를 선택합니다.
5. Lightsail 인스턴스에 할당할 도메인 이름을 선택합니다. 고정 IP가 아직 인스턴스에 연결되어 있지 않은 경우 연결하라는 메시지가 표시됩니다.
6. 라우팅 정보가 올바른지 확인한 다음 Assign(할당)을 선택합니다.

선택 사항

리소스에서 도메인 할당을 편집하거나 제거하려면 도메인 이름 옆에 있는 편집 아이콘 또는 휴지통 아이콘을 선택합니다.

Lightsail 로드 밸런서로 도메인 연결

암호화된(HTTPS) 트래픽을 처리할 도메인이 본인이 관리하는 도메인임을 확인한 후 Lightsail 로드 밸런서에 도메인을 지정하는 도메인의 DNS 호스팅 제공업체에 주소(A) 레코드를 추가해야 합니다. 이

설명서에서는 Lightsail DNS 영역과 Amazon Route 53 호스팅 영역에 A 레코드를 추가하는 방법을 안내합니다.

DNS 영역 - 과제 페이지를 사용하여 A 레코드 추가

1. 왼쪽 탐색 창에서 도메인 및 DNS를 선택합니다.
2. 관리하려는 DNS 영역을 선택합니다.
3. 과제 탭을 선택합니다.
4. Add assignment(할당 추가)를 선택합니다.
5. 도메인 이름 선택 필드에서 도메인 이름을 사용할지 아니면 도메인의 하위 도메인을 사용할지 선택합니다.
6. 리소스 선택 드롭다운에서 도메인을 할당하려는 로드 밸런서를 선택합니다.
7. 할당을 선택합니다.

인터넷의 DNS를 통해 변경 사항이 전파될 때까지 기다립니다. 몇 분에서 몇 시간 정도 걸릴 수 있습니다.

DNS 영역 - DNS 레코드 페이지를 사용하여 A 레코드 추가

1. 왼쪽 탐색 창에서 도메인 및 DNS를 선택합니다.
2. 관리하려는 DNS 영역을 선택합니다.
3. DNS records(DNS 레코드) 탭을 선택합니다.
4. DNS 영역의 현재 상태에 따라 다음 단계 중 하나를 완료합니다.

   - A 레코드를 추가하지 않은 경우 레코드 추가(Add record)를 선택합니다.
   - 이전에 A, AAAA 또는 CNAME 레코드를 추가한 경우 페이지에 나열된 기존 A, AAAA 또는 CNAME 레코드 옆에 있는 편집 아이콘을 선택한 다음, 이 절차의 5단계로 건너뜁니다.
5. 레코드 유형 드롭다운 메뉴에서 A 레코드(A record)를 선택합니다.
6. Record name(레코드 이름) 텍스트 상자에 다음 옵션 중 하나를 입력합니다.

   - @을 입력하여 도메인의 정점(예: `example.com`)에 대한 트래픽 경로를 로드 밸런서로 지정합니다.
   - www를 입력하여 www 하위 도메인(예: `www.example.com`) 트래픽 경로를 로드 밸런서로 지정합니다.
7. 확인(Resolves to) 텍스트 상자에서 Lightsail 로드 밸런서 이름을 선택합니다.

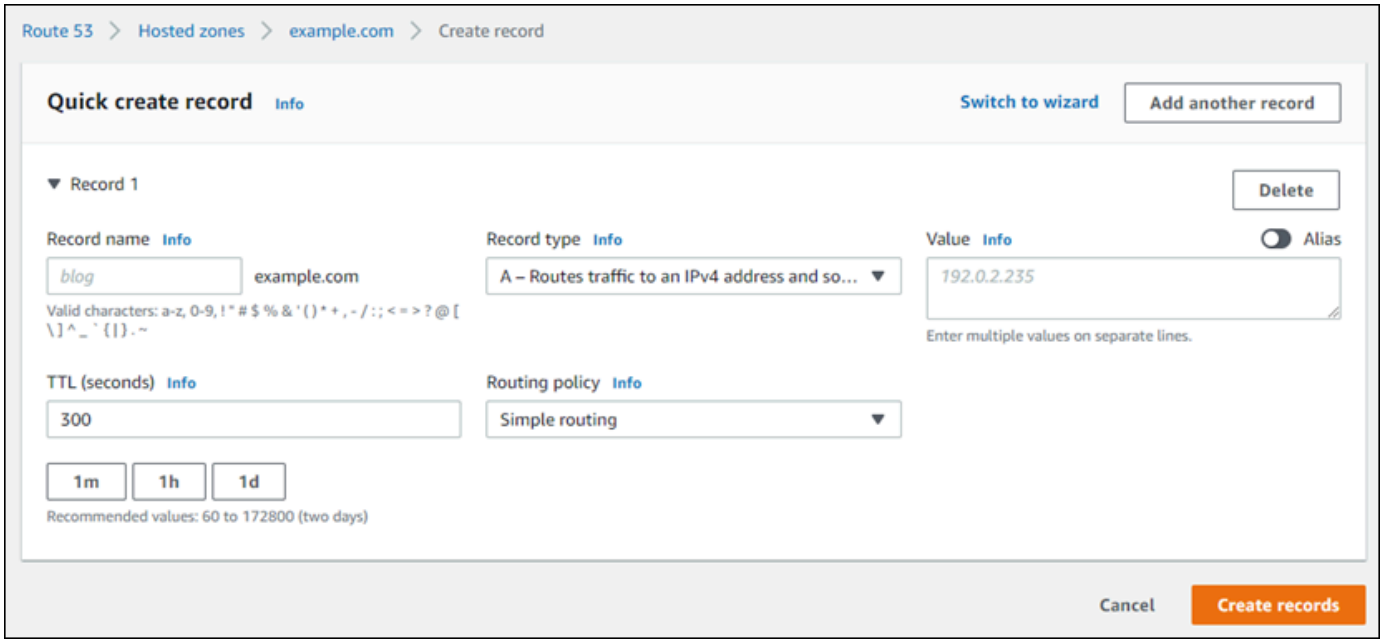
8. 저장 아이콘을 선택합니다.

인터넷의 DNS를 통해 변경 사항이 전파될 때까지 기다립니다. 몇 분에서 몇 시간 정도 걸릴 수 있습니다.

Route 53에 A 레코드 추가

1. Route 53 콘솔에 로그인합니다.
2. 탐색 창에서 호스팅 영역(Hosted zones)을 선택합니다.
3. 트래픽 경로를 로드 밸런서로 지정하는 데 사용할 도메인 이름이 있는 호스팅 영역을 선택합니다.
4. 레코드 세트 생성을 선택합니다.

   레코드 빠른 생성(Quick create record) 페이지가 나타납니다.



> **Note**
>
> 라우팅 정책 선택(Choose routing policy) 페이지가 표시되면 빠른 생성으로 전환(Switch to quick create)을 선택하여 빠른 생성 마법사로 전환한 후 다음 단계를 이어서 진행하세요.

5. 레코드 이름(Record name)에 www 하위 도메인(예: `www.example.com`)을 사용할 경우 www를 입력하거나 도메인의 정점(예: `example.com`)을 사용할 경우 비워 둡니다.
6. 레코드 유형에서 A - IPv4 주소 및 일부 AWS 리소스로 트래픽 라우팅을 선택합니다.

7. 별칭(Alias) 토글을 선택하여 별칭 리소스를 활성화합니다.
8. 트래픽 라우팅 대상(Route traffic to)에 대해 다음 옵션을 선택합니다.

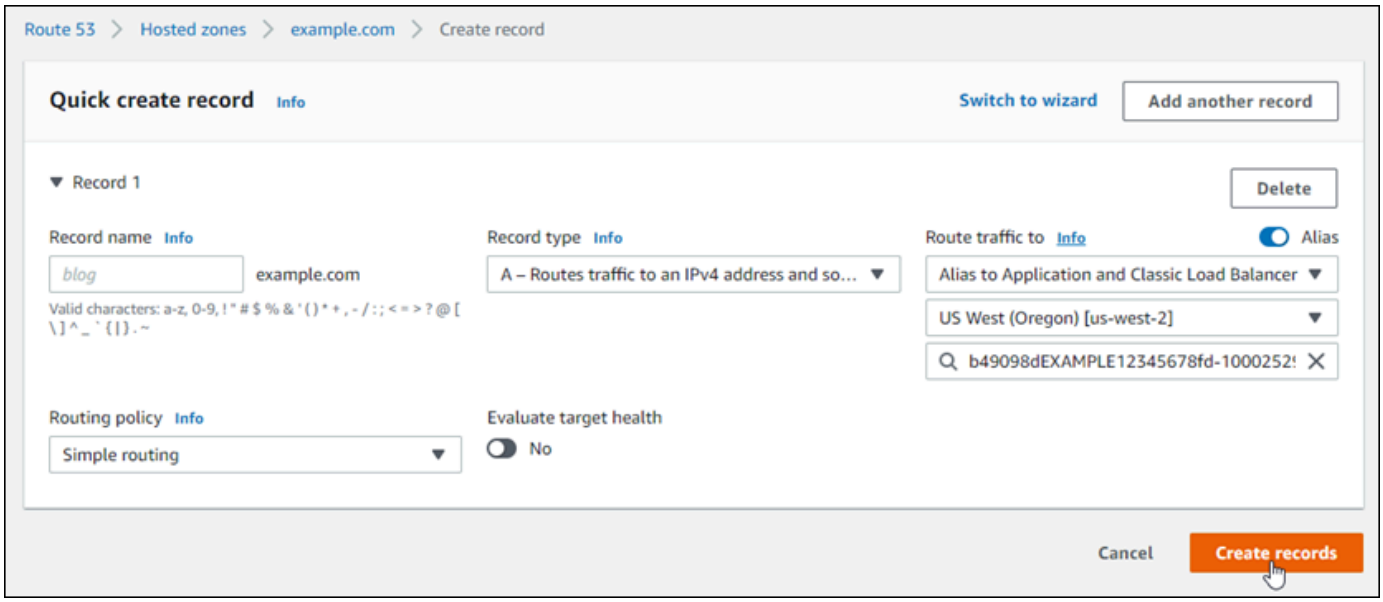
   a. 엔드포인트 선택(Choose endpoint)에서 애플리케이션 및 Classic Load Balancer 대한 별칭(Alias to Application and Classic Load Balancer)을 선택합니다.

   b. 리전 선택(Choose Region)에서 Lightsail 로드 밸런서를 생성한 AWS 리전을 선택합니다.

   c. 로드 밸런서 선택(Choose load balancer)에 Lightsail 로드 밸런서의 엔드포인트 URL(예: DNS 이름)을 입력하거나 붙여넣습니다.
9. 라우팅 정책(Routing Policy)으로 단순 라우팅(Simple routing)을 선택하고 대상 상태 평가(Evaluate target health) 토글을 비활성으로 설정합니다.

   Lightsail이 이미 로드 밸런서의 상태 확인을 수행했습니다. 자세한 내용은 로드 밸런서에 대한 상태 확인을 참조하세요.

   레코드는 다음 예시와 같은 형식이어야 합니다.



10. 레코드 생성(Create records)을 선택하여 호스팅 영역에 레코드를 추가합니다.

> **Note**
>
> 인터넷의 DNS를 통해 변경 사항이 전파될 때까지 기다립니다. 몇 분에서 몇 시간 정도 걸릴 수 있습니다.

Lightsail 도메인에 대한 전송 DNS 관리

Amazon Lightsail DNS 영역을 사용하여 Lightsail 사용을 통해 등록한 도메인의 DNS 레코드를 관리할 수 있습니다. 또는 원하는 경우 도메인의 DNS 레코드 관리를 다른 DNS 호스팅 공급자에게 이전할 수 있습니다. 이 안내서에서는 Lightsail에 등록한 도메인의 DNS 레코드 관리를 다른 DNS 호스팅 공급자에게 이전하는 방법을 보여줍니다.

> **Important**
>
> 도메인의 DNS를 변경하면 인터넷의 DNS를 통해 전파하는 데 몇 시간이 걸릴 수 있습니다. 따라서 관리 이전이 완료될 때까지 현재 DNS 호스팅 공급자에 도메인의 DNS 레코드를 보관해야 합니다. 이렇게 하면 전송이 진행되는 동안 도메인의 트래픽이 중단되지 않고 리소스로 계속 라우팅됩니다.

목차

- 사전 조건 충족
- DNS 영역에 레코드 추가

사전 조건 완료

아직 수행하지 않은 경우 다음 사전 조건을 완료하세요.

1. 도메인 이름을 등록합니다. Lightsail을 사용하여 도메인 이름을 등록할 수 있습니다. 자세한 내용은 새 도메인 등록을 참조하세요.
2. DNS 서비스가 제공하는 프로세스를 이용해 도메인에 대한 이름 서버를 얻습니다.

DNS 영역에 레코드 추가

다음 절차를 완료하여 다른 DNS 호스팅 공급자의 이름 서버를 Lightsail의 등록된 도메인에 추가합니다.

1. Lightsail 콘솔에 로그인합니다.
2. Domains & DNS(도메인 및 DNS) 탭을 선택합니다.
3. 다른 DNS 서비스를 사용하도록 구성하고자 하는 도메인의 이름을 선택합니다.

4. Edit Name Servers(이름 서버 편집)를 선택합니다.
5. 사전 조건을 완료하면 이름 서버의 이름을 DNS 서비스로부터 얻은 이름 서버로 변경합니다.
6. 저장을 선택합니다.

Amazon Route 53를 사용하여 도메인을 Lightsail 인스턴스에 지정

Amazon Lightsail의 DNS 영역은 등록된 도메인 이름(예: `example.com`)을 Lightsail 인스턴스에서 실행 중인 웹 사이트에 쉽게 지정할 수 있습니다. 최대 6개의 Lightsail DNS 영역을 생성할 수 있으며 모든 DNS 레코드 유형이 지원되는 것은 아닙니다. Lightsail DNS 영역에 대한 자세한 내용은 DNS를 참조하세요.

Lightsail DNS 영역이 너무 제한적인 경우 Amazon Route 53 호스팅 영역을 사용하여 도메인의 DNS 레코드를 관리하는 것이 좋습니다. Route 53를 사용하여 최대 500개의 도메인에 대한 DNS를 관리할 수 있으며 이는 다양한 DNS 레코드 유형을 지원합니다. 또는 이미 Route 53를 사용하여 도메인의 DNS 레코드를 관리하고 있으며 계속 사용하는 것을 선호할 수도 있습니다. 이 설명서에서는 Route 53에서 관리되는 도메인의 DNS 레코드를 편집하여 Lightsail 인스턴스에 지정하는 방법을 보여줍니다.

사전 조건

아직 수행하지 않은 경우 다음 사전 조건을 완료하세요.

- Route 53를 사용하여 도메인 이름 등록 자세한 내용은 Route 53 설명서의 새 도메인 등록을 참조하세요.
- 이미 도메인을 등록했지만 Route 53를 사용하여 레코드를 관리하지 않는 경우 도메인의 DNS 레코드 관리를 Route 53에 전송해야 합니다. 자세한 내용은 Route 53 설명서의 Amazon Route 53를 기존 도메인에 대한 DNS 서비스로 설정을 참조하세요.
- Route 53에서 도메인에 대한 퍼블릭 호스팅 영역을 생성합니다. 자세한 내용은 Route 53 설명서의 퍼블릭 호스팅 영역 생성을 참조하세요.
- 고정 IP를 만들어 Lightsail 인스턴스에 연결합니다. 이 설명서에서는 인스턴스의 고정 IP 주소(퍼블릭 IP 주소)로 확인되는 도메인의 Route 53 호스팅 영역에 DNS 레코드를 생성합니다. 자세한 내용은 고정 IP를 생성하여 인스턴스에 연결을 참조하세요.

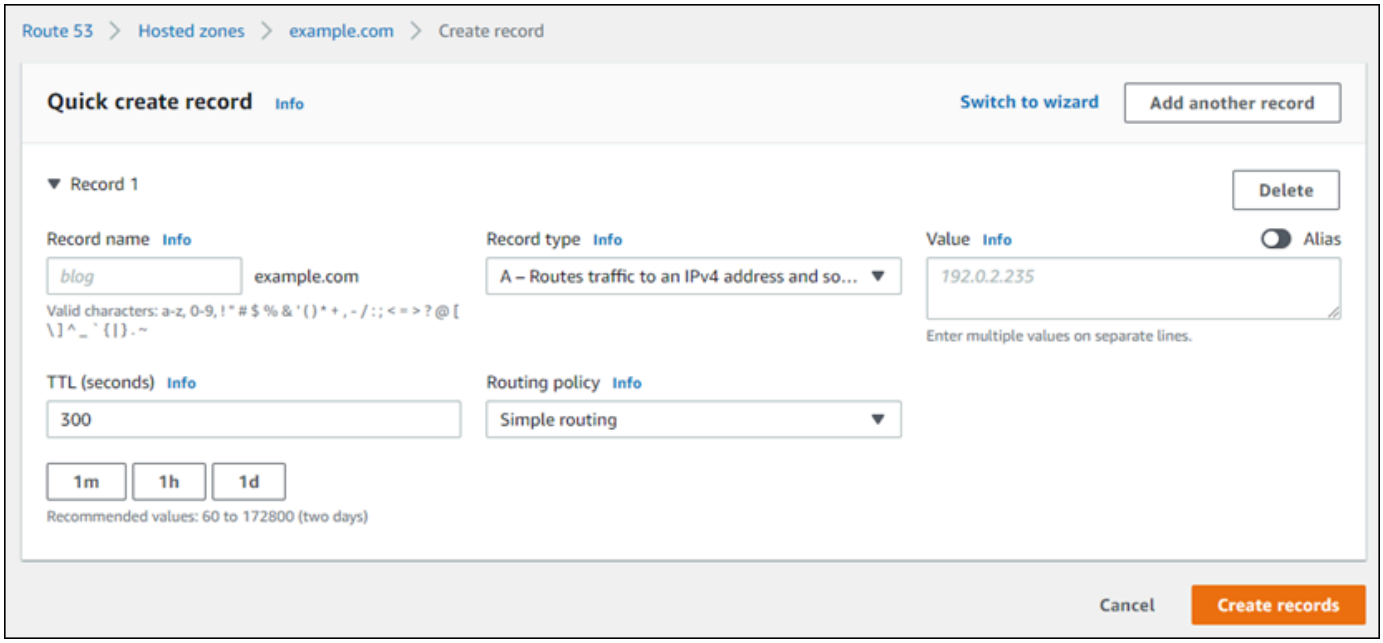
Route 53를 사용하여 도메인을 Lightsail 인스턴스에 지정

도메인을 Lightsail 인스턴스로 지정하려면 다음 단계를 완료하여 Route 53에서 가장 일반적인 두 DNS 레코드인 주소 및 정식 이름을 구성합니다.

> **Note**
>
> 이 절차는 Route 53 개발자 안내서에도 문서화되어 있습니다. 자세한 내용은 Route 53 설명서에서 Amazon Route 53 콘솔을 사용하여 레코드 생성을 참조하세요.

1. Route 53 콘솔에 로그인합니다.
2. 탐색 창에서 호스팅 영역(Hosted zones)을 선택합니다.
3. 트래픽 경로를 로드 밸런서로 지정하는 데 사용할 도메인 이름이 있는 호스팅 영역을 선택합니다.
4. 레코드 세트 생성을 선택합니다.

   레코드 빠른 생성(Quick create record) 페이지가 나타납니다.



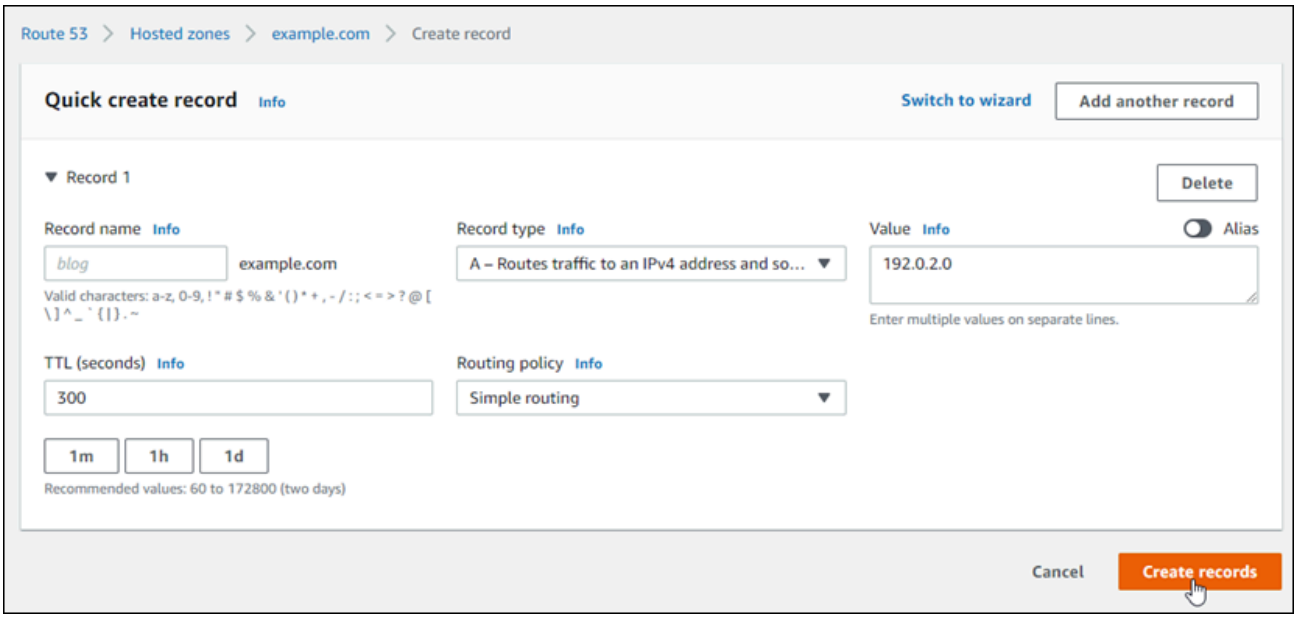
> **Note**
>
> 라우팅 정책 선택(Choose routing policy) 페이지가 표시되면 빠른 생성으로 전환(Switch to quick create)을 선택하여 빠른 생성 마법사로 전환한 후 다음 단계를 이어서 진행합니다.

5. 레코드 유형(Record type)에서 다음 옵션 중 하나를 선택합니다.

A - IPv4 주소 및 일부 AWS 리소스로 트래픽 라우팅(A - Routes traffic to an IPv4 address and some AWS resources)

주소 (A) 레코드는 도메인(예: `example.com`)이나 하위 도메인(예: `blog.example.com`)을 웹 서버의 IP 주소로 매핑합니다(예: `192.0.2.255`).

1. 레코드 이름(Record name) 텍스트 상자를 비워두고 도메인의 정점(예: `example.com`)을 IP 주소로 연결하거나 하위 도메인을 입력합니다.
2. 레코드 유형(Record type) 드롭다운 메뉴에서 A - IPv4 주소 및 일부 AWS 리소스로 트래픽 라우팅(A - Routes traffic to an IPv4 address and some AWS resources)을 선택합니다.
3. 값(Value) 텍스트 상자에 Lightsail 인스턴스의 고정 IP 주소(퍼블릭 IP 주소)를 입력합니다.
4. TTL을 300으로 유지하고 라우팅 정책을 단순 라우팅으로 유지합니다.

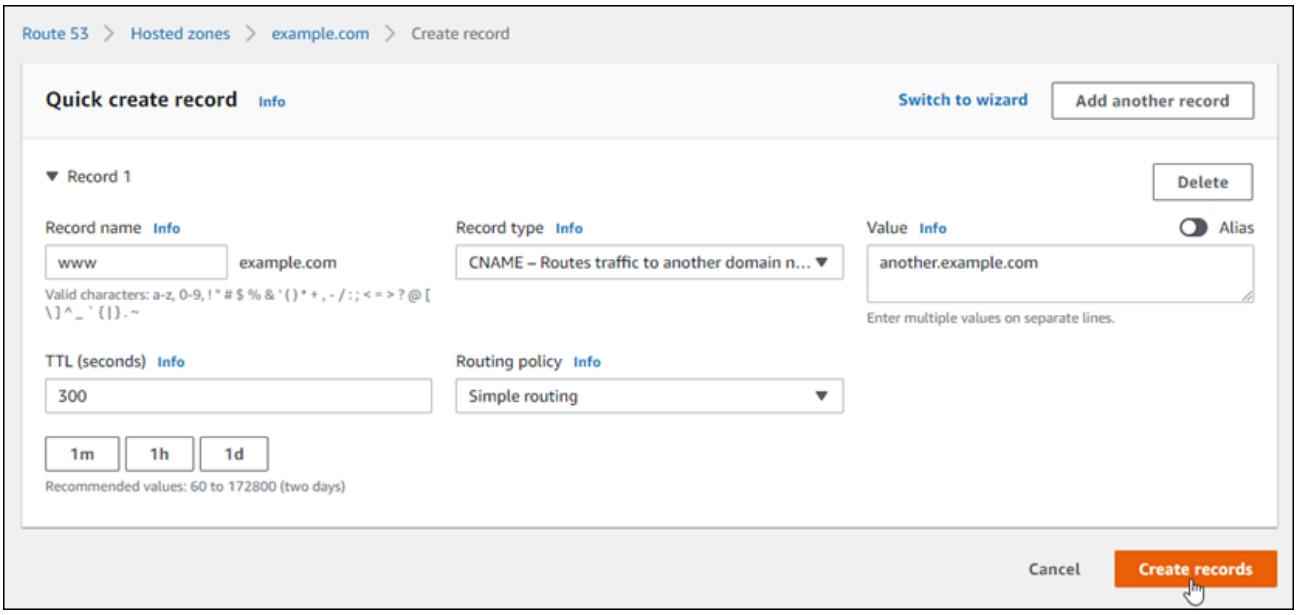


CNAME - 트래픽을 다른 도메인 이름 및 일부 AWS 리소스로 라우팅합니다.

정식 이름(CNAME) 레코드는 별칭 또는 하위 도메인(예: `www.example.com`)을 도메인(예: `example.com`)이나 하위 도메인(예: `www2.example.com`)에 매핑합니다. CNAME 레코드는 도메인을 다른 도메인으로 리디렉션합니다.

1. 레코드 이름(Record name) 텍스트 상자에 하위 도메인을 입력합니다.
2. 레코드 유형(Record type) 드롭다운 메뉴에서 CNAME - 트래픽을 다른 도메인 이름과 일부 AWS 리소스로 라우팅(CNAME - Routes traffic to another domain name and to some AWS resources)을 선택합니다.
3. 값(Value) 텍스트 상자에 도메인(예: `example.com`) 또는 하위 도메인(예: `another.example.com`)을 입력합니다.

4. TTL을 300으로 유지하고 라우팅 정책을 단순 라우팅으로 유지합니다.



6. 레코드 생성을 선택하여 호스팅 영역에 레코드를 추가합니다.

> **Note**
>
> 인터넷의 DNS를 통해 변경 사항이 전파될 때까지 기다립니다. 몇 분에서 몇 시간 정도 걸릴 수 있습니다.

Route 53 호스팅 영역에서 기존 레코드 세트를 편집하려면 편집할 레코드를 선택하고 변경 내용을 입력한 다음 저장을 선택합니다.

Lightsail에 도메인 등록

Amazon Lightsail을 사용하여 새 도메인을 등록할 수 있습니다. Lightsail 도메인은 가용성과 확장성이 뛰어난 DNS 웹 서비스인 Amazon Route 53를 통해 등록됩니다. 다른 공급자에 등록된 도메인이 있는 경우 해당 도메인의 DNS 관리를 Lightsail로 이전할 수 있습니다. 해당 도메인을 Lightsail 리소스로 지정할 수도 있습니다.

다음 절차를 하나를 선택하여 Lightsail에 새 도메인 등록:

- 새 도메인을 등록하려면 Lightsail을 사용하여 새 도메인 등록을 참조하세요.
- 기존 도메인의 경우 DNS 영역을 생성하여 도메인의 DNS 레코드 관리를 참조하세요.

- 도메인을 다른 등록 대행자로 옮기려는 경우 Amazon Route 53에서 Lightsail 도메인 관리를 참조하세요.

시작하기 전에 도메인 등록에 대한 다음 고려 사항을 참고하세요.

도메인 등록 요금

도메인 등록 요금에 대한 자세한 내용은 Amazon Route 53 요금 안내를 참조하세요.

도메인 서비스 할당량

등록할 수 있는 도메인 수에는 제한이 있습니다. 자세한 내용은 Amazon Route 53 개발자 안내서의 서비스 할당량을 참조하세요. 이러한 한도를 높이려는 경우 Route 53에 문의하세요.

지원되는 도메인

Lightsail은 모든 일반적인 최상위 도메인(TLD)의 등록을 지원합니다. 지원되는 TLD 목록은 Amazon Route 53 개발자 안내서의 Amazon Route 53에 등록할 수 있는 도메인을 참조하세요.

지리적 최상위 도메인을 등록하려면 Route 53를 사용해야 합니다. 자세한 내용은 Amazon Route 53 개발자 안내서의 지리적 최상위 도메인을 참조하세요.

등록 후 도메인 이름 변경 불가

실수로 잘못된 도메인 이름을 등록한 경우 변경할 수 없습니다. 대신 다른 도메인 이름을 등록하고 올바른 이름을 지정해야 합니다. 실수로 등록한 도메인 이름은 환불되지 않습니다.

DNS 영역 요금

Lightsail에 도메인을 등록하면 도메인에 대해 자동으로 DNS 영역이 생성됩니다. Lightsail에서는 DNS 영역에 대한 요금을 부과하지 않습니다.

Lightsail을 사용하여 새 도메인 등록

주제

- 새 도메인을 등록하기 위한 사전 조건
- 새 도메인 등록
- 도메인 연락처 정보 확인

새 도메인을 등록하기 위한 사전 조건

도메인에 필요한 DNS 레코드 유형이 Lightsail DNS 영역에서 지원되는지 확인합니다. 현재 Lightsail DNS 영역에서는 주소(A), 정식 이름(CNAME), 메일 교환기(MX), 이름 서버(NS), 서비스 로케이터(SRV) 및 텍스트(TXT) 레코드 유형이 지원됩니다. NS 레코드에는 와일드카드 DNS 레코드 항목을 사용할 수 있습니다.

도메인에 필요한 DNS 레코드 유형이 Lightsail DNS 영역에서 지원되지 않으면 Route 53를 도메인의 DNS 호스팅 공급자로 사용할 수 있습니다. Route 53은 더 많은 레코드 유형을 지원합니다. 자세한 내용은 Amazon Route 53 개발자 안내서의 지원되는 DNS 레코드 유형 및 Amazon Route 53를 기존 도메인에 대한 DNS 서비스로 설정을 참조하세요.

새 도메인 등록

새 도메인 등록

1. Lightsail 콘솔에 로그인합니다.
2. Domains & DNS(도메인 및 DNS) 탭을 선택합니다.
3. Register domain(도메인 등록)을 선택하고 등록할 도메인을 지정합니다.
   a. 등록하고자 하는 도메인 이름을 입력하고, Check(확인)를 선택하여 그 도메인 이름이 사용 가능한지 알아봅니다. 도메인이 사용 가능한 경우 Automatic domain renewal(자동 도메인 갱신)을 계속 진행합니다.
   b. 도메인을 사용할 수 없는 경우 처음 선택한 도메인 대신에 또는 처음 선택한 도메인 외에 추가로 등록할 수 있는 다른 도메인의 목록이 표시됩니다. 등록할 도메인에 대해 Select(선택)를 선택합니다.
4. 만료 날짜 전에 도메인 등록을 자동으로 갱신할지 여부를 선택합니다. 도메인 이름을 등록하면 기본적으로 1년 동안 소유하게 됩니다. 도메인 이름 등록을 갱신하지 않으면 등록이 만료되고 다른 누군가가 도메인 이름을 등록할 수 있습니다. 도메인 이름을 확실히 유지하려면 매년 자동으로 갱신하거나 더 긴 기간을 선택할 수 있습니다.
5. Domain contact information(도메인 연락처 정보) 섹션에서 도메인 등록자, 관리자, 기술 담당자의 연락처 정보를 입력합니다. 자세한 내용은 도메인을 등록하거나 이전할 때 지정하는 값을 참조하세요.

   다음과 같은 고려 사항에 유의합니다.

   이름과 성

   First name(이름)과 Last name(성)의 경우 공인 신분증에 표시된 이름을 지정하는 것이 좋습니다. 도메인 설정에 대한 일부 변경 사항의 경우, 일부 도메인은 신분 증명서를 제공하도록 요구합

니다. ID에 표시된 이름이 해당 도메인의 등록자 연락처에 기재된 이름과 정확히 일치해야 합니다.

다른 연락처

기본값으로 세 사람의 연락처에 대해 같은 정보를 사용합니다. 하나 이상의 연락처에 대해 다른 정보를 입력하려면 Same as registrant(등록자와 동일) 확인란을 선택 취소하고 새 연락처 정보를 입력합니다.

6. Privacy protection(개인 정보 보호) 섹션에서 WHOIS 쿼리에서 연락처 정보를 숨길지 여부를 선택합니다.

   자세한 내용은 다음 항목을 참조하세요.

   - 개인 정보 보호
   - Amazon Route 53에 등록할 수 있는 도메인

7. Register domain(도메인 등록)을 선택하여 계속합니다. DNS zones(DNS 영역) 및 Summary(요약) 섹션에는 도메인의 DNS 영역, 요금 및 갱신 일정에 대한 정보가 표시됩니다.
8. 도메인을 등록하려면 먼저 Amazon Route 53 도메인 이름 등록 계약을 수락해야 합니다.

도메인 연락처 정보 확인

도메인을 등록한 후 등록자 연락처 이메일 주소가 유효한지 확인해야 합니다.

다음 이메일 주소 중 하나에서 확인 이메일이 자동으로 전송됩니다.

- noreply@registrar.amazon - 등록 대행자가 Amazon Registrar인 도메인의 경우.
- noreply@domainnameverification.net - 등록 대행자가 등록 대행 협력사 Gandi인 도메인의 경우. TLD의 등록 대행자를 확인하려면 Amazon Route 53 개발자 안내서의 Amazon Route 53에 등록할 수 있는 도메인을 참조하세요.

다음 절차를 사용하여 도메인 확인 프로세스를 완료합니다.

도메인 확인 완료

1. 확인 이메일을 받은 경우 이메일 주소가 유효한지 확인하는 이메일의 링크를 선택합니다. 이메일이 즉시 도착하지 않으면 스팸 메일함을 살펴보세요.
2. Lightsail 콘솔로 돌아갑니다. 상태가 Verified(확인됨)로 자동으로 업데이트되지 않으면 Refresh status(상태 새로 고침)를 선택합니다.

> **Important**
>
> 등록자 연락처는 이메일의 지시 사항에 따라 이메일을 받았다는 사실을 확인해야 합니다. 그렇지 않으면 ICANN에서 요구할 경우 도메인을 일시 중지합니다. 도메인이 일시 중지되면 인터넷에서 접속할 수 없습니다.

3. 도메인 등록이 완료되면 DNS 서비스로 Lightsail을 사용할지, 아니면 다른 DNS 서비스를 사용할지 선택합니다.

   - Lightsail

     도메인을 등록할 때 Lightsail이 생성한 DNS 영역에서 레코드를 생성하여 도메인 및 하위 도메인의 트래픽을 라우팅하는 방식을 Lightsail에 지시합니다.

     예를 들어, 누군가 브라우저에 도메인 이름을 입력하고 그 쿼리가 Lightsail에 전달될 때, Lightsail이 웹 서버 IP 주소로 그 쿼리에 응답하길 원하십니까, 아니면 로드 밸런서의 이름으로 응답하길 원하십니까? 자세한 내용은 DNS 영역 편집 또는 삭제를 참조하세요.

   - 다른 DNS 서비스 사용

     새 도메인이 DNS 쿼리를 Lightsail을 제외한 다른 DNS 서비스로 라우팅하도록 구성합니다. 자세한 내용은 다른 DNS 서비스를 이용하고자 할 때 도메인에 대한 이름 서버 업데이트를 참조하세요.

Amazon Registrar에 등록된 도메인의 등록 세부 정보 보기

Amazon Lightsail 및 Amazon Route 53를 사용하여 등록했으며 Amazon Registrar가 등록 대행자인 .com, .net 및 .org 도메인에 대한 정보를 볼 수 있습니다. 이 정보에는 도메인을 최초로 등록한 시기와 도메인 소유자, 기술 담당자 및 관리자에 대한 연락처 정보와 같은 세부 정보가 포함되어 있습니다.

다음 사항에 유의하세요.

개인 정보 보호가 활성 상태인 경우 도메인 연락처에 이메일 보내기

도메인에 대해 개인 정보 보호가 활성 상태인 경우 등록자, 기술 담당자 및 관리자에 대한 연락처 정보가 Amazon Registrar 개인 정보 보호 서비스의 연락처 정보로 바뀝니다. 예를 들어 example.com 도메인이 Amazon Registrar에 등록되고 개인 정보 보호가 활성 상태인 경우, WHOIS 쿼리에 대한 응답의 Registrant Email(등록자 이메일) 값은 `owner1234@example.com.whoisprivacyservice.org`와 유사합니다.

개인 정보 보호가 활성 상태일 때 하나 이상의 도메인 연락처에 연락하려면 해당 이메일 주소로 이메일을 보냅니다. 이메일은 해당 연락처로 자동 전달됩니다.

악용 사례 신고

부적절한 콘텐츠, 피싱, 맬웨어 또는 스팸을 포함하여 사용 제한 정책 위반 또는 불법 활동을 보고하려면 trustandsafety@support.aws.com으로 이메일을 보내세요.

Amazon Registrar에 등록된 도메인에 대한 정보 보기

1. 웹 브라우저에서 다음 웹 사이트 중 하나로 이동합니다. 두 웹 사이트 모두 동일한 정보를 표시합니다. 하지만 서로 다른 프로토콜을 사용하며 정보를 다른 형식으로 표시합니다.
   - WHOIS: https://registrar.amazon.com/whois
   - RDAP: https://registrar.amazon.com/rdap
2. 정보를 보려는 도메인의 이름을 입력하고 검색을 선택합니다. 검색한 도메인이 Amazon Lightsail 또는 Route 53를 사용하여 등록되지 않은 경우 해당 도메인이 등록 대행자 데이터베이스에 없다는 메시지가 표시됩니다.

Lightsail에서 도메인 이름 형식 지정

사람들이 웹 사이트나 애플리케이션에 액세스하는 데 도움이 되려면 기억하기 쉬운 도메인 이름을 선택해야 합니다. 도메인 이름(및 DNS 영역의 이름과 레코드의 이름)은 마침표(.)로 구분된 일련의 레이블로 구성됩니다. 명명 요구 사항은 도메인 이름 등록 여부나 DNS 영역 또는 레코드 이름 지정 여부에 따라 다릅니다.

다음 가이드라인을 따라 도메인 이름 형식을 지정합니다.

목차

- 도메인 이름 등록에 대한 도메인 이름 형식 지정
- DNS 영역 및 레코드에 대한 도메인 이름 형식 지정
- DNS 영역 및 레코드의 이름에 별표(*) 사용
- 다음 단계

도메인 이름 등록에 대한 도메인 이름 형식 지정

도메인 이름 등록 시 도메인 이름은 1~255자여야 합니다. 도메인 이름에 대해 유효한 문자는 (a-z), (A-Z), (0-9), 하이픈(-) 및 마침표(.)입니다.

도메인 이름의 처음이나 끝에 공백을 사용하거나 하이픈을 넣을 수 없습니다. Lightsail에서는 유효한 일반 최상위 도메인(TLD) 이름을 지원합니다. 자세한 내용은 Amazon Route 53 개발자 안내서의 일반 최상위 도메인을 참조하세요.

DNS 영역 및 레코드에 대한 도메인 이름 형식 지정

DNS 영역 및 레코드의 경우 도메인 이름은 1~255자여야 합니다. 도메인 이름에 대해 유효한 문자는 (a-z), (A-Z), (0-9), 하이픈(-) 및 마침표(.)입니다. 공백을 사용할 수는 없습니다.

Lightsail은 대문자(A-Z)로 지정한 경우에도 알파벳 문자를 소문자(a-z)로 저장합니다.

Lightsail은 일반 TLD와 지리적 TLD 모두에 대해 DNS 영역을 지원합니다. 지리적 TLD의 추가 예시는 Amazon Route 53 개발자 안내서의 지리적 최상위 도메인을 참조하세요.

DNS 영역 및 레코드의 이름에 별표(*) 사용

DNS는 이름에서 별표가 표시되는 위치에 따라 별표(*) 문자를 와일드카드 문자로 처리합니다. 와일드카드 DNS 레코드는 아직 정의하지 않은 하위 도메인에 대한 DNS 요청에 응답하는 레코드입니다. Lightsail에서는 다음과 같은 조건에 따라 이름에 별표(*)가 포함된 DNS 영역 및 레코드를 생성할 수 있습니다.

DNS 영역

- 도메인 이름의 맨 왼쪽 라벨에 별표(*)를 포함할 수 없습니다. 예를 들어 *.example.com은 사용할 수 없습니다.
- 다른 위치에 별표(*)를 포함시키면 DNS가 이를 와일드카드가 아닌 ASCII 42 문자로 처리합니다. ASCII 문자에 대한 자세한 내용은 Wikipedia에서 ASCII를 참조하세요.

DNS 레코드

DNS 레코드 이름에서 별표(*)를 와일드카드로 사용할 때 다음 제한 사항에 유의하세요.

- 와일드카드로서 별표는 도메인 이름의 제일 왼쪽 라벨을 바꿔야 합니다(예: *.example.com 또는 *.acme.example.com). 다른 위치에 별표를 포함시키면(예: prod.*.example.com) DNS가 이를 와일드카드가 아닌 ASCII 42 문자로 처리합니다.
- 별표가 전체 라벨을 대신해야 합니다. 예를 들어 *prod.example.com 또는 prod.*.example.com을 지정할 수 없습니다.

- 구체적인 도메인 이름이 우선순위입니다. 예를 들어 *.example.com 및 acme.example.com에 대한 레코드를 생성하는 경우, acme.example.com에 대한 DNS 쿼리는 acme.example.com 레코드의 값으로 응답합니다.
- 별표는 별표 및 해당 하위 도메인의 모든 하위 도메인을 비롯한 하위 도메인 수준의 DNS 쿼리에 적용됩니다. 예를 들어 *.example.com이라는 레코드를 생성한 경우 *.example.com에 대한 DNS 쿼리는 다음에 응답합니다.

  zenith.example.com

  acme.zenith.example.com

  pinnacle.acme.zenith.example.com(해당 DNS 영역에 대해 어떠한 유형의 레코드도 없는 경우)

*.example.com이라는 레코드를 생성했는데 example.com 레코드가 없는 경우 Lightsail은 example.com DNS 쿼리에 NXDOMAIN(존재하지 않는 도메인)으로 응답합니다.

Lightsail을 구성하여 동일한 수준의 모든 하위 도메인 및 도메인 이름에 따라 DNS 쿼리에 동일한 응답을 반환할 수 있습니다. 예를 들어 acme.example.com 및 zenith.example.com과 같은 DNS 쿼리에 acme.example.com 레코드를 사용하여 응답하도록 Lightsail을 구성할 수 있습니다. 다음 단계를 수행하여 하위 도메인의 트래픽을 example.com 최상위 도메인으로 라우팅:

1. 도메인의 레코드(예: example.com)를 생성합니다.
2. 하위 도메인의 별칭 레코드(예: *.example.com)를 생성합니다. 이전 단계에서 생성한 기록을 별칭 레코드 대상으로 지정합니다.

다음 단계

자세한 내용은 다음 항목을 참조하세요.

- DNS 영역을 생성하여 도메인의 DNS 레코드 관리
- DNS

고급 Route 53 기능을 사용하여 Lightsail 도메인 관리

Amazon Lightsail은 가용성과 확장성이 뛰어난 DNS 웹 서비스인 Amazon Route 53를 통해 도메인을 등록합니다. Lightsail을 사용하여 도메인을 등록하면 Lightsail 및 Route 53 모두에서 도메인을 관리할 수 있습니다.

도메인 등록, 도메인의 트래픽을 Lightsail 리소스로 라우팅하는 등의 작업은 Lightsail 콘솔에서 실시됩니다. 자세한 내용은 Amazon Lightsail에서 도메인 등록을 참조하세요.

도메인 이전 및 등록 삭제와 같은 고급 작업은 Amazon Route 53 콘솔에서 수행해야 합니다.

이 설명서는 Route 53 콘솔을 사용하여 완료할 수 있는 일부 고급 관리 작업에 대한 정보를 제공합니다. Route 53의 전체 개요는 Amazon Route 53 개발자 안내서의 Amazon Route 53은 무엇인가요?을 참조하세요.

목차

- 도메인 등록 상태 보기
- 다른 등록 대행자로의 무단 이전을 방지하기 위해 도메인 잠그기
- 만료되거나 삭제된 도메인 복원
- 도메인 이전
- 도메인 이름 등록 삭제

도메인 등록 상태 보기

도메인 이름에는 확장성 프로비저닝 프로토콜(EPP) 상태 코드라고 하는 상태가 있습니다. 도메인 이름의 중앙 데이터베이스를 유지 관리하는 조직인 ICANN은 EPP 상태 코드를 개발했습니다. EPP 상태 코드는 다양한 작업의 상태를 알려줍니다. 예를 들어, 도메인 이름 등록, 도메인 이름 등록 갱신 등이 있습니다. 모든 등록 대행자는 이 상태 코드를 사용합니다. 도메인의 상태 코드를 보려면 Amazon Route 53 개발자 안내서의 도메인 등록 상태 보기를 참조하세요.

다른 등록 대행자로의 무단 이전을 방지하기 위해 도메인 잠그기

모든 일반적인 최상위 도메인(TLD)의 도메인 등록 기관에서는 다른 사람이 허가 없이 도메인을 다른 등록 대행자로 이전하지 못하도록 도메인을 잠글 수 있습니다. 자세한 내용은 Amazon Route 53 개발자 안내서의 다른 등록 대행자로의 무단 이전을 방지하기 위해 도메인 잠그기를 참조하세요.

만료되거나 삭제된 도메인 복원

추가 갱신 기간 내에 도메인을 갱신하지 않거나 도메인을 부주의로 삭제한 경우, 다른 사용자가 해당 도메인을 등록하기 전에 상위 도메인(TLD)의 일부 레지스트리를 사용하여 도메인을 복원할 수 있습니다. 링크로 연결된 절차를 사용하여 도메인 등록을 복원해 보세요. 자세한 내용은 Amazon Route 53 개발자 안내서의 만료되거나 삭제된 도메인 복원을 참조하세요.

도메인 등록 이전

도메인 등록을 다른 등록 기관에서 Route 53로, AWS 계정 간 또는 Route 53에서 다른 등록 기관으로 이전할 수 있습니다. 자세한 내용은 Amazon Route 53 개발자 안내서의 도메인 이전을 참조하세요.

도메인 이름 등록 삭제

TLD(최상위 도메인)에 대해 등록을 더 이상 원하지 않는 경우 등록을 삭제할 수 있습니다. 등록 기관에서 등록을 삭제하도록 허용하는 경우 이 항목의 절차를 수행하십시오. 자세한 내용은 Amazon Route 53 개발자 가이드의 도메인 이름 등록 삭제를 참조하십시오.

Lightsail에서 도메인 등록 또는 이전 시 도메인 정보 제공

Amazon Lightsail을 사용하여 도메인을 등록할 때는 등록 기간(기간) 및 도메인 연락처 정보와 같은 도메인 정보를 제공하게 됩니다. 또한 자동 도메인 갱신 및 개인 정보 보호를 구성합니다.

그 밖에 현재 Lightsail을 통해 등록된 도메인의 정보를 변경할 수도 있습니다.

> **Note**
>
> - 도메인에 대한 연락처 정보를 변경하면 그 변경에 관해 등록자 연락처로 이메일 알림이 전송됩니다. 이 이메일은 noreply@registrar.amazon에서 발송됩니다. 대부분의 경우, 등록자 연락처가 그 이메일에 응답할 필요는 없습니다.
> - 소유권 변경으로 여겨지는 연락처 정보의 변경에 대해서는 등록자 연락처로 추가 이메일이 전송됩니다. 도메인 이름의 중앙 데이터베이스를 유지 관리하는 조직인 ICANN은 등록자 연락처에 이메일 수신을 확인하도록 요구합니다. 자세한 내용은 이 섹션에서 이름, 성 및 조직을 참조하세요.

기존 도메인에 대한 연락처 정보 변경에 대한 자세한 내용은 도메인 연락처 정보 업데이트를 참조하세요.

제공하는 도메인 정보

- Term
- 자동 도메인 갱신
- 등록자, 관리자, 기술 담당자 및 청구 담당자 연락처
- 연락처 유형

- 이름, 성
- 조직
- 이메일
- 전화번호
- 주소 1
- 주소 2
- 국가
- 시/도
- 구/군/시
- 우편번호
- 개인 정보 보호

Term

도메인의 등록 기간입니다. 기간은 일반적으로 1년이지만 도메인을 등록하는 동안 기간을 최대 10년까지 늘릴 수 있습니다.

자동 도메인 갱신

Lightsail에서 도메인을 등록할 때 도메인이 자동 갱신되도록 구성됩니다. 자동 갱신 기간은 일반적으로 1년입니다. Lightsail이 도메인 만료 전에 도메인을 자동으로 갱신하길 원하는지 여부를 선택합니다. 등록 요금은 AWS 계정에 청구됩니다. 자세한 내용은 도메인 등록 갱신을 참조하세요.

> **Important**
>
> 자동 도메인 갱신을 비활성화하면 만료 날짜가 지날 때 도메인에 대한 등록이 갱신되지 않습니다. 그에 따라 도메인 이름에 대한 통제권을 상실할 수 있습니다.

등록자, 관리자, 기술 담당자 및 청구 담당자 연락처

도메인을 등록할 때는 다음과 같은 연락처가 필요합니다.

- 등록자 - 도메인 소유자.
- 관리자 - 도메인 관리를 책임지는 담당자.

- 기술 담당자 - 도메인에 대한 기술 변경 수행을 책임지는 담당자.
- 청구 담당자 - 도메인에 대한 청구 문의를 책임지는 담당자.

> **Note**
>
> 기본적으로 등록자에 대해 지정한 것과 동일한 정보를 사용되고 이를 다른 연락처에 적용합니다. 특정 연락처에 다른 정보를 입력하려면 등록자와 동일 선택을 취소합니다.

연락처 유형

이 연락처의 범주.

> **Note**
>
> - Company(회사) 또는 Association(협회) 옵션을 선택한 경우 조직 이름을 입력해야 합니다.
> - 일부 최상위 도메인(TLD)의 경우 개인 정보 보호 가용성은 Contact Type(연락처 유형)에서 선택한 값에 따라 다릅니다. TLD의 개인 정보 보호 설정은 Amazon Route 53에 등록할 수 있는 도메인을 참조하세요.

이름, 성

연락할 사람의 이름과 성. First Name(이름)과 Last Name(성)에는 공식 ID에 표시된 이름을 지정하는 것이 좋습니다. 도메인 설정에 대한 일부 변경 사항의 경우, 신분 증명서를 제공해야 합니다. 이 경우 ID에 표시된 이름이 해당 도메인의 등록자 연락처에 기재된 이름과 정확히 일치해야 합니다.

등록자 연락처의 이메일 주소를 변경하면, 이 이메일이 이전 이메일 및 새 이메일 주소 모두로 전송됩니다.

조직

연락처와 관련된 조직. 등록자 및 관리자 연락처의 경우에는 일반적으로 도메인을 등록하는 조직을 말합니다. 기술 담당자 연락처의 경우에는 도메인을 관리하는 조직일 수 있습니다.

연락처 유형이 Person(사람)을 제외한 다른 값이고 등록자 연락처에 대해 Organization(조직) 필드를 변경하면 도메인 소유자를 변경하게 됩니다. ICANN은 이 경우 등록자 연락처에 이메일을 보내 승인을 얻도록 규정하고 있습니다. 이메일은 다음 이메일 주소 중 하나에서 발송합니다.

- noreply@registrar.amazon - 등록 대행자가 Amazon Registrar인 도메인의 경우.
- noreply@domainnameverification.net - 등록 대행자가 등록 대행 협력사 Gandi인 도메인의 경우.

TLD의 등록 대행자를 확인하려면 Amazon Route 53에 등록할 수 있는 도메인을 참조하세요.

등록자 연락처의 이메일 주소를 변경하면, 이 이메일이 이전 이메일 및 새 이메일 주소 모두로 전송됩니다.

이메일

연락 대상의 이메일 주소.

> **Note**
>
> 등록자 연락처의 이메일 주소를 변경하면, 알림 이메일이 이전 이메일 및 새 이메일 주소 모두로 전송됩니다. 이 이메일은 noreply@registrar.amazon에서 발송됩니다.

전화번호

연락 대상의 전화번호.

- 미국 또는 캐나다 지역의 전화번호를 입력하는 경우 1을 입력한 다음 지역 번호가 포함된 10자리 전화번호를 입력합니다.
- 기타 지역의 전화번호를 입력하는 경우 국가 코드를 입력한 다음 나머지 전화번호를 입력합니다. 국가별 전화 코드 목록은 Wikipedia에서 국가별 전화 코드 목록을 참조하세요.

주소 1

연락처의 거리 주소 또는 사서함.

주소 2

아파트, 스위트, 유닛, 빌딩, 층 또는 사서함과 같은 연락처에 대한 추가 주소 정보.

국가

연락 대상의 국가.

시/도

연락 대상의 주 또는 도(해당되는 경우).

구/군/시

연락 대상의 도시.

우편번호

연락 대상의 우편 번호.

개인 정보 보호

WHOIS 쿼리로부터 연락처 정보를 감추길 원하는지 여부를 선택합니다. 도메인 연락처 정보에 대한 개인 정보 보호를 활성화하면 WHOIS 쿼리는 개인 정보 대신 도메인 등록 대행자의 연락처 정보를 반환합니다. 도메인 등록 대행자는 도메인 이름 등록을 관리하는 회사입니다.

> **Note**
>
> 관리자, 등록자 및 기술 담당자에 대해 동일한 개인 정보 설정이 적용됩니다.

도메인 연락처 정보에 대한 개인 정보 보호를 비활성화하면 지정한 이메일 주소로 더 많은 스팸 이메일을 받게 됩니다.

누구나 도메인에 대한 WHOIS 쿼리를 전송하여 그 도메인에 대한 연락처 정보 전체를 받을 수 있습니다. WHOIS 명령은 많은 운영 체제에서 사용할 수 있고, 많은 웹 사이트에서 웹 애플리케이션으로도 사용 가능합니다.

> **Important**
>
> 도메인 연락처 정보에 대한 합법적 사용자도 있지만, 가장 일반적인 사용자들은 원하지 않는 이메일 및 가짜 제안으로 도메인 연락처를 노리는 스패머들입니다. 일반적으로 연락처 정보에 대해서는 개인 정보 보호를 활성화된 상태로 두는 것이 좋습니다.

개인 정보 보호에 대한 자세한 내용은 다음 주제를 참조하세요.

- 도메인의 개인 정보 보호 관리

- Amazon Route 53에 등록할 수 있는 도메인

Lightsail에서 도메인 등록 갱신 또는 비활성화

Amazon Lightsail에서 도메인을 등록할 때 도메인이 기본적으로 자동 갱신되도록 구성됩니다. 일부 TLD(최상위 도메인) 등록 기관의 갱신 기간은 더 길지만, 기본 자동 갱신 기간은 1년입니다. 모든 일반 TLD를 통해 도메인 등록을 장기간(일반적으로 1년씩 연장하여 최대 10년) 연장할 수 있습니다.

> **Note**
>
> 를 닫으려는 경우 자동 갱신을 비활성화해야 합니다 AWS 계정. 그렇지 않으면 계정을 해지한 후에도 도메인 등록이 갱신됩니다.

목차

- 자동 갱신
- 도메인 등록 도중 도메인에 대한 자동 갱신 구성
- 이미 등록된 도메인에 대한 자동 갱신 구성

자동 갱신

다음 타임라인은 자동 갱신이 활성화된 경우 어떻게 되는지 보여줍니다.

만료 45일 전

자동 갱신이 활성 상태임을 알리는 이메일이 등록자 연락처로 전송됩니다. 이메일에는 자동 갱신을 비활성화하는 방법에 대한 지침도 포함되어 있습니다. 이 이메일을 놓치지 않도록 등록자 연락처 이메일 주소를 최신 상태로 유지하세요.

만료 35일 또는 30일 전

.com.ar, .com.br 및 .jp 도메인을 제외한 모든 도메인의 도메인 등록은 만료 35일 전에 갱신됩니다. 이렇게 하면 도메인 이름이 만료되기 전에 갱신 관련 문제를 해결할 시간을 확보할 수 있습니다.

.com.ar, .com.br 및 .jp 도메인에 대한 등록의 경우 만료 전 30일 이내에 도메인을 갱신해야 합니다. 등록 대행 협력사인 Gandi에서 만료 30일 전에 갱신 이메일을 보냅니다. 자동 갱신이 활성 상태인 경우 이 이메일은 도메인이 갱신되는 날에 발송됩니다.

자동 갱신이 비활성 상태인 경우 다음 타임라인은 도메인 이름 만료 날짜가 다가옴에 따라 어떻게 되는지 보여줍니다.

만료 45일 전

등록자 연락처에 자동 갱신이 현재 비활성 상태임을 알리는 이메일이 발송됩니다. 이메일에는 자동 갱신을 활성화하는 방법에 대한 지침도 포함되어 있습니다. 이 이메일을 놓치지 않도록 등록자 연락처 이메일 주소를 최신 상태로 유지하세요.

만료 35일 및 7일 전

도메인에 대한 자동 갱신이 비활성 상태인 경우 도메인 등록 관리 기관인 ICANN에서 등록 대행자에게 이메일을 등록자 연락처로 보내도록 요구합니다. 이메일은 다음 이메일 주소 중 하나에서 발송합니다.

noreply@registrar.amazon - 등록 대행자가 Amazon Registrar인 도메인의 경우.

noreply@domainnameverification.net - 등록 대행자가 등록 대행 협력사 Gandi인 도메인의 경우.

만료 전 30일 이내에 자동 갱신을 활성화하면 24시간 이내에 도메인 등록이 갱신됩니다.

갱신 기간에 대한 자세한 내용은 Amazon Route 53 개발자 안내서의 Amazon Route 53에 등록할 수 있는 도메인에서 TLD에 대한 '도메인 갱신 및 복원 기한' 섹션을 참조하세요.

만료 날짜 후

대부분의 도메인은 만료 후 짧은 시간 동안 등록 기관이 보유하므로 만료 날짜 후에는 만료된 도메인을 갱신하지 못할 수도 있습니다. 따라서 도메인을 유지하려면 자동 갱신을 활성화한 상태로 유지하는 것이 좋습니다. 만료 날짜 후의 도메인 갱신에 대한 자세한 내용은 Amazon Route 53 개발자 안내서의 만료되거나 삭제된 도메인 복원을 참조하세요.

도메인이 만료되었지만 추가 기간에 갱신이 가능한 경우 표준 갱신 요금으로 도메인을 갱신할 수 있습니다. 도메인이 추가 갱신 기간에 포함되는지 확인하려면 Amazon Route 53 개발자 안내서의 도메인의 등록 기간 연장 절차를 수행하세요. 도메인이 목록에 아직 있으면 아직 추가 갱신 기간이 지나지 않은 것입니다.

도메인 등록 도중 도메인에 대한 자동 갱신 구성

Lightsail에서 새 도메인 이름을 등록할 때 도메인이 자동 갱신되도록 구성됩니다. 도메인 등록 절차 중에 자동 도메인 갱신을 비활성화하도록 선택할 수 있습니다.

1. Lightsail 콘솔에 로그인합니다.

2. Domains & DNS(도메인 및 DNS) 탭을 선택합니다.
3. Register domain(도메인 등록) 버튼을 선택합니다.
4. Lightsail에 등록하려는 도메인 이름을 지정하고, 가용성 확인을 선택합니다.
5. 도메인 이름을 사용할 수 있는 경우 도메인 등록 페이지가 표시됩니다. Automatic domain renewal(자동 도메인 갱신) 섹션에서 토글 스위치를 켜거나 끄면 자동 도메인 갱신을 활성화 또는 비활성화할 수 있습니다.

이미 등록된 도메인에 대한 자동 갱신 구성

만료 날짜 직전에 Lightsail이 도메인 등록을 자동으로 갱신할지 여부를 변경하거나 자동 갱신의 현재 설정을 보려면 다음 절차를 수행합니다.

1. Lightsail 콘솔에 로그인합니다.
2. Domains & DNS(도메인 및 DNS) 탭을 선택합니다.
3. 보거나 업데이트하려는 도메인을 선택합니다.
4. Contact info(연락처 정보) 탭을 선택합니다.
5. 5. Automatic domain renewal(자동 도메인 갱신) 섹션에서 토글 스위치를 켜거나 끄면 도메인의 등록 기간에 대한 자동 갱신을 활성화 또는 비활성화할 수 있습니다.

Lightsail에서 도메인 연락처의 개인 정보 보호 관리

Amazon Lightsail에 도메인을 등록할 때 기본적으로 모든 도메인 연락처에 대해 개인 정보 보호가 활성화됩니다. 이렇게 하면 일반적으로 대부분의 연락처 정보가 WHOIS("Who is") 쿼리에서 숨겨지며 수신하는 스팸 메일의 양이 줄어듭니다. 연락처 정보는 등록 대행자의 연락처 정보 또는 "REDACTED FOR PRIVACY(개인정보 보호를 위해 편집됨)"라는 문구로 대체됩니다. 개인 정보 보호 사용에 대한 요금은 없습니다.

개인 정보 보호를 비활성화하면 누구나 도메인에 대한 WHOIS 쿼리를 보낼 수 있으며, 대부분의 최상위 도메인(TLD)의 경우 도메인 등록 시 제공한 모든 연락처 정보를 받을 수 있습니다. 여기에는 이름, 주소, 전화번호 및 이메일 주소가 포함됩니다. WHOIS 명령은 널리 사용되고 있습니다. 여러 운영 체제에 내장되어 있으며, 또한 여러 웹 사이트에서 웹 애플리케이션으로도 사용 가능합니다.

Lightsail을 사용하여 등록한 도메인의 개인 정보 보호를 관리하려면 다음 절차를 수행합니다.

목차

- 사전 조건 충족
- 도메인의 개인 정보 보호 관리

사전 조건 완료

Lightsail에 도메인을 등록합니다. 자세한 내용은 새 도메인 등록을 참조하세요.

도메인의 개인 정보 보호 관리

1. Lightsail 콘솔에 로그인합니다.
2. Domains & DNS(도메인 및 DNS) 탭을 선택합니다.
3. 개인 정보 보호를 변경하려는 도메인의 이름을 선택합니다.
4. Contact info(연락처 정보)를 선택합니다.
5. Privacy protection(개인 정보 보호) 토글 스위치를 켜거나 끄면 연락처 정보의 개인 정보 보호를 관리할 수 있습니다.

Lightsail에서 도메인 연락처 정보 업데이트

Amazon Lightsail에 도메인을 등록할 때는 도메인의 연락처 정보를 지정해야 합니다. 도메인의 연락처 정보는 도메인 소유권을 확인하고 도메인 이름과 관련된 정보를 업데이트하는 데 사용됩니다. 도메인 등록 중에 필요한 정보에 대한 자세한 내용은 Lightsail에서 도메인 등록 또는 이전 시 도메인 정보 제공 섹션을 참조하세요.

주제

- 도메인 소유자는 누구입니까?
- 도메인의 연락처 정보 업데이트

도메인 소유자는 누구입니까?

연락처 유형이 [Person]이고 등록자 연락처의 [First Name] 또는 [Last Name] 필드를 변경하는 것은 도메인 소유자를 변경하는 것이나 마찬가지입니다.

연락처 유형이 Person 외의 다른 값일 경우 Organization을 변경하면 도메인 소유자가 변경됩니다.

다음 작업은 현재 Lightsail에 등록된 도메인의 연락처 정보를 변경할 때 발생합니다.

- 도메인에 대한 연락처 정보를 변경하면 그 변경에 관해 등록자 연락처로 이메일 알림이 전송됩니다. 이 이메일은 noreply@registrar.amazon에서 발송됩니다. 대부분의 경우, 등록자 연락처가 그 이메일에 응답할 필요는 없습니다.
- 소유권 변경으로 여겨지는 연락처 정보의 변경에 대해서는 등록자 연락처로 추가 이메일이 전송됩니다. 도메인 이름의 중앙 데이터베이스를 유지 관리하는 조직인 ICANN은 등록자 연락처에 이메일 수신을 확인하도록 요구합니다.

도메인의 연락처 정보 업데이트

도메인의 연락처 정보를 업데이트하려면 다음과 같이 하세요.

1. Lightsail 콘솔에 로그인합니다.
2. Domains & DNS(도메인 및 DNS) 탭을 선택합니다.
3. 업데이트하려는 도메인의 이름을 선택합니다.
4. Contact info(연락처 정보) 탭을 선택합니다. 그런 다음 Edit contact(연락처 편집)를 선택합니다.
5. 관련 값들을 업데이트합니다. 자세한 내용은 Amazon Route 53 개발자 안내서의 도메인을 등록하거나 이전할 때 지정하는 값을 참조하세요.
6. 저장을 선택합니다.

Lightsail에서 관계형 데이터베이스 생성 및 관리

몇 개의 단계를 통해 Amazon Lightsail에서 MySQL 또는 PostgreSQL 관리형 데이터베이스를 생성할 수 있습니다. Lightsail에서는 일반적인 유지 관리 및 보안 작업을 관리하여 데이터베이스를 더 효율적으로 관리할 수 있습니다. Lightsail 콘솔을 사용하여 다음과 같은 작업을 수행할 수 있습니다.

- 데이터베이스를 스냅샷에 백업합니다.
- 스냅샷에서 새로운 더 큰 데이터베이스를 생성합니다.
- 브라우저 기반 로그 및 지표로 일반적인 문제를 해결합니다.
- 특정 시점 백업 및 복원 작업을 사용하여 데이터를 복구합니다.

Lightsail 인스턴스에 애플리케이션을 빌드하고 Lightsail 관리형 데이터베이스에 연결할 수 있습니다. 또한 독립 실행형 데이터베이스를 생성하고 회사에 대한 분석 또는 쿼리 도구를 연결할 수 있습니다. 사전 구성된 데이터베이스, SSD 기반 스토리지 및 고정된 월간 요금에 대한 데이터 전송 할당을 포함하는 고가용성 데이터베이스 플랜 또는 표준 데이터베이스 플랜 중에서 선택합니다. 또한 Lightsail(AWS Command Line Interface), API 또는 SDK를 사용하여 AWS CLI 데이터베이스를 관리할 수도 있습니다.

Lightsail 데이터베이스 생성

몇 분 안에 Amazon Lightsail에서 관리형 데이터베이스를 생성합니다. MySQL 또는 PostgreSQL의 최신 메이저 버전 중에서 선택하고 표준 또는 고가용성 플랜으로 데이터베이스를 구성할 수 있습니다.

> **Note**
>
> Lightsail의 관리형 데이터베이스에 대한 자세한 내용은 데이터베이스 선택을 참조하세요.

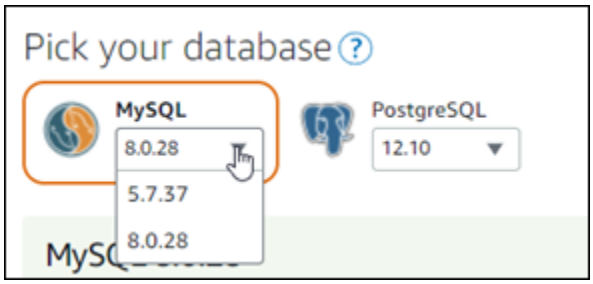
데이터베이스를 생성하려면

1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 데이터베이스를 선택합니다.
3. 데이터베이스 생성을 선택합니다.
4. 데이터베이스에 대한 AWS 리전 및 가용 영역을 선택합니다.

    1. AWS 리전 및 가용 영역 변경을 선택한 다음 리전을 선택합니다.

2. 가용 영역 변경을 선택한 다음 가용 영역을 선택합니다.

5. 데이터베이스 유형을 선택합니다. 사용 가능한 데이터베이스 엔진 옵션 중 하나에서 드롭다운 메뉴를 선택한 다음, Lightsail에서 지원되는 최신 메이저 데이터베이스 버전 두 개 중에서 하나를 선택합니다.



6. 필요한 경우 다음 옵션 중 하나를 선택합니다.

- Specify login credentials(로그인 자격 증명 지정) - 자체 데이터베이스 사용자 이름 및 암호를 지정합니다. 그렇지 않으면 Lightsail에서 자동으로 사용자 이름을 지정하고 강력한 암호를 생성합니다.
  - 자체 사용자 이름을 지정하려면 Specify login credentials(로그인 자격 증명 지정)를 선택하고 텍스트 상자에 사용자 이름을 입력합니다. 선택한 데이터베이스 엔진에 따라 다음 제약 조건이 적용됩니다.

    (MySQL)

    - MySQL의 경우 필수로 적용됩니다.
    - 1에서 16자리의 문자 또는 숫자여야 합니다.
    - 첫 번째 문자는 글자이어야 합니다.
    - 선택한 데이터베이스 엔진의 예약어는 사용할 수 없습니다. MySQL의 예약어에 대한 자세한 내용은 MySQL 5.6, MySQL 5.7 또는 MySQL 8.0용 키워드 및 예약어 문서를 참조하세요.

    (PostgreSQL)

    - PostgreSQL의 경우 필수로 적용됩니다.
    - 1~63자리의 문자 또는 숫자여야 합니다.
    - 첫 번째 문자는 글자이어야 합니다.
    - 선택한 데이터베이스 엔진의 예약어는 사용할 수 없습니다. PostgreSQL의 예약어에 대한 자세한 내용은 PostgreSQL 9.6, PostgreSQL 10, PostgreSQL 11 또는 PostgreSQL 12용 SQL 키워드 문서를 참조하세요.

- 자체 암호를 지정하려면 Create a strong password for me(나를 위한 강력한 암호 생성) 확인란을 지우고 텍스트 상자에 암호를 입력합니다. 암호에는 "/", """, "@"을 제외하고 인쇄 가능한 모든 ASCII 문자를 사용할 수 있습니다. MySQL 데이터베이스의 경우 암호에 8~41자를 포함할 수 있습니다. PostgreSQL 데이터베이스의 경우 암호에 8~128자를 포함할 수 있습니다.

- 기본 데이터베이스 이름 지정(Specify the master database name) - 자체 프라이머리 데이터베이스 이름을 지정합니다. 지정하지 않으면 Lightsail에서 자동으로 이름을 설정합니다. 자체 프라이머리 데이터베이스 이름을 지정하려면 기본 데이터베이스 이름 지정(Specify the master database name)을 선택하고 이름을 텍스트 상자에 입력합니다. 선택한 데이터베이스 엔진에 따라 다음 제약 조건이 적용됩니다.

  (MySQL)

  - 1~64개 문자 또는 숫자를 포함해야 합니다.
  - 문자로 시작해야 합니다. 이후 문자는 글자, 밑줄 또는 숫자(0~9)가 될 수 있습니다.
  - 선택한 데이터베이스 엔진의 예약어는 사용할 수 없습니다. MySQL의 예약어에 대한 자세한 내용은 MySQL 5.6, MySQL 5.7 또는 MySQL 8.0용 키워드 및 예약어 문서를 참조하세요.

  (PostgreSQL)

  - 1~63개 문자, 숫자 또는 밑줄을 포함해야 합니다.
  - 문자로 시작해야 합니다. 이후 문자는 글자, 밑줄 또는 숫자(0~9)가 될 수 있습니다.
  - 선택한 데이터베이스 엔진의 예약어는 사용할 수 없습니다. PostgreSQL의 예약어에 대한 자세한 내용은 PostgreSQL 9.6, PostgreSQL 10, PostgreSQL 11 또는 PostgreSQL 12용 SQL 키워드 문서를 참조하세요.

7. 고가용성 또는 표준 데이터베이스 플랜을 선택합니다.

   고가용성 플랜으로 생성된 데이터베이스에는 기본 데이터베이스가 있으며 장애 조치 지원을 위해 다른 가용 영역에 보조 예비 데이터베이스를 둡니다. 자세한 내용은 고가용성 데이터베이스를 참조하세요. 메모리, 처리, 스토리지 공간 및 전송 속도의 수준이 각각 다르며 요금이 다르게 책정된 데이터베이스 번들 옵션을 사용할 수 있습니다.

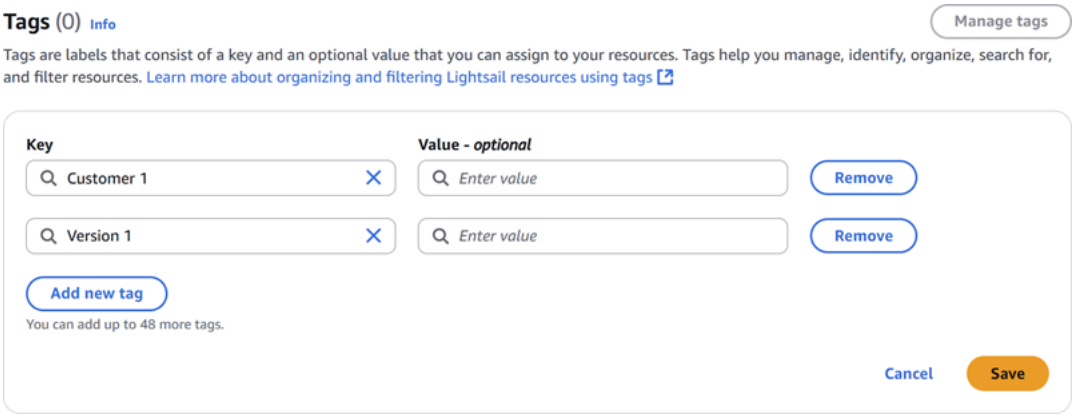
8. 데이터베이스의 이름을 입력합니다.

   리소스 이름:

   - Lightsail 계정의 각 AWS 리전 내에서 고유해야 합니다.
   - 2~255자의 문자로 구성되어야 합니다.
   - 영숫자 문자 또는 숫자로 시작하고 끝나야 합니다.
   - 영숫자 문자, 숫자, 마침표, 대시, 밑줄이 포함될 수 있습니다.

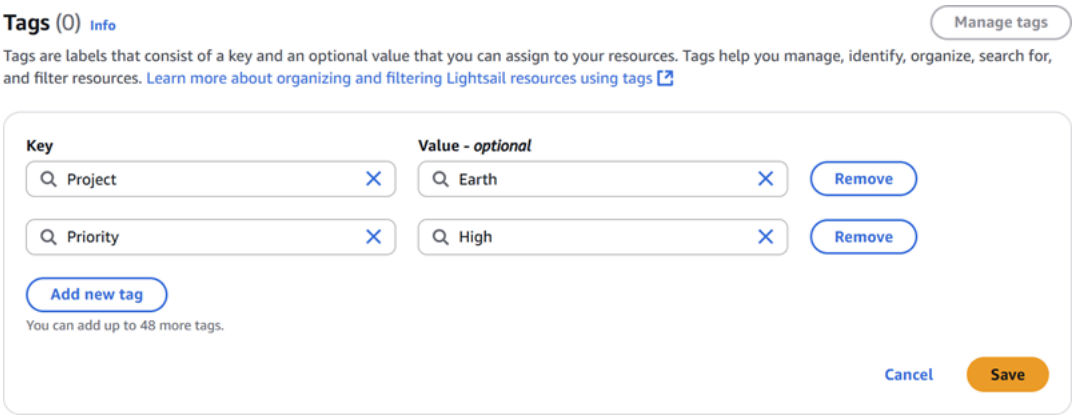
9. 다음 옵션 중 하나를 선택하여 데이터베이스에 태그를 추가합니다.

   - 키 전용 태그를 추가하거나 태그를 관리(이미 태그를 추가한 경우)합니다. 새 태그를 태그 키 텍스트 상자에 입력하고 Enter를 누릅니다. 태그를 추가하려면 태그 입력이 완료될 때 저장을 선택하고, 태그를 추가하지 않으려면 취소를 선택합니다.



   - 키-값 태그를 생성한 다음 키를 키 텍스트 상자에 입력하고, 값을 값 텍스트 상자에 입력합니다. 태그 입력이 완료되면 저장을 선택하고, 태그를 추가하지 않으려면 취소를 선택합니다.

     키-값 태그는 저장 전에 한 번에 하나씩만 추가할 수 있습니다. 둘 이상의 키-값 태그를 추가하려면 이전 단계를 반복하세요.



> **Note**
>
> 키 전용 태그 및 키-값 태그에 대한 자세한 내용은 태그를 참조하세요.

10. 데이터베이스 생성을 선택합니다.

    몇 분 내에 Lightsail 데이터베이스가 준비 완료됩니다. 데이터 가져오기에 대해 데이터베이스 구성을 시작하거나, 데이터베이스 클라이언트를 사용하여 데이터베이스에 연결할 수 있습니다.

다음 단계

다음은 새 데이터베이스를 준비 및 실행한 후 Lightsail에서 새 데이터베이스를 관리하는 데 도움이 되는 몇 가지 안내서입니다.

- 데이터베이스에 대해 데이터 가져오기 모드 구성
- 에서 데이터베이스에 대한 퍼블릭 모드 구성Amazon Lightsail
- 데이터베이스 암호 관리
- MySQL 데이터베이스에 연결
- PostgreSQL 데이터베이스에 연결
- MySQL 데이터베이스로 데이터 가져오기
- PostgreSQL 데이터베이스로 데이터 가져오기
- 데이터베이스의 스냅샷 생성

프로젝트에 적합한 Lightsail 데이터베이스 엔진 선택

Amazon Lightsail은 MySQL 및 PostgreSQL 데이터베이스의 최신 메이저 버전을 제공합니다. 이 안내서는 프로젝트에 적합한 데이터베이스를 결정하는 데 도움이 됩니다.

Lightsail은 또한 SQL 서버가 설치된 Windows Server 2022 인스턴스도 제공합니다. 자세한 내용은 Amazon Lightsail 인스턴스 이미지 선택을 참조하세요.

Lightsail에서 관리형 데이터베이스 비교

MySQL

MySQL 5.7 및 8.0은 Lightsail에서 사용할 수 있습니다. MySQL은 가장 널리 채택되는 오픈 소스 관계형 데이터베이스이며, 널리 사용되는 다양한 웹 사이트, 애플리케이션 및 상용 제품에 대한 기본 관계형 데이터 스토어 역할을 합니다. MySQL은 신뢰할 수 있고, 안정적이고, 안전한 SQL 기반 데이터베이스 관리 시스템으로, 20년 이상 커뮤니티에 개발 및 지원을 제공하고 있습니다. MySQL 데이터베이스는 미션 크리티컬 앱 및 동적 웹 사이트를 비롯한 광범위한 사용 사례에 적합합니다. 또한 소프트웨어, 하드웨어 및 어플라이언스에 대한 내장 데이터베이스 기능을 담당합니다.

> **Important**
>
> 2024년 6월 30일부터 Lightsail은 더 이상 MySQL 5.7을 지원하지 않으며 이 블루프린트로 새 데이터베이스를 생성할 수 없습니다. 데이터베이스 인스턴스의 메이저 버전을 업그레이드하는 방법을 알아보려면 Lightsail 데이터베이스의 메이저 버전 업그레이드를 참조하세요.

자세한 내용은 다음 MySQL 설명서를 참조하세요.

- MySQL 5.7 설명서
- MySQL 8.0 설명서

PostgreSQL

Lightsail에서 PostgreSQL 12, 13, 14, 15 및 16을 이용할 수 있습니다. PostgreSQL은 30년 넘게 활발하게 개발되어 온 강력한 오픈 소스 객체 관계형 데이터베이스 시스템으로서, 신뢰성, 강력한 기능 및 성능으로 좋은 평판을 얻어왔습니다.

공식 설명서를 통해 PostgreSQL의 설치 및 사용 방법에 대한 풍부한 정보를 제공하고 있습니다. PostgreSQL 커뮤니티는 기술을 익히고, 작동 방법을 알아보고, 취업 기회를 찾는 데 도움이 되는 다양한 소스를 제공합니다.

> **Important**
>
> - 2024년 6월 30일부터 Lightsail은 더 이상 PostgreSQL 11을 지원하지 않으며 이 블루프린트로 새 데이터베이스를 생성할 수 없습니다. 데이터베이스 인스턴스의 메이저 버전을 업그레이드하는 방법을 알아보려면 Lightsail 데이터베이스의 메이저 버전 업그레이드를 참조하세요.
> - PostgreSQL 커뮤니티는 PostgreSQL 2024년 11월 14일에 12의 사용을 중단할 계획이며, 이 블루프린트에서 시작된 Lightsail 인스턴스는 이 날짜 이후에는 보안 패치를 받지 않습니다. 따라서 Amazon Lightsail은 2025년 2월 28일에 PostgreSQL 12의 표준 지원을 종료합니다. 2025년 2월 28일부터 PostgreSQL 12를 사용하여 새 Lightsail 데이터베이스를 생성할 수 없습니다. 자세한 내용은 PostgreSQL 웹사이트를 참조하세요.

자세한 내용은 다음 PostgreSQL 설명서를 참조하세요.

- PostgreSQL 11 설명서

- PostgreSQL 12 설명서
- PostgreSQL 13 설명서
- PostgreSQL 14 설명서
- PostgreSQL 15 설명서
- PostgreSQL 16 설명서

데이터 가져오기 최적화

Lightsail에서는 각각 특정 메모리, vCPU, 스토리지 및 데이터 전송 허용 사양이 있는 여러 데이터베이스 플랜을 사용할 수 있습니다. 각 데이터베이스 플랜에는 이러한 사양이 있으므로 새 Lightsail 데이터베이스로 가져올 데이터 양에 적절한 크기의 데이터베이스 플랜을 선택하는 것이 중요합니다. 크기 요구 사항보다 낮은 플랜을 선택하면 데이터 가져오기 속도가 느려질 수 있습니다. 다음 지침을 통해 데이터 가져오기 요구 사항에 적합한 데이터베이스 플랜을 선택하세요.

- 월별 15 USD의 마이크로 데이터베이스 플랜 - 10GB 이상의 데이터를 전송할 경우 데이터 가져오기 속도가 느려질 수 있습니다.
- 월별 30 USD의 스몰 데이터베이스 플랜 - 20GB 이상의 데이터를 전송할 경우 데이터 가져오기 속도가 느려질 수 있습니다.
- 월별 60달러의 중간 데이터베이스 플랜 - 85GB 이상의 데이터를 전송할 경우 데이터 가져오기 속도가 느려질 수 있습니다.
- 월별 115 USD의 라지 데이터베이스 플랜 - 156GB 이상의 데이터를 전송할 경우 데이터 가져오기 속도가 느려질 수 있습니다.
- Xlarge 230 USD/월 데이터베이스 플랜 - 156GB 이상의 데이터를 전송하는 경우 데이터 가져오기 속도가 느려질 수 있습니다.
- 2Xlarge 460 USD/월 데이터베이스 플랜 - 156GB 이상의 데이터를 전송하는 경우 데이터 가져오기 속도가 느려질 수 있습니다.

> **Note**
>
> 데이터베이스에 데이터를 가져오는 방법에 대한 자세한 내용은 MySQL 데이터베이스로 데이터 가져오기 또는 PostgreSQL 데이터베이스로 데이터 가져오기를 참조하세요.

Lightsail의 고가용성 데이터베이스

Lightsail 고가용성 관리형 데이터베이스는 장애 조치 지원을 위해 한 가용 영역에 기본 데이터베이스를 두고, 다른 가용 영역에 보조 예비 데이터베이스를 둡니다. 사용량이 많으며 데이터 중복성이 필요한 프로덕션 워크로드에 고가용성 데이터베이스를 사용하는 것이 좋습니다. 개발 및 테스트 목적으로는 고가용성이 아닌 표준 데이터베이스를 사용할 수 있습니다.

고가용성 데이터베이스를 생성하려면 관리형 데이터베이스를 생성할 때 Lightsail에서 사용 가능한 고가용성 데이터베이스 플랜 중 하나를 선택하세요. 자세한 내용은 데이터베이스 생성을 참조하세요. 표준 데이터베이스를 고가용성 데이터베이스로 변경할 수도 있습니다. 표준 데이터베이스의 스냅샷을 생성하고, 이 스냅샷에서 새 데이터베이스를 생성하고, 고가용성 플랜을 선택하면 됩니다. 자세한 내용은 스냅샷에서 데이터베이스 생성을 참조하세요.

클라이언트 앱에서 Lightsail MySQL 데이터베이스에 연결

Amazon Lightsail에 MySQL 관리형 데이터베이스를 생성한 후 표준 MySQL 클라이언트 애플리케이션 또는 유틸리티를 사용하여 해당 데이터베이스에 연결할 수 있습니다. Lightsail 콘솔의 데이터베이스 관리 페이지에서 데이터베이스 엔드포인트, 포트, 사용자 이름 및 암호를 가져와야 합니다. 클라이언트 또는 웹 애플리케이션에서 데이터베이스 연결을 구성할 때 해당 값을 지정합니다.

이 안내서에서는 필수 연결 정보를 가져오고 MySQL Workbench를 구성하여 관리형 데이터베이스에 연결하는 방법을 보여줍니다.

> **Note**
>
> PostgreSQL 데이터베이스 연결에 대한 자세한 내용은 PostgreSQL 데이터베이스에 연결을 참조하세요.

1단계: MySQL 데이터베이스 연결 세부 정보 가져오기

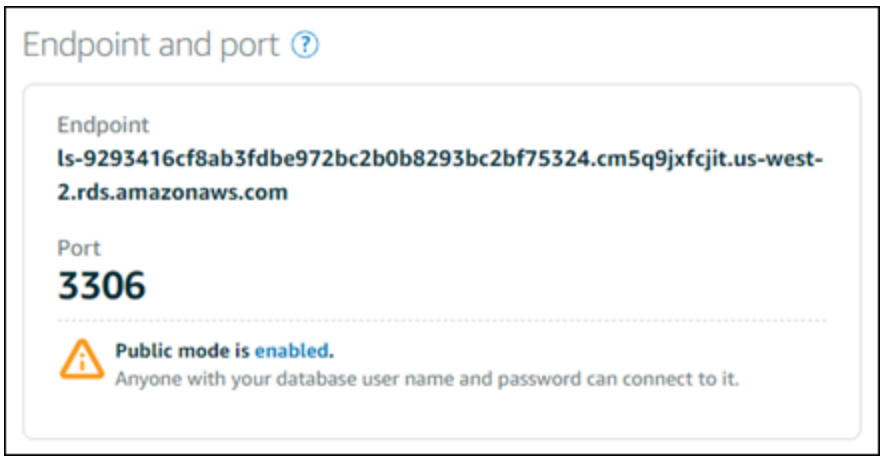
Lightsail 콘솔에서 데이터베이스 엔드포인트 및 포트 정보를 가져옵니다. 나중에 클라이언트를 구성하여 데이터베이스에 연결할 때 해당 정보를 사용합니다.

데이터베이스 연결 세부 정보를 가져오려면

1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 데이터베이스를 선택합니다.
3. 연결할 데이터베이스의 이름을 선택합니다.

4. 연결 탭의 엔드포인트 및 포트 섹션에 엔드포인트와 포트 정보를 기록합니다.

   엔드포인트를 잘못 입력하지 않도록 클립보드에 엔드포인트를 복사하는 것이 좋습니다. 이렇게 하려면 엔드포인트를 강조 표시하고 Windows를 사용하고 있는 경우 Ctrl+C를, macOS를 사용하고 있는 경우 Cmd+C를 눌러 클립보드에 복사합니다. 그런 다음 Ctrl+V 또는 Cmd+V를 적절하게 눌러 붙여넣습니다.



5. 연결(Connect) 탭의 사용자 이름 및 암호(User name and passwords) 섹션에서 사용자 이름을 기록한 다음, 암호(Password) 섹션 아래에서 표시(Show)를 선택하여 현재 데이터베이스 암호를 확인합니다.

   관리형 암호는 복잡하므로 잘못 입력하지 않도록 암호를 복사하고 붙여넣는 것이 좋습니다. 관리형 암호를 강조 표시하고 Windows를 사용하고 있는 경우 Ctrl+C를, macOS를 사용하고 있는 경우 Cmd+C를 눌러 클립보드에 복사합니다. 그런 다음 Ctrl+V 또는 Cmd+V를 적절하게 눌러 붙여넣습니다.

2단계: MySQL 데이터베이스의 퍼블릭 가용성 구성

외부에서 또는 데이터베이스 AWS 리전 와 다른의 Lightsail 인스턴스에서 데이터베이스를 연결하려면 퍼블릭 모드를 활성화해야 합니다. 퍼블릭 모드가 활성화되면 데이터베이스 사용자 이름 및 암호를 보유한 사용자는 누구나 데이터베이스에 연결할 수 있습니다. 데이터베이스의 퍼블릭 가용성을 구성하려면 데이터베이스에 대한 퍼블릭 모드 구성 가이드의 단계를 따르세요.

> **Note**
>
> 데이터베이스와 동일한 리전에 있는 Lightsail 인스턴스 중 하나에서 데이터베이스에 연결하도록 계획할 경우에는 3단계로 건너뜁니다.

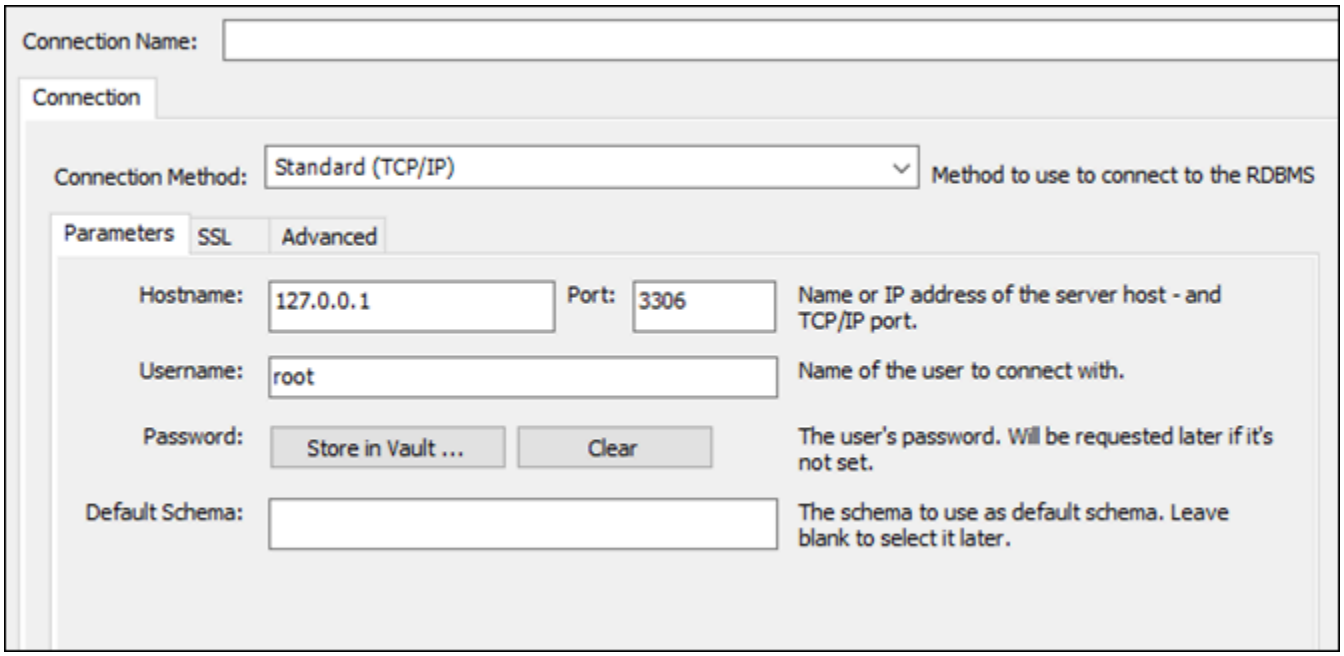
3단계: 데이터베이스 클라이언트를 구성하여 MySQL 데이터베이스에 연결

MySQL 데이터베이스에 연결하려면 이전에 받은 엔드포인트 및 포트를 사용하도록 데이터베이스 클라이언트를 구성합니다. 다음 단계들은 MySQL Workbench를 구성하는 방법을 보여주지만, 이러한 절차는 다른 클라이언트와 다를 수 있습니다.

> **Note**
>
> MySQL Workbench 사용에 대한 자세한 내용은 MySQL Workbench 매뉴얼을 참조하십시오.

MySQL Workbench를 구성하여 데이터베이스에 연결하려면

1. MySQL Workbench를 엽니다.
2. Database(데이터베이스) 메뉴를 선택한 다음 Manage connections(연결 관리)를 선택합니다.
3. 표시되는 양식에 다음 정보를 입력합니다.

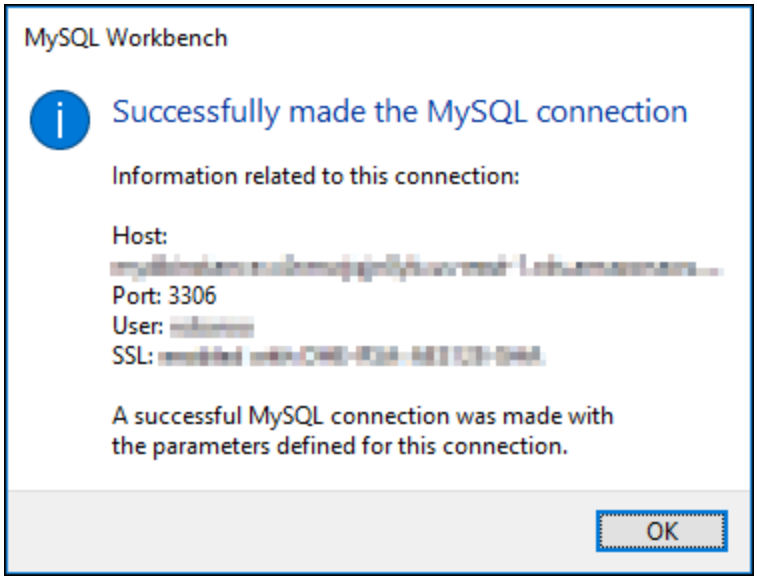


- Connection Name(연결 이름) - 연결에 대해 데이터베이스와 비슷한 이름을 사용하는 것이 좋습니다. 이렇게 하면 나중에 연결을 식별하는 데 도움이 됩니다.
- Connection Method(연결 방법) - Standard (TCP/IP)(표준 (TCP/IP))를 선택합니다.
- 포트 — 이전에 얻은 데이터베이스의 포트를 입력합니다. MySQL의 기본 포트는 3306입니다.

- Hostname(호스트 이름) - 이전에 얻은 데이터베이스 엔드포인트를 입력합니다. Lightsail 콘솔에서 데이터베이스 엔드포인트를 복사했지만 여전히 클립보드에 있는 경우 Windows를 사용하고 있는 경우 Ctrl+V를, macOS를 사용하고 있는 경우 Cmd+V를 눌러 붙여넣습니다.
- 사용자 이름 - 이전에 받은 데이터베이스 사용자 이름을 입력합니다.
- Password(암호) - Store in vault(볼트에 저장)를 선택합니다. 나타나는 창에 이전에 받은 데이터베이스 암호를 입력합니다. Lightsail 콘솔에서 암호를 복사했지만 여전히 클립보드에 있는 경우 Windows를 사용하고 있는 경우 Ctrl+V를, macOS를 사용하고 있는 경우 Cmd+V를 눌러 붙여넣습니다. OK(확인)를 선택하여 암호를 저장합니다.
- Default Schema(기본 스키마) - 이 텍스트 상자는 비워 둡니다.

4. Test connection(연결 테스트)을 선택하여 클라이언트가 데이터베이스에 연결할 수 있는지 여부를 확인합니다.

   연결이 성공하면 다음 예제와 비슷한 프롬프트가 표시됩니다. 정보를 읽은 후에는 OK(확인)를 선택하여 닫습니다.

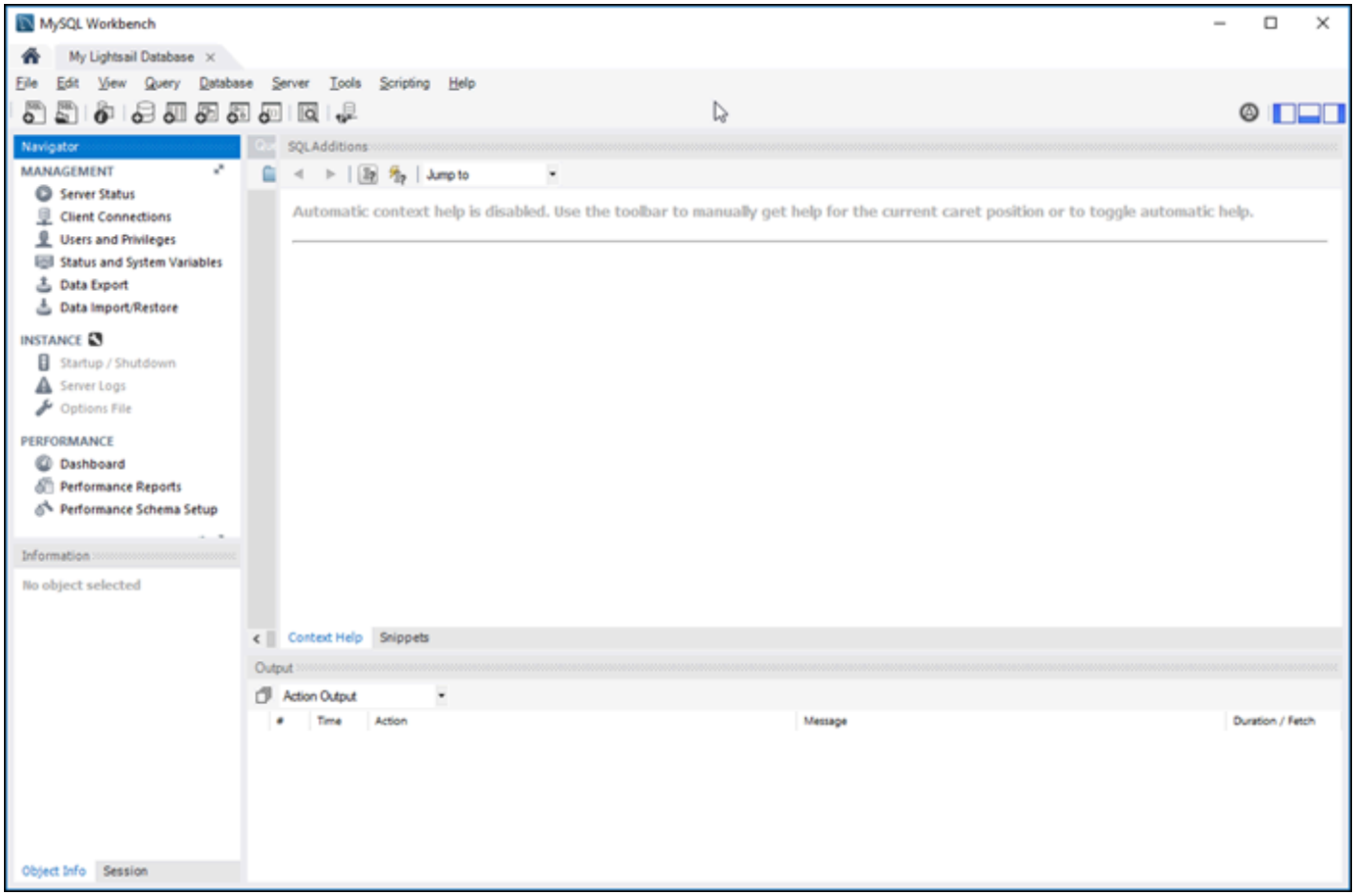


5. 신규를 선택하여 새 연결 세부 정보를 저장한 다음 Close(닫기)를 선택하여 연결 관리 창을 닫습니다.

   새 데이터베이스 연결이 MySQL Workbench 애플리케이션의 홈 페이지에 있는 MySQL Connections(MySQL 연결) 섹션에 나타납니다.

6. 데이터베이스에 연결하려면 새 데이터베이스 연결을 선택합니다.

   연결이 성공하면 다음 예제와 비슷한 창이 표시됩니다.



다음 단계

다음은 Lightsail의 데이터베이스에 데이터를 가져오는 데 도움이 되는 안내서입니다.

- MySQL 데이터베이스로 데이터 가져오기

SSL, TLS를 사용하여 Lightsail MySQL 데이터베이스에 안전하게 연결

Amazon Lightsail은 SSL 인증서를 생성하며, 프로비저닝될 때 MySQL 관리형 데이터베이스에 설치합니다. 인증서는 인증 기관(CA)에 의해 서명되고, SSL 인증서에 스푸핑 공격으로부터 보호해주는 SSL 인증서를 위한 일반 이름(CN)으로 데이터베이스 엔드포인트를 포함합니다.

Lightsail에서 생성하는 SSL 인증서는 신뢰할 수 있는 루트 개체이므로 대부분의 경우에 작동하지만 애플리케이션에서 인증서 체인을 수락하지 않을 경우 실패할 수 있습니다. 애플리케이션에서 인증서 체인을 허용하지 않는 경우 중간 인증서로 사용 중인 AWS 리전에 연결해야 할 수도 있습니다.

관리형 데이터베이스의 CA 인증서, 지원되는 AWS 리전및 애플리케이션에 대한 중간 인증서를 다운로드하는 방법에 대한 자세한 내용은 관리형 데이터베이스의 SSL 인증서 다운로드를 참조하세요

지원되는 연결

MySQL은 다음과 같은 버전에서 보안 연결을 위해 yaSSL을 사용합니다.

- MySQL 버전 5.7.19 및 5.7 이전 버전
- MySQL 버전 5.6.37 및 5.6 이전 버전
- MySQL 버전 5.5.57 및 5.5 이전 버전

MySQL은 다음과 같은 버전에서 보안 연결을 위해 OpenSSL을 사용합니다.

- MySQL 버전 8.0
- MySQL 버전 5.7.21 및 5.7 이후 버전
- MySQL 버전 5.6.39 및 5.6 이후 버전
- MySQL 버전 5.5.59 및 5.5 이후 버전

MySQL 관리형 데이터베이스는 TLS(전송 계층 보안) 버전 1.0, 1.1 및 1.2를 지원합니다. 다음 목록은 MySQL 버전에 대한 TLS 지원을 보여줍니다.

- MySQL 8.0 - TLS1.0, TLS 1.1 및 TLS 1.2
- MySQL 5.7 - TLS1.0 및 TLS 1.1 TLS 1.2는 MySQL 5.7.21 이후 버전에서만 지원됩니다.
- MySQL 5.6 - TLS1.0
- MySQL 5.5 - TLS1.0

사전 조건

- 데이터베이스에 연결하는 데 사용할 컴퓨터에 MySQL 서버를 설치합니다. 자세한 내용은 MySQL 웹 사이트에서 MySQL 커뮤니티 서버 다운로드를 참조하십시오.
- 데이터베이스에 적합한 인증서를 다운로드합니다. 자세한 내용은 관리형 데이터베이스의 SSL 인증서 다운로드를 참조하세요

SSL을 사용하여 MySQL 데이터베이스에 연결

SSL을 사용하여 MySQL 데이터베이스에 연결하려면 다음 단계를 수행합니다.

1. 터미널 또는 명령 프롬프트 창을 엽니다.
2. MySQL 데이터베이스 버전에 따라 다음 명령 중 하나를 입력합니다.

   - MySQL 5.7 이후 버전의 데이터베이스에 연결하려면 다음 명령을 입력합니다.

     ```
     mysql -h DatabaseEndpoint --ssl-ca=/path/to/certificate/rds-combined-ca-bundle.pem --ssl-mode=VERIFY_IDENTITY -u UserName -p
     ```

     명령에서 다음과 같이 바꿉니다.

     - *DatabaseEndpoint*를 데이터베이스의 엔드포인트로 바꿉니다.
     - */path/to/certificate/rds-combined-ca-bundle.pem*을 이전에 다운로드하고 데이터베이스에 대한 인증서를 저장한 로컬 경로로 바꿉니다.
     - *UserName*을 데이터베이스의 사용자 이름으로 바꿉니다.

     예:

     ```
     mysql -h ls-1c51a7c70a4fb55e542829a4e4e0d735ba42.czowadgeezqi.us-west-2.rds.amazonaws.com --ssl-ca=/home/ec2-user/rds-combined-ca-bundle.pem --ssl-mode=VERIFY_IDENTITY -u dbmasteruser -p
     ```

   - MySQL 6.7 이전 버전의 데이터베이스에 연결하려면 다음 명령을 입력합니다.

     ```
     mysql -h DatabaseEndpoint --ssl-ca=/path/to/certificate/rds-combined-ca-bundle.pem --ssl-verify-server-cert -u UserName -p
     ```

     명령에서 다음과 같이 바꿉니다.

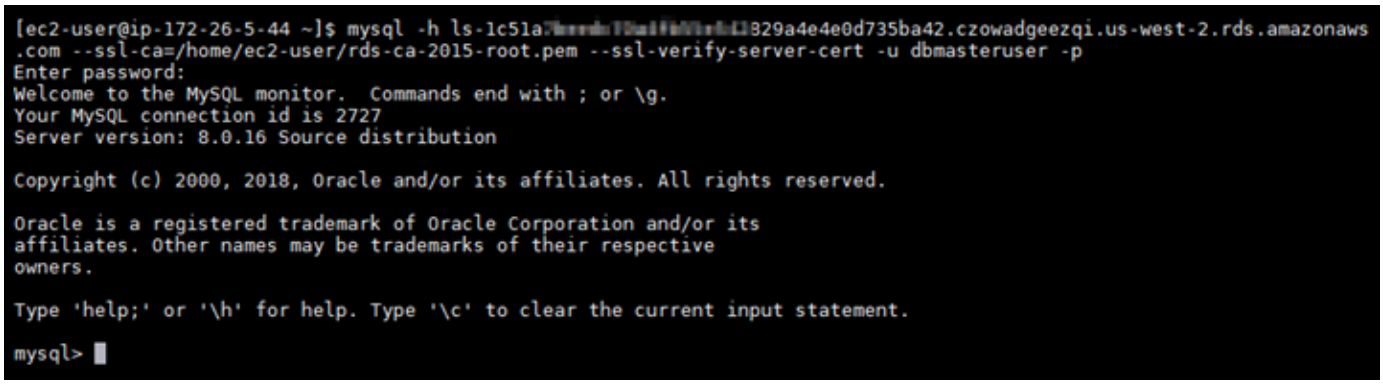
     - *DatabaseEndpoint*를 데이터베이스의 엔드포인트로 바꿉니다.
     - */path/to/certificate/rds-combined-ca-bundle.pem*을 이전에 다운로드하고 데이터베이스에 대한 인증서를 저장한 로컬 경로로 바꿉니다.
     - *UserName*을 데이터베이스의 사용자 이름으로 바꿉니다.

     예:

```
mysql -h ls-1c51a7c70a4fb55e542829a4e4e0d735ba42.czowadgeezqi.us-
west-2.rds.amazonaws.com --ssl-ca=/home/ec2-user/rds-combined-ca-bundle.pem --
ssl-verify-server-cert -u dbmasteruser -p
```

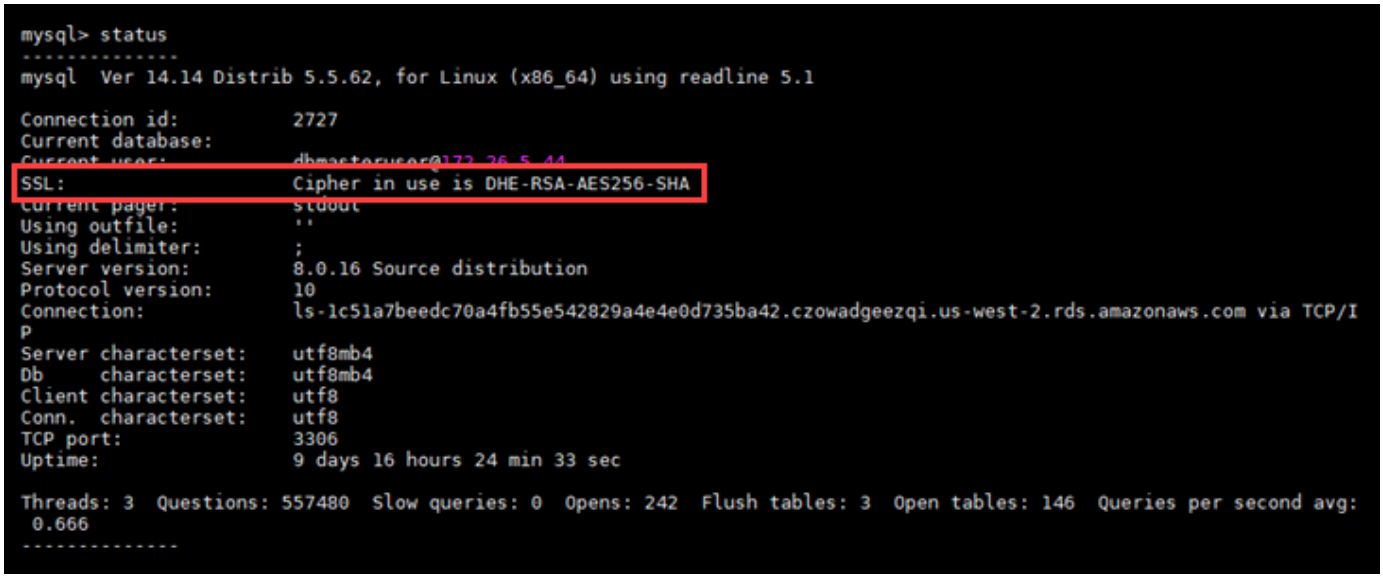
3. 메시지가 나타나면 이전 명령에서 지정한 데이터베이스 사용자의 암호를 입력하고 Enter 키를 누릅니다.

   다음 예와 비슷한 결과가 나타나야 합니다.



4. **status**를 입력하고 Enter 키를 눌러 연결 상태를 확인합니다.

   SSL 옆에 “사용 중인 암호” 값이 표시되면 연결이 암호화된 것입니다.



Lightsail PostgreSQL 데이터베이스 인스턴스에 연결

Amazon Lightsail에 PostgreSQL 관리형 데이터베이스를 생성한 후 표준 PostgreSQL 클라이언트 애플리케이션 또는 유틸리티를 사용하여 해당 데이터베이스에 연결할 수 있습니다. Lightsail 콘솔의 데이터

베이스 관리 페이지에서 데이터베이스 엔드포인트, 포트, 사용자 이름 및 암호를 가져와야 합니다. 클라이언트 또는 웹 애플리케이션에서 데이터베이스 연결을 구성할 때 해당 값을 지정합니다.

이 안내서에서는 필수 연결 정보를 가져오고 pgAdmin 클라이언트를 구성하여 관리형 데이터베이스에 연결하는 방법을 보여줍니다.

> **Note**
>
> MySQL 데이터베이스 연결에 대한 자세한 내용은 MySQL 데이터베이스에 연결을 참조하세요.

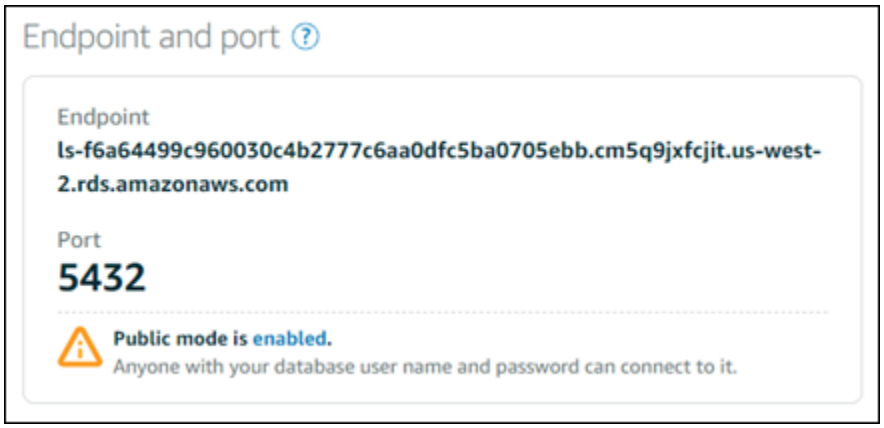
1단계: PostgreSQL 데이터베이스 연결 세부 정보 가져오기

Lightsail 콘솔에서 데이터베이스 엔드포인트 및 포트 정보를 가져옵니다. 나중에 클라이언트를 구성하여 데이터베이스에 연결할 때 해당 정보를 사용합니다.

데이터베이스 연결 세부 정보를 가져오려면

1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 데이터베이스를 선택합니다.
3. 연결할 데이터베이스의 이름을 선택합니다.
4. 연결 탭의 엔드포인트 및 포트 섹션에 엔드포인트와 포트 정보를 기록합니다.

   엔드포인트를 잘못 입력하지 않도록 클립보드에 엔드포인트를 복사하는 것이 좋습니다. 이렇게 하려면 엔드포인트를 강조 표시하고 Windows를 사용하고 있는 경우 Ctrl+C를, macOS를 사용하고 있는 경우 Cmd+C를 눌러 클립보드에 복사합니다. 그런 다음 Ctrl+V 또는 Cmd+V를 적절하게 눌러 붙여넣습니다.



5. 연결(Connect) 탭의 사용자 이름 및 암호(User name and passwords) 섹션에서 사용자 이름을 기록한 다음, 암호(Password) 섹션 아래에서 표시(Show)를 선택하여 현재 데이터베이스 암호를 확인합니다.

   관리형 암호는 복잡하므로 잘못 입력하지 않도록 암호를 복사하고 붙여넣는 것이 좋습니다. 관리형 암호를 강조 표시하고 Windows를 사용하고 있는 경우 Ctrl+C를, macOS를 사용하고 있는 경우 Cmd+C를 눌러 클립보드에 복사합니다. 그런 다음 Ctrl+V 또는 Cmd+V를 적절하게 눌러 붙여넣습니다.

2단계: PostgreSQL 데이터베이스의 퍼블릭 가용성 구성

외부에서 또는 데이터베이스와 다른 리전의 Lightsail 인스턴스에서 데이터베이스에 연결하려면 데이터베이스의 퍼블릭 모드를 활성화해야 합니다. 퍼블릭 모드가 활성화되면 데이터베이스 사용자 이름 및 암호를 보유한 사용자는 누구나 데이터베이스에 연결할 수 있습니다. 데이터베이스의 퍼블릭 가용성을 구성하려면 데이터베이스에 대한 퍼블릭 모드 구성 가이드의 단계를 따르세요.

> **Note**
>
> 데이터베이스와 동일한 리전에 있는 Lightsail 인스턴스 중 하나에서 데이터베이스에 연결하도록 계획할 경우에는 3단계로 건너뜁니다.

3단계: 데이터베이스 클라이언트를 구성하여 PostgreSQL 데이터베이스에 연결

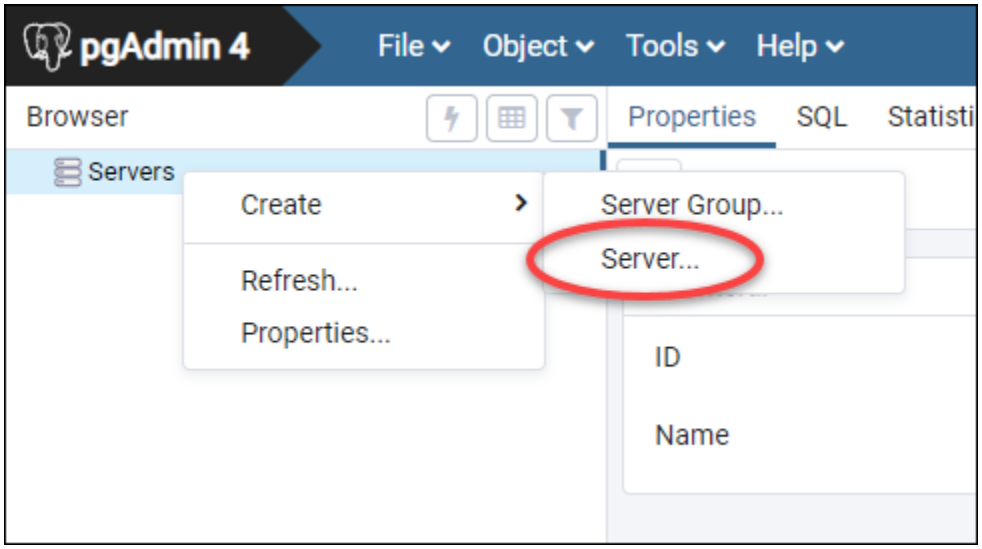
PostgreSQL 데이터베이스에 연결하려면 이전에 얻은 엔드포인트 및 포트를 사용하도록 데이터베이스 클라이언트를 구성합니다. 다음 단계들은 pgAdmin을 구성하는 방법을 보여주지만, 이러한 절차는 다른 클라이언트와 다를 수 있습니다.

> **Note**
>
> pgAdmin 사용에 대한 자세한 내용은 pgAdmin 설명서를 참조하십시오.

pgAdmin를 구성하여 데이터베이스에 연결하려면

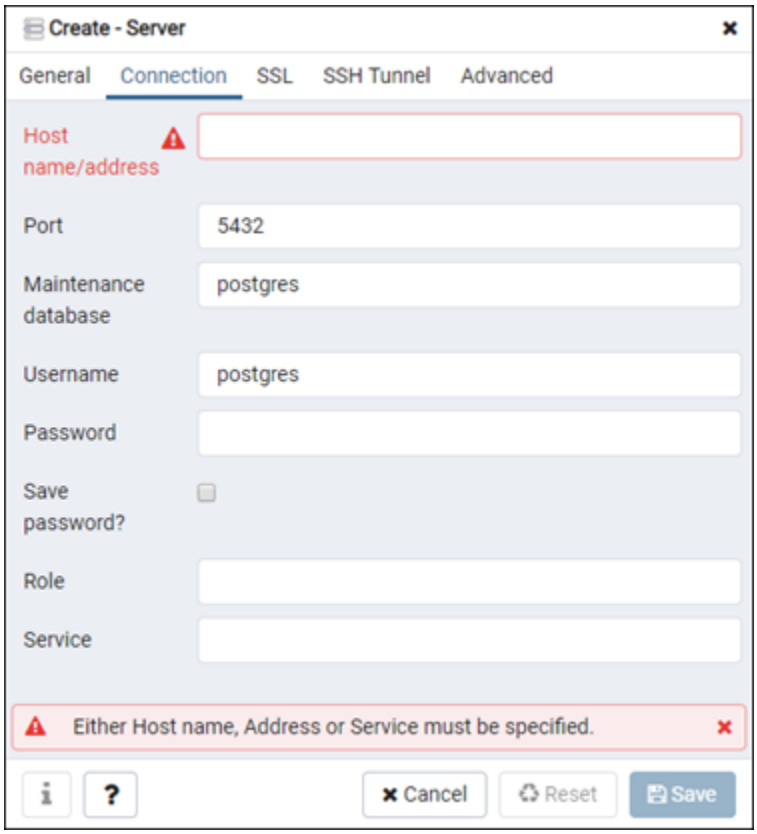
1. pgAdmin을 엽니다.
2. 왼쪽 탐색 창에서 Servers(서버)를 선택합니다.

3. 생성과 서버를 차례로 선택합니다.

4. 

5. Create - Server(생성 - 서버) 양식에 서버의 이름을 입력합니다. 연결에 대해 데이터베이스와 비슷한 이름을 사용하는 것이 좋습니다. 이렇게 하면 나중에 연결을 식별하는 데 도움이 됩니다.

6. 연결 탭을 선택한 다음, 표시된 양식에 다음 정보를 입력합니다.



- Host name/address(호스트 이름/주소) — 이전에 얻은 데이터베이스 엔드포인트를 입력합니다. Lightsail 콘솔에서 데이터베이스 엔드포인트를 복사했지만 여전히 클립보드에 있는 경우

Windows를 사용하고 있는 경우 Ctrl+V를, macOS를 사용하고 있는 경우 Cmd+V를 눌러 붙여넣습니다.

- 포트 — 이전에 얻은 데이터베이스의 포트를 입력합니다. PostgreSQL의 기본값은 5432입니다.
- Maintenance database(유지 관리 데이터베이스) - 클라이언트가 연결될 초기 데이터베이스의 이름을 지정합니다. 이는 Lightsail에서 PostgreSQL 데이터베이스를 생성할 때 지정했던 프라이머리 데이터베이스 이름입니다.

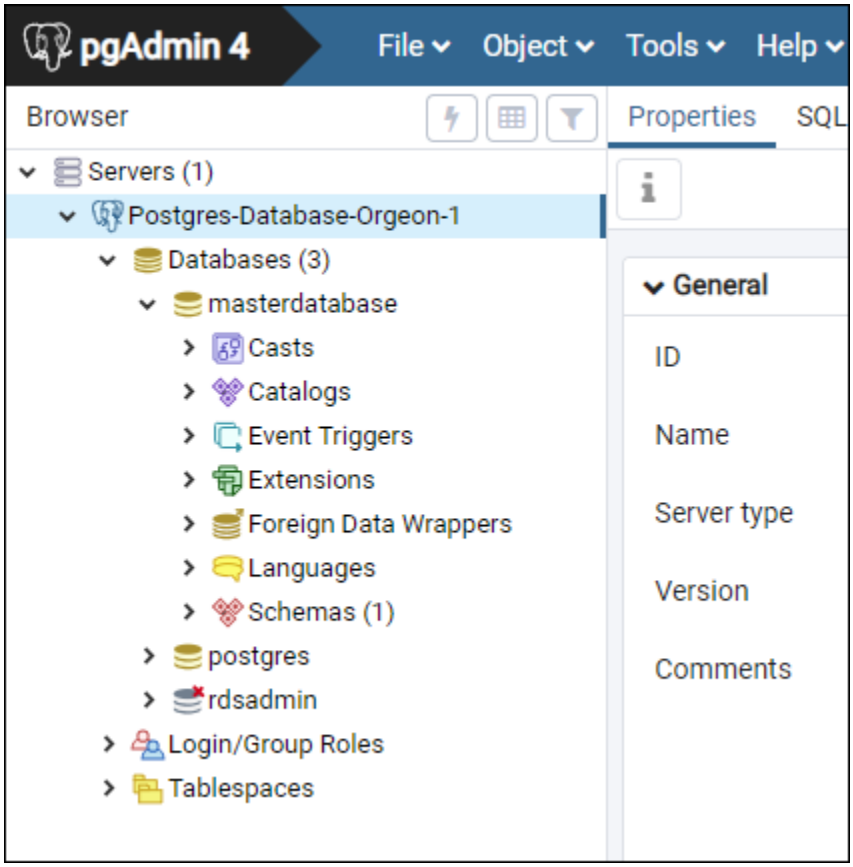
  프라이머리 데이터베이스의 이름이 기억나지 않으면 `postgres`를 입력합니다. 모든 PostgreSQL 관리형 데이터베이스에는 PostgreSQL 관리형 데이터베이스의 기타 모든 데이터베이스에 액세스할 수 없게 된 이후에 연결할 수 있는 `postgres` 데이터베이스가 포함되어 있습니다.
- 사용자 이름 - 이전에 받은 데이터베이스 사용자 이름을 입력합니다.
- 암호 - 이전에 얻은 데이터베이스 암호를 입력합니다. Lightsail 콘솔에서 암호를 복사했지만 여전히 클립보드에 있는 경우 Windows를 사용하고 있는 경우 Ctrl+V를, macOS를 사용하고 있는 경우 Cmd+V를 눌러 붙여넣습니다. Save password(암호 저장)를 선택하여 암호를 저장합니다.
- 역할 및 서비스 - 이들 필드는 비워둡니다.

7. 저장을 선택하여 새 서버 세부 정보를 저장합니다.

   Servers(서버) 섹션 아래 pgAdmin 애플리케이션의 왼쪽 탐색 메뉴에 새로운 데이터베이스 연결이 표시됩니다.

8. 데이터베이스에 연결하려면 새 데이터베이스 연결을 두 번 클릭합니다.

   연결에 성공하면 해당 데이터베이스에서 사용 가능한 리소스의 목록이 표시됩니다.



다음 단계

다음은 Lightsail의 데이터베이스에 데이터를 가져오는 데 도움이 되는 안내서입니다.

- PostgreSQL 데이터베이스로 데이터 가져오기

SSL을 사용하여 Lightsail PostgreSQL 데이터베이스에 안전하게 연결

Amazon Lightsail은 SSL 인증서를 생성하며, 프로비저닝될 때 PostgreSQL(Postgres) 관리형 데이터베이스에 설치합니다. 인증서는 인증 기관(CA)에 의해 서명되고, SSL 인증서에 스푸핑 공격으로부터 보호해주는 SSL 인증서를 위한 일반 이름(CN)으로 데이터베이스 엔드포인트를 포함합니다.

Lightsail에서 생성하는 SSL 인증서는 신뢰할 수 있는 루트 개체이므로 대부분의 경우에 작동하지만 애플리케이션에서 인증서 체인을 수락하지 않을 경우 실패할 수 있습니다. 애플리케이션에서 인증서 체인을 허용하지 않는 경우 중간 인증서로 사용 중인 AWS 리전에 연결해야 할 수도 있습니다.

관리형 데이터베이스의 CA 인증서, 지원되는 AWS 리전및 애플리케이션에 대한 중간 인증서를 다운로드하는 방법에 대한 자세한 내용은 관리형 데이터베이스의 SSL 인증서 다운로드를 참조하세요

사전 조건

- 데이터베이스에 연결하는 데 사용할 컴퓨터에 PostgreSQL 서버를 설치합니다. 자세한 내용은 Postgres 웹 사이트의 PostgreSQL 다운로드를 참조하십시오.
- 데이터베이스에 적합한 인증서를 다운로드합니다. 자세한 내용은 관리형 데이터베이스의 SSL 인증서 다운로드를 참조하세요

SSL을 사용하여 Postgres 데이터베이스에 연결

SSL을 사용하여 Postgres 데이터베이스에 연결하려면 다음 단계를 수행합니다.

1. 터미널 또는 명령 프롬프트 창을 엽니다.
2. PostgreSQL 데이터베이스에 연결하려면 다음 명령을 입력합니다.

   ```
   psql -h DatabaseEndpoint -p 5432 "dbname=DatabaseName user=UserName sslrootcert=/
   path/to/certificate/rds-combined-ca-bundle.pem sslmode=verify-full"
   ```

   명령에서 다음과 같이 바꿉니다.

   - *DatabaseEndpoint*를 데이터베이스의 엔드포인트로 바꿉니다.
   - *DatabaseName*을 연결할 데이터베이스 이름으로 바꿉니다.
   - *UserName*을 데이터베이스의 사용자 이름으로 바꿉니다.
   - */path/to/certificate/rds-combined-ca-bundle.pem*을 이전에 다운로드하고 데이터베이스에 대한 인증서를 저장한 로컬 경로로 바꿉니다.

   예:

   ```
   psql -h ls-8e81e07f8b821917b11e1c6a0e26cb73c203.czowadgeezqi.us-
   west-2.rds.amazonaws.com -p 5432 "dbname=dbmaster user=dbmasteruser sslrootcert=/
   home/ec2-user/rds-combined-ca-bundle.pem sslmode=verify-full"
   ```

3. 메시지가 나타나면 이전 명령에서 지정한 데이터베이스 사용자의 암호를 입력하고 Enter 키를 누릅니다.

다음 예제와 비슷한 결과가 나타나야 합니다. “SSL 연결” 값이 표시되면 연결이 암호화된 것입니다.

Lightsail 데이터베이스 삭제 및 최종 스냅샷 생성

더 이상 필요하지 않은 경우 Amazon Lightsail에서 관리형 데이터베이스를 삭제합니다. 삭제하는 즉시 데이터베이스에 대한 요금 발생이 중지됩니다.

> **Note**
>
> 삭제된 데이터베이스는 복구할 수 없습니다. 이 안내서에서 다룬 단계의 일환으로 데이터베이스의 최종 스냅샷을 생성하거나 삭제 프로세스와 별도로 스냅샷을 생성할 수 있습니다. 자세한 내용은 데이터베이스의 스냅샷 생성을 참조하세요.

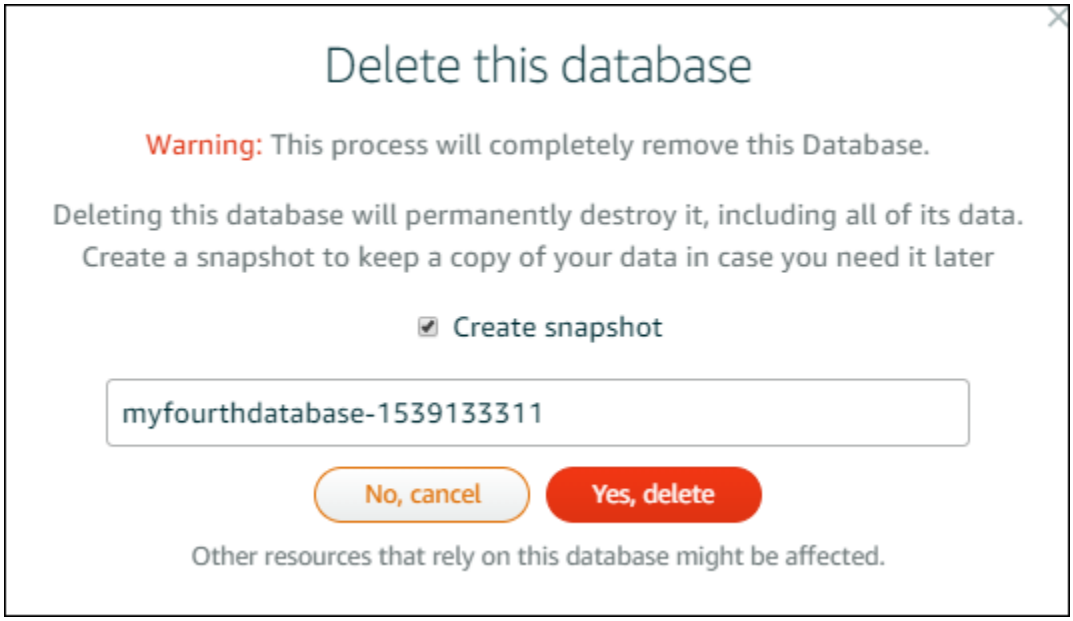
데이터베이스를 삭제하려면

1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 데이터베이스를 선택합니다.
3. 삭제할 데이터베이스의 이름을 선택합니다.
4. 삭제 탭을 선택합니다.
5. 삭제 전 스냅샷 생성 옆에 확인 표시를 추가하여 데이터베이스를 삭제하기 전에 최종 스냅샷을 생성합니다. 그런 다음 스냅샷 이름을 입력합니다.

   리소스 이름:

   - Lightsail 계정 AWS 리전 의 각 내에서 고유해야 합니다.
   - 2~255자의 문자로 구성되어야 합니다.
   - 영숫자 문자 또는 숫자로 시작하고 끝나야 합니다.
   - 영숫자 문자, 숫자, 마침표, 대시, 밑줄이 포함될 수 있습니다.

6. Delete database(데이터베이스 삭제)를 선택합니다.
7. 예, 삭제를 선택하여 삭제를 확인합니다.



삭제하기 전에 스냅샷을 생성하도록 선택한 경우 Lightsail 홈 페이지의 스냅샷 섹션에서 스냅샷을 볼 수 있습니다.

지연 없이 대용량 데이터 세트를 Lightsail 데이터베이스로 가져오기

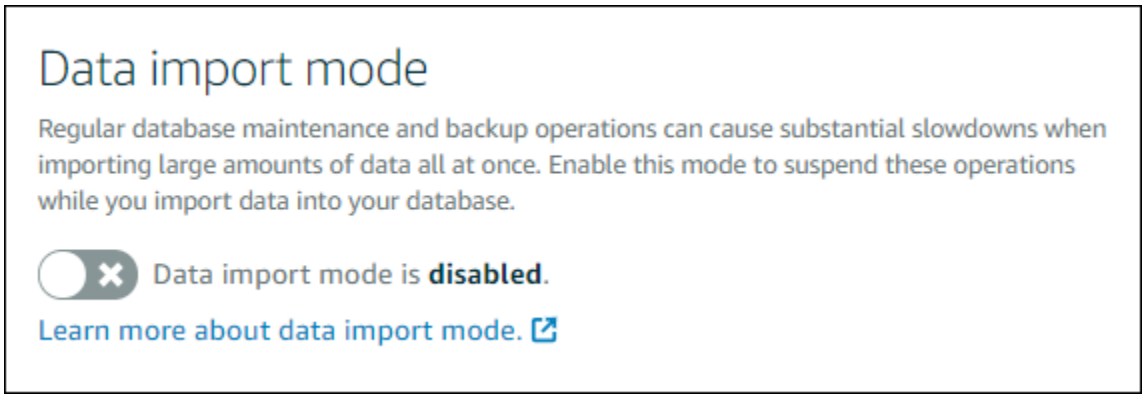
정기적인 데이터베이스 백업 작업 중에 대량의 데이터를 한 번에 가져오면 속도가 상당히 지연되거나 느려질 수 있습니다. 대량의 데이터를 가져오는 동안 Amazon Lightsail 관리형 데이터베이스에 대한 데이터 가져오기 모드를 활성화하여 이러한 작업을 일시 중지합니다.

> **Important**
>
> 데이터 가져오기 모드가 활성화되면 모든 비상 복원 백업이 삭제됩니다. 데이터 가져오기 모드가 활성화되기 전에 백업을 하고 싶은 경우에는 데이터베이스의 스냅샷을 생성합니다. 자세한 내용은 데이터베이스의 스냅샷 생성을 참조하세요.

데이터베이스에 대해 데이터 가져오기 모드를 구성하려면

1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 데이터베이스를 선택합니다.
3. 데이터 가져오기 모드를 구성하려는 데이터베이스의 이름을 선택합니다.

4. 연결 탭의 Data import mode(데이터 가져오기 모드) 섹션에서 토글을 사용하여 데이터 가져오기 모드를 켭니다. 마찬가지로 가져오기가 완료되면 토글을 사용하여 해당 모드를 끕니다.



이제 데이터 가져오기 모드가 활성화되었으므로 데이터베이스에 백업 작업이 일시 중지됩니다. 데이터 가져오기 모드는 일시적으로 활성화하는 것이 좋습니다. 대량의 데이터를 데이터베이스로 가져올 때 필요할 경우에만 이 모드를 사용하십시오. 완료한 즉시 데이터 가져오기 모드를 비활성화하여 백업 작업을 복원합니다.

> **Note**
>
> 가져오기를 수행 중인 데이터의 양에 따라 가져오기 속도가 느려질 수 있습니다. 자세한 내용은 데이터 가져오기 최적화를 참조하십시오.

Lightsail MySQL 데이터베이스로 SQL 데이터 가져오기

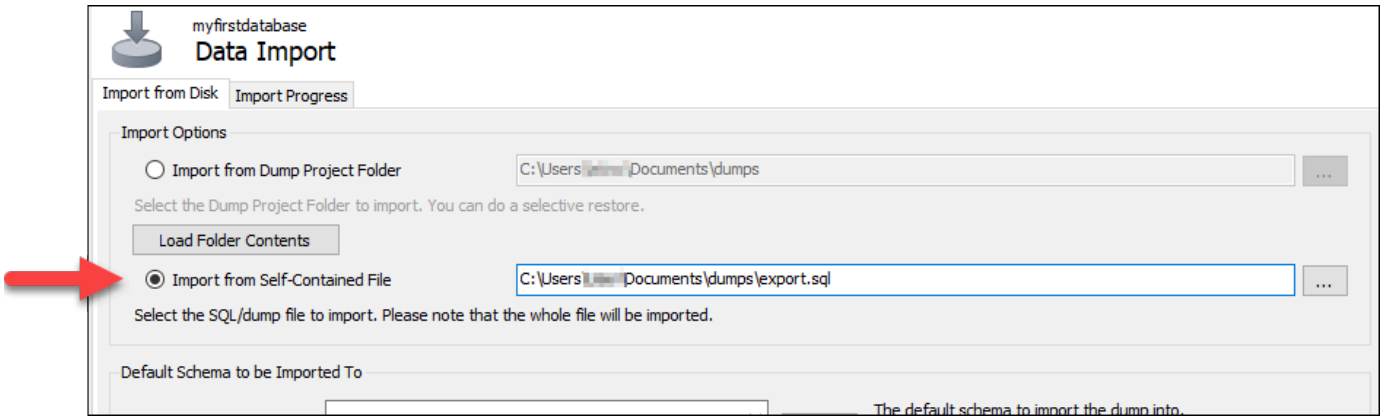
MySQL Workbench를 사용하여 Amazon Lightsail의 MySQL 관리형 데이터베이스로 SQL 파일(.SQL)을 가져올 수 있습니다.

> **Note**
>
> MySQL Workbench를 데이터베이스에 연결하는 방법에 대한 자세한 내용은 MySQL 데이터베이스에 연결을 참조하세요.

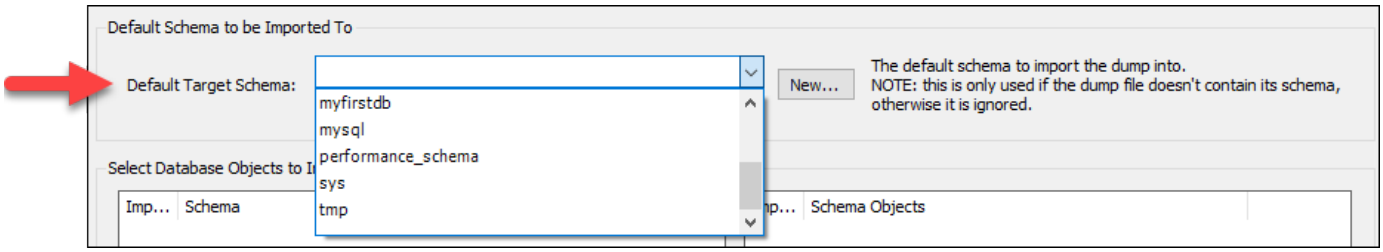
데이터베이스로 데이터를 가져오려면

1. MySQL Workbench를 엽니다.
2. MySQL 연결 목록에서 MySQL 관리형 데이터베이스를 선택합니다.

3. 왼쪽 탐색 메뉴에서 Data Import/Restore(데이터 가져오기/복원)를 선택합니다.
4. 데이터 가져오기(Data Import) 창의 가져오기 옵션(Import Options) 섹션에서 자체 포함 파일에서 가져오기(Import from Self-Contained File)를 선택합니다.

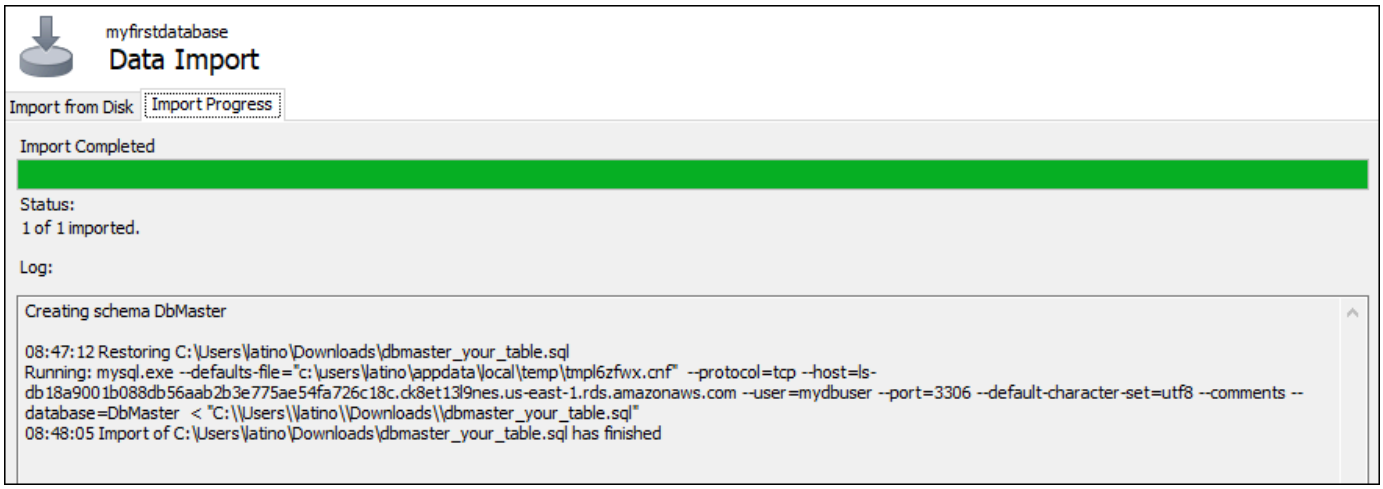


5. 줄임표 버튼을 선택하여 가져오려는 .SQL 파일에 대한 로컬 드라이브를 찾습니다.
6. 가져올 .SQL 파일을 선택한 다음 Open(열기)을 선택합니다.
7. Default Target Schema(기본 대상 스키마) 드롭다운 메뉴를 선택한 다음 파일을 가져올 기존 데이터베이스를 선택합니다. 신규를 선택하여 새 데이터베이스를 생성할 수도 있습니다.



8. Start Import(가져오기 시작)를 선택하여 가져오기를 시작합니다.

   .SQL 파일의 크기에 따라 가져오기는 몇 분 또는 그 이상 걸릴 수 있습니다. 가져오기가 완료되면 다음과 유사한 메시지가 표시되어야 합니다.



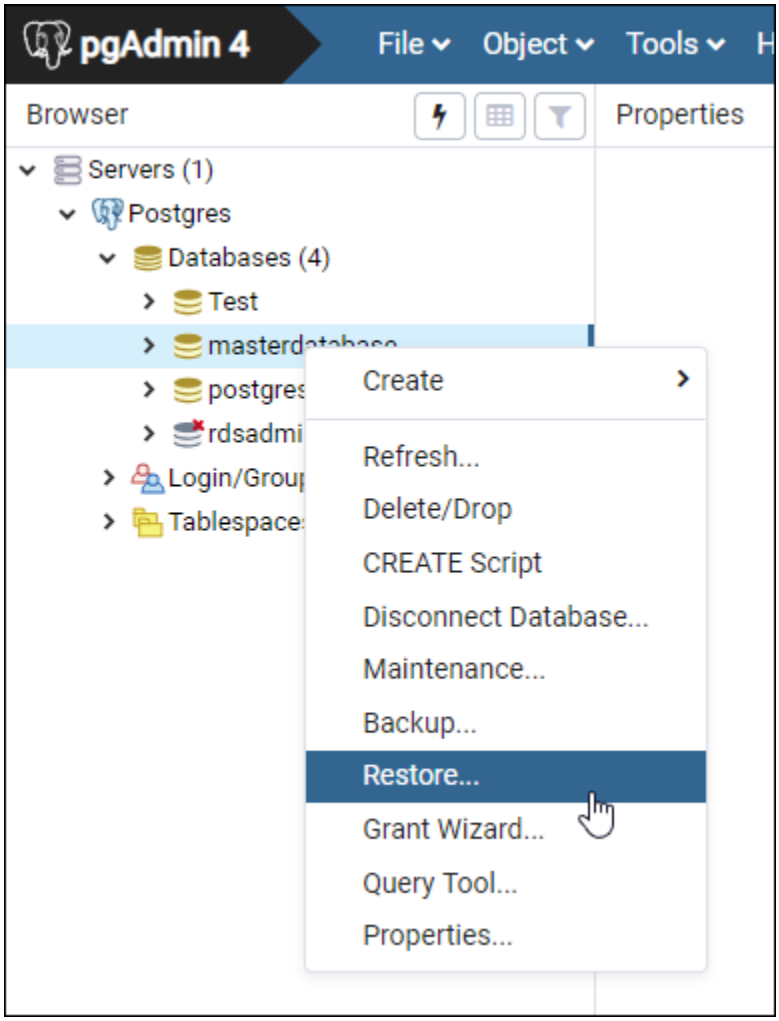
PostgreSQL 데이터베이스 백업을 Lightsail 관리형 데이터베이스로 가져오기

pgAdmin을 사용하여 Amazon Lightsail에서 PostgreSQL 관리형 데이터베이스로 데이터베이스 백업 파일을 가져올 수 있습니다.

> **Note**
>
> pgAdmin을 데이터베이스에 연결하는 방법에 대한 자세한 내용은 PostgreSQL 데이터베이스에 연결을 참조하세요. 다른 데이터베이스로 가져올 수 있는 PostgreSQL 데이터베이스 백업 생성에 대한 자세한 내용은 pgAdmin 설명서의 Backup Dialog를 참조하십시오.

데이터베이스로 백업 파일을 가져오려면

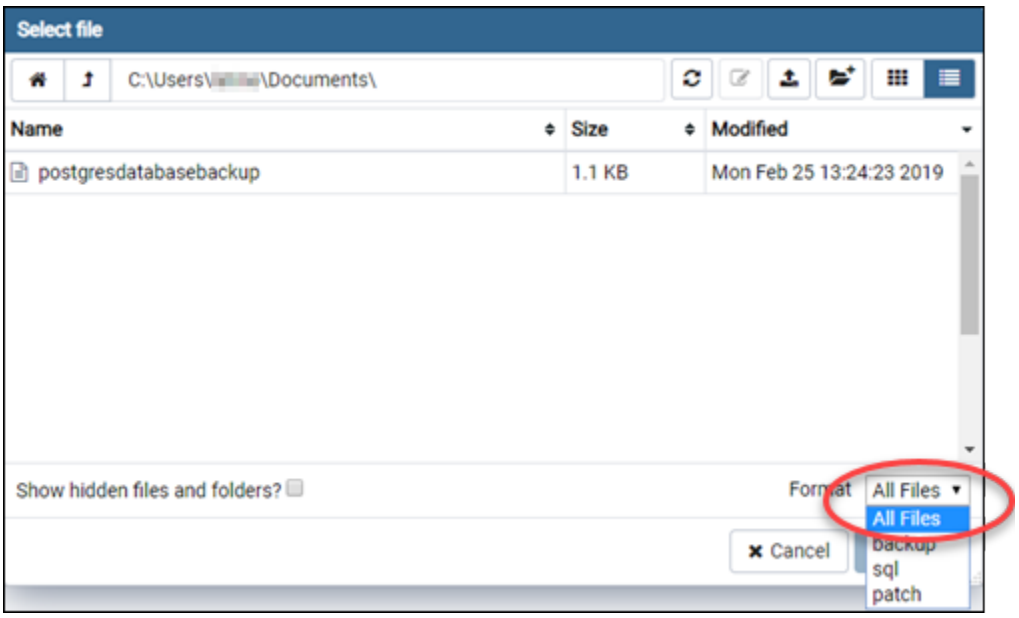
1. pgAdmin을 엽니다.
2. 서버 연결 목록에서 Amazon Lightsail의 PostgreSQL 관리형 데이터베이스를 두 번 클릭하여 연결합니다.
3. Databases(데이터베이스) 노드를 확장합니다.
4. 데이터베이스 백업 파일에서 데이터를 가져오려는 데이터베이스를 마우스 오른쪽 버튼으로 클릭한 다음 Restore(복원)를 선택합니다.



5. Restore(복원) 양식에서 다음 필드를 작성합니다.

   - Format(형식) - 백업 파일의 형식을 선택합니다.
   - Filename(파일 이름) - 줄임표 아이콘을 선택한 다음 로컬 드라이브에서 데이터베이스 백업 파일을 찾고 선택합니다. 파일이 강조 표시되면 Select(선택)를 선택하여 Restore(복원) 프롬프트로 돌아갑니다.

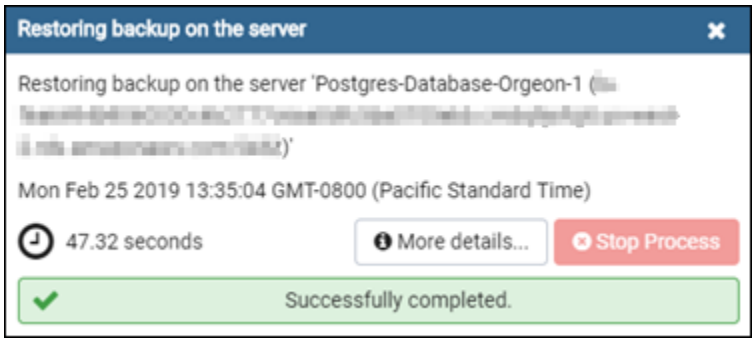
   > **Note**
   >
   > Format(형식) 드롭다운 메뉴를 선택하고 All files(모든 파일)를 선택하여 로컬 드라이브의 모든 파일 형식을 봅니다. 백업 파일은 기본적으로 선택된 파일 형식(sql)과 다른 파일 유형으로 저장될 수 있습니다.



- Number of jobs(작업 개수) 및 Role name(역할 이름) — 이 필드는 비워 둡니다.

6. Restore(복원)를 선택하여 가져오기를 시작합니다.

   데이터베이스 백업 파일의 크기에 따라 가져오기는 몇 분 또는 그 이상 걸릴 수 있습니다. 가져오기가 완료되면 다음과 유사한 메시지가 표시되어야 합니다.



Lightsail 데이터베이스 로그 및 기록 확인

데이터베이스 로그 및 변경 기록을 Amazon Lightsail 콘솔에서 확인하세요. 데이터베이스 로그는 데이터베이스 관련 문제를 진단하는 데 도움이 되는 유용한 정보를 제공합니다. 마찬가지로 데이터베이스 기록은 데이터베이스 변경 사항을 보여주므로 문제를 최근 변경 사항과 연관시킬 수 있습니다.

데이터베이스 로그를 보려면

1. Lightsail 콘솔에 로그인합니다.

2. 왼쪽 탐색 창에서 데이터베이스를 선택합니다.
3. 로그를 보려는 데이터베이스의 이름을 선택합니다.
4. Logs and history(로그 및 기록) 탭을 선택합니다.

   이 페이지에는 데이터베이스 로그 및 데이터베이스 변경 기록이 표시됩니다.
5. 데이터베이스 로그를 선택합니다. 다음 데이터베이스 로그를 사용할 수 있습니다.

   MySQL 데이터베이스 로그

   - 오류 로그 - mysqld 시작 및 종료 횟수 기록입니다. 또한 서버 시작과 종료 중에 또는 서버 실행 중에 발생하는 오류, 경고, 참고 사항과 같은 진단 메시지도 포함되어 있습니다. 자세한 내용은 MySQL 5.6, MySQL 5.7 또는 MySQL 8.0 설명서의 오류 로그 문서를 참조하세요.
   - 일반 로그 - mysqld이 수행 중인 작업에 대한 일반적인 기록입니다. 서버는 클라이언트가 연결하거나 연결을 끊을 때 이 로그에 정보를 쓰고, 클라이언트로부터 수신하는 각 SQL 문을 기록합니다. 자세한 내용은 MySQL 5.6, MySQL 5.7 또는 MySQL 8.0 설명서의 일반 쿼리 로그 문서를 참조하세요.
   - 느린 쿼리 로그 - 실행하는 데 long_query_time초 이상 걸리고 검토할 행이 최소 min_examined_row_limit개 필요한 SQL 문 기록입니다. 자세한 내용은 MySQL 5.6, MySQL 5.7 또는 MySQL 8.0 설명서의 느린 쿼리 로그 문서를 참조하세요.

   > **Note**
   >
   > 일반 쿼리 로그 및 느린 쿼리 로그는 MySQL 데이터베이스에서 기본적으로 비활성화됩니다. 몇 가지 데이터베이스 파라미터를 업데이트하면 이러한 로그를 활성화하고 데이터 수집을 시작할 수 있습니다. 자세한 내용은 Amazon Lightsail에서 MySQL 데이터베이스 일반 쿼리 로그 및 느린 쿼리 로그 활성화를 참조하세요.

   PostgreSQL 데이터베이스 로그

   - Postgres 로그 - 데이터베이스 시작 및 종료 횟수 기록입니다. 또한 데이터베이스 시작, 종료 및 데이터베이스 실행 중에 발생하는 오류, 경고, 알림 및 디버그 메시지와 같은 진단을 포함할 수 있습니다. 자세한 내용은 PostgreSQL 9.6, PostgreSQL 10, PostgreSQL 11 또는 PostgreSQL 12 설명서의 오류 보고 및 로깅 문서를 참조하세요.

주제

- 일반 및 느린 쿼리 로그인으로 Lightsail에서 MySQL 쿼리 성능 모니터링

일반 및 느린 쿼리 로그인으로 Lightsail에서 MySQL 쿼리 성능 모니터링

일반 쿼리 로그 및 느린 쿼리 로그는 Amazon Lightsail의 MySQL 데이터베이스에서 기본적으로 비활성화됩니다. 몇 가지 데이터베이스 파라미터를 업데이트하면 이러한 로그를 활성화하고 데이터 수집을 시작할 수 있습니다. Lightsail API, AWS Command Line Interface (AWS CLI) 또는 SDKs. 이 가이드에서는 AWS CLI 를 사용하여 데이터베이스 파라미터를 업데이트하고 일반 쿼리 로그와 느린 쿼리 로그를 활성화하는 방법을 보여줍니다. 또한 일반 쿼리 로그 및 느린 쿼리 로그를 제어하는 것과 로그 데이터 보존을 처리하는 방법에 대한 추가 옵션을 제공합니다.

사전 조건

AWS CLI를 아직 설치 및 구성하지 않았다면 설치하고 구성합니다. 자세한 내용은 AWS Command Line Interface 가 Amazon Lightsail과 함께 작동하도록 구성을 참조하세요.

Lightsail 콘솔에서 일반 쿼리 로그 및 느린 쿼리 로그를 활성화

Lightsail 콘솔에서 일반 쿼리 로그 및 느린 쿼리 로그를 활성화하려면 `general_log` 및 `slow_query_log` 데이터베이스 파라미터를 `1`의 값으로 업데이트하고 `log_output` 파라미터를 `FILE`의 값으로 업데이트해야 합니다.

Lightsail 콘솔에서 일반 쿼리 로그 및 느린 쿼리 로그를 활성화하려면

1. 터미널 또는 명령 프롬프트 창을 엽니다.
2. `general_log` 파라미터를 `1`의 값으로 업데이트하려면 다음 명령을 입력하세요. 이 값은 true 또는 enabled입니다.

   ```
   aws lightsail update-relational-database-parameters --
   region Region --relational-database-name DatabaseName --parameters
    "parameterName=general_log,parameterValue=1,applyMethod=pending-reboot"
   ```

   명령에서 다음과 같이 바꿉니다.

   - *DatabaseName*을 데이터베이스 이름으로 바꿉니다.
   - 데이터베이스 AWS 리전 의가 있는 *##*입니다.
3. `slow_query_log` 파라미터를 `1`의 값으로 업데이트하려면 다음 명령을 입력하세요. 이 값은 true 또는 enabled입니다.

```
aws lightsail update-relational-database-parameters --
region Region --relational-database-name DatabaseName --parameters
 "parameterName=slow_query_log,parameterValue=1,applyMethod=pending-reboot"
```

명령에서 다음과 같이 바꿉니다.

- *DatabaseName*을 데이터베이스 이름으로 바꿉니다.
- 데이터베이스 AWS 리전 의가 있는 *##*입니다.

4. log_output 파라미터를 FILE의 값으로 업데이트하려면 다음 명령을 입력하십시오. 이 값은 로그 데이터를 시스템 파일에 쓰고 Lightsail 콘솔에 데이터를 표시하도록 활성화합니다.

```
aws lightsail update-relational-database-parameters --
region Region --relational-database-name DatabaseName --parameters
 "parameterName=log_output,parameterValue=FILE,applyMethod=pending-reboot"
```

명령에서 다음과 같이 바꿉니다.

- *DatabaseName*을 데이터베이스 이름으로 바꿉니다.
- 데이터베이스 AWS 리전 의가 있는 *##*입니다.

5. 다음 명령을 입력하여 데이터베이스를 재부팅하고 변경 사항을 적용합니다.

```
aws lightsail reboot-relational-database --region Region --relational-database-
name DatabaseName
```

명령에서 다음과 같이 바꿉니다.

- *DatabaseName*을 데이터베이스 이름으로 바꿉니다.
- 데이터베이스 AWS 리전 의가 있는 *##*입니다.

이 시점에서 데이터베이스는 재부팅하는 동안 사용할 수 없게 됩니다. 몇 분 정도 기다린 다음, Lightsail 콘솔에 로그인하여 데이터베이스에 대한 일반 쿼리 로그 및 느린 쿼리 로그를 봅니다. 자세한 내용은 Amazon Lightsail에서 데이터베이스 로그 및 이력 보기를 참조하십시오.

> **Note**
>
> 데이터베이스 파라미터 업데이트에 대한 자세한 내용은 Amazon Lightsail에서 데이터베이스 파라미터 업데이트를 참조하십시오.

추가적인 데이터베이스 로그 옵션 제어

MySQL 일반 쿼리 로그 및 느린 쿼리 로그에 대한 추가 옵션을 제어하려면 다음과 같은 파라미터을 업데이트하십시오.

- `log_output` - 이 파라미터를 `TABLE`로 설정합니다. 이렇게 하면 `mysql.general_log` 테이블에 일반 쿼리 쓰기가 실행되며, `mysql.slow_log` 테이블에 느린 쿼리 쓰기가 실행됩니다. 또한 `log_output` 파라미터를 `NONE`으로 설정하면 로깅을 비활성화할 수 있습니다.

  > **Note**
  >
  > `log_output` 파라미터를 `TABLE`로 설정하면 Lightsail 콘솔에서 일반 쿼리 로그 및 느린 쿼리 로그의 데이터 표시가 비활성화됩니다. 그 대신 로그 데이터를 보려면 데이터베이스에서 `mysql.general_log` 및 `mysql.slow_log` 테이블을 참조해야 합니다.

- `long_query_time` - 빠르게 실행되는 쿼리가 느린 쿼리 로그에 기록되지 않도록 하려면 로그에 기록할 쿼리의 최단 실행 시간 값(초)을 지정하십시오. 기본값은 10초이고, 최소값은 0초입니다. `log_output` 파라미터가 `FILE`로 설정된 경우, 마이크로초 단위까지 부동 소수점 값을 지정할 수 있습니다. `log_output` 파라미터가 `TABLE`로 설정된 경우, 초 단위로 정수 값을 지정해야 합니다. 실행 시간이 `long_query_time` 파라미터 값을 초과하는 쿼리만 로그에 기록됩니다. 예를 들어 `long_query_time`을 0.1로 설정하면 100밀리초 미만의 시간 동안 작동하는 쿼리가 로그에 기록되지 않습니다.
- `log_queries_not_using_indexes` - 인덱스를 사용하지 않는 모든 쿼리를 느린 쿼리 로그에 기록하려면 1로 설정합니다. 기본값은 0입니다. 인덱스를 사용하지 않는 쿼리는 실행 시간이 `long_query_time` 파라미터의 값보다 짧아도 로그에 기록됩니다.

로그 데이터 보존

로깅을 사용하는 경우, 테이블 로그가 순환되거나 정기적으로 로그 파일이 삭제됩니다. 이러한 예방 조치를 취하면 데이터베이스 사용에 방해가 되거나 성능에 영향을 미치는 큰 로그 파일이 생성될 가능성

을 줄일 수 있습니다. log_output 파라미터가 FILE 또는 TABLE로 설정된 경우, 로깅은 다음과 같이 처리됩니다.

- FILE 로깅을 사용하는 경우, 로그 파일은 매시간 검사되며 24시간 이상 지난 로그 파일은 삭제됩니다. 경우에 따라 삭제 후 나머지 로그 파일의 총 크기가 데이터베이스에 할당된 공간 중 2%의 임계값을 초과할 수 있습니다. 이러한 경우 로그 파일의 전체 크기가 임계값 이하로 작아질 때까지 가장 큰 로그 파일부터 차례대로 삭제됩니다.
- TABLE 로깅이 활성화되면 로그 테이블은 경우에 따라 24시간마다 순환됩니다.

  테이블 로그에서 사용되는 공간이 할당된 스토리지 공간 중 20% 이상을 차지하거나 모든 로그의 총 크기가 10GB 이상일 경우 이 순환이 발생합니다.

  데이터베이스에 대해 사용된 공간의 양이 데이터베이스의 할당된 스토리지 공간 중 90% 이상을 차지할 경우, 로그 순환을 위한 임계값은 줄어듭니다.

  테이블 로그에서 사용되는 공간이 할당된 스토리지 공간 중 10% 이상을 차지하거나 모든 로그의 총 크기가 5GB 이상일 경우, 로그 테이블은 순환됩니다.

  공간 확보를 위해 로그 테이블이 순환되면 알림을 받으려면 low_free_storage 이벤트를 구독할 수 있습니다.

  - 로그 테이블이 순환되면 현재 로그 테이블은 백업 로그 테이블에 복사되며 현재 로그 테이블의 해당 항목들은 제거됩니다. 백업 로그 테이블이 이미 존재할 경우, 현재 로그 테이블이 백업으로 복사되기 전에 백업 로그 테이블이 삭제됩니다. 백업 로그 테이블을 쿼리할 수 있습니다. mysql.general_log 테이블에 대한 백업 로그 테이블 이름은 mysql.general_log_backup으로 지정됩니다. mysql.slow_log 테이블에 대한 백업 로그 테이블 이름은 mysql.slow_log_backup으로 지정됩니다.
  - mysql.rds_rotate_general_logprocedure를 호출하면 mysql.general_log 테이블을 순환할 수 있습니다. mysql.rds_rotate_slow_logprocedure를 호출하면 mysql.slow_log 테이블을 순환할 수 있습니다.
  - 데이터베이스 버전 업그레이드가 진행되는 동안 테이블 로그가 순환됩니다.

Lightsail 데이터베이스에 대한 특정 시점 백업 비활성화

다음 절차에 따라 Lightsail 관리형 데이터베이스의 특정 시점 백업을 비활성화할 수 있습니다.

> **Important**
>
> 특정 시점 백업을 사용하면 데이터베이스가 실패할 경우 데이터를 쉽게 복구할 수 있습니다. Lightsail 관리형 데이터베이스에 대해 특정 시점 백업을 활성화한 상태로 두는 것이 좋습니다.

사전 조건

AWS Command Line Interface (AWS CLI) 또는 AWS CloudShell 를 사용하여 Lightsail 데이터베이스에 대한 point-in-time 백업을 활성화하거나 비활성화합니다. CloudShell은 브라우저 기반의 사전 인증된 셸로, Lightsail 콘솔에서 직접 실행할 수 있습니다. 자세한 내용은 Lightsail 작업을 AWS CLI 위한 설정 및 구성 및 를 사용하여 Lightsail 리소스 관리 AWS CloudShell 섹션을 참조하세요.

데이터베이스 특정 시점 백업 비활성화

Lightsail에서 관리형 데이터베이스에 대한 특정 시점 백업을 비활성화하려면 AWS CLI의 `update-relational-database` Lightsail 명령을 사용하여 데이터베이스를 업데이트해야 합니다. 자세한 내용은 AWS CLI 명령 레퍼런스의 update-relational-database를 참조하세요.

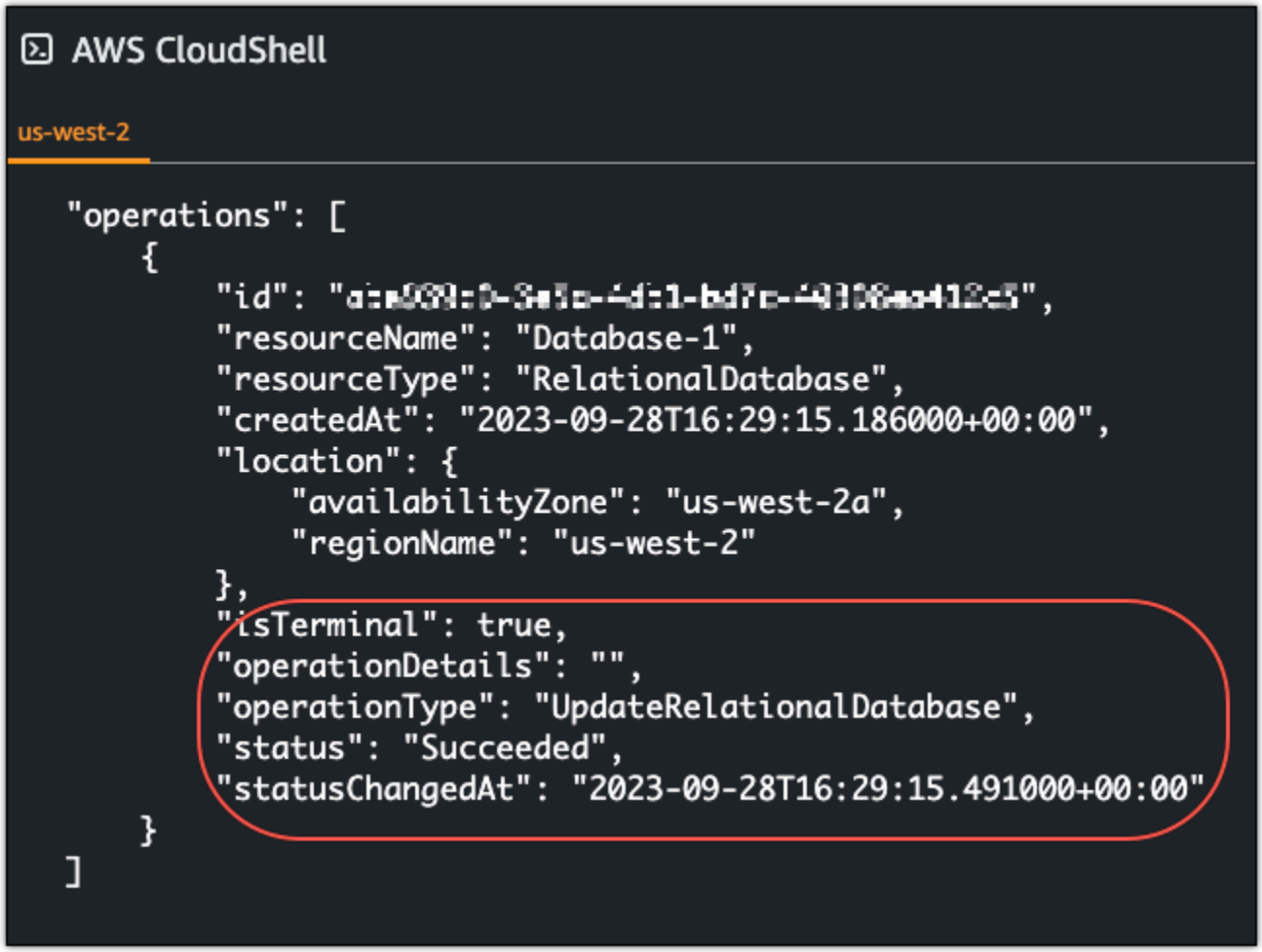
- 터미널, 명령 프롬프트, CloudShell 창에서 다음 명령을 입력합니다.

  ```
  aws lightsail update-relational-database --region Region --relational-database-
  name DatabaseName --disable-backup-retention --apply-immediately
  ```

  명령의 `--disable-backup-retention` 값은 지정된 데이터베이스에 대한 특정 시점 백업을 끕니다. 명령에서 다음과 같이 바꿉니다.

  - *DatabaseName*을 데이터베이스 이름으로 바꿉니다.
  - 데이터베이스 AWS 리전 의가 있는 *##*입니다.

상태가 `Succeeded`인 작업 응답이 표시됩니다. 데이터베이스 상태가 업데이트되는 동안 잠시 동안 수정 중으로 변경됩니다. 데이터베이스 상태가 사용 가능으로 다시 변경되면 다음 예제와 같이 특정 시점 복원 옵션이 비활성화됩니다.



> **Note**
>
> 특정 시점 백업을 활성화하려면 앞에서 나열한 것과 동일한 명령을 실행하되 대신 `--enable-backup-retention` 파라미터를 사용합니다.

스냅샷으로 Lightsail 데이터베이스 백업

Amazon Lightsail에서 관리형 데이터베이스의 스냅샷을 생성할 수 있습니다. 스냅샷은 문제가 발생할 경우 복원하는 데 사용할 수 있는 데이터베이스의 복사본입니다. 스냅샷을 사용하여 고가용성 또는 표준 플랜과 같은 다른 플랜을 사용하는 새 데이터베이스를 생성할 수도 있습니다.

표준 데이터베이스의 스냅샷을 생성할 때 크기에 따라 몇 초에서 몇 분 동안 데이터베이스가 사용할 수 없게 됩니다. 스냅샷은 예비 데이터베이스를 사용하여 생성되므로 고가용성 데이터베이스는 스냅샷 작업의 영향을 받지 않습니다.

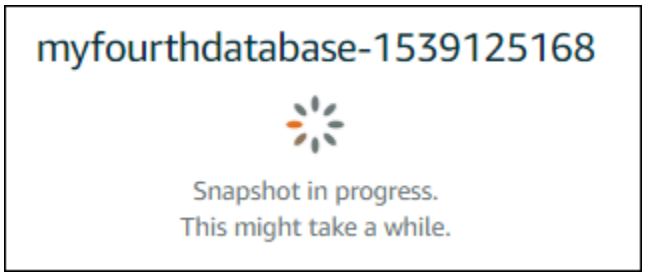
데이터베이스의 스냅샷을 생성하려면

1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 데이터베이스를 선택합니다.
3. 스냅샷을 생성하려는 데이터베이스의 이름을 선택합니다.
4. Snapshots & restore(스냅샷 및 복원) 탭을 선택합니다.
5. 페이지의 수동 스냅샷 섹션에서 스냅샷 생성을 선택한 다음 스냅샷 이름을 입력합니다.

   리소스 이름:

   - Lightsail 계정 AWS 리전 의 각 내에서 고유해야 합니다.
   - 2~255자의 문자로 구성되어야 합니다.
   - 영숫자 문자 또는 숫자로 시작하고 끝나야 합니다.
   - 영숫자 문자, 숫자, 마침표, 대시, 밑줄이 포함될 수 있습니다.
6. 생성(Create)을 선택합니다.

   스냅샷 생성 프로세스가 시작되고 스냅샷 진행 중 상태가 표시됩니다.



   스냅샷 생성 프로세스가 완료되면 새 스냅샷이 최근 스냅샷 섹션에 나열됩니다. Lightsail 홈 페이지의 스냅샷 탭에서 계정에 대한 모든 스냅샷을 볼 수도 있습니다.

다음 단계

스냅샷이 준비 완료되면 스냅샷에서 새 데이터베이스를 생성할 수 있습니다. 이는 원본 데이터베이스의 복제본입니다. 자세한 내용은 스냅샷에서 데이터베이스 생성을 참조하세요.

주제

- Lightsail의 특정 시점 백업에서 데이터베이스 복원
- Lightsail에서 스냅샷에서 관리 데이터베이스 생성

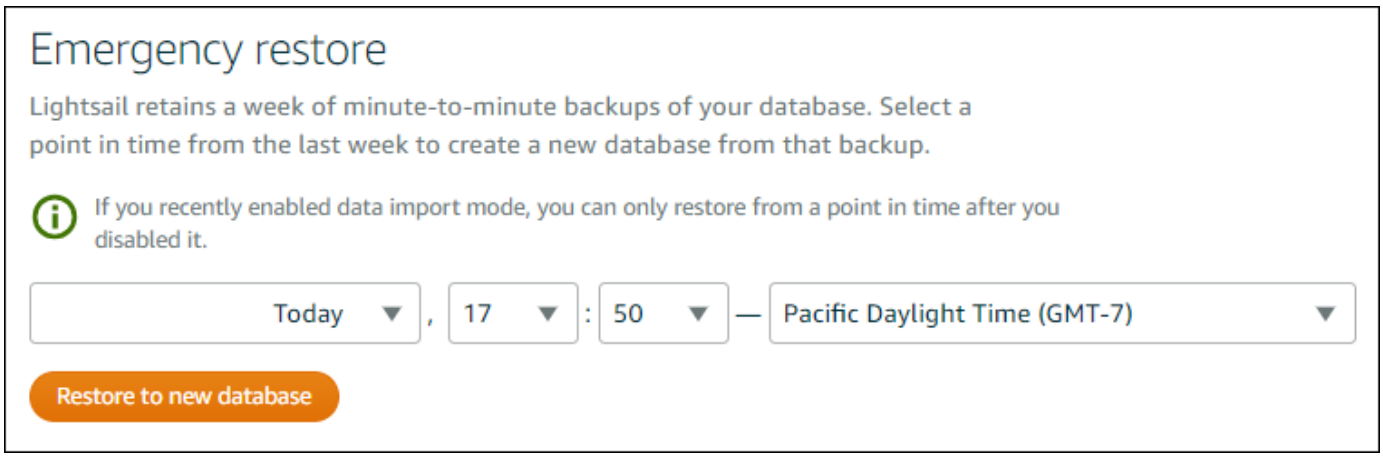
Lightsail의 특정 시점 백업에서 데이터베이스 복원

Amazon Lightsail의 특정 시점 백업을 사용하여 새 관리형 데이터베이스를 생성할 수 있습니다. 데이터베이스의 특정 시점 백업은 이전 7일 동안 5분 단위로 사용할 수 있습니다. 이 기능을 사용하면 실패한 데이터베이스를 지난 주의 특정 날짜 및 시간으로 복원할 수 있습니다.

스냅샷에서 새 데이터베이스를 생성할 수도 있습니다. 자세한 내용은 Amazon Lightsail의 스냅샷에서 데이터베이스 생성을 참조하세요.

특정 시점 백업에서 데이터베이스를 생성하려면

1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 데이터베이스를 선택합니다.
3. 플랜을 변경하려는 데이터베이스의 이름을 선택합니다.
4. Snapshots and restore(스냅샷 및 복원) 탭을 선택합니다.
5. Emergency restore(비상 복원) 섹션에서 새 데이터베이스에 사용할 백업 날짜 및 시간을 선택합니다.



6. Restore to new database(새 데이터베이스로 복원)를 선택합니다.

7. Create a new database(새 데이터베이스 생성) 페이지에서 영역 변경을 선택하여 다른 가용 영역을 선택합니다. 그러면 새 데이터베이스가 이전에 선택한 스냅샷과 동일한 AWS 리전에 생성됩니다.

8. 새 데이터베이스 플랜을 선택합니다.

   고가용성 또는 표준 데이터베이스 플랜을 선택합니다. 고가용성 플랜으로 생성된 데이터베이스에는 기본 데이터베이스가 있으며 장애 조치 지원을 위해 다른 가용 영역에 보조 예비 데이터베이스를 둡니다. 자세한 내용은 고가용성 데이터베이스를 참조하세요.

   > **Note**
   >
   > 원본 데이터베이스 플랜보다 작은 데이터베이스 플랜을 선택할 수는 없습니다.

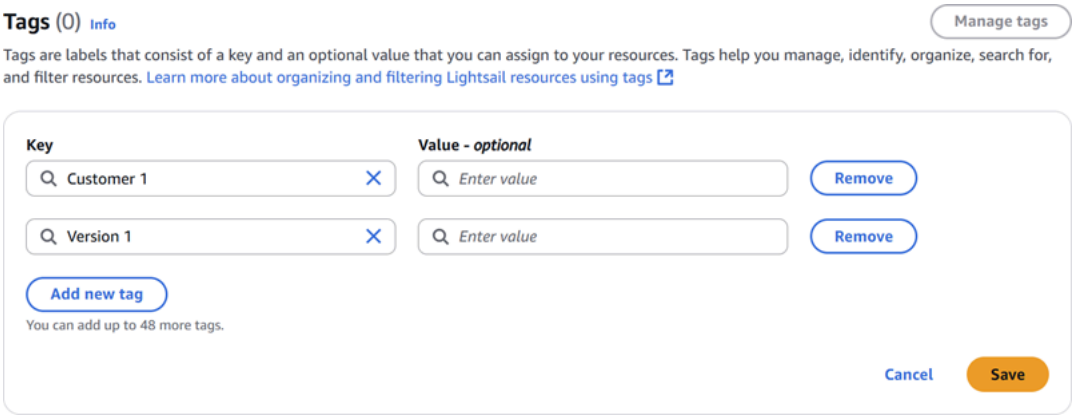
9. 데이터베이스의 이름을 입력합니다.

   리소스 이름:

   - Lightsail 계정 AWS 리전 의 각 내에서 고유해야 합니다.
   - 2~255자의 문자로 구성되어야 합니다.
   - 영숫자 문자 또는 숫자로 시작하고 끝나야 합니다.
   - 영숫자 문자, 숫자, 마침표, 대시, 밑줄이 포함될 수 있습니다.

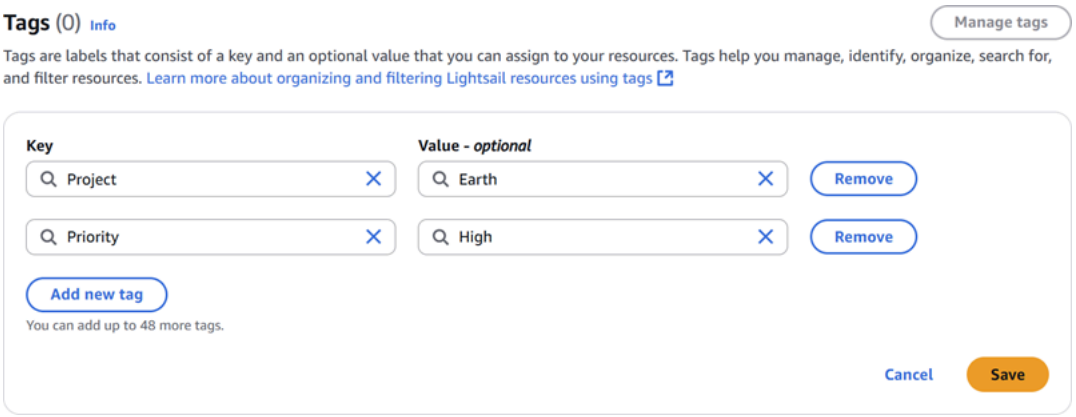
10. 다음 옵션 중 하나를 선택하여 데이터베이스에 태그를 추가합니다.

    - 키 전용 태그를 추가하거나 태그를 관리(이미 태그를 추가한 경우)합니다. 새 태그를 태그 키 텍스트 상자에 입력하고 Enter를 누릅니다. 태그를 추가하려면 태그 입력이 완료될 때 저장을 선택하고, 태그를 추가하지 않으려면 취소를 선택합니다.

    - 키-값 태그를 생성한 다음 키를 키 텍스트 상자에 입력하고, 값을 값 텍스트 상자에 입력합니다. 태그 입력이 완료되면 저장을 선택하고, 태그를 추가하지 않으려면 취소를 선택합니다.

키-값 태그는 저장 전에 한 번에 하나씩만 추가할 수 있습니다. 둘 이상의 키-값 태그를 추가하려면 이전 단계를 반복하세요.



> **Note**
>
> 키 전용 태그 및 키-값 태그에 대한 자세한 내용은 태그를 참조하세요.

11. 데이터베이스 생성을 선택합니다.

    몇 분 내에 새 데이터베이스 플랜 또는 번들로 새 Lightsail 데이터베이스가 준비 완료됩니다.

다음 단계

새 데이터베이스가 준비 및 실행된 후 다음 작업을 완료하세요.

- 더 이상 필요하지 않은 경우 원본 데이터베이스를 삭제합니다. 자세한 내용은 데이터베이스 삭제를 참조하세요.
- 특정 시점 백업에서 생성된 데이터베이스는 Lightsail에서 생성된 강력한 암호를 사용하도록 구성됩니다. 자세한 내용은 데이터베이스 암호 관리를 참조하세요.

Lightsail에서 스냅샷에서 관리 데이터베이스 생성

원본 데이터베이스에 문제가 발생하는 경우 Amazon Lightsail의 스냅샷에서 새 관리형 데이터베이스를 생성할 수 있습니다. 고가용성 또는 표준 플랜과 같이 다른 플랜으로 데이터베이스를 변경할 수도 있습니다. 원본 데이터베이스의 특정 시점 백업에서 새 데이터베이스를 생성할 수도 있습니다. 자세한 내용은 Amazon Lightsail의 특정 시점 백업에서 데이터베이스 생성을 참조하세요.

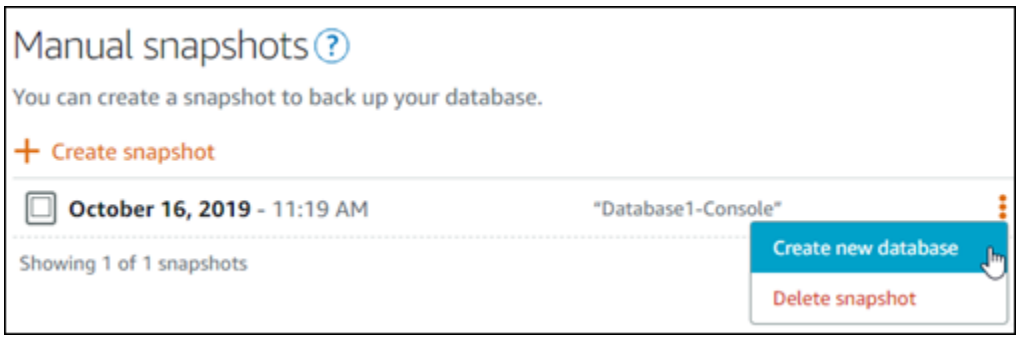
중복 데이터베이스를 생성한 경우 원본 데이터베이스와 다르거나 더 큰 플랜을 선택할 수 있습니다. 그러나 원본 데이터베이스보다 더 작은 플랜은 선택할 수 없습니다.

> **Note**
>
> 고가용성 플랜으로 생성된 데이터베이스에는 기본 데이터베이스가 있으며 장애 조치 지원을 위해 다른 가용 영역에 보조 예비 데이터베이스를 둡니다. 자세한 내용은 고가용성 데이터베이스를 참조하세요.

스냅샷에서 데이터베이스를 생성하려면

1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 데이터베이스를 선택합니다.
3. 스냅샷에서 새 데이터베이스를 생성하여 복제하려는 데이터베이스의 이름을 선택합니다.
4. Snapshots & restore(스냅샷 및 복원) 탭을 선택합니다.
5. 페이지의 수동 스냅샷 섹션에서 새 데이터베이스를 생성할 스냅샷 옆에 있는 작업 메뉴 아이콘(⋮)을 선택한 후 새 데이터베이스 생성을 선택합니다.

   > **Note**
   >
   > 작업할 데이터베이스의 스냅샷이 필요합니다. 아직 스냅샷을 생성하지 않은 경우 데이터베이스의 스냅샷 생성을 참조하세요.



6. Create new database(새 데이터베이스 생성)를 선택합니다.
7. Create a new database(새 데이터베이스 생성) 페이지에서 영역 변경을 선택하여 다른 가용 영역을 선택합니다. 새 데이터베이스가 이전에 선택한 스냅샷과 동일한 AWS 리전에 생성됩니다.
8. 새 데이터베이스 플랜을 선택합니다.

고가용성 또는 표준 데이터베이스 플랜을 선택합니다. 고가용성 플랜으로 생성된 데이터베이스에는 기본 데이터베이스가 있으며 장애 조치 지원을 위해 다른 가용 영역에 보조 예비 데이터베이스를 둡니다. 자세한 내용은 고가용성 데이터베이스를 참조하세요.

> **Note**
>
> 스냅샷을 생성하는 데 사용된 원본 데이터베이스 플랜보다 작은 데이터베이스 플랜을 선택할 수는 없습니다.

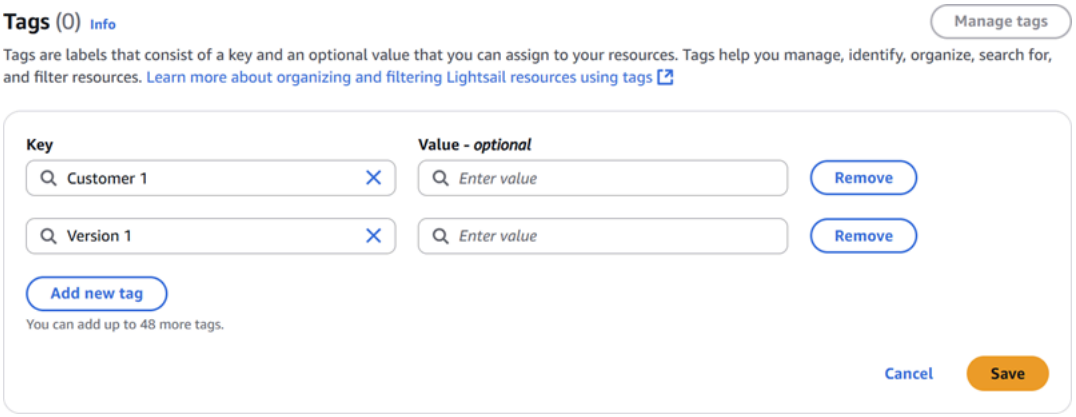
9. 데이터베이스의 이름을 입력합니다.

   리소스 이름:

   - Lightsail 계정의 각 AWS 리전 내에서 고유해야 합니다.
   - 2~255자의 문자로 구성되어야 합니다.
   - 영숫자 문자 또는 숫자로 시작하고 끝나야 합니다.
   - 영숫자 문자, 숫자, 마침표, 대시, 밑줄이 포함될 수 있습니다.

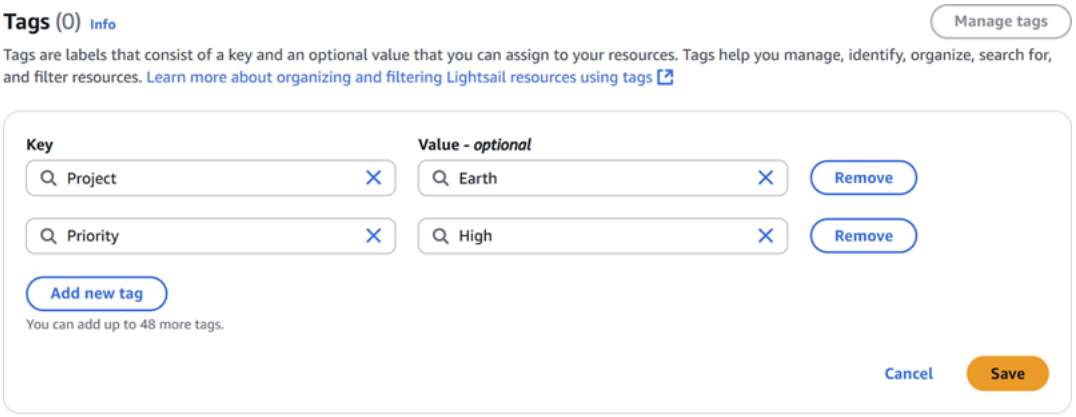
10. 다음 옵션 중 하나를 선택하여 데이터베이스에 태그를 추가합니다.

    - 키 전용 태그를 추가하거나 태그를 관리(이미 태그를 추가한 경우)합니다. 새 태그를 태그 키 텍스트 상자에 입력하고 Enter를 누릅니다. 태그를 추가하려면 태그 입력이 완료될 때 저장을 선택하고, 태그를 추가하지 않으려면 취소를 선택합니다.



    - 키-값 태그를 생성한 다음 키를 키 텍스트 상자에 입력하고, 값을 값 텍스트 상자에 입력합니다. 태그 입력이 완료되면 저장을 선택하고, 태그를 추가하지 않으려면 취소를 선택합니다.

      키-값 태그는 저장 전에 한 번에 하나씩만 추가할 수 있습니다. 둘 이상의 키-값 태그를 추가하려면 이전 단계를 반복하세요.



> **Note**
>
> 키 전용 태그 및 키-값 태그에 대한 자세한 내용은 태그를 참조하세요.

11. 데이터베이스 생성을 선택합니다.

    몇 분 내에 새 데이터베이스 플랜 또는 번들로 새 Lightsail 데이터베이스가 준비 완료됩니다.

다음 단계

새 데이터베이스가 준비 및 실행된 후 다음 작업을 완료하세요.

- 새 데이터베이스를 생성하여 기존 데이터베이스를 대체하려고 할 때 사용자의 애플리케이션이 기존 데이터베이스에 종속되는 경우 애플리케이션 종속성을 새 데이터베이스로 업데이트해야 합니다.
- 더 이상 필요하지 않은 경우 원본 데이터베이스를 삭제합니다. 자세한 내용은 데이터베이스 삭제를 참조하세요.
- 스냅샷에서 생성된 데이터베이스는 Lightsail에서 생성된 강력한 암호를 사용하도록 구성됩니다. 자세한 내용은 데이터베이스 암호 관리를 참조하세요.

Lightsail 데이터베이스에 대한 안전한 앱 연결을 위한 SSL/TLS 인증서 다운로드

애플리케이션에서 SSL(Secure Socket Layer) 또는 TLS(전송 계층 보안)를 사용하여 MySQL 또는 PostgreSQL을 실행하는 Amazon Lightsail의 관리형 데이터베이스에 대한 연결을 암호화할 수 있습니다. 각 DB 엔진에는 SSL/TLS를 구현하기 위한 고유한 프로세스가 있습니다. 자세한 내용은 SSL을 사

용하여 MySQL 데이터베이스에 연결 또는 SSL을 사용하여 PostgreSQL 데이터베이스에 연결을 참조하세요.

> **Note**
>
> 다운로드할 수 있는 인증서는 Amazon Relational Database Service(RDS)로 표시되어 있지만 Lightsail의 관리형 데이터베이스에서도 작동합니다.

모든 AWS 리전에 대한 인증서 번들

모든 AWS 리전에 대한 중간 인증서와 루트 인증서가 모두 포함된 인증서 번들을 가져오거나 애플리케이션이 Microsoft Windows에 있고 PKCS7 파일이 필요한 경우 Amazon Relational Database Service 사용 설명서의 모든 AWS 리전에 대한 인증서 번들을 참조하세요.

이 루트 인증서는 신뢰할 수 있는 루트 개체이므로 대부분의 경우에 작동합니다. 그러나 애플리케이션에서 인증서 체인을 수락하지 않는 경우 실패할 수 있습니다. 애플리케이션에서 인증서 체인을 허용하지 않는 경우 이 문서의 다음 섹션을 계속 진행합니다.

특정 AWS 리전에 대한 인증서 번들

특정에 대한 중간 인증서와 루트 인증서가 모두 포함된 인증서 번들을 가져오려면 Amazon Relational Database Service 사용 설명서의 특정 AWS 리전에 대한 인증서 번들을 AWS 리전참조하세요.

Lightsail 데이터베이스의 CA 인증서 버전 업데이트

Amazon Lightsail에서는 SSL/TLS를 사용하여 관리형 데이터베이스에 연결할 수 있도록 새로운 인증기관(CA)를 게시하였습니다. 이 안내서에서는 새 CA 인증서로 업그레이드하는 방법을 설명합니다. update-relational-database API 작업을 사용해야만 인증서를 업그레이드할 수 있습니다. 새 인증서를 `rds-ca-rsa2048-g1`, `rds-ca-rsa4096-g1` 및 `rds-ca-ecc384-g1`(이)라고 합니다. 이전 인증서를 `rds-ca-2019`(이)라고 합니다. CA 인증서는 AWS 보안 모범 사례로 제공됩니다. 관리형 데이터베이스의 CA 인증서, 지원되는 AWS 리전에 대한 자세한 내용은 관리형 데이터베이스의 SSL 인증서 다운로드를 참조하세요

이전 CA 인증서(`rds-ca-2019`)는 2024년 8월 22일에 만료됩니다. 따라서 새 인증서를 사용하도록 관리형 데이터베이스를 수정하기 위해 가능한 빨리 이 안내서의 단계를 완료하는 것이 좋습니다. 애플리케이션이 SSL/TLS를 사용하여 Lightsail 관리형 데이터베이스에 연결되지 않는 경우 아무 작업도 필요하지 않습니다. 이러한 단계가 완료되지 않으면 2024년 8월 22일 이후 애플리케이션은 SSL/TLS를 사용하여 관리형 데이터베이스에 연결할 수 없게 됩니다.

2024년 1월 26일 이후에 생성된 모든 새 관리형 데이터베이스에서는 기본적으로 `rds-ca-rsa2048-g1` 인증서를 사용합니다. 이전 인증서(`rds-ca-2019`)를 사용하도록 새 관리형 데이터베이스를 일시적으로 수정하려는 경우 AWS Command Line Interface (AWS CLI)을 사용하여 수정할 수 있습니다. 2024년 1월 26일 이전에 생성된 모든 관리형 데이터베이스에서는 `rds-ca-rsa2048-g1`, `rds-ca-rsa4096-g1` 및 `rds-ca-ecc384-g1` 인증서로 업데이트하기 전까지 `rds-ca-2019` 인증서를 사용합니다.

> **Note**
>
> 프로덕션 환경에서 사용하기 전에 개발 또는 스테이징 환경에서 이 안내서의 단계를 테스트하세요.

사전 조건

- 새 SSL/TLS 인증서를 사용하려면 다음 단계를 완료하기 전에 데이터베이스 클라이언트 애플리케이션을 업데이트해야 합니다.

  새 SSL/TLS 인증서를 위해 애플리케이션을 업데이트하는 방법은 애플리케이션에 따라 다릅니다. 애플리케이션 개발자와 함께 애플리케이션의 SSL/TLS 인증서를 업데이트하세요. 새로운 SSL/TLS 인증서에 대해 애플리케이션을 업데이트하는 방법에 대한 자세한 내용은 Amazon Relational Database Service 사용 설명서의 새 SSL/TLS 인증서를 사용하여 MySQL DB 인스턴스에 연결할 애플리케이션 업데이트 또는 새 SSL/TLS 인증서를 사용하여 PostgreSQL DB 인스턴스에 연결할 애플리케이션 업데이트를 참조하세요.
- 이 가이드에서는 AWS CloudShell 를 사용하여 업그레이드를 수행합니다. CloudShell은 브라우저 기반의 사전 인증된 셸로, Lightsail 콘솔에서 직접 실행할 수 있습니다. CloudShell을 사용하면 Bash, PowerShell 또는 Z 쉘과 같은 원하는 쉘을 사용하여 AWS Command Line Interface (AWS CLI) 명령을 실행할 수 있습니다. 명령줄 도구를 다운로드하거나 설치할 필요 없이 이 작업을 수행할 수 있습니다. CloudShell을 설정하고 사용하는 방법은 Lightsail의AWS CloudShell 섹션을 참조하세요.

관리형 데이터베이스의 활성 CA 인증서 식별

다음 단계를 완료하여 Lightsail 데이터베이스 인스턴스의 활성 CA 인증서를 식별합니다.

1. 터미널, AWS CloudShell 또는 명령 프롬프트 창을 엽니다.
2. 다음 명령을 입력하여 관리형 데이터베이스에서 활성 CA 인증서를 식별합니다.

```
aws lightsail get-relational-database --relational-database-name DatabaseName --
region DatabaseRegion | grep "caCertificateIdentifier"
```

명령에서 *DatabaseName*을 수정하려는 데이터베이스의 이름으로 바꾸고 *DatabaseRegion*을 데이터베이스 인스턴스가 있는 AWS 리전 으로 바꿉니다.

예제

```
aws lightsail get-relational-database --relational-database-name Database-1 --
region us-east-1 | grep "caCertificateIdentifier"
```

명령은 데이터베이스에 대한 활성 CA 인증서의 ID를 반환합니다.

예제

```
"caCertificateIdentifier": "rds-ca-rsa2048-g1"
```

새 CA 인증서를 사용하도록 관리형 데이터베이스 수정

새 CA 인증서(`rds-ca-rsa2048-g1`, `rds-ca-rsa4096-g1` 및 `rds-ca-ecc384-g1`)를 사용하도록 Lightsail에서 관리형 데이터베이스를 수정하려면 다음 단계를 수행하세요.

> **Important**
>
> 데이터베이스에서 CA 인증서를 업데이트하기 전에 CA 인증서를 사용하는 클라이언트 애플리케이션을 업데이트합니다.

1. 터미널, AWS CloudShell 또는 명령 프롬프트 창을 엽니다.
2. 다음 명령을 입력하여 관리형 데이터베이스에서 새 인증서를 사용합니다.

```
aws lightsail update-relational-database --relational-database-name DatabaseName --
region DatabaseRegion --ca-certificate-identifier rds-ca-rsa2048-g1
```

명령에서 *DatabaseName*을 수정하려는 데이터베이스의 이름으로 바꾸고 *DatabaseRegion*을 데이터베이스 인스턴스가 있는 AWS 리전 으로 바꿉니다.

예제

```
aws lightsail update-relational-database --relational-database-name Database-1 --
region us-east-1 --ca-certificate-identifier rds-ca-rsa2048-g1
```

관리형 데이터베이스에서 사용하는 CA 인증서는 데이터베이스의 다음 유지 관리 기간 동안 또는 `--apply-immediately` 파라미터를 명령 끝에 추가하는 경우 즉시 업데이트됩니다.

이전 CA 인증서를 사용하도록 관리형 데이터베이스 수정

이전 CA 인증서(Lightsail)를 사용하도록 `rds-ca-2019`에서 관리형 데이터베이스를 수정하려면 다음 단계를 수행하세요. 새 인증서(`rds-ca-rsa2048-g1`, `rds-ca-rsa4096-g1` 및 `rds-ca-ecc384-g1`) 중 하나에 심각한 문제가 발생하여 이전 인증서를 일시적으로 되돌려야 하는 경우에만 이 작업을 수행하세요.

> **Important**
>
> 데이터베이스에서 CA 인증서를 업데이트하기 전에 CA 인증서를 사용하는 클라이언트 애플리케이션을 업데이트합니다.

1. 터미널, AWS CloudShell 또는 명령 프롬프트 창을 엽니다.
2. 다음 명령을 입력하여 관리형 데이터베이스에서 `rds-ca-2019`를 사용합니다.

```
aws lightsail update-relational-database --relational-database-name DatabaseName --
region DatabaseRegion --ca-certificate-identifier rds-ca-2019
```

명령에서 *DatabaseName*을 수정하려는 데이터베이스의 이름으로 바꾸고 *DatabaseRegion*을 데이터베이스 인스턴스가 있는 AWS 리전 으로 바꿉니다.

예제

```
aws lightsail update-relational-database --relational-database-name Database-1 --
region us-east-1 --ca-certificate-identifier rds-ca-2019
```

관리형 데이터베이스에서 사용하는 CA 인증서는 데이터베이스의 다음 유지 관리 기간 동안 또는 `--apply-immediately` 파라미터를 명령 끝에 추가하는 경우 즉시 업데이트됩니다.

Lightsail 데이터베이스에 대한 유지 관리 및 백업 예약

Amazon Lightsail에서 새 데이터베이스 버전이 지원되는 경우 기존 관리형 데이터베이스를 해당 버전으로 업그레이드할 수 있습니다. 마이너 버전 업그레이드와 메이저 버전 업그레이드라는 두 가지 업그레이드가 있습니다. 현재 Lightsail에서는 마이너 버전 업그레이드만 지원합니다.

마이너 버전 업그레이드 및 기타 데이터베이스 유지 관리 작업은 데이터베이스에 대한 기본 유지 관리 기간 동안 자동으로 수행됩니다. 기본 유지 관리 기간은 각각 8시간 블록 시간 중에서 임의로 선택한 30분 기간입니다 AWS 리전. 이 기간은 임의의 요일에 발생합니다. 데이터베이스 백업은 기본 백업 기간 동안 수행됩니다. 기본 백업 기간은 각각 8시간 블록 시간 중에서 임의로 선택한 30분 기간입니다 AWS 리전. 이 기간도 임의의 요일에 발생합니다.

> **Note**
>
> 각 리전에 대한 기본 유지 관리 기간 시간 블록의 자세한 내용은 Amazon RDS(Amazon Relational Database Service) 설명서의 DB 인스턴스 유지 관리 안내서를 참조하십시오. 각 리전에 대한 기본 백업 기간 시간 블록의 자세한 내용은 Amazon RDS 설명서의 백업 작업 안내서를 참조하십시오.

이 안내서에서는 데이터베이스가 가장 낮은 로드 상태일 때 발생하도록 기본 유지 관리 기간 및 백업 기간을 변경하는 방법을 보여줍니다.

사전 조건

데이터베이스 기본 유지 관리 및 백업 기간을 변경하려면 AWS Command Line Interface (AWS CLI)를 사용해야 합니다.

다음 사전 조건을 완료합니다.

- 설치 AWS CLI - 자세한 내용은 AWS CLI I 설치를 참조하세요.
- 구성 AWS CLI- 자세한 내용은 구성을 참조하세요 AWS CLI.

데이터베이스 유지 관리 기간 변경

유지 관리 작업 또는 백업 작업 중에는 데이터베이스가 사용할 수 없게 될 수 있습니다. 따라서, 기본 유지 관리 기간 또는 백업 기간을 데이터베이스가 가장 낮은 로드 상태인 시간으로 변경하려고 할 수 있습니다.

데이터베이스 유지 관리 기간을 변경하려면

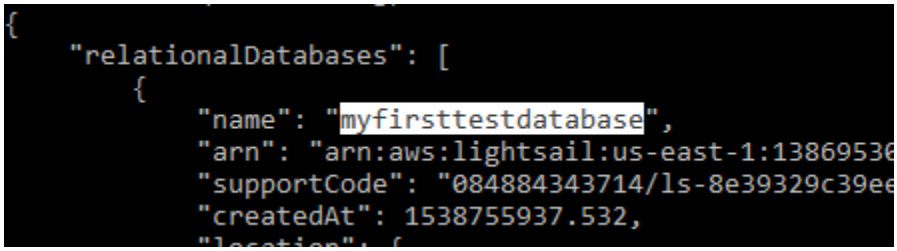
1. 터미널 또는 명령 프롬프트 창을 엽니다.
2. 다음 명령을 입력하여 유지 관리 기간을 변경하려는 데이터베이스의 이름을 가져옵니다.

```
aws lightsail get-relational-databases
```

다음 예와 비슷한 결과가 나타나야 합니다.

> **Note**
>
> 수정하려는 데이터베이스가 나열되지 않은 경우 데이터베이스 AWS 리전 가 있는에 대해 가 AWS CLI 구성되어 있는지 확인합니다. 자세한 내용은 AWS CLI구성을 참조하세요.

3. 수정할 데이터베이스의 이름을 강조 표시하고 다음 단계에서 사용할 수 있도록 Windows를 사용하는 경우 Ctrl+C를, macOS를 사용하는 경우 Cmd+C를 눌러 클립보드에 복사합니다.



4. 변경하는 기본 기간에 따라 다음 명령 중 하나를 입력합니다.

   - 다음 명령을 입력하여 데이터베이스 유지 관리 기간을 변경합니다.

     ```
     aws lightsail update-relational-database --relational-database-name DatabaseName
      --preferred-maintenance-window MaintenanceWindow
     ```

     명령에서 다음과 같이 바꿉니다.

     - *DatabaseName*을 데이터베이스 이름으로 바꿉니다.
     - *MaintenanceWindow*를 새 유지 관리 기간으로 바꿉니다.

       기본 유지 관리 기간 시간을 ddd:hh24:mi-ddd:hh24:mi 형식으로 정의합니다. 또한 UTC(협정 세계시) 형식이어야 하며 30분의 최소 기간으로 정의되어야 합니다. 기본 유지 관리 기간은 기본 백업 기간과 중첩될 수 없습니다.

     예:

     ```
     aws lightsail update-relational-database --relational-database-
     name myproductiondb --preferred-maintenance-window Tue:16:00-Tue:16:30
     ```

   - 다음 명령을 입력하여 데이터베이스 백업 기간을 변경합니다.

     ```
     aws lightsail update-relational-database --relational-database-name DatabaseName
      --preferred-backup-window BackupWindow
     ```

     명령에서 다음과 같이 바꿉니다.

     - *DatabaseName*을 데이터베이스 이름으로 바꿉니다.
     - *BackupWindow*를 새 백업 기간으로 바꿉니다.

       기본 백업 기간 시간을 hh24:mi-hh24:mi 형식으로 정의합니다. 또한 UTC(협정 세계시) 형식이어야 하며 30분의 최소 기간으로 정의되어야 합니다. 기본 백업 기간은 기본 유지 관리 기간과 중첩될 수 없습니다.

예:

```
aws lightsail update-relational-database --relational-database-
name myproductiondb --preferred-backup-window 14:00-14:30
```

다음 예와 비슷한 결과가 나타나야 합니다.

다음 단계

다음은 데이터베이스를 관리하는 데 도움이 되는 몇 가지 안내서입니다.

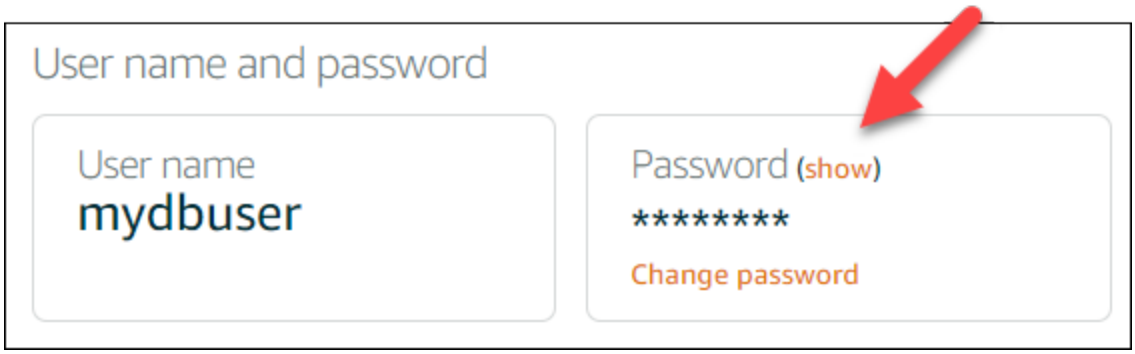
- 데이터베이스에 대해 데이터 가져오기 모드 구성
- 데이터베이스에 대해 퍼블릭 모드 구성
- 데이터베이스 암호 관리
- MySQL 데이터베이스에 연결
- PostgreSQL 데이터베이스에 연결
- MySQL 데이터베이스로 데이터 가져오기
- PostgreSQL 데이터베이스로 데이터 가져오기
- 데이터베이스의 스냅샷 생성

Lightsail 데이터베이스 암호 변경

Amazon Lightsail에서 새 데이터베이스를 생성할 때 Lightsail에서 강력한 암호를 자동으로 생성하도록 하거나 자체 암호를 지정할 수 있습니다. Lightsail 콘솔에서 언제든지 현재 데이터베이스 암호를 보거나 변경할 수 있습니다.

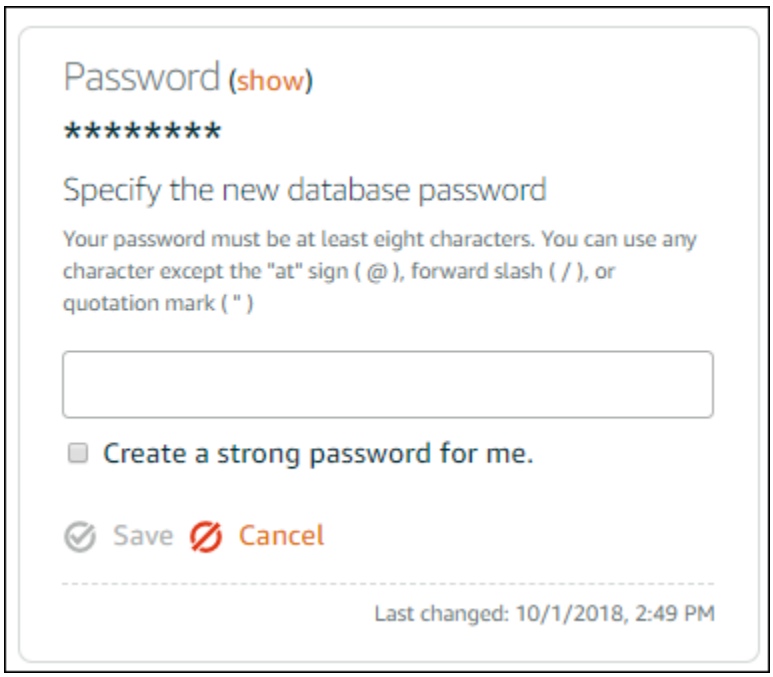
데이터베이스 암호를 관리하려면

1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 데이터베이스를 선택합니다.
3. 암호를 관리하려는 데이터베이스의 이름을 선택합니다.
4. 연결 탭의 User name and passwords(사용자 이름 및 암호) 섹션에서 표시를 선택하여 현재 데이터베이스 암호를 확인합니다.



5. 데이터베이스 암호를 변경하려면 Change password(암호 변경)를 선택합니다.

   Lightsail에서 강력한 암호를 자동으로 생성하도록 선택하거나 텍스트 상자에 자체 암호를 입력할 수 있습니다. 암호에는 "/", """, "@"을 제외하고 인쇄 가능한 모든 ASCII 문자를 사용할 수 있습니다. MySQL 데이터베이스의 경우 암호에 8~41자를 포함해야 합니다. PostgreSQL의 경우 암호에 8~128자를 포함해야 합니다.



6. 완료되면 저장을 선택합니다.

   데이터베이스 암호 변경이 즉시 적용됩니다. 자체 암호를 입력하면 암호가 즉시 저장됩니다. Lightsail이 자동으로 암호를 생성할 경우 몇 초 내에 생성됩니다. 새 암호를 보려면 표시를 선택합니다.

다음 단계

다음은 Lightsail에서 데이터베이스를 관리하는 데 도움이 되는 몇 가지 안내서입니다.

- MySQL 데이터베이스에 연결
- PostgreSQL 데이터베이스에 연결
- 데이터베이스의 스냅샷 생성

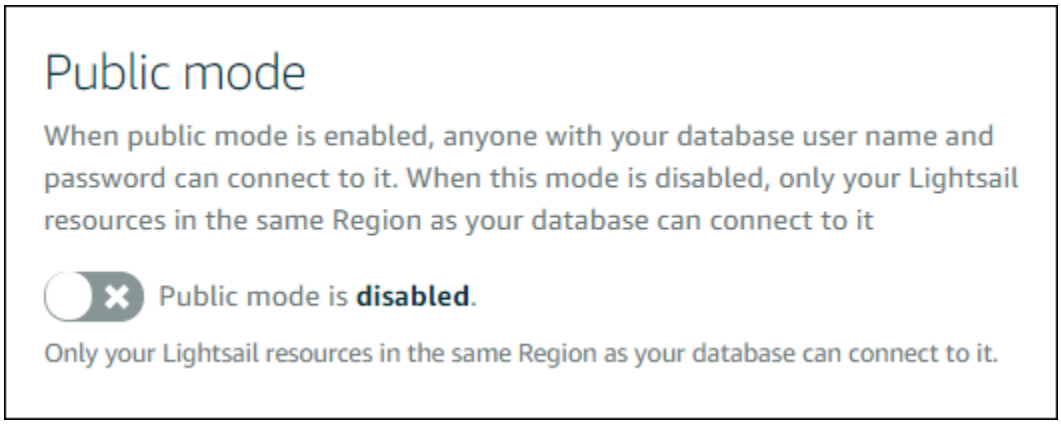
Lightsail 데이터베이스에 대해 퍼블릭 액세스 구성

Amazon Lightsail의 관리형 데이터베이스는 동일한 Lightsail 계정에 있는 Lightsail 리소스(인스턴스, 로드 밸런서 등)에서만 액세스할 수 있습니다. 한 가지 일반적인 시나리오는 일반에 공개되어 있는 웹 애플리케이션을 포함한 Lightsail 인스턴스와 공개적으로 액세스할 수 없는 Lightsail 데이터베이스를 모두 생성하여 연결하는 것입니다.

퍼블릭 모드 기능을 활성화하여 데이터베이스에 공개적으로 액세스할 수 있도록 합니다. 이 방법을 사용하면 데이터베이스 엔드포인트, 포트, 사용자 이름 및 암호를 보유한 사용자는 누구나 데이터베이스에 연결할 수 있습니다. 자세한 내용은 MySQL 데이터베이스에 연결 또는 PostgreSQL 데이터베이스에 연결을 참조하세요.

데이터베이스에 대해 퍼블릭 모드를 구성하려면

1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 데이터베이스를 선택합니다.
3. 퍼블릭 모드를 구성하려는 데이터베이스의 이름을 선택합니다.
4. 네트워킹 탭을 선택합니다.
5. Public mode(퍼블릭 모드) 섹션에서 토글을 사용하여 해당 모드를 켭니다. 마찬가지로 토글을 사용하여 해당 모드를 끕니다.



퍼블릭 접근성 설정은 즉시 적용되지만, 완료하려면 몇 분 정도 걸릴 수 있습니다. 이 시간 동안 데이터베이스의 상태가 수정 중으로 변경됩니다. 퍼블릭 접근성 설정이 적용되면 데이터베이스의 상태가 사용 가능으로 변경됩니다.

다음 단계

다음은 데이터베이스를 관리하는 데 도움이 되는 몇 가지 안내서입니다.

- 데이터베이스에 대해 데이터 가져오기 모드 구성
- 데이터베이스 암호 관리
- MySQL 데이터베이스에 연결
- PostgreSQL 데이터베이스에 연결
- MySQL 데이터베이스로 데이터 가져오기

- PostgreSQL 데이터베이스로 데이터 가져오기
- 데이터베이스의 스냅샷 생성

파라미터 업데이트로 Lightsail 데이터베이스 성능 최적화

데이터베이스 시스템 변수라고도 하는 데이터베이스 파라미터는 Amazon Lightsail에서 관리형 데이터베이스의 기본 속성을 정의합니다. 예를 들어 데이터베이스 파라미터를 정의하여 데이터베이스 연결 수를 제한하거나 다른 파라미터를 정의하여 데이터베이스 버퍼 풀 크기를 제한할 수 있습니다. 이 가이드에서는 관리형 데이터베이스의 파라미터 목록을 가져오는 방법과 AWS Command Line Interface ()를 사용하여 업데이트하는 방법을 보여줍니다AWS CLI.

> **Note**
>
> MySQL 시스템 변수에 대한 자세한 내용은 MySQL 5.6 및 MySQL 5.7 또는 MySQL 8.0 설명서를 참조하세요. PostgreSQL 시스템 변수에 대한 자세한 내용은 PostgreSQL 9.6, PostgreSQL 10, PostgreSQL 11 또는 PostgreSQL 12 설명서를 참조하세요.

사전 조건

- AWS CLI를 아직 설치 및 구성하지 않았다면 설치하고 구성합니다. 자세한 내용은 에서 작동 AWS CLI 하도록 구성을 Lightsail참조하세요.

사용 가능한 데이터베이스 파라미터 목록 가져오기

데이터베이스 파라미터는 데이터베이스 엔진에 따라 다릅니다. 따라서 관리형 데이터베이스에 사용할 수 있는 파라미터 목록을 가져와야 합니다. 그러면 수정할 파라미터와 해당 파라미터가 적용되는 방식을 결정할 수 있습니다.

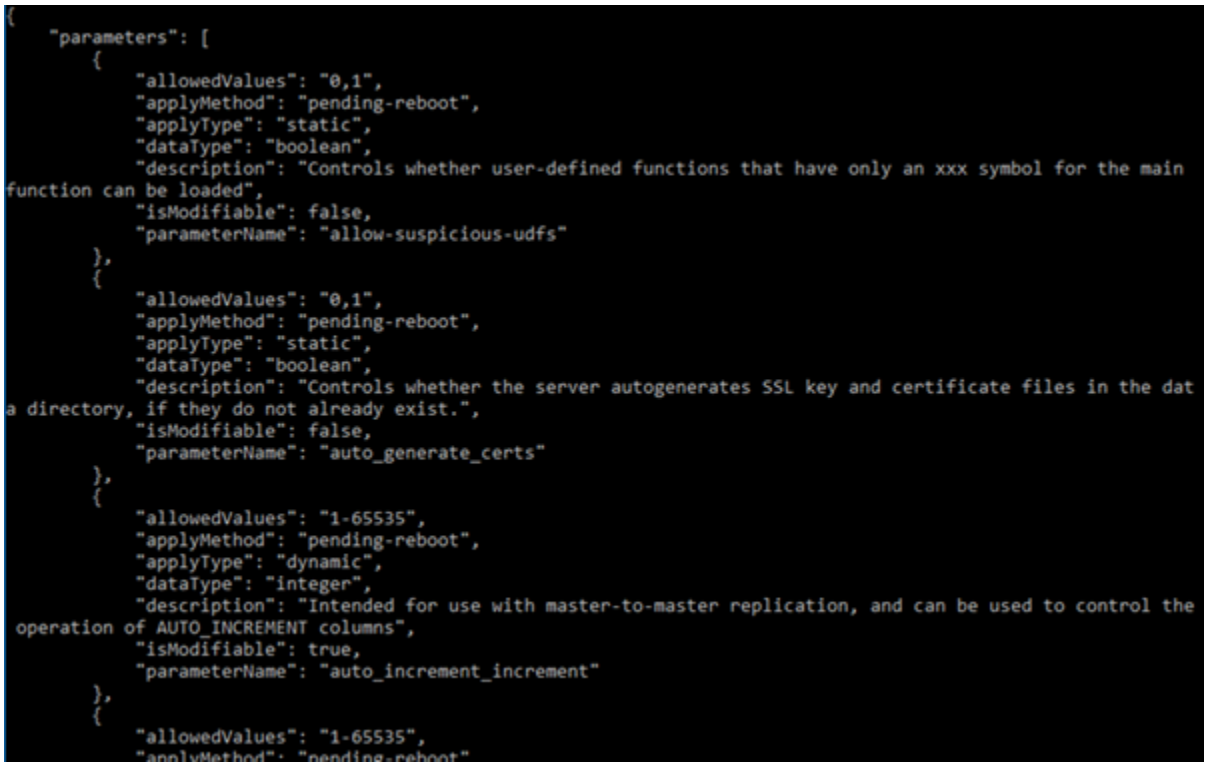
사용 가능한 데이터베이스 파라미터 목록을 가져오려면

1. 터미널 또는 명령 프롬프트 창을 엽니다.
2. 다음 명령을 입력하여 데이터베이스에 대한 파라미터 목록을 가져옵니다.

```
aws lightsail get-relational-database-parameters --relational-database-
name DatabaseName
```

명령에서 *DatabaseName* 을 데이터베이스 이름으로 바꿉니다.

다음 예와 비슷한 결과가 나타나야 합니다.



> **Note**
>
> 파라미터 결과의 페이지 번호가 매겨지면 다음 페이지 토큰 ID가 나열됩니다. 다음 페이지 토큰 ID를 기록해 두고 다음 단계에 표시된 대로 이 토큰 ID를 사용하여 파라미터 결과의 다음 페이지를 확인하세요.

3. 결과의 페이지 번호가 매겨진 경우 다음 명령을 사용하여 추가 파라미터 세트를 확인합니다. 그렇지 않은 경우 다음 단계로 건너뜁니다.

```
aws lightsail get-relational-database-parameters --relational-database-
name DatabaseName --page-token NextPageTokenID
```

명령에서 다음과 같이 바꿉니다.

- *DatabaseName*을 데이터베이스 이름으로 바꿉니다.
- *NextPageTokenID*를 다음 페이지 토큰 ID로 바꿉니다.

각 데이터베이스 파라미터에 대해 다음 정보가 결과에 표시됩니다.

- 허용된 값 - 파라미터의 유효한 값 범위를 지정합니다.
- Apply method(적용 방법) - 파라미터 변경이 적용되는 시기를 지정합니다. 허용 가능한 옵션은 `immediate` 또는 `pending-reboot`입니다. 적용 방법을 정의하는 방법에 대한 자세한 내용은 적용 유형을 참조하세요.
- 적용 유형 - 엔진별 제출 유형을 지정합니다. `dynamic`이 나열되면 파라미터를 `immediate` 적용 방법을 사용하여 적용할 수 있으며, 그 경우 데이터베이스에 새 파라미터 값이 즉시 사용됩니다. `static`이 나열되면 파라미터를 `pending-reboot` 적용 방법으로만 적용할 수 있으며, 그 경우 데이터베이스를 다시 시작한 후에야 새 파라미터가 사용됩니다.
- 데이터 형식 - 파라미터의 유효한 데이터 형식을 지정합니다.
- 설명 - 파라미터에 대한 설명을 제공합니다.
- 수정 가능 - 파라미터를 수정할 수 있는지 여부를 나타내는 부울 값입니다. `true`가 나열되면 파라미터를 수정할 수 있습니다.
- 파라미터 이름 - 파라미터 이름을 지정합니다. 이 값은 `update relational database` 작업 및 `parameter name` 파라미터와 함께 사용합니다.

4. 변경할 파라미터를 찾은 다음 파라미터 이름, 허용되는 값 및 적용 방법을 기록합니다. 파라미터 이름을 잘못 입력하지 않도록 클립보드에 복사해 두는 것이 좋습니다. 이렇게 하려면 파라미터 이름을 강조 표시하고 Windows를 사용하고 있는 경우 Ctrl+C를, macOS를 사용하고 있는 경우 Cmd+C를 눌러 클립보드에 복사합니다. 그런 다음 Ctrl+V 또는 Cmd+V를 적절하게 눌러 붙여넣습니다.

   수정할 파라미터의 이름을 확인한 후에 이 가이드의 다음 섹션을 계속 진행하여 파라미터를 원하는 값으로 변경합니다.

데이터베이스 파라미터 업데이트

변경할 파라미터의 이름을 확인한 후에는 다음 단계를 수행하여 Lightsail에서 관리형 데이터베이스의 파라미터를 수정합니다.

데이터베이스 파라미터를 업데이트하려면

- 터미널 또는 명령 프롬프트 창에 다음 명령을 입력하여 관리형 데이터베이스의 파라미터를 업데이트합니다.

```
aws lightsail update-relational-database-parameters
 --relational-database-name DatabaseName --parameters
 "parameterName=ParameterName,parameterValue=NewParameterValue,applyMethod=ApplyMethod"
```

명령에서 다음과 같이 바꿉니다.

- *DatabaseName*을 데이터베이스 이름으로 바꿉니다.
- *ParameterName*을 수정하려는 파라미터의 이름으로 바꿉니다.
- *NewParameterValue*를 파라미터의 새 값으로 바꿉니다.
- *ApplyMethod*를 파라미터의 적용 방법으로 바꿉니다.

  파라미터의 적용 유형이 dynamic인 경우 파라미터를 immediate 적용 방법을 사용하여 적용할 수 있으며, 그 경우 데이터베이스에 새 파라미터 값이 즉시 사용됩니다. 그러나 파라미터 적용 유형이 static인 경우에는 파라미터를 pending-reboot 적용 방법으로만 적용할 수 있으며, 그 경우 데이터베이스를 다시 시작한 후에야 새 파라미터가 사용됩니다.

다음 예와 비슷한 결과가 나타나야 합니다.


```
{
    "operations": [
        {
            "id": "2c650987-11e8-463f-94d5-0c15aacaf12b",
            "resourceName": "myfirsttestdatabase",
            "resourceType": "RelationalDatabase",
            "createdAt": 1539204831.214,
            "location": {
                "availabilityZone": "us-east-1a",
                "regionName": "us-east-1"
            },
            "isTerminal": true,
            "operationType": "UpdateRelationalDatabaseParameters",
            "status": "Succeeded",
            "statusChangedAt": 1539204831.214
        }
    ]
}
```


데이터베이스 파라미터는 사용된 적용 방법에 따라 업데이트됩니다.

Lightsail 데이터베이스의 메이저 버전 업그레이드

Amazon Lightsail에서 새 데이터베이스 엔진 버전을 지원하는 경우, 데이터베이스를 새 버전으로 업그레이드할 수 있습니다. Lightsail은 MySQL과 PostgreSQL이라는 두 가지 데이터베이스 블루프린트를 제공합니다. 이 안내서에서는 MySQL 또는 PostgreSQL 데이터베이스 인스턴스의 메이저 버전을 업그레이드하는 방법을 설명합니다. update-relational-database API 작업을 사용해야만 데이터베이스 메이저 버전을 업그레이드할 수 있습니다.

AWS CloudShell 를 사용하여 업그레이드를 수행합니다. CloudShell은 브라우저 기반의 사전 인증된 셸로, Lightsail 콘솔에서 직접 실행할 수 있습니다. CloudShell을 사용하면 Bash, PowerShell 또는 Z 쉘과 같은 원하는 쉘을 사용하여 AWS Command Line Interface (AWS CLI) 명령을 실행할 수 있습니다. 명령줄 도구를 다운로드하거나 설치할 필요 없이 이 작업을 수행할 수 있습니다. CloudShell을 설정하고 사용하는 방법은 Lightsail의AWS CloudShell 섹션을 참조하세요.

변경 사항 이해

메이저 버전 업그레이드는 이전 버전과 호환되지 않는 여러 가지 문제를 일으킬 수 있습니다. 이러한 비호환성으로 인해 업그레이드하는 동안 문제가 발생할 수 있습니다. 업그레이드에 성공하려면 데이터베이스를 준비해야 할 수 있습니다. 데이터베이스의 메이저 버전 업그레이드에 대한 자세한 내용은 MySQL 및 PostgreSQL 웹 사이트의 다음 주제를 참조하세요.

- 업그레이드를 위한 설치 준비
- MySQL 업그레이드 검사기 유틸리티
- PostgreSQL 클러스터 업그레이드

사전 조건

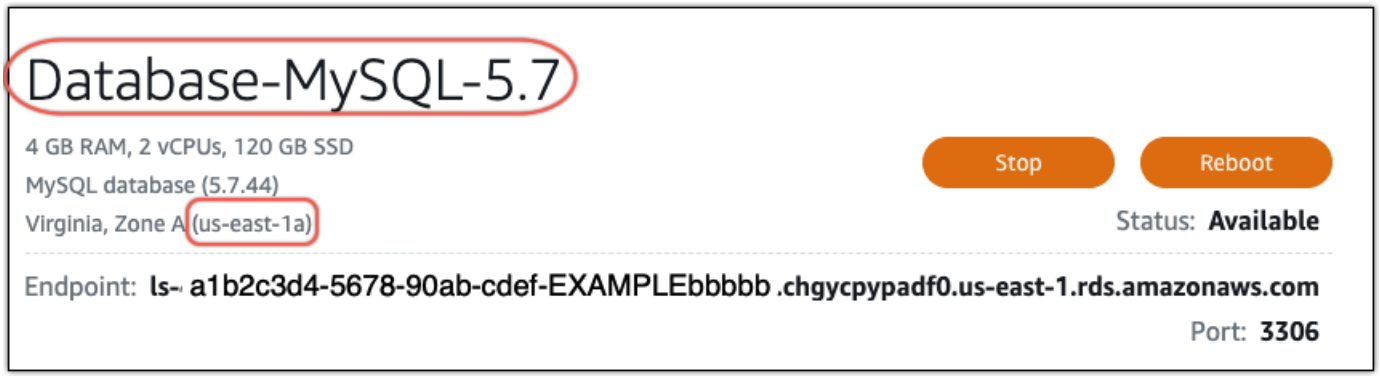
1. 애플리케이션이 데이터베이스의 두 메이저 버전을 모두 지원하는지 확인합니다.
2. 변경하기 전에 데이터베이스 인스턴스의 스냅샷을 생성하는 것이 좋습니다. 자세한 내용은 Lightsail 데이터베이스의 스냅샷 생성을 참조하세요.
3. (선택 사항) 방금 생성한 스냅샷에서 새 데이터베이스 인스턴스를 생성합니다. 데이터베이스 업데이트에는 가동 중지 시간이 필요하므로 현재 활성 상태인 데이터베이스를 업그레이드하기 전에 새 데이터베이스에서 업그레이드를 테스트할 수 있습니다. 데이터베이스 복사에 대한 자세한 내용은 Lightsail 데이터베이스 스냅샷 생성을 참조하세요.

데이터베이스 메이저 버전 업데이트

Lightsail은 MySQL 및 PostgreSQL 데이터베이스 인스턴스에 대한 메이저 버전 업그레이드를 지원합니다. MySQL 데이터베이스는 다음 절차의 예로 사용됩니다. 하지만 프로세스와 명령은 PostgreSQL 데이터베이스에서 동일합니다.

Lightsail 데이터베이스의 데이터베이스 메이저 버전을 업그레이드하려면 다음 절차를 완료하세요.

1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 데이터베이스를 선택합니다.
3. 업그레이드하려는 데이터베이스 인스턴스의 이름과 AWS 리전 를 기록해 둡니다.



4. Lightsail 콘솔 왼쪽 하단에서 CloudShell을 선택합니다. 동일한 브라우저 탭에서 CloudShell 터미널이 열립니다. 명령 프롬프트가 표시되면 셸이 상호 작용할 준비가 된 것입니다.
5. CloudShell 프롬프트에 다음 명령을 입력하여 사용 가능한 데이터베이스 블루프린트 ID 목록을 가져옵니다.

```
aws lightsail get-relational-database-blueprints
```

6. 업그레이드하려는 메이저 버전의 블루프린트 ID에 대한 참고 사항입니다. 예를 들어 `mysql_8_0`입니다.

7. 다음 명령을 입력하여 데이터베이스의 메이저 버전으로 업그레이드합니다. 업그레이드는 데이터베이스의 다음 유지 관리 기간 동안 수행됩니다. 명령에서 *DatabaseName*을 데이터베이스 이름으로 바꾸고, *blueprintId*를 업그레이드하려는 메이저 버전의 블루프린트 ID로 바꾸고, *DatabaseRegion*을 데이터베이스가 AWS 리전 있는 로 바꿉니다.

```
aws lightsail update-relational-database \
 --relational-database-name DatabaseName \
 --relational-database-blueprint-id blueprintId \
 --region DatabaseRegion
```

(선택 사항) 업그레이드를 즉시 적용하려면 명령에 `--apply-immediately` 파라미터를 포함합니다. 다음 예제와 유사한 응답이 표시되고 업그레이드가 적용되는 동안 데이터베이스를 사용할 수 없게 됩니다. 자세한 내용을 알아보려면 https://docs.aws.amazon.com/lightsail/2016-11-28/api-reference/API_UpdateRelationalDatabase.html API 레퍼런스의 update-relational-databaseLightsail 섹션을 참조하세요.

```
% aws lightsail update-relational-database \
--relational-database-name "Database-Mysql-5.7" \
--relational-database-blueprint-id "mysql_8_0" \
--apply-immediately \
[--region us-east-1
{
    "operations": [
        {
            "id": "a1b2c3d4-5678-90ab-cdef-EXAMPLEbbbbb",
            "resourceName": "Database-Mysql-5.7",
            "resourceType": "RelationalDatabase",
            "createdAt": 2024-01-01T00:00:00.00000+00:00",
            "location": {
                "availabilityZone": "us-east-1a",
                "regionName": "us-east-1"
            },
            "isTerminal": true,
            "operationDetails": "",
            "operationType": "UpdateRelationalDatabase",
            "status": "Succeeded",
            "statusChangedAt": 2024-01-01T00:00:00.00000+00:00",
        }
```

8. 다음 명령을 입력하여 메이저 버전 업그레이드가 다음 데이터베이스 유지 관리 기간에 예약되어 있는지 확인합니다. 명령에서 *DatabaseName*을 데이터베이스 이름으로 바꾸고 *DatabaseRegion*을 데이터베이스가 AWS 리전 있는 로 바꿉니다.

```
aws lightsail get-relational-database \
 --relational-database-name DatabaseName \
 --region DatabaseRegion
```

`get-relational-database` 응답에서 데이터베이스 state은(는) 다음 유지 관리 기간 동안 보류 중인 메이저 버전 업그레이드를 알려줍니다. 응답의 preferredMaintenanceWindow 섹션에서 다음 유지 관리 기간의 날짜 및 시간을 찾을 수 있습니다.

데이터베이스 인스턴스 상태

```
"state": "upgrading",
  "backupRetentionEnabled": true,
```

```
  "pendingModifiedValues": {
  "engineVersion": "8.0.36"
```

유지 관리 윈도우

```
"preferredMaintenanceWindow": "wed: 09:22-wed: 09:52"
```

다음 단계

테스트 데이터베이스를 생성한 경우 애플리케이션이 업그레이드된 데이터베이스와 함께 작동하는지 확인한 후 삭제할 수 있습니다. 이전 데이터베이스로 돌아가야 할 경우를 대비하여 이전 데이터베이스에서 생성한 스냅샷을 보관합니다. 또한 업그레이드된 데이터베이스의 스냅샷을 생성하여 새 특정 시점 복사본을 확보해야 합니다.

MySQL 5.6 데이터베이스의 데이터를 Lightsail의 최신 데이터베이스 버전으로 마이그레이션

이 자습서에서는 MySQL 5.6 데이터베이스에서 Amazon Lightsail의 새로운 MySQL 5.7 데이터베이스로 데이터를 마이그레이션하는 방법을 안내합니다. 마이그레이션을 수행하려면 사용 중인 MySQL 5.6 데이터베이스에 연결하고 기존 데이터를 내보내야 합니다. 그런 다음 MySQL 5.7 데이터베이스에 연결하고 데이터를 가져오면 됩니다. 새 데이터베이스에 필수 데이터가 있으면 새 데이터베이스에 연결하도록 애플리케이션을 다시 구성할 수 있습니다.

목차

- 1단계: 변경 사항 이해
- 2단계: 사전 조건 완료
- 3단계: MySQL 5.6 데이터베이스에 연결 및 데이터 내보내기
- 4단계: MySQL 5.7 데이터베이스에 연결 및 데이터 가져오기
- 5단계: 애플리케이션 테스트 및 마이그레이션 완료

1단계: 변경 사항 이해

MySQL 5.6 데이터베이스에서 MySQL 5.7 데이터베이스로의 마이그레이션은 주요 버전 업그레이드로 간주됩니다. 메이저 버전 업그레이드에는 기존 애플리케이션과 호환되지 않는 데이터베이스 변경 사

항이 포함될 수 있습니다. 모든 업그레이드는 프로덕션 환경의 인스턴스에 적용하기 전에 반드시 철저하게 테스트하는 것이 좋습니다. 자세한 내용은 MySQL 문서의 MySQL 5.7의 변경 사항을 참조하세요.

먼저 기존 MySQL 5.6 데이터베이스에서 새로운 MySQL 5.7 데이터베이스로 데이터를 마이그레이션하는 것이 좋습니다. 그런 다음 사전 프로덕션 인스턴스에서 새 MySQL 5.7 데이터베이스로 애플리케이션을 테스트합니다. 애플리케이션이 예상대로 동작하는 경우 프로덕션 인스턴스의 애플리케이션에 변경 사항을 적용합니다. 한 단계 더 나아가, 기존 MySQL 5.7 데이터베이스에서 최신 MySQL 8.0 데이터베이스로 데이터를 마이그레이션하고 사전 프로덕션 환경에서 애플리케이션을 다시 테스트한 후 프로덕션 환경의 애플리케이션에 변경 사항을 적용할 수 있습니다.

2단계: 사전 조건 완료

이 자습서의 다음 섹션을 이어서 진행하기 전에 다음 사전 조건을 완료해야 합니다.

- 데이터베이스에 연결하여 데이터를 내보내고 가져오는 데 사용할 MySQL Workbench를 로컬 컴퓨터에 설치합니다. 자세한 내용은 MySQL 웹 사이트의 MySQL Workbench 다운로드를 참조하세요.
- Lightsail에서 MySQL 5.7 데이터베이스를 생성합니다. 자세한 내용은 Amazon Lightsail에서 데이터베이스 생성을 참조하세요.
- 데이터베이스에 대해 퍼블릭 모드를 활성화합니다. 이렇게 하면 MySQL Workbench를 사용하여 연결할 수 있습니다. 데이터 내보내기 및 가져오기가 완료된 후에 데이터베이스의 퍼블릭 모드를 비활성화하면 됩니다. 자세한 내용은 데이터베이스에 대한 퍼블릭 모드 구성을 참조하세요.
- MySQL Workbench를 구성하여 데이터베이스에 연결합니다. 자세한 내용은 MySQL 데이터베이스에 연결을 참조하세요.

3단계: MySQL 5.6 데이터베이스에 연결 및 데이터 내보내기

이 자습서 섹션에서는 MySQL 5.6 데이터베이스에 연결하고 MySQL Workbench를 사용하여 데이터베이스에서 데이터를 내보냅니다. MySQL Workbench를 사용하여 데이터를 내보내는 방법에 대한 자세한 내용은 MySQL Workbench 설명서의 SQL 데이터 내보내기 및 가져오기 마법사를 참조하세요.

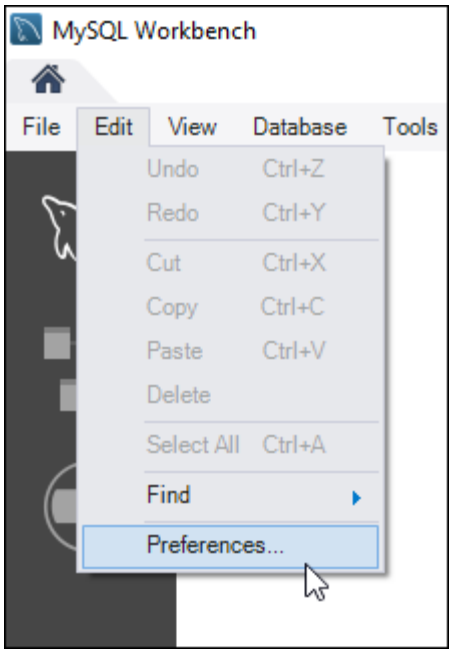
1. MySQL Workbench를 사용하여 MySQL 5.6 데이터베이스에 연결합니다.

   MySQL Workbench는 mysqldump를 사용하여 데이터를 내보냅니다. MySQL Workbench에서 사용하는 mysqldump 버전은 데이터를 내보낼 MySQL 데이터베이스 버전과 동일하거나 그 이후 버전이어야 합니다. 예를 들어, MySQL 5.6.51 데이터베이스에서 데이터를 내보내는 경우 mysqldump 버전 5.6.51 이상을 사용해야 합니다. 올바른 mysqldump 버전을 사용하려면 로컬 컴퓨터에 적절한 MySQL 서버 버전을 다운로드하여 설치해야 할 수 있습니다. 특정 MySQL 서버 버전을 다운로드하려면 MySQL 웹 사이트의 MySQL 커뮤니티 다운로드를 참조하세요. Windows

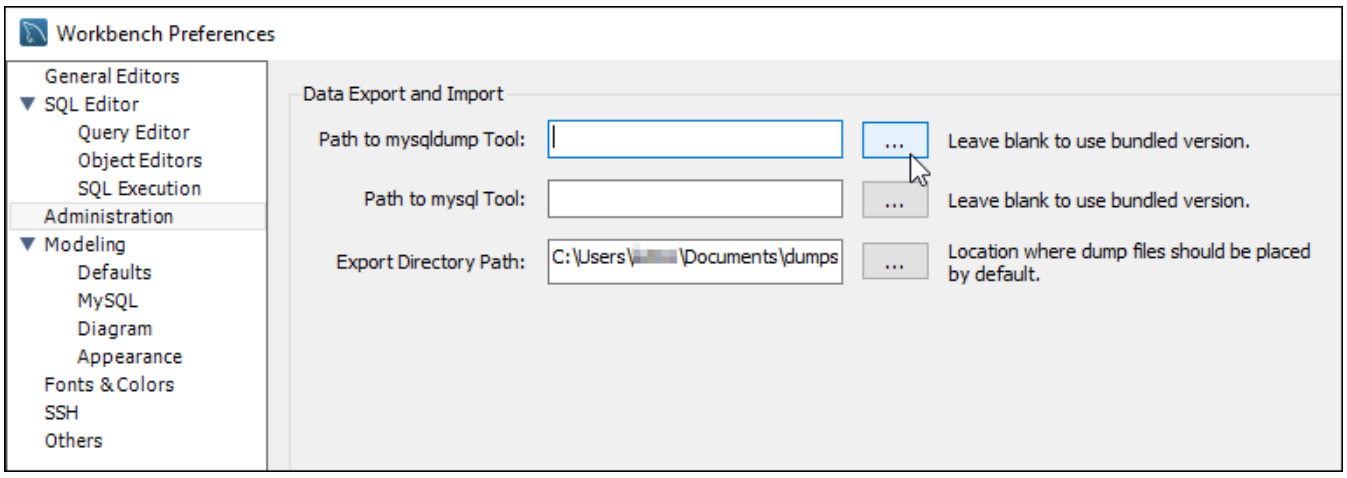
MSI용 MySQL 설치 프로그램은 모든 버전의 MySQL 서버를 다운로드할 수 있는 옵션을 제공합니다.

MySQL Workbench에서 사용할 올바른 mysqldump 버전을 선택하려면 다음 단계를 수행하세요.

1. MySQL Workbench에서 편집(Edit)과 환경설정(Preferences)을 차례로 선택합니다.

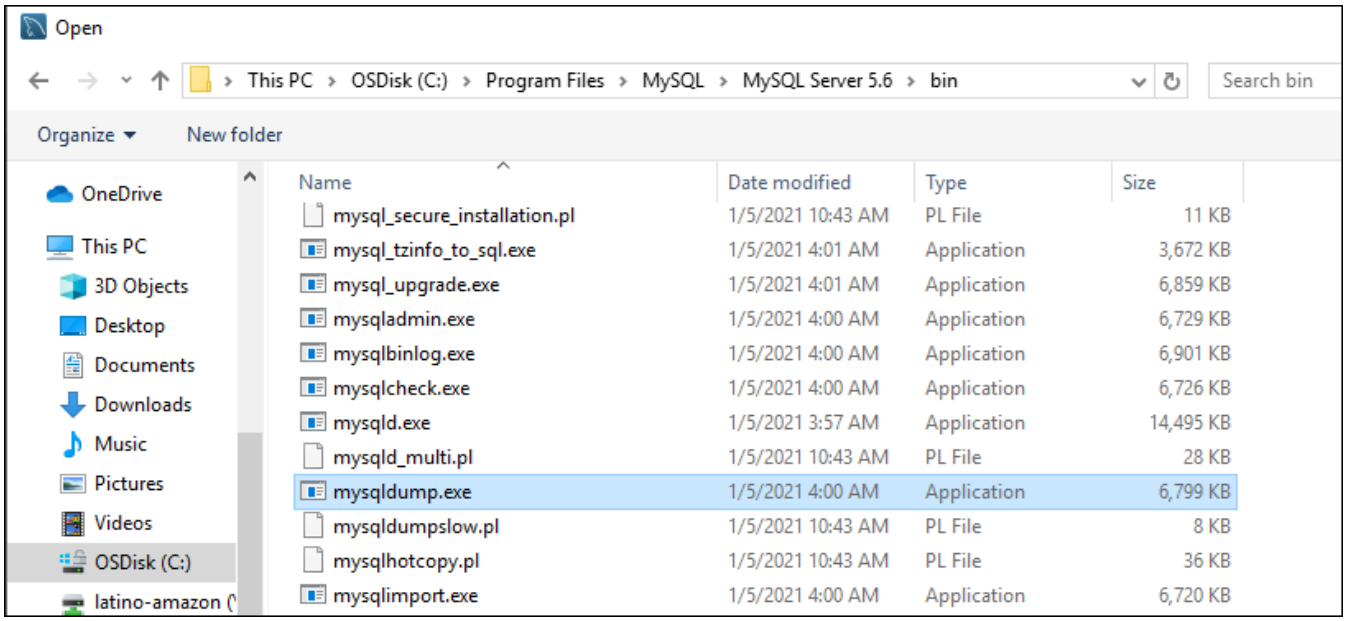


2. 탐색 창에서 관리(Administration)를 선택합니다.
3. Workbench 환경설정(Workbench Preferences) 창이 열리면 mysqldump 도구로의 경로(Path to mysqldump Tool) 텍스트 상자 옆에 있는 줄임표 버튼을 선택합니다.

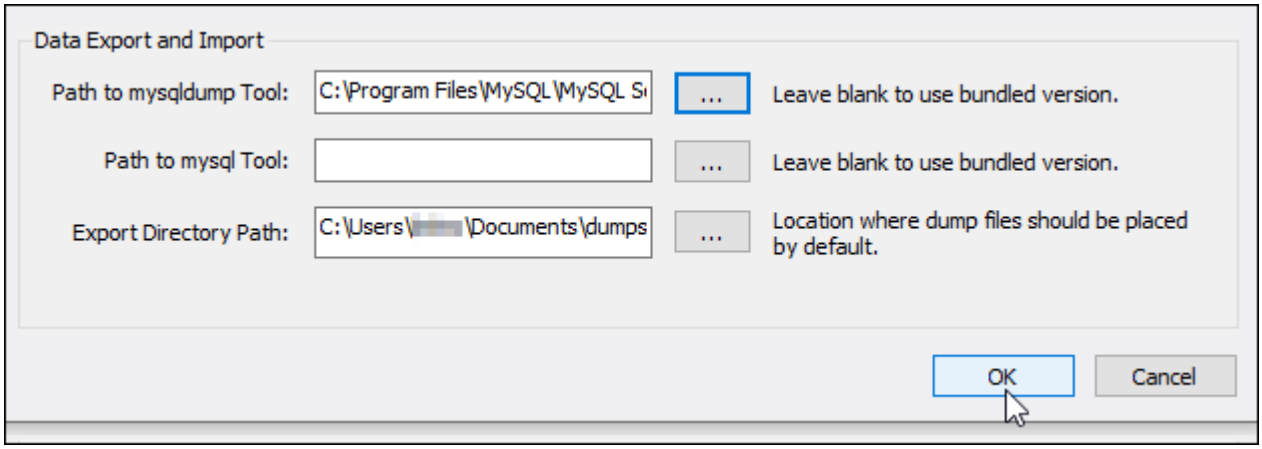


4. 적절한 `mysqldump` 실행 파일의 위치를 찾아 두 번 클릭합니다.

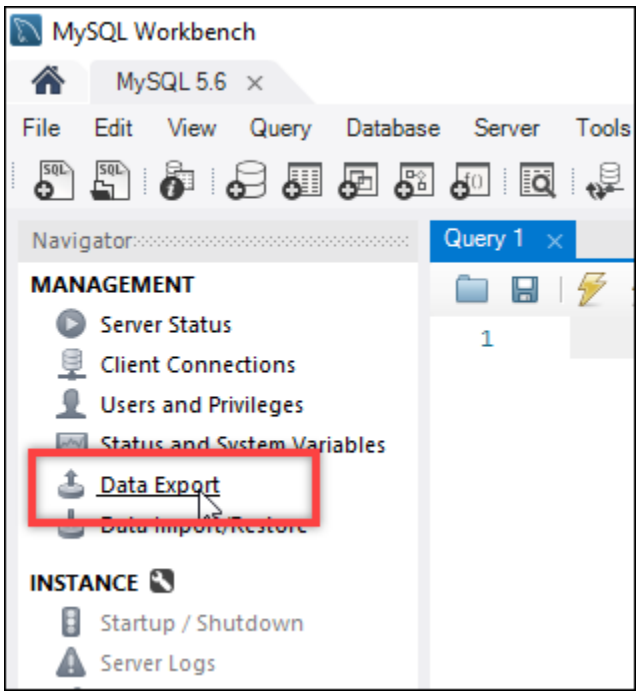
Windows에서 `mysqldump.exe` 파일은 일반적으로 `C:\Program Files\MySQL\MySQL Server 5.6\bin` 디렉터리에 있습니다. Linux의 경우 터미널에서 `which mysqldump`를 입력하여 mysqldump 파일이 있는 위치를 확인합니다.



5. Workbench 환경설정(Workbench Preferences) 창에서 확인(OK)을 선택합니다.

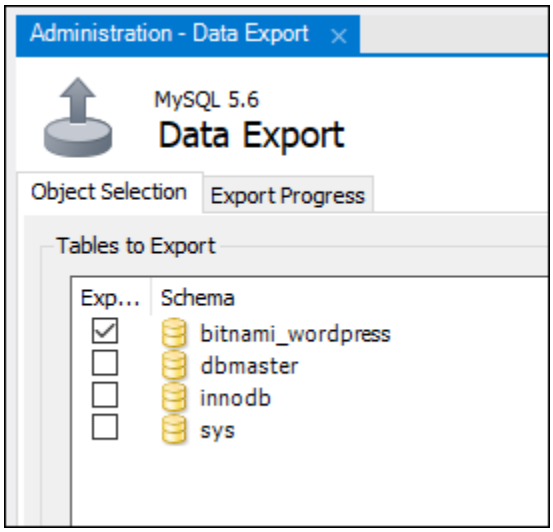


2. 탐색(Navigator) 창에서 데이터 내보내기(Data Export)를 선택합니다.

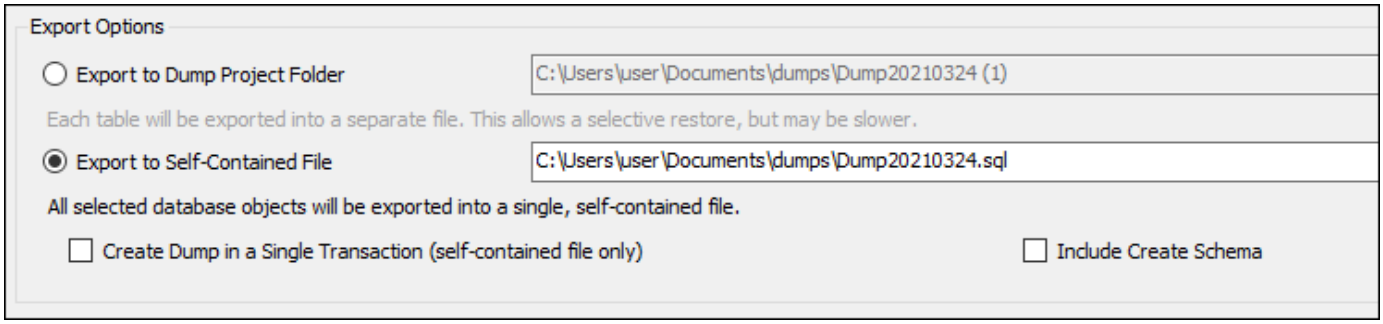


3. 데이터 내보내기 탭이 표시되면 내보낼 테이블 옆에 확인 표시를 추가합니다.

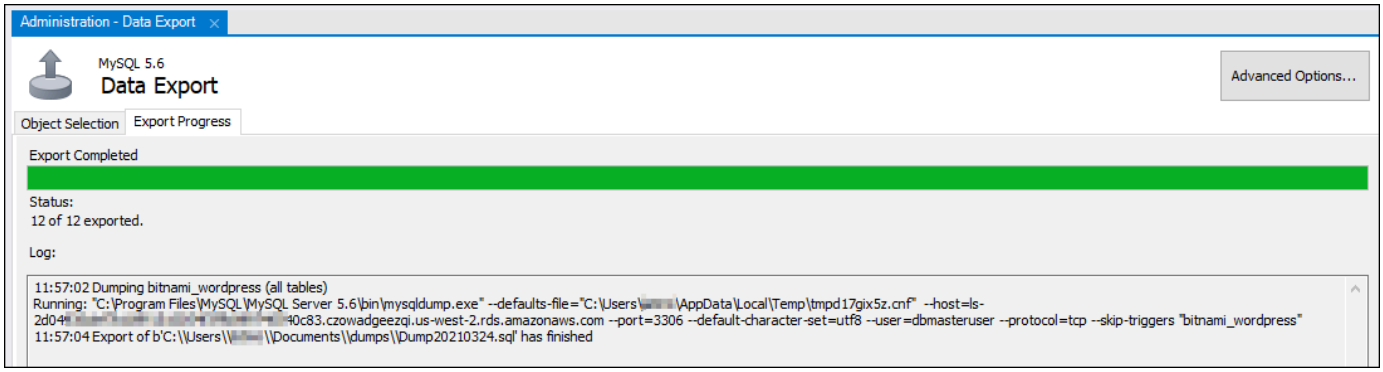
> **Note**
>
> 이 예에서는 'Bitnami 인증' WordPress 인스턴스의 WordPress 웹 사이트에 대한 데이터가 들어 있는 `bitnami_wordpress` 테이블을 선택했습니다.



4. 내보내기 옵션(Export Options) 섹션에서 자체 포함 파일로 내보내기(Export to Self-Contained File)를 선택한 후 내보낸 파일이 저장될 디렉터리를 기록합니다.



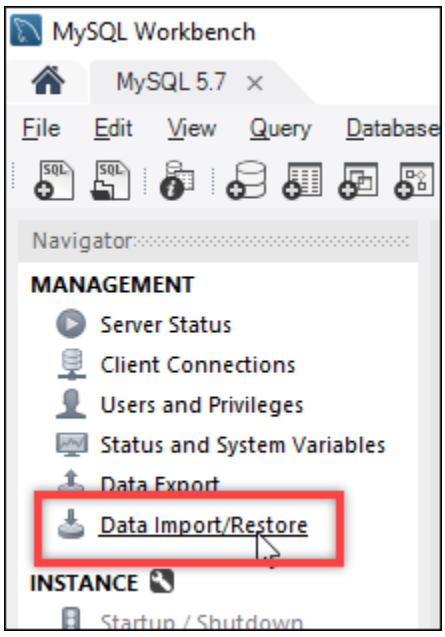
5. 내보내기 시작(Start Export)을 선택합니다.
6. 내보내기가 완료될 때까지 기다린 후 이 자습서의 다음 섹션을 이어서 진행합니다.



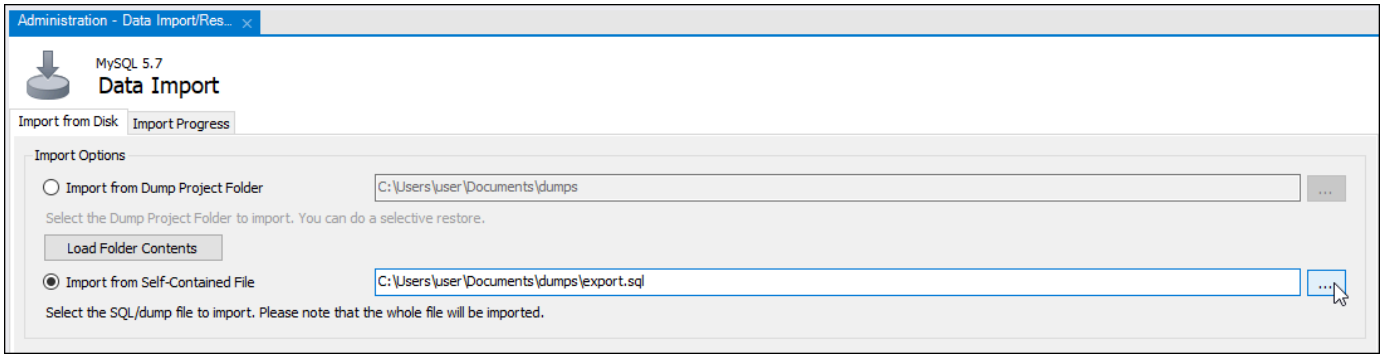
4단계: MySQL 5.7 데이터베이스에 연결 및 데이터 가져오기

이 자습서 섹션에서는 MySQL 5.7 데이터베이스에 연결하고 MySQL Workbench를 사용하여 데이터베이스로 데이터를 가져옵니다.

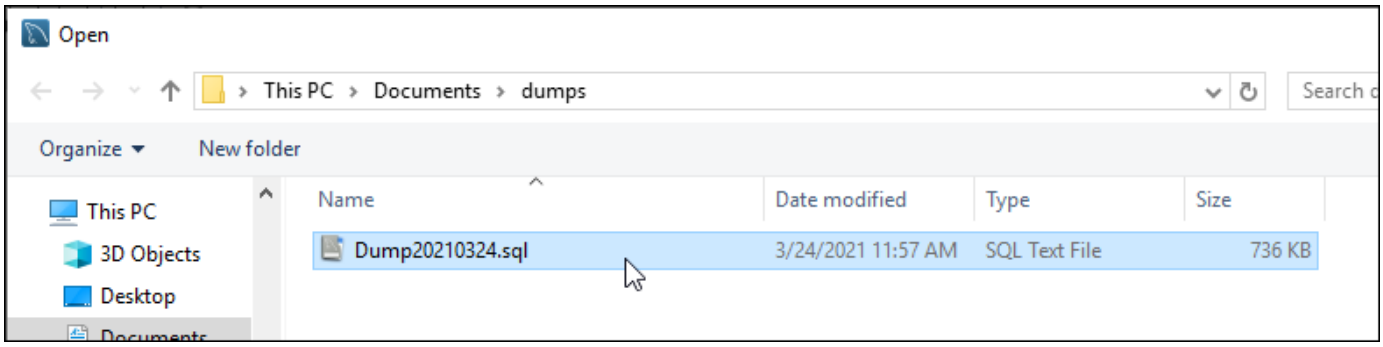
1. 로컬 컴퓨터에서 MySQL Workbench를 사용하여 MySQL 5.7 데이터베이스에 연결합니다.
2. 탐색(Navigator) 창에서 데이터 가져오기/복원(Data Import/Restore)을 선택합니다.



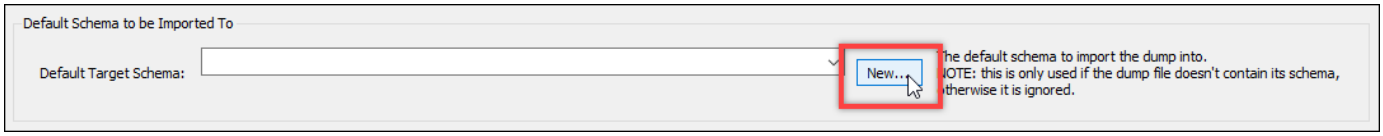
3. 데이터 가져오기(Data Import) 탭이 표시되면 자체 포함 파일에서 가져오기(Import from Self-Contained File)를 선택한 다음, 텍스트 상자 옆에 있는 줄임표 버튼을 선택합니다.



4. 내보낸 파일이 저장된 위치를 찾아서 두 번 클릭합니다.

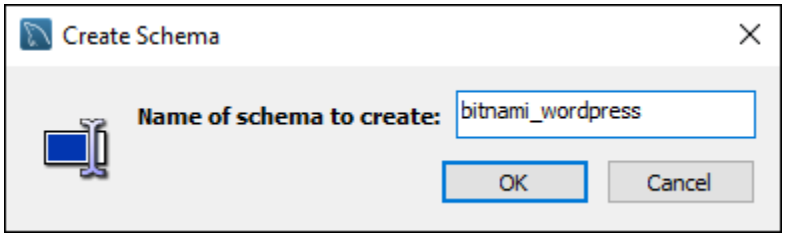


5. 다음으로 가져올 기본 스키마(Default Schema to be imported To) 섹션에서 신규(New)를 선택합니다.

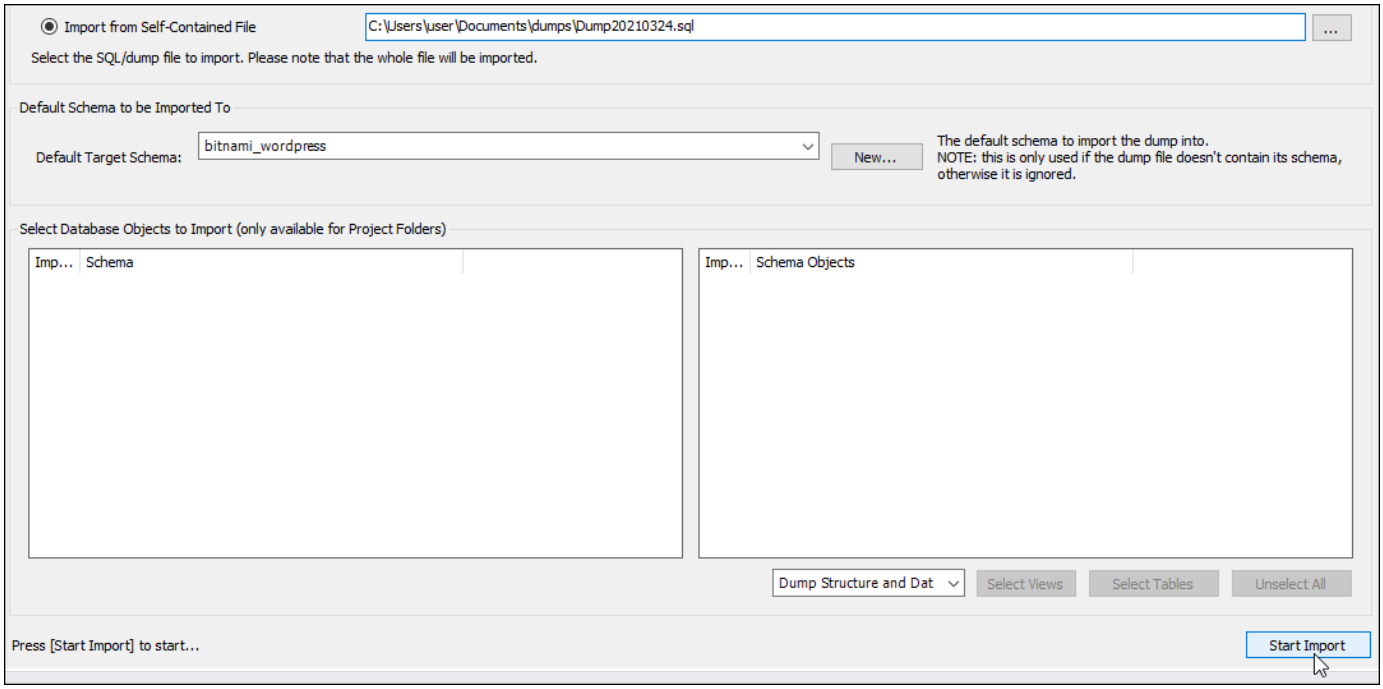


6. 스키마 생성(Create Schema) 창이 열리면 스키마 이름을 입력합니다.

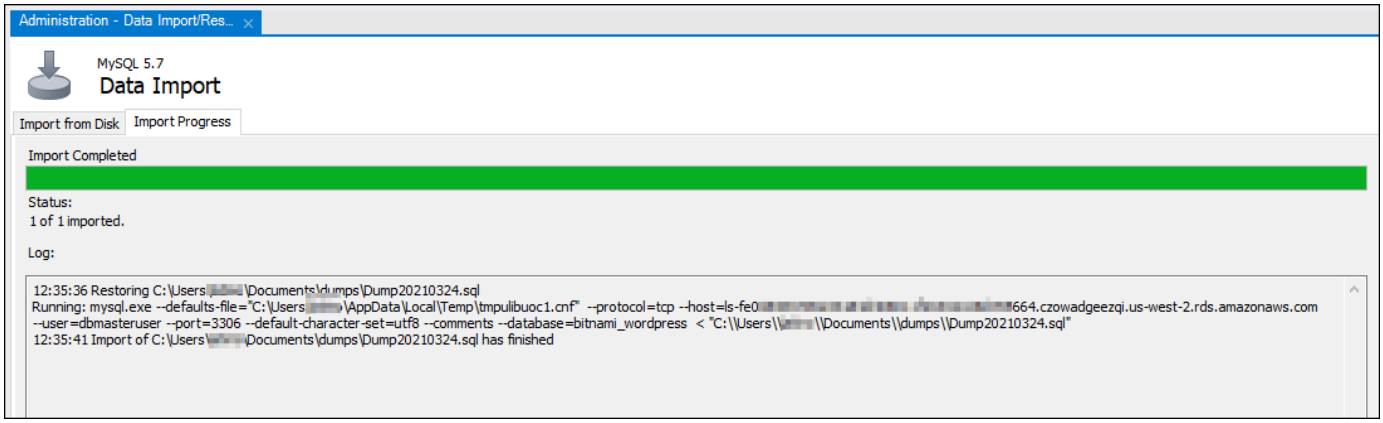
> **Note**
>
> 이 예에서는 내보낸 데이터베이스 테이블의 이름인 `bitnami_wordpress`를 입력합니다.



7. 가져오기 시작(Start Import)을 선택합니다.



8. 가져오기가 완료될 때까지 기다린 후 이 자습서의 다음 섹션을 이어서 진행합니다.



5단계: 애플리케이션 테스트 및 마이그레이션 완료

이제 데이터가 새로운 MySQL 5.7 데이터베이스에 저장되었습니다. 사전 프로덕션 환경에서 애플리케이션을 구성하고 애플리케이션과 새 MySQL 5.7 데이터베이스 간의 연결을 테스트합니다. 애플리케이션이 예상대로 동작하면 프로덕션 환경에서 애플리케이션을 변경합니다.

마이그레이션이 완료되면 데이터베이스의 퍼블릭 모드를 비활성화해야 합니다. MySQL 5.6 데이터베이스는 더 이상 필요하지 않다면 삭제해도 됩니다. 단, MySQL 5.6 데이터베이스를 삭제하기 전에 스냅샷을 생성해야 합니다. 그러는 중에 새로운 MySQL 5.7 데이터베이스의 스냅샷도 생성해야 합니다. 자세한 내용은 데이터베이스 스냅샷 생성을 참조하세요.

Lightsail 로드 밸런서를 사용하여 웹 트래픽 배포

Lightsail 로드 밸런서는 수신된 웹 트래픽을 여러 가용 영역의 여러 Lightsail 인스턴스에 분산시킵니다. 로드 밸런서는 인스턴스에서 애플리케이션의 가용성과 내결함성을 높입니다. 애플리케이션에 대한 요청의 전체적인 흐름을 방해하지 않고 필요에 따라 Lightsail 로드 밸런서에서 인스턴스를 추가 및 제거할 수 있습니다.

Lightsail 로드 밸런싱 상태에서 DNS 호스트 이름을 만들고, 이 호스트 이름으로 보내는 모든 요청을 대상 Lightsail 인스턴스 풀로 라우팅합니다. Lightsail 계정의 총 인스턴스 수 할당량을 넘어서지 않는 한, 대상 인스턴스를 원하는 만큼 로드 밸런서에 추가할 수 있습니다.

로드 밸런서 기능

Lightsail 로드 밸런서는 다음 기능을 제공합니다.

- HTTPS 암호화 - 기본적으로 Lightsail 로드 밸런서는 포트 80을 통해 암호화되지 않은(HTTP) 트래픽 요청을 처리합니다. 검증된 Lightsail SSL/TLS 인증서를 로드 밸런서에 연결하여 HTTPS 암호화를 활성화합니다. 이를 통해 로드 밸런서는 포트 443을 통해 암호화된(HTTPS) 트래픽 요청을 처리할 수 있습니다. 자세한 내용은 SSL/TLS 인증서를 참조하세요.

  다음 기능은 로드 밸런서에 HTTPS 암호화를 활성화한 후 사용할 수 있습니다.

  - HTTP에서 HTTPS로의 리디렉션 - HTTP에서 HTTPS로의 리디렉션을 활성화하여 HTTP 요청을 HTTPS 암호화된 연결로 자동 리디렉션합니다. 자세한 내용은 로드 밸런서에 대한 HTTP에서 HTTPS로의 리디렉션 구성을 참조하세요.
  - TLS 보안 정책 - 로드 밸런서에 TLS 보안 정책을 구성합니다. 자세한 내용은 Amazon Lightsail 로드 밸런서에 TLS 보안 정책 구성을 참조하십시오.
- 상태 확인 - 기본적으로 상태 확인은 실행 중인 웹 애플리케이션의 루트에 있는 연결된 인스턴스에 대해 수행됩니다. 상태 확인은 인스턴스의 상태를 모니터링하는 데 사용되며, 로드 밸런서는 이를 통해 정상적인 인스턴스에만 요청을 보낼 수 있습니다. 자세한 내용은 Lightsail 로드 밸런서 상태 확인을 참조하십시오.
- 세션 지속성 - 세션 정보를 웹 사이트 방문자의 브라우저에 로컬로 조정하는 경우 세션 지속성을 구성합니다. 예를 들어 Magento 전자 상거래 애플리케이션을 실행하면서 장바구니를 로드 밸런싱된 Lightsail 인스턴스에 둘 수 있습니다. 세션 지속성을 활성화하면, 웹 사이트 방문자가 장바구니에 품목을 추가하고 세션을 종료한 뒤 나중에 다시 돌아오면 장바구니에 해당 품목이 그대로 있습니다. 자세한 내용은 로드 밸런서에 대한 세션 지속성 활성화를 참조하세요.

로드 밸런서의 사용 시점

트래픽이 간헐적으로 급증하는 웹 사이트 또는 너무 많은 방문객이 동시에 그 콘텐츠를 요청하여 인스턴스에 큰 부하가 걸릴 수 있는 콘텐츠를 호스팅하는 웹 사이트가 있다면 로드 밸런서를 사용해야 합니다. 예를 들어, 웹 사이트에서 이미지를 많이 사용하는 경우에는 이미지 요청을 다른 페이지 요청과 조화롭게 로드 밸런싱할 수 있습니다. 이렇게 하면 페이지 로딩 속도가 빨라지고 사용자의 만족도는 높아집니다.

로드 밸런서를 사용하여 가용성이 우수한 웹 사이트를 만들어 보십시오. 고가용성이란 주어진 기간 중 웹 사이트 또는 애플리케이션이 얼마나 오래 가동 상태를 유지하는지를 나타내는 말입니다. 한 번이라도 사이트 중단 사태를 겪은 적이 있다면 로드 밸런서로 가동 시간을 늘릴 수 있습니다. Lightsail 로드 밸런서를 사용하여 대상 인스턴스를 추가하고 여러 가용 영역에 이를 분산시킴으로써 애플리케이션의 가용성을 높일 수 있습니다.

내결함성도 이와 관련된 개념입니다. 인스턴스 또는 데이터베이스 중 하나에 장애가 발생하더라도 사이트가 계속 작동한다면 내결함성이 있는 사이트입니다. 로드 밸런서는 내결함성 애플리케이션 또는 웹 사이트를 구축하는 데 도움이 됩니다.

로드 밸런싱에 권장되는 애플리케이션

로드 밸런서가 필요 없는 Lightsail 애플리케이션도 있습니다. 로드 밸런싱 기능의 애플리케이션을 만들기로 했다면 먼저 그 애플리케이션을 구성해야 합니다. 예를 들어 LAMP 스택 애플리케이션을 로드 밸런싱하기 위해서는 먼저 모든 대상 인스턴스가 읽고 쓸 수 있는 전용 중앙 데이터베이스를 만들어야 합니다. Lightsail 객체 스토리지 버킷과 같은 중앙 집중식 미디어 스토리지를 만드는 것도 고려해 볼 수 있습니다. 자세한 내용은 로드 밸런싱을 위한 인스턴스 구성을 참조하세요.

로드 밸런서 사용 시작하기

Lightsail 콘솔, AWS Command Line Interface (AWS CLI) 또는 Lightsail API를 사용하여 로드 밸런서를 생성할 수 있습니다. 로드 밸런싱을 위한 인스턴스도 구성해야 합니다.

로드 밸런서를 생성하여 구성된 인스턴스에 연결했으면, 이제 다음 주제에 따라 HTTPS를 활성화할 수 있습니다. 자세한 내용은 로드 밸런서용 SSL/TLS 인증서 생성을 참조하세요.

Lightsail 로드 밸런서를 사용하여 웹 트래픽 배포

로드 밸런서를 생성하여 애플리케이션의 중복성을 높이거나 웹 트래픽을 더 많이 처리할 수 있습니다. 로드 밸런서를 생성한 후에 균형을 조정할 Lightsail 인스턴스를 연결할 수 있습니다. 자세한 내용은 로드 밸런서를 참조하세요.

사전 조건

시작하기 전에 Lightsail 인스턴스를 로드 밸런싱용으로 준비해 두어야 합니다. 자세한 내용은 로드 밸런싱을 위한 인스턴스 구성을 참조하세요.

로드 밸런서 생성

1. Lightsail 콘솔에 로그인합니다.
2. 네트워킹 탭을 선택합니다.
3. 로드 밸런서 생성을 선택하세요.
4. 로드 밸런서가 생성될 AWS 리전 을 확인하거나 리전 변경을 선택하여 다른 리전을 선택합니다.

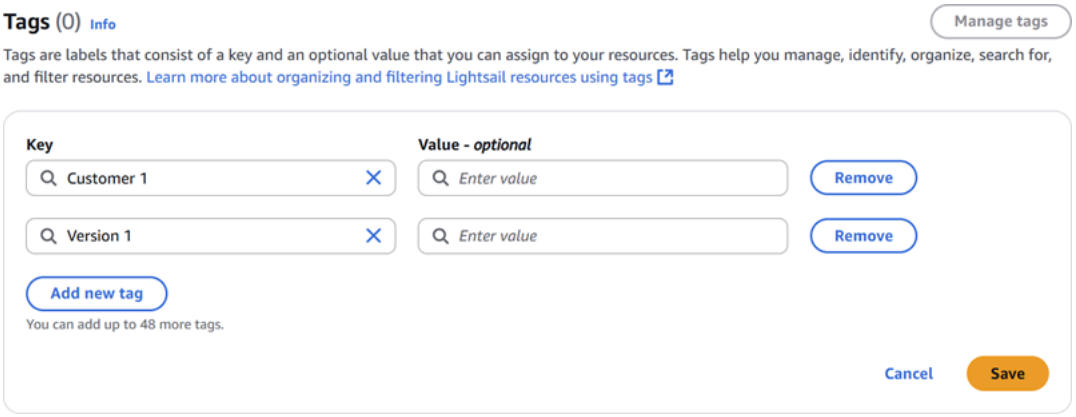
   > **Note**
   >
   > 기본적으로, HTTP 요청을 수락할 수 있도록 포트 80을 열고 로드 밸런서를 생성합니다. 로드 밸런서를 생성한 뒤에는 SSL/TLS 인증서를 생성하고 HTTPS를 구성할 수 있습니다. 자세한 내용은 로드 밸런서용 SSL/TLS 인증서 생성을 참조하세요.

5. 로드 밸런서의 이름을 입력합니다.

   리소스 이름:

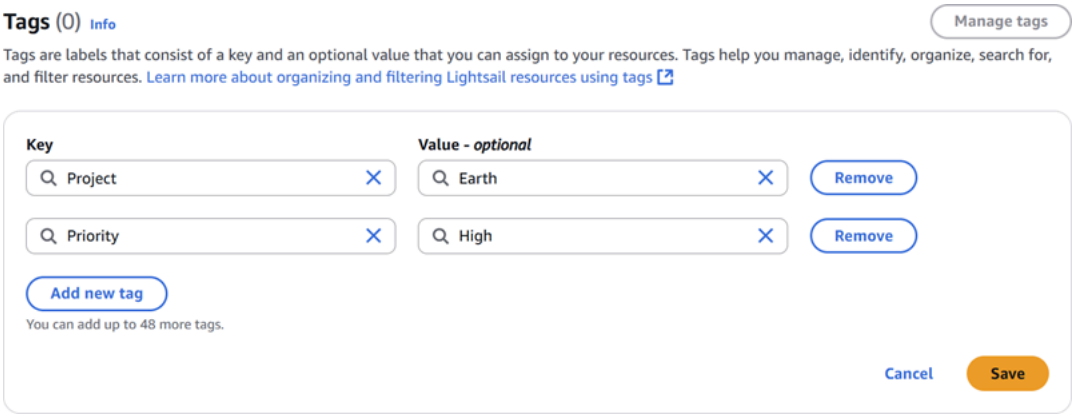
   - Lightsail 계정 AWS 리전 의 각 내에서 고유해야 합니다.
   - 2~255자의 문자로 구성되어야 합니다.
   - 영숫자 문자 또는 숫자로 시작하고 끝나야 합니다.
   - 영숫자 문자, 숫자, 마침표, 대시, 밑줄이 포함될 수 있습니다.
6. 다음 옵션 중 하나를 선택하여 로드 밸런서에 태그를 추가합니다.

   - 키 전용 태그를 추가하거나 태그를 관리(이미 태그를 추가한 경우)합니다. 새 태그를 태그 키 텍스트 상자에 입력하고 Enter를 누릅니다. 태그를 추가하려면 태그 입력이 완료될 때 저장을 선택하고, 태그를 추가하지 않으려면 취소를 선택합니다.



- 키-값 태그를 생성한 다음 키를 키 텍스트 상자에 입력하고, 값을 값 텍스트 상자에 입력합니다. 태그 입력이 완료되면 저장을 선택하고, 태그를 추가하지 않으려면 취소를 선택합니다.

  키-값 태그는 저장 전에 한 번에 하나씩만 추가할 수 있습니다. 둘 이상의 키-값 태그를 추가하려면 이전 단계를 반복하세요.



> **Note**
>
> 키 전용 태그 및 키-값 태그에 대한 자세한 내용은 태그를 참조하세요.

7. 로드 밸런서 생성을 선택하세요.

로드 밸런서에 인스턴스 연결

로드 밸런서를 생성하고 나면 Lightsail에서 로드 밸런서 관리 페이지로 이동합니다. 해당 페이지를 다시 찾아야 하는 경우 Lightsail 홈 페이지에서 네트워킹(Networking) 탭을 선택한 다음 관리할 Lightsail 로드 밸런서의 이름을 선택합니다.

> **Note**
>
> Lightsail 인스턴스를 로드 밸런서에 연결하려면 인스턴스가 실행 중이어야 합니다.

1. 로드 밸런서 관리 페이지에서 대상 인스턴스를 선택합니다.
2. 대상 인스턴스 드롭다운 메뉴에서 인스턴스를 선택합니다.
3. 연결을 선택합니다. 연결하는 데 몇 분 정도 걸릴 수 있습니다.

   다른 항목 연결을 선택하고 앞의 단계를 반복하여 다른 인스턴스를 로드 밸런서에 연결합니다.

다음 단계

로드 밸런서가 생성되고 인스턴스가 연결되면 다음 단계를 완료하여 로드 밸런스를 구성합니다.

- 로드 밸런서용 SSL/TLS 인증서 생성
- 로드 밸런서에 대한 상태 확인 사용자 지정

로드 밸런서에 문제가 있는 경우 로드 밸런서 문제 해결을 참조하세요.

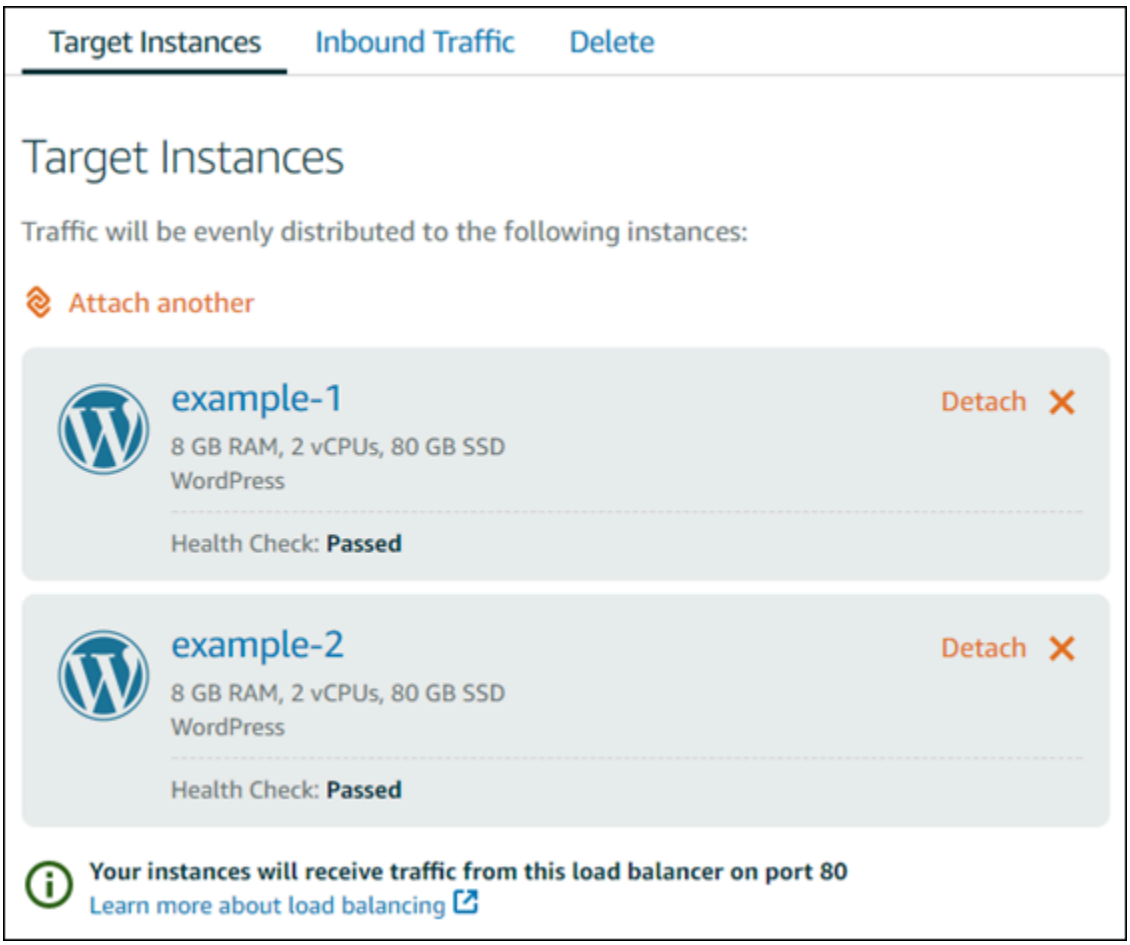
Lightsail 로드 밸런서 상태 확인 및 HTTPS 설정 사용자 지정

Lightsail 로드 밸런서를 생성할 때 AWS 리전과 이름을 선택합니다. 이 주제에서는 로드 밸런서를 업데이트하여 더 많은 옵션을 활성화하는 방법에 대해 설명합니다.

아직 수행하지 않은 경우 로드 밸런서를 생성해야 합니다. 로드 밸런서 생성

상태 확인

가장 먼저 로드 밸런싱을 위해 인스턴스를 구성해야 합니다. 이 구성을 마치고 나면 로드 밸런서에 인스턴스를 연결할 수 있습니다. 인스턴스를 연결하면 상태 확인 프로세스가 시작되고, 로드 밸런서 관리 페이지에 통과 또는 실패 메시지가 표시됩니다.



상태 확인 경로를 사용자 지정할 수도 있습니다. 예를 들어, 홈 페이지가 느리게 로드되거나 홈 페이지에 이미지가 많다면 더 빨리 로드되는 다른 페이지를 확인하도록 Lightsail을 구성할 수 있습니다. 로드 밸런서 상태 확인 경로 사용자 지정

암호화된 트래픽(HTTPS)

HTTPS를 설정하여 웹 사이트 사용자에게 보다 안전한 환경을 제공할 수 있습니다. 로드 밸런서를 설정한 다음 이 3단계 프로세스에 따라 SSL/TLS 인증서를 생성하고 확인하면 됩니다.

HTTPS에 대해 자세히 알아보기

세션 지속성

세션 지속성은 세션 정보를 사용자의 브라우저에 로컬로 저장하는 경우에 유용합니다. 예를 들어 장바구니를 Lightsail에 두고 Magento 전자 상거래 애플리케이션을 실행 중일 수 있습니다. 세션 지속성을 활성화하면, 사용자가 장바구니에 품목을 넣고 세션을 종료하더라도 나중에 다시 돌아왔을 때 장바구니에 그 품목이 그대로 있습니다.

쿠키의 지속 기간도 지속적 세션에 알맞게 조정할 수 있습니다. 이 기능은 지속 기간을 특히 길거나 짧게 하려 할 때 유용합니다. 자세한 내용은 로드 밸런서에 대한 세션 지속성 활성화를 참조하세요.

로드 밸런싱을 위한 Lightsail 인스턴스 구성

Amazon Lightsail 로드 밸런서에 인스턴스를 연결하기 전에 애플리케이션의 구성을 평가해야 합니다. 예를 들어 로드 밸런서는 데이터 계층이 애플리케이션의 나머지 부분과 분리되어 있을 경우 기능을 더 잘 발휘합니다. 이 단원에서는 각 Lightsail 인스턴스에 대해 설명하고, 로드 밸런싱 수행 여부(또는 수평 확장 조정)와 애플리케이션을 구성하는 최선의 방법에 대해 제안합니다.

일반 지침: 데이터베이스를 사용하는 애플리케이션

데이터베이스를 사용하는 Lightsail 애플리케이션의 경우 데이터베이스 인스턴스가 한 개가 되도록, 데이터베이스 인스턴스와 애플리케이션의 나머지 부분을 분리할 것을 권장합니다. 이렇게 하는 가장 큰 이유는 데이터베이스에 데이터를 여러 번 쓰지 않도록 하기 위해서입니다. 데이터베이스 인스턴스가 하나가 아니면 인스턴스 히트와 상관없이 데이터베이스게 데이터를 쓰게 됩니다.

WordPress

수평 확장 조정이 가능합니까? WordPress 블로그나 웹 사이트에 대해 가능합니다.

Lightsail 로드 밸런서를 사용하기 전 구성 권장 사항

- 로드 밸런서에 대해 실행 중인 모든 WordPress 인스턴스가 동일한 장소에서 정보를 저장하고 가져올 수 있도록 데이터베이스를 분리합니다. 데이터베이스 성능을 높여야 할 경우, 웹 서버와 독립적으로 처리 능력과 메모리를 복제하거나 변경할 수 있습니다.
- 파일과 정적 콘텐츠를Lightsail 버킷으로 오프로드하세요. 이렇게 하려면 WP Offload Media Lite 플러그 인을 WordPress 웹 사이트에 설치하고 Lightsail 버킷에 연결하도록 구성해야 합니다. 자세한 내용은 자습서: 스토리지 버킷에 WordPress 인스턴스 연결을 참조하세요.

Node.js

수평 확장 조정이 가능합니까? 예, 몇 가지를 고려하여 가능합니다.

Lightsail 로드 밸런서를 사용하기 전 구성 권장 사항

- Lightsail에서, Bitnami에서 패키징한 Node.js 스택에는 Node.js, Apache, Redis(인메모리 데이터베이스), Python이 있습니다. 배포하는 애플리케이션에 따라 일부 서버에 대해 로드 밸런싱을 수행할

수 있습니다. 하지만 모든 웹 서버에 대해 트래픽 로드 밸런싱을 수행하고 Redis를 다른 서버로 이동하도록 로드 밸런서를 구성해야 합니다.

- Redis 서버를 모든 인스턴스와 통신하도록 다른 서버로 분할합니다. 필요한 경우 데이터베이스 서버를 추가합니다.
- Redis의 주요 사용 사례 중 하나는 중앙 데이터베이스를 계속 히트하지 않아도 되도록 데이터를 로컬 캐시에 저장하는 것입니다. 세션 지속성을 활성화하여 Redis의 성능 향상을 활용하는 것이 좋습니다. 자세한 내용은 로드 밸런서에 대한 세션 지속성 활성화를 참조하세요.
- Redis 노드를 공유할 수도 있으므로 세션 지속성을 사용하여 각 머신에서 로컬 캐시를 사용하거나 노드를 공유할 수 있습니다.
- Apache를 사용하여 로드 밸런서를 배포하려면 Apache 서버에 `mod_proxy_balancer`를 포함시켜 보세요.

자세한 내용은 Node.js 애플리케이션 확장 조정을 참조하세요.

Magento

수평 확장 조정이 가능합니까? 예.

Lightsail 로드 밸런서를 사용하기 전 구성 권장 사항

- Amazon RDS 데이터베이스: Terraform Magento Adobe Commerce on과 같은 추가 구성 요소를 사용하는 Magento AWS의 AWS 참조 배포를 사용할 수 있습니다.
- 세션 지속성을 활성화하세요. Magento는 장바구니를 사용하며, 이는 여러 세션에서 여러 곳을 방문하는 고객이 새로운 세션을 위해 돌아왔을 때 장바구니에 품목을 유지할 수 있게 해줍니다. 자세한 내용은 로드 밸런서에 대한 세션 지속성 활성화를 참조하세요.

GitLab

수평 확장 조정이 가능합니까? 예, 고려 사항이 있으나 가능합니다.

Lightsail 로드 밸런서를 사용하기 전 구성 권장 사항

다음 요소를 갖춰야 합니다.

- 실행 중이고 사용 가능한 Redis 노드.
- 공유된 NFS(Network Storage Server).

- 애플리케이션의 중앙 데이터베이스(MySQL 또는 PostgreSQL). 위의 데이터베이스에 대한 일반 지침을 참조하세요.

자세한 내용은 GitLab 웹 사이트의 고가용성(High Availability)을 참조하세요.

> **Note**
>
> 위에서 언급한 공유된 네트워크 스토리지 서버(NFS)는 현재로서는 GitLab 블루프린트에서 사용할 수 없습니다.

Drupal

수평 확장 조정이 가능합니까? 예. Drupal은 애플리케이션을 수평 확장 조정하는 방법을 설명하는 공식 문서인 서버 확장 조정을 제공합니다.

Lightsail 로드 밸런서를 사용하기 전 구성 권장 사항

여러 인스턴스 간에 파일을 동기화하도록 Drupal 모듈을 설정해야 합니다. Drupal 웹 사이트에서 몇 가지 모듈을 제공하지만, 이들은 프로덕션 용도보다는 프로토타입에 더 적합할 수 있습니다.

Amazon S3에 파일을 저장할 수 있는 모듈을 사용합니다. 이렇게 하면 각 대상 인스턴스에 별도의 사본을 두는 대신 한곳에서 파일을 관리할 수 있습니다. 즉, 파일을 편집할 경우 중앙 저장소에서 업데이트본을 가져오므로 히트 인스턴스에 상관없이 사용자가 동일한 파일을 볼 수 있습니다.

- Amazon S3 파일 시스템
- 콘텐츠 동기화

자세한 내용은 클라우드에서의 Drupal 수평 확장 조정을 참조하세요.

LAMP 스택

수평 확장 조정이 가능합니까? 예.

Lightsail 로드 밸런서를 사용하기 전 구성 권장 사항

- 별도의 인스턴스에 데이터베이스를 만들어야 합니다. 로드 밸런싱 대상의 모든 인스턴스가 동일한 장소에서 정보를 저장하고 가져올 수 있도록 이 데이터베이스 인스턴스를 지정해야 합니다.

- 배포하려는 애플리케이션에 따라 파일 시스템을 고려하는 방법(NFS, Lightsail 블록 스토리지 디스크, Amazon S3 스토리지)을 생각해 보세요.

MEAN 스택

수평 확장 조정이 가능합니까? 예.

Lightsail 로드 밸런서를 사용하기 전 구성 권장 사항

MongoDB를 다른 머신으로 옮기고 Lightsail 인스턴스에서 루트 문서를 공유하는 방법을 구성합니다.

Redmine

수평 확장 조정이 가능합니까? 예.

Lightsail 로드 밸런서를 사용하기 전 구성 권장 사항

- Redmine_S3 플러그인을 받아서 로컬 파일 시스템이 아니라 Amazon S3에 첨부 파일을 저장합니다.
- 데이터베이스를 다른 인스턴스와 분리합니다.

Nginx

수평 확장 조정이 가능합니까? 예.

한 개 또는 여러 개의 Lightsail 인스턴스를 Nginx에서 실행하고 Lightsail 로드 밸런서에 연결할 수 있습니다. 자세한 내용은 NGINX로 웹 애플리케이션 확장 조정, 1부: 로드 밸런싱을 참조하세요.

Joomla!

수평 확장 조정이 가능합니까? 예, 고려 사항이 있으나 가능합니다.

Lightsail 로드 밸런서를 사용하기 전 구성 권장 사항

Joomla! 웹 사이트에 대한 공식 문서는 없지만 해당 커뮤니티 포럼에 몇 가지 설명이 있습니다. 일부 사용자는 클러스터가 있는 Joomla! 인스턴스를 다음과 같은 구성으로 수평 확장 조정할 수 있었습니다.

- 세션 지속성을 활성화하도록 구성한 Lightsail 로드 밸런서. 자세한 내용은 로드 밸런서에 대한 세션 지속성 활성화를 참조하세요.

- Joomla!가 실행되는 여러 Lightsail 인스턴스가 동기화된 Joomla!의 문서 루트가 있는 로드 밸런서에 연결됨. 이렇게 하려면 Rsync와 같은 도구를 사용하여 모든 Lightsail 인스턴스 간의 콘텐츠 동기화를 담당하는 NFS 서버를 두거나, AWS를 사용하여 파일을 공유하면 됩니다.
- 복제 클러스터로 구성된 데이터베이스 서버 여러 개.
- 각 Lightsail 인스턴스에 구성된 동일한 캐시 시스템. JotCache 등과 같은 유용한 확장이 몇 가지 제공됩니다.

Lightsail 로드 밸런서에 TLS 보안 정책 구성

Amazon Lightsail 로드 밸런서에 HTTPS를 활성화한 후, 암호화된 연결에 대한 TLS 보안 정책을 구성할 수 있습니다. 이 가이드에서는 Lightsail 로드 밸런서에 구성할 수 있는 보안 정책에 대한 정보 및 로드 밸런서의 보안 정책을 업데이트하는 절차를 제공합니다. 로드 밸런서에 대한 자세한 내용은 로드 밸런서를 참조하세요.

보안 정책 개요

Lightsail 로드 밸런싱은 보안 정책이라고 하는 Secure Socket Layer(SSL) 협상 구성을 사용해 클라이언트와 로드 밸런서 간의 연결을 협상합니다. 보안 정책은 프로토콜과 암호의 조합입니다. 프로토콜은 클라이언트와 서버 간에 보안 연결을 설정하여 클라이언트와 로드 밸런서 간에 전달되는 모든 데이터를 안전하게 보호합니다. 암호는 코딩된 메시지를 생성하기 위해 암호화 키를 사용하는 암호화 알고리즘입니다. 프로토콜은 여러 개의 암호를 사용해 인터넷 상의 데이터를 암호화합니다. 연결 협상이 이루어지는 동안 클라이언트와 로드 밸런서는 각각이 지원하는 암호 및 프로토콜 목록을 선호도 순으로 표시합니다. 기본적으로 서버의 목록에서 클라이언트의 암호 중 하나와 일치하는 첫 번째 암호가 보안 연결을 위해 선택됩니다. Lightsail 로드 밸런서는 클라이언트 또는 대상 연결에 대한 SSL 재협상을 지원하지 않습니다.

Lightsail 로드 밸런서에 HTTPS를 사용하도록 설정하면 `TLS-2016-08` 보안 정책이 기본적으로 구성됩니다. 이 가이스의 뒷부분에 설명된 대로 필요에 따라 다른 보안 정책을 구성할 수 있습니다. 프런트엔드 연결에만 사용되는 보안 정책을 선택할 수 있습니다. `TLS-2016-08` 보안 정책은 항상 백엔드 연결에 사용됩니다. Lightsail 로드 밸런서는 사용자 지정 보안 정책을 지원하지 않습니다.

지원되는 보안 정책 및 프로토콜

Lightsail 로드 밸런서는 다음과 같은 보안 정책 및 프로토콜을 사용하여 구성할 수 있습니다.

Security policies	TLS-2016-08 (default)	TLS-FS-1-2-Res-2019-08
TLS Protocols		
Protocol-TLSv1	✓	
Protocol-TLSv1.1	✓	
Protocol-TLSv1.2	✓	✓
TLS Ciphers		
ECDHE-ECDSA-AES128-GCM-SHA256	✓	✓
ECDHE-RSA-AES128-GCM-SHA256	✓	✓
ECDHE-ECDSA-AES128-SHA256	✓	✓
ECDHE-RSA-AES128-SHA256	✓	✓
ECDHE-ECDSA-AES128-SHA	✓	
ECDHE-RSA-AES128-SHA	✓	
ECDHE-ECDSA-AES256-GCM-SHA384	✓	✓
ECDHE-RSA-AES256-GCM-SHA384	✓	✓
ECDHE-ECDSA-AES256-SHA384	✓	✓
ECDHE-RSA-AES256-SHA384	✓	✓
ECDHE-RSA-AES256-SHA	✓	
ECDHE-ECDSA-AES256-SHA	✓	
AES128-GCM-SHA256	✓	
AES128-SHA256	✓	
AES128-SHA	✓	

사전 조건 완료

아직 수행하지 않은 경우 다음 사전 조건을 완료합니다.

- 로드 밸런서 생성하고 여기에 인스턴스를 연결합니다. 자세한 내용은 로드 밸런스를 생성하여 인스턴스 연결을 참조하세요.
- SSL/TLS 인증서를 생성하고 이를 로드 밸런서에 연결하여 HTTPS를 활성화합니다. 자세한 내용은 Lightsail 로드 밸런서용 SSL/TLS 인증서 생성을 참조하세요. 인증서에 대한 자세한 내용은 SSL/TLS 인증서를 참조하세요.

Lightsail 콘솔을 사용하여 보안 정책 구성

Lightsail 콘솔을 사용하여 보안 정책을 구성하려면 다음 절차를 완료합니다.

1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 네트워킹을 선택합니다.
3. TLS 보안 정책을 구성할 로드 밸런서의 이름을 선택합니다.
4. 인바운드 트래픽 탭을 선택합니다.
5. 페이지의 TLS 보안 프로토콜 섹션 아래에 있는 프로토콜 변경을 선택합니다.
6. 지원되는 프로토콜 드롭다운 메뉴에서 다음 옵션 중 하나를 선택합니다.
   - TLS 버전 1.2 - 이 옵션은 가장 안전하지만 오래된 브라우저가 연결할 수 없을 수 있습니다.
   - TLS 버전 1.0, 1.1, 1.2 - 이 옵션은 브라우저와의 호환성이 가장 좋습니다.
7. 저장을 선택하여 선택된 프로토콜을 로드 밸런서에 적용합니다.

   변경 사항이 적용되는 데에는 약간의 시간이 걸립니다.

를 사용하여 보안 정책 구성 AWS CLI

AWS Command Line Interface (AWS CLI)을 사용하여 보안 정책을 구성하려면 다음 절차를 완료합니다. `update-load-balancer-attribute` 명령을 사용하여 이 작업을 수행할 수 있습니다. 자세한 내용은 AWS CLI 명령 레퍼런스의 update-load-balancer-attribute를 참조하세요.

> **Note**
>
> 이 절차를 계속하기 Lightsail 전에 AWS CLI 를 설치하고에 대해 구성해야 합니다. 자세한 내용은 에서 작동 AWS CLI 하도록 구성을 Lightsail참조하세요.

1. 명령 프롬프트 또는 터미널 창을 엽니다.
2. 로드 밸런서의 TLS 보안 정책을 변경하려면 다음 명령을 입력합니다.

```
aws lightsail update-load-balancer-attribute --load-balancer-name LoadBalancerName
 --attribute-name TlsPolicyName --attribute-value AttributeValue
```

명령에서 아래 예 텍스트를 사용자의 값으로 대체합니다.

- *LoadBalancerName*을 TLS 보안 정책을 변경하려는 로드 밸런서의 이름으로 대체합니다.
- *AttributeValue*를 TLS-2016-08 또는 TLS-FS-1-2-Res-2019-08 보안 정책으로 대체합니다.

> **Note**
>
> 명령의 TlsPolicyName 속성은 로드 밸런서에 구성된 TLS 보안 정책을 편집하도록 지정합니다.

예제:

```
aws lightsail update-load-balancer-attribute --load-balancer-name MyLoadBalancer --
attribute-name TlsPolicyName --attribute-value TLS-2016-08
```

변경 사항이 적용되는 데에는 약간의 시간이 걸립니다.

Lightsail 로드 밸런서에 대해 HTTP에서 HTTPS로 리디렉션

Amazon Lightsail 로드 밸런서에 HTTPS를 구성했으면 HTTPS로 HTTPS 리디렉션을 구성할 수 있습니다. 이렇게 하면 HTTP 연결을 사용하여 웹 사이트 또는 웹 애플리케이션을 탐색하는 사용자가 암호화된 HTTPS 연결로 자동 리디렉션됩니다. 로드 밸런서에 대한 자세한 내용은 로드 밸런서를 참조하세요.

사전 조건 완료

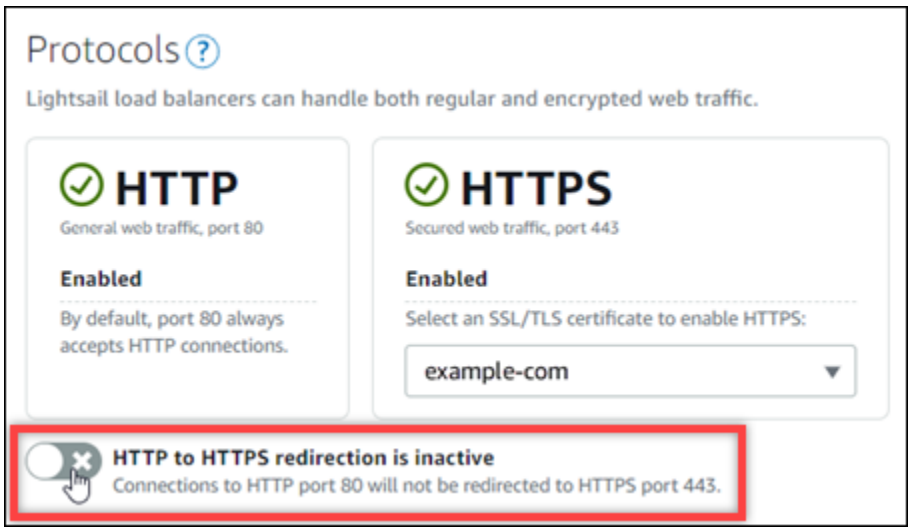
아직 수행하지 않은 경우 다음 사전 조건을 완료합니다.

- 로드 밸런서 생성하고 여기에 인스턴스를 연결합니다. 자세한 내용은 로드 밸런스를 생성하여 인스턴스 연결을 참조하세요.
- SSL/TLS 인증서를 생성하고 이를 로드 밸런서에 연결하여 HTTPS를 활성화합니다. 자세한 내용은 Lightsail 로드 밸런서용 SSL/TLS 인증서 생성을 참조하세요. 인증서에 대한 자세한 내용은 SSL/TLS 인증서를 참조하세요.

Lightsail 콘솔을 사용하여 로드 밸런서에 HTTPS 리디렉션 구성

다음 절차를 완료하여 Lightsail 콘솔에서 로드 밸런서에 HTTPS 리디렉션을 구성합니다.

1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 네트워킹을 선택합니다.
3. HTTPS 리디렉션을 구성할 로드 밸런서의 이름을 선택합니다.
4. 인바운드 트래픽 탭을 선택합니다.
5. 페이지의 프로토콜 섹션에서 다음 작업 중 하나를 수행할 수 있습니다.



   - HTTP에서 HTTPS로의 리디렉션을 켜려면 방향 옵션을 활성으로 전환합니다.
   - HTTP에서 HTTPS로의 리디렉션을 끄려면 방향 옵션을 비활성으로 전환합니다.

   변경 사항이 적용되는 데에는 약간의 시간이 걸립니다.

를 사용하여 로드 밸런서에 대한 HTTP에서 HTTPS로의 리디렉션 구성 AWS CLI

AWS Command Line Interface ()를 사용하여 로드 밸런서에 HTTPS 리디렉션을 구성하려면 다음 절차를 완료하세요AWS CLI. `update-load-balancer-attribute` 명령을 사용하여 이 작업을 수행할 수 있습니다. 자세한 내용은 AWS CLI 명령 레퍼런스의 update-load-balancer-attribute를 참조하세요.

> **Note**
>
> 이 절차를 계속하기 Lightsail 전에 AWS CLI 를 설치하고에 대해 구성해야 합니다. 자세한 내용은 에서 작동 AWS CLI 하도록 구성을 Lightsail참조하세요.

1. 명령 프롬프트 또는 터미널 창을 엽니다.
2. 다음 명령을 입력하여 로드 밸런서에 HTTPS 리디렉션을 구성합니다.

```
aws lightsail update-load-balancer-attribute --load-balancer-name LoadBalancerName
 --attribute-name HttpsRedirectionEnabled --attribute-value AttributeValue
```

명령에서 아래 예 텍스트를 사용자의 값으로 대체합니다.

- *LoadBalancerName*을 HTTP에서 HTTPS로의 리디렉션을 활성화하거나 비활성화하려는 로드 밸런서의 이름으로 대체합니다.
- 리디렉션을 활성화하려면 *AttributeValue*를 `true`로, 리디렉션을 비활성화하려면 `false`로 대체합니다.

> **Note**
>
> 명령의 `HttpsRedirectionEnabled` 속성은 지정된 로드 밸런서에 대해 HTTPS 리디렉션을 활성화 또는 비활성화할지 여부를 편집하도록 지정합니다.

예시:

- 로드 밸런서에 HTTP에서 HTTPS로의 리디렉션을 활성화합니다.

```
aws lightsail update-load-balancer-attribute --load-balancer-name MyLoadBalancer
 --attribute-name HttpsRedirectionEnabled --attribute-value true
```

- 로드 밸런서에 HTTP에서 HTTPS로의 리디렉션을 비활성화합니다.

```
aws lightsail update-load-balancer-attribute --load-balancer-name MyLoadBalancer
 --attribute-name HttpsRedirectionEnabled --attribute-value false
```

변경 사항이 적용되는 데에는 약간의 시간이 걸립니다.

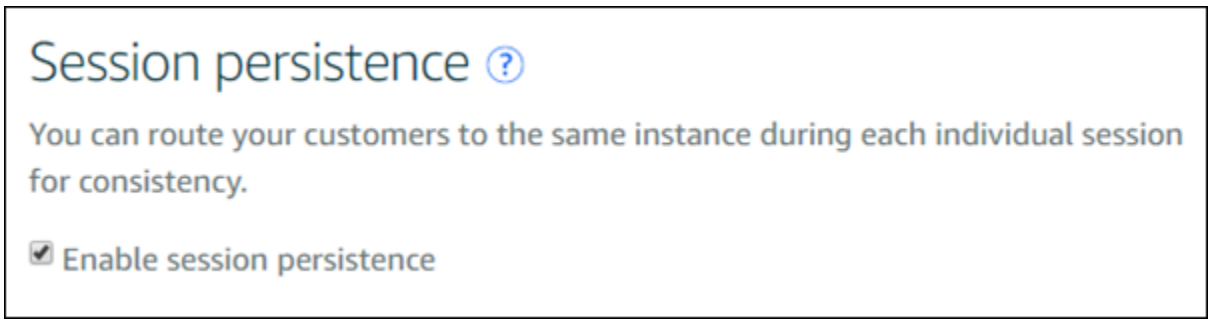
Lightsail 로드 밸런서에 대한 세션 지속성 활성화

사용자에 대해 세션 지속성을 활성화할 수 있습니다. 이 기능은 세션 정보를 사용자의 브라우저에 로컬로 저장하는 경우에 유용합니다. 예를 들어 장바구니를 Amazon Lightsail에 두고 Magento 전자 상거래 애플리케이션을 실행 중일 수 있습니다. 세션 지속성을 활성화하면, 사용자가 자신의 장바구니에 품목을 추가한 후 사이트를 떠나도 다시 돌아왔을 때 장바구니에 항목을 유지할 수 있습니다.

AWS Command Line Interface (AWS CLI) 또는 Lightsail API를 사용하여 쿠키 기간을 조정할 수도 있습니다.

세션 지속성 활성화

1. 왼쪽 탐색 창에서 네트워킹을 선택합니다.
2. 관리할 로드 밸런서를 선택합니다.
3. 인바운드 트래픽 탭을 선택합니다.
4. 세션 지속성 활성화를 선택합니다.



쿠키 지속 기간 조정

쿠키의 지속 기간도 지속적 세션에 알맞게 조정할 수 있습니다. 이 기능은 지속 기간을 특히 길거나 짧게 하려 할 때 유용합니다. 예를 들어 많은 전자 상거래 사이트는 지속 기간이 매우 깁니다. 이렇게 하면 고객이 사이트를 떠났다가 다시 돌아와도 장바구니의 항목을 그대로 유지할 수 있습니다.

아직 설정하지 않은 경우를 설정하고 AWS CLI 구성합니다.

AWS Command Line Interface 가 Amazon Lightsail과 함께 작동하도록 구성

1. 명령 프롬프트 또는 터미널 창을 엽니다.
2. 다음 AWS CLI 명령을 입력하여 쿠키 기간을 3일(259,200초)로 늘립니다.

```
aws lightsail update-load-balancer-attribute --load-balancer-name LoadBalancerName
 --attribute-name SessionStickiness_LB_CookieDurationSeconds --attribute-value
 259200
```

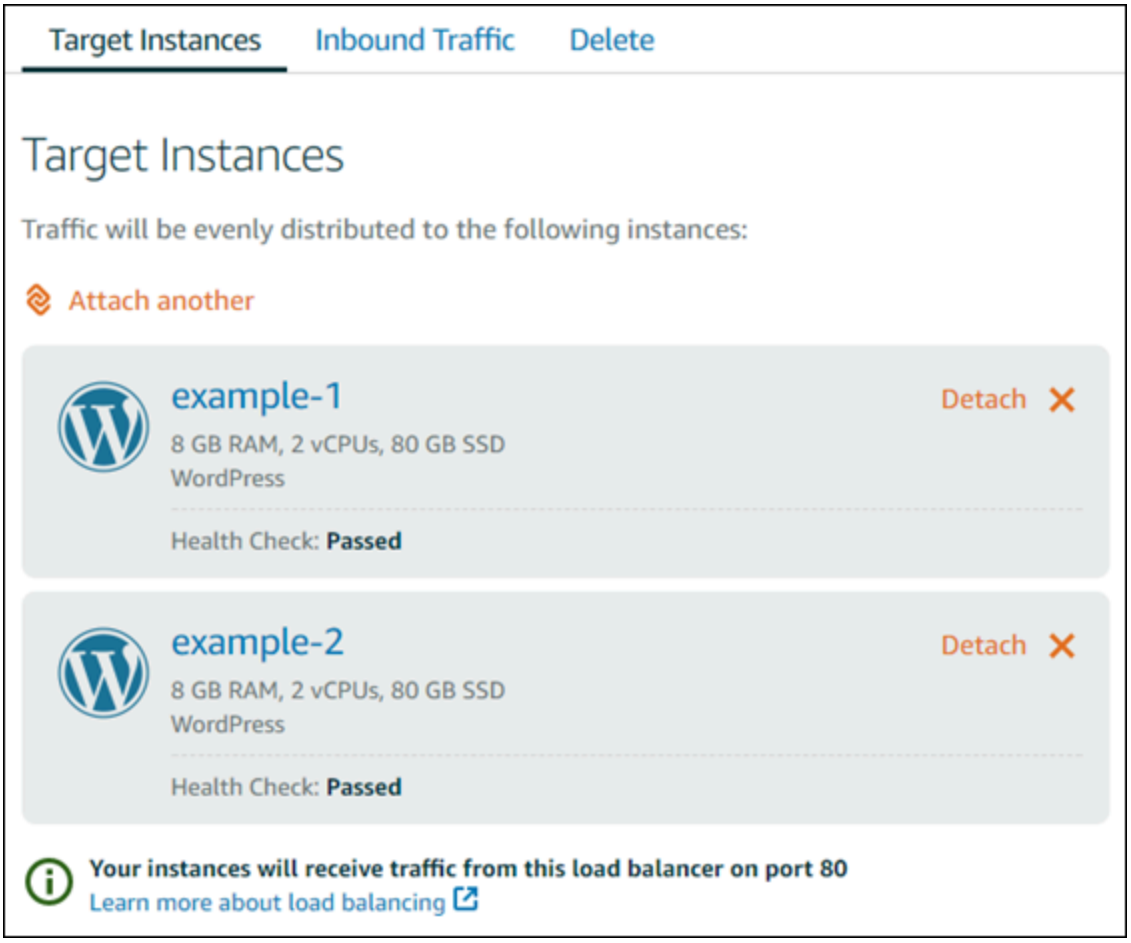
명령에서 *LoadBalancerName*을 로드 밸런서 이름으로 바꿉니다.

성공하면 다음 응답이 표시됩니다.

```
{
    "operations": [
        {
            "status": "Succeeded",
            "resourceType": "LoadBalancer",
            "isTerminal": true,
            "operationDetails": "SessionStickiness_LB_CookieDurationSeconds",
            "statusChangedAt": 1511758936.174,
            "location": {
                "availabilityZone": "all",
                "regionName": "us-west-2"
            },
            "operationType": "UpdateLoadBalancerAttribute",
            "resourceName": "example-load-balancer",
            "id": "681c2bd9-9a51-402b-8ad2-12345EXAMPLE",
            "createdAt": 1511758936.174
        }
    ]
}
```

Lightsail 로드 밸런서에 대한 상태 확인 설정 구성

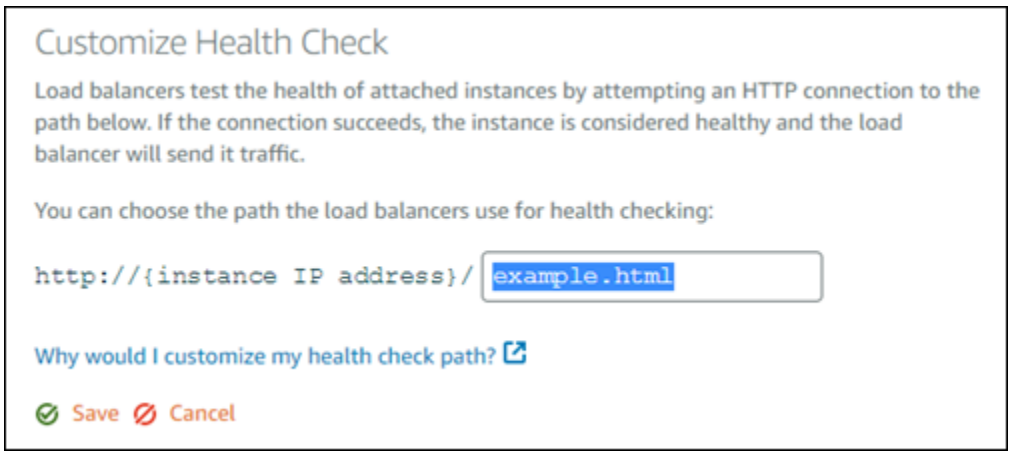
Lightsail 인스턴스를 로드 밸런서에 연결하는 즉시 상태 확인이 시작되고, 이후 30분마다 수행됩니다. 로드 밸런서 관리 페이지에서 상태 확인 현황을 볼 수 있습니다.



상태 확인 경로 사용자 지정

상태 확인 경로를 사용자 지정해야 하는 경우가 있습니다. 예를 들어, 홈 페이지가 느리게 로드되거나 홈 페이지에 이미지가 많다면 더 빨리 로드되는 다른 페이지를 확인하도록 Lightsail을 구성할 수 있습니다.

1. 왼쪽 탐색 창에서 네트워킹을 선택합니다.
2. 관리할 로드 밸런서를 선택합니다.
3. 대상 인스턴스 탭에서 상태 검사 사용자 지정을 선택합니다.
4. 유효한 상태 확인 경로를 입력한 후 저장을 선택합니다.



상태 확인 지표

다음 지표는 상태 확인의 문제를 진단하는 데 도움이 됩니다. AWS Command Line Interface 또는 Lightsail API를 사용하여 특정 상태 확인 지표에 대한 정보를 반환합니다.

- **ClientTLSNegotiationErrorCount** - 로드 밸런서와 세션을 구성하지 않은 클라이언트에서 시작된 TLS 연결 수. 가능한 원인으로는 암호 또는 프로토콜 불일치가 있습니다.

  Statistics: 가장 유용한 통계는 Sum입니다.

- **HealthyHostCount** - 정상으로 간주되는 대상의 수입니다.

  Statistics: 가장 유용한 통계는 Average, Minimum 및 Maximum입니다.

- **UnhealthyHostCount** - 비정상으로 간주되는 대상의 수입니다.

  Statistics: 가장 유용한 통계는 Average, Minimum 및 Maximum입니다.

- **HTTPCode_LB_4XX_Count** - 로드 밸런서에서 생성된 HTTP 4XX 클라이언트 오류 코드 수. 클라이언트 오류는 요청 형식이 잘못되었거나 불완전할 때 생성됩니다. 이러한 요청은 대상 인스턴스에 수신되지 않습니다. 대상 인스턴스에서 생성된 응답 코드는 이 숫자에 포함되지 않습니다.

  Statistics: 가장 유용한 통계는 Sum입니다. Minimum, Maximum 및 Average는 모두 1을 반환합니다.

- **HTTPCode_LB_5XX_Count** - 로드 밸런서에서 생성된 HTTP 5XX 서버 오류 코드 수. 대상 인스턴스에서 생성된 응답 코드는 이 숫자에 포함되지 않습니다.

  Statistics: 가장 유용한 통계는 Sum입니다. Minimum, Maximum 및 Average는 모두 1을 반환합니다. Minimum, Maximum 및 Average는 모두 1을 반환합니다.

- **HTTPCode_Instance_2XX_Count** - 대상 인스턴스에 의해 생성된 HTTP 응답 코드 수. 단, 로드 밸런서에서 생성된 응답 코드 수는 여기에 포함되지 않습니다.

  Statistics: 가장 유용한 통계는 Sum입니다. Minimum, Maximum 및 Average는 모두 1을 반환합니다.

- **HTTPCode_Instance_3XX_Count** - 대상 인스턴스에 의해 생성된 HTTP 응답 코드 수. 단, 로드 밸런서에서 생성된 응답 코드 수는 여기에 포함되지 않습니다.

  Statistics: 가장 유용한 통계는 Sum입니다. Minimum, Maximum 및 Average는 모두 1을 반환합니다.

- **HTTPCode_Instance_4XX_Count** - 대상 인스턴스에 의해 생성된 HTTP 응답 코드 수. 단, 로드 밸런서에서 생성된 응답 코드 수는 여기에 포함되지 않습니다.

  Statistics: 가장 유용한 통계는 Sum입니다. Minimum, Maximum 및 Average는 모두 1을 반환합니다.

- **HTTPCode_Instance_5XX_Count** - 대상 인스턴스에 의해 생성된 HTTP 응답 코드 수. 단, 로드 밸런서에서 생성된 응답 코드 수는 여기에 포함되지 않습니다.

  Statistics: 가장 유용한 통계는 Sum입니다. Minimum, Maximum 및 Average는 모두 1을 반환합니다.

- **InstanceResponseTime** - 로드 밸런서에서 요청을 보낸 후 대상 인스턴스로부터 응답 신호를 받을 때까지의 경과 시간(초).

  Statistics: 가장 유용한 통계는 Average입니다.

- **RejectedConnectionCount** - 로드 밸런서가 최대 연결 수에 도달하여 거부된 연결 수

  Statistics: 가장 유용한 통계는 Sum입니다.

- **RequestCount** - IPv4를 통해 처리된 요청 수. 로드 밸런서의 대상 인스턴스에서 응답을 생성한 요청만 이 개수에 포함됩니다.

  Statistics: 가장 유용한 통계는 Sum입니다. Minimum, Maximum 및 Average는 모두 1을 반환합니다.

주제

- Lightsail 로드 밸런서 상태 확인 구성

Lightsail 로드 밸런서 상태 확인 구성

기본적으로 Lightsail은 웹 애플리케이션의 루트("/")에서 인스턴스 상태 확인을 수행합니다. 상태 확인은 등록된 인스턴스의 상태를 모니터링하는 데 사용되며, 로드 밸런서는 이를 통해 정상적인 인스턴스에만 요청을 보낼 수 있습니다. 상태 확인은 인스턴스를 로드 밸런서에 연결하면 바로 시작됩니다.

다음 상태 중 하나가 반환됩니다.

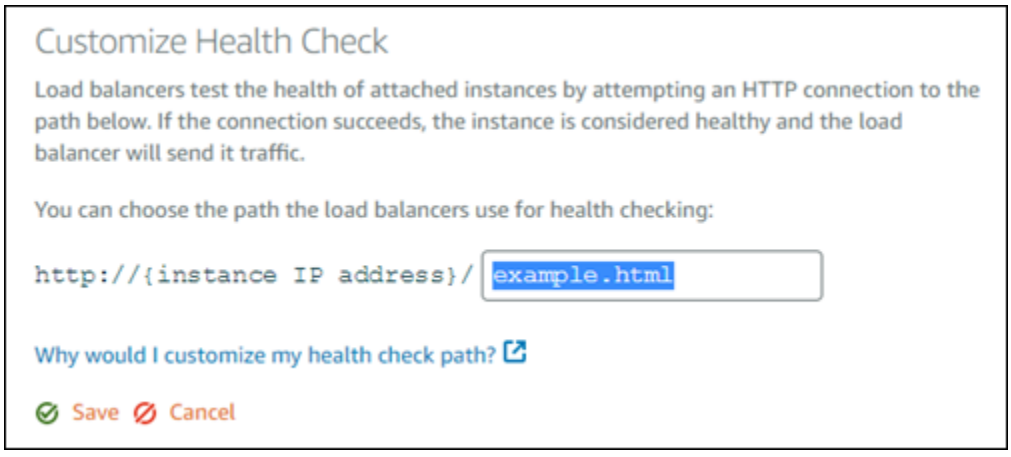
- 통과
- 실패

상태 확인이 실패하는 경우 AWS Command Line Interface 또는 Lightsail API를 사용하여 무엇이 잘못되었는지 알아볼 수 있습니다. 자세한 내용은 문제 해결 안내서를 참조하십시오.

상태 확인 경로 사용자 지정

상태 확인 경로를 사용자 지정해야 하는 경우가 있습니다. 예를 들어, 홈 페이지가 느리게 로드되거나 홈 페이지에 이미지가 많다면 더 빨리 로드되는 다른 페이지를 확인하도록 Lightsail을 구성할 수 있습니다.

1. 왼쪽 탐색 창에서 네트워킹을 선택합니다.
2. 관리할 로드 밸런서를 선택합니다.
3. 대상 인스턴스 탭에서 상태 검사 사용자 지정을 선택합니다.
4. 유효한 상태 확인 경로를 입력한 후 저장을 선택합니다.



Lightsail 로드 밸런서에서 인스턴스 분리

Amazon Lightsail 로드 밸런서에 더 이상 인스턴스를 연결해 둘 필요가 없다면 분리할 수 있습니다. 로드 밸런서에서 Lightsail 인스턴스를 분리하면 지정된 인스턴스가 더 이상 필요하지 않을 때까지 기다린 후 분리됩니다.

1. 왼쪽 탐색 창에서 네트워킹을 선택합니다.
2. 관리할 로드 밸런서를 선택합니다.
3. 대상 인스턴스 탭에서 분리할 로드 밸런서 옆에 있는 분리를 선택합니다.

Lightsail 로드 밸런서 삭제

더 이상 필요 없는 Lightsail 로드 밸런서를 삭제할 수 있습니다. 로드 밸런서를 삭제하면 연결되어 있는 Lightsail 인스턴스도 모두 분리되지만, Lightsail 인스턴스 자체는 삭제되지 않습니다. SSL/TLS 인증서를 사용하여 암호화된(HTTPS) 트래픽을 활성화한 경우, 로드 밸런서를 삭제하면 그 로드 밸런서에 연결된 SSL/TLS 인증서도 모두 영구적으로 삭제됩니다.

> **Important**
>
> Lightsail 로드 밸런서 및 연결된 인증서 삭제는 최종적이며 취소할 수 없는 작업입니다.

1. 왼쪽 탐색 창에서 네트워킹을 선택합니다.
2. 삭제할 로드 밸런서를 선택합니다.
3. 삭제를 선택합니다.
4. 로드 밸런서 삭제를 선택합니다.
5. 예, 삭제(Yes, delete)를 선택합니다.

Lightsail 콘텐츠 전송 배포를 통해 전 세계적으로 웹 콘텐츠 제공

Lightsail 배포는 엣지 로케이션이라고도 하는 전 세계에 분산된 서버 네트워크를 사용하여 사용자에게 콘텐츠를 더 빠르게 전달합니다. 배포를 사용하려면 먼저 웹 사이트나 웹 애플리케이션을 Lightsail 인스턴스 또는 컨테이너 서비스 또는 Lightsail 로드 밸런서에 연결된 여러 인스턴스에서 생성하고 호스팅하거나 정적 콘텐츠를 Lightsail 버킷에 저장해야 합니다. 그런 다음 인스턴스, 컨테이너 서비스, 로드 밸런서 또는 버킷에서 콘텐츠를 가져오고 캐싱하고 제공할 Lightsail 배포를 생성하고 구성합니다. 배포의 오리진이라고도 하는 인스턴스, 컨테이너 서비스, 로드 밸런서 또는 버킷은 콘텐츠의 최종 소스입니다.

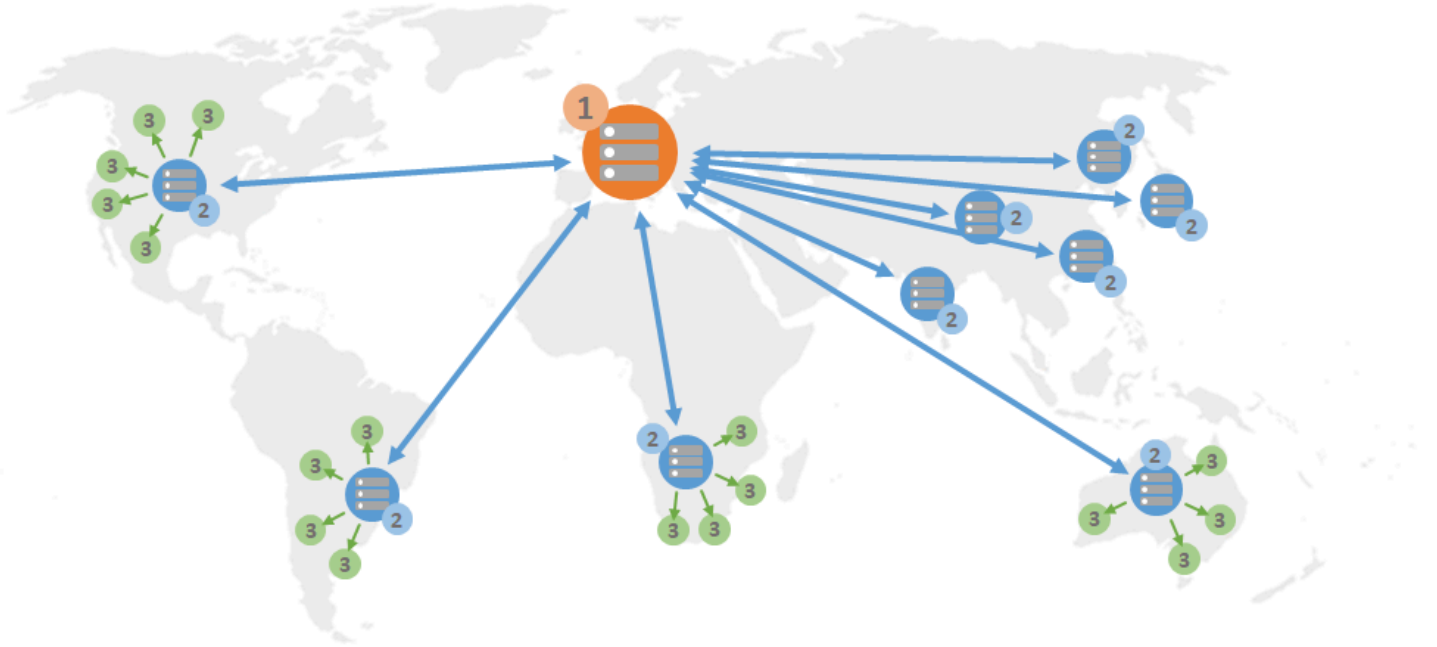
사용자가 배포를 통해 제공되는 웹 사이트를 방문하여 콘텐츠를 요청하면 요청이 지연 시간을 기준으로 가장 가까운 위치로 라우팅됩니다. 그런 다음 배포에서 다음 작업 중 하나를 수행합니다.

- 콘텐츠가 이미 엣지 로케이션에 캐싱되어 있는 경우 배포는 콘텐츠를 즉시 사용자에게 제공합니다.
- 콘텐츠가 아직 해당 엣지 로케이션에 캐싱되지 않은 경우 배포는 지정된 오리진에서 콘텐츠를 검색하고 캐싱하여 사용자에게 제공합니다.

콘텐츠는 배포에 대해 지정한 캐시 수명 기간(유지 시간(TTL)) 동안 엣지 로케이션에서 캐싱되므로, 동일한 위치의 다른 요청이 즉시 수행됩니다. 캐싱된 콘텐츠가 캐시 수명에 도달하면 엣지 로케이션에서 지워집니다. 배포는 콘텐츠 요청이 다음에 엣지 로케이션으로 라우팅될 때 콘텐츠를 검색하고 캐싱하여 제공합니다.

아래 다이어그램은 다음과 같이 해석할 수 있습니다.

- 1은 웹 사이트를 호스팅하는 Lightsail 인스턴스 또는 컨테이너 서비스, 인스턴스가 연결된 로드 밸런서 또는 정적 콘텐츠를 호스팅하는 버킷과 같이 배포의 오리진을 나타냅니다.
- 2는 오리진에서 콘텐츠를 가져오고 캐싱하여 제공하는 엣지 로케이션 또는 배포를 나타냅니다.
- 3은 엣지 로케이션에서 콘텐츠를 제공하는 대상인 사용자를 나타냅니다.

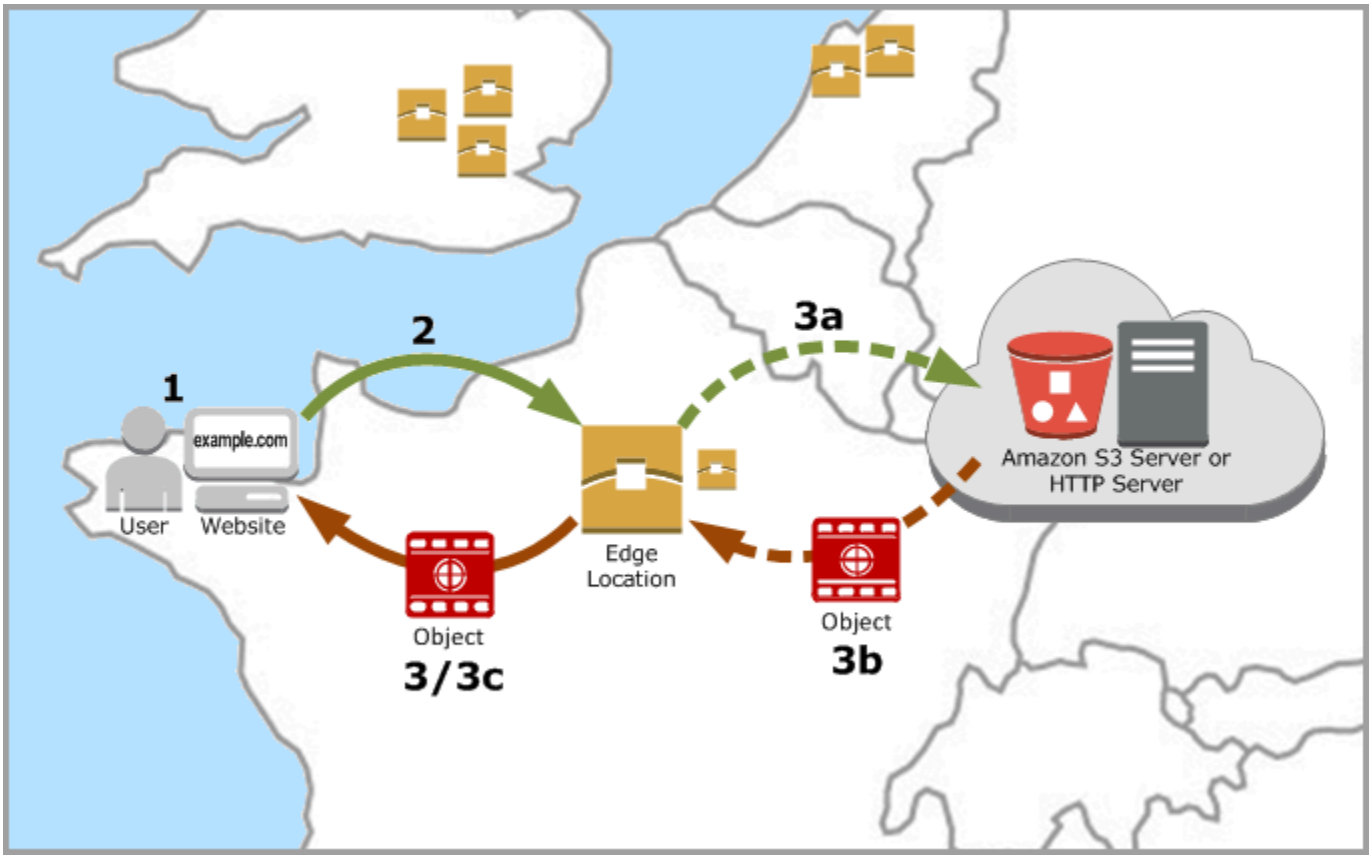


> **Note**
>
> 이 다이어그램은 설명용이며 실제 엣지 로케이션을 반영하지는 않습니다. 엣지 로케이션에 대한 자세한 내용은 가이드 후반부에 나와 있는 엣지 로케이션 및 IP 주소 범위를 참조하세요.

예를 들어 웹 사이트가 프랑스에서 호스팅되고 프랑스의 다른 지역에 있는 사람이 콘텐츠를 보려는 경우 페이지가 밀리초 단위로 로드됩니다.

방문자가 근처에 있지 않으면 상황이 조금 어려워집니다.

호주에 있는 사람이 콘텐츠를 보려는 경우, 브라우저는 프랑스에 있는 서버에서 콘텐츠를 가져온 다음 수천 마일 떨어진 해당 사용자에게 보여줘야 합니다. 다른 국가의 사용자가 동시에 동일한 콘텐츠를 요청하는 경우, 서버가 요청청으로 인해 느려지고 콘텐츠를 로드하고 제공하는 데 시간이 오래 걸립니다. 이는 최종 사용자에게 콘텐츠가 로드되는 속도에 영향을 줍니다.



CDN은 웹 사이트 콘텐츠를 엣지 로케이션에 캐싱하여 이러한 상황을 해결합니다. 이러한 콘텐츠 제공 방식은 중앙 리소스 한 곳에서 콘텐츠를 제공하는 기존의 방법보다 빠르고 효율적입니다. 최종 사용자가 웹 사이트나 애플리케이션을 통해 요청하면, DNS는 이 요청을 사용자의 요청을 가장 잘 처리할 수 있는 위치로 라우팅합니다. 모든 사용자가 멀리 떨어진 하나의 중앙 리소스에 액세스하는 것이 아니라, 사용자가 가까운 위치에서 콘텐츠에 액세스할 수 있습니다.

사용 사례

빠르고 안전한 웹 사이트 제공

Lightsail 배포는 웹 사이트 페이지, 이미지, 스타일 시트, JavaScript 등 콘텐츠를 세계 곳곳에 있는 최종 사용자에게 전달하는 속도를 향상시킵니다. 배포를 사용하면 AWS 백본 네트워크와 엣지 서버의 장점을 활용하여 해당 웹 사이트를 방문하는 최종 사용자에게 빠르고 안전하며 신뢰할 수 있는 환경을 제공할 수 있습니다.

사이트 보안 향상

암호화 처리 작어을 배포로 오프로드하여 오리진의 부하를 줄여주는 TLS 종료 기능을 활용하여 웹 사이트를 강화하고 성능을 향상할 수 있습니다. 등록된 도메인 이름과 Lightsail SSL/TLS 인증서를

함께 사용하여 배포에 Hypertext Transfer Protocol Secure(HTTPS)를 사용할 수 있습니다. 사용자는 배포에 암호화된 HTTPS 연결을 설정하고, 배포는 HTTP를 사용하여 오리진에서 콘텐츠를 가져오게 됩니다.

애플리케이션 최적화

WordPress 및 정적 웹 사이트를 비롯한 다양한 애플리케이션에 맞게 배포를 쉽게 최적화할 수 있습니다. 배포를 사용하여 콘텐츠를 캐싱하고 제공하면 대부분의 요청이 인스턴스, 컨테이너 서비스, 로드 밸런서 또는 버킷이 아닌 배포에서 처리되기 때문에 오리진이 받는 부하도 줄어듭니다.

배포 구성

다음은 Lightsail 인스턴스와 배포를 사용하여 웹 사이트 또는 웹 애플리케이션을 제공하기 위해 따라야 하는 일반적인 단계입니다.

1. 배포에 인스턴스, 컨테이너 서비스 또는 버킷을 사용할지에 따라 다음 중 하나를 완료합니다.
   - 콘텐츠를 호스스팅할 Lightsail 인스턴스를 생성합니다. 인스턴스는 배포의 오리진 역할을 합니다. 이 오리진은 콘텐츠의 최종 원본 버전을 저장합니다. 자세한 내용은 인스턴스 생성을 참조하세요.

     Lightsail 고정 IP를 인스턴스에 연결합니다. 인스턴스를 중지했다가 시작하면 인스턴스의 기본 퍼블릭 IP 주소가 변경되어 배포와 오리진 인스턴스 간의 연결이 끊어집니다. 인스턴스를 중지했다가 시작해도 고정 IP는 변경되지 않습니다. 자세한 내용은 고정 IP를 생성하여 인스턴스에 연결을 참조하세요.

     인스턴스에 콘텐츠와 파일을 업로드합니다. 객체라고도 하는 파일은 일반적으로 웹 페이지, 이미지 및 미디어 파일을 포함하지만 HTTP를 통해 제공될 수 있는 모든 항목이 될 수 있습니다.
   - 웹 사이트 또는 웹 애플리케이션을 호스팅할 Lightsail 컨테이너 서비스를 생성합니다. 컨테이너 서비스는 배포의 오리진 역할을 합니다. 이 오리진은 콘텐츠의 최종 원본 버전을 저장합니다. 자세한 내용은 Amazon Lightsail 컨테이너 서비스 생성을 참조하세요.
   - 정적 콘텐츠를 저장할 Lightsail 버킷을 생성합니다. 버킷은 배포의 오리진 역할을 합니다. 이 오리진은 콘텐츠의 최종 원본 버전을 저장합니다. 자세한 내용은 버킷 생성을 참조하세요.

     Lightsail 콘솔, AWS Command Line Interface (AWS CLI) 및 AWS APIs. 파일 업로드에 대한 자세한 내용은 버킷으로 파일 업로드를 참조하세요.
2. (선택 사항) 인스턴스에서 호스팅되는 웹 사이트에서 내결함성을 필요로 하는 경우 Lightsail 로드 밸런서를 생성합니다. 그런 다음 여러 인스턴스 복사본을 로드 밸런서에 연결합니다. 인스턴스를 오리진으로 구성하는 대신, 하나 이상의 인스턴스가 연결된 상태인 로드 밸런서를 배포의 오리진으로 구성할 수 있습니다. 자세한 내용은 로드 밸런스를 생성하여 인스턴스 연결을 참조하세요.

3. Lightsail 배포를 생성하고 인스턴스, 컨테이너 서비스, 로드 밸런서 또는 버킷을 오리진으로 구성합니다. 이와 동시에 콘텐츠의 캐시 수명, 캐싱된 웹 사이트 또는 웹 애플리케이션 요소와 같은 세부 정보를 지정합니다. 자세한 내용은 배포 생성을 참조하세요.
4. (선택 사항) 배포의 오리진이 WordPress 인스턴스인 경우 WordPress 웹 사이트를 배포와 함께 사용할 수 있도록 인스턴스의 WordPress 구성 파일을 편집해야 합니다. 자세한 내용은 배포와 함께 사용할 수 있도록 WordPress 인스턴스 구성을 참조하세요.
5. (선택 사항) Lightsail 콘솔에서 도메인의 DNS를 관리할 Lightsail DNS 영역을 생성합니다. 이렇게 하면 도메인을 Lightsail 리소스에 쉽게 매핑할 수 있습니다. 자세한 내용은 DNS 영역을 생성하여 도메인의 DNS 레코드 관리를 참조하세요. 아니면 현재 호스팅 중인 도메인의 DNS를 계속 호스팅할 수도 있습니다.
6. 배포와 함께 사용할 도메인의 Lightsail SSL/TLS 인증서를 생성합니다. Lightsail 배포에는 HTTPS가 필요하므로, 배포에 사용하려면 먼저 도메인에 대한 SSL/TLS 인증서를 요청해야 합니다. 자세한 내용은 배포용 SSL/TLS 인증서 생성을 참조하세요.
7. 등록된 도메인 이름을 배포에 사용하려면 배포에 사용자 지정 도메인을 사용하도록 설정합니다. 도메인에 대해 생성한 Lightsail SSL/TLS 인증서를 지정해서 사용자 지정 도메인을 사용할 수 있습니다. 이렇게 하면 도메인이 배포에 추가되고 HTTPS가 사용됩니다. 자세한 내용은 배포용 사용자 지정 도메인 사용을 참조하세요.
8. 도메인의 트래픽을 해당 배포로 라우팅하려면 별칭 레코드를 도메인의 DNS에 추가합니다. 별칭 레코드를 추가하면 도메인을 방문한 사용자가 배포를 통해 라우팅됩니다. 자세한 내용은 배포로 도메인 연결을 참조하세요.
9. 배포가 콘텐츠를 캐싱하고 있는지 테스트합니다. 자세한 내용은 배포 테스트를 참조하세요.

엣지 로케이션 및 IP 주소 범위

Lightsail 배포에서는 Amazon CloudFront와 동일한 엣지 서버 및 IP 주소 범위를 사용합니다. CloudFront 엣지 서버의 위치 목록은 Amazon CloudFront 제품 세부 정보 페이지를 참조하세요. CloudFront IP 범위 목록은 CloudFront 전역 IP 목록을 참조하세요.

Lightsail 콘텐츠 전송 네트워크 배포 생성

이 가이드에서는 Lightsail 콘솔을 사용하여 Amazon Lightsail 배포를 생성하는 방법을 안내하고 구성할 수 있는 배포 설정에 대해 설명합니다. 배포에 대한 자세한 내용은 콘텐츠 전송 네트워크 배포를 참조하세요.

목차

- 사전 조건
- 오리진 리소스
- 오리진 프로토콜 정책
- 캐싱 동작 및 캐싱 사전 설정
- WordPress 캐싱 사전 설정 모범 사례
- 기본 동작
- 디렉터리 및 파일 재정의
- 고급 캐시 설정
- 배포 플랜
- 배포 생성
- 다음 단계

사전 조건

배포를 생성하기 전에 다음 필수 사전 조건을 완료하세요.

1. 배포에 인스턴스, 컨테이너 서비스 또는 버킷을 사용할지에 따라 다음 중 하나를 완료합니다.
   - 콘텐츠를 호스스팅할 Lightsail 인스턴스를 생성합니다. 인스턴스는 배포의 오리진 역할을 합니다. 이 오리진은 콘텐츠의 최종 원본 버전을 저장합니다. 자세한 내용은 인스턴스 생성을 참조하세요.

     Lightsail 고정 IP를 인스턴스에 연결합니다. 인스턴스를 중지했다가 시작하면 인스턴스의 기본 퍼블릭 IP 주소가 변경되어 배포와 오리진 인스턴스 간의 연결이 끊어집니다. 인스턴스를 중지했다가 시작해도 고정 IP는 변경되지 않습니다. 자세한 내용은 고정 IP를 생성하여 인스턴스에 연결을 참조하세요.

     인스턴스에 콘텐츠와 파일을 업로드합니다. 객체라고도 하는 파일은 일반적으로 웹 페이지, 이미지 및 미디어 파일을 포함하지만 HTTP를 통해 제공될 수 있는 모든 항목이 될 수 있습니다.
   - 웹 사이트 또는 웹 애플리케이션을 호스팅할 Lightsail 컨테이너 서비스를 생성합니다. 컨테이너 서비스는 배포의 오리진 역할을 합니다. 이 오리진은 콘텐츠의 최종 원본 버전을 저장합니다. 자세한 내용은 Amazon Lightsail 컨테이너 서비스 생성을 참조하세요.
   - 정적 콘텐츠를 저장할 Lightsail 버킷을 생성합니다. 버킷은 배포의 오리진 역할을 합니다. 이 오리진은 콘텐츠의 최종 원본 버전을 저장합니다. 자세한 내용은 버킷 생성을 참조하세요.

     Lightsail 콘솔, AWS Command Line Interface (AWS CLI) 및 AWS APIs. 파일 업로드에 대한 자세한 내용은 버킷으로 파일 업로드를 참조하세요.

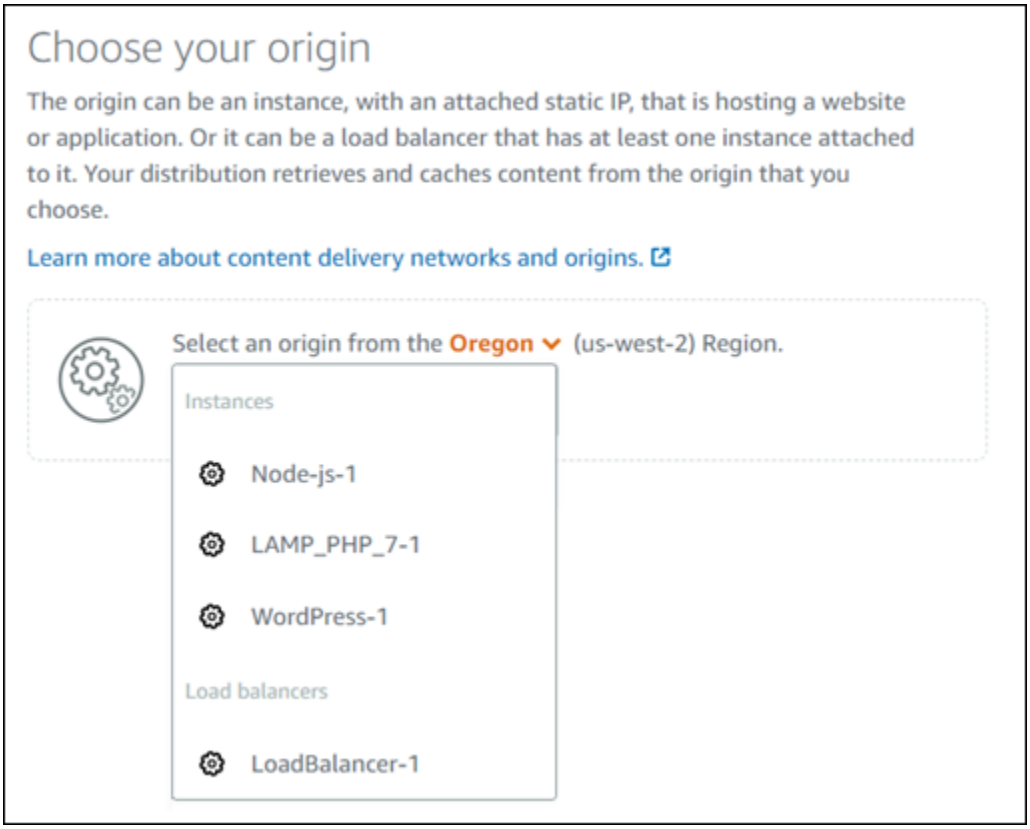
2. (선택 사항) 웹 사이트에서 내결함성을 필요로 하는 경우 Lightsail 로드 밸런서를 생성합니다. 그런 다음 여러 인스턴스 복사본을 로드 밸런서에 연결합니다. 인스턴스를 오리진으로 구성하는 대신, 하나 이상의 인스턴스가 연결된 상태인 로드 밸런서를 배포의 오리진으로 구성할 수 있습니다. 자세한 내용은 로드 밸런스를 생성하여 인스턴스 연결을 참조하세요.

오리진 리소스

오리진은 배포의 최종 콘텐츠 소스입니다. 배포를 생성할 때 웹 사이트나 웹 애플리케이션의 콘텐츠를 호스팅하는 Lightsail 인스턴스, 컨테이너 서비스, 버킷 또는 하나 이상의 인스턴스가 연결된 로드 밸런서를 선택할 수 있습니다.

> **Note**
>
> IPv6 전용 인스턴스는 현재 Lightsail 콘텐츠 전송 네트워크(CDN) 배포의 오리진으로 구성할 수 없습니다.

배포당 하나의 오리진만 선택할 수 있습니다. 배포를 생성하고 나서 언제든지 오리진을 변경할 수 있습니다. 자세한 내용은 배포의 오리진 변경을 참조하세요.



오리진 프로토콜 정책

오리진 프로토콜 정책은 배포에서 오리진으로부터 콘텐츠를 가져올 때 사용하는 프로토콜 정책입니다. 배포의 오리진을 선택한 후 오리진에서 콘텐츠를 가져올 때 배포에서 Hypertext Transfer Protocol(HTTP) 또는 Hypertext Transfer Protocol Secure(HTTPS)를 사용해야 하는지 정해야 합니다. 오리진이 HTTPS에 대해 구성되지 않은 경우 HTTP를 사용해야 합니다.

배포에 대해 다음 오리진 프로토콜 정책 중 하나를 선택할 수 있습니다.

- HTTP 전용(HTTP Only) - 배포에서 HTTP만 사용하여 오리진에 액세스합니다. 기본 설정입니다.
- HTTPS 전용(HTTPS Only) - 배포에서 HTTPS만 사용하여 오리진에 액세스합니다.

오리진 프로토콜 정책을 편집하는 단계는 이 가이드 뒷부분의 배포 생성(Create a distribution) 섹션에 나와 있습니다.

> **Note**
>
> Lightsail 버킷을 배포의 오리진으로 선택하면 오리진 프로토콜 정책이 기본적으로 HTTPS 전용(HTTPS Only)으로 설정됩니다. 버킷이 배포의 오리진인 경우에는 오리진 프로토콜 정책을 변경할 수 없습니다.

캐싱 동작 및 캐싱 사전 설정

캐싱 사전 설정은 오리진에서 호스팅하는 콘텐츠 유형에 대한 배포 설정을 자동으로 구성합니다. 예를 들어, 정적 콘텐츠에 가장 적합한 사전 설정을 선택하면 정적 웹 사이트에서 가장 효과적인 설정으로 배포를 자동으로 구성합니다. 웹 사이트가 WordPress 인스턴스에서 호스팅되는 경우 WordPress에 가장 적합한 사전 설정을 선택하여 WordPress 웹 사이트에서 배포를 사용할 수 있도록 자동으로 구성하면 됩니다.

> **Note**
>
> 배포의 오리진으로 Lightsail 버킷을 선택하는 경우 캐싱 사전 설정 옵션을 사용할 수 없습니다. 버킷에 저장되는 정적 콘텐츠에 가장 적합한 배포 설정을 자동으로 적용합니다.

배포에 대해 다음 캐싱 사전 설정 중 하나를 선택할 수 있습니다.

- 정적 콘텐츠에 가장 적합 - 이 사전 설정은 모든 항목을 캐싱하도록 배포를 구성합니다. 이 사전 설정은 오리진에서 정적 콘텐츠(예: 정적 HTML 페이지)를 호스팅하거나 웹 사이트를 방문하는 각 사용자에 대해 변함없는 콘텐츠를 호스팅하는 경우에 사용하면 좋습니다. 이 사전 설정을 선택하면 배포의 모든 콘텐츠가 캐싱됩니다.
- 동적 콘텐츠에 가장 적합 - 이 사전 설정은 배포 생성(Create a distribution) 페이지의 디렉터리 및 파일 재정의(Directory and file overrides) 섹션에서 캐시로 지정한 파일을 제외한 다른 파일을 캐싱하지 않도록 배포를 구성합니다. 자세한 내용은 가이드 후반부에 나와 있는 디렉터리 및 파일 재정의를 참조하세요. 이 사전 설정은 오리진에서 동적 콘텐츠를 호스팅하거나, 웹 사이트에 방문하거나 웹 애플리케이션을 이용하는 각 사용자에게 맞춰 변경될 수 있는 콘텐츠를 호스팅하는 경우에 적합합니다.
- WordPress에 가장 적합 - 이 사전 설정은 WordPress 인스턴스의 `wp-includes/` 및 `wp-content/` 디렉터리에 있는 파일 외에는 어떤 것도 캐싱하지 않도록 배포를 구성합니다. 이 사전 설정은 오리진이 Bitnami 및 Automattic 인증 WordPress 블루프린트(Multisite 블루프린트 제외)를 사용하는 인스턴스인 경우 유용합니다. 이 사전 설정에 대한 자세한 내용은 WordPress 캐싱 사전 설정 모범 사례를 참조하세요.

> **Note**
>
> 이 사용자 지정 설정 사전 설정은 선택할 수 없습니다. 이 설정은 사전 설정을 선택했으나 추후 배포 설정을 수동으로 수정하는 경우 자동으로 선택됩니다.

Lightsail 콘솔에서만 캐싱 사전 설정을 지정할 수 있으며, Lightsail API AWS CLI, 및 SDKs를 사용하여 지정할 수 없습니다.

WordPress 캐싱 사전 설정 모범 사례

배포 오리진으로 Bitnami 및 Automattic 인증 WordPress 블루프린트를 사용하는 인스턴스를 선택하면 Lightsail에서 배포에 WordPress에 가장 적합한 캐싱 사전 설정을 적용할지 묻는 메시지가 표시됩니다. 해당 사전 설정을 적용하면 배포가 WordPress 웹 사이트에서 가장 효과적으로 작동하도록 자동으로 구성됩니다. 다른 배포 설정을 적용하지 않아도 됩니다. WordPress 웹 사이트의 `wp-includes/` 및 `wp-content/` 디렉터리에 있는 파일 외에는 캐싱하지 않는 WordPress에 가장 적합한 사전 설정. 또한, 매일 캐시를 지우고(캐시 수명 1일), 모든 HTTP 메서드를 허용하고, `Host` 헤더만 전달하고, 쿠키를 전달하지 않고, 모든 쿼리 문자열을 전달하도록 배포를 구성합니다.

> **Important**
>
> WordPress 웹 사이트를 배포와 함께 사용할 수 있도록 하려면 인스턴스에서 WordPress 구성 파일을 편집해야 합니다. 자세한 내용은 배포와 함께 사용할 수 있도록 WordPress 인스턴스 구성을 참조하세요.

기본 동작

기본 동작은 배포에서 콘텐츠 캐싱을 처리하는 방법을 좌우합니다. 선택한 캐싱 사전 설정에 따라 배포의 기본 동작이 자동으로 지정됩니다. 다른 기본 동작을 선택하면 캐싱 사전 설정이 자동으로 사용자 지정 설정으로 변경됩니다.

> **Note**
>
> 배포의 오리진으로 Lightsail 버킷을 선택하는 경우 기본 동작 옵션을 사용할 수 없습니다. 버킷에 저장되는 정적 콘텐츠에 가장 적합한 배포 설정을 자동으로 적용합니다.

배포에 대해 다음 기본 동작 중 하나를 선택할 수 있습니다.

- 모든 항목 캐싱 - 이 동작은 전체 웹 사이트를 캐싱하고 정적 콘텐츠로 제공하도록 배포를 구성합니다. 이 옵션은 오리진이 보는 사람에 따라 변경되지 않는 콘텐츠를 호스팅하거나 웹 사이트에서 콘텐츠를 개인화하는 데 쿠키, 헤더 또는 쿼리 문자열을 활용하지 않는 경우에 사용하면 좋습니다.
- 캐싱하지 않음 - 이 동작은 지정한 오리진 파일과 폴더 경로만 캐싱하도록 배포를 구성합니다. 이 옵션은 웹 사이트 또는 웹 응용 프로그램에서 쿠키, 헤더 및 쿼리 문자열을 활용하여 개별 사용자에게 맞춰 콘텐츠를 개인화하는 경우에 적합합니다. 이 옵션을 선택하는 경우 캐시에 디렉터리 및 파일 경로 재정의를 지정해야 합니다.

디렉터리 및 파일 재정의

디렉터리 및 파일 재정의는 선택한 기본 동작을 재정의하거나 예외를 추가하는 데 사용할 수 있습니다. 예를 들어, 모든 항목을 캐싱하도록 선택한 경우 재정의를 통해 배포에서 캐싱하지 않아야 하는 디렉터리, 파일 또는 파일 유형을 지정할 수 있습니다. 반대로 어떤 것도 캐싱하지 않도록 선택한 경우 재정의를 통해 배포에서 캐싱해야 하는 디렉터리, 파일 또는 파일 유형을 지정하면 됩니다.

페이지의 디렉터리 및 파일 재정의(Directory and file overrides) 섹션에서 캐싱하거나 캐싱하지 않을 디렉터리 또는 파일 경로를 지정할 수 있습니다. 별표 기호를 사용하여 와일드카드 디렉터리(`path/`

to/assets/*) 및 파일 유형(*.html, *jpg, *js)을 지정합니다. 디렉터리와 파일 경로는 대/소문자를 구분합니다.

> **Note**
>
> 배포의 오리진으로 Lightsail 버킷을 선택하는 경우 디렉터리 및 파일 재정의 옵션을 사용할 수 없습니다. 선택한 버킷에 저장된 모든 항목이 캐싱됩니다.

다음은 디렉터리 및 파일 재정의를 지정하는 몇 가지 방법입니다.

- Lightsail 인스턴스에서 실행 중인 Apache 웹 서버의 문서 루트에 있는 모든 파일을 캐싱하려면 다음을 지정하세요.

  ```
  var/www/html/
  ```

- Apache 웹 서버의 문서 루트에 있는 인덱스 페이지만 캐싱하려면 다음 파일을 지정하세요.

  ```
  var/www/html/index.html
  ```

- Apache 웹 서버의 문서 루트에 있는 .html 파일만 캐싱하려면 다음을 지정하세요.

  ```
  var/www/html/*.html
  ```

- Apache 웹 서버 문서 루트의 이미지 하위 디렉터리에 있는 .jpg, .png 및 .gif 파일만 캐싱하려면 다음을 지정하세요.

  ```
  var/www/html/images/*.jpg
  ```

  ```
  var/www/html/images/*.png
  ```

  ```
  var/www/html/images/*.gif
  ```

  Apache 웹 서버 문서 루트의 이미지 하위 디렉터리에 있는 모든 파일을 캐싱하려면 다음을 지정하세요.

  ```
  var/www/html/images/
  ```

고급 캐시 설정

고급 설정을 사용하여 배포에 있는 콘텐츠의 캐시 수명, 허용되는 HTTP 메서드, HTTP 헤더 전달, 쿠키 전달 및 쿼리 문자열 전달을 지정할 수 있습니다. 지정한 고급 설정은 캐시로 지정한 디렉터리 및 파일을 재정의하는 등 배포가 캐싱하는 디렉터리 및 파일에만 적용됩니다.

> **Note**
>
> 배포의 오리진으로 Lightsail 버킷을 선택하면 배포 생성(Create distribution) 페이지에서 고급 캐시 설정을 사용할 수 없습니다. 버킷에 저장되는 정적 콘텐츠에 가장 적합한 배포 설정을 자동으로 적용합니다. 단, 배포 생성 후 배포 관리 페이지에서 고급 캐시 설정을 수정할 수 있습니다.

다음과 같은 고급 설정을 구성할 수 있습니다.

캐시 수명(TTL)

배포가 오리진으로 다른 요청을 전달하여 콘텐츠가 업데이트되었는지 확인하기 전에 콘텐츠가 배포의 캐시에 남아 있는 시간을 제어합니다. 기본값은 1일입니다. 이 기간을 줄여 보다 효과적인 동적 콘텐츠를 제공할 수 있습니다. 이 기간이 늘어나면 파일이 엣지 로케이션에서 바로 제공될 가능성이 크므로 사용자에게 제공되는 성능이 향상됩니다. 또한, 기간을 늘리면 배포에서 콘텐츠를 가져오는 빈도가 줄어들어 오리진에 가해지는 부하도 줄어듭니다.

> **Note**
>
> 지정한 캐시 수명 값은 오리진이 콘텐츠에 `Cache-Control max-age`, `Cache-Control s-maxage` 및 `Expires`와 같은 HTTP 헤더를 추가하지 않을 경우에만 적용됩니다.

허용되는 HTTP 메서드

배포가 처리하고 오리진으로 전달하는 HTTP 메서드를 제어합니다. HTTP 메서드는 오리진에서 수행되기를 바라는 작업을 나타냅니다. 예를 들어, GET 메서드는 오리진에서 데이터를 검색하고 PUT 메서드는 함께 제공되는 엔터티를 오리진에 저장하도록 요청합니다.

배포에 대해 다음 HTTP 메서드 옵션 중 하나를 선택할 수 있습니다.

- GET, HEAD, OPTIONS, PUT, PATCH, POST 및 DELETE 메서드 허용

- GET, HEAD 및 OPTIONS 메서드 허용
- GET 및 HEAD 메서드 허용

배포는 항상 GET 및 HEAD 요청에 대한 응답을 캐싱합니다. 또한, 이러한 요청을 허용하도록 선택한 경우 배포에서는 OPTIONS 요청에 대한 응답도 캐싱합니다. 배포는 다른 HTTP 메서드에 대한 응답을 캐싱하지 않습니다. 자세한 내용은 HTTP 메서드를 참조하세요.

> **Important**
>
> 지원되는 모든 HTTP 메서드를 허용하도록 배포를 구성할 경우 모든 메서드를 처리하도록 오리진 인스턴스를 구성해야 합니다. 예를 들어, POST를 사용하기 위해 이러한 메서드를 허용하도록 배포를 구성하는 경우, 최종 사용자가 원치 않는 리소스를 삭제할 수 없도록 오리진 서버가 DELETE 요청을 적절히 처리할 수 있게 구성해야 합니다. 자세한 내용은 웹 사이트 또는 웹 애플리케이션 문서를 살펴보세요.

HTTP 헤더 전달

지정된 헤더 값을 기준으로 배포가 콘텐츠를 캐싱할지와 캐싱한다면 어떤 헤더 값을 캐싱할지 제어합니다. HTTP 헤더는 클라이언트 브라우저, 요청된 페이지, 오리진 등에 대한 정보를 전달합니다. 예를 들어, `Accept-Language` 헤더는 클라이언트 언어(예: 영어의 경우 `en-US`)를 전송하여 오리진이 사용 가능한 경우 클라이언트 언어로 된 내용으로 응답할 수 있도록 합니다.

배포에 대해 다음 HTTP 헤더 옵션 중 하나를 선택할 수 있습니다.

- 헤더 전달 안 함(Forward no headers)
- 지정한 헤더만 전달

헤더 전달 안 함(Forward no headers)을 선택하면 배포가 헤더 값을 기반으로 콘텐츠를 캐싱하지 않습니다. 선택한 옵션에 관계없이 배포는 구체적인 헤더를 오리진으로 전달하고 전달한 헤더를 기반으로 특정 작업을 수행합니다. 배포에서 헤더 전달을 처리하는 방법에 대한 자세한 내용은 HTTP 요청 헤더 및 배포 동작을 참조하세요.

쿠키 전달

배포에서 쿠키를 오리진에 전달할지 여부와 전달할 경우 어떤 쿠키를 전달할지 제어합니다. 쿠키에는 방문자의 이름 및 관심사와 같은 방문자가 제공한 정보뿐만 아니라 오리진의 웹 페이지에서 방문자가 보인 행동에 대한 정보처럼 오리진으로 전송된 작은 데이터 조각이 포함됩니다.

배포에 대해 다음 쿠키 전달 옵션 중 하나를 선택할 수 있습니다.

- 쿠키 전달 안 함(Don't forward cookies)
- 모든 쿠키 전달(Forward all cookies)
- 지정한 쿠키 전달(Forward cookies I specify)

모든 쿠키 전달(Forward all cookies)을 선택하는 경우 배포에서는 애플리케이션에서 사용하는 쿠키 수와 관계없이 모든 쿠키를 전달합니다. 지정한 쿠키 전달(Forward cookies I specify)을 선택한 경우 표시되는 텍스트 상자에 배포에서 전달할 쿠키의 이름을 입력합니다. 쿠키 이름을 지정할 때 다음 와일드카드를 지정할 수 있습니다.

- *는 쿠키 이름의 0개 이상의 문자에 해당합니다.
- ?는 쿠키 이름의 정확히 1문자에 해당합니다.

예를 들어, `userid_`*`member-number`*라는 이름의 쿠키가 포함된 객체에 대한 최종 사용자 요청을 가정해 봅니다. 각각의 사용자는 `member-number`(`userid_123`, `userid_124`, `userid_125` 등)에 대한 고유한 값을 갖습니다. 이 경우에는 배포에서 구성원별로 개별 콘텐츠 버전을 캐싱하려 합니다. 모든 쿠키를 오리진에 전달하여 이를 달성할 수는 있지만, 최종 사용자 요청에 배포에서 캐싱하길 원치 않는 쿠키가 일부 포함됩니다. 다음 값을 쿠키 이름으로 지정할 수 있습니다. 이렇게 하면 배포에서 `userid_`로 시작하는 모든 쿠키를 오리진으로 전달합니다. `userid_*`

쿼리 문자열 전달

배포에서 쿼리 문자열을 오리진에 전달할지 여부와 전달할 경우 어떤 쿼리 문자열을 전달할지 제어합니다. 쿼리 문자열은 지정된 파라미터에 값을 할당하는 URL의 일부입니다. 예를 들어, `https://example.com/over/there?name=ferret` URL은 `name=ferret` 쿼리 문자열을 포함합니다. 서버가 이러한 페이지에 대한 요청을 수신하면 프로그램을 실행하여 `name=ferret` 쿼리 문자열을 변경하지 않고 프로그램에 전달할 수 있습니다. 물음표는 구분자로 사용되며 쿼리 문자열의 일부를 구성하지 않습니다.

배포에서 쿼리 문자열 없이 전달하거나 지정한 쿼리 문자열만 전달하도록 선택할 수 있습니다. 오리진에서 쿼리 문자열 파라미터 값과 상관없이 동일한 콘텐츠 버전을 반환할 경우 쿼리 문자열을 전달하지 않도록 선택합니다. 이렇게 하면 배포에서 캐시로부터 요청을 제공할 수 있는 가능성을 높이고, 그에 따라 성능이 향상되고 오리진에 걸리는 부하가 줄어듭니다. 오리진 서버에서 하나 이상의 쿼리 문자열 파라미터를 기준으로 다른 콘텐츠 버전을 반환할 경우 지정한 쿼리 문자열만 전달하도록 선택합니다.

배포 플랜

배포 플랜에 따라 월별 데이터 전송 할당량 및 배포 비용이 지정됩니다. 배포가 플랜의 월별 데이터 전송 할당량보다 많은 데이터를 전송하는 경우 초과 요금이 청구됩니다. 자세한 내용은 Lightsail 요금 페이지를 참조하세요.

요금을 초과로 사용하지 않으려면 현재 배포 플랜을 배포가 월별 할당량을 초과하기 전에 매달 더 많은 데이터 전송량을 제공하는 다른 플랜으로 변경하면 됩니다. 각 AWS 결제 주기 동안 배포의 플랜을 한 번만 변경할 수 있습니다. 배포 플랜을 생성한 후 변경하는 방법은 배포의 플랜 변경을 참조하세요.

배포 생성

배포를 생성하려면 다음 절차를 완료하세요.

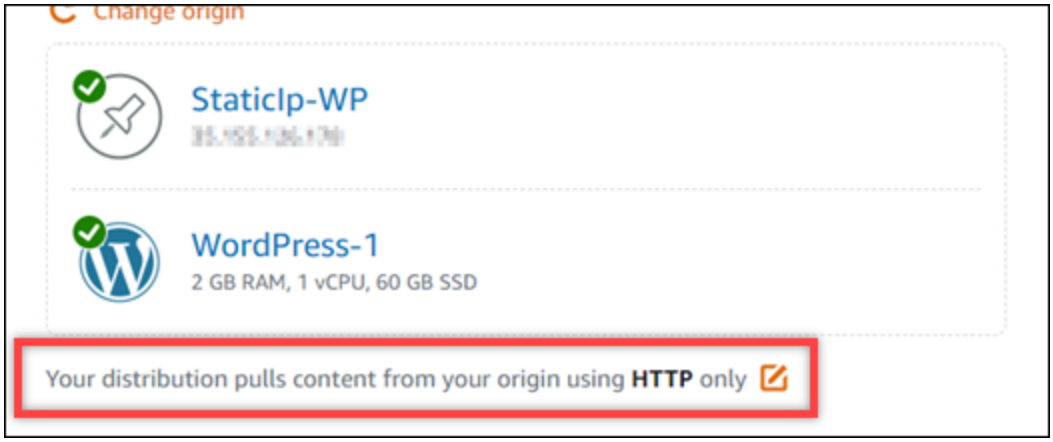
1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 네트워킹을 선택합니다.
3. 배포 생성을 선택합니다.
4. 이 페이지의 오리진 선택 섹션에서 오리진 리소스를 생성한 AWS 리전 을 선택합니다.

   배포는 전역 리소스입니다. 모든에서 오리진을 참조 AWS 리전하고 콘텐츠를 전 세계에 배포할 수 있습니다.
5. 오리진을 선택합니다. 오리진은 Lightsail 인스턴스, 컨테이너 서비스, 버킷 또는 하나 이상의 인스턴스가 연결된 로드 밸런서일 수 있습니다. 자세한 내용은 오리진 리소스를 참조하세요.

   > **Important**
   >
   > Lightsail 컨테이너 서비스를 배포의 오리진으로 선택하는 경우, Lightsail은 배포의 기본 도메인 이름을 컨테이너 서비스에 사용자 지정 도메인으로 자동으로 추가합니다. 이렇게 하면 트래픽을 배포와 컨테이너 서비스 간에 라우팅할 수 있습니다. 그러나 컨테이너 서비스에 배포의 기본 도메인 이름을 수동으로 추가해야 하는 경우도 있습니다. 자세한 내용은 배포의 기본 도메인을 컨테이너 서비스에 추가를 참조하세요.
6. (선택 사항) 오리진 프로토콜 정책을 변경하려면 배포에서 사용하는 현재 오리진 프로토콜 정책 옆에 표시된 연필 아이콘을 선택합니다. 자세한 내용은 오리진 프로토콜 정책을 참조하세요.

   이 옵션은 페이지의 오리진 선택(Choose your origin) 섹션에서 배포에 대해 선택한 오리진 리소스 아래에 나와 있습니다.

> **Note**
>
> Lightsail 버킷을 배포의 오리진으로 선택하면 오리진 프로토콜 정책이 기본적으로 HTTPS 전용(HTTPS Only)으로 설정됩니다. 버킷이 배포의 오리진인 경우에는 오리진 프로토콜 정책을 변경할 수 없습니다.



7. 배포에 대한 캐싱 동작(캐싱 사전 설정이라고도 함)을 선택합니다. 자세한 정보는 캐싱 동작 및 캐싱 사전 설정을 참조하세요.

> **Note**
>
> 배포의 오리진으로 Lightsail 버킷을 선택하는 경우 캐싱 사전 설정 옵션을 사용할 수 없습니다. 버킷에 저장되는 정적 콘텐츠에 가장 적합한 배포 설정을 자동으로 적용합니다.

8. (선택 사항) 배포에 대한 추가 캐싱 동작 설정을 보려면 모든 설정 표시를 선택합니다.

> **Note**
>
> 배포의 오리진으로 Lightsail 버킷을 선택하는 경우 캐싱 동작 설정을 사용할 수 없습니다. 버킷에 저장되는 정적 콘텐츠에 가장 적합한 배포 설정을 자동으로 적용합니다.

9. (선택 사항) 배포의 기본 동작을 선택합니다. 자세한 내용은 기본 동작을 참조하세요.

> **Note**
>
> 배포의 오리진으로 Lightsail 버킷을 선택하는 경우 기본 동작 옵션을 사용할 수 없습니다. 버킷에 저장되는 정적 콘텐츠에 가장 적합한 배포 설정을 자동으로 적용합니다.

10. (선택 사항) 경로 추가(Add path)를 선택하여 배포의 캐싱 동작에 디렉터리 및 파일 재정의를 추가합니다. 자세한 내용은 디렉터리 및 파일 재정의를 참조하세요.

> **Note**
>
> 배포의 오리진으로 Lightsail 버킷을 선택하는 경우 디렉터리 및 파일 재정의 옵션을 사용할 수 없습니다. 버킷에 저장되는 정적 콘텐츠에 가장 적합한 배포 설정을 자동으로 적용합니다.

11. (선택 사항) 배포에 대해 편집할 고급 설정 옆에 표시된 연필 아이콘을 선택합니다. 자세한 내용은 고급 캐시 설정을 참조하세요.

> **Note**
>
> 배포의 오리진으로 Lightsail 버킷을 선택하면 배포 생성(Create distribution) 페이지에서 고급 캐시 설정을 사용할 수 없습니다. 버킷에 저장되는 정적 콘텐츠에 가장 적합한 배포 설정을 자동으로 적용합니다. 단, 배포 생성 후 배포 관리 페이지에서 고급 캐시 설정을 수정할 수 있습니다.

12. 배포 플랜을 선택합니다. 자세한 내용은 배포 플랜을 참조하세요.
13. 배포의 이름을 입력합니다.

    리소스 이름:

    - Lightsail 계정 AWS 리전 의 각 내에서 고유해야 합니다.
    - 2~255자의 문자로 구성되어야 합니다.
    - 영숫자 문자 또는 숫자로 시작하고 끝나야 합니다.
    - 영숫자 문자, 숫자, 마침표, 대시, 밑줄이 포함될 수 있습니다.
14. 배포 비용을 검토합니다.
15. 배포 생성을 선택합니다.

    몇 분 정도 지나면 배포가 생성됩니다.

다음 단계

배포를 설정하고 실행한 후 다음 단계를 완료하는 것이 좋습니다.

1. 배포의 오리진이 WordPress 인스턴스인 경우 WordPress 웹 사이트를 배포와 함께 사용할 수 있도록 인스턴스의 WordPress 구성 파일을 편집해야 합니다. 자세한 내용은 배포와 함께 사용할 수 있도록 WordPress 인스턴스 구성을 참조하세요.
2. (선택 사항) Lightsail 콘솔에서 도메인의 DNS를 관리할 Lightsail DNS 영역을 생성합니다. 이렇게 하면 도메인을 Lightsail 리소스에 쉽게 매핑할 수 있습니다. 자세한 내용은 DNS 영역을 생성하여 도메인의 DNS 레코드 관리를 참조하세요. 아니면 현재 호스팅 중인 도메인의 DNS를 계속 호스팅할 수도 있습니다.
3. 배포와 함께 사용할 도메인의 Lightsail SSL/TLS 인증서를 생성합니다. Lightsail 배포에는 HTTPS가 필요하므로, 배포에 사용하려면 먼저 도메인에 대한 SSL/TLS 인증서를 요청해야 합니다. 자세한 내용은 배포용 SSL/TLS 인증서 생성을 참조하세요.
4. 도메인을 배포와 함께 사용하려면 배포에 대해 사용자 지정 도메인을 활성화하면 됩니다. 도메인에 대해 생성한 Lightsail SSL/TLS 인증서를 지정해서 사용자 지정 도메인을 활성화할 수 있습니다. 이렇게 하면 도메인이 배포에 추가되고 HTTPS가 활성화됩니다. 자세한 내용은 배포용 사용자 지정 도메인 사용을 참조하세요.
5. 도메인의 트래픽을 해당 배포로 라우팅하려면 별칭 레코드를 도메인의 DNS에 추가합니다. 별칭 레코드를 추가하면 도메인을 방문한 사용자가 배포를 통해 라우팅됩니다. 자세한 내용은 배포로 도메인 연결을 참조하세요.
6. 배포가 콘텐츠를 캐싱하고 있는지 테스트합니다. 자세한 내용은 배포 테스트를 참조하세요.

Lightsail 배포 삭제

Amazon Lightsail 배포를 더 이상 사용하지 않을 경우 언제든지 삭제할 수 있습니다.

배포 삭제

배포를 삭제하려면 다음 절차를 완료하세요.

1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 네트워킹을 선택합니다.
3. 삭제할 배포의 이름을 선택합니다.
4. 배포의 관리 페이지에서 삭제(Delete) 탭을 선택합니다.
5. 배포 삭제(Delete distribution)를 선택하여 배포를 삭제합니다.

6. 예, 삭제를 선택하여 삭제를 확인합니다.

Lightsail 배포에 대한 캐싱 구성

캐시 동작을 사용하면 Amazon Lightsail 배포에 의해 오리진에서 캐싱되거나 캐싱되지 않는 항목을 구성할 수 있습니다. 예를 들어, 오리진에서 개별 디렉터리, 파일 또는 파일 유형을 캐싱하도록 지정할 수 있습니다. 오리진에 전달되는 HTML 메서드와 헤더를 지정할 수도 있습니다. 이 가이드에서는 배포의 캐싱 동작을 변경하는 방법을 안내합니다. 배포에 대한 자세한 내용은 콘텐츠 전송 네트워크 배포를 참조하세요.

목차

- 캐싱 사전 설정
- WordPress 캐싱 사전 설정 모범 사례
- 기본 동작
- 디렉터리 및 파일 재정의
- 고급 캐시 설정
- 배포의 캐시 동작 변경

캐싱 사전 설정

캐싱 사전 설정은 오리진에서 호스팅하는 콘텐츠 유형에 대한 배포 설정을 자동으로 구성합니다. 예를 들어, 정적 콘텐츠에 가장 적합한 사전 설정을 선택하면 정적 웹 사이트에서 가장 효과적인 설정으로 배포를 자동으로 구성합니다. 웹 사이트가 WordPress 인스턴스에서 호스팅되는 경우 WordPress에 가장 적합한 사전 설정을 선택하여 WordPress 웹 사이트에서 배포를 사용할 수 있도록 자동으로 구성하면 됩니다.

배포에 대해 다음 캐싱 사전 설정 중 하나를 선택할 수 있습니다.

- 정적 콘텐츠에 가장 적합 - 이 사전 설정은 모든 항목을 캐싱하도록 배포를 구성합니다. 이 사전 설정은 오리진에서 정적 콘텐츠(예: 정적 HTML 페이지)를 호스팅하거나 웹 사이트를 방문하는 각 사용자에 대해 변함없는 콘텐츠를 호스팅하는 경우에 사용하면 좋습니다. 이 사전 설정을 선택하면 배포의 모든 콘텐츠가 캐싱됩니다.
- 동적 콘텐츠에 가장 적합 - 이 사전 설정은 배포 생성(Create a distribution) 페이지의 디렉터리 및 파일 재정의(Directory and file overrides) 섹션에서 캐시로 지정한 파일을 제외한 다른 파일을 캐싱하지 않도록 배포를 구성합니다. 자세한 내용은 가이드 후반부에 나와 있는 디렉터리 및 파일 재정의를

참조하세요. 이 사전 설정은 오리진에서 동적 콘텐츠를 호스팅하거나, 웹 사이트에 방문하거나 웹 애플리케이션을 이용하는 각 사용자에게 맞춰 변경될 수 있는 콘텐츠를 호스팅하는 경우에 적합합니다.

- WordPress에 가장 적합 - 이 사전 설정은 WordPress 인스턴스의 `wp-includes/` 및 `wp-content/` 디렉터리에 있는 파일 외에는 어떤 것도 캐싱하지 않도록 배포를 구성합니다. 이 사전 설정은 오리진이 Bitnami 및 Automattic 인증 WordPress 블루프린트(Multisite 블루프린트 제외)를 사용하는 인스턴스인 경우 유용합니다. 이 사전 설정에 대한 자세한 내용은 WordPress 캐싱 사전 설정 모범 사례를 참조하세요.

> **Note**
>
> 이 사용자 지정 설정 사전 설정은 선택할 수 없습니다. 이 설정은 사전 설정을 선택했으나 추후 배포 설정을 수동으로 수정하는 경우 자동으로 선택됩니다.

Lightsail 콘솔에서만 캐싱 사전 설정을 지정할 수 있으며, Lightsail API AWS CLI, 및 SDKs를 사용하여 지정할 수 없습니다.

WordPress 캐싱 사전 설정 모범 사례

배포 오리진으로 Bitnami 및 Automattic 인증 WordPress 블루프린트를 사용하는 인스턴스를 선택하면 Lightsail에서 배포에 WordPress에 가장 적합한 캐싱 사전 설정을 적용할지 묻는 메시지가 표시됩니다. 해당 사전 설정을 적용하면 배포가 WordPress 웹 사이트에서 가장 효과적으로 작동하도록 자동으로 구성됩니다. 다른 배포 설정을 적용하지 않아도 됩니다. WordPress 웹 사이트의 `wp-includes/` 및 `wp-content/` 디렉터리에 있는 파일 외에는 캐싱하지 않는 WordPress에 가장 적합한 사전 설정. 또한, 매일 캐시를 지우고(캐시 수명 1일), 모든 HTTP 메서드를 허용하고, `Host` 헤더만 전달하고, 쿠키를 전달하지 않고, 모든 쿼리 문자열을 전달하도록 배포를 구성합니다.

> **Important**
>
> WordPress 웹 사이트를 배포와 함께 사용할 수 있도록 하려면 인스턴스에서 WordPress 구성 파일을 편집해야 합니다. 자세한 내용은 배포와 함께 사용할 수 있도록 WordPress 인스턴스 구성을 참조하세요.

기본 동작

기본 동작은 배포에서 콘텐츠 캐싱을 처리하는 방법을 좌우합니다. 선택한 캐싱 사전 설정에 따라 배포의 기본 동작이 자동으로 지정됩니다. 다른 기본 동작을 선택하면 캐싱 사전 설정이 자동으로 사용자 지정 설정으로 변경됩니다.

배포에 대해 다음 기본 동작 중 하나를 선택할 수 있습니다.

- 모든 항목 캐싱 - 이 동작은 전체 웹 사이트를 캐싱하고 정적 콘텐츠로 제공하도록 배포를 구성합니다. 이 옵션은 오리진이 보는 사람에 따라 변경되지 않는 콘텐츠를 호스팅하거나 웹 사이트에서 콘텐츠를 개인화하는 데 쿠키, 헤더 또는 쿼리 문자열을 활용하지 않는 경우에 사용하면 좋습니다.
- 캐싱하지 않음 - 이 동작은 지정한 오리진 파일과 폴더 경로만 캐싱하도록 배포를 구성합니다. 이 옵션은 웹 사이트 또는 웹 응용 프로그램에서 쿠키, 헤더 및 쿼리 문자열을 활용하여 개별 사용자에게 맞춰 콘텐츠를 개인화하는 경우에 적합합니다. 이 옵션을 선택하는 경우 캐시에 디렉터리 및 파일 경로 재정의를 지정해야 합니다.

디렉터리 및 파일 재정의

디렉터리 및 파일 재정의는 선택한 기본 동작을 재정의하거나 예외를 추가하는 데 사용할 수 있습니다. 예를 들어, 모든 항목을 캐싱하도록 선택한 경우 재정의를 통해 배포에서 캐싱하지 않아야 하는 디렉터리, 파일 또는 파일 유형을 지정할 수 있습니다. 반대로 어떤 것도 캐싱하지 않도록 선택한 경우 재정의를 통해 배포에서 캐싱해야 하는 디렉터리, 파일 또는 파일 유형을 지정하면 됩니다.

페이지의 디렉터리 및 파일 재정의(Directory and file overrides) 섹션에서 캐싱하거나 캐싱하지 않을 디렉터리 또는 파일 경로를 지정할 수 있습니다. 별표 기호를 사용하여 와일드카드 디렉터리(`path/to/assets/*`) 및 파일 유형(`*.html`, `*jpg`, `*js`)을 지정합니다. 디렉터리와 파일 경로는 대/소문자를 구분합니다.

다음은 디렉터리 및 파일 재정의를 지정하는 몇 가지 방법입니다.

- Lightsail 인스턴스에서 실행 중인 Apache 웹 서버의 문서 루트에 있는 모든 파일을 캐싱하려면 다음을 지정하세요.

```
var/www/html/
```

- Apache 웹 서버의 문서 루트에 있는 인덱스 페이지만 캐싱하려면 다음을 지정하세요.

```
var/www/html/index.html
```

- Apache 웹 서버의 문서 루트에 있는 .html 파일만 캐싱하려면 다음을 지정하세요.

```
var/www/html/*.html
```

- Apache 웹 서버 문서 루트의 이미지 하위 디렉터리에 있는 .jpg, .png 및 .gif 파일만 캐싱하려면 다음을 지정하세요.

```
var/www/html/images/*.jpg
```

```
var/www/html/images/*.png
```

```
var/www/html/images/*.gif
```

Apache 웹 서버 문서 루트의 이미지 하위 디렉터리에 있는 모든 파일을 캐싱하려면 다음을 지정하세요.

```
var/www/html/images/
```

고급 캐시 설정

고급 설정을 사용하여 배포에 있는 콘텐츠의 캐시 수명, 허용되는 HTTP 메서드, HTTP 헤더 전달, 쿠키 전달 및 쿼리 문자열 전달을 지정할 수 있습니다. 지정한 고급 설정은 캐시로 지정한 디렉터리 및 파일을 재정의하는 등 배포가 캐싱하는 디렉터리 및 파일에만 적용됩니다.

다음과 같은 고급 설정을 구성할 수 있습니다.

캐시 수명(TTL)

배포가 오리진으로 다른 요청을 전달하여 콘텐츠가 업데이트되었는지 확인하기 전에 콘텐츠가 배포의 캐시에 남아 있는 시간을 제어합니다. 기본값은 1일입니다. 이 기간을 줄여 보다 효과적인 동적 콘텐츠를 제공할 수 있습니다. 이 기간이 늘어나면 파일이 엣지 로케이션에서 바로 제공될 가능성이 크므로 사용자에게 제공되는 성능이 향상됩니다. 또한, 기간을 늘리면 배포에서 콘텐츠를 가져오는 빈도가 줄어들어 오리진에 가해지는 부하도 줄어듭니다.

> **Note**
>
> 지정한 캐시 수명 값은 오리진이 콘텐츠에 `Cache-Control max-age`, `Cache-Control s-maxage` 및 `Expires`와 같은 HTTP 헤더를 추가하지 않을 경우에만 적용됩니다.

허용되는 HTTP 메서드

배포가 처리하고 오리진으로 전달하는 HTTP 메서드를 제어합니다. HTTP 메서드는 오리진에서 수행되기를 바라는 작업을 나타냅니다. 예를 들어, GET 메서드는 오리진에서 데이터를 검색하고 PUT 메서드는 함께 제공되는 엔터티를 오리진에 저장하도록 요청합니다.

배포에 대해 다음 HTTP 메서드 옵션 중 하나를 선택할 수 있습니다.

- GET, HEAD, OPTIONS, PUT, PATCH, POST 및 DELETE 메서드 허용
- GET, HEAD 및 OPTIONS 메서드 허용
- GET 및 HEAD 메서드 허용

배포는 항상 GET 및 HEAD 요청에 대한 응답을 캐싱합니다. 또한, 이러한 요청을 허용하도록 선택한 경우 배포에서는 OPTIONS 요청에 대한 응답도 캐싱합니다. 배포는 다른 HTTP 메서드에 대한 응답을 캐싱하지 않습니다.

> **Important**
>
> 지원되는 모든 HTTP 메서드를 허용하도록 배포를 구성할 경우 모든 메서드를 처리하도록 오리진 인스턴스를 구성해야 합니다. 예를 들어, POST를 사용하기 위해 이러한 메서드를 허용하도록 배포를 구성하는 경우, 최종 사용자가 원치 않는 리소스를 삭제할 수 없도록 오리진 서버가 DELETE 요청을 적절히 처리할 수 있게 구성해야 합니다. 자세한 내용은 웹 사이트 또는 웹 애플리케이션 문서를 살펴보세요.

HTTP 헤더 전달

지정된 헤더 값을 기준으로 배포가 콘텐츠를 캐싱할지와 캐싱한다면 어떤 헤더 값을 캐싱할지 제어합니다. HTTP 헤더는 클라이언트 브라우저, 요청된 페이지, 오리진 등에 대한 정보를 전달합니다. 예를 들어, `Accept-Language` 헤더는 클라이언트 언어(예: 영어의 경우 `en-US`)를 전송하여 오리진이 사용 가능한 경우 클라이언트 언어로 된 내용으로 응답할 수 있도록 합니다.

배포에 대해 다음 HTTP 헤더 옵션 중 하나를 선택할 수 있습니다.

- 헤더 전달 안 함(Forward no headers)
- 지정한 헤더만 전달

헤더 전달 안 함(Forward no headers)을 선택하면 배포가 헤더 값을 기반으로 콘텐츠를 캐싱하지 않습니다. 선택한 옵션에 관계없이 배포는 구체적인 헤더를 오리진으로 전달하고 전달한 헤더를 기반으로 특정 작업을 수행합니다.

쿠키 전달

배포에서 쿠키를 오리진에 전달할지 여부와 전달할 경우 어떤 쿠키를 전달할지 제어합니다. 쿠키에는 방문자의 이름 및 관심사와 같은 방문자가 제공한 정보뿐만 아니라 오리진의 웹 페이지에서 방문자가 보인 행동에 대한 정보처럼 오리진으로 전송된 작은 데이터 조각이 포함됩니다.

배포에 대해 다음 쿠키 전달 옵션 중 하나를 선택할 수 있습니다.

- 쿠키 전달 안 함(Don't forward cookies)
- 모든 쿠키 전달(Forward all cookies)
- 지정한 쿠키 전달(Forward cookies I specify)

모든 쿠키 전달(Forward all cookies)을 선택하는 경우 배포에서는 애플리케이션에서 사용하는 쿠키 수와 관계없이 모든 쿠키를 전달합니다. 지정한 쿠키 전달(Forward cookies I specify)을 선택한 경우 표시되는 텍스트 상자에 배포에서 전달할 쿠키의 이름을 입력합니다. 쿠키 이름을 지정할 때 다음 와일드카드 기호를 지정할 수 있습니다.

- *는 쿠키 이름의 0개 이상의 문자에 해당합니다.
- ?는 쿠키 이름의 정확히 1문자에 해당합니다.

예를 들어, `userid_`*`member-number`*라는 이름의 쿠키가 포함된 객체에 대한 최종 사용자 요청을 가정해 봅니다. 각각의 사용자는 `member-number`(`userid_123`, `userid_124`, `userid_125` 등)에 대한 고유한 값을 갖습니다. 이 경우에는 배포에서 구성원별로 개별 콘텐츠 버전을 캐싱하려 합니다. 모든 쿠키를 오리진에 전달하여 이를 달성할 수는 있지만, 최종 사용자 요청에 배포에서 캐싱하길 원치 않는 쿠키가 일부 포함됩니다. 다음 값을 쿠키 이름으로 지정할 수 있습니다. 이렇게 하면 배포에서 `userid_`로 시작하는 모든 쿠키를 오리진으로 전달합니다. `userid_*`

쿼리 문자열 전달

배포에서 쿼리 문자열을 오리진에 전달할지 여부와 전달할 경우 어떤 쿼리 문자열을 전달할지 제어합니다. 쿼리 문자열은 지정된 파라미터에 값을 할당하는 URL의 일부입니다. 예를 들어, `https://example.com/over/there?name=ferret` URL은 `name=ferret` 쿼리 문자열을 포함합니다. 서버가 이러한 페이지에 대한 요청을 수신하면 프로그램을 실행하여 `name=ferret` 쿼리 문자열을 변경하지 않고 프로그램에 전달할 수 있습니다. 물음표는 구분자로 사용되며 쿼리 문자열의 일부를 구성하지 않습니다.

배포에서 쿼리 문자열 없이 전달하거나 지정한 쿼리 문자열만 전달하도록 선택할 수 있습니다. 오리진에서 쿼리 문자열 파라미터 값과 상관없이 동일한 콘텐츠 버전을 반환할 경우 쿼리 문자열을 전달하지 않도록 선택합니다. 이렇게 하면 배포에서 캐시로부터 요청을 제공할 수 있는 가능성을 높이고, 그에 따라 성능이 향상되고 오리진에 걸리는 부하가 줄어듭니다. 오리진 서버에서 하나 이상의 쿼리 문자열 파라미터를 기준으로 다른 콘텐츠 버전을 반환할 경우 지정한 쿼리 문자열만 전달하도록 선택합니다.

배포의 캐시 동작 변경

배포의 기본 캐시 동작을 변경하려면 다음 절차를 수행하세요.

1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 네트워킹을 선택합니다.
3. 기본 캐시 동작을 변경하려는 배포의 이름을 선택합니다.
4. 배포의 관리 페이지에서 캐시(Cache) 탭을 선택합니다.
5. 페이지의 캐싱 구성(Configure caching) 섹션에서 배포의 캐싱 사전 설정을 선택합니다. 자세한 정보는 캐싱 사전 설정을 참조하세요.
6. 기본 캐시 동작 변경(Change default cache behavior)을 선택하여 배포에 대한 기본 동작을 변경합니다. 그런 다음 배포에 대한 기본 동작을 선택합니다. 자세한 내용은 기본 동작을 참조하세요.
7. 경로 추가(Add path)를 선택하여 배포의 캐싱 동작에 디렉터리 및 파일 재정의를 추가합니다. 자세한 내용은 디렉터리 및 파일 재정의를 참조하세요.
8. 배포에 대해 편집할 고급 설정 옆에 표시된 연필 아이콘을 선택합니다. 자세한 내용은 고급 캐시 설정을 참조하세요.

배포 구성에 대한 변경 내용을 저장하면 배포에서 이러한 변경 내용을 모든 엣지 로케이션으로 전파하기 시작합니다. 구성이 엣지 로케이션에서 업데이트될 때까지 배포는 이전 구성에 따라 해당 위치에서 콘텐츠를 계속 제공합니다. 구성이 엣지 로케이션에서 업데이트된 후에는 배포가 즉시 새 구성에 따라 해당 위치에서 콘텐츠를 제공하기 시작합니다.

변경 내용이 모든 엣지 로케이션으로 즉시 전파되지는 않습니다. 전파가 완료되면 배포의 상태가 진행 중(InProgress)에서 사용(Enabled)으로 변경됩니다. 배포가 변경 내용을 전파하고 있는 동안은 지정된

엣지 로케이션이 이전 구성과 새 구성 중 어느 것에 기초하여 콘텐츠를 제공하는지 확인할 수 없습니다.

주제

- Lightsail 배포의 캐시 재설정

Lightsail 배포의 캐시 재설정

캐시 수명(유지 시간(TTL)) 설정은 콘텐츠가 Amazon Lightsail 배포 캐시에 남아 있는 시간을 제어합니다. 캐시 수명 주기 전에 캐시를 지워야 하는 경우 배포에서 캐시를 수동으로 재설정할 수도 있습니다. 캐시를 지운 후 다음에 사용자가 콘텐츠를 요청하면 배포에서 오리진의 최신 콘텐츠 버전을 가져와 캐싱합니다. 이 가이드에서는 배포에서 캐시를 수동으로 재설정하는 방법을 안내합니다. 배포에 대한 자세한 내용은 콘텐츠 전송 네트워크 배포를 참조하세요.

배포의 캐시 재설정

배포의 캐시를 재설정하려면 다음 절차를 완료하세요.

1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 네트워킹을 선택합니다.
3. 캐시를 재설정하려는 배포의 이름을 선택합니다.
4. 배포의 관리 페이지에서 캐시(Cache) 탭을 선택합니다.
5. 페이지의 캐시 재설정(Reset cache) 섹션으로 스크롤하고 캐시 재설정(Reset cache)을 선택합니다.
6. 확인 프롬프트에서 예, 재설정합니다(Yes, reset)를 선택하여 배포 캐시를 재설정할지 확인합니다. 배포 캐시를 재설정하지 않으려면 아니요, 취소합니다(No, cancel)를 선택합니다.

Lightsail 배포의 콘텐츠 오리진 변경

이 가이드에서는 Amazon Lightsail 배포의 오리진을 생성하고 나서 변경하는 방법을 안내합니다. 오리진은 배포의 최종 콘텐츠 소스입니다. 배포를 생성할 때 웹 사이트나 웹 애플리케이션의 콘텐츠를 호스팅하는 Lightsail 인스턴스, Lightsail 버킷 또는 하나 이상의 인스턴스가 연결된 Lightsail 로드 밸런서를 선택할 수 있습니다. 자세한 내용은 콘텐츠 전송 네트워크 배포를 참조하세요.

배포를 생성하고 나서 언제든지 오리진을 변경할 수 있습니다. 오리진을 변경할 경우 배포에서는 엣지 로케이션에 대한 변경 사항을 즉시 복제하기 시작합니다. 배포가 엣지 로케이션에서 새 오리진으로 업데이트될 때까지 지정된 엣지 로케이션의 이전 오리진으로 요청이 계속 전달됩니다.

오리진을 변경하더라도 배포에서는 엣지 캐시를 새 오리진의 콘텐츠로 다시 채울 필요가 없습니다. 웹 사이트 또는 웹 애플리케이션의 사용자 요청이 변경되지 않는 한, 배포에서는 콘텐츠의 캐시 수명이 만료될 때까지 계속해서 엣지 캐시에 이미 존재하는 콘텐츠를 제공합니다.

오리진 프로토콜 정책

오리진 프로토콜 정책은 배포에서 오리진으로부터 콘텐츠를 가져올 때 사용하는 프로토콜 정책입니다. 배포의 오리진을 선택한 후 오리진에서 콘텐츠를 가져올 때 배포에서 Hypertext Transfer Protocol(HTTP) 또는 Hypertext Transfer Protocol Secure(HTTPS)를 사용해야 하는지 정해야 합니다. 오리진이 HTTPS에 대해 구성되지 않은 경우 HTTP를 사용해야 합니다.

배포에 대해 다음 오리진 프로토콜 정책 중 하나를 선택할 수 있습니다.

- HTTP 전용(HTTP Only) - 배포에서 HTTP만 사용하여 오리진에 액세스합니다. 이것이 기본 설정입니다.
- HTTPS 전용(HTTPS Only) - 배포에서 HTTPS만 사용하여 오리진에 액세스합니다.

오리진 프로토콜 정책을 편집하는 단계는 이 가이드의 배포 오리진 변경(Change your distribution's origin) 섹션에 나와 있습니다.

배포 오리진 변경

배포 오리진을 변경하려면 다음 절차를 수행하세요.

1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 네트워킹을 선택합니다.
3. 오리진을 변경하려는 배포의 이름을 선택합니다.
4. 배포의 관리 페이지에서 세부 정보(Details) 탭을 선택하고, 해당 페이지의 오리진 선택(Choose your origin) 섹션으로 스크롤합니다.

   이 페이지의 오리진 선택(Select your origin) 섹션에 배포의 현재 오리진이 표시됩니다.
5. 오리진 생성(Create Origin)을 선택합니다.
6. 오리진 리소스가 생성된 AWS 리전을 선택합니다.

   배포는 전역 리소스입니다. 모든 AWS 리전에서 오리진을 참조하고 콘텐츠를 전 세계에 배포할 수 있습니다.
7. 오리진을 선택합니다. 오리진은 인스턴스, 버킷 또는 하나 이상의 인스턴스가 연결된 로드 밸런서일 수 있습니다.

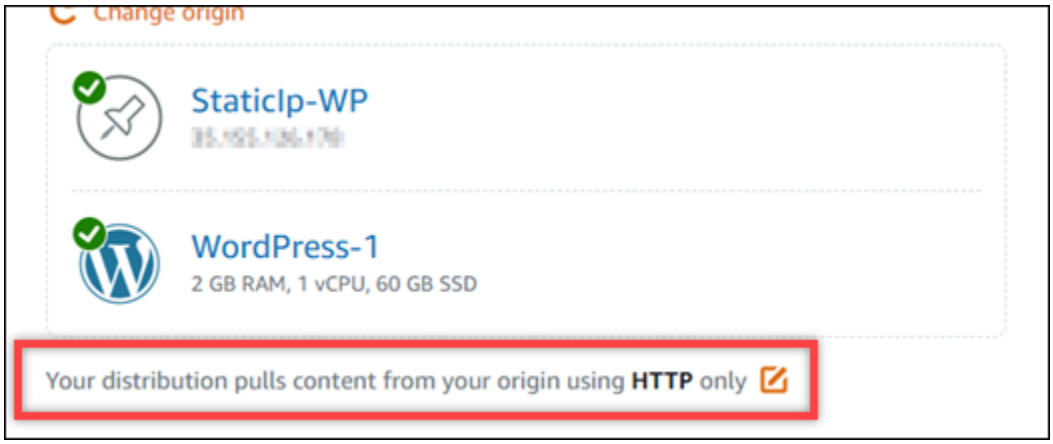
8. 저장(Save)을 선택하여 배포를 새 오리진으로 업데이트합니다.

   배포의 오리진을 선택한 후 오리진에서 콘텐츠를 가져올 때 배포에서 Hypertext Transfer Protocol(HTTP) 또는 Hypertext Transfer Protocol Secure(HTTPS)를 사용해야 하는지 정해야 합니다.

9. (선택 사항) 오리진 프로토콜 정책을 변경하려면 배포에서 사용하는 현재 오리진 프로토콜 정책 옆에 표시된 연필 아이콘을 선택합니다. 자세한 내용은 오리진 프로토콜 정책을 참조하세요.

   이 옵션은 페이지의 오리진 선택(Choose your origin) 섹션에서 배포에 대해 선택한 오리진 리소스 아래에 나와 있습니다.

   > **Note**
   >
   > Lightsail 버킷을 배포의 오리진으로 선택하면 오리진 프로토콜 정책이 기본적으로 HTTPS 전용(HTTPS Only)으로 설정됩니다. 버킷이 배포의 오리진인 경우에는 오리진 프로토콜 정책을 변경할 수 없습니다.



10. HTTP 전용(HTTP Only) 또는 HTTPS 전용(HTTPS Only)을 선택한 다음 저장(Save)을 선택하여 오리진 프로토콜 정책을 저장합니다.

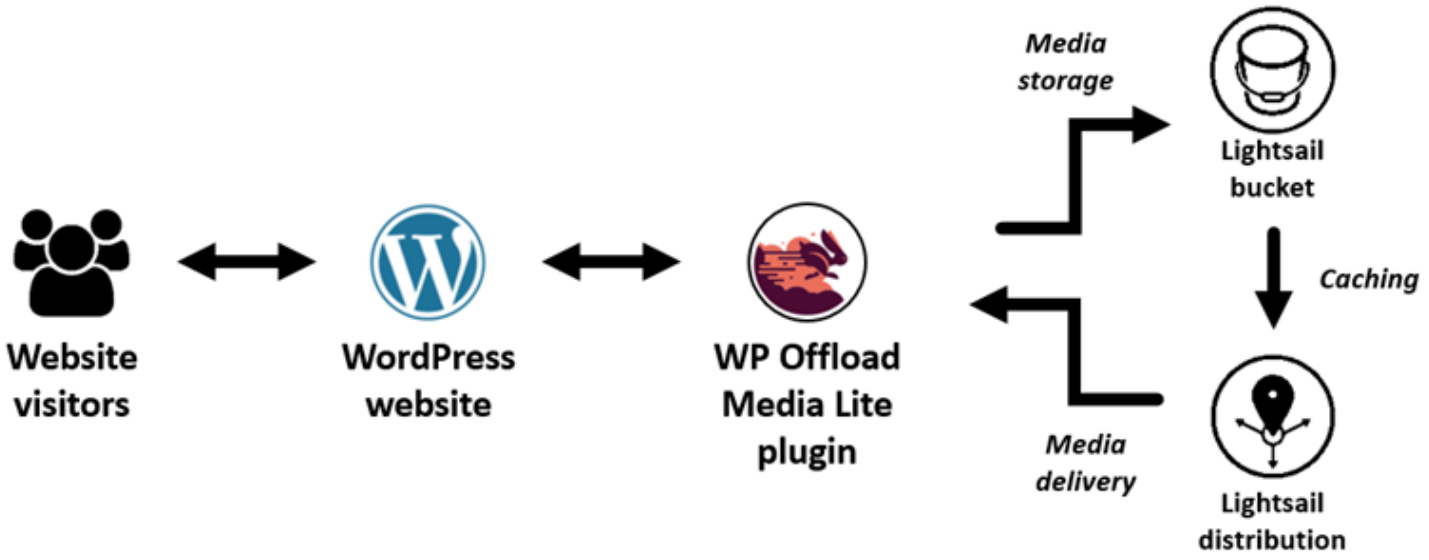
배포 구성에 대한 변경 내용을 저장하면 배포에서 이러한 변경 내용을 모든 엣지 로케이션으로 전파하기 시작합니다. 구성이 엣지 로케이션에서 업데이트될 때까지 배포는 이전 구성에 따라 해당 위치에서 콘텐츠를 계속 제공합니다. 구성이 엣지 로케이션에서 업데이트된 후에는 배포가 즉시 새 구성에 따라 해당 위치에서 콘텐츠를 제공하기 시작합니다.

변경 내용이 모든 엣지 로케이션으로 즉시 전파되지는 않습니다. 전파가 완료되면 배포의 상태가 진행 중(InProgress)에서 사용(Enabled)으로 변경됩니다. 배포가 변경 내용을 전파하고 있는 동안은 지정된

엣지 로케이션이 이전 구성과 새 구성 중 어느 것에 기초하여 콘텐츠를 제공하는지 확인할 수 없습니다.

Lightsail 버킷 및 CDN 배포를 통해 미디어 파일을 효율적으로 제공

이 자습서에서는 Amazon Lightsail 버킷을 Lightsail 콘텐츠 전송 네트워크(CDN) 배포의 오리진으로 구성하는 데 필요한 단계를 설명합니다. 또한, 이미지 및 동영상 파일과 같은 미디어를 버킷에 업로드 및 저장하고 배포에서 미디어를 전달하도록 WordPress 웹 사이트를 구성하는 방법에 대해서도 안내합니다. 이렇게 하는 1가지 방법은 WP Offload Media Lite 플러그 인을 사용하는 것입니다. 다음 다이어그램에서는 이 구성을 보여줍니다.



웹 사이트 미디어를 Lightsail 버킷에 저장하면 인스턴스가 해당 파일을 저장하고 제공해야 하는 부담을 줄일 수 있습니다. Lightsail 배포에서 미디어를 캐싱하고 제공하면 해당 파일이 웹 사이트 방문자에게 더 빠르게 전달되며 전반적인 웹 사이트 성능을 향상시킬 수 있습니다. 배포에 대한 자세한 내용은 콘텐츠 전송 네트워크 배포를 참조하세요. 버킷에 대한 자세한 내용은 객체 스토리지를 참조하세요.

목차

- 1단계: 필수 구성 요소 완성
- 2단계: 버킷 권한 수정
- 3단계: 버킷을 오리진으로 하는 배포 생성
- 4단계: 배포용 사용자 지정 하위 도메인 활성화
- 5단계: WordPress 웹 사이트에 WP Offload Media Lite 플러그 인 설치
- 6단계: WordPress 웹 사이트와 Lightsail 버킷 및 배포 간의 연결 테스트

1단계: 필수 구성 요소 완성

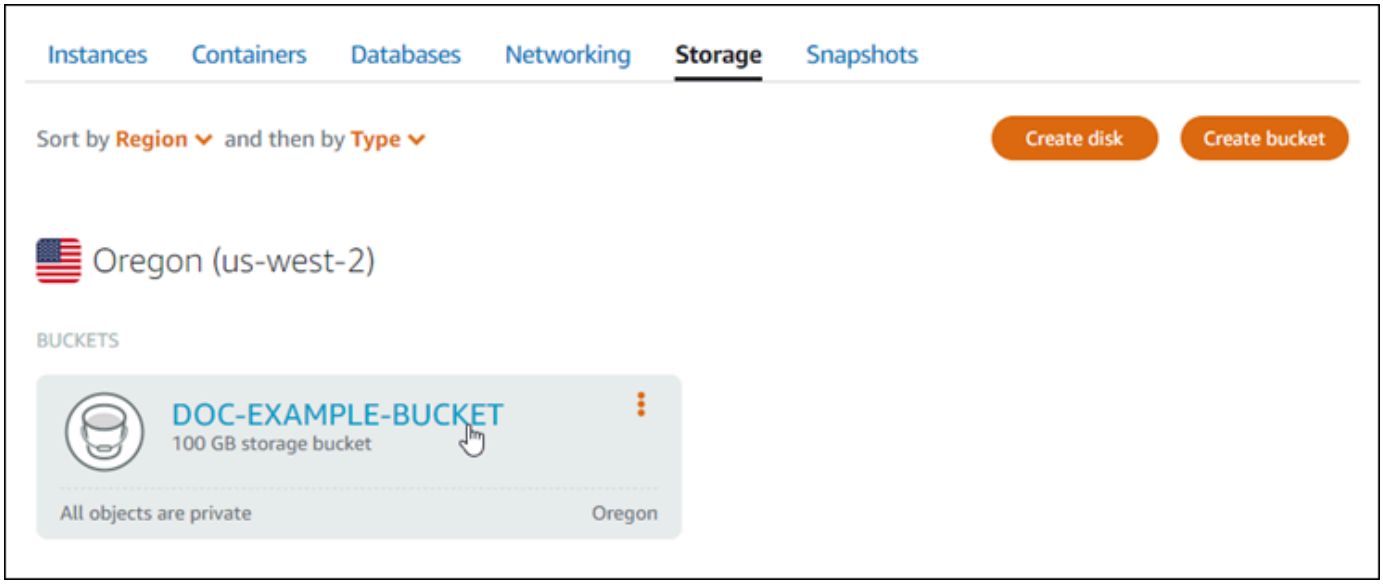
아직 수행하지 않은 경우 다음 사전 조건을 완료합니다.

- Lightsail에서 WordPress 인스턴스를 생성하여 구성하고 관리 대시보드에 로그인할 암호를 가져옵니다. 자세한 내용은 자습서: Amazon Lightsail에서 WordPress 인스턴스 시작 및 구성을 참조하세요.
- Lightsail 객체 스토리지 서비스에서 버킷을 생성합니다. 자세한 내용은 Lightsail에서 버킷 생성을 참조하세요.

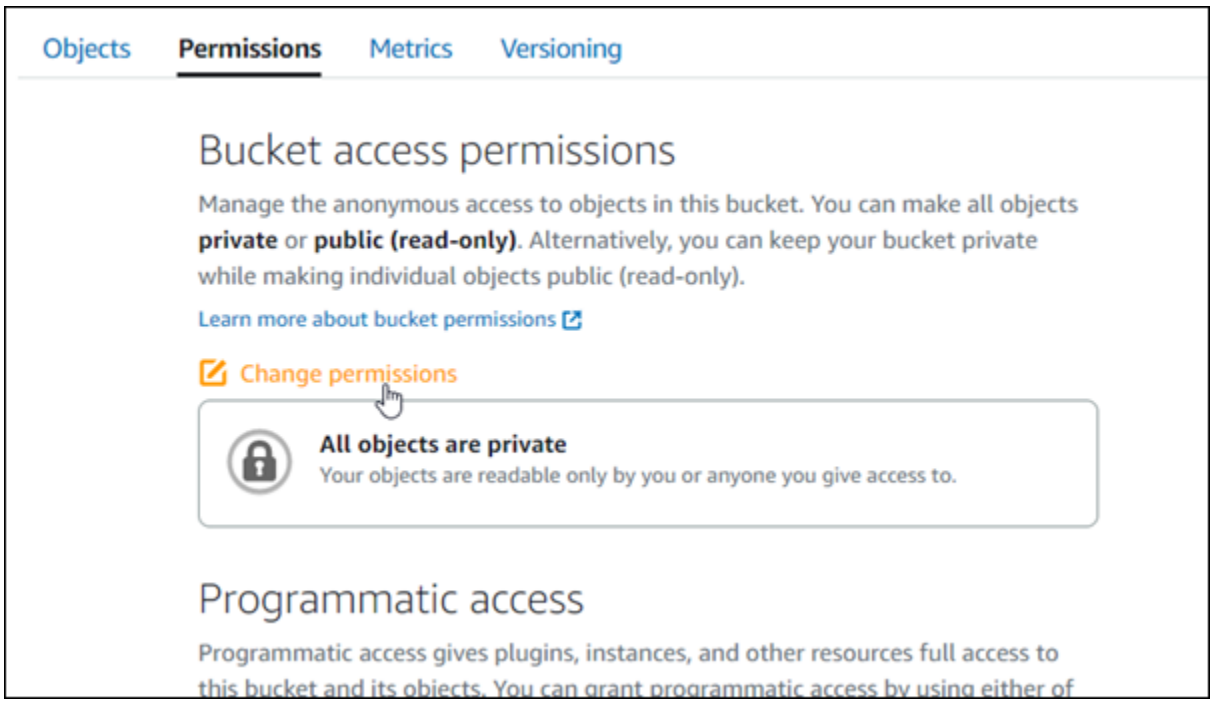
2단계: 버킷 권한 수정

WordPress 인스턴스와 WP Offload Media Lite 플러그 인에 버킷에 액세스할 권한을 부여하려면 다음 절차를 완료하세요. 버킷의 권한은 개별 객체 공개 가능(읽기 전용)(Individual objects can be made public (read-only))으로 설정되어야 합니다. 또한, WordPress 인스턴스를 버킷에 연결해야 합니다. 버킷 권한에 대한 자세한 내용은 버킷 권한을 참조하세요.

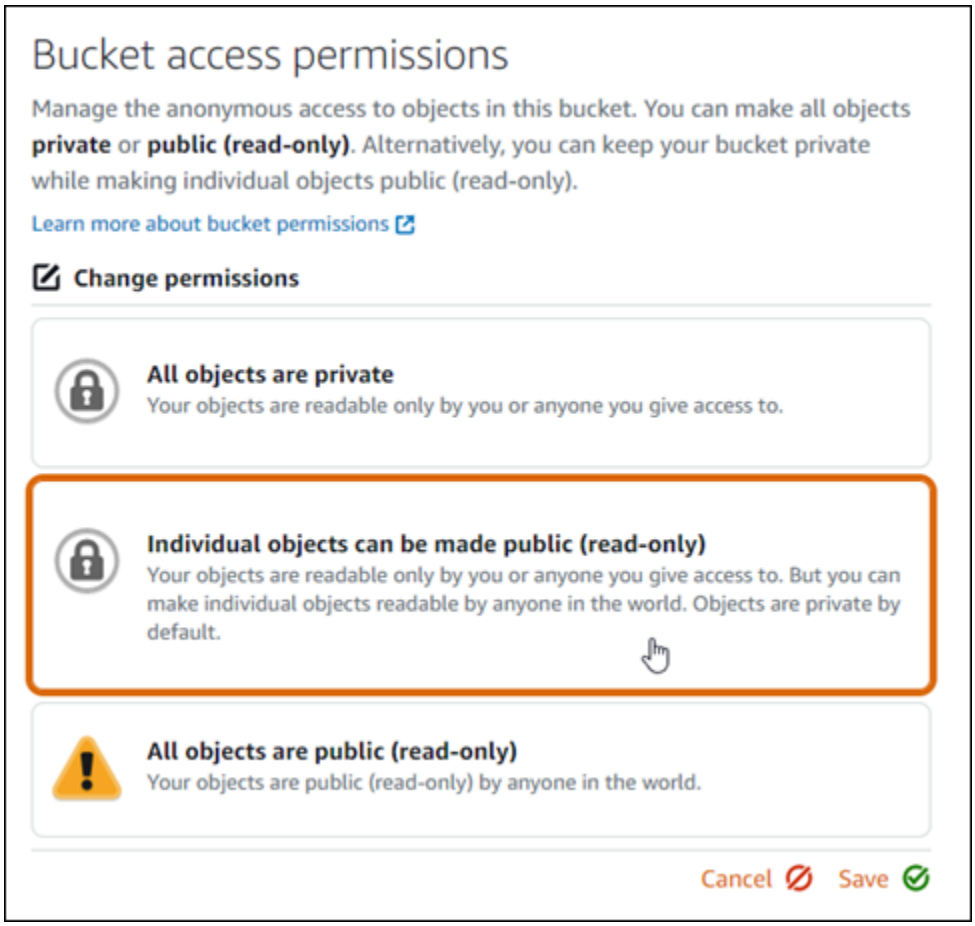
1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 스토리지를 선택합니다.
3. WordPress 웹 사이트에서 사용하려는 버킷의 이름을 선택합니다.



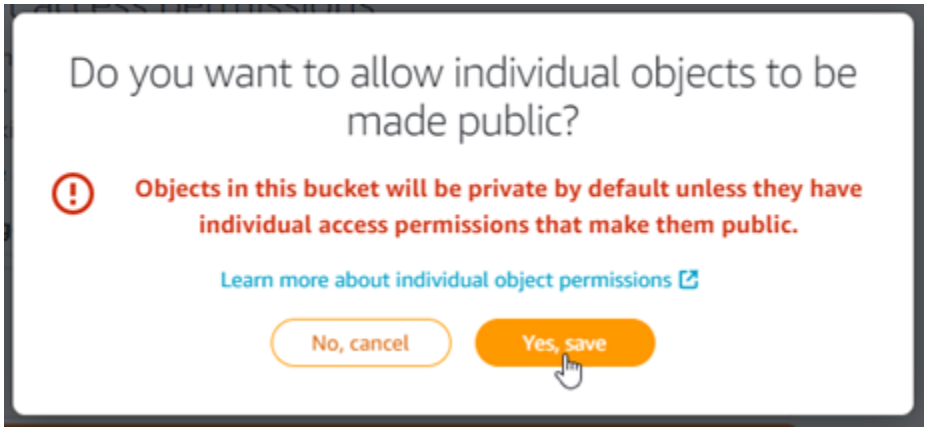
4. 버킷 관리(Bucket management) 페이지에서 권한(Permissions) 탭을 선택합니다.
5. 페이지의 버킷 액세스 권한(Bucket access permissions) 섹션에서 권한 변경(Change permissions)을 선택합니다.



6. 개별 객체 공개 가능 및 읽기 전용(Individual objects can be made public and read only)을 선택합니다.

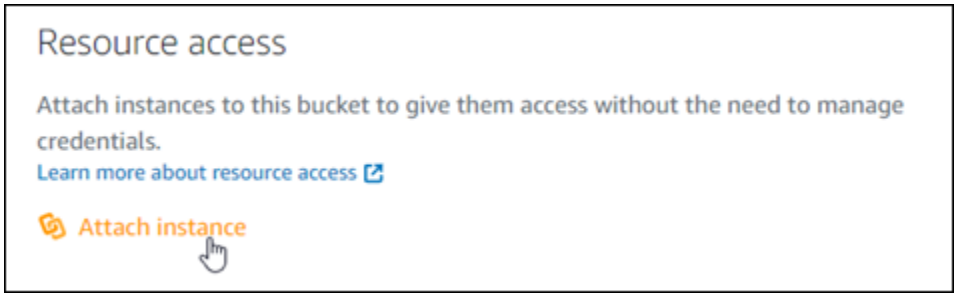


7. 저장을 선택합니다.
8. 표시되는 확인 프롬프트에서 예, 저장합니다(Yes, save)를 선택합니다.

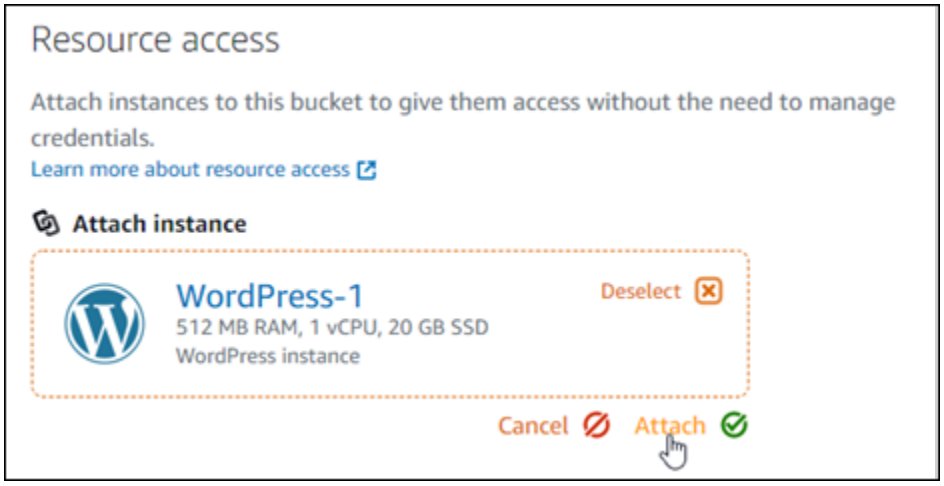


잠시 후 버킷이 개별 객체 액세스를 허용하도록 구성됩니다. 이렇게 하면 Offload Media Lite 플러그 인을 사용하여 WordPress 웹 사이트에서 버킷에 업로드된 객체를 고객이 읽을 수 있습니다.

9. 페이지의 리소스 액세스(Resource access) 섹션으로 스크롤하고 인스턴스 연결(Attach instance)을 선택합니다.



10. 표시되는 드롭다운 메뉴에서 WordPress 인스턴스의 이름을 선택한 후 연결(Attach)을 선택합니다.



몇 분 정도 지나면 WordPress 인스턴스가 버킷에 연결됩니다. 이렇게 하면 WordPress 인스턴스에 버킷과 해당 객체를 관리할 액세스 권한을 부여할 수 있습니다.

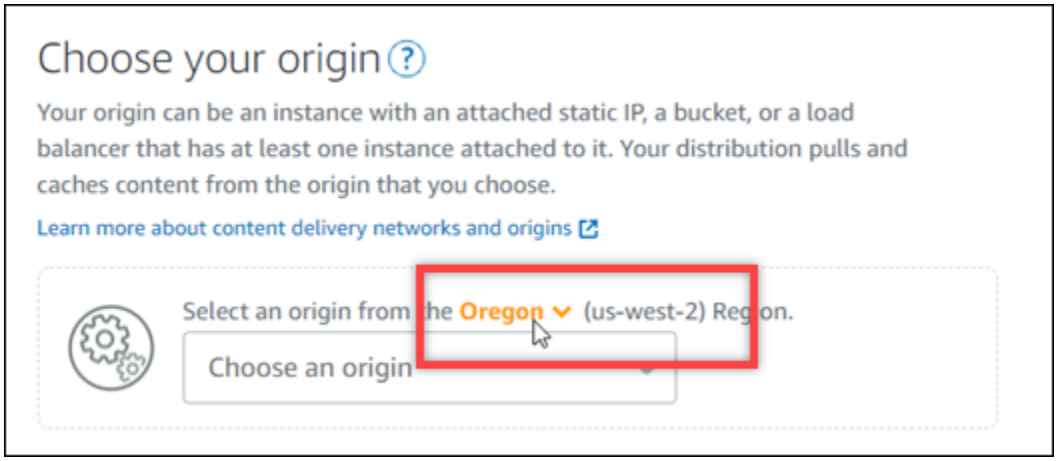
3단계: 버킷을 오리진으로 하는 배포 생성

Lightsail 배포를 생성하고 Lightsail 버킷을 오리진으로 선택하려면 다음 절차를 완료하세요.

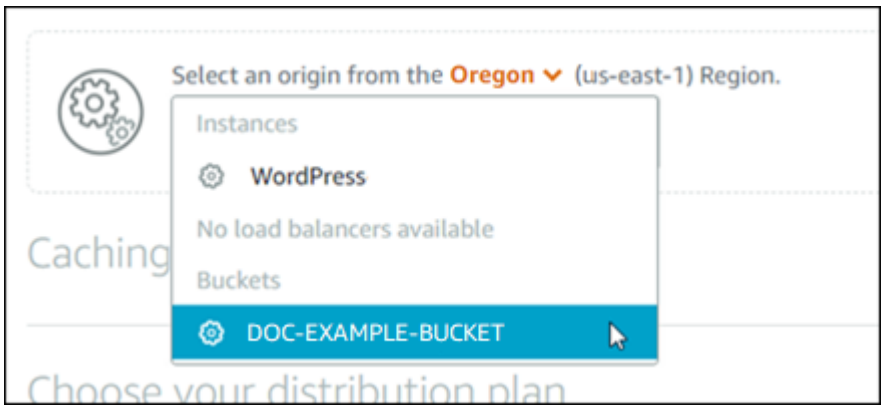
1. Lightsail 콘솔의 상단 탐색 메뉴에서 홈(Home)을 선택합니다.
2. 왼쪽 탐색 창에서 네트워킹을 선택합니다.
3. 배포 생성을 선택합니다.

4. 페이지의 오리진 선택 섹션에서 버킷을 생성한 AWS 리전 을 선택합니다.

   배포는 전역 리소스입니다. 모든에서 버킷을 참조 AWS 리전하고 해당 콘텐츠를 전 세계에 배포할 수 있습니다.

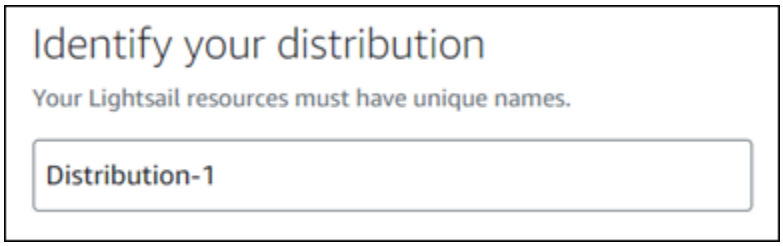


5. 오리진으로 버킷을 선택합니다.



> **Note**
>
> 버킷의 권한은 개별 객체 공개 가능(읽기 전용)(Individual objects can be made public (read-only))으로 설정되어야 합니다. 퍼블릭인 개별 객체만 배포에 의해 캐시되고 제공됩니다. 배포 오리진으로 버킷을 선택하면 오리진 프로토콜 정책, 캐싱 동작, 기본 동작, 디렉터리 및 파일 재정의를 지정하는 옵션을 사용할 수 없으며 이를 편집할 수 없습니다. 오리진 프로토콜 정책은 버킷에 대해서만 HTTP 전용으로 기본 설정되고 캐싱 동작은 기본적으로 모든 항목 캐싱으로 설정됩니다. 배포가 생성된 후에 배포의 고급 캐시 설정을 변경할 수 있습니다.

6. 배포 플랜을 선택합니다.
7. 배포의 이름을 입력합니다.



배포 이름:

- Lightsail 계정 AWS 리전 의 각 내에서 고유해야 합니다.
- 2–255자로 구성되어야 합니다.
- 영숫자 문자 또는 숫자로 시작하고 끝나야 합니다.
- 영숫자 문자, 숫자, 마침표, 대시, 밑줄이 포함될 수 있습니다.

8. 배포 생성을 선택합니다.

몇 분 정도 지나면 배포가 생성됩니다. 새 배포가 활성화(Enabled) 상태로 변하면 버킷에 있는 객체를 제공하고 캐싱할 준비가 된 것입니다.

4단계: 배포용 사용자 지정 하위 도메인 활성화

배포를 생성할 때 배포는 `123abc.cloudfront.net`과 유사한 기본 도메인으로 구성됩니다. WP Offload Media Lite 플러그 인을 구성할 때 미디어 파일의 소스로 기본 도메인을 지정할 수 있습니다. 그러나 배포에 대해 사용자 지정 도메인을 활성화하는 것이 좋습니다. 배포에 활성화하는 사용자 지정 도메인은 WordPress 웹 사이트에서 사용하는 도메인의 하위 도메인이어야 합니다. 예를 들어, WordPress 웹 사이트에서 `mycustomdomain.com`을 사용하는 경우 배포에 사용자 지정 도메인 `media.mycustomdomain.com`을 사용하도록 선택할 수 있습니다. WordPress 웹 사이트와 배포 간에 동일한 도메인 및 하위 도메인 조합을 사용하면 웹 사이트의 검색 엔진 최적화 점수를 향상시킬 수 있습니다.

배포에 사용할 사용자 지정 도메인을 구성하려면 다음 단계를 완료하세요.

1. 배포와 함께 사용할 도메인의 Lightsail SSL/TLS 인증서를 생성합니다. Lightsail 배포에는 HTTPS가 필요하므로, 배포에 사용하려면 먼저 도메인에 대한 SSL/TLS 인증서를 요청해야 합니다. 자세한 내용은 배포용 SSL/TLS 인증서 생성을 참조하세요.
2. 도메인을 배포와 함께 사용하려면 배포에 대해 사용자 지정 도메인을 활성화하면 됩니다. 도메인에 대해 생성한 Lightsail SSL/TLS 인증서를 지정해서 사용자 지정 도메인을 활성화할 수 있습니다. 이렇게 하면 도메인이 배포에 추가되고 HTTPS가 활성화됩니다. 자세한 내용은 배포용 사용자 지정 도메인 사용을 참조하세요.
3. 도메인의 DNS에 별칭 레코드를 추가합니다. 별칭 레코드를 추가하면 도메인을 방문한 사용자가 배포를 통해 라우팅됩니다. 자세한 내용은 배포로 도메인 연결을 참조하세요.

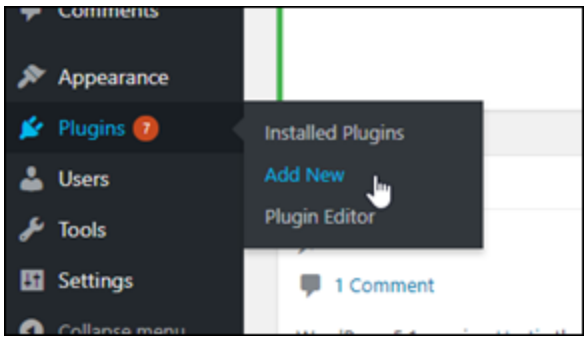
5단계: WordPress 웹 사이트에 WP Offload Media Lite 플러그 인 설치

WP Offload Media Lite 플러그 인을 WordPress 웹 사이트에 설치하려면 다음 절차를 완료하세요. 이 플러그 인은 WordPress 미디어 업로더를 통해 추가된 이미지, 비디오, 문서 및 기타 미디어를 Lightsail 버킷에 자동으로 복사합니다. 또한, Lightsail 배포를 통해 버킷에서 미디어를 제공하도록 구성할 수도 있습니다. 자세한 내용은 WordPress 웹 사이트의 WP Offload Media Lite를 참조하세요.

1. WordPress 웹 사이트의 대시보드에 관리자로 로그인합니다.

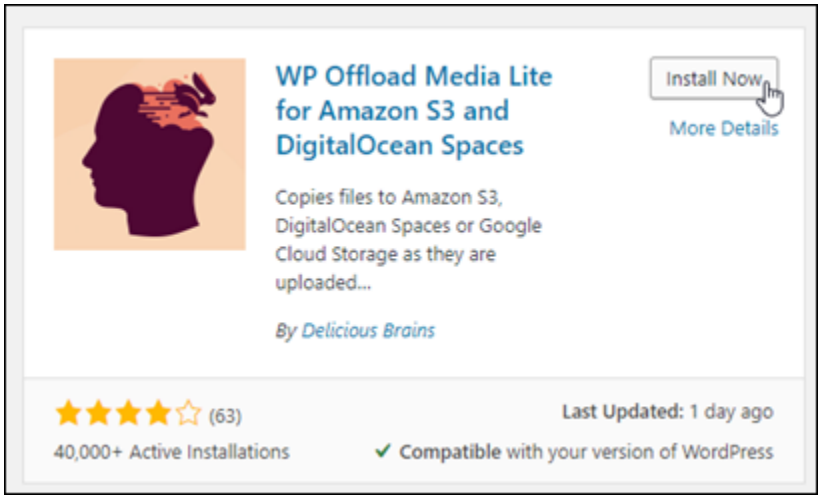
   자세한 내용은 Amazon Lightsail의 Bitnami 인스턴스에 대한 애플리케이션 사용자 이름과 암호 가져오기를 참조하세요.

2. 왼쪽 탐색 메뉴에서 플러그 인(Plugins)을 일시 중지하고 새로 추가(Add New)를 선택합니다.

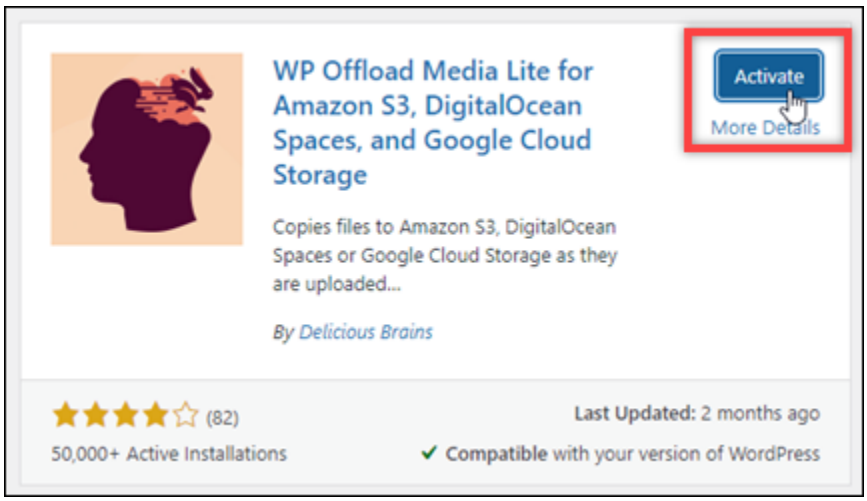


3. WP Offload Media Lite를 검색합니다.

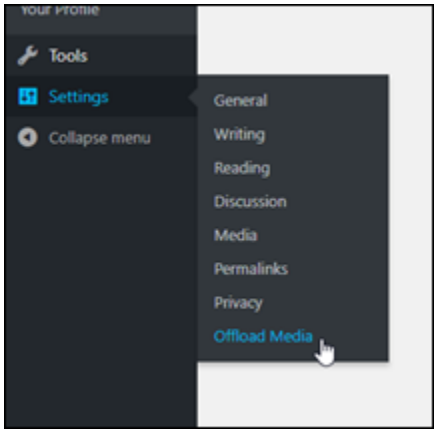
4. 검색 결과에서 WP Offload Media Lite 플러그 인 옆에 있는 지금 설치(Install Now)를 선택합니다.



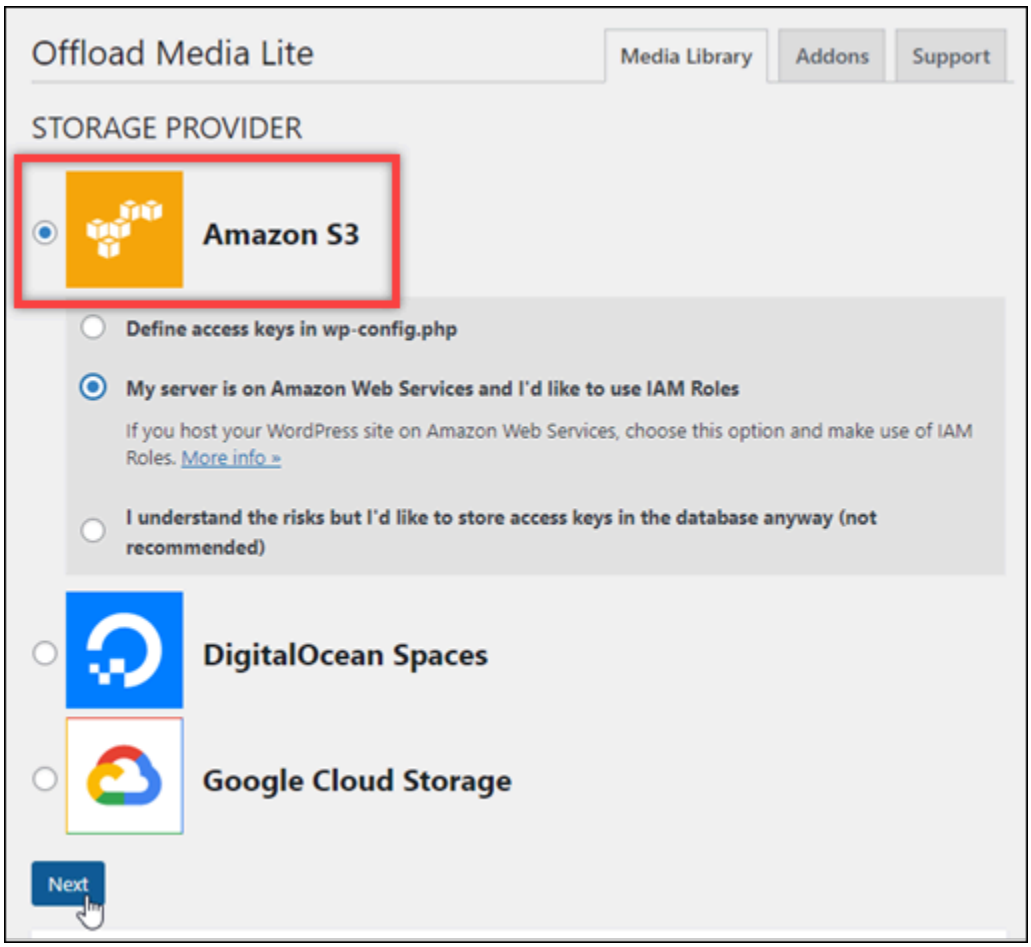
5. 플러그 인 설치가 끝나면 활성화(Activate)를 선택합니다.



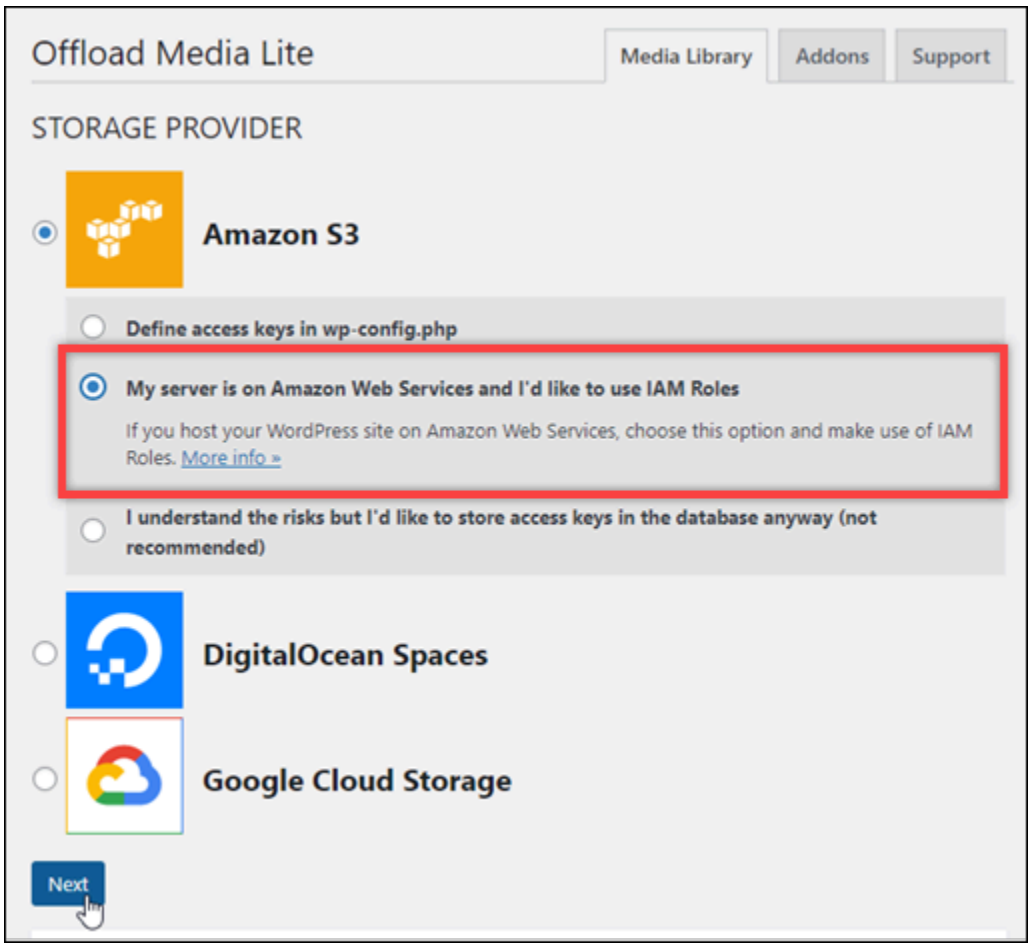
6. 왼쪽 탐색 메뉴에서 설정을 선택한 후 Offload Media를 선택합니다.



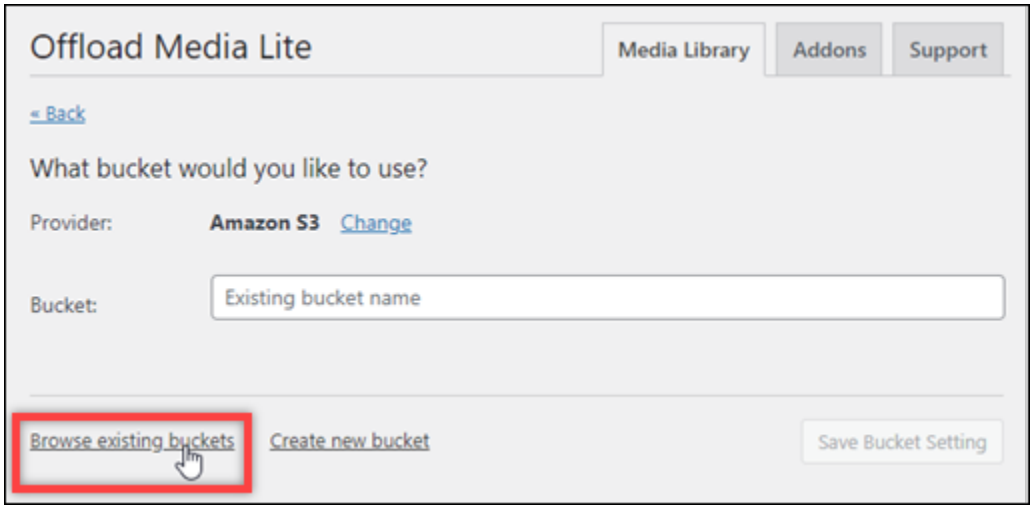
7. Offload Media Lite 페이지에서 Amazon S3를 스토리지 제공자로 지정합니다.



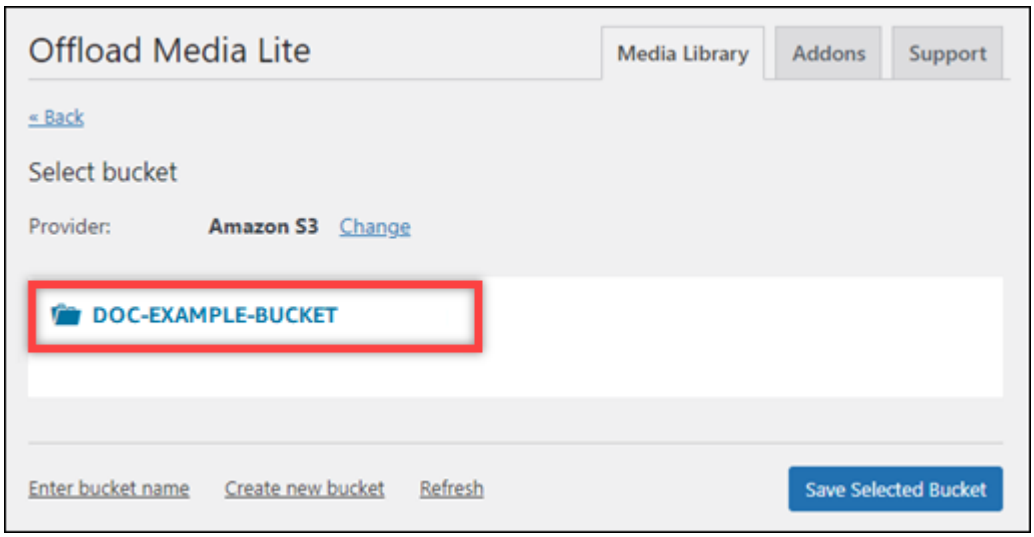
8. 서버가 Amazon Web Services에 있으며 IAM 역할을 사용하려고 합니다(My server is on Amazon Web Services and I'd like to use IAM Roles)를 선택합니다.



9. 다음을 선택합니다.
10. 표시되는 어느 버킷을 사용하시겠어요?(What bucket would you like to use?) 페이지에서 기존 버킷 찾아보기(Browse existing buckets)를 선택합니다.



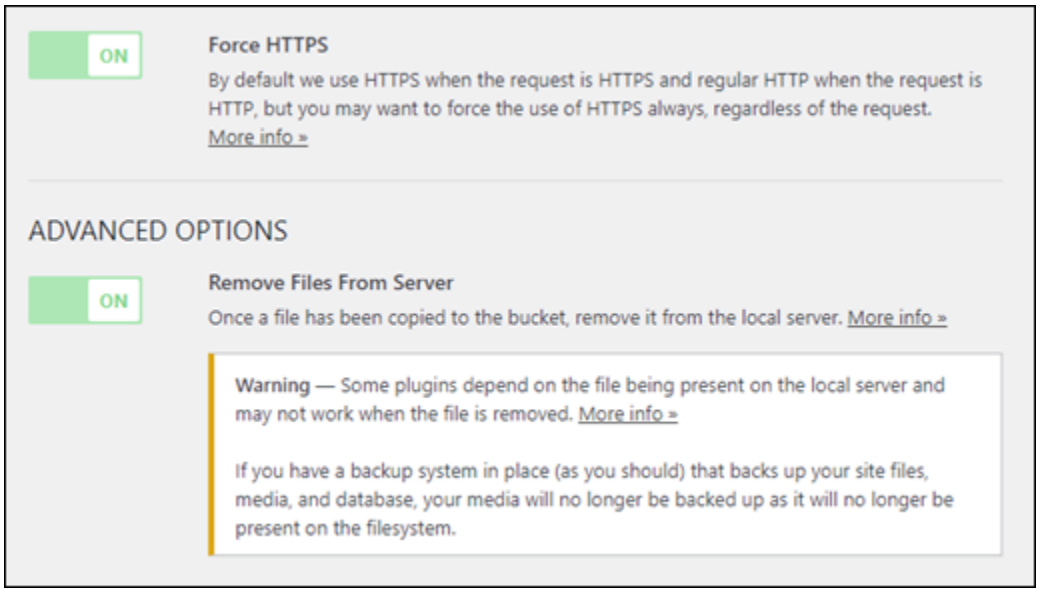
11. 직접 생성하여 WordPress 인스턴스에서 사용하려는 버킷의 이름을 선택합니다.



12. Offload Media Lite 설정(Offload Media Lite Settings) 페이지가 표시되면 HTTPS 강제 실행(Force HTTPS) 및 서버에서 파일 제거(Remove Files From Server)를 선택하여 켭니다.

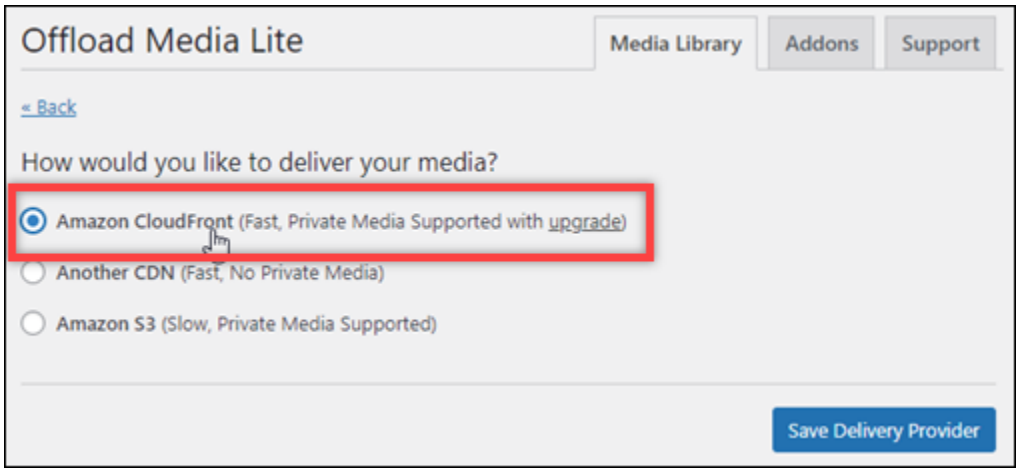
    - Lightsail 버킷은 기본적으로 HTTPS를 사용하여 미디어 파일을 제공하므로 HTTPS 강제 실행(Force HTTPS) 설정을 켜야 합니다. 이 기능을 켜지 않으면 WordPress 웹 사이트에서 Lightsail 버킷에 업로드된 미디어 파일이 웹 사이트 방문자에게 올바르게 제공되지 않습니다.

      서버에서 파일 제거(Remove Files From Server) 설정을 사용하면 Lightsail 버킷에 업로드된 미디어가 인스턴스의 디스크에도 저장되지 않습니다. 이 기능을 켜지 않으면 Lightsail 버킷에 업로드된 미디어 파일이 WordPress 인스턴스의 로컬 스토리지에도 저장됩니다.

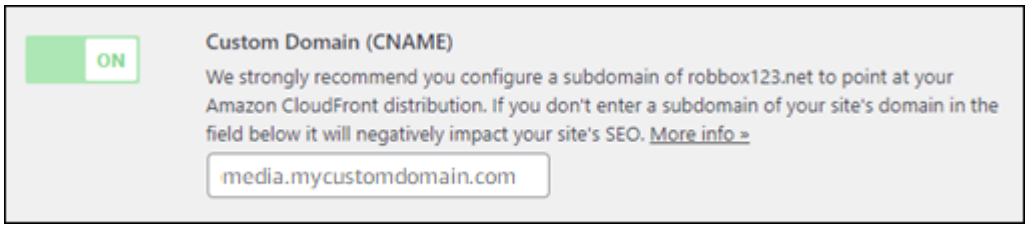


13. 페이지의 전송(Delivery) 섹션에서 Amazon S3 레이블 옆의 변경(Change)을 선택합니다.

14. 미디어를 전송할 방법 페이지가 표시되면 Amazon CloudFront를 선택합니다.



15. 전송 제공자 저장(Save Delivery Provider)을 선택합니다.
16. Offload Media Lite 설정(Offload Media Lite Settings) 페이지가 표시되면 사용자 지정 도메인(CNAME)(Custom Domain (CNAME))을 선택하여 켭니다. 그런 다음 텍스트 상자에 Lightsail 배포의 도메인을 입력합니다. 이 도메인은 배포의 기본 도메인(예: `123abc.cloudfront.net`)이거나 배포의 사용자 지정 도메인(예: `media.mycustomdomain.com`)일 수 있습니다(활성화한 경우).



17. 변경 사항 저장(Save Changes)을 선택합니다.

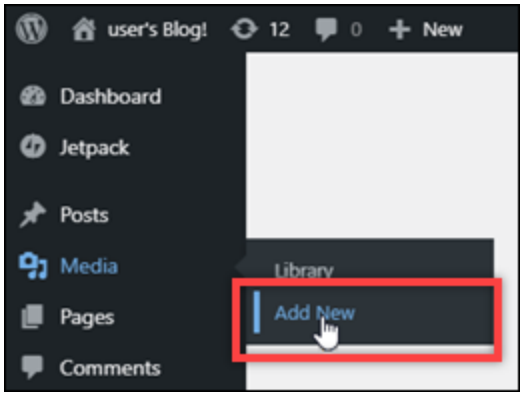
> **Note**
>
> 나중에 Offload Media Lite 설정(Offload Media Lite Settings) 페이지로 돌아가려면 왼쪽 탐색 메뉴에서 설정(Settings)을 일시 중지하고 Offload Media를 선택하면 됩니다.

이제 WordPress 웹 사이트가 Media Lite 플러그 인을 사용하도록 구성되었습니다. 다음에 WordPress를 통해 미디어 파일을 업로드하면 해당 파일이 Lightsail 버킷에 자동으로 업로드되어 배포에서 제공됩니다. 구성을 테스트하려면 이 자습서의 다음 섹션을 계속 진행합니다.

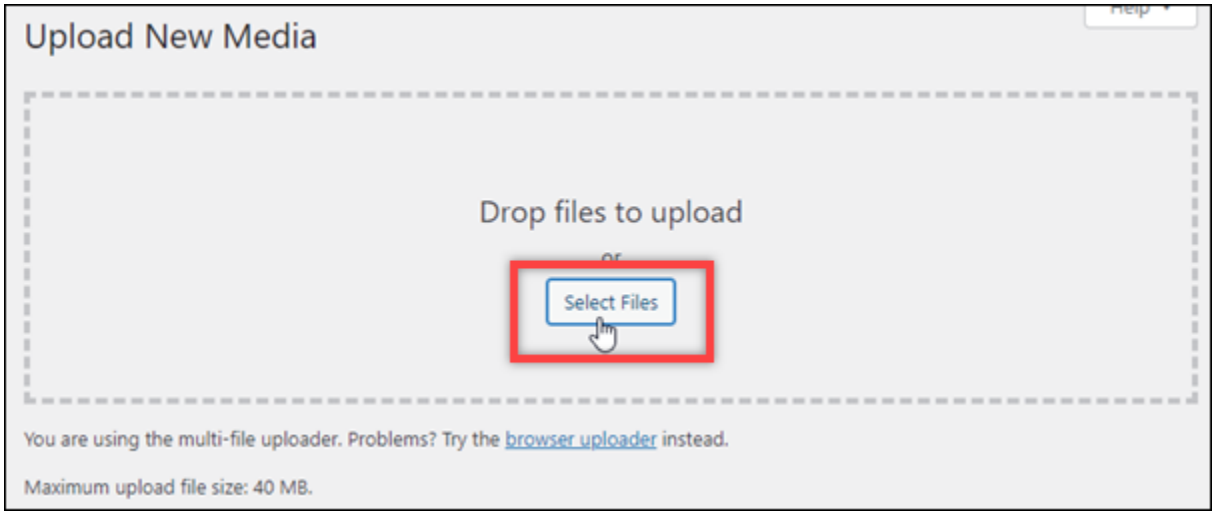
6단계: WordPress 웹 사이트와 Lightsail 버킷 및 배포 간의 연결 테스트

WordPress 인스턴스에 미디어 파일을 업로드하고 해당 파일이 Lightsail 버킷에 업로드되어 배포에서 제공되는지 확인하려면 다음 절차를 완료하세요.

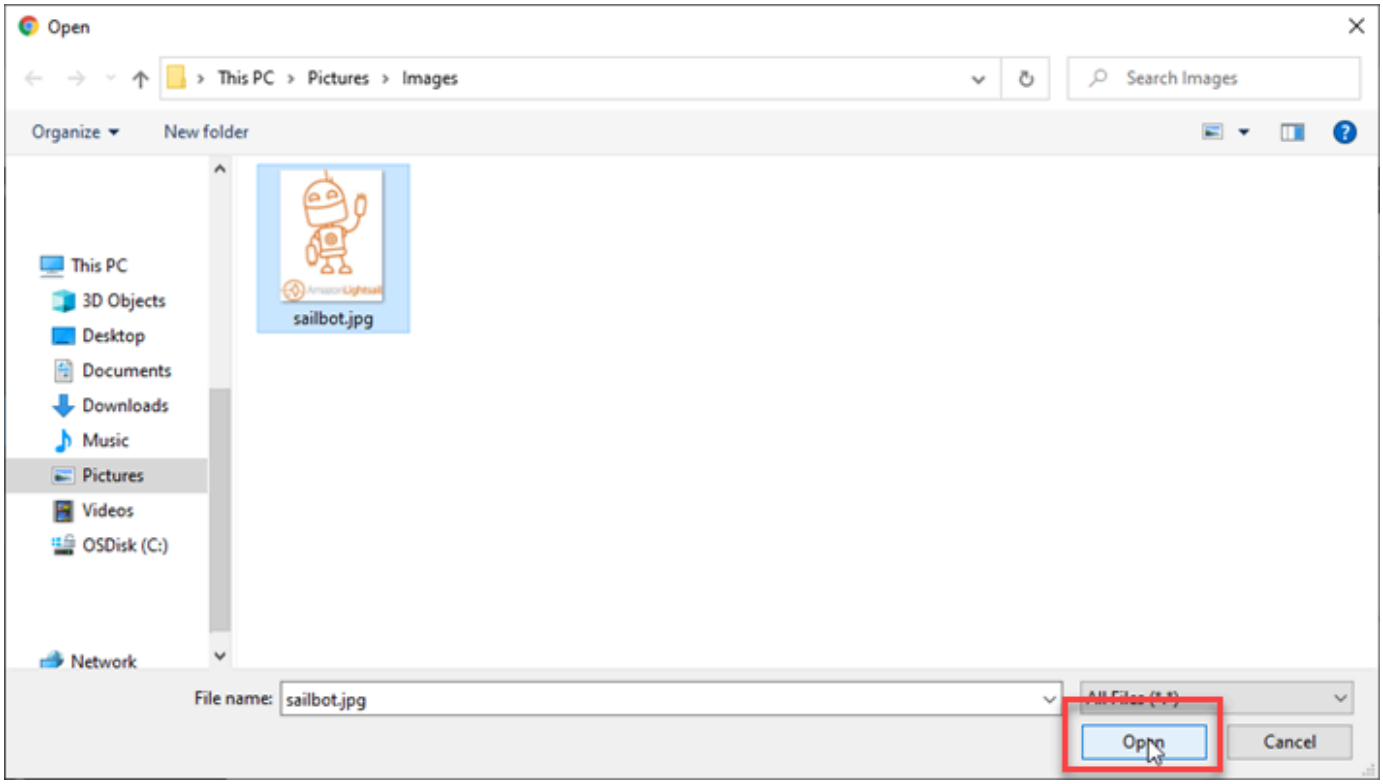
1. WordPress 대시보드의 왼쪽 탐색 메뉴에서 미디어(Media)를 일시 중지하고 새로 추가(Add New)를 선택합니다.



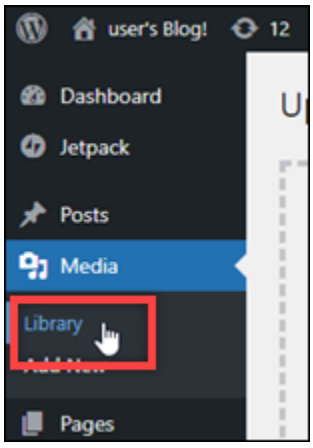
2. 표시되는 새 미디어 업로드(Upload New Media) 페이지에서 파일 선택(Select Files)을 선택합니다.



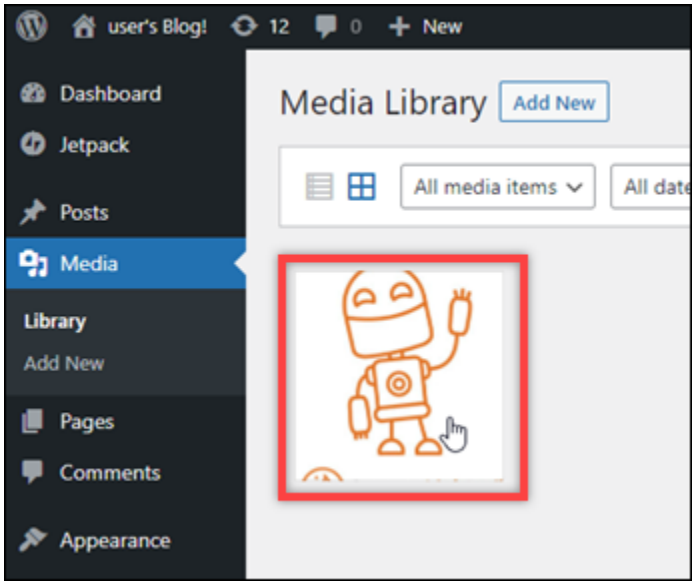
3. 로컬 컴퓨터에서 업로드할 미디어 파일을 선택하고 열기(Open)를 선택합니다.



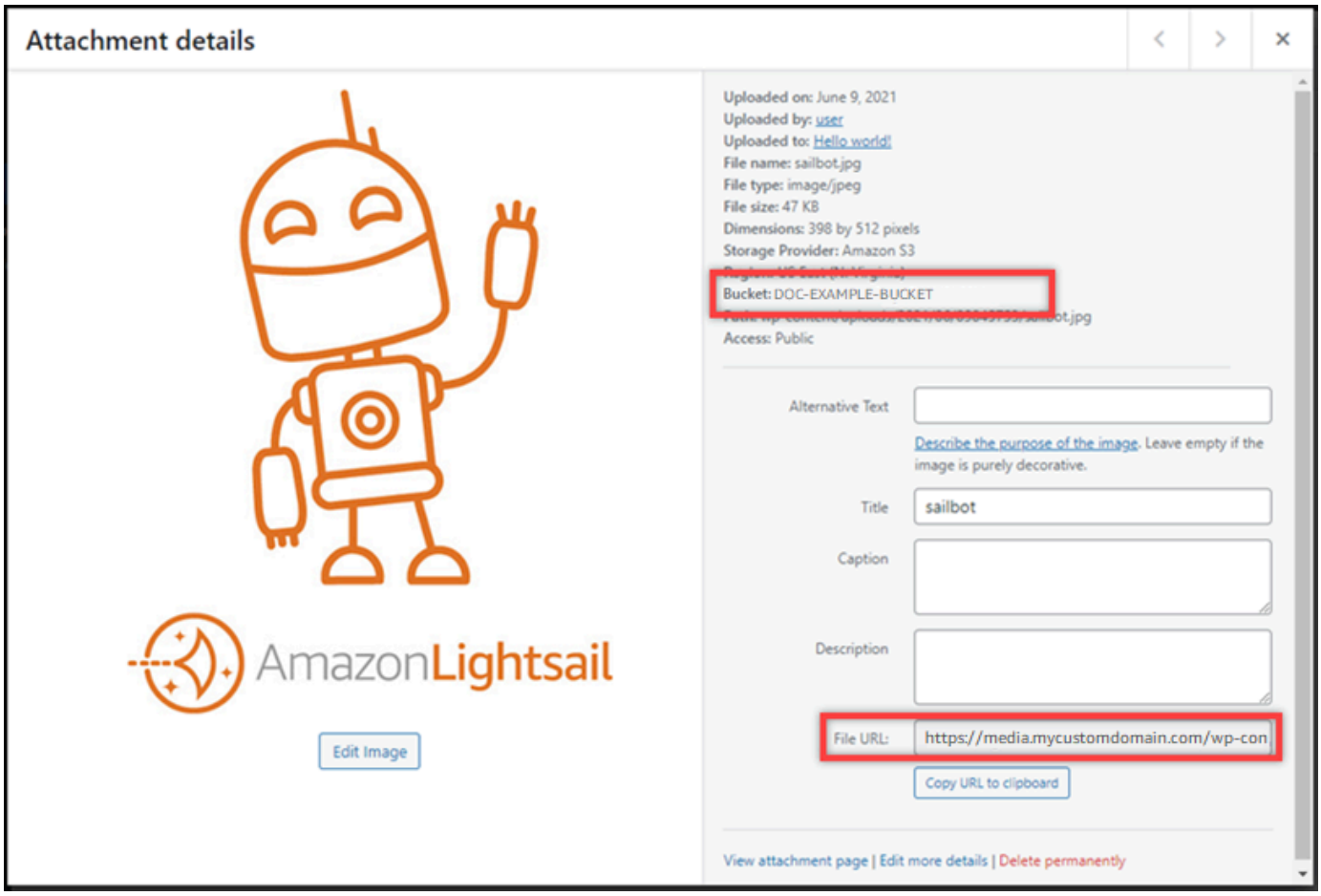
4. 파일 업로드가 완료되면 왼쪽 탐색 메뉴에서 미디어(Media) 아래의 라이브러리(Library)를 선택합니다.



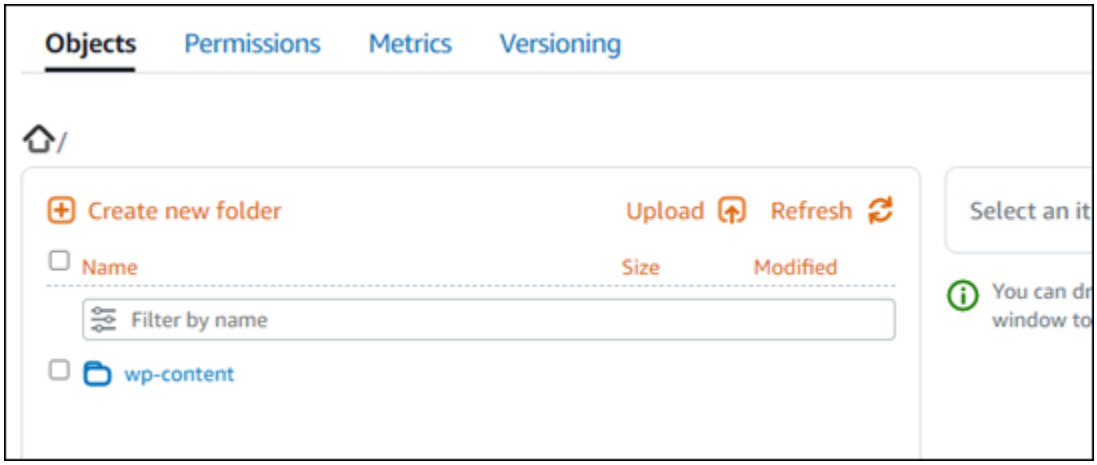
5. 최근에 업로드한 파일을 선택합니다.



6. 파일의 세부 정보 패널에서 버킷 이름이 버킷(Bucket) 필드에 나타납니다. 배포의 URL이 파일 URL(File URL) 필드에 표시됩니다.



7. Lightsail 버킷 관리 페이지의 객체(Objects) 탭으로 이동하면 wp-content 폴더가 표시됩니다. 이 폴더는 Offload Media Lite 플러그 인에 의해 생성되며 업로드된 미디어 파일을 저장하는 데 사용됩니다.



버킷 및 객체 관리

다음은 Lightsail 객체 스토리지 버킷을 관리하는 일반적인 단계입니다.

1. Amazon Lightsail 객체 스토리지 서비스의 객체와 버킷에 대해 알아보세요. 자세한 내용은 Amazon Lightsail의 객체 스토리지를 참조하세요.
2. Amazon Lightsail에서 버킷에 지정할 수 있는 이름에 대해 알아봅니다. 자세한 내용은 Amazon Lightsail의 버킷 이름 지정 규칙을 참조하세요.
3. 버킷을 생성하여 Lightsail 객체 스토리지 서비스를 시작합니다. 자세한 내용은 Amazon Lightsail에서 버킷 생성을 참조하세요.
4. 버킷의 보안 모범 사례와 버킷에 구성할 수 있는 액세스 권한에 대해 알아봅니다. 버킷의 모든 객체를 공개 또는 비공개로 설정하거나 개별 객체를 공개로 설정하도록 선택할 수 있습니다. 액세스 키를 생성하고, 버킷에 인스턴스를 연결하고, 다른 AWS 계정에 액세스 권한을 부여하여 버킷에 대한 액세스 권한을 부여할 수도 있습니다. 자세한 내용은 Amazon Lightsail 객체 스토리지에 대한 보안 모범 사례 및 Amazon Lightsail의 버킷 권한 이해를 참조하세요.

   버킷 액세스 권한에 대해 알아본 후 다음 가이드를 참조하여 버킷에 대한 액세스 권한을 부여합니다.

   - Amazon Lightsail의 버킷에 대한 퍼블릭 액세스 차단
   - Amazon Lightsail에서 버킷 액세스 권한 구성
   - Amazon Lightsail에서 버킷의 개별 객체에 대한 액세스 권한 구성
   - Amazon Lightsail에서 버킷의 액세스 키 생성

- Amazon Lightsail에서 버킷에 대한 리소스 액세스 구성
- Amazon Lightsail에서 버킷의 크로스 계정 액세스 구성

5. 버킷에 대한 액세스 로깅을 활성화하는 방법과 액세스 로그를 사용하여 버킷의 보안을 감사하는 방법에 대해 알아봅니다. 자세한 내용은 다음 안내서를 참조하세요.
   - Amazon Lightsail 객체 스토리지 서비스의 버킷에 대한 액세스 로깅
   - Amazon Lightsail 객체 스토리지 서비스에서 버킷에 대한 액세스 로그 형식
   - Amazon Lightsail 객체 스토리지 서비스에서 버킷에 대한 액세스 로깅 활성화
   - Amazon Lightsail에서 버킷에 대한 액세스 로그 사용하여 요청을 식별
6. Lightsail에서 버킷을 관리할 수 있는 기능을 사용자에게 부여하는 IAM 정책을 생성합니다. 자세한 내용은 Amazon Lightsail에서 버킷을 관리하기 위한 IAM 정책을 참조하세요.
7. 버킷의 객체에 라벨을 지정하고 식별하는 방식에 대해 알아봅니다. 자세한 내용은 Amazon Lightsail에서 객체 키 이름 이해를 참조하세요.
8. 파일을 업로드하고 버킷의 객체를 관리하는 방법에 대해 알아봅니다. 자세한 내용은 다음 안내서를 참조하세요.
   - Amazon Lightsail에서 버킷으로 파일 업로드
   - 멀티파트 업로드를 사용하여 Amazon Lightsail에서 버킷으로 파일 업로드
   - Amazon Lightsail에서 버킷의 객체 확인
   - Amazon Lightsail에서 버킷의 객체 복사 또는 이동
   - Amazon Lightsail에서 버킷의 객체 다운로드
   - Amazon Lightsail에서 버킷의 객체 필터링
   - Amazon Lightsail에서 버킷의 객체 태그 지정
   - Amazon Lightsail에서 버킷의 객체 삭제
9. 객체 버전 관리를 활성화하면 버킷에 저장된 모든 버전의 객체를 모두 보존, 검색 및 복원할 수 있습니다. 자세한 내용은 Amazon Lightsail에서 버킷의 객체 버전 사용 설정 및 사용 중지를 참조하세요.
10. 객체 버전 관리를 활성화한 후 버킷에 있는 객체의 이전 버전을 복원할 수 있습니다. 자세한 내용은 Amazon Lightsail에서 버킷의 이전 객체 버전 복원을 참조하세요.
11. 버킷 사용률을 모니터링합니다. 자세한 내용은 Amazon Lightsail에서 버킷에 대한 지표 보기를 참조하세요.
12. 버킷 사용률이 임곗값을 초과할 때 알림을 받도록 버킷 지표에 대한 경보를 구성합니다. 자세한 내용은 Amazon Lightsail에서 버킷 지표 경보 생성을 참조하세요.
13. 스토리지 및 네트워크 전송이 부족한 경우 버킷의 스토리지 플랜을 변경합니다. 자세한 내용은 Amazon Lightsail에서 버킷 플랜 변경을 참조하세요.

14.버킷을 다른 리소스에 연결하는 방법에 대해 알아봅니다. 자세한 내용은 다음 자습서를 참조하세요.

- 자습서: WordPress 인스턴스를 Amazon Lightsail 버킷에 연결
- 자습서: Lightsail 콘텐츠 전송 네트워크 배포와 함께 Amazon Lightsail 버킷 사용

15.버킷을 더 이상 사용하지 않는 경우 삭제할 수 있습니다. 자세한 내용은 Amazon Lightsail에서 버킷 삭제를 참조하세요.

Lightsail 배포에 대한 데이터 전송 할당량 조정

Amazon Lightsail 배포를 생성할 때 월별 데이터 전송 할당량과 배포 비용을 정하는 배포 플랜을 선택합니다. 배포가 플랜의 월별 데이터 전송 할당량보다 많은 데이터를 전송하는 경우 초과 요금이 청구됩니다. 초과 요금에 대한 자세한 내용은 Lightsail 요금 페이지를 참조하세요.

요금을 초과로 사용하지 않으려면 현재 배포 플랜을 배포가 월별 할당량을 초과하기 전에 매달 더 많은 데이터 전송량을 제공하는 다른 플랜으로 변경하면 됩니다. 각 AWS 결제 주기 동안 배포의 플랜을 한 번만 변경할 수 있습니다. 이 가이드에서는 배포 플랜을 변경하는 방법을 안내합니다.

배포에 대한 자세한 내용은 콘텐츠 전송 네트워크 배포를 참조하세요.

배포 플랜 변경하기

배포 플랜을 변경하려면 다음 절차를 완료합니다.

1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 네트워킹을 선택합니다.
3. 현재 월별 데이터 전송을 보려는 배포의 이름을 선택합니다.
4. 배포의 관리 페이지에서 세부 정보(Details) 탭을 선택합니다.
5. 페이지의 데이터 전송(Data transfer) 섹션에서 배포 플랜 변경(Change distribution plan)을 선택합니다.
6. 확인 프롬프트에서 예, 변경합니다(Yes, change)를 선택하여 배포 플랜을 변경할지 확인합니다.
7. 다음 프롬프트에서 배포에 대한 새 플랜을 선택하고 플랜 선택(Select plan)을 선택합니다.
8. 다음 프롬프트에서 예, 적용합니다(Yes, apply)를 선택하여 배포에 새 플랜을 적용할지 확인합니다. 배포에 새 플랜을 적용하지 않으려면 아니요, 돌아갑니다(No, go back)를 선택합니다.

Lightsail 배포용 사용자 지정 도메인과 함께 콘텐츠 제공

배포에서 등록된 도메인 이름을 사용하려면 Amazon Lightsail 배포에 사용자 지정 도메인을 활성화해야 합니다. 사용자 지정 도메인을 활성화하기 전에 배포는 처음 생성 시 배포에 연결된 기본 도메인(예: `123456abcdef.cloudfront.net`)의 트래픽만 수락합니다. 사용자 지정 도메인을 활성화할 때는 배포에서 사용하려는 도메인에 대해 생성한 Lightsail SSL/TLS 인증서를 선택해야 합니다. 사용자 지정 도메인을 활성화한 후에는 배포가 선택한 인증서와 연결된 모든 도메인의 트래픽을 수락합니다.

> **Important**
>
> 배포당 1번에 1개의 인증서만 사용할 수 있습니다. 배포에서 사용자 지정 도메인을 비활성화하면 사용자 지정 도메인을 다시 활성화할 때까지 등록된 도메인의 HTTPS 트래픽을 배포에서 더 이상 처리할 수 없습니다.
>
> SSL/TLS 인증서와 연결된 도메인 이름은 Amazon CloudFront 서비스의 배포를 포함하여 모든 Amazon Web Services(AWS) 계정의 다른 배포에서 사용할 수 없습니다. 도메인에 대한 인증서를 생성할 수 있지만, 배포와 함께 사용할 수는 없습니다.

배포에 대한 자세한 내용은 콘텐츠 전송 네트워크 배포를 참조하세요.

사전 조건

시작하기 전에 Lightsail 배포를 생성해야 합니다. 자세한 내용은 배포 생성을 참조하세요.

또한, 배포용 SSL/TLS 인증서를 생성하고 검증해야 합니다. 자세한 내용은 배포용 SSL/TLS 인증서 생성 및 배포용 SSL/TLS 인증서 검증을 참조하세요.

배포용 사용자 지정 도메인 활성화

배포의 사용자 지정 도메인을 활성화하려면 다음 절차를 완료하세요.

1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 네트워킹을 선택합니다.
3. 사용자 지정 도메인을 활성화하려는 배포의 이름을 선택합니다.
4. 배포의 관리 페이지에서 사용자 지정 도메인(Custom domains) 탭을 선택합니다.
5. 인증서 연결(Attach certificate)을 선택합니다.

   인증서가 없는 경우 먼저 도메인에 대한 SSL/TLS 인증서를 생성하고 검증한 후 배포에 인증서를 연결해야 합니다. 자세한 내용은 배포용 SSL/TLS 인증서 생성을 참조하세요.

6. 드롭다운 메뉴가 표시되면 배포에 사용할 도메인에 대해 유효한 인증서를 선택합니다.
7. 인증서 정보가 올바른지 확인한 다음 Attach(연결)를 선택합니다.
8. 배포의 Status(상태)가 Updating(업데이트 중)으로 변경됩니다. 상태가 Enabled(활성화됨)로 변경된 후 인증서의 도메인이 Custom domains(사용자 지정 도메인) 섹션에 표시됩니다.
9. Add domain assignment(도메인 할당 추가)를 선택하여 도메인을 배포에 연결합니다.
10. 인증서와 DNS 정보가 올바른지 확인한 다음 Add assignment(할당 추가)를 선택합니다. 잠시 후 배포에서 선택한 도메인의 트래픽을 수락하기 시작합니다.

주제

- 사용자 지정 도메인을 Lightsail 배포로 지정
- Lightsail 배포용 SSL/TLS 인증서 업데이트
- Lightsail 배포용 사용자 지정 도메인 사용 중지
- 배포의 기본 도메인을 Lightsail 컨테이너 서비스에 추가

사용자 지정 도메인을 Lightsail 배포로 지정

배포에서 사용자 지정 도메인을 활성화한 후에는 등록된 도메인 이름을 Amazon Lightsail 배포로 연결해야 합니다. 배포와 함께 사용하는 인증서에 지정된 각 도메인의 DNS 영역에 별칭 레코드를 추가하면 됩니다. 추가하는 모든 레코드를 배포의 기본 도메인(예: `123456abcdef.cloudfront.net`)으로 지정해야 합니다.

이 가이드에서는 Lightsail DNS 영역을 사용하여 도메인을 배포로 연결하는 절차를 안내합니다. Domain.com 또는 GoDaddy와 같은 다른 DNS 호스팅 제공자를 사용하여 도메인을 배포로 연결하는 절차와 유사할 수 있습니다. Lightsail DNS 영역에 대한 자세한 내용은 DNS를 참조하세요.

배포에 대한 자세한 내용은 배포 생성을 참조하세요.

목차

- 1단계: 사전 조건 완료
- 2단계: 배포의 기본 도메인 가져오기
- 3단계: 도메인의 DNS 영역에 레코드 추가

1단계: 사전 조건 완료

시작하기 전에 Lightsail 배포에서 사용자 지정 도메인을 활성화해야 합니다. 자세한 내용은 배포용 사용자 지정 도메인 사용을 참조하세요.

2단계: 배포의 기본 도메인 가져오기

별칭 레코드를 도메인의 DNS에 추가할 때 지정하는 배포의 기본 도메인 이름을 가져오려면 다음 절차를 완료하세요.

1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 네트워킹을 선택합니다.
3. 기본 도메인 이름을 가져오려는 배포의 이름을 선택합니다.
4. 배포 관리 페이지의 헤더 섹션에 있는 배포의 기본 도메인 이름을 기록해 둡니다. 배포의 기본 도메인 이름은 `123456abcdef.cloudfront.net`과 유사합니다.

   이 값은 도메인의 DNS에서 별칭 레코드의 일부로 추가해야 합니다. 나중에 참조할 수 있도록 이 값을 복사하여 텍스트 파일에 붙여넣는 것이 좋습니다. 이 자습서의 다음 3단계: 도메인의 DNS 영역에 레코드 추가 섹션으로 이동합니다.

3단계: 도메인의 DNS 영역에 레코드 추가

도메인의 DNS 영역에 레코드를 추가하려면 다음 절차를 완료하세요.

1. 왼쪽 탐색 창에서 도메인 및 DNS를 선택합니다.
2. 페이지의 DNS 영역(DNS zones) 섹션에서 도메인의 트래픽을 배포로 연결할 레코드를 추가하려는 도메인 이름을 선택합니다.
3. DNS records(DNS 레코드) 탭을 선택합니다. 그런 다음 Add record(레코드 추가)를 선택합니다.
4. 배포로 연결하려는 도메인 유형에 따라 다음 단계 중 하나를 완료합니다.

   - 주소 (A) 레코드를 선택하여 도메인의 정점(예: `example.com`)을 배포로 연결합니다.

     도메인의 정점에 대한 A 레코드가 DNS 영역에 이미 있으면 다른 A 레코드를 추가하는 대신 해당 기존 레코드를 편집해야 합니다.

   - 하위 도메인(예: `website.example.com`)을 배포로 연결하려면 표준 이름(CNAME)을 선택합니다.

5. A 레코드를 추가하는 경우 확인(Resolves to) 텍스트 상자에서 배포 이름을 선택합니다. CNAME 레코드를 추가하는 경우 다음으로 매핑(Maps to) 텍스트 상자에 배포의 기본 도메인 이름을 입력합니다.

   > **Note**
   >
   > DNS 영역에 A 레코드를 추가하고 배포 이름을 선택하면 실제로 주소 레코드와 다른 별칭 레코드를 추가하게 됩니다. Lightsail에서는 일반적으로 다른 DNS 호스팅 제공자에서 필요로 하는 단계를 별도로 수행하지 않고도 별칭 레코드를 쉽게 추가할 수 있습니다.

6. DNS 영역에 레코드를 저장하려면 저장 아이콘을 선택합니다.

   이 단계를 반복하여 배포와 함께 사용 중인 인증서의 도메인에 대한 별도의 DNS 레코드를 추가합니다. 인터넷의 DNS를 통해 변경 사항이 전파될 때까지 기다립니다. 몇 분 후 도메인이 배포로 연결되었는지 확인합니다. 또한, 배포를 테스트해야 합니다. 자세한 내용은 아래 배포 테스트를 참조하세요.

Lightsail 배포용 SSL/TLS 인증서 업데이트

Amazon Lightsail 배포에서 이용하는 사용자 지정 도메인을 다른 도메인 또는 도메인 집합으로 변경할 수 있습니다. 이렇게 하려면 먼저 배포에 사용할 도메인의 새 SSL/TLS 인증서를 생성해야 합니다. 자세한 내용은 배포용 SSL/TLS 인증서 생성을 참조하세요. 새 인증서를 검증한 후 이전 인증서를 새 인증서로 교체하여 배포의 사용자 지정 도메인을 변경합니다.

배포에 대한 자세한 내용은 배포 생성을 참조하세요.

배포용 사용자 지정 도메인 변경

배포의 사용자 지정 도메인을 변경하려면 다음 절차를 수행하세요.

1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 네트워킹을 선택합니다.
3. 사용자 지정 도메인을 변경할 배포의 이름을 선택합니다.
4. 배포의 관리 페이지에서 사용자 지정 도메인(Custom domains) 탭을 선택합니다.
5. 현재 배포에 연결된 SSL/TLS 인증서를 분리합니다.

   배포 상태가 In progress(진행 중)로 변경됩니다.

6. 배포 상태가 다시 Enabled(활성화됨)로 변경된 후 Attach certificate(인증서 연결)을 선택합니다.
7. 드롭다운 메뉴가 표시되면 배포에 사용할 도메인에 대해 유효한 인증서를 선택합니다.
8. 인증서 정보가 올바른지 확인한 다음 Attach(연결)를 선택합니다.
9. 도메인의 DNS에 도메인 할당을 추가하여 도메인을 배포에 연결합니다.

   배포의 Status(상태)가 Updating(업데이트 중)으로 변경됩니다. 상태가 Ready(준비됨) 변경된 후 인증서의 Custom domains(사용자 지정 도메인) 섹션에 표시됩니다. Add domain assignment(도메인 할당 추가)를 선택하여 도메인을 배포에 연결합니다.
10. Add assignment(할당 추가)를 선택합니다. 잠시 후 배포에서 선택한 도메인의 트래픽을 수락하기 시작합니다.
11. 저장을 선택합니다.

Lightsail 배포용 사용자 지정 도메인 사용 중지

Amazon Lightsail 배포용 사용자 지정 도메인을 사용 중지하면 배포에 등록된 도메인 이름을 더 이상 사용할 수 없습니다. 사용자 지정 도메인을 사용 중지하도록 설정하면 배포에서 처음 생성 시 배포에 연결된 기본 도메인(예: `123456abcdef.cloudfront.net`)의 트래픽만 수락하며, 이전에 연결된 사용자 지정 도메인의 트래픽에는 403 오류가 표시됩니다.

배포에 대한 자세한 내용은 콘텐츠 전송 네트워크 배포를 참조하세요.

배포용 사용자 지정 도메인 사용 중지

배포용 사용자 지정 도메인을 사용 중지하려면 다음 절차를 완료하세요.

1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 네트워킹을 선택합니다.
3. 사용자 지정 도메인을 사용 중지할 배포의 이름을 선택합니다.
4. 배포의 관리 페이지에서 사용자 지정 도메인(Custom domains) 탭을 선택합니다.

   Custom domains(사용자 지정 도메인) 페이지에 현재 배포에 연결된 SSL/TLS 인증서가 표시됩니다(있는 경우).
5. 다음 옵션 중 하나를 선택하세요.

   1. 이전에 선택한 도메인을 선택 취소하거나 배포와 연결된 도메인을 더 선택하려면 Configure distribution domains(배포 도메인 구성)를 선택합니다.

2. 분리를 선택하여 배포에서 인증서를 분리하고, 에서 연결된 모든 도메인을 제거합니다.

6. 사용자 지정 도메인 사용 중지 요청이 제출되고 배포 상태가 진행 중(In progress)으로 변경됩니다. 잠시 후 배포의 상태가 사용(Enabled)으로 변경됩니다.

사용자 지정 도메인을 사용 중지하도록 설정하면 배포에서 처음 생성 시 배포에 연결된 기본 도메인(예: `123456abcdef.cloudfront.net`)의 트래픽만 수락하며, 이전에 연결된 사용자 지정 도메인의 트래픽에는 403 오류가 표시됩니다. 도메인의 트래픽이 다른 리소스로 연결되도록 도메인의 DNS 레코드를 업데이트해야 합니다.

배포의 기본 도메인을 Lightsail 컨테이너 서비스에 추가

Amazon Lightsail 컨테이너 서비스를 콘텐츠 전송 네트워크(CDN) 배포의 오리진으로 선택할 수 있습니다. 그러면 배포가 컨테이너 서비스에 호스팅된 웹 사이트 또는 웹 애플리케이션을 캐싱하여 제공합니다. Lightsail 배포를 Lightsail 컨테이너 서비스와 함께 사용하는 경우, Lightsail은 배포의 기본 도메인 이름을 컨테이너 서비스에 사용자 지정 도메인으로 자동으로 추가합니다. 이렇게 하면 트래픽을 배포와 컨테이너 서비스 간에 라우팅할 수 있습니다. 그러나 다음과 같은 경우에는 반드시 이 가이드에 설명된 단계를 따라 배포의 기본 도메인 이름을 컨테이너 서비스에 수동으로 추가해야 합니다.

- 문제가 발생하여 컨테이너 서비스에 배포의 기본 도메인 이름이 자동으로 추가되지 않는 경우.
- 컨테이너 서비스에 Lightsail 배포가 아닌 다른 배포를 사용하는 경우.

AWS Command Line Interface ()를 사용해야만 배포의 기본 도메인 이름을 컨테이너 서비스에 수동으로 추가할 수 있습니다AWS CLI. 컨테이너 서비스에 대한 자세한 내용은 컨테이너 서비스를 참조하세요. 배포에 대한 자세한 내용은 객체 스토리지를 참조하세요.

배포의 기본 도메인을 컨테이너 서비스에 추가

AWS Command Line Interface ()를 Lightsail 사용하여에서 배포의 기본 도메인을 컨테이너 서비스에 추가하려면 다음 절차를 완료합니다AWS CLI. `update-container-service` 명령을 사용하여 이 작업을 수행할 수 있습니다. 자세한 내용은 AWS CLI 명령 레퍼런스의 update-container-service를 참조하세요.

> **Note**
>
> 이 절차를 계속하기 Lightsail 전에 AWS CLI 를 설치하고에 대해 구성해야 합니다. 자세한 내용은 에서 작동 AWS CLI 하도록 구성을 Lightsail참조하세요.

1. 명령 프롬프트 또는 터미널 창을 엽니다.
2. 다음 명령 중 하나를 입력하여 배포의 기본 도메인을 컨테이너 서비스에 추가합니다.

> **Note**
>
> 컨테이너 서비스에 사용자 지정 도메인을 추가한 경우에는 사용자 지정 도메인과 배포의 기본 도메인을 모두 지정해야 합니다.

사용자 지정 도메인이 컨테이너 서비스에 구성되어 있지 않습니다:

```
aws lightsail update-container-service --service-name ContainerServiceName --
public-domain-names '{"_": ["DistributionDefaultDomain"]}'
```

하나 이상의 사용자 지정 도메인이 컨테이너 서비스에 구성되어 있습니다:

```
aws lightsail update-container-service --service-name ContainerServiceName
 --public-domain-names '{"CertificateName": ["ExistingCustomDomain"],"_":
 ["DistributionDefaultDomain"]}'
```

명령에서 아래 예 텍스트를 사용자의 값으로 대체합니다.

- *ContainerServiceName* - 배포의 오리진으로 지정했던 Lightsail 컨테이너 서비스의 이름입니다.
- *DistributionDefaultDomain* - 컨테이너 서비스를 오리진으로 사용하는, 배포의 기본 도메인입니다. 예를 들어 `example123.cloudfront.net`입니다.
- *CertificateName*" - 현재 컨테이너 서비스에 연결되어 있는 사용자 지정 도메인(있는 경우)의 Lightsail 인증서 이름입니다. 컨테이너 서비스에 연결된 사용자 지정 도메인이 없는 경우에는 사용자 지정 도메인이 컨테이너 서비스에 구성되어 있지 않습니다라는 레이블이 지정된 명령을 사용합니다.
- *DistributionDefaultDomain* - 컨테이너 서비스에 현재 연결된 사용자 지정 도메인입니다.

예시:

- 사용자 지정 도메인이 컨테이너 서비스에 구성되어 있지 않습니다:

```
aws lightsail update-container-service --service-name ContainerServiceName --
public-domain-names '{"_": ["example123.cloudfront.net"]}'
```

- 하나 이상의 사용자 지정 도메인이 컨테이너 서비스에 구성되어 있습니다:

```
aws lightsail update-container-service --service-name ContainerServiceName
 --public-domain-names '{"example-com": ["example.com"],"_":
 ["example123.cloudfront.net"]}'
```

Lightsail 배포에 대한 요청 및 응답 동작 관리

이 가이드에서는 요청을 오리진으로 처리 및 전달하고 오리진에서 응답을 처리할 때 Amazon Lightsail 배포의 동작 방식을 설명합니다. 배포에 대한 자세한 내용은 콘텐츠 전송 네트워크 배포를 참조하세요.

주제

- 배포에서 요청을 처리하고 오리진에 요청을 전달하는 방법
- 배포에서 오리진의 응답을 처리하는 방법

배포에서 요청을 처리하고 오리진에 요청을 전달하는 방법

이 섹션에서는 배포에서 최종 사용자 요청을 처리하고 요청을 오리진으로 전달하는 방법에 대한 정보를 제공합니다.

목차

- Authentication
- 캐싱 기간
- 클라이언트 IP 주소
- 클라이언트측 SSL 인증
- 압축
- 조건부 요청
- 쿠키
- 교차 오리진 리소스 공유(CORS)

- 암호화(Encryption)
- 본문이 포함되는 GET 요청
- HTTP 메소드
- HTTP 요청 헤더 및 배포 동작
- HTTP 버전
- 요청의 최대 길이 및 최대 URL 길이
- OCSP 스테이플링
- 지속적인 연결
- 프로토콜
- 쿼리 문자열
- 오리진 연결 제한 시간 및 시도 횟수
- 오리진 응답 제한 시간
- 동일 객체에 대한 동시 요청(트래픽 스파이크)
- User-agent 헤더

Authentication

DELETE, GET, HEAD, PATCH, POST, PUT 요청에 대해 Authorization 헤더를 오리진에 전달하도록 배포를 구성하는 경우, 클라이언트 인증을 요청하도록 오리진 서버를 구성할 수 있습니다.

OPTIONS 요청에 대해서는 다음 배포 설정을 사용하는 경우에만 클라이언트 인증을 요청하도록 오리진 서버를 구성할 수 있습니다.

- Authorization 헤더를 오리진에 전달하도록 배포를 구성합니다.
- OPTIONS 요청에 대한 응답을 캐싱하지 않도록 배포를 구성합니다.

HTTP 또는 HTTPS를 사용하여 오리진에 요청을 전달하도록 배포를 구성할 수 있습니다.

캐싱 기간

다음과 같은 방법으로 배포에서 다른 요청을 오리진에 전달하기 전에 객체를 배포의 캐시에 보관하는 기간을 제어할 수 있습니다.

- Cache-Control 또는 Expires 헤더 파일을 각 객체에 추가하도록 오리진을 구성합니다.

- 캐시 수명 주기(TTL)를 기본값인 1일로 설정합니다.

자세한 내용은 배포 고급 설정을 참조하세요.

클라이언트 IP 주소

최종 사용자가 배포에 요청을 보내고 X-Forwarded-For 요청 헤더를 포함하지 않는 경우, 배포는 TCP 연결에서 최종 사용자의 IP 주소를 가져오고 IP 주소를 포함하는 X-Forwarded-For 헤더를 추가하고 오리진에 요청을 전달합니다. 예를 들어, 배포가 TCP 연결에서 IP 주소 192.0.2.2를 가져오면 오리진에 다음 헤더를 전달합니다.

```
X-Forwarded-For: 192.0.2.2
```

최종 사용자가 배포에 요청을 보내고 X-Forwarded-For 요청 헤더를 포함하는 경우, 배포는 TCP 연결에서 최종 사용자의 IP 주소를 가져와 X-Forwarded-For 헤더의 끝에 첨부하고 오리진에 요청을 전달합니다. 예를 들어, 최종 사용자 요청에 X-Forwarded-For: 192.0.2.4,192.0.2.3이 포함되고 배포가 TCP 연결에서 IP 주소 192.0.2.2를 가져오면 오리진에 다음 헤더를 전달합니다.

```
X-Forwarded-For: 192.0.2.4,192.0.2.3,192.0.2.2
```

로드 밸런서, 웹 애플리케이션 방화벽, 역방향 프록시, 침입 방지 시스템 및 API Gateway와 같은 일부 애플리케이션은 해당 요청을 전달한 배포 엣지 서버의 IP 주소를 X-Forwarded-For 헤더의 끝에 추가합니다. 예를 들어, 배포가 ELB에 전달하는 요청에 X-Forwarded-For: 192.0.2.2를 포함시키고 배포 엣지 서버의 IP 주소가 192.0.2.199이면 인스턴스가 수신하는 요청에는 다음과 같은 헤더가 들어 있습니다.

```
X-Forwarded-For: 192.0.2.2,192.0.2.199
```

> **Note**
>
> X-Forwarded-For 헤더에는 IPv4 주소(예: 192.0.2.44)와 IPv6 주소(예: 2001:0db8:85a3:0000:0000:8a2e:0370:7334)가 포함됩니다.

클라이언트측 SSL 인증

Lightsail 배포는 클라이언트 측 SSL 인증서와 클라이언트 인증을 지원하지 않습니다. 오리진이 클라이언트 측 인증서를 요청하는 경우 배포에서는 이 요청을 삭제합니다.

압축

Lightsail 배포에서는 `Accept-Encoding` 필드 값이 `"identity"` 및 `"gzip"`인 요청을 전달합니다.

조건부 요청

배포가 엣지 캐시에서 만료된 객체에 대한 요청을 수신하면 이 요청을 오리진으로 전달하여 최신 버전의 객체를 가져오거나 배포 엣지 캐시에 이미 최신 버전이 있는지 오리진의 확인을 받습니다. 대개 오리진에서 마지막으로 배포에 객체를 보낼 때는 `ETag` 값, `LastModified` 값 또는 두 값 모두를 응답에 포함하여 보냅니다. 배포는 배포가 오리진으로 전달하는 새 요청에 다음 중 하나 또는 모두를 추가합니다.

- 만료된 버전의 객체에 대한 `If-Match` 값을 포함하는 `If-None-Match` 또는 `ETag` 헤더
- 만료된 버전의 객체에 대한 `If-Modified-Since` 값을 포함하는 `LastModified` 헤더

오리진에서는 이 정보를 사용하여 객체가 배포에 업데이트되는지와, 이에 따라 전체 객체를 배포에 반환할지 아니면 HTTP 304 상태 코드(수정되지 않음)만 반환할지를 결정합니다.

쿠키

쿠키를 오리진에 전달하도록 배포를 구성할 수 있습니다. 자세한 내용은 배포 고급 설정을 참조하세요.

교차 오리진 리소스 공유(CORS)

배포에서 cross-origin 리소스 공유 설정을 준수하도록 하려는 경우, `Origin` 헤더를 오리진에 전달하도록 오리진을 구성합니다.

암호화(Encryption)

최종 사용자가 HTTPS를 사용하여 배포에 연결하도록 하고 배포에서 HTTP 또는 HTTPS를 사용하여 요청을 오리진에 전달하도록 할 수 있습니다.

배포에서는 SSLv3, TLSv1.0, TLSv1.1 및 TLSv1.2 프로토콜을 사용하여 HTTPS 요청을 오리진에 전달합니다. 다른 버전의 SSL 및 TLS는 지원되지 않습니다.

본문이 포함되는 GET 요청

최종 사용자 `GET` 요청에 본문이 포함되는 경우, 배포에서는 HTTP 상태 코드 403(금지됨)을 최종 사용자에게 반환합니다.

HTTP 메소드

지원되는 모든 HTTP 메서드를 허용하도록 배포를 구성하는 경우, 배포에서는 최종 사용자의 다음 요청을 수락하고 이를 오리진에 전달합니다.

- DELETE
- GET
- HEAD
- OPTIONS
- PATCH
- POST
- PUT

배포는 항상 GET 및 HEAD 요청에 대한 응답을 캐싱합니다. OPTIONS 요청에 대한 응답을 캐싱하지 않도록 배포를 구성할 수도 있습니다. 배포에서는 다른 메서드를 사용하는 요청에 대한 응답을 캐싱하지 않습니다.

오리진에서 이러한 메소드를 처리할지를 구성하는 방법에 대한 자세한 내용은 오리진 설명서를 참조하세요.

> **Important**
>
> 배포에서 지원되는 모든 HTTP 메서드를 허용하고 오리진에 전달하도록 배포를 구성한 경우, 모든 메서드를 처리하도록 오리진 서버를 구성합니다. 예를 들어, POST를 사용할 의도로 이러한 메서드를 허용 및 전달하도록 배포를 구성한 경우, 원치 않는 리소스를 최종 사용자가 삭제할 수 없도록 DELETE 요청을 적절히 처리하여 오리진 서버를 구성해야 합니다. 자세한 내용은 HTTP 서버에 대한 설명서를 참조하십시오.

HTTP 요청 헤더 및 배포 동작

다음 목록에는 예외 사항과 함께 오리진으로 전달할 수 있는 HTTP 요청 헤더가 포함되어 있습니다. 각 헤더의 목록에는 다음과 같은 정보가 포함되어 있습니다.

- 지원 여부 - 해당 헤더에 대해 헤더 값에 따라 객체를 캐싱하도록 배포를 구성할 수 있는지 여부입니다.

Date 및 User-Agent 헤더의 값에 따라 객체를 캐싱하도록 배포를 구성할 수 있지만, 권장되지는 않습니다. 이러한 헤더는 여러 가지 값을 가질 수 있으며, 헤더의 값에 따른 캐싱으로 인해 배포에서 오리진에 전달되는 요청이 눈에 띄게 증가할 수 있습니다.

- 구성하지 않은 경우의 동작 - 헤더를 오리진에 전달하도록 배포를 구성하지 않은 경우의 배포 동작으로, 배포에서 헤더 값에 따라 객체를 캐싱하는 원인이 됩니다.

- 헤더 - 기타 정의 헤더

  지원 여부 - 지원됨

  구성하지 않은 경우의 동작 - 배포에서 오리진에 헤더를 전달합니다.

- 헤더 - Accept

  지원 여부 - 지원됨

  구성하지 않은 경우의 동작 - 배포에서 헤더를 제거합니다.

- 헤더 - Accept-Charset

  지원 여부 - 지원됨

  구성하지 않은 경우의 동작 - 배포에서 헤더를 제거합니다.

- 헤더 - Accept-Encoding

  지원 여부 - 지원됨

  구성하지 않은 경우의 동작 - 값에 gzip이 포함된 경우, 배포에서 오리진에 Accept-Encoding: gzip을 전달합니다. 이 값에 gzip이 포함되어 있지 않은 경우, 배포에서는 오리진으로 요청을 전달하기 전에 Accept-Encoding 헤더 필드를 제거합니다.

- 헤더 - Accept-Language

  지원 여부 - 지원됨

  구성하지 않은 경우의 동작 - 배포에서 헤더를 제거합니다.

- 헤더 - Authorization

  지원 여부 - 지원됨

  구성하지 않은 경우의 동작:

- GET 및 HEAD 요청 - 배포에서는 요청을 오리진에 전달하기 전에 Authorization 헤더 필드를 제거합니다.
- OPTIONS 요청 - OPTIONS 요청에 대한 응답을 캐싱하도록 배포를 구성한 경우, 배포에서는 요청을 오리진에 전달하기 전에 Authorization 헤더 필드를 제거합니다.

  OPTIONS 요청에 대한 응답을 캐싱하도록 배포를 구성하지 않은 경우, 배포에서는 Authorization 헤더 필드를 오리진에 전달합니다.
- DELETE, PATCH, POST 및 PUT 요청 - 배포에서는 요청을 오리진에 전달하기 전에 헤더 필드를 제거하지 않습니다.

- 헤더 - Cache-Control

  지원 여부 - 지원되지 않음

  구성하지 않은 경우의 동작 - 배포에서 오리진에 헤더를 전달합니다.
- 헤더 - CloudFront-Forwarded-Proto

  지원 여부 - 지원됨

  구성하지 않은 경우의 동작 - 배포에서 오리진에 요청을 전달하기 전에 헤더를 추가하지 않습니다.
- 헤더 - CloudFront-Is-Desktop-Viewer

  지원 여부 - 지원됨

  구성하지 않은 경우의 동작 - 배포에서 오리진에 요청을 전달하기 전에 헤더를 추가하지 않습니다.
- 헤더 - CloudFront-Is-Mobile-Viewer

  지원 여부 - 지원됨

  구성하지 않은 경우의 동작 - 배포에서 오리진에 요청을 전달하기 전에 헤더를 추가하지 않습니다.
- 헤더 - CloudFront-Is-Tablet-Viewer

  지원 여부 - 지원됨

  구성하지 않은 경우의 동작 - 배포에서 오리진에 요청을 전달하기 전에 헤더를 추가하지 않습니다.
- 헤더 - CloudFront-Viewer-Country

  지원 여부 - 지원됨

  구성하지 않은 경우의 동작 - 배포에서 오리진에 요청을 전달하기 전에 헤더를 추가하지 않습니다.

- 헤더 - Connection

  지원 여부 - 지원되지 않음

  구성하지 않은 경우의 동작 - 배포에서 요청을 오리진으로 전달하기 전에 이 헤더를 Connection: Keep-Alive로 바꿉니다.

- 헤더 - Content-Length

  지원 여부 - 지원되지 않음

  구성하지 않은 경우의 동작 - 배포에서 오리진에 헤더를 전달합니다.

- 헤더 - Content-MD5

  지원 여부 - 지원됨

  구성하지 않은 경우의 동작 - 배포에서 오리진에 헤더를 전달합니다.

- 헤더 - Content-Type

  지원 여부 - 지원됨

  구성하지 않은 경우의 동작 - 배포에서 오리진에 헤더를 전달합니다.

- 헤더 - Cookie

  지원 여부 - 지원되지 않음

  구성하지 않은 경우의 동작- 쿠키를 전달하도록 배포를 구성하면 Cookie 헤더 필드가 오리진으로 전달됩니다. 구성하지 않은 경우 배포에서는 Cookie 헤더 필드를 제거합니다.

- 헤더 - Date

  지원 여부 - 지원되나 권장되지 않음

  구성하지 않은 경우의 동작 - 배포에서 오리진에 헤더를 전달합니다.

- 헤더 - Expect

  지원 여부 - 지원됨

  구성하지 않은 경우의 동작 - 배포에서 헤더를 제거합니다.

- 헤더 - From

  지원 여부 - 지원됨

구성하지 않은 경우의 동작 - 배포에서 오리진에 헤더를 전달합니다.

- 헤더 - Host

  지원 여부 - 지원됨

  구성하지 않은 경우의 동작 - 배포에서 요청된 객체와 연결된 오리진의 도메인 이름으로 값을 설정합니다.

- 헤더 - If-Match

  지원 여부 - 지원됨

  구성하지 않은 경우의 동작 - 배포에서 오리진에 헤더를 전달합니다.

- 헤더 - If-Modified-Since

  지원 여부 - 지원됨

  구성하지 않은 경우의 동작 - 배포에서 오리진에 헤더를 전달합니다.

- 헤더 - If-None-Match

  지원 여부 - 지원됨

  구성하지 않은 경우의 동작 - 배포에서 오리진에 헤더를 전달합니다.

- 헤더 - If-Range

  지원 여부 - 지원됨

  구성하지 않은 경우의 동작 - 배포에서 오리진에 헤더를 전달합니다.

- 헤더 - If-Unmodified-Since

  지원 여부 - 지원됨

  구성하지 않은 경우의 동작 - 배포에서 오리진에 헤더를 전달합니다.

- 헤더 - Max-Forwards

  지원 여부 - 지원되지 않음

  구성하지 않은 경우의 동작 - 배포에서 오리진에 헤더를 전달합니다.

- 헤더 - Origin

지원 여부 - 지원됨

구성하지 않은 경우의 동작 - 배포에서 오리진에 헤더를 전달합니다.

- 헤더 - Pragma

  지원 여부 - 지원되지 않음

  구성하지 않은 경우의 동작 - 배포에서 오리진에 헤더를 전달합니다.

- 헤더 - Proxy-Authenticate

  지원 여부 - 지원되지 않음

  구성하지 않은 경우의 동작 - 배포에서 헤더를 제거합니다.

- 헤더 - Proxy-Authorization

  지원 여부 - 지원되지 않음

  구성하지 않은 경우의 동작 - 배포에서 헤더를 제거합니다.

- 헤더 - Proxy-Connection

  지원 여부 - 지원되지 않음

  구성하지 않은 경우의 동작 - 배포에서 헤더를 제거합니다.

- 헤더 - Range

  지원 여부 - 기본적으로 지원됨

  구성하지 않은 경우의 동작 - 배포에서 오리진에 헤더를 전달합니다.

- 헤더 - Referer

  지원 여부 - 지원됨

  구성하지 않은 경우의 동작 - 배포에서 헤더를 제거합니다.

- 헤더 - Request-Range

  지원 여부 - 지원되지 않음

  구성하지 않은 경우의 동작 - 배포에서 오리진에 헤더를 전달합니다.

- 헤더 - TE

지원 여부 - 지원되지 않음

구성하지 않은 경우의 동작 - 배포에서 헤더를 제거합니다.

- 헤더 - Trailer

  지원 여부 - 지원되지 않음

  구성하지 않은 경우의 동작 - 배포에서 헤더를 제거합니다.

- 헤더 - Transfer-Encoding

  지원 여부 - 지원되지 않음

  구성하지 않은 경우의 동작 - 배포에서 오리진에 헤더를 전달합니다.

- 헤더 - Upgrade

  지원 여부 - 지원되지 않음(WebSocket 연결 제외)

  구성하지 않은 경우의 동작 - WebSocket 연결이 설정되지 않은 경우 배포에서 헤더를 제거합니다.

- 헤더 - User-Agent

  지원 여부 - 지원되나 권장되지 않음

  구성하지 않은 경우의 동작 - 배포에서 이 헤더 필드의 값을 Amazon CloudFront로 바꿉니다.

- 헤더 - Via

  지원 여부 - 지원됨

  구성하지 않은 경우의 동작 - 배포에서 오리진에 헤더를 전달합니다.

- 헤더 - Warning

  지원 여부 - 지원됨

  구성하지 않은 경우의 동작 - 배포에서 오리진에 헤더를 전달합니다.

- 헤더 - X-Amz-Cf-Id

  지원 여부 - 지원되지 않음

  구성하지 않은 경우의 동작 - 배포에서 요청을 오리진으로 전달하기 전에 최종 사용자 요청에 헤더를 추가합니다. 헤더 값에는 요청을 고유하게 식별하는 암호화된 문자열이 포함됩니다.

- 헤더 - `X-Edge-*`

  지원 여부 - 지원되지 않음

  구성하지 않은 경우의 동작 - 배포에서 모든 `X-Edge-*` 헤더를 제거합니다.
- 헤더 - `X-Forwarded-For`

  지원 여부 - 지원됨

  구성하지 않은 경우의 동작 - 배포에서 오리진에 헤더를 전달합니다.
- 헤더 - `X-Forwarded-Proto`

  지원 여부 - 지원되지 않음

  구성하지 않은 경우의 동작 - 배포에서 헤더를 제거합니다.
- 헤더 - `X-Real-IP`

  지원 여부 - 지원되지 않음

  구성하지 않은 경우의 동작 - 배포에서 헤더를 제거합니다.

HTTP 버전

배포에서는 HTTP/1.1을 사용하여 오리진에 요청을 전달합니다.

요청의 최대 길이 및 최대 URL 길이

경로, 쿼리 문자열(있는 경우), 헤더를 모두 포함한 요청의 최대 길이는 20,480바이트입니다.

배포에서는 이 요청으로부터 URL을 구성합니다. 이 URL의 최대 길이는 8,192바이트입니다.

요청 또는 URL이 이 최대값을 초과할 경우, 배포에서는 HTTP 상태 코드 413(요청 엔터티가 너무 큼)을 반환한 후 최종 사용자와의 TCP 연결을 종료합니다.

OCSP 스테이플링

최종 사용자가 객체에 대한 HTTPS 요청을 제출할 때 배포 또는 최종 사용자는 인증 기관(CA)을 통해 도메인의 SSL 인증서가 해지되지 않았는지 확인해야 합니다. OCSP 스테이플링은 배포에서 인증서를 검증하고 CA로부터 응답을 캐싱할 수 있도록 지원하여 클라이언트가 CA를 통해 직접 인증서를 검증하지 않아도 되므로 인증서 검증 속도가 향상됩니다.

배포에서 동일한 도메인에 있는 객체에 대한 수많은 HTTPS 요청을 수신하는 경우 OCSP 스테이플링으로 인한 성능 개선이 더욱 확연히 드러납니다. 배포 엣지 로케이션의 각 서버는 개별적인 검증 요청을 제출해야 합니다. 배포에서 같은 도메인에 대해 다수의 HTTPS 요청을 수신하는 경우, 엣지 로케이션의 각 서버에서는 곧 CA로부터 SSL 핸드셰이크의 패킷에 '스테이플'할 수 있다는 응답을 받습니다. 최종 사용자의 인증서가 유효하다는 조건을 충족하면 배포에서 요청된 객체를 제공할 수 있습니다. 배포가 엣지 로케이션에서 많은 양의 트래픽을 받지 않는 경우, 아직 CA를 통해 인증서를 검증하지 않은 서버로 새로운 요청이 리디렉션될 가능성이 큽니다. 이 경우 최종 사용자가 검증 단계를 개별적으로 수행해야 하며, 배포 서버에서는 객체를 제공합니다. 해당 배포 서버에서 또한 검증 요청을 CA에 제출하고, 다음에 동일한 도메인 이름을 포함한 요청을 수신하면 CA로부터 검증 응답을 받습니다.

지속적인 연결

배포가 오리진으로부터 응답을 받는 경우, 해당 시간 동안 다른 요청이 도착하면 몇 초 정도 연결을 유지하려고 합니다. 지속적인 연결을 유지하면 TCP 연결 재설정 및 이후 요청에 대한 별도의 TLS 핸드쉐이크 수행에 필요한 시간이 절약됩니다.

프로토콜

배포는 Lightsail 콘솔의 오리진 프로토콜 정책 필드 값을 기준으로 오리진 서버로 HTTP 또는 HTTPS 요청을 전달합니다. Lightsail 콘솔에는 HTTP 전용 및 HTTPS 전용 옵션이 있습니다.

HTTP 전용(HTTP Only) 또는 HTTPS 전용(HTTPS Only)을 지정하는 경우, 배포에서는 최종 사용자 요청의 프로토콜과 관계없이 지정된 프로토콜을 사용하여 오리진으로 요청을 전달합니다.

> **Important**
>
> 배포가 HTTPS 프로토콜을 사용하여 오리진으로 요청을 전달하고 오리진 서버가 잘못된 인증서 또는 자체 서명한 인증서를 반환하는 경우, 배포에서는 TCP 연결을 끊습니다.

쿼리 문자열

배포에서 쿼리 문자열 파라미터를 오리진에 전달할지를 구성할 수 있습니다.

오리진 연결 제한 시간 및 시도 횟수

기본적으로 배포는 30초(각 10초씩 3번 시도) 동안 기다린 후 최종 사용자에게 오류 응답을 반환합니다.

오리진 응답 제한 시간

오리진 응답 제한 시간(오리진 읽기 제한 시간 또는 오리진 요청 제한 시간이라고도 함)은 다음 두 값에 모두 적용됩니다.

- 배포가 오리진에 요청을 전달한 후 응답을 기다리는 시간(초).
- 배포에서 오리진으로부터 응답 패킷을 수신한 후 다음 패킷을 수신할 때까지 대기하는 시간(초).

배포 동작은 최종 사용자 요청의 HTTP 메서드에 따라 달라집니다.

- GET 및 HEAD 요청 - 오리진에서 응답 제한 시간 내에 응답하지 않거나 응답이 중지된 경우, 배포에서는 연결을 끊습니다. 지정된 오리진 연결 시도 횟수가 1보다 많으면 배포가 다시 완료 응답을 수신하려고 시도합니다. 배포는 오리진 연결 시도 설정 값에 따라 최대 3회까지 시도합니다. 오리진이 마지막 시도에 응답하지 않는 경우, 배포에서는 동일한 오리진의 콘텐츠에 대해 다른 요청을 받을 때까지 다시 시도하지 않습니다.
- DELETE, OPTIONS, PATCH, PUT 및 POST 요청 - 오리진에서 30초 내에 응답하지 않는 경우, 배포에서는 연결을 끊고 오리진에 다시 연결을 시도하지 않습니다. 필요한 경우 클라이언트는 요청을 다시 제출할 수 있습니다.

동일 객체에 대한 동시 요청(트래픽 스파이크)

배포 엣지 로케이션에서 객체에 대한 요청을 받을 때 객체가 현재 캐시에 있지 않거나 객체가 만료된 경우, 배포에서는 즉시 요청을 오리진으로 보냅니다. 트래픽 스파이크가 있는 경우 - 오리진에서 첫 번째 요청에 응답하기 전에 동일 객체에 대한 추가 요청이 엣지 로케이션에 도착하는 경우 배포는 객체에 대한 추가 요청을 오리진에 전달하기 전에 작업을 일시 중단합니다. 일반적으로는 후속 요청에 응답하기 전에 첫 번째 요청에 대한 응답이 배포 엣지 로케이션에 도착하게 됩니다. 이러한 일시 중단은 오리진 서버에 불필요하게 로드가 걸리는 것을 줄여 줍니다. 요청 헤더나 쿠키에 따라 캐싱하도록 배포를 구성하는 등의 이유로 추가 요청이 동일하지 않은 경우, 배포에서는 모든 고유한 요청을 오리진에 전달합니다.

User-agent 헤더

배포에서 사용자가 콘텐츠를 보는 데 이용 중인 디바이스에 따라 여러 객체 버전을 캐싱하도록 하려는 경우, 다음과 같이 하나 이상의 헤더를 오리진에 전달하도록 배포를 구성하는 것이 좋습니다.

- `CloudFront-Is-Desktop-Viewer`
- `CloudFront-Is-Mobile-Viewer`

- CloudFront-Is-SmartTV-Viewer
- CloudFront-Is-Tablet-Viewer

User-Agent 헤더의 값에 따라 배포에서는 요청을 오리진에 전달하기 전에 이러한 헤더의 값을 true 또는 false로 설정합니다. 디바이스가 둘 이상의 범주에 해당하는 경우 둘 이상의 값이 true일 수 있습니다. 예를 들어, 일부 태블릿 디바이스의 경우 배포에서 CloudFront-Is-Mobile-Viewer 및 CloudFront-Is-Tablet-Viewer를 모두 true로 설정할 수 있습니다.

User-Agent 헤더의 값에 따라 객체를 캐싱하도록 배포를 구성할 수 있지만, 권장되지는 않습니다. User-Agent 헤더는 여러 가지 값을 가질 수 있으며, 그러한 값에 따른 캐싱으로 인해 배포에서 오리진에 전달되는 요청이 눈에 띄게 증가할 수 있습니다.

User-Agent 헤더의 값에 따라 객체를 캐싱하도록 배포를 구성하지 않는 경우, 배포에서는 오리진에 요청을 전달하기 전에 다음 값으로 User-Agent 헤더를 추가합니다.

```
User-Agent = Amazon CloudFront
```

배포에서는 최종 사용자의 요청에 User-Agent 헤더가 포함되어 있는지와 관계없이 이 헤더를 추가합니다. 최종 사용자의 요청에 User-Agent 헤더가 포함되어 있는 경우, 배포에서는 이를 제거합니다.

배포에서 오리진의 응답을 처리하는 방법

이 섹션에는 배포가 오리진의 응답을 처리하는 방법에 대한 정보가 포함되어 있습니다.

목차

- 100-Continue 응답
- 캐싱
- 취소된 요청
- 콘텐츠 협상
- 쿠키
- TCP 연결 끊김
- 배포에서 제거하거나 대체하는 HTTP 응답 헤더
- 최대 파일 크기
- 오리진 사용 불가
- 리디렉션

- 전송 인코딩

100-Continue 응답

오리진은 배포에 두 개 이상의 100-Continue 응답을 전송할 수 없습니다. 첫 번째 100-Continue 응답 후에 배포는 HTTP 200 OK 응답을 예상합니다. 오리진이 첫 번째 응답 후 또 다른 100-Continue 응답을 전송하면 배포는 오류를 반환합니다.

캐싱

- 오리진에서 `Date` 및 `Last-Modified` 헤더 필드에 유효하고 정확한 값을 설정했는지 확인합니다.
- 최종 사용자의 요청에 `If-Match` 또는 `If-None-Match` 요청 헤더 필드가 포함된 경우, `ETag` 응답 헤더 필드를 설정합니다. `ETag` 값을 지정하지 않은 경우, 배포에서는 후속 `If-Match` 또는 `If-None-Match` 헤더를 무시합니다.
- 배포에서는 오리진의 응답으로 보통 `Cache-Control: no-cache` 헤더를 신뢰합니다. 예외 사항은 동일한 객체에 대한 동시 요청(트래픽 스파이크)을 참조하세요.

취소된 요청

객체가 엣지 캐시에 있지 않은 경우 최종 사용자가 배포에서 오리진으로부터 객체를 가져온 뒤 요청된 객체를 제공하기 전에 브라우저 닫기 등으로 세션을 종료하면, 배포에서는 엣지 로케이션의 객체를 캐싱하지 않습니다.

콘텐츠 협상

오리진이 응답에서 `Vary:*`를 반환하는 경우와 해당 캐시 동작의 Minimum TTL 값이 0인 경우, 배포에서는 객체를 캐싱하지만 오리진에 객체의 모든 후속 요청을 전달하여 캐시에 객체의 최신 버전이 포함되어 있음을 확인합니다. 배포에는 `If-None-Match` 또는 `If-Modified-Since`와 같은 조건부 헤더가 포함되지 않습니다. 결과적으로 오리진은 모든 요청에 응답하여 객체를 배포에 반환합니다.

오리진이 응답에서 `Vary:*`를 반환하는 경우와 해당 캐시 동작의 최소 TTL 값이 다른 값인 경우, CloudFront에서는 배포에서 삭제 또는 대체하는 HTTP 응답 헤더에 설명된 `Vary` 헤더를 처리합니다.

쿠키

캐시 동작에 대한 쿠키를 사용하는 경우 오리진에서 객체를 통해 쿠키를 반환하면 배포에서는 객체와 쿠키를 모두 캐싱합니다. 이를 통해 객체에 대한 캐싱 가능성이 줄어듭니다.

TCP 연결 끊김

오리진이 객체를 배포에 반환하는 동안 배포와 오리진 간의 TCP 연결이 끊어진 경우, 배포 동작은 오리진이 응답에 Content-Length 헤더를 포함했는지 여부에 따라 달라집니다.

- Content-Length 헤더 - 배포가 오리진에서 객체를 가져올 때 최종 사용자에게 객체를 반환합니다. 그러나 Content-Length 헤더의 값이 객체의 크기와 일치하지 않으면 배포가 객체를 캐싱하지 않습니다.
- Transfer-Encoding: 청크 분할 - 배포가 오리진에서 객체를 가져올 때 최종 사용자에게 객체를 반환합니다. 그러나 청크 응답이 완전하지 않을 경우 배포가 객체를 캐싱하지 않습니다.
- Content-Length 헤더 없음 - 배포가 최종 사용자에게 객체를 반환하고 캐싱하지만, 객체가 불완전할 수 있습니다. Content-Length 헤더가 없으면 TCP 연결이 실수로 또는 고의로 끊어졌는지 여부를 배포가 확인할 수 없습니다.

배포가 부분적인 객체를 캐싱하는 것을 방지하기 위해 Content-Length 헤더를 추가하도록 HTTP 서버를 구성하는 것이 좋습니다.

배포에서 제거하거나 대체하는 HTTP 응답 헤더

배포는 오리진에서 최종 사용자에게 응답을 전달하기 전에 다음 헤더 필드를 제거하거나 업데이트합니다.

- Set-Cookie - 쿠키를 전달하도록 배포를 구성하면 Set-Cookie 헤더 필드가 클라이언트에 전달됩니다.
- Trailer
- Transfer-Encoding - 오리진에서 이 헤더 필드를 반환할 경우 배포에서는 최종 사용자에게 응답을 반환하기 전에 값을 chunked로 설정합니다.
- Upgrade
- Vary – 다음을 참조하십시오.
  - 디바이스별 헤더 중 하나를 오리진(CloudFront-Is-Desktop-Viewer, CloudFront-Is-Mobile-Viewer, CloudFront-Is-SmartTV-Viewer, CloudFront-Is-Tablet-Viewer)에 전달하도록 배포를 구성하고 Vary:User-Agent를 배포에 반환하도록 오리진을 구성하는 경우, 배포에서는 최종 사용자에게 Vary:User-Agent를 반환합니다.
  - Vary 헤더에서 Accept-Encoding 또는 Cookie를 포함하도록 오리진을 구성하는 경우, 배포는 최종 사용자에 대한 응답에 값을 포함합니다.

  - 헤더 허용 목록을 오리진에 전달하도록 배포를 구성하는 경우와 Vary 헤더(예: Vary:Accept-Charset,Accept-Language)의 배포에 헤더 이름을 반환하도록 오리진을 구성하는 경우, 배포에서는 최종 사용자에게 Vary 헤더를 해당 값과 함께 반환합니다.
  - 배포가 Vary 헤더의 * 값을 처리하는 방법에 대한 자세한 내용은 콘텐츠 협상을 참조하세요.
  - Vary 헤더에 다른 값을 포함하도록 오리진을 구성하는 경우, 배포는 최종 사용자에게 응답을 반환하기 전에 값을 제거합니다.
- Via - 배포는 최종 사용자에 대한 응답으로 값을 다음과 같이 설정합니다.

  ```
  Via: http-version alphanumeric-string.cloudfront.net (CloudFront)
  ```

  예를 들어, 클라이언트가 HTTP/1.1을 통해 요청한 경우 값은 다음과 같습니다.

  ```
  Via: 1.1 1026589cc7887e7a0dc7827b4example.cloudfront.net (CloudFront)
  ```

최대 파일 크기

배포에서 최종 사용자에게 반환하는 응답 본문의 최대 크기는 20GB입니다. 여기에는 Content-Length 헤더 값으로 지정하지 않은 조각난 전송 응답이 포함됩니다.

오리진 사용 불가

오리진 서버를 사용할 수 없고 배포에서 엣지 캐시에 있지만 만료된(Cache-Control max-age 명령에 지정된 기간이 지났다는 이유 등으로) 객체에 대한 요청을 받은 경우, 배포에서는 만료된 버전의 객체를 제공하거나 사용자 지정 오류 페이지를 표시합니다.

경우에 따라 자주 요청되지는 않는 객체는 제거되고 엣지 캐시에서 더 이상 사용할 수 없게 됩니다. 배포에서는 제거된 객체를 제공할 수 없습니다.

리디렉션

오리진 서버에 있는 객체의 위치를 변경하는 경우, 요청을 새 위치로 리디렉션하도록 웹 서버를 구성할 수 있습니다. 리디렉션을 구성한 뒤 처음으로 최종 사용자가 객체에 대한 요청을 제출할 때 배포에서는 요청을 오리진에 전송하고 오리진에서는 리디렉션으로 응답합니다(예: 302 Moved Temporarily). 배포에서는 이 리디렉션을 캐싱하고 최종 사용자에게 반환합니다. 배포에서는 리디렉션을 따라가지 않습니다.

다음 위치 중 하나로 요청을 리디렉션하도록 웹 서버를 구성할 수 있습니다.

- 오리진 서버에 있는 객체의 새 URL입니다. 최종 사용자가 새 URL에 대한 리디렉션을 따라갈 때 최종 사용자는 배포를 우회하고 오리진으로 직행합니다. 따라서 오리진에 있는 객체의 새 URL로 요청을 리디렉션하지 않는 것이 좋습니다.
- 객체에 대한 새 배포 URL. 최종 사용자가 새 배포 URL을 포함한 요청을 제출할 때 배포는 오리진의 새 위치에서 객체를 가져오고, 이를 엣지 로케이션에서 캐싱하여 최종 사용자에게 객체를 반환합니다. 이 객체에 대한 후속 요청은 엣지 로케이션에서 제공됩니다. 이를 통해 최종 사용자의 오리진에서의 객체 요청과 관련된 시간 지연과 로드 발생을 피할 수 있습니다. 그러나 객체에 대한 새로운 요청이 발생할 때마다 배포에 대한 두 요청 비용이 부과됩니다.

전송 인코딩

Lightsail 배포에서는 `Transfer-Encoding` 헤더에 대해 `chunked` 값만 지원합니다. 오리진에서 `Transfer-Encoding: chunked`를 반환하는 경우, 배포에서는 객체를 엣지 로케이션에서 수신한 객체로 클라이언트에 반환하고 후속 요청에 대해 chunked 형식의 객체를 캐싱합니다.

최종 사용자가 `Range GET` 요청을 하고 오리진이 `Transfer-Encoding: chunked`를 반환하는 경우, 배포에서는 최종 사용자에게 요청한 범위가 아닌 전체 객체를 반환합니다.

응답의 콘텐츠 길이를 사전에 결정할 수 없는 경우 chunked 인코딩을 사용하는 것이 좋습니다. 자세한 내용은 TCP 연결 끊김을 참조하세요.

Lightsail 배포의 콘텐츠 캐싱 검증

이 설명서에서는 Amazon Lightsail 배포가 오리진의 콘텐츠를 캐싱하고 제공하고 있는지 테스트하는 방법에 대해 알아봅니다. 등록된 도메인 이름을 배포에 추가하고 나서 테스트를 수행해야 합니다. 배포에 대한 자세한 내용은 콘텐츠 전송 네트워크 배포를 참조하세요.

배포 테스트

배포를 테스트하려면 다음 절차를 완료하세요. 이 절차에서는 Chrome 웹 브라우저를 사용합니다. 다른 브라우저에서도 비슷한 단계를 활용할 수 있습니다.

1. Chrome 웹 브라우저를 엽니다.
2. 브라우저 창의 오른쪽 상단 모서리에 있는 Chrome 메뉴(Chrome Menu)를 열고 추가 도구(More Tools) > 개발자 도구(Developer Tools)를 선택합니다.

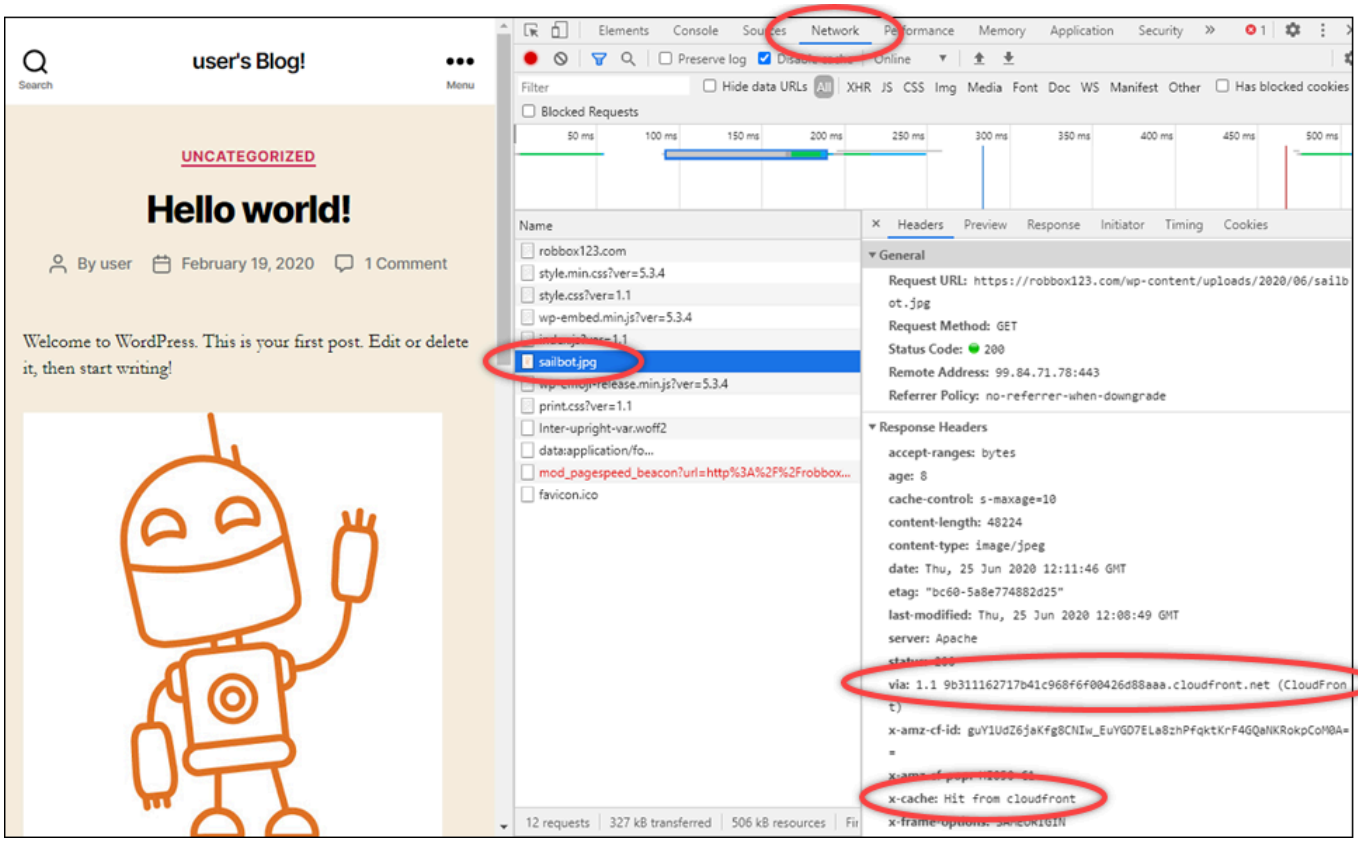
   Option + ⌘ + J(macOS) 또는 Shift + Ctrl + J (Windows/Linux) 단축키를 사용할 수도 있습니다.
3. 개발자 도구 창에서 네트워크(Network) 탭을 선택합니다.

4. 배포의 도메인(예:`https://www.example.com`)으로 이동합니다.

   Chrome 개발자 도구의 네트워크(Network) 탭이 웹 사이트의 객체 목록으로 채워집니다.

5. 이미지 파일(.jpg, .png, .gif)과 같은 정적 객체를 선택합니다.

6. 나타나는 헤더 패널에서 `via` 및 `x-cache` 헤더 모두 CloudFront를 언급해야 합니다. 이것으로 배포가 오리진의 콘텐츠를 캐싱하고 제공한다는 것을 확인할 수 있습니다.



Amazon Lightsail의 네트워킹 리소스

Lightsail 네트워킹 리소스는 사용자와 외부 서비스가 Lightsail 인스턴스에 연결하는 방식을 개선합니다.

로드 밸런서

로드 밸런서를 만들어 중복성을 높이거나 더 많은 트래픽을 처리할 수 있습니다. 자세한 내용은 로드 밸런서를 참조하세요.

고정 IP

고정 IP 주소를 생성하면 인스턴스를 재부팅할 때마다 동일한 IP 주소를 유지할 수 있습니다. 자세한 내용은 고정 IP 주소를 참조하세요.

Lightsail 리소스의 IP 주소 보기 및 관리

Lightsail 인스턴스 및 기타 Lightsail 리소스와 IP 주소를 사용하여 통신할 수 있습니다. 예를 들어, 인스턴스의 퍼블릭 IP 주소를 사용하여 인스턴스의 네트워크 상태를 확인(ping 사용)하고, 인스턴스에 SSH 연결을 설정하고, 사용자 지정 도메인 이름에서 인스턴스로 트래픽을 라우팅할 수 있습니다. Lightsail 리소스의 IP 주소로 보다 다양한 작업을 수행할 수 있습니다.

Lightsail 인스턴스, 컨테이너 서비스, 로드 밸런서는 IPv4 및 IPv6 주소 지정 프로토콜을 모두 지원합니다. 이러한 리소스는 기본적으로 IPv4 주소 지정 프로토콜을 사용하므로, 이 동작은 비활성화할 수 없습니다. 필요에 따라 인스턴스, 컨테이너 서비스, 로드 밸런서에 IPv6를 활성화할 수 있습니다.

이 가이드에서는 Lightsail의 IP 주소에 대해 알아야 하는 내용을 다룹니다.

목차

- 인스턴스의 프라이빗 및 퍼블릭 IPv4 주소
- 인스턴스의 고정 IP 주소
- 인스턴스, 컨테이너 서비스, CDN 배포 및 로드 밸런서의 IPv6

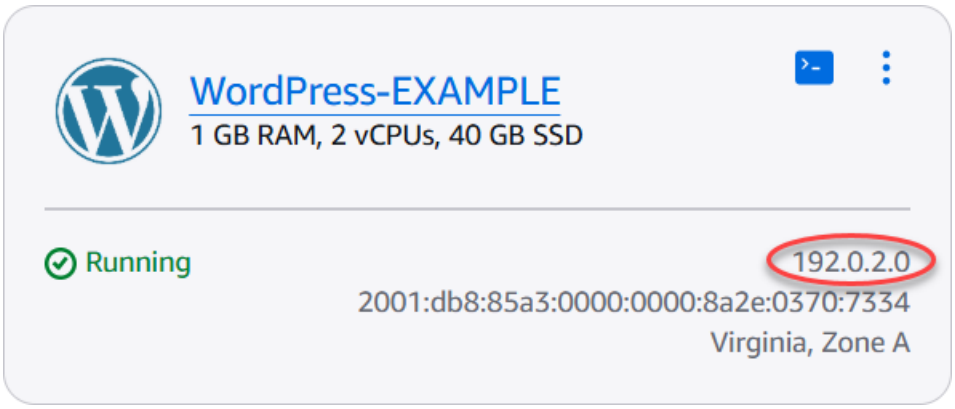
인스턴스의 프라이빗 및 퍼블릭 IPv4 주소

Lightsail 인스턴스를 만들면 퍼블릭 및 프라이빗 IPv4 주소가 할당됩니다. 퍼블릭 IP 주소는 인터넷에 액세스할 수 있지만, 프라이빗 IP 주소는 동일한 AWS 리전에 있는 Lightsail 계정의 리소스에만 액세스할 수 있습니다.

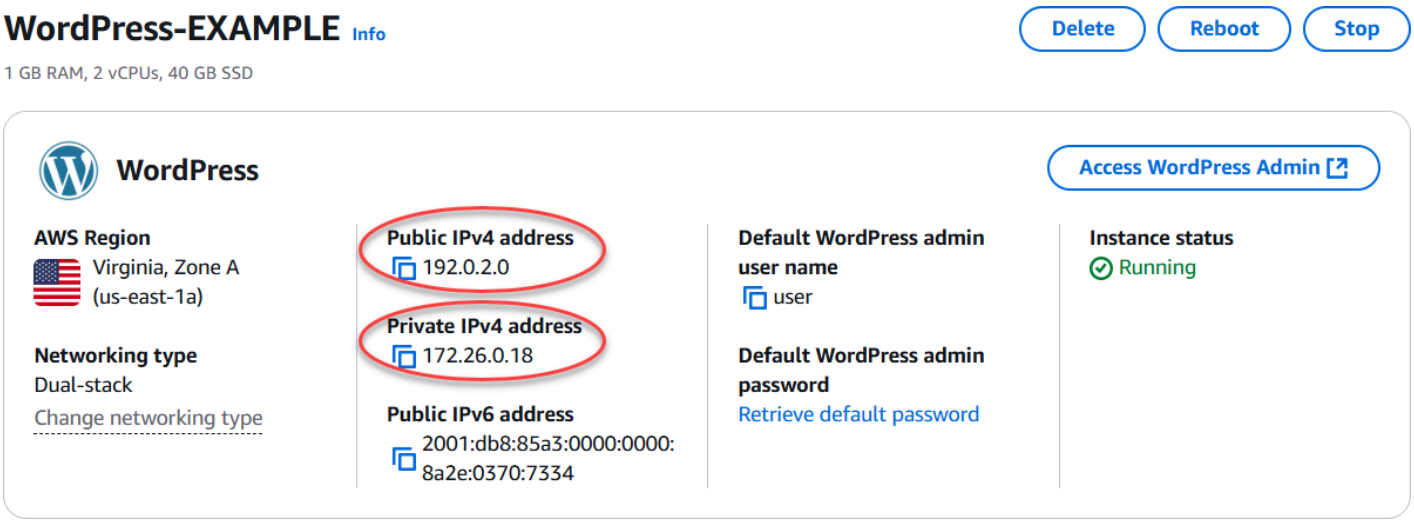
> **Note**
>
> 인스턴스의 프라이빗 IP 주소는 동일한 AWS 리전의 다른 AWS 리소스에 액세스할 수 있지만, VPC 피어링을 활성화한 경우 Lightsail 계정 외부에서 액세스할 수 있습니다. 자세한 내용은 Lightsail 외부의 AWS 리소스와 함께 작동하도록 Amazon VPC 피어링 설정을 참조하세요.

인스턴스의 IP 주소는 Lightsail 콘솔의 다음 영역에 표시됩니다.

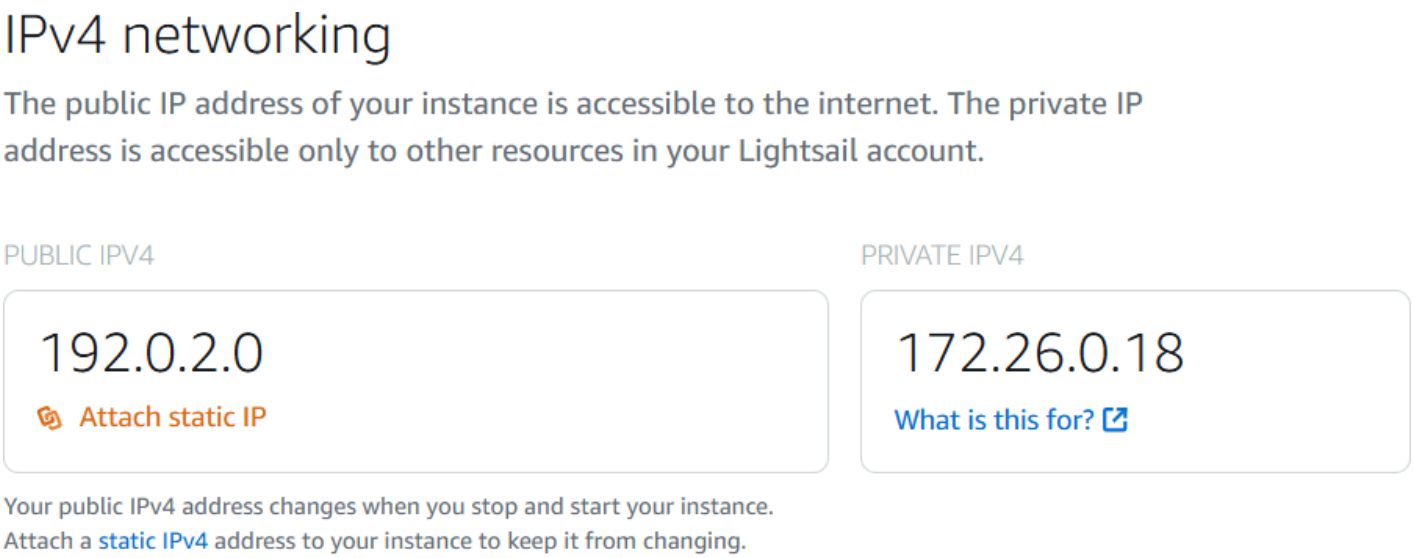
- 다음 예는 Lightsail 홈 페이지에 있는 인스턴스의 퍼블릭 IPv4 주소를 보여줍니다.



- 다음 예는 인스턴스 관리 페이지의 머리말 영역에 있는 인스턴스의 퍼블릭 및 프라이빗 IPv4 주소를 보여줍니다.



- 다음 예는 인스턴스 관리 페이지의 네트워킹 탭에서 인스턴스의 퍼블릭 및 프라이빗 IPv4 주소를 보여줍니다.



인스턴스의 IPv4 주소를 사용할 때는 다음 사항에 유의하세요.

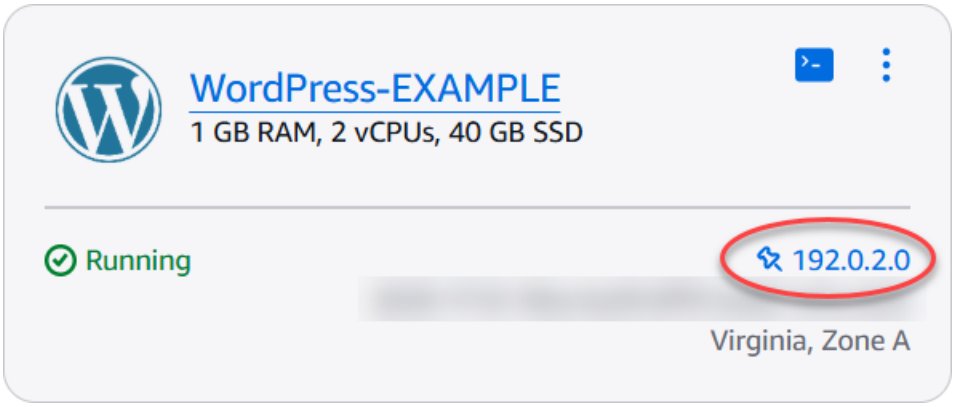
- 인스턴스의 퍼블릭 IP 주소는 변경될 수 있습니다. 인스턴스에 변경되지 않는 IP 주소를 지정하려면 고정 IP를 연결하면 됩니다. 자세한 내용은 가이드의 인스턴스의 고정 IP 주소 섹션을 참조하세요.
- Lightsail은 기본적으로 IPv4 주소를 사용합니다. 단, 필요에 따라 2021년 1월 12일 이전에 생성된 일부 Lightsail 리소스에서 IPv6를 활성화할 수 있습니다. 2021년 1월 12일 이후에 생성된 리소스는 기본적으로 IPv6를 활성화하도록 설정되어 있습니다. 자세한 내용은 가이드의 인스턴스, 컨테이너 서비스, CDN 배포 및 로드 밸런서의 IPv6 섹션을 참조하세요.
- 인스턴스 방화벽에 연결할 수 있는 트래픽을 제어하는 규칙을 추가합니다. 자세한 내용은 인스턴스 방화벽을 참조하세요.

인스턴스의 고정 IPv4 주소

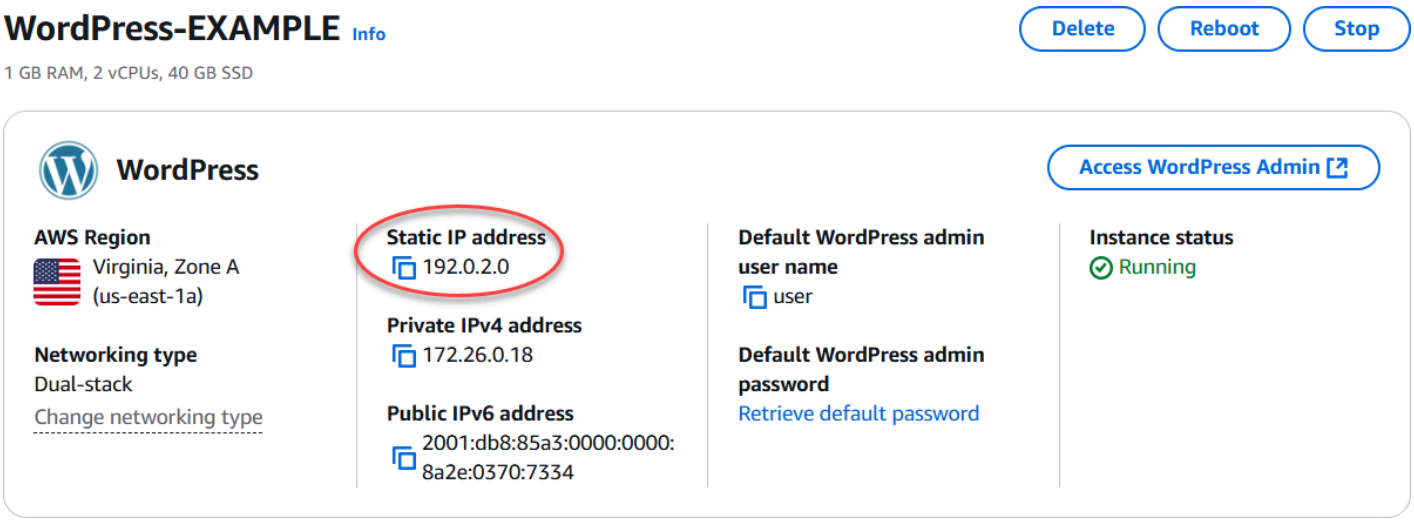
인스턴스를 생성할 때 인스턴스에 할당된 기본 퍼블릭 IPv4 주소는 인스턴스를 중지하고 시작할 때 변경됩니다. 고정 IPv4 주소를 생성하여 인스턴스에 연결할 수도 있습니다. 고정 IPv4 주소는 인스턴스의 기본 퍼블릭 IPv4 주소를 대체하며 인스턴스를 중지하고 시작할 때 동일하게 유지됩니다. 한 인스턴스에 한 개의 고정 IP를 연결할 수 있습니다. 자세한 내용은 고정 IP를 생성하여 인스턴스에 연결을 참조하세요.

고정 IP를 생성하여 인스턴스에 연결하면 Lightsail 콘솔의 다음 영역에 표시됩니다.

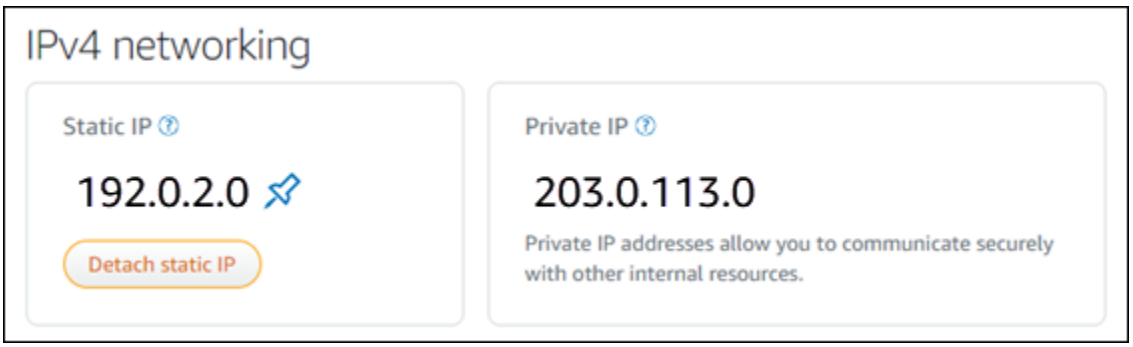
- 다음 예는 Lightsail 홈 페이지에 있는 인스턴스의 고정 IP 주소를 보여줍니다. 압정 아이콘은 퍼블릭 IP 주소가 고정되어 있음을 나타냅니다.



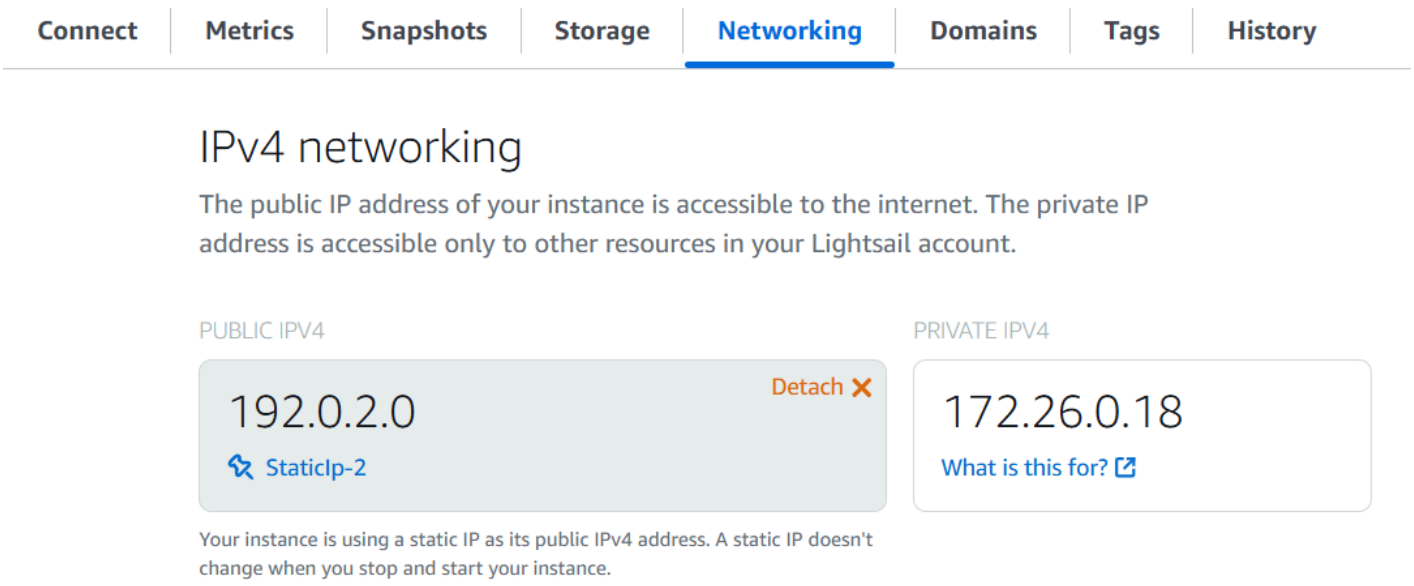
- 다음 예는 인스턴스 관리 페이지의 머리말 영역에 있는 인스턴스의 고정 IP 주소를 보여줍니다. 압정 아이콘은 퍼블릭 IP 주소가 고정되어 있음을 나타냅니다.



- 다음 예는 인스턴스 관리 페이지의 네트워킹(Networking) 탭에 있는 인스턴스의 고정 IP 주소를 보여줍니다. 기본 퍼블릭 IP 주소는 더 이상 나열되지 않으며 고정 IP 주소로 대체되었습니다. 압정 아이콘은 퍼블릭 IP 주소가 고정되어 있음을 나타냅니다.



- 다음 예와 같이 Lightsail 홈 페이지의 네트워킹(Networking) 탭으로 이동하여 생성한 고정 IP를 모두 볼 수 있습니다.



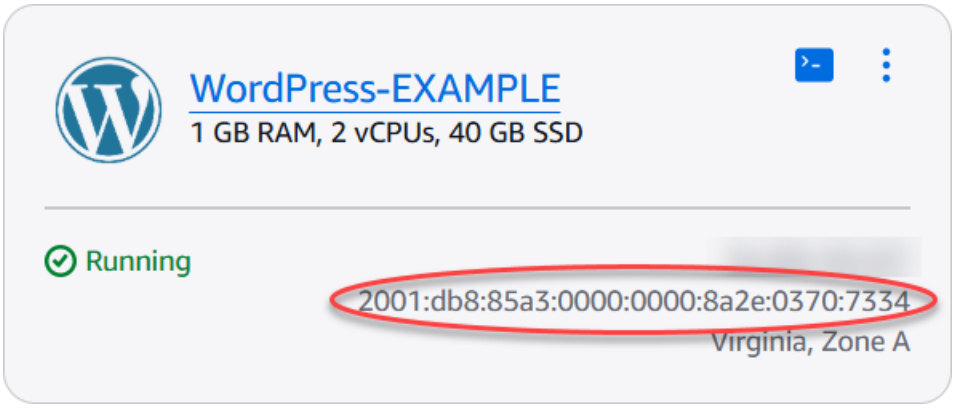
인스턴스, 컨테이너 서비스, CDN 배포 및 로드 밸런서의 IPv6

IPv6는 2021년 1월 12일 이후에 생성된 Lightsail 인스턴스, 컨테이너 서비스, CDN 배포 및 로드 밸런서에 대해 기본적으로 사용하도록 설정됩니다. 필요에 따라 2021년 1월 12일 이전에 생성된 리소스에서 IPv6를 실행할 수 있습니다. 특정 리소스에 대해 IPv6를 활성화하면 Lightsail에서 해당 리소스에 IPv6 주소를 자동으로 할당하므로 IPv6 주소를 직접 선택하거나 지정할 수 없습니다. 자세한 내용은 IPv6 사용 또는 사용 중지를 참조하세요.

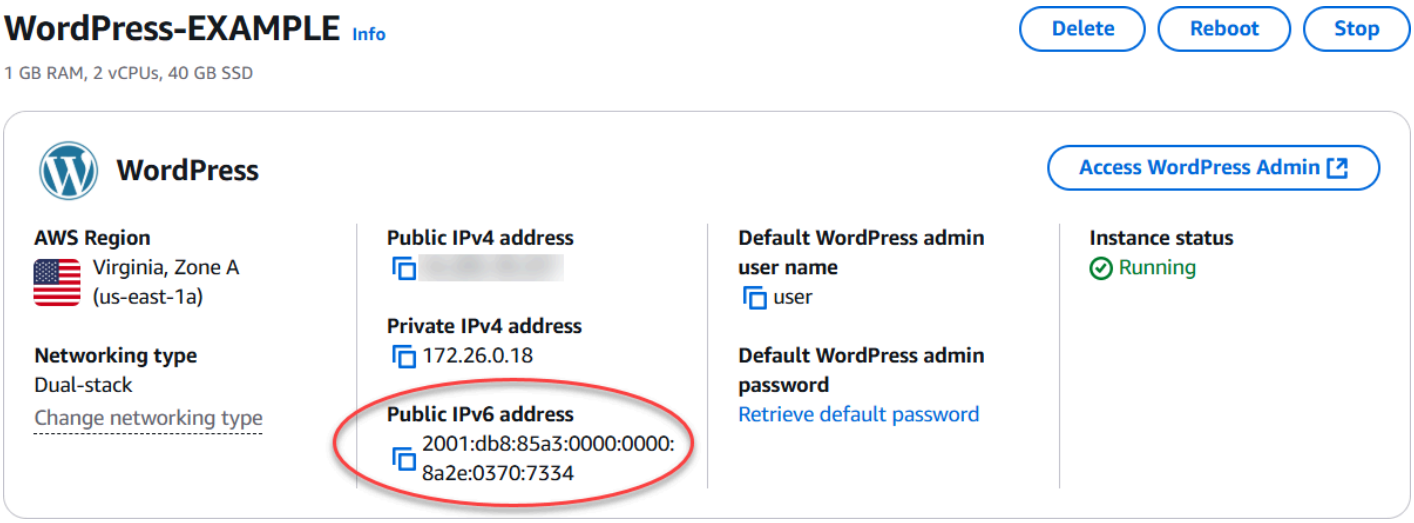
IPv6 전용 인스턴스를 생성할 수도 있습니다. IPv6 전용 인스턴스는 IPv6를 통해서만 공개적으로 통신할 수 있으며 퍼블릭 IPv4 주소가 없습니다. 자세한 내용은 Lightsail 인스턴스 IPv6 전용 네트워킹 구성 섹션을 참조하세요.

인스턴스의 IPv6 주소는 Lightsail 콘솔의 다음 영역에 표시됩니다.

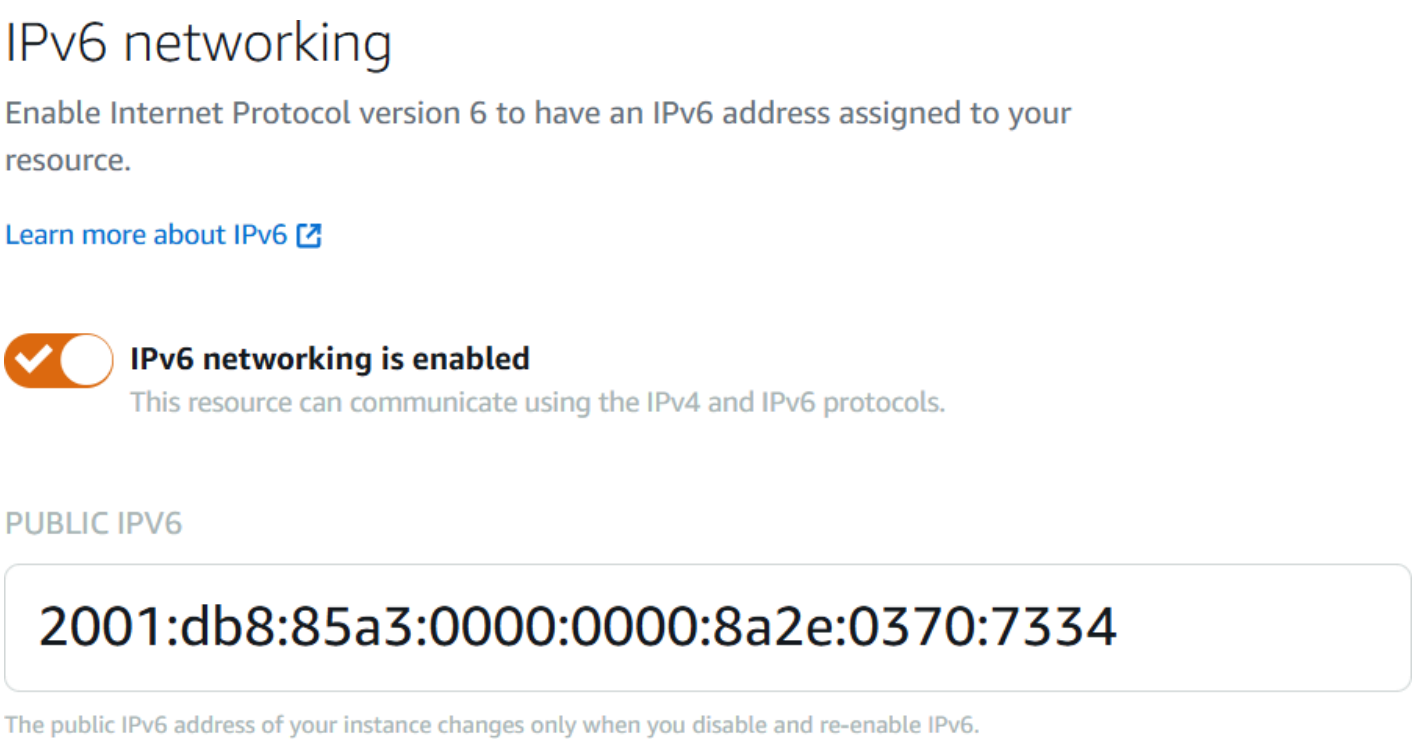
- 다음 예는 Lightsail 홈 페이지에 있는 인스턴스의 IPv6 주소를 보여줍니다.



- 다음 예는 리소스 관리 페이지의 머리말 영역에 있는 리소스의 IPv6 주소를 보여줍니다.



- 다음 예는 리소스 관리 페이지의 네트워킹(Networking) 탭에서 리소스의 IPv6 주소를 보여줍니다.



리소스에 IPv6를 활성화하여 사용할 때는 다음 사항에 유의하세요.

- 리소스에 IPv6를 사용할 때 리소스는 IPv4를 통해서만 통신하거나 IPv4 및 IPv6(이중 스택 모드)를 통해 통신할 수 있습니다.
- 리소스에 대해 IPv6를 활성화하면 Lightsail에서 해당 리소스에 IPv6 주소를 자동으로 할당하므로 IPv6 주소를 직접 선택하거나 지정할 수 없습니다. 리소스에 대해 IPv6를 활성화하면 IPv6 프로토콜을 통한 네트워크 트래픽을 수락하기 시작합니다.
- 인스턴스를 중지했다가 시작해도 인스턴스의 IPv6 주소는 유지됩니다. 인스턴스를 삭제하거나 인스턴스에 대해 IPv6를 사용 중지하도록 설정한 경우에만 해제됩니다. 이러한 작업을 수행하고 나면 IPv6 주소를 다시 가져올 수 없습니다.
- 인스턴스에 할당된 모든 IPv6 주소는 퍼블릭 주소이며 인터넷을 통해 접속할 수 있습니다. 인스턴스에는 프라이빗 IPv6 주소가 할당되지 않습니다.
- 인스턴스의 IPv4 및 IPv6 주소는 서로 독립적입니다. IPv4 및 IPv6의 인스턴스 방화벽 규칙은 별도로 구성해야 합니다. 자세한 내용은 인스턴스 방화벽을 참조하세요.
- IPv6를 사용하도록 설정한 경우 Lightsail에서 사용할 수 있는 인스턴스 블루프린트 일부가 IPv6에 대해 자동으로 구성되지 않습니다. 다음 블루프린트를 사용하는 인스턴스의 경우 IPv6를 사용하도록 설정한 후 구성 단계를 추가로 거쳐야 합니다.
  - cPanel – 자세한 내용은 cPanel에서 IPv6 구성을 참조하세요.
  - GitLab – 자세한 내용은 GitLab에서 IPv6 구성을 참조하세요.

- Nginx – 자세한 내용은 Nginx에서 IPv6 구성을 참조하세요.
- Plesk – 자세한 내용은 Plesk에서 IPv6 구성을 참조하세요.

> **Note**
>
> PrestaShop은 현재 IPv6 주소를 지원하지 않습니다. 인스턴스에 IPv6를 활성화할 수 있지만, PrestaShop 소프트웨어는 IPv6 네트워크를 통한 요청에 응답하지 않습니다.

Lightsail의 고정 IP 주소

고정 IP는 인스턴스 또는 다른 리소스에 할당하고 재할당할 수 있는 고정된 퍼블릭 IP 주소입니다. 고정 IP 주소를 설정하지 않은 경우 인스턴스를 중지하거나 다시 시작할 때마다 Lightsail이 새 퍼블릭 IP 주소를 할당합니다.

고정 IP 주소가 Lightsail 인스턴스에 연결되어 있을 때에는 관련 비용이 발생하지 않습니다. 그러나 고정 IP 주소가 인스턴스에 연결되지 않았을 때에는 요금이 부과됩니다. 자세한 내용은 Lightsail 고정 IPv4 주소의 사용 요금은 얼마인가요? 단원을 참조하십시오.

> **Important**
>
> 먼저 고정 IP 주소를 생성하여 인스턴스에 연결하지 않고 인스턴스를 중지하거나 다시 시작하면 인스턴스가 다시 시작될 때 IP 주소를 잃게 됩니다. 인스턴스가 항상 동일한 퍼블릭 IP 주소를 갖도록 하려면 고정 IP 주소를 생성하여 인스턴스에 연결해야 합니다. 자세한 내용은 고정 IP 주소 생성을 참조하세요.

내용

- 고정 IP를 만들어 Lightsail 인스턴스를 생성하고 연결합니다.
- Lightsail에서 고정 IP 주소 삭제

고정 IP를 만들어 Lightsail 인스턴스를 생성하고 연결합니다.

Amazon Lightsail 인스턴스에 연결된 기본 동적 퍼블릭 IP 주소는 인스턴스를 중지하고 다시 시작할 때마다 변경됩니다. 퍼블릭 IP 주소가 변경되지 않도록 고정 IP 주소를 생성하고 인스턴스에 연결하세요. 나중에 인스턴스에 등록된 도메인 이름을 지정하면 인스턴스를 중지하고 다시 시작할 때마다 도메인

의 DNS 레코드를 업데이트할 필요가 없습니다. 한 인스턴스에 한 개의 고정 IP를 연결할 수 있습니다. 자세한 내용은 고정 IP 주소를 참조하세요.

사전 조건

Lightsail에서 실행 중인 이중 스택 인스턴스가 하나 이상 필요합니다. 하나 생성하려면 인스턴스 생성을 참조하세요.

고정 IP 주소를 생성하여 인스턴스에 할당

Lightsail에서 새 고정 IP 주소를 생성하여 인스턴스에 연결하려면 다음 단계를 따르세요.

1. https://lightsail.aws.amazon.com/ Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 네트워킹을 선택합니다.
3. 고정 IP 생성을 선택합니다.
4. 고정 IP를 생성할 AWS 리전 를 선택합니다.

   > **Note**
   >
   > 고정 IP 주소는 동일한 리전의 인스턴스에만 연결할 수 있습니다.

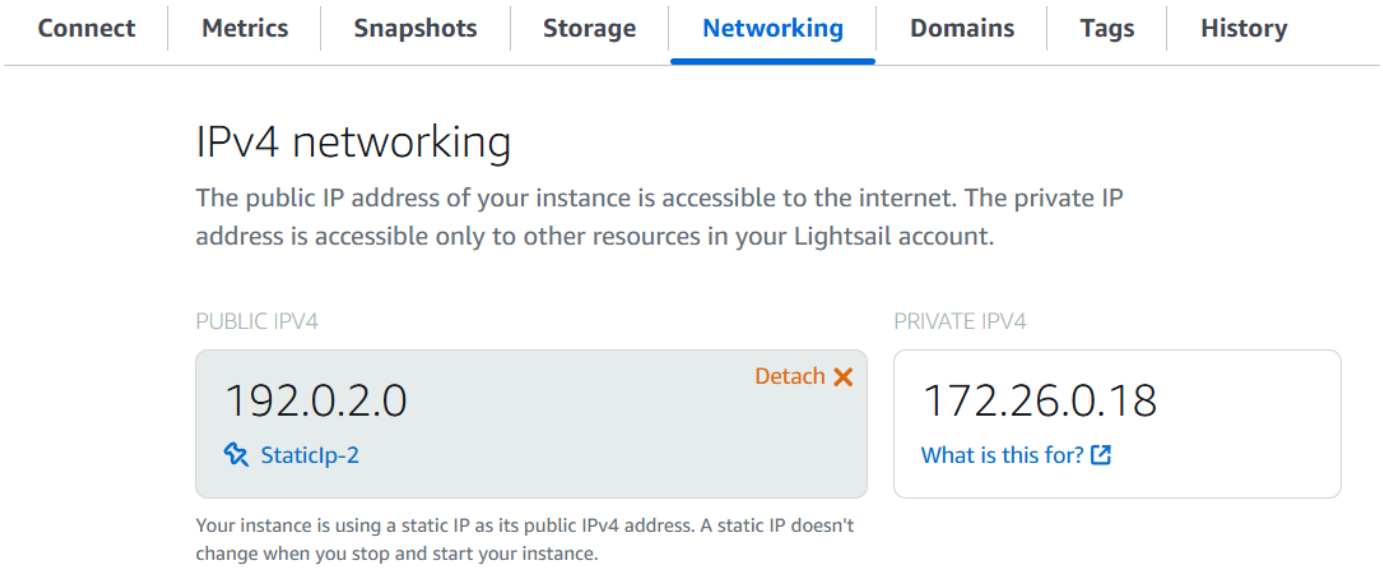
5. 고정 IP를 연결하려는 Lightsail 리소스를 선택합니다.
6. 고정 IP의 이름을 입력합니다.

   리소스 이름:

   - Lightsail 계정 AWS 리전 의 각 내에서 고유해야 합니다.
   - 2~255자의 문자로 구성되어야 합니다.
   - 영숫자 문자 또는 숫자로 시작하고 끝나야 합니다.
   - 영숫자 문자, 숫자, 마침표, 대시, 밑줄이 포함될 수 있습니다.
7. 생성을 선택합니다.

   홈 페이지로 이동하면 관리할 수 있는 고정 IP 주소가 보입니다.

또한, 인스턴스의 관리 페이지에 있는 네트워킹 탭에서 퍼블릭 IP 주소 옆에는 파란색 압정이 보입니다. 이는 IP 주소가 현재 고정 IP 주소임을 나타냅니다.



자세한 내용은 퍼블릭 IP 주소 및 프라이빗 IP 주소를 참조하세요.

Lightsail에서 고정 IP 주소 삭제

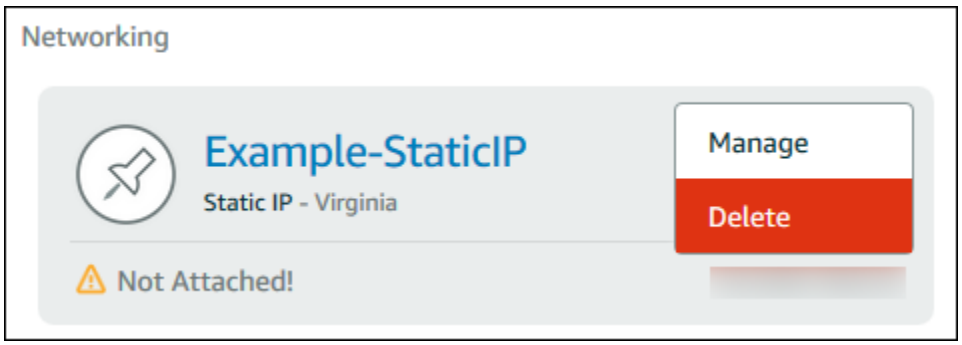
Amazon Lightsail 계정 AWS 리전 에서 당 최대 5개의 고정 IPs 생성할 수 있습니다. 고정 IP 주소가 연결된 인스턴스를 삭제해도 고정 IP 주소는 계정에 남아 있습니다. 고정 IP 주소가 더 이상 필요하지 않은 경우 Lightsail 콘솔 또는 AWS Command Line Interface ()를 사용하여 삭제할 수 있습니다AWS CLI. 이 안내서에서는 Lightsail 계정에서 고정 IP 주소를 삭제하는 방법을 보여 줍니다. 고정 IP에 대한 자세한 내용은 IP 주소를 참조하세요.

> **Important**
>
> 고정 IP를 삭제하면 Lightsail 계정에서 고정 IP가 완전히 제거됩니다. 해당 고정 IP를 사용하는 리소스(예: 인스턴스)가 영향을 받습니다. 고정 IP를 삭제한 후에는 고정 IP를 다시 가져올 수 없습니다.

Lightsail 콘솔을 사용하여 고정 IP 삭제

Lightsail 콘솔을 사용하여 고정 IP를 삭제하려면 다음 절차를 완료합니다.

1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 네트워킹을 선택합니다.
3. 네트워킹 페이지에서 삭제하려는 고정 IP 주소 옆의 세로 줄임표(⋮) 아이콘을 선택한 다음 삭제를 선택합니다.



를 사용하여 고정 IP 삭제 AWS CLI

AWS CLI를 사용하여 고정 IP를 삭제하려면 다음 절차를 완료합니다. Lightsail 계정에서 고정 IP를 삭제하는 명령은 release-static-ip입니다. 고정 IP를 생성할 경우 실제로는 고정 IP를 할당하는 것입니다. 따라서 고정 IP를 삭제하는 것이 아니라 실제로는 해제하는 것입니다.

사전 조건

먼저 아직 설치하지 않았다면를 설치해야 합니다 AWS CLI. 자세한 내용은 AWS Command Line Interface설치 섹션을 참조하세요. AWS CLI를 구성해야 합니다.

고정 IP를 해제하려면 고정 IP의 이름이 필요합니다. `get-static-ips` AWS CLI 명령을 사용하여 이를 얻을 수 있습니다.

1. 다음 명령을 입력합니다.

```
aws lightsail get-static-ips
```

다음과 유사한 출력 화면이 표시되어야 합니다.

```
{
    "staticIps": [
        {
            "name": "Example-StaticIP",
            "resourceType": "StaticIp",
            "attachedTo": "MyInstance",
            "arn": "arn:aws:lightsail:us-east-2:123456789101:StaticIp/5282f35e-
c720-4e5a-1234-12345EXAMPLE",
            "isAttached": true,
            "ipAddress": "192.0.2.0",
            "createdAt": 1489750629.026,
            "location": {
                "availabilityZone": "all",
                "regionName": "us-east-2"
            }
        },
        {
            "name": "my-other-static-ip",
            "resourceType": "StaticIp",
            "arn": "arn:aws:lightsail:us-east-2:123456789101:StaticIp/
f5885e14-8984-49e5-1234-12345EXAMPLE",
            "isAttached": false,
            "ipAddress": "192.0.2.2",
            "createdAt": 1483653597.815,
            "location": {
                "availabilityZone": "all",
                "regionName": "us-east-2"
            }
        }
    ]
}
```

2. 해제하려는 고정 IP의 이름 값을 선택하고 다음 단계에서 사용할 수 있도록 기록해 둡니다.

   예를 들어, 값을 클립보드에 복사할 수 있습니다.

3. 다음 명령을 입력합니다.

```
aws lightsail release-static-ip --static-ip-name StaticIpName
```

명령에서 *StaticIpName*을 고정 IP의 이름으로 바꿉니다.

성공한 경우 다음과 유사한 출력 화면이 표시되어야 합니다.

```
{
    "operations": [
        {
            "status": "Succeeded",
            "resourceType": "StaticIp",
            "isTerminal": true,
            "statusChangedAt": 1489860944.19,
            "location": {
                "availabilityZone": "all",
                "regionName": "us-east-2"
            },
            "operationType": "ReleaseStaticIp",
            "resourceName": "Example-StaticIP",
            "id": "92a2f0d2-eef2-4e6f-1234-12345EXAMPLE",
            "createdAt": 1489860944.19
        }
    ]
}
```

Lightsail 리소스에 대한 듀얼 스택 네트워킹 활성화 또는 비활성화

IPv6는 2021년 1월 12일 이후에 생성된 Lightsail 듀얼 스택 인스턴스, 컨테이너 서비스 및 로드 밸런서에 대해 기본적으로 사용하도록 설정됩니다. 필요에 따라 2021년 1월 12일 이전에 생성된 리소스에서 IPv6를 실행할 수 있습니다. 이 가이드에서는 듀얼 스택 인스턴스에 대해 IPv6 네트워킹을 활성화 또는 비활성화하는 방법을 보여줍니다. IPv6에 대한 자세한 내용은 IP 주소를 참조하세요.

듀얼 스택 고려 사항

IPv6는 2021년 1월 12일부터 Lightsail에서 지원되므로, 다음 지침에 따라 일부 리소스에 대해 IPv6를 수동으로 사용하거나 사용 중지하도록 설정해야 할 수 있습니다.

- 1월 12일 이전에 생성된 인스턴스 및 로드 밸런서의 경우 IPv6가 사용하도록 설정될 때까지 사용 중지됩니다. 그러나 1월 12일 이후에 생성된 인스턴스 및 로드 밸런서의 경우 생성 시 IPv6를 사용하도록 설정됩니다.
- 1월 12일 이전이나 이후에 생성된 컨테이너 서비스는 IPv6를 사용하도록 설정되어 있습니다.
- 언제든지 인스턴스 및 로드 밸런서에 대해 IPv6를 수동으로 사용하거나 사용 중지하도록 설정할 수 있습니다. 컨테이너 서비스의 경우 사용 중지할 수 없습니다.

IPv6를 사용하도록 설정하고 사용할 때는 다음 사항에 유의해야 합니다.

- 리소스에 IPv6를 사용할 때 리소스는 IPv4를 통해서만 통신하거나 IPv4 및 IPv6(듀얼 스택 모드)를 통해 통신할 수 있습니다.
- 인스턴스에 IPv6를 사용하도록 설정하면 Lightsail은 자동으로 해당 인스턴스에 IPv6 주소를 할당하므로, 사용자가 직접 IPv6 주소를 선택하거나 지정할 수 없습니다. 컨테이너 서비스 또는 로드 밸런서에 대해 IPv6를 사용하도록 설정하면 해당 리소스가 IPv6를 통한 인터넷 트래픽을 수락하기 시작합니다.
- 인스턴스를 중지했다가 시작해도 인스턴스의 IPv6 주소는 유지됩니다. 인스턴스를 삭제하거나 인스턴스에 대해 IPv6를 사용 중지하도록 설정한 경우에만 해제됩니다. 이러한 작업을 수행하고 나면 IPv6 주소를 다시 가져올 수 없습니다.
- 인스턴스에 할당된 모든 IPv6 주소는 퍼블릭 주소이며 인터넷을 통해 접속할 수 있습니다. 인스턴스에는 프라이빗 IPv6 주소가 할당되지 않습니다.
- 인스턴스의 IPv4 및 IPv6 주소는 서로 독립적입니다. IPv4 및 IPv6의 인스턴스 방화벽 규칙은 별도로 구성해야 합니다. 자세한 내용은 인스턴스 방화벽을 참조하세요.
- IPv6를 사용하도록 설정한 경우 Lightsail에서 사용할 수 있는 인스턴스 블루프린트 일부가 IPv6에 대해 자동으로 구성되지 않습니다. 다음 블루프린트를 사용하는 인스턴스의 경우 IPv6를 사용하도록 설정한 후 구성 단계를 추가로 거쳐야 합니다.
  - cPanel – 자세한 내용은 cPanel에서 IPv6 구성을 참조하세요.
  - GitLab – 자세한 내용은 GitLab에서 IPv6 구성을 참조하세요.
  - Nginx – 자세한 내용은 Nginx에서 IPv6 구성을 참조하세요.
  - Plesk – 자세한 내용은 Plesk에서 IPv6 구성을 참조하세요.

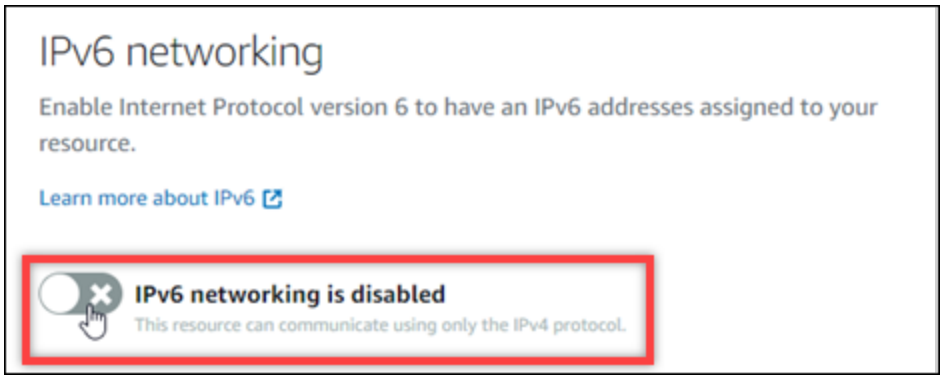
주제

- Lightsail 리소스에 대한 IPv6 네트워킹 활성화
- Lightsail 리소스에 대한 IPv6 네트워킹 비활성화

Lightsail 리소스에 대한 IPv6 네트워킹 활성화

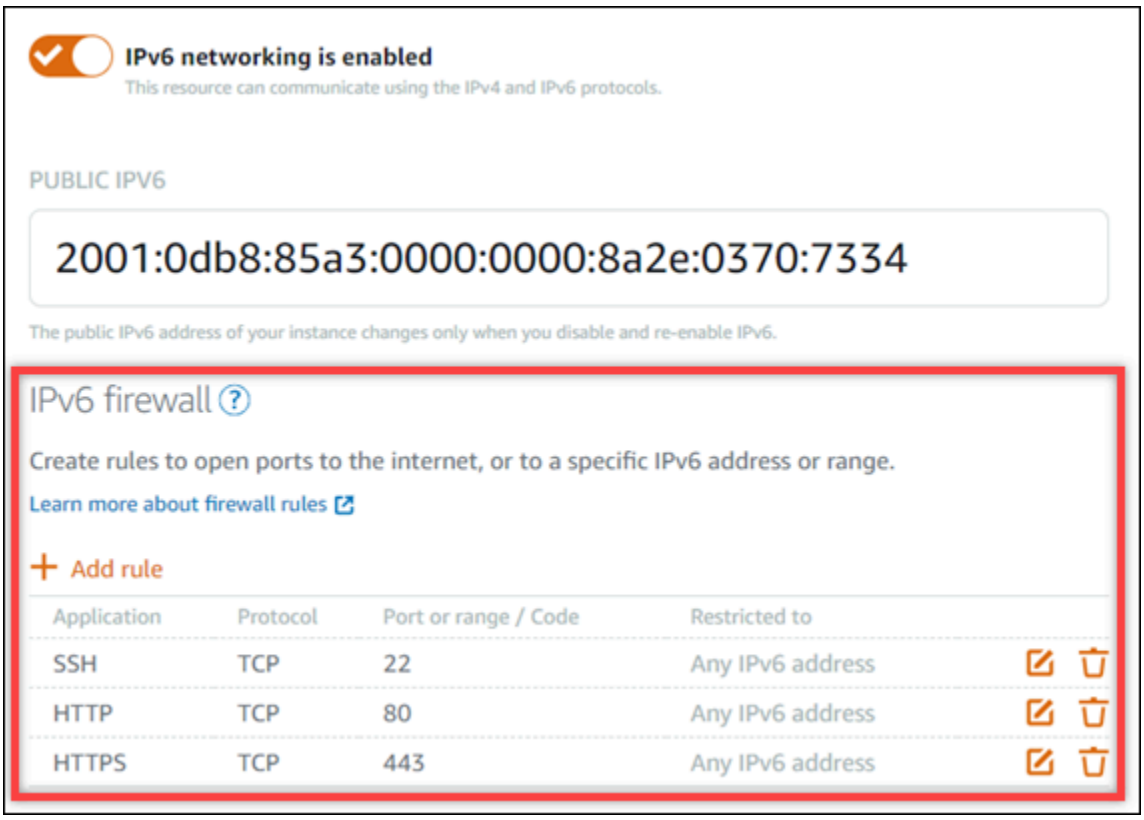
인스턴스, CDN 배포 및 로드 밸런서에 IPv6를 사용하도록 설정하려면 다음 절차를 완료하세요.

1. Lightsail 콘솔에 로그인합니다.
2. IPv6를 사용하도록 설정할 리소스에 따라 다음 단계 중 하나를 완료합니다.
   - 인스턴스에서 IPv6를 사용하려면 Lightsail 홈 페이지에서 인스턴스 탭을 선택한 다음, IPv6를 사용하도록 설정할 인스턴스의 이름을 선택합니다.
   - CDN 배포 또는 로드 밸런서에서 IPv6를 사용하도록 설정하려면 왼쪽 탐색 창에서 네트워킹 탭을 선택한 다음, IPv6를 사용하도록 설정할 CDN 배포 또는 로드 밸런서의 이름을 선택합니다.
3. 리소스 관리 페이지에서 네트워킹(Networking) 탭을 선택합니다.
4. 페이지의 IPv6 네트워킹(IPv6 Networking) 섹션에서 토글 버튼을 선택하여 리소스에서 IPv6를 사용하도록 설정합니다.



리소스에서 IPv6를 사용하도록 설정한 후에는 다음 항목에 유의해야 합니다.

- CDN 배포 또는 로드 밸런서에서 IPv6를 사용하도록 설정하면 해당 리소스가 IPv6 트래픽을 수락하기 시작합니다. 인스턴스에서 IPv6를 사용하도록 설정하면 IPv6 주소가 할당되고 다음 예와 같이 IPv6 방화벽을 사용할 수 있게 됩니다.



- 다음 블루프린트를 사용하는 인스턴스는 IPv6를 사용하도록 설정한 후 인스턴스가 새 IPv6 주소를 인식하도록 단계를 추가로 수행해야 합니다.
  - cPanel – 자세한 내용은 cPanel에서 IPv6 구성을 참조하세요.
  - GitLab – 자세한 내용은 GitLab에서 IPv6 구성을 참조하세요.
  - Nginx – 자세한 내용은 Nginx에서 IPv6 구성을 참조하세요.
  - Plesk – 자세한 내용은 Plesk에서 IPv6 구성을 참조하세요.
- 트래픽을 인스턴스, 컨테이너 서비스, CDN 배포 또는 로드 밸런서로 연결하는 등록된 도메인 이름이 있는 경우, 도메인의 DNS에 IPv6 주소 레코드(AAA)를 생성하여 IPv6 트래픽 경로를 리소스로 지정해야 합니다.

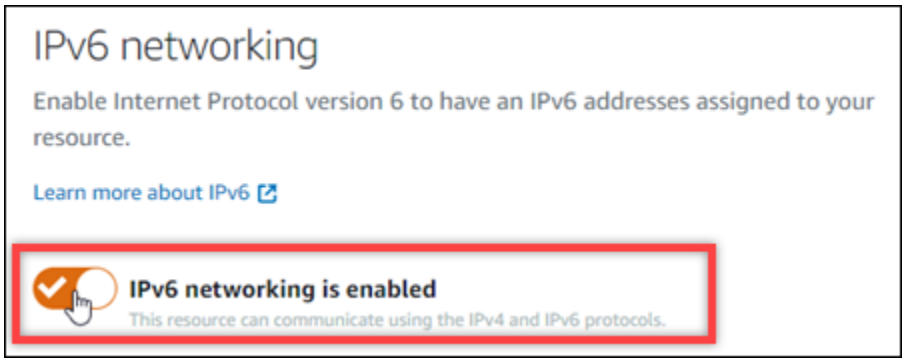
Lightsail 리소스에 대한 IPv6 네트워킹 비활성화

인스턴스, CDN 배포 및 로드 밸런서에서 IPv6를 사용 중지하려면 다음 절차를 완료하세요.

1. Lightsail 콘솔에 로그인합니다.
2. IPv6를 사용 중지할 리소스에 따라 다음 단계 중 하나를 완료합니다.

- 인스턴스에서 IPv6를 사용하지 않으려면 Lightsail 홈 페이지에서 인스턴스 탭을 선택한 다음, IPv6를 사용 중지할 인스턴스의 이름을 선택합니다.
- CDN 배포 또는 로드 밸런서에서 IPv6를 사용하지 않으려면 왼쪽 탐색 창에서 네트워킹 탭을 선택한 다음, IPv6를 사용 중지할 CDN 배포 또는 로드 밸런서의 이름을 선택합니다.

3. 리소스 관리 페이지에서 네트워킹(Networking) 탭을 선택합니다.
4. 페이지의 IPv6 네트워킹(IPv6 Networking) 섹션에서 토글 버튼을 선택하여 리소스에서 IPv6를 사용 중지합니다.



Lightsail 인스턴스 IPv6 전용 네트워킹 구성

Lightsail 인스턴스는 이중 스택 네트워킹(IPv4 및 IPv6)과 IPv6-only 네트워킹이라는 두 가지 유형의 네트워킹을 지원합니다. 이중 스택 네트워킹을 사용하면 인스턴스에 퍼블릭 IPv4와 퍼블릭 IPv6 주소가 할당됩니다. 듀얼 스택 네트워킹을 사용하는 인스턴스의 경우 필요에 따라 IPv6를 활성화하거나 비활성화할 수 있습니다.

IPv6 전용 네트워킹을 사용하면 인스턴스에 퍼블릭 IPv6 주소가 할당되고 퍼블릭 IPv4 트래픽을 지원하지 않습니다. 모든 Lightsail 블루프린트가 IPv6와 호환되는 것은 아닙니다. IPv6-only를 지원하는 블루프린트를 알아보려면 IPv6 호환 블루프린트 섹션을 참조하세요. 또는 IPv6 전용 네트워킹을 사용하는 인스턴스는 Lightsail 콘텐츠 전송 네트워크(CDN) 배포의 오리진 리소스로 구성할 수 없습니다. Lightsail 배포에 대한 자세한 내용은 Lightsail 콘텐츠 전송 배포를 통해 전 세계적으로 웹 콘텐츠 제공 섹션을 참조하세요.

퍼블릭 IPv4 주소가 필요하지 않은 경우 IPv6 전용 네트워킹을 사용합니다. 하지만 먼저 로컬 네트워크, 컴퓨터, 디바이스 및 최종 사용자가 IPv6를 사용하여 통신할 수 있는지 확인합니다. 자세한 내용은 Lightsail 인스턴스의 IPv6 도달 가능성 확인의 IPv6 도달 가능성을 참조하세요.

지원되는 블루프린트가 있는 기존 인스턴스의 경우, 네트워킹 유형을 듀얼 스택 네트워킹과 IPv6 전용 네트워킹 간에 변경할 수 있습니다. IPv6 전용 네트워킹의 고려 사항을 검토하고 기존 인스턴스를 변경하려면 인스턴스 네트워킹 유형을 Lightsail의 IPv6 또는 이중 스택으로 전환 섹션을 참조하세요.

주제

- 인스턴스 네트워킹 유형을 Lightsail의 IPv6 또는 이중 스택으로 전환
- IPv6 호환 블루프린트

인스턴스 네트워킹 유형을 Lightsail의 IPv6 또는 이중 스택으로 전환

인스턴스의 네트워킹 유형에 따라 인터넷 통신에 사용하는 프로토콜이 결정됩니다. 인스턴스를 생성할 때 이중 스택 또는 IPv6 전용 네트워킹 중에서 선택합니다. 기존 인스턴스의 네트워킹 유형을 이중 스택에서 IPv6 전용으로 변경할 수도 있습니다. 단계별 안내 워크플로를 사용하거나 개별 단계를 완료하여 네트워킹 유형을 변경합니다.

안내형 워크플로를 사용하면 새 네트워킹 유형이 구성된 동안 인스턴스가 계속 실행됩니다. 변경 사항이 적용되는 동안 인터넷을 통해 인스턴스에 연결할 수 있도록 하려면 이 옵션을 사용합니다. 하지만 먼저 로컬 네트워크, 컴퓨터, 디바이스 및 최종 사용자가 IPv6를 사용하여 통신할 수 있는지 확인해야 합니다. 자세한 내용은 Lightsail 인스턴스의 IPv6 도달 가능성 확인 섹션을 참조하세요.

개별 단계를 통해 인스턴스의 스냅샷을 생성한 다음 스냅샷에서 새 인스턴스를 생성합니다. 새 인스턴스를 생성할 때 다른 네트워킹 유형을 선택할 수 있습니다. 다른 인스턴스의 구성을 변경하기 전에 이 옵션을 사용하여 IPv6 호환성을 확인합니다. 시작하기 전에 먼저 IPv6 전용 고려 사항을 확인하는 것이 좋습니다.

IPv6 전용 고려 사항

다음과 같은 고려 사항을 검토합니다.

- 인스턴스 플랜은 네트워킹 유형이 변경될 때마다 변경됩니다. 자세한 내용은 AWS 컴퓨팅 블로그에서 Amazon Lightsail의 IPv6 인스턴스 번들 및 요금 업데이트 발표를 참조하세요.
- 인스턴스는 IPv6를 통해 공개적으로 통신합니다. 수신 또는 발신 퍼블릭 IPv4 트래픽은 지원하지 않습니다. Lightsail 계정의 다른 리소스와 통신할 수 있는 프라이빗 IPv4 주소가 수신됩니다. 자세한 내용은 Lightsail 리소스의 IP 주소 보기 및 관리 단원을 참조하십시오.
- IPv6 전용 인스턴스는 Lightsail 콘텐츠 전송 네트워크(CDN) 배포의 오리진으로 구성할 수 없습니다.
- IPv6 전용 인스턴스를 Lightsail 로드 밸런서에 추가할 수 있습니다.
- 네트워킹 유형을 변경하면 인스턴스의 데이터 전송 플랜에 대한 허용량이 이월됩니다. 초기화되지는 않습니다.
- 로컬 디바이스, 네트워크 및 인터넷 서비스 제공업체(ISP)가 IPv6와 호환되는지 확인합니다. 자세한 내용은 Lightsail 인스턴스의 IPv6 도달 가능성 확인 섹션을 참조하세요.

옵션: 안내형 워크플로

마법사를 사용하여 인스턴스 네트워킹 유형 구성

1. 인스턴스 관리 페이지의 정보 패널에서 네트워킹 유형 변경을 선택합니다.
2. 네트워킹 유형 선택에서 이중 스택 또는 IPv6 전용을 선택합니다. 선택한 옵션 아래에 강조 표시된 정보를 검토한 후 다음을 선택합니다.
3. 리소스 검토에서 현재 인스턴스와 연결된 리소스에 대해 변경되는 사항을 검토합니다. 리소스는 고정 IP 주소 또는 Lightsail 로드 밸런서일 수 있습니다. 인스턴스에 연결된 리소스가 없으면 변경되지 않습니다. 다음 단계에서 워크플로를 완료할 때까지 리소스 변경은 수행되지 않습니다. 다음을 선택하여 계속 진행합니다.
4. 변경 사항 확인에서 새 인스턴스 네트워킹 유형, 요금 및 리소스 변경 사항을 검토하고 변경 사항 확인을 선택합니다. Lightsail 리소스를 구성하기 시작합니다.
5. (선택 사항) 워크플로가 완료된 후 인스턴스 구성을 업데이트합니다. 예를 들어 인스턴스에 고정 IP를 연결하거나 IPv4의 경우 DNS A 레코드를 업데이트하고 IPv6의 경우 AAAA 레코드를 업데이트합니다. 다음 단계는 이 가이드의 the section called “다음 단계” 섹션을 참조하세요.

옵션: 개별 단계

개별 단계를 완료하여 인스턴스 네트워킹 유형 구성

1. 인스턴스 관리 페이지의 스냅샷 탭에서 스냅샷 생성을 선택합니다. 자세한 내용은 다음 중 하나의 주제를 참조하세요:
   - 스냅샷으로 Linux/Unix Lightsail 인스턴스 백업
   - Lightsail Windows Server 인스턴스의 스냅샷 생성
2. 스냅샷에 이름을 지정한 다음 생성을 선택합니다.
3. 스냅샷 작업 메뉴(⋮)에서 새 인스턴스 생성을 선택합니다. 자세한 내용은 스냅샷에서 Lightsail 인스턴스 생성 섹션을 참조하세요.
4. 네트워킹 유형 선택 섹션에서 이중 스택 또는 IPv6 전용을 선택합니다.
5. 남은 옵션을 검토하고 인스턴스 생성을 선택합니다. 새 인스턴스가 생성됩니다.
6. (선택 사항) 워크플로가 완료된 후 인스턴스 구성을 업데이트합니다. 예를 들어 인스턴스에 고정 IP를 연결하거나 IPv4의 경우 DNS A 레코드를 업데이트하고 IPv6의 경우 AAAA 레코드를 업데이트합니다. 다음 단계는 이 가이드의 the section called “다음 단계” 섹션을 참조하세요.

다음 단계

인스턴스의 네트워킹 유형을 변경한 후 수행할 수 있는 몇 가지 추가 작업이 있습니다.

- (IPv6 전용) 애플리케이션과 사용자가 IPv6를 통해 통신할 수 있는지 확인합니다. 자세한 내용은 Lightsail 인스턴스의 IPv6 도달 가능성 확인 섹션을 참조하세요.
- (이중 스택) 고정 IP 주소를 인스턴스에 연결합니다. 자세한 내용은 고정 IP를 인스턴스에 연결을 참조하세요.
- (이중 스택) 인스턴스를 Lightsail 배포의 오리진으로 구성합니다. 자세한 내용은 Lightsail의 CDN 배포를 참조하세요.
- (둘 다) 인스턴스의 방화벽 설정을 추가하거나 업데이트합니다. 자세한 내용은 Lightsail의 인스턴스 방화벽을 참조하세요.
- (둘 다) IPv4의 DNS A 레코드와 IPv6의 AAAA 레코드를 추가하거나 업데이트합니다. 자세한 내용은 도메인을 인스턴스로 지정을 참조하세요.
- (둘 다) 인스턴스를 Lightsail 로드 밸런서에 추가합니다. 자세한 내용은 Lightsail의 로드 밸런서를 참조하세요.

IPv6 호환 블루프린트

다음 Lightsail 블루프린트는 IPv6 전용 인스턴스 플랜과 호환됩니다.

- Windows Server 2022
- Windows Server 2019
- Windows Server 2016
- Amazon Linux 2023
- Amazon Linux 2
- AlmaLinux OS 9
- CentOS Stream 9
- Debian 11, 12, and 13
- FreeBSD 13, 14, and 15
- Ubuntu 20, 22, and 24
- SQL Server 2022 Express
- SQL Server 2019 Express
- SQL Server 2016 Express

- LAMP stack (PHP 8) packaged by Bitnami
- MEAN stack packaged by Bitnami
- Redmine packaged by Bitnami

Lightsail 블루프린트에 대한 자세한 내용은 the section called “블루프린트” 섹션을 참조하세요.

Lightsail의 리전 및 가용 영역

에서 리소스를 생성할 때 사용자에게 AWS 리전 더 가까운에서 리소스를 Amazon Lightsail생성합니다. 또한 애플리케이션의 내결함성 및 고가용성을 위해 여러 가용 영역을 활용할 수 있습니다. 사용자들에게 더 가깝게 리소스를 프로비저닝하면 사용자에게 짧은 지연 시간과 향상된 성능을 제공할 수 있습니다. 예를 들어, 블로그 트래픽이 대부분 스위스에서 비롯된다면 프랑크푸르트 또는 파리를 선택합니다.

Lightsail에 지원되는 옵트인 리전(기본적으로 비활성화된 리전)에서 애플리케이션을 생성하는 것이 도움이 되는 경우 리전을 활성화할 수 있습니다. 나중에 옵트인 리전에서 리소스를 더 이상 운영하지 않기로 결정하는 경우 리전을 비활성화할 수 있습니다. 옵트인 리전에 대한 자세한 내용은 AWS 용어집의 리전을 활성화하기 전 고려 사항 및 옵트인 리전 항목을 참조하세요.

> **Note**
>
> DNS 영역 및 Lightsail 배포와 같은 글로벌 리소스는 미국 동부(버지니아 북부)(us-east-1) 리전에서만 생성되지만 모든 AWS 리전에서 지원되는 모든 리소스를 참조할 수 있습니다.

주제

- Lightsail 리전
- SSH 키 및 Lightsail 리전
- Lightsail 리전으로 작업하기 위한 팁
- Lightsail 가용 영역
- 가용 영역과 Lightsail 애플리케이션
- Lightsail에 대한 옵트인 리전 활성화
- Lightsail에 대한 옵트인 리전 비활성화

Lightsail 리전

Lightsail은 다음 리전에서 사용 가능합니다.

- 미국 동부(버지니아 북부)(us-east-1)
- 미국 동부(오하이오)(us-east-2)
- 미국 서부(오리건)(us-west-2)
- 아시아 태평양(자카르타)(ap-southeast-3)*
- 아시아 태평양(뭄바이)(ap-south-1)
- 아시아 태평양(서울)(ap-northeast-2)
- 아시아 태평양(싱가포르)(ap-southeast-1)
- 아시아 태평양(시드니)(ap-southeast-2)
- 아시아 태평양(도쿄)(ap-northeast-1)
- 캐나다(중부)(ca-central-1)
- EU(프랑크푸르트)(eu-central-1)
- EU(아일랜드)(eu-west-1)
- EU(런던)(eu-west-2)
- EU(파리) (eu-west-3)
- EU(스톡홀름)(eu-north-1)

* 이 기능은 기본적으로 비활성화되어 있습니다. * 이 리전을 사용하려면 먼저 리전을 활성화해야 합니다.

SSH 키 및 Lightsail 리전

Lightsail에서는 AWS 리전에 인스턴스를 생성하는 즉시 그 리전에 기본 SSH 키를 생성해야 합니다. 이 기본 키는 해당 리전의 인스턴스에 연결하는 데만 사용할 수 있습니다. 자세한 내용은 SSH 키 페어를 참조하세요.

Lightsail 리전으로 작업하기 위한 팁

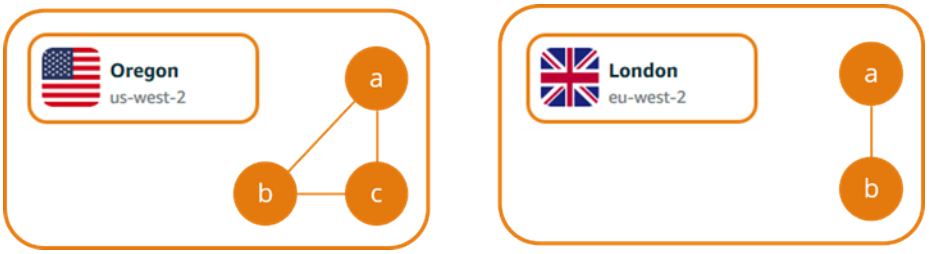
각 AWS 리전 는 서로 완전히 격리되도록 설계되었습니다 AWS 리전. 이를 통해 가장 강력한 내결함성 및 안정성을 달성할 수 있습니다.

리전 간 통신은 모두 퍼블릭 인터넷을 통해 이루어집니다. 따라서 데이터 보호를 위해 적절한 암호화 방법을 사용해야 합니다. 리전 간 데이터 전송에는 요금이 발생합니다. 자세한 내용은 Amazon EC2 요금 - 데이터 전송을 참조하세요.

AWS Command Line Interface (AWS CLI)를 사용하여 Lightsail 인스턴스로 작업하는 경우 명령의 `--region` 옵션을 AWS CLI 사용하여 리전의 코드를 지정하는 것이 좋습니다. 예를 들어 미국 동부(버지니아 북부)(us-east-1) 리전의 DNS 영역 및 네트워크 리소스에 대한 정보를 반환하도록 **`us-east-1`**을 지정할 수 있습니다. 옵션 사용에 AWS CLI `--region` 대한 자세한 내용은 AWS CLI 참조의 일반 옵션을 참조하세요.

Lightsail 가용 영역

각 AWS 리전 에는 리전 이름() 뒤에 문자로 표시되는 격리된 가용 영역이 여러 개 있습니다`us-east-2`*a*. 가용 영역은 물리적으로 구분된 자체 독립 인프라에서 운영되는 데이터 센터를 모은 것입니다. 가용 영역은 높은 안정성을 갖추도록 설계되었습니다. 발전기 및 냉각 장비 등에 발생하는 일반적인 장애 사항은 가용 영역 간에 공유되지 않습니다. 가용 영역 역시 물리적으로는 분리되어 있으므로, 화재, 토네이도, 홍수 등 극한의 재해 상황이 발생하더라도 그에 해당하는 단일 가용 영역에만 영향을 미치게 됩니다.



한 번에 가용 영역 하나에서만 Lightsail 인스턴스를 생성할 수 있습니다. 인스턴스를 만드는 시점에는 일부 가용 영역이 보이지 않을 수도 있습니다. 가용 영역 목록이 전혀 보이지 않으면 이전 단계에서 리전을 선택했는지 확인하세요.

가용 영역과 Lightsail 애플리케이션

각각의 개별적인 가용 영역에서 인스턴스를 시작함으로써 단일 위치에서 장애가 발생할 경우 애플리케이션을 보호할 수 있습니다. 이러한 구성으로 진행하기 전에 일반 및 애플리케이션별 지침을 검토하세요. 자세한 내용은 로드 밸런싱을 위한 Lightsail 인스턴스 구성을 참조하세요.

기존 인스턴스를 확장하여 애플리케이션을 여러 가용 영역에서 사용할 수 있도록 하려면 먼저 인스턴스의 스냅샷을 생성합니다. 다음으로, 자신이 생성한 스냅샷에서 새 인스턴스를 생성할 때 다른 가용 영역을 선택합니다. 스냅샷에서 생성된 새 인스턴스를 사용하면 Lightsail 로드 밸런서를 사용하여 인스턴스에 웹 트래픽을 분산할 수 있습니다.

Lightsail에 대한 옵트인 리전 활성화

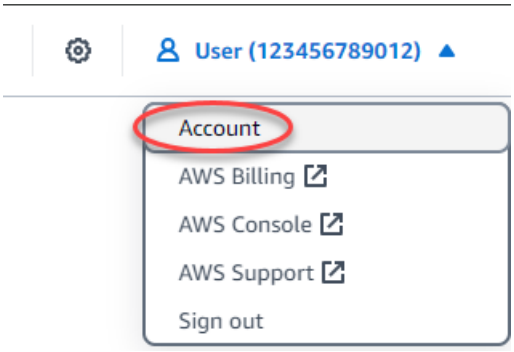
추가 비용 없이에 Lightsail대해 지원되는 옵트인 리전을 활성화할 수 있습니다. 새로 활성화된 리전에 생성한 리소스에만 요금이 부과됩니다. 이 프로세스는 대부분의 계정에서 몇 분이 걸립니다. 리전 작업에 대한 자세한 내용은 계정에서 AWS 리전 활성화 또는 비활성화를 참조하세요.

> **Note**
>
> 옵트인 리전을 Lightsail와 함께 사용하려면 Lightsail 콘솔에서 옵트인 상태가 활성화로 되어 있어야 합니다.

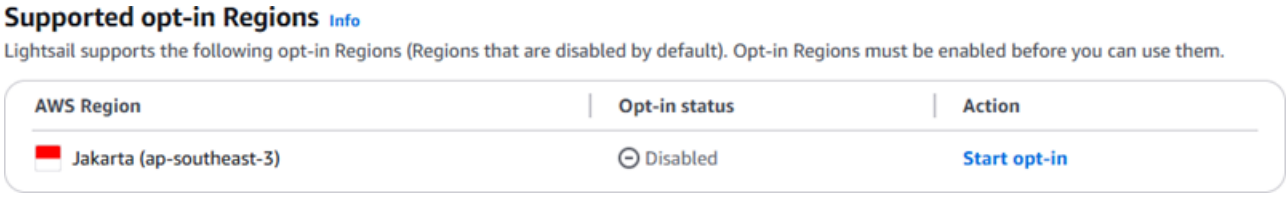
이 절차는 Lightsail 콘솔에서 옵트인 리전의 활성화를 시작하는 방법을 자세히 설명합니다.

Lightsail에 대한 옵트인 리전을 활성화하려면

1. Lightsail 콘솔에 로그인합니다.
2. Lightsail 홈 페이지의 상단 탐색 메뉴에서 계정을 선택합니다.
3. 드롭다운 메뉴에서 계정(Account)을 선택합니다.

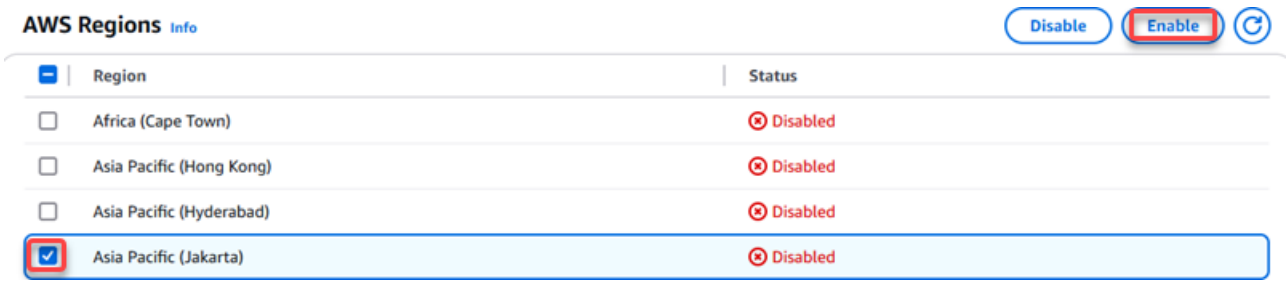


4. 계정 페이지에서 프로필 탭을 선택합니다.
5. 지원되는 옵트인 리전 섹션에서 활성화하려는 리전에 대해 옵트인 시작을 선택합니다.

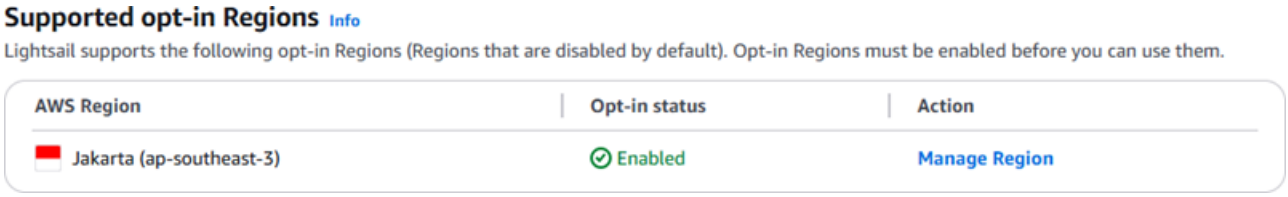


6. 옵트인 정보를 검토하고 옵트인 시작을 선택합니다.
7. 필요한 단계를 검토하고 AWS 프로필 관리를 선택하여 계속 진행합니다. AWS Account Management 콘솔 페이지가 열립니다. 이후 단계를 위해 Lightsail 콘솔을 열어 둡니다.
8. AWS 리전 섹션에서 활성화할 리전을 선택한 다음 활성화를 선택합니다.

> **Note**
>
> 의 리전 이름은 다른 AWS 서비스의 리전 이름과 Lightsail 약간 다릅니다. 예를 들어 Lightsail 콘솔의 자카르타(ap-southeast-3)는 AWS Account Management 콘솔의 아시아 태평양(자카르타)과 동일합니다.



9. 표시되는 추가 정보를 검토한 다음 리전 활성화를 선택하여 작업을 진행합니다.
10. Lightsail 콘솔의 계정 페이지로 돌아가 리전의 옵트인 상태 값을 주기적으로 확인하세요. 옵트인 상태는 프로세스가 완료되고 활성화됨으로 업데이트될 때까지 활성화 중으로 표시됩니다. 이제 새 리전에서 리소스를 프로비저닝할 수 있습니다.



Lightsail에 대한 옵트인 리전 비활성화

옵트인 리전에서 리소스를 더 이상 운영하지 않기로 결정하는 경우 리전을 비활성화할 수 있습니다. 리전을 비활성화하기 전에 해당 리전에 삭제할 리소스가 있는지 평가해야 합니다. 이 작업은 Lightsail 콘솔에서 리소스를 검토하거나 Billing and Cost Management 콘솔을 확인하여 수행할 수 있습니다. Billing and Cost Management 콘솔에서 비용을 검토하면 리전의 모든 AWS 서비스 에서 사용되는 리소스를 식별하는 데 도움이 될 수 있습니다. 리전 작업에 대한 자세한 내용은 계정 AWS 리전 에서 활성화 또는 비활성화를 참조하세요.

비활성화된 옵트인 리전에 있는 리소스의 동작

리소스를 아직 포함하고 있는 리전을 비활성화하더라도 해당 리소스(있는 경우)에 대한 요금은 표준 요율로 계속 부과됩니다. 또한 리소스가 계속 실행되더라도 리전에서 리소스를 관리할 수 있는 액세스 권한을 잃게 됩니다. 따라서 Lightsail 콘솔, Lightsail API AWS CLI또는 SDKs를 사용하여 이러한 리소스로 작업할 수 없습니다. 비활성화된 리전에서 리소스를 삭제하려면 해당 리전을 다시 활성화하게 해당 리소스에 대한 조치를 취해야 합니다.

예를 들어 실행 중인 WordPress 인스턴스가 있고 해당 인스턴스를 먼저 삭제하지 않고 인스턴스가 있는 옵트인 리전을 비활성화하면 요금이 계속 발생합니다. 또한 인스턴스는 계속 실행되며 인터넷에서 사용할 수 있습니다. 이 예제에서는 비활성화된 리전에서 WordPress 인스턴스를 더 이상 관리할 수 없습니다.

리소스가 리전에 있는지 확인

다음 단계를 사용하여 비활성화하려는 리전의 Lightsail 리소스에 대한 요금이 있는지 확인할 수 있습니다. 리전의 리소스에 대한 요금이 있는 경우 계속하기 전에 리소스를 삭제할 수 있습니다.

리전별 Lightsail 비용을 검토하려면

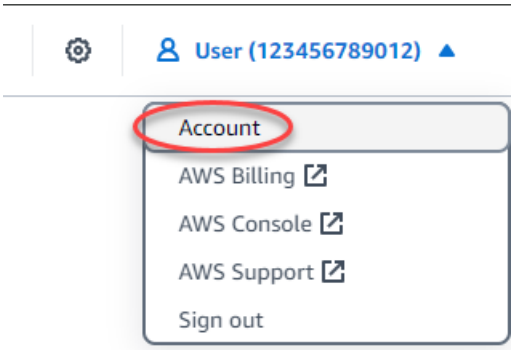
1. Billing and Cost Management 콘솔에 로그인합니다.
2. 왼쪽 탐색 창의 청구 및 결제에서 청구서를 선택합니다.
3. 서비스별 요금 탭의 Amazon Web Services, Inc. 서비스별 요금에서 Lightsail에 대한 항목을 확장합니다.
4. 비활성화하려는 리전에서 Lightsail에 대한 요금이 있는 경우 리전 이름 옆의 확장 아이콘을 선택합니다.
5. 리전 내에서 요금이 발생하는 리소스 유형의 목록을 검토합니다. 예를 들어 Amazon Lightsail 번들에 대한 요금은 리전에 Lightsail 인스턴스가 생성되었음을 가리킵니다.
6. 추가 요금이 발생하는 것을 중지하려면 비활성화하기 전에 리전 내의 모든 리소스를 삭제합니다.

옵트인 리전 비활성화

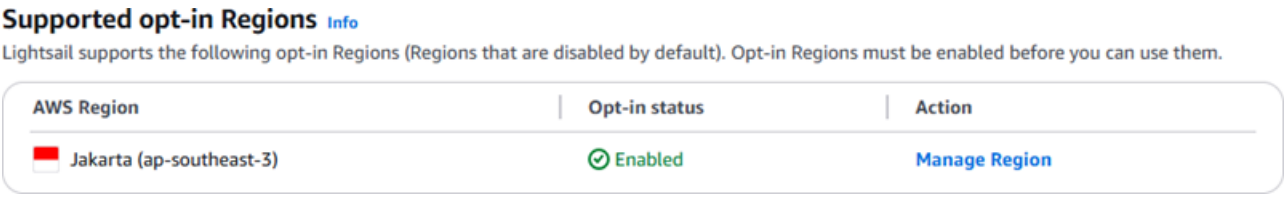
이 절차를 사용하여 옵트인 리전을 비활성화할 수 있습니다. 계속하기 전에 먼저 리소스가 리전에 있는지 확인 섹션을 검토했는지 확인합니다.

Lightsail에 대한 옵트인 리전을 비활성화하려면

1. Lightsail 콘솔에 로그인합니다.
2. Lightsail 홈 페이지의 상단 탐색 메뉴에서 계정을 선택합니다.
3. 드롭다운 메뉴에서 계정(Account)을 선택합니다.

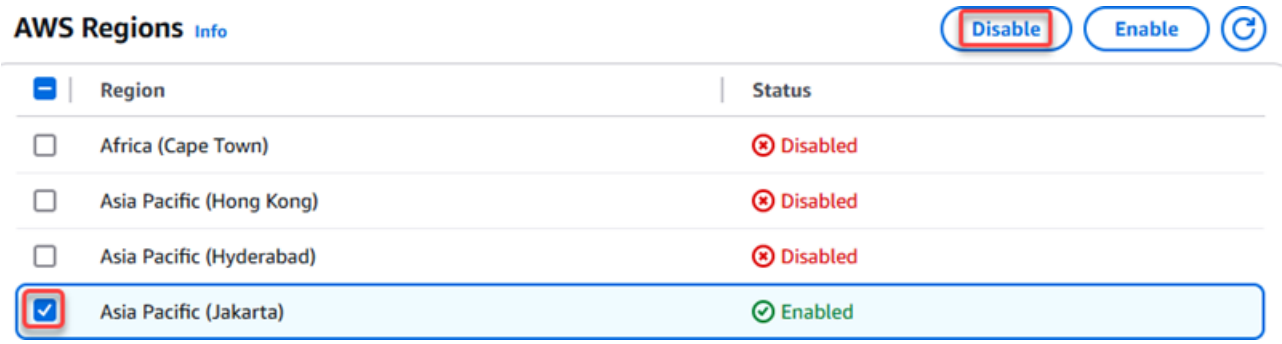


4. 프로필 탭의 옵트인 리전 지원에서 리전 관리를 선택합니다.

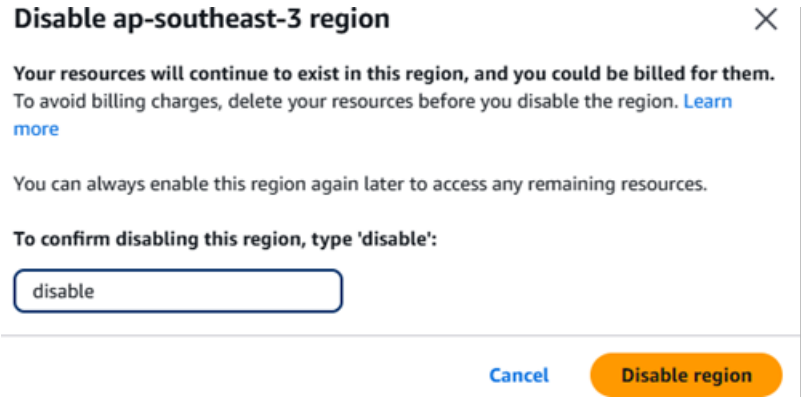


5. 리전 내에서 먼저 리소스를 삭제하는 방법에 대한 메시지를 검토합니다. 계속할 준비가 되면 AWS 프로필 관리를 선택합니다.
6. 필요한 단계를 검토하고 AWS 계정관리를 선택하여 계속 진행합니다. AWS Account Management 페이지가 열립니다.
7. AWS 리전 섹션에서 비활성화할 리전을 선택한 다음 비활성화를 선택합니다.

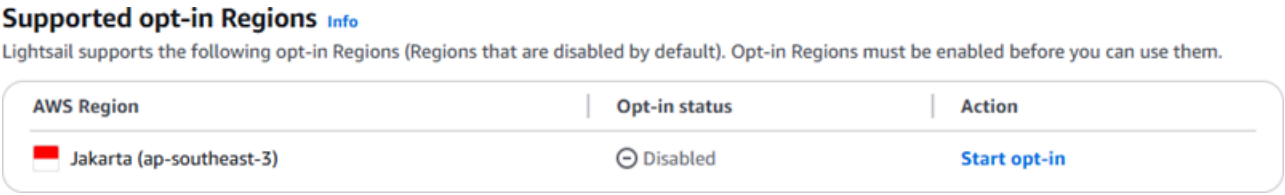
> **Note**
>
> 의 리전 이름은 다른 AWS 서비스의 리전 이름과 Lightsail 약간 다릅니다. 예를 들어 Lightsail 콘솔의 자카르타(ap-southeast-3)는 AWS Account Management 콘솔의 아시아 태평양(자카르타)과 동일합니다.



8. 표시되는 추가 정보를 검토한 다음 **`disable`**을 입력하고 리전 비활성화를 선택하여 작업을 진행합니다.



9. Lightsail 콘솔의 계정 페이지로 돌아가 리전의 옵트인 상태 값을 주기적으로 확인하세요. 옵트인 상태는 프로세스가 완료되고 비활성화됨으로 업데이트될 때까지 비활성화 중으로 표시됩니다.



VPC 피어링을 사용하여 Lightsail 리소스와 AWS 서비스 연결

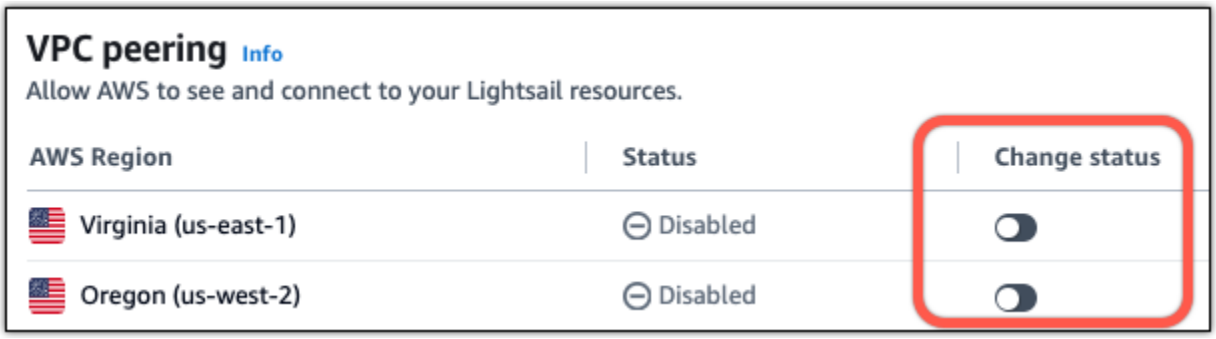
를 사용하면 Virtual Private Cloud(VPC) 피어링을 통해 Amazon RDS 데이터베이스와 같은 AWS 리소스에 연결할 Amazon Lightsail수 있습니다. VPC는 AWS 계정 전용 가상 네트워크입니다. Lightsail 내에서 만드는 모든 것은 VPC 내에 있으므로 Lightsail VPC를 Amazon VPC에 연결할 수 있습니다.

Amazon S3, Amazon CloudFront 및 Amazon DynamoDB와 같은 일부 AWS 리소스에서는 VPC 피어링을 활성화할 필요가 없습니다.

> **Note**
>
> 에서 VPC 피어링을 활성화하려면에 기본 VPC가 있어야 Lightsail합니다 AWS 리전. 피어링 관계는 Lightsail의 리소스와 VPC 피어링을 활성화하려는 리전의 기본 VPC의 리소스 간에 이루어집니다. 기본 Amazon VPC 인증서가 없는 경우 하나를 생성할 수 있습니다. 자세한 내용은 Amazon VPC 사용 설명서의 기본 VPC와 기본 VPC 생성을 참조하세요.
> AWS 리전는 서로 격리되어 있으므로 VPC는 VPC를 생성한 리전에서도 격리됩니다. 다른 Lightsail 리소스를 연결하려는 리소스 AWS 리전 가 있는 각에서 VPC 피어링을 활성화해야 합니다.

기본 Amazon VPC가 있으면 다음 지침에 따라 Lightsail VPC를 Amazon VPC와 피어링하세요.

1. Lightsail 콘솔의 상단 탐색 메뉴에서 사용자 이름을 선택합니다.
2. 드롭다운에서 계정을 선택합니다.
3. [Advanced] 탭을 선택합니다.
4. VPC 피어링을 활성화하려는 AWS 리전 옆의 상태를 전환합니다.



피어링 연결에 실패할 경우 VPC 피어링을 다시 활성화해보세요. 그래도 문제가 해결되지 않으면 AWS Support에 문의하세요.

피어링 요청이 성공하면 AWS 계정에 피어링 연결이 생성됩니다. Amazon VPC 대시보드로 이동한 다음 탐색 창에서 피어링 연결을 선택하여 생성된 피어링 연결을 확인합니다.

Amazon VPC에 대한 자세한 내용은 Amazon VPC 사용 설명서의 VPC 및 서브넷을 참조하세요.

다른 AWS 서비스와의 통신 허용

VPC 피어링이 활성화되면 연결하려는 다른 AWS 서비스의 리소스가 Lightsail 리소스에서 인바운드 트래픽을 수락하는지 확인해야 합니다. 다른 AWS 서비스의 리소스를 Lightsail 인스턴스에 연결하려면

필요한 인바운드 트래픽을 허용하는 방화벽 규칙을 추가할 수 있습니다. 자세한 내용은 Lightsail 인스턴스 방화벽 규칙 추가를 참조하세요.

수행할 수 있는 단계는 작업 중인 서비스 및 트래픽 유형에 따라 달라집니다. Lightsail 인스턴스를 Amazon RDS 데이터베이스에 연결하기 위해 수행할 수 있는 단계의 예는 Amazon Lightsail 데이터베이스 팁 및 요령 블로그 게시물을 참조하세요 AWS . VPC 피어링을 사용하여 Lightsail과 통합할 수 있는 서비스에 대한 자세한 내용은 Lightsail 다른 AWS 서비스와 VPC 피어링 통합 섹션을 참조하세요.

Lightsail의 SSL/TLS 인증서

Amazon Lightsail은 SSL/TLS 인증서를 사용하여 Lightsail 로드 밸런서, 콘텐츠 전송 네트워크(CDN) 배포, 컨테이너 서비스와 함께 사용할 수 있는 사용자 지정(등록된) 도메인을 검증합니다. 검증된 인증서를 해당 Lightsail 리소스에 연결한 후 도메인을 통해 해당 리소스로 라우팅되는 트래픽은 HTTPS(Hypertext Transfer Protocol Secure)를 사용하여 암호화됩니다.

Amazon Lightsail에서 전송 계층 보안(TLS) 인증서를 생성하여 Lightsail 로드 밸런서, 콘텐츠 전송 네트워크 배포, 컨테이너 서비스와 함께 사용하려는 사용자 지정(등록된) 도메인의 암호화된 웹 트래픽을 활성화할 수 있습니다. TLS는 보안 소켓 계층(SSL)을 한층 안전하게 업데이트한 버전입니다. Lightsail 문서와 콘솔 전체에서는 이를 SSL/TLS라고 부릅니다.

> **Important**
>
> 로드 밸런서, CDN 배포 및 컨테이너 서비스에 연결할 수 있는 Lightsail 인증서는 AWS Certificate Manager (ACM) 서비스에서 발급합니다. 2022년 10월 11일부터 로드 밸런서, CDN 배포, 컨테이너 서비스에 대해 Lightsail을 통해 취득한 모든 퍼블릭 인증서는 ACM이 관리하는 여러 중간 인증 기관(ICA) 또는 하위 CA 중 하나에서 발급됩니다. 자세한 내용은 AWS Security Blog(AWS 보안 블로그)의 Amazon introduces dynamic intermediate certificate authorities(Amazon, 동적 중간 인증 기관 도입)을 참조하세요.

HTTPS를 사용하는 이유

가장 중요한 첫 번째 이유는 보안입니다. HTTPS는 TLS를 통해 데이터를 이동하기 때문에 보안 계층이 하나 더 늘어납니다. HTTPS 암호화를 사용하면 웹 서버와 클라이언트의 브라우저만 해당 트래픽을 해독할 수 있기 때문에 웹 서버와 클라이언트 브라우저 간에 기밀이 유지됩니다. 클라이언트가 서버와 서로 교환하는 데이터를 다른 관계자는 수정할 수 없다는 것도 HTTPS 연결이 더 안전한 이유입니다.

위에서 언급한 보안 이점 외에, HTTP와 함께 HTTPS도 사용해야 하는 이유가 있습니다. 예를 들어, Google은 2014년부터 검색 결과에서 보안 웹 사이트의 순위를 높게 매기기 시작했습니다. 다시 말해,

HTTPS를 사용하는 사이트가 HTTP만 사용하는 사이트보다 검색 결과에서 순위가 높아졌습니다(다른 조건은 모두 동일한 경우).

순위 지표로 사용되는 HTTPS에 대해 자세히 알아보기

프로세스 개요

Lightsail 인증서를 사용하는 프로세스는 간단합니다. 여기에는 다음 단계가 포함됩니다.

1. Lightsail 인증서를 사용할 수 있는 Lightsail 리소스(예: 로드 밸런서, CDN 배포 또는 컨테이너 서비스)를 생성합니다.
2. Lightsail을 사용하는 도메인의 인증서를 생성합니다.
3. 도메인의 DNS에 표준 이름(CNAME) 레코드를 추가하여 인증서를 검증합니다.
4. 검증된 인증서를 Lightsail 리소스에 연결합니다.
5. 도메인의 DNS를 수정하여 Lightsail 리소스로 트래픽을 라우팅합니다.

인증서가 리소스에 연결되면 도메인을 통해 해당 리소스로 라우팅되는 트래픽이 HTTPS를 사용하여 암호화됩니다.

배포 또는 컨테이너 서비스와 함께 SSL/TLS 인증서 사용

Lightsail 배포 및 컨테이너 서비스에는 HTTPS가 필요합니다. 이러한 리소스 중 하나를 생성하면 HTTPS가 리소스의 기본 도메인(예: 배포의 경우 `https://123456abcdef.cloudfront.net/` 또는 컨테이너 서비스의 경우 `https://container-service-1.123456abcdef.us-west-2.cs.amazonlightsail.com/`)에 대해 기본적으로 활성화됩니다. 등록된 도메인 이름(예: `example.com`)을 배포 또는 컨테이너 서비스와 함께 사용하려면 Lightsail SSL/TLS 인증서를 생성하고 도메인 이름으로 검증한 다음 리소스에서 사용자 지정 도메인을 활성화해야 합니다. 배포 또는 컨테이너 서비스에서 사용자 지정 도메인을 활성화하면 도메인의 검증된 인증서도 리소스에 연결됩니다.

다음 링크를 따라 배포에서 사용자 지정 도메인 및 HTTPS를 활성화할 수 있습니다.

- 배포용 SSL/TLS 인증서 생성
- 배포용 SSL/TLS 인증서 검증
- 배포용 SSL/TLS 인증서 확인
- 배포용 사용자 지정 도메인 활성화
- 배포로 도메인 연결

배포에 대한 자세한 내용은 콘텐츠 전송 네트워크 배포를 참조하세요.

다음 링크를 따라 컨테이너 서비스에서 사용자 지정 도메인 및 HTTPS를 활성화할 수 있습니다.

- 컨테이너 서비스 SSL/TLS 인증서 생성
- 컨테이너 서비스 SSL/TLS 인증서 검증
- 사용자 지정 도메인 활성화 및 관리

컨테이너 서비스에 대한 자세한 내용은 컨테이너 서비스를 참조하세요.

로드 밸런서와 함께 SSL/TLS 인증서 사용

Lightsail 로드 밸런서를 생성하면 포트 80은 일반 HTTP 트래픽을 처리하기 위해 기본적으로 열려 있습니다. 포트 443을 통한 HTTPS 트래픽을 사용하려면 SSL/TLS 인증서를 생성하고 도메인 이름으로 검증한 다음 로드 밸런서에 연결해야 합니다.

로드 밸런서마다 SSL/TLS 인증서를 2개까지 만들 수 있습니다. 로드 밸런서당 1번에 1개의 인증서만 사용할 수 있습니다. 로드 밸런서에서 사용 중인 유효한 인증서를 삭제하면 다른 유효한 인증서를 연결할 때까지 로드 밸런서가 더 이상 지정된 도메인의 HTTPS 트래픽을 처리할 수 없습니다.

다음 링크를 따라 로드 밸런서에서 HTTPS를 활성화할 수 있습니다.

- 로드 밸런서 생성 및 인스턴스 연결
- SSL/TLS 인증서 생성
- 도메인 소유권 확인
- 확인된 인증서를 연결하여 HTTPS 활성화

로드 밸런서에 대한 자세한 내용은 로드 밸런서를 참조하세요.

보안 Lightsail 컨테이너 서비스 도메인에 대한 SSL/TLS 인증서 생성

Lightsail 컨테이너 서비스용 Amazon Lightsail SSL/TLS 인증서를 생성할 수 있습니다. 인증서를 생성할 때 인증서의 기본 도메인 이름과 대체 도메인 이름을 지정해야 합니다. 컨테이너 서비스에 사용자 지정 도메인을 사용하도록 설정하고 인증서를 선택하면 컨테이너 서비스의 사용자 지정 도메인으로 추가할 인증서의 도메인을 최대 4개까지 선택할 수 있습니다. 컨테이너 서비스에 트래픽을 직접 전달하도록 도메인의 DNS 레코드를 업데이트하면 서비스가 트래픽을 허용하고 HTTPS를 사용하여 콘텐츠를 제공합니다. 생성할 수 있는 인증서 수에 대한 할당량이 있습니다. 자세한 내용은 Lightsail 서비스 할당량을 참조하세요.

SSL/TLS 인증서에 대한 자세한 내용은 컨테이너 서비스 인증서를 참조하세요.

사전 조건

시작하기 전에 Lightsail 컨테이너 서비스를 생성해야 합니다. 자세한 내용은 컨테이너 서비스 생성 및 컨테이너 서비스를 참조하세요.

컨테이너 서비스용 SSL/TLS 인증서 생성

컨테이너 서비스용 SSL/TLS 인증서를 생성하려면 다음 절차를 완료하세요.

1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창의 이미지에서 컨테이너를 선택합니다.
3. 인증서를 생성하려는 컨테이너 서비스 이름을 선택합니다.
4. 컨테이너 서비스 관리 페이지에서 사용자 지정 도메인 탭을 선택합니다.
5. 페이지에서 아래로 스크롤하여 Attached certificates(연결된 인증서) 섹션으로 이동합니다.

   다른 Lightsail 리소스용으로 생성된 인증서 및 사용 중인 인증서와 사용하지 않는 인증서를 포함하여 모든 인증서가 페이지의 Attached certificates(연결된 인증서) 섹션 아래에 나열됩니다.
6. [Create certificate]를 선택합니다.
7. 인증서를 식별하도록 Certificate name(인증서 이름) 텍스트 상자에 고유한 이름을 입력합니다. 그런 다음 Continue(계속)를 선택합니다.
8. 인증서에 사용할 기본 도메인 이름(예: `example.com`)을 Specify up to 10 domains or subdomains(최대 10개의 도메인 또는 하위 도메인 지정) 필드에 입력합니다.
9. (선택 사항) 다른 도메인 이름(예: www.example.com))을 Specify up to 10 domains or subdomains(최대 10개의 도메인 또는 하위 도메인 지정) 필드에 입력합니다.

인증서에는 최대 9개의 대체 도메인을 추가할 수 있습니다. 사용자 지정 도메인을 사용하도록 설정하고 서비스에 대한 인증서를 선택한 후 최대 4개의 인증서 도메인을 컨테이너 서비스와 함께 사용할 수 있습니다.

10. [Create certificate]를 선택합니다.

    인증서 요청이 제출되고, 새 인증서의 상태는 Attempting to validate your certificate(인증서 검증을 시도하는 중)로 변경됩니다. 이 기간 도중 Lightsail에서는 인증서의 검증 레코드를 기본 도메인의 DNS에 추가하려 합니다. 그 동안 상태는 Valid(유효함)로 변경됩니다.

    자동 검증이 실패하는 경우 컨테이너 서비스에 사용하기 전에 도메인에 대해 인증서를 검증해야 합니다. 자세한 내용은 컨테이너 서비스 SSL/TLS 인증서 검증을 참조하세요.

주제

- Lightsail 컨테이너 서비스 SSL/TLS 인증서 검증
- Lightsail 컨테이너 서비스 SSL/TLS 인증서 보기

Lightsail 컨테이너 서비스 SSL/TLS 인증서 검증

Amazon Lightsail SSL/TLS 인증서는 생성 후 Lightsail 컨테이너 서비스에 사용하기 전에 검증되어야 합니다. 인증서 요청이 제출된 후 새 인증서의 상태는 Attempting to validate your certificate(인증서 검증을 시도하는 중)로 변경됩니다. 이 기간 도중 Lightsail에서는 인증서의 검증 레코드를 인증서에 대해 지정한 도메인 이름의 DNS에 추가하려 합니다. 그 동안 상태는 Valid(유효함) 또는 Validation timed out(검증 시간 초과)으로 변경됩니다.

자동 검증이 실패하는 경우 인증서 생성 시 지정한 모든 도메인 이름을 제어하는지 확인해야 합니다. 인증서를 검증하려면 인증서에 지정된 각 도메인의 DNS 영역에 표준 이름(CNAME) 레코드를 추가하면 됩니다. 추가해야 하는 레코드는 인증서의 Validation details(검증 세부 정보)에 나열됩니다.

이 안내서에서는 Lightsail DNS 영역을 통해 인증서를 수동으로 검증하는 절차를 안내합니다. Domain.com 또는 GoDaddy와 같은 다른 DNS 호스팅 제공자를 사용하여 인증서를 검증하는 절차와 유사할 수 있습니다. Lightsail DNS 영역에 대한 자세한 내용은 DNS를 참조하세요.

SSL/TLS 인증서에 대한 자세한 내용은 SSL/TLS 인증서를 참조하세요.

사전 조건

시작하기 전에 컨테이너 서비스에 대한 SSL/TLS 인증서를 생성해야 합니다. 자세한 내용은 컨테이너 서비스용 SSL/TLS 인증서 생성을 참조하세요.

CNAME 레코드 값을 가져와 인증서 검증

인증서를 검증하기 위해 도메인에 추가해야 하는 CNAME 레코드를 가져오려면 다음 절차를 완료하세요.

1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창의 이미지에서 컨테이너를 선택합니다.
3. 인증서를 생성하려는 컨테이너 서비스 이름을 선택합니다.
4. 컨테이너 서비스 관리 페이지에서 사용자 지정 도메인 탭을 선택합니다.
5. 페이지에서 아래로 스크롤하여 Attached certificates(연결된 인증서) 섹션으로 이동합니다.

   다른 Lightsail 리소스용으로 생성된 인증서 및 검증 보류 중인 인증서를 포함하여 모든 인증서가 페이지의 Attached certificates(연결된 인증서) 섹션에 나열됩니다.
6. 검증할 인증서를 찾고, Validation details(검증 세부 정보)를 확장한 다음 각 도메인에 대해 추가해야 하는 CNAME 레코드의 이름 및 값을 기록합니다.

   이러한 레코드를 나열된 대로 정확하게 추가해야 합니다. 나중에 참조할 수 있도록 이 값을 복사하여 텍스트 파일에 붙여넣는 것이 좋습니다. 자세한 내용은 가이드의 도메인의 DNS 영역에 CNAME 레코드 추가 섹션을 참조하세요.

CNAME 레코드를 도메인의 DNS 영역에 추가

도메인의 DNS 영역에 CNAME 레코드를 추가하려면 다음 절차를 완료하세요.

1. 왼쪽 탐색 창에서 도메인 및 DNS를 선택합니다.
2. 페이지의 DNS 영역 섹션에서 인증서를 검증하기 위해 CNAME 레코드를 추가할 도메인 이름을 선택합니다.
3. DNS records(DNS 레코드) 탭을 선택합니다.
4. DNS 레코드 관리 페이지에서 Add record(레코드 추가)를 선택합니다.
5. Record type(레코드 유형) 드롭다운에서 CNAME을 선택합니다.
6. Record name(레코드 이름) 텍스트 상자에서 인증서에서 가져온 CNAME 레코드의 Name(이름) 값을 입력합니다.

   Lightsail 콘솔은 도메인의 정점 부분을 미리 채웁니다. 예를 들어, `www.example.com` 하위 도메인을 추가하려면 텍스트 상자에 `www`만 입력하면 되며, 레코드를 저장할 때 Lightsail이 `.example.com` 부분을 추가합니다.

7. Route traffic to(다음으로 트래픽 라우팅) 텍스트 상자에서 인증서에서 가져온 CNAME 레코드의 Value(값) 부분을 입력합니다.
8. 입력한 값이 검증할 인증서에 나열된 값과 일치하는지 확인합니다.
9. DNS 영역에 레코드를 저장하려면 저장 아이콘을 선택합니다.

   다음 단계를 반복하여 검증해야 할 인증서의 도메인에 대한 CNAME 레코드를 추가합니다. 인터넷의 DNS를 통해 변경 사항이 전파될 때까지 기다립니다. 몇 분 후에 인증서 상태가 유효(Valid)로 변경됩니다. 자세한 내용은 이 설명서의 다음 인증서의 상태 확인 섹션을 참조하세요.

인증서 상태 확인

SSL/TLS 인증서의 상태를 확인하려면 다음 절차를 완료하세요.

1. 왼쪽 탐색 창의 이미지에서 컨테이너를 선택합니다.
2. 인증서 상태를 보려는 컨테이너 서비스의 이름을 선택합니다.
3. 컨테이너 서비스 관리 페이지에서 사용자 지정 도메인 탭을 선택합니다.
4. 페이지에서 아래로 스크롤하여 Attached certificates(연결된 인증서) 섹션으로 이동합니다.

   모든 인증서가 페이지의 Attached certificates(연결된 인증서) 섹션 아래에 나열되며, 여기에는 검증 상태가 Pending(보류 중) 및 Valid(유효함) 상태인 인증서가 포함됩니다.

   > **Note**
   >
   > 인증서를 검증하는 동안 Custom domains(사용자 지정 도메인) 페이지를 열어 둔 경우 업데이트된 인증서 상태를 보려면 새로 고쳐야 할 수 있습니다.

   유효 상태는 도메인에 추가한 CNAME 레코드로 인증서를 검증했음을 확인합니다. Details(세부 정보)를 선택하여 인증서의 중요 날짜, 암호화 세부 정보, 식별 및 검증 레코드를 확인합니다. 인증서의 유효 기간은 인증일로부터 13개월이며, 이후 Lightsail은 인증서의 자동 재검증을 시도합니다. 도메인에 추가한 CNAME 레코드는 나열된 다음까지 유효 날짜에 인증서를 다시 검증할 때 필요하므로 삭제하지 마세요.

   SSL/TLS 인증서를 검증한 후 서비스에서 인증서의 도메인 이름을 사용하도록 컨테이너 서비스용 사용자 지정 도메인을 활성화해야 합니다. 자세한 내용은 컨테이너 서비스용 사용자 지정 도메인 사용 설정 및 관리를 참조하세요.

Lightsail 컨테이너 서비스 SSL/TLS 인증서 보기

Lightsail 컨테이너 서비스에 대해 생성한 Amazon Lightsail SSL/TLS 인증서를 볼 수 있습니다. 이 작업은 Lightsail 콘솔에 있는 컨테이너 서비스의 관리 페이지에 액세스하여 수행합니다.

SSL/TLS 인증서에 대한 자세한 내용은 SSL/TLS 인증서를 참조하세요.

사전 조건

시작하기 전에 Lightsail 컨테이너 서비스를 생성해야 합니다. 자세한 내용은 Amazon Lightsail 컨테이너 서비스 생성 및 컨테이너 서비스를 참조하세요.

컨테이너 서비스용 SSL/TLS 인증서도 생성해야 합니다. 자세한 내용은 컨테이너 서비스 SSL/TLS 인증서 생성을 참조하세요.

컨테이너 서비스 SSL/TLS 인증서 확인

컨테이너 서비스 SSL/TLS 인증서를 확인하려면 다음 절차를 완료하세요.

1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창의 이미지에서 컨테이너를 선택합니다.
3. 컨테이너 서비스의 이름을 선택합니다.

   선택한 컨테이너 서비스에 관계없이 모든 인증서를 볼 수 있습니다.
4. 컨테이너 서비스 관리 페이지에서 사용자 지정 도메인 탭을 선택합니다.
5. 페이지에서 아래로 스크롤하여 Attached certificates(연결된 인증서) 섹션으로 이동합니다.

   모든 인증서는 페이지의 Attached certificates(연결된 인증서) 섹션에 나열됩니다. Details(세부 정보)를 선택하여 인증서의 중요 날짜, 암호화 세부 정보, 식별 및 도메인을 확인합니다. Validation details(검증 세부 정보)를 선택하여 인증서의 검증 레코드를 확인합니다. 인증서는 생성 날짜로부터 13개월 동안 유효하며, 이후 Lightsail이 인증서의 자동 재검증을 시도합니다. 도메인에 추가한 CNAME 레코드는 나열된 다음까지 유효(Valid until) 날짜에 인증서를 다시 검증할 때 필요하므로 삭제하지 마세요.

   컨테이너 서비스와 함께 사용할 유효한 SSL/TLS 인증서가 있으면 서비스에서 인증서의 도메인 이름을 사용할 수 있도록 사용자 지정 도메인을 활성화해야 합니다. 자세한 내용은 사용자 지정 도메인 활성화 및 관리를 참조하세요.

SSL/TLS 인증서로 Lightsail CDN 배포 보호

Lightsail 배포용 Amazon Lightsail TLS/SSL 인증서를 생성할 수 있습니다. 인증서를 생성할 때 인증서의 기본 도메인 이름과 대체 도메인 이름을 지정해야 합니다. 배포용 사용자 지정 도메인을 사용하도록 설정하고 인증서를 선택하면 해당 도메인이 배포의 사용자 지정 도메인으로 추가됩니다. 배포를 가리키도록 도메인의 DNS 레코드를 업데이트하면 배포에서 트래픽을 허용하고 HTTPS를 사용하여 콘텐츠를 제공합니다. 생성할 수 있는 인증서 수에 대한 할당량이 있습니다. 자세한 내용은 Lightsail 서비스 할당량을 참조하세요.

SSL/TLS 인증서에 대한 자세한 내용은 SSL/TLS 인증서를 참조하세요.

> **Important**
>
> 배포용 SSL/TLS 인증서를 생성할 때 지정한 도메인 이름은 Amazon CloudFront 서비스의 배포를 포함하여 모든 Amazon Web Services(AWS) 계정의 다른 배포에서 사용할 수 없습니다. 도메인에 대한 인증서를 생성할 수 있지만, 배포와 함께 인증서를 사용할 수는 없습니다.

사전 조건

시작하기 전에 Lightsail 배포를 생성해야 합니다. 자세한 내용은 배포 생성 및 콘텐츠 전송 네트워크 배포를 참조하세요.

배포용 SSL/TLS 인증서 생성

배포용 SSL/TLS 인증서를 생성하려면 다음 절차를 완료하세요.

1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 네트워킹을 선택합니다.
3. 인증서를 생성하려는 배포 이름을 선택합니다.
4. 배포의 관리 페이지에서 사용자 지정 도메인 탭을 선택합니다.
5. 페이지에서 아래로 스크롤하여 Attached certificates(연결된 인증서) 섹션으로 이동합니다.

   다른 배포용으로 생성된 인증서 및 사용 중인 인증서와 사용하지 않는 인증서를 포함하여 모든 배포 인증서가 페이지의 Attached certificates(연결된 인증서) 섹션 아래에 나열됩니다.
6. [Create certificate]를 선택합니다.
7. 인증서를 식별하도록 Certificate name(인증서 이름) 텍스트 상자에 고유한 이름을 입력합니다. 그런 다음 Continue(계속)를 선택합니다.

8. 인증서에 사용할 기본 도메인 이름(예: `example.com`)을 Specify up to 10 domains or subdomains(최대 10개의 도메인 또는 하위 도메인 지정) 필드에 입력합니다.
9. (선택 사항) 대체 도메인 이름(예: `www.example.com`)을 나머지 Specify up to 10 domains or subdomains(최대 10개의 도메인 또는 하위 도메인 지정) 필드에 입력합니다.

   인증서에는 최대 9개의 대체 도메인을 추가할 수 있습니다. 사용자 지정 도메인을 사용하도록 설정하고 배포용 인증서를 선택하고 나면 배포와 함께 인증서의 모든 도메인을 사용할 수 있습니다.
10. 생성(Create)을 선택합니다.

    인증서 요청이 제출되고, 새 인증서의 상태는 Attempting to validate your certificate(인증서 검증을 시도하는 중)로 변경됩니다. 이 기간 도중 Lightsail에서는 인증서의 검증 레코드를 기본 도메인의 DNS에 추가하려 합니다. 그 동안 상태는 Valid(유효함)로 변경됩니다.

    자동 검증이 실패하는 경우 배포에 사용하기 전에 도메인에 대해 인증서를 검증해야 합니다. 자세한 내용은 배포용 SSL/TLS 인증서 검증을 참조하세요.

주제

- Lightsail 배포용 SSL/TLS 인증서 확인
- Lightsail 배포용 SSL/TLS 인증서 검증
- 최소 TLS 프로토콜 버전으로 Lightsail 배포 보안 유지
- Lightsail 배포에서 미사용 SSL/TLS 인증서 삭제

Lightsail 배포용 SSL/TLS 인증서 확인

Lightsail 배포용으로 생성한 Amazon Lightsail SSL/TLS 인증서를 볼 수 있습니다. 이렇게 하려면 Lightsail 콘솔에 있는 배포의 관리 페이지에 액세스해야 합니다.

SSL/TLS 인증서에 대한 자세한 내용은 SSL/TLS 인증서를 참조하세요.

사전 조건

시작하기 전에 Lightsail 배포를 생성해야 합니다. 자세한 내용은 배포 생성 및 콘텐츠 전송 네트워크 배포를 참조하세요.

또한, 배포용 SSL/TLS 인증서를 생성해야 합니다. 자세한 내용은 배포용 SSL/TLS 인증서 생성을 참조하세요.

배포용 SSL/TLS 인증서 확인

배포용 SSL/TLS 인증서를 보려면 다음 절차를 완료하세요.

1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 네트워킹을 선택합니다.
3. 배포의 이름을 선택합니다.

   선택한 배포에 관계없이 모든 인증서를 볼 수 있습니다.
4. 배포의 관리 페이지에서 사용자 지정 도메인 탭을 선택합니다.
5. 페이지에서 아래로 스크롤하여 Attached certificates(연결된 인증서) 섹션으로 이동합니다.

   모든 배포 인증서는 페이지의 Attached certificates(연결된 인증서) 섹션에 나열됩니다. Validation details(검증 세부 정보)를 확장하여 인증서의 중요 날짜, 암호화 세부 정보, 식별 및 검증 레코드를 확인합니다. 인증서는 생성 날짜로부터 13개월 동안 유효하며, 이후 Lightsail이 인증서의 자동 재검증을 시도합니다. 도메인에 추가한 CNAME 레코드는 나열된 유효 기간 만료(Valid until) 날짜에 인증서를 다시 검증할 때 필요하므로 삭제하지 마세요.

   배포와 함께 사용할 유효한 SSL/TLS 인증서가 있으면 배포에서 인증서의 도메인 이름을 사용할 수 있도록 사용자 지정 도메인을 활성화해야 합니다. 자세한 내용은 배포용 사용자 지정 도메인 사용을 참조하세요.

Lightsail 배포용 SSL/TLS 인증서 검증

Amazon Lightsail SSL/TLS 인증서는 생성 후 Lightsail 배포에 사용하기 전에 검증되어야 합니다. 인증서 요청이 제출된 후 새 인증서의 상태는 Attempting to validate your certificate(인증서 검증을 시도하는 중)로 변경됩니다. 이 기간 도중 Lightsail에서는 인증서의 검증 레코드를 인증서에 대해 지정한 도메인 이름의 DNS에 추가하려 합니다. 그 동안 상태는 Valid(유효함) 또는 Validation timed out(검증 시간 초과)으로 변경됩니다.

자동 검증이 실패하는 경우 인증서 생성 시 지정한 모든 도메인 이름을 제어하는지 확인해야 합니다. 인증서를 검증하려면 인증서에 지정된 각 도메인의 DNS 영역에 표준 이름(CNAME) 레코드를 추가하면 됩니다. 추가해야 하는 레코드는 인증서의 Validation details(검증 세부 정보)에 나열됩니다.

이 안내서에서는 Lightsail DNS 영역을 통해 인증서를 수동으로 검증하는 절차를 안내합니다. Domain.com 또는 GoDaddy와 같은 다른 DNS 호스팅 제공자를 사용하여 인증서를 검증하는 절차와 유사할 수 있습니다. Lightsail DNS 영역에 대한 자세한 내용은 DNS를 참조하세요.

SSL/TLS 인증서에 대한 자세한 내용은 SSL/TLS 인증서를 참조하세요.

목차

- 사전 조건
- CNAME 레코드 값을 가져와 인증서 검증
- CNAME 레코드를 도메인의 DNS 영역에 추가
- 배포 인증서 상태 확인

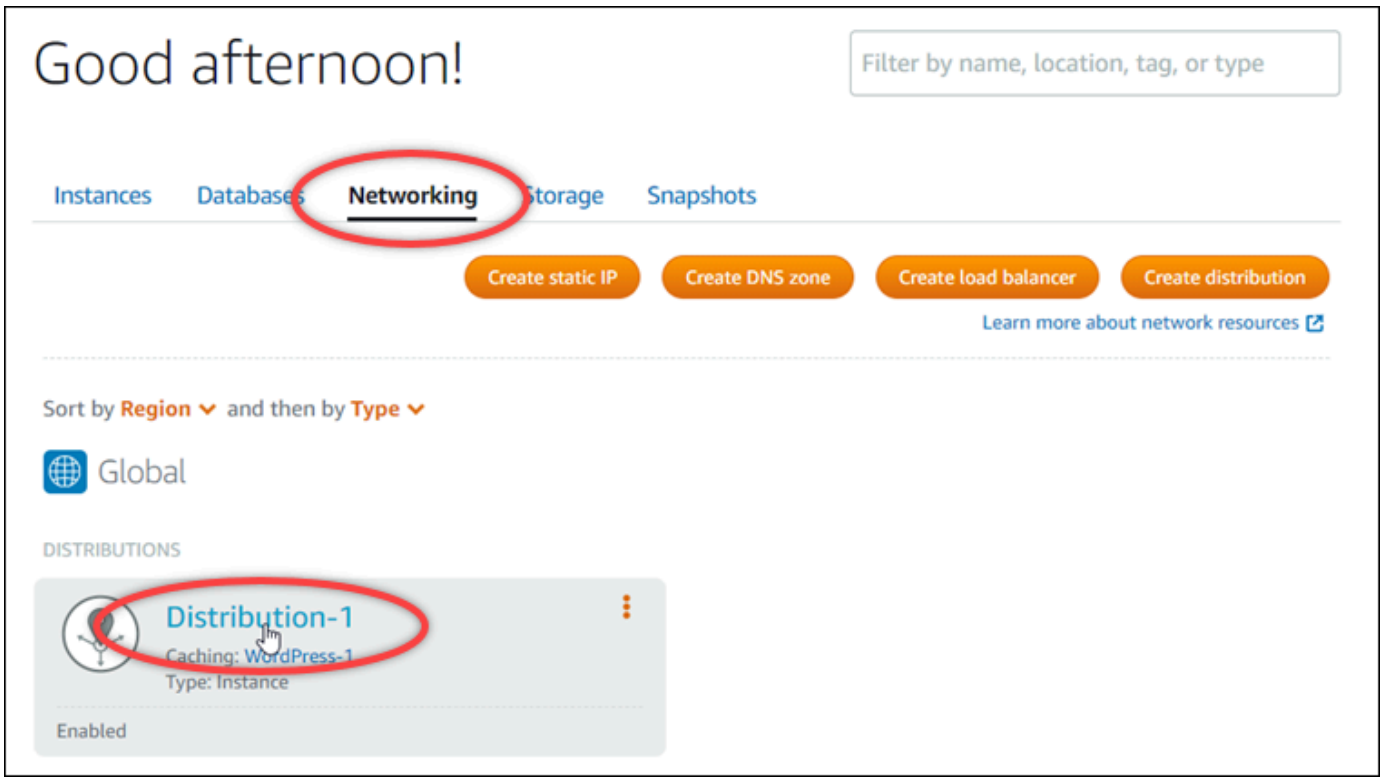
사전 조건

시작하기 전에 베포에 대한 SSL/TLS 인증서를 생성해야 합니다. 자세한 내용은 배포용 SSL/TLS 인증서 생성을 참조하세요.

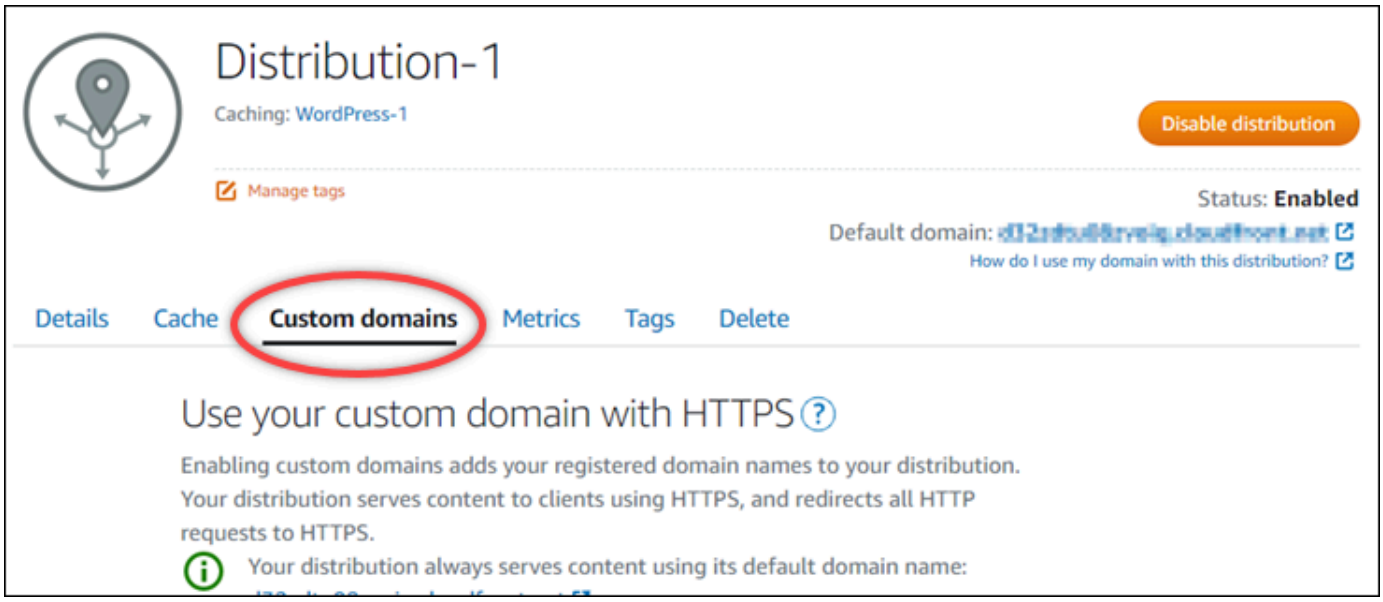
CNAME 레코드 값을 가져와 인증서 검증

인증서를 검증하기 위해 도메인에 추가해야 하는 CNAME 레코드를 가져오려면 다음 절차를 완료하세요.

1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 네트워킹을 선택합니다.
3. 인증서의 CNAME 레코드 값을 가져오려는 배포의 이름을 선택합니다.



4. 배포의 관리 페이지에서 사용자 지정 도메인(Custom domains) 탭을 선택합니다.



5. 페이지에서 아래로 스크롤하여 Attached certificates(연결된 인증서) 섹션으로 이동합니다.

   다른 Lightsail 리소스용으로 생성된 인증서 및 검증 보류 중인 인증서를 포함하여 모든 배포 인증서가 페이지의 Attached certificates(연결된 인증서) 섹션에 나열됩니다.

6. 검증할 인증서를 찾고, Validation details(검증 세부 정보)를 확장한 다음 각 도메인에 대해 추가해야 하는 CNAME 레코드의 Name(이름) 및 Value(값)를 기록합니다.

   이러한 레코드를 나열된 대로 정확하게 추가해야 합니다. 나중에 참조할 수 있도록 이 값을 복사하여 텍스트 파일에 붙여넣는 것이 좋습니다. 자세한 내용은 가이드의 도메인의 DNS 영역에 CNAME 레코드 추가 섹션을 참조하세요.

CNAME 레코드를 도메인의 DNS 영역에 추가

도메인의 DNS 영역에 CNAME 레코드를 추가하려면 다음 절차를 완료하세요.

1. 왼쪽 탐색 창에서 도메인 및 DNS를 선택합니다.
2. 페이지의 DNS 영역 섹션에서 인증서를 검증하기 위해 CNAME 레코드를 추가할 도메인 이름을 선택합니다.
3. DNS records(DNS 레코드) 탭을 선택합니다.
4. DNS 레코드 관리 페이지에서 Add record(레코드 추가)를 선택합니다.
5. Record type(레코드 유형) 드롭다운에서 CNAME을 선택합니다.

6. Record name(레코드 이름) 텍스트 상자에서 인증서에서 가져온 CNAME 레코드의 Name(이름) 값을 입력합니다.

   Lightsail 콘솔은 도메인의 정점 부분을 미리 채웁니다. 예를 들어, `www.example.com` 하위 도메인을 추가하려면 텍스트 상자에 `www`만 입력하면 되며, 레코드를 저장할 때 Lightsail이 `.example.com` 부분을 추가합니다.

7. Route traffic to(다음으로 트래픽 라우팅) 텍스트 상자에서 인증서에서 가져온 CNAME 레코드의 Value(값) 부분을 입력합니다.

8. 입력한 값이 검증할 인증서에 나열된 값과 일치하는지 확인합니다.

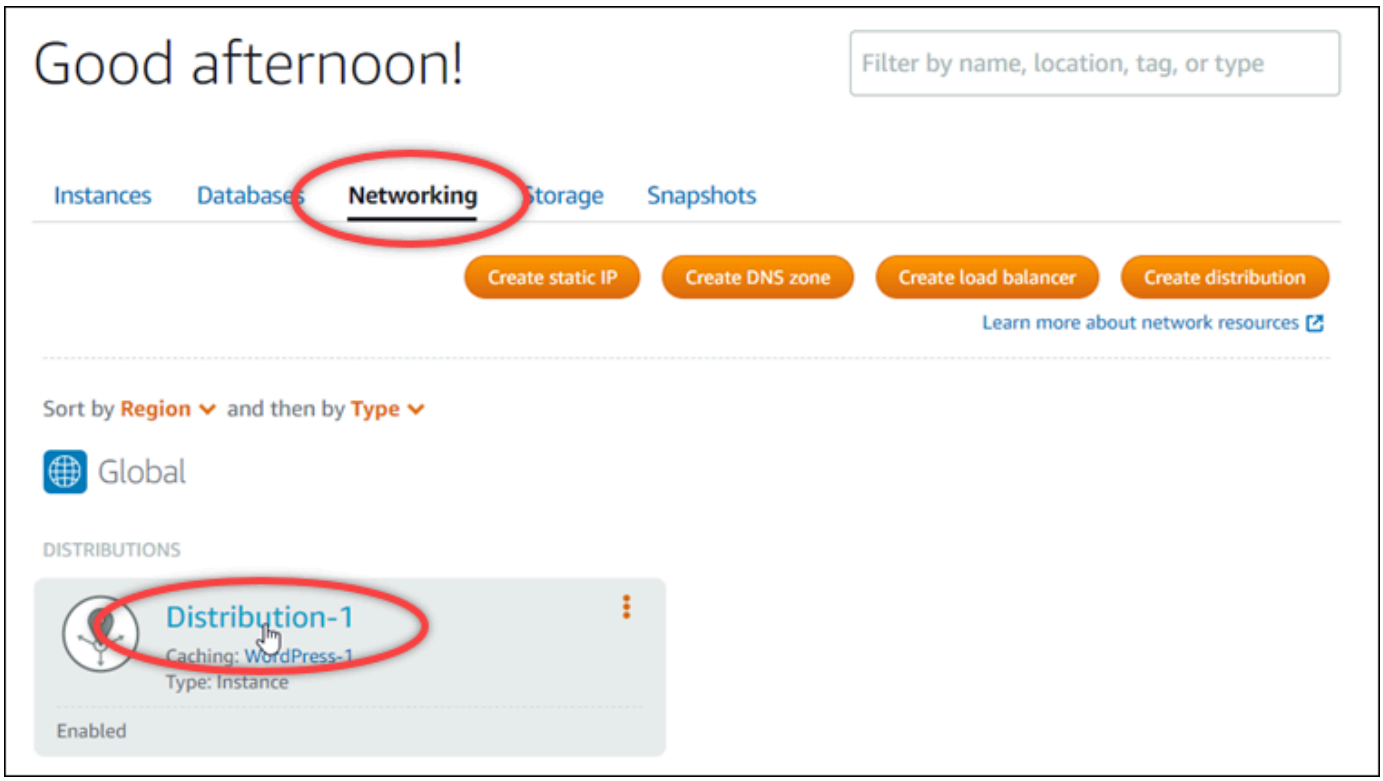
9. DNS 영역에 레코드를 저장하려면 저장 아이콘을 선택합니다.

   다음 단계를 반복하여 검증해야 할 인증서의 도메인에 대한 CNAME 레코드를 추가합니다. 인터넷의 DNS를 통해 변경 사항이 전파될 때까지 기다립니다. 몇 분 후에 배포 인증서 상태가 유효(Valid)로 변경됩니다. 자세한 내용은 이 설명서의 다음 배포 인증서의 상태 확인 섹션을 참조하세요.

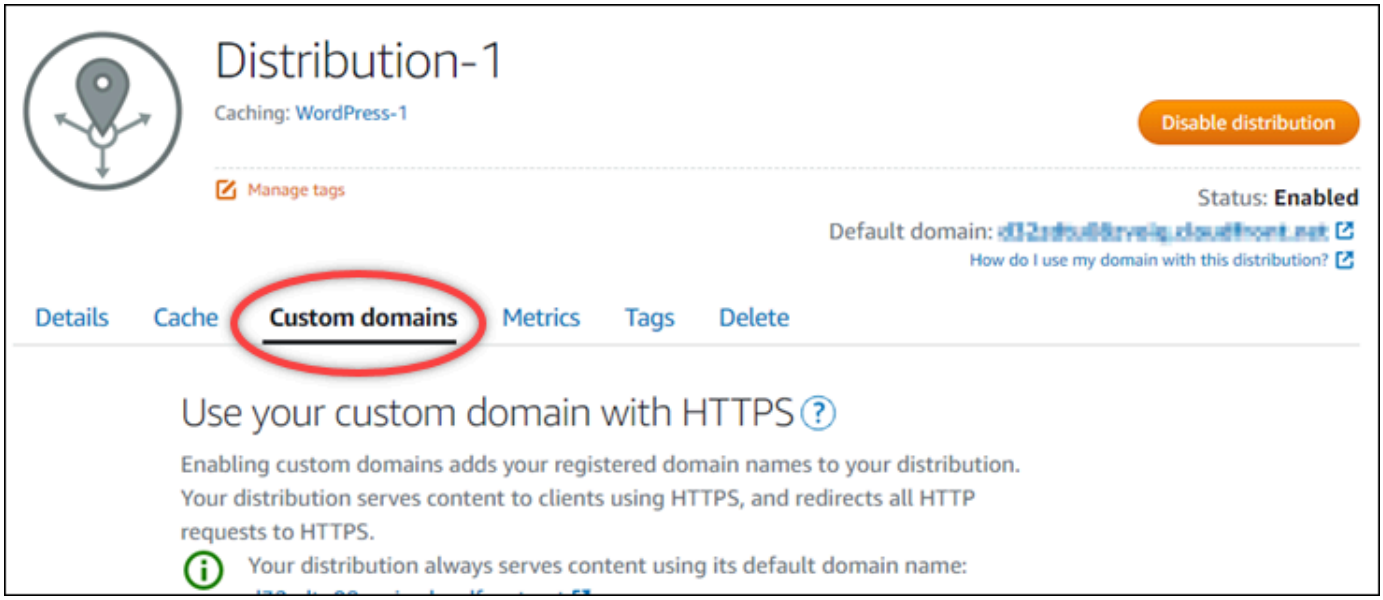
배포 인증서 상태 확인

배포용 SSL/TLS 인증서의 상태를 확인하려면 다음 절차를 완료하세요.

1. 왼쪽 탐색 창에서 네트워킹을 선택합니다.
2. 인증서 상태를 보려는 배포의 이름을 선택합니다.

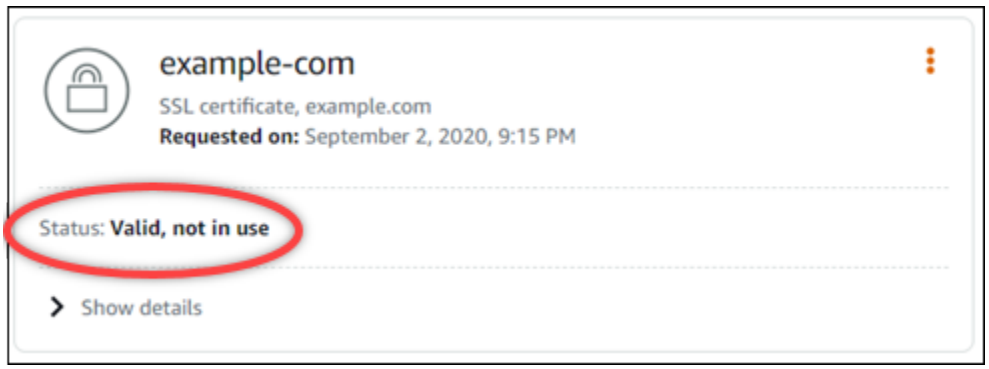


3. 배포의 관리 페이지에서 사용자 지정 도메인(Custom domains) 탭을 선택합니다.



4. 페이지에서 아래로 스크롤하여 Attached certificates(연결된 인증서) 섹션으로 이동합니다.

   모든 배포 인증서가 페이지의 Attached certificates(연결된 인증서) 섹션 아래에 나열되며, 여기에는 Pending validation(검증 보류 중) 및 Valid(유효함) 상태인 인증서가 포함됩니다.



유효 상태는 도메인에 추가한 CNAME 레코드로 인증서를 검증했음을 확인합니다. Details(세부 정보)를 선택하여 인증서의 중요 날짜, 암호화 세부 정보, 식별 및 검증 레코드를 확인합니다. 인증서의 유효 기간은 인증일로부터 13개월이며, 이후 Lightsail은 인증서의 자동 재검증을 시도합니다. 도메인에 추가한 CNAME 레코드는 나열된 다음까지 유효(Valid until) 날짜에 인증서를 다시 검증할 때 필요하므로 삭제하지 마세요.

SSL/TLS 인증서를 검증한 후 배포에서 인증서의 도메인 이름을 사용하도록 배포용 사용자 지정 도메인을 활성화해야 합니다. 자세한 내용은 배포용 사용자 지정 도메인 사용을 참조하세요.

최소 TLS 프로토콜 버전으로 Lightsail 배포 보안 유지

Amazon Lightsail은 SSL/TLS 인증서를 사용하여 Lightsail 배포에 사용할 수 있는 사용자 지정(등록) 도메인을 검증합니다. 이 안내서는 SSL/TLS 인증서에 대해 구성할 수 있는 최종 사용자 최소 TLS 프로토콜 버전(프로토콜 버전)에 대한 정보를 제공합니다. SSL/TLS 인증서에 대한 자세한 내용은 Lightsail의 SSL/TLS 인증서를 참조하세요. 뷰어는 Lightsail 배포와 연결된 엣지 로케이션에 HTTP 요청을 수행하는 애플리케이션입니다. 배포에 대한 자세한 내용은 Lightsail에서 콘텐츠 전송 네트워크 배포를 참조하세요.

배포에 사용자 지정 도메인을 활성화하면 `TLSv1.2_2021` 프로토콜 버전이 기본적으로 구성됩니다. 다른 프로토콜 버전 구성은 이 가이드의 뒷부분에 설명되어 있습니다. Lightsail 배포는 사용자 지정 TLS 프로토콜 버전을 지원하지 않습니다.

지원되는 프로토콜

Lightsail 배포는 다음 TLS 프로토콜로 구성할 수 있습니다.

- (권장) TLSv1.2_2021
- TLSv1.2_2019
- TLSv1.2_2018

- TLSv1.1_2016

사전 조건

아직 수행하지 않은 경우 다음 사전 조건을 완료합니다.

- Lightsail 콘텐츠 전송 네트워크 배포 생성
- 배포용 SSL/TLS 인증서 생성
- 배포용 SSL/TLS 인증서 검증
- 배포용 사용자 지정 도메인 활성화
- 배포로 도메인 연결

배포를 위한 최소 TLS 프로토콜 버전 식별

Lightsail 배포에 사용할 최소 TLS 프로토콜 버전을 식별하려면 다음 단계를 완료하세요.

> **Note**
>
> 이 가이드에서는 AWS CloudShell 를 사용하여 업그레이드를 수행합니다. CloudShell은 브라우저 기반의 사전 인증된 셸로, Lightsail 콘솔에서 직접 실행할 수 있습니다. CloudShell을 사용하면 Bash, PowerShell 또는 Z 쉘과 같은 원하는 쉘을 사용하여 AWS CLI 명령을 실행할 수 있습니다. 명령줄 도구를 다운로드하거나 설치할 필요 없이 이 작업을 수행할 수 있습니다. CloudShell을 설정하고 사용하는 방법에 대한 자세한 내용은 Lightsail의AWS CloudShell 섹션을 참조하세요.

1. 터미널, AWS CloudShell 또는 명령 프롬프트 창을 엽니다.
2. 다음 명령을 입력하여 Lightsail 배포의 최소 TLS 프로토콜 버전을 식별합니다.

   ```
   aws lightsail get-distributions --distribution-name DistributionName --region us-east-1 | grep "viewerMinimumTlsProtocolVersion"
   ```

   명령에서 *DistributionName*을 수정하려는 배포의 이름으로 바꿉니다.

   예

```
aws lightsail get-distributions --distribution-name Distribution-1 --region us-
east-1 | grep "viewerMinimumTlsProtocolVersion"
```

명령은 배포를 위한 최소 TLS 프로토콜 버전의 ID를 반환합니다.

예

```
"viewerMinimumTlsProtocolVersion": "TLSv1.2_2021"
```

를 사용하여 최소 TLS 프로토콜 버전 구성 AWS CLI

AWS Command Line Interface (AWS CLI)를 사용하여 TLS 프로토콜 버전을 구성하려면 다음 절차를 완료합니다. `update-distribution` 명령을 사용하여 이 작업을 수행할 수 있습니다. 자세한 내용은 AWS CLI 명령 레퍼런스의 update-distribution 속성을 참조하세요.

1. 터미널, AWS CloudShell 또는 명령 프롬프트 창을 엽니다.
2. 다음 명령을 입력하여 배포의 최소 TLS 프로토콜 버전을 변경합니다.

```
aws lightsail update-distribution --distribution-name DistributionName --viewer-
minimum-tls-protocol-version ProtocolVersion
```

명령에서 아래 예 텍스트를 사용자의 값으로 대체합니다.

- 업데이트할 배포 이름이 있는 *DistributionName*입니다.
- 유효한 TLS 프로토콜 버전이 있는 *ProtocolVersion*입니다. 예를 들면 `TLSv1.2_2021` 또는 `TLSv1.2_2019`입니다.

예제:

```
aws lightsail update-distribution --distribution-name  MyDistribution --viewer-
minimum-tls-protocol-version TLSv1.2_2021
```

변경 사항이 적용되는 데에는 약간의 시간이 걸립니다.

Lightsail 배포에서 미사용 SSL/TLS 인증서 삭제

> **Warning**
>
> SSL/TLS 인증서 삭제는 최종적이며 취소할 수 없는 작업입니다.

Amazon Lightsail SSL/TLS 인증서를 더 이상 사용하지 않는 경우 삭제할 수 있습니다. 예를 들면 인증서가 만료되었고 확인을 마친 업데이트 인증서를 이미 연결한 경우가 여기에 해당합니다. 인증서에 대한 자세한 내용은 SSL/TLS 인증서를 참조하세요. 배포에 대한 자세한 내용은 콘텐츠 전송 네트워크 배포를 참조하세요.

365일 동안 생성할 수 있는 인증서 할당량이 있습니다. 자세한 내용은 AWS 일반 참조의 Lightsail 서비스 할당량을 참조하세요.

배포용 SSL/TLS 인증서 삭제

> **Important**
>
> 삭제할 인증서가 사용 중인 경우에는 삭제(Delete) 옵션이 지원되지 않습니다. 사용 중인 인증서를 삭제하려면 먼저 인증서를 사용하는 배포의 사용자 지정 도메인을 변경하거나 인증서를 사용하는 배포의 사용자 지정 도메인을 비활성화해야 합니다.

배포용 SSL/TLS 인증서를 삭제하려면 다음 절차를 완료하세요.

1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 네트워킹을 선택합니다.
3. SSL/TLS 인증서를 삭제할 배포 이름을 선택합니다. 인증서를 현재 사용하지 않는 경우 인증서 전체가 모든 배포에 나열되므로 원하는 배포를 선택할 수 있습니다.
4. 배포의 관리 페이지에서 사용자 지정 도메인(Custom domains) 탭을 선택합니다.
5. 페이지의 인증서 섹션에서 삭제할 인증서의 줄임표 아이콘(:)을 선택하고 삭제를 선택합니다.

   삭제할 인증서가 사용 중인 경우에는 삭제(Delete) 옵션이 지원되지 않습니다. 사용 중인 인증서를 삭제하려면 먼저 인증서를 사용 중인 배포의 사용자 지정 도메인을 변경하거나 인증서를 사용 중인 배포에서 사용자 지정 도메인을 사용하지 않도록 설정해야 합니다. 자세한 내용은 배포용 사용자 지정 도메인 변경 및 배포용 사용자 지정 도메인 사용을 참조하세요.

6. 예, 삭제를 선택하여 삭제를 확인합니다.

Lightsail 로드 밸런서용 SSL/TLS와 함께 HTTPS 활성화

Lightsail 로드 밸런서를 생성한 뒤 TLS(전송 계층 보안) 인증서를 연결하여 HTTPS를 지원할 수 있습니다. SSL/TLS 인증서가 있으면 로드 밸런서에서 암호화된 웹 트래픽을 처리할 수 있으며, 따라서 사용자에게 보다 안전한 환경을 제공할 수 있습니다. 자세한 내용은 SSL/TLS 인증서를 참조하세요.

사전 조건

시작하기 전에 다음을 준비해야 합니다.

- Lightsail 로드 밸런서입니다. 자세한 내용은 로드 밸런서 생성을 참조하세요.

인증서 생성 요청

1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 네트워킹을 선택합니다.
3. SSL/TLS 인증서를 구성하려는 로드 밸런서의 이름을 선택합니다.
4. Custom domains(사용자 지정 도메인) 탭을 선택합니다.
5. [Create certificate]를 선택합니다.
6. 인증서에 대한 이름을 입력하거나 기본값을 수락합니다.

   리소스 이름:

   - Lightsail 계정 AWS 리전 의 각 내에서 고유해야 합니다.
   - 2~255자의 문자로 구성되어야 합니다.
   - 영숫자 문자 또는 숫자로 시작하고 끝나야 합니다.
   - 영숫자 문자, 숫자, 마침표, 대시, 밑줄이 포함될 수 있습니다.
7. 기본 도메인(`www.example.com`)과 최대 9개의 대체 도메인 또는 하위 도메인을 입력합니다.

   자세한 내용은 SSL/TLS 인증서에 대체 도메인 및 하위 도메인 추가를 참조하십시오.
8. [Create certificate]를 선택합니다.

   Lightsail에서 확인 프로세스를 시작합니다. 72시간 안에 본인 소유의 도메인임을 확인해야 합니다.

인증서를 생성하고 나면 해당 인증서와 도메인 이름 및 모든 대체 도메인과 하위 도메인이 표시됩니다. 각각의 도메인과 하위 도메인에 대해 DNS 레코드를 생성해야 합니다.

다음 단계

- 본인 소유의 도메인 확인

주제

- Lightsail SSL/TLS 인증서에 대체 도메인 및 하위 도메인 추가
- Lightsail에서 CNAME 레코드를 사용하여 SSL/TLS 인증서 도메인 확인
- 확인된 SSL/TLS 인증서를 Lightsail 로드 밸런서에 연결
- Lightsail 로드 밸런서에서 SSL/TLS 인증서 제거

Lightsail SSL/TLS 인증서에 대체 도메인 및 하위 도메인 추가

Lightsail 로드 밸런서용 SSL/TLS 인증서를 생성할 때 대체 도메인과 하위 도메인을 추가할 수 있습니다. 이러한 대체 이름은 로드 밸런서로 이동하는 모든 트래픽을 암호화하는 데 도움이 됩니다.

기본 도메인을 지정할 때는 `www.example.com` 같은 정식 도메인 이름이나 `example.com` 같은 약식 도메인 이름을 사용할 수 있습니다.

도메인과 하위 도메인의 수를 합해서 10개를 초과할 수 없으므로, 인증서에 대체 도메인과 하위 도메인을 최대 9개까지 추가할 수 있습니다. 다음 목록과 비슷한 항목을 추가하게 될 것입니다.

- example.com
- example.net
- blog.example.com
- myexamples.com

대체 도메인 및 하위 도메인으로 인증서를 생성하려면

1. 아직 로드 밸런서가 없으면 로드 밸런서를 생성합니다.
2. 왼쪽 탐색 창에서 네트워킹을 선택합니다.
3. Lightsail 로드 밸런서를 선택합니다.

4. Custom domains(사용자 지정 도메인) 탭을 선택합니다.
5. [Create certificate]를 선택합니다.
6. 인증서에 대한 이름을 입력하거나 기본 이름을 수락합니다.

   리소스 이름:

   - Lightsail 계정 AWS 리전 의 각 내에서 고유해야 합니다.
   - 2~255자의 문자로 구성되어야 합니다.
   - 영숫자 문자 또는 숫자로 시작하고 끝나야 합니다.
   - 영숫자 문자, 숫자, 마침표, 대시, 밑줄이 포함될 수 있습니다.
7. 기본 도메인(`www.example.com`)과 최대 9개의 대체 도메인 또는 하위 도메인을 입력합니다.
8. [Create certificate]를 선택합니다.

   도메인을 생성한 뒤 72시간 안에 본인 소유의 도메인임을 확인해야 합니다.

다음 단계

- DNS를 사용하여 도메인 소유권 확인

  확인을 마친 뒤, 검증된 인증서를 선택하여 Lightsail 로드 밸런서에 연결할 수 있습니다.
- 세션 지속성 활성화

Lightsail에서 CNAME 레코드를 사용하여 SSL/TLS 인증서 도메인 확인

Lightsail에서 SSL/TLS 인증서를 생성한 후에는 인증서에 추가한 모든 도메인과 하위 도메인을 제어하고 있는지 확인해야 합니다.

목차

- 1단계: 도메인의 Lightsail DNS 영역 생성
- 2단계: 도메인의 DNS 영역에 TXT 레코드 추가
- 다음 단계

1단계: 도메인의 Lightsail DNS 영역 생성

아직 생성하지 않은 경우, 도메인의 Lightsail DNS 영역을 생성합니다. 자세한 내용은 DNS 영역을 생성하여 도메인의 DNS 레코드 관리를 참조하세요.

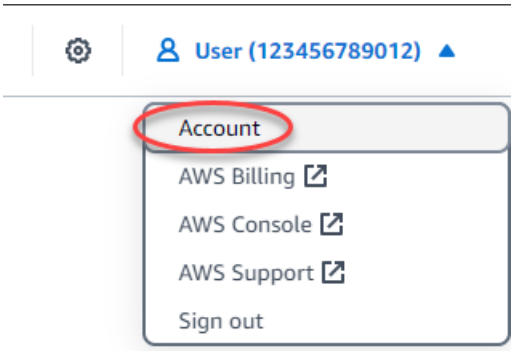
2단계: 도메인의 DNS 영역에 TXT 레코드 추가

생성한 인증서는 정식 이름(CNAME) 레코드 집합을 제공합니다. 도메인의 DNS 영역에 이러한 레코드를 추가하여 해당 도메인을 소유하거나 제어함을 확인합니다.

> **Important**
>
> Lightsail은 인증서를 생성하는 동안 지정한 도메인 또는 하위 도메인을 제어하는지 자동으로 확인하려고 시도합니다. Create certificate(인증서 생성)을 선택한 후 CNAME 레코드는 도메인의 DNS 영역에 추가됩니다. 자동 검증이 성공적인 경우 인증서의 상태가 Attempting to validate your certificate(인증서 검증을 시도하는 중)에서 Valid, in use(유효함, 사용 중)로 변경됩니다.
> 자동 검증이 실패하는 경우 다음 단계를 진행합니다.

다음 단계에서는 Lightsail 콘솔에서 CNAME 레코드를 가져와서 도메인의 DNS 영역에 추가하는 방법을 보여 줍니다.

1. Lightsail 콘솔에 로그인합니다.
2. Lightsail 홈 페이지의 상단 탐색 메뉴에서 계정을 선택합니다.
3. 드롭다운 메뉴에서 계정(Account)을 선택합니다.



4. 인증서 탭을 선택합니다.
5. 확인할 인증서를 찾고 각 도메인에 대해 추가해야 하는 CNAME 레코드의 Name(이름) 및 Value(값)를 기록합니다.

   Ctrl+C(Windows를 사용하는 경우) 또는 Cmd+C(Mac을 사용하는 경우)를 눌러 클립보드에 복사합니다.

6. 텍스트 편집기를 엽니다(Windows를 사용하는 경우 메모장, Mac을 사용하는 경우 텍스트 편집기). 텍스트 파일에서 Ctrl+V(Windows를 사용하는 경우) 또는 Cmd+V(Mac을 사용하는 경우)를 눌러 값을 텍스트 파일에 붙여넣습니다.

   이 텍스트 파일을 열어 둡니다. 이 설명서에서 나중에 도메인의 DNS 영역에 레코드를 추가할 때 이러한 CNAME 값이 필요합니다.

7. Lightsail 콘솔의 상단 탐색 모음에서 홈(Home)을 선택합니다.
8. Lightsail 홈 페이지에서 Domains & DNS(도메인 및 DNS)를 선택합니다.
9. 인증서를 사용할 도메인의 DNS 영역을 선택합니다.
10. DNS records(DNS 레코드) 탭에서 Add record(레코드 추가)를 선택합니다.
11. 레코드 유형의 CNAME을 선택합니다.
12. 인증서의 CNAME 레코드를 포함하는 텍스트 파일로 전환합니다.

    CNAME 레코드의 이름을 복사합니다. 예를 들어 `_1bfb0b9ef15a50f9041e559d2c67b760`입니다.
13. DNS 레코드 페이지로 전환하고 Record name(레코드 이름) 필드에 Name(이름)을 붙여넣습니다.

> **Important**
>
> 도메인 이름(예: `.example.com`)이 포함된 CNAME 레코드를 추가하면 도메인 이름이 중복될 수 있습니다(예: `.example.com.example.com`). 중복을

방지하려면 필요한 CNAME의 일부만 추가되도록 항목을 편집합니다. 그러면 `_1bfb0b9ef15a50f9041e559d2c67b760`이 됩니다.

14. CNAME 레코드의 값을 복사합니다. 예를 들어 `_c9a0c385eda13283350e35f297469a13.hkvuiqjoua.acm-validations.aws.`입니다.
15. DNS 레코드 페이지로 전환하고 Route traffic to(트래픽 라우팅 대상) 필드에 Value(값)를 붙여넣습니다.
16. Save(저장) 아이콘을 선택하여 레코드를 추가합니다.
17. 대체 하위 도메인이 있는 경우, 레코드 추가를 선택하여 다른 레코드를 추가합니다.

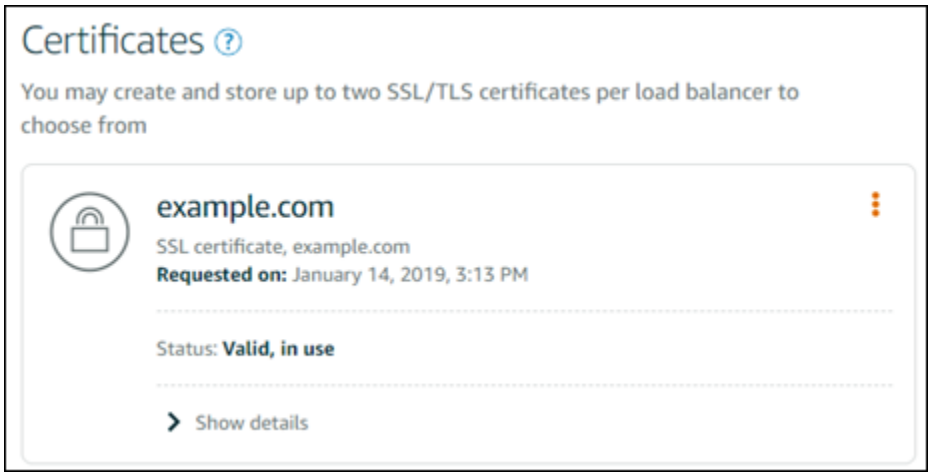
> **Note**
>
> 대체 도메인 또는 하위 도메인에 대해 자세히 알아보려면 Amazon Lightsail의 SSL/TLS 인증서에 대체 도메인 및 하위 도메인 추가를 참조하세요.

18. 11~17단계를 반복하여 대체 하위 도메인의 CNAME 레코드를 추가합니다.

DNS 영역 관리 페이지에 있는 동안 별칭(A) 레코드를 추가하여 로드 밸런서, 또는 다른 Lightsail 리소스에 연결할 수 있습니다.

완료하면 DNS 영역이 다음 스크린샷과 같이 보일 것입니다.

시간이 지나 도메인이 확인되면 인증서에 다음 메시지가 표시됩니다.

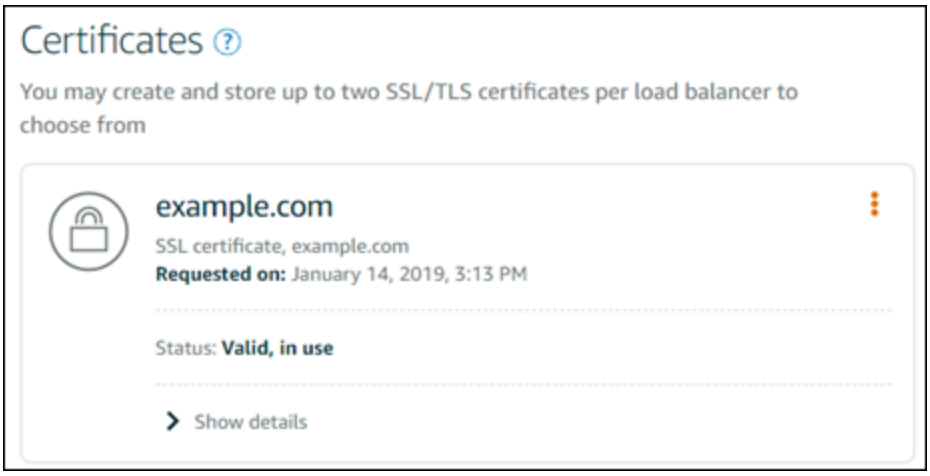


다음 단계

도메인이 확인되면 [검증된 SSL/TLS 인증서를 로드 밸런서에 연결](#)할 준비가 됩니다.

확인된 SSL/TLS 인증서를 Lightsail 로드 밸런서에 연결

도메인을 제어하고 있는지 확인한 후 인증서 상태가 Valid(유효함)로 변경됩니다.



다음 단계는 이 인증서를 Lightsail 로드 밸런서에 연결하는 것입니다.

1. Lightsail 홈 페이지에서 네트워킹을 선택합니다.
2. 로드 밸런서를 선택합니다.
3. Custom domains(사용자 지정 도메인) 탭을 선택합니다.
4. Certificates(인증서) 섹션에서 Attach certificate(인증서 연결)을 선택합니다.
5. 드롭다운 목록에서 인증서를 선택합니다.
6. 연결을 선택하여 인증서를 연결합니다.

Lightsail 로드 밸런서에서 SSL/TLS 인증서 제거

더 이상 사용하지 않는 SSL/TLS 인증서는 삭제할 수 있습니다. 예를 들면 인증서가 만료되었고 확인을 마친 업데이트 인증서를 이미 연결한 경우가 여기에 해당합니다. 인증서를 삭제하기 전에 복제해 두려면 아래의 5단계와 동일한 바로 가기 메뉴에서 복제를 선택하면 됩니다.

> **Important**
>
> 현재 사용 중인 유효한 인증서를 삭제하면, 로드 밸런서는 더 이상 암호화된(HTTPS) 트래픽을 처리할 수 없게 됩니다. 그러나 Lightsail 로드 밸런서는 암호화되지 않은(HTTP) 트래픽은 여전히 처리합니다.

SSL/TLS 인증서 삭제는 최종적이며 취소할 수 없는 작업입니다. 365일 동안 생성할 수 있는 인증서 할당량이 있습니다. 자세한 내용은 AWS Certificate Manager 사용 설명서의 할당량을 참조하세요.

1. 왼쪽 탐색 창에서 네트워킹을 선택합니다.
2. SSL/TLS 인증서가 연결되어 있는 로드 밸런서를 선택합니다.
3. 로드 밸런서 관리 페이지에서 인바운드 트래픽(Inbound traffic) 탭을 선택합니다.
4. 페이지의 인증서 섹션에서 삭제할 인증서의 줄임표 아이콘(⋮)을 선택하고 삭제를 선택합니다.

   삭제할 인증서가 사용 중인 경우에는 삭제(Delete) 옵션이 지원되지 않습니다. 사용 중인 인증서를 삭제하려면 먼저 인증서를 사용하는 로드 밸런서의 인증서를 변경하거나 인증서를 사용하는 로드 밸런서에서 HTTPS를 비활성화해야 합니다.

Lightsail 인스턴스의 이메일 스팸을 방지하도록 역방향 DNS 구성

역방향 DNS(도메인 이름 시스템) 검색은 이메일 서버에서 메시지가 시작된 위치를 추적하고, 메시지가 스팸 또는 악성이 아닌지 확인하는 데 사용됩니다. 역방향 DNS 검색은 IP 주소의 도메인 이름을 반환합니다. 이는 도메인의 IP 주소를 반환하는 정방향 DNS 검색과 대비됩니다.

예를 들어, IP 주소 `192.168.1.2`의 역방향 DNS 검색이 하위 도메인 `mail.example.com`을 반환하고, 하위 도메인 `mail.example.com`의 정방향 DNS 검색이 IP 주소 `192.168.1.2`를 반환하면 IP 주소 `192.168.1.2`에 대한 역방향 DNS는 정방향-확인(forward-confirmed)됩니다. 자세한 내용은 Wikipedia의 Forward-confirmed reverse DNS(정방향-확인 역방향 DNS)를 참조하십시오.

사전 조건을 완료한 다음 아웃바운드 메시지 할당량을 제거하도록 AWS Support에 요청을 제출하여 Amazon Lightsail 인스턴스에 대한 역방향 DNS를 구성할 수 있습니다. 이러한 단계는 다음 단원에서 다룹니다.

사전 조건

역방향 DNS를 구성하려면 표시된 순서로 다음 사전 조건을 완료하십시오.

1. 이메일 서버로 사용될 Lightsail 인스턴스를 생성합니다. 자세한 내용은 인스턴스 생성을 참조하세요.
2. 역방향 DNS 레코드에 사용할 고정 IP를 생성하여 실행 중인 인스턴스에 연결합니다. 자세한 내용은 고정 IP를 생성하여 인스턴스에 연결을 참조하세요.

> **Important**
>
> 인스턴스를 처음 생성할 때 연결되는 기본 퍼블릭 IP를 역방향 DNS에 사용할 수 없습니다. 이는 인스턴스를 중지한 다음 시작하면 인스턴스에 대한 기본 퍼블릭 IP가 변경되기 때문입니다.

3. 도메인의 DNS 영역에서 `mail.example.com`과 같이 하위 도메인을 가리키는 별칭 레코드(A 레코드)를 실행 중인 인스턴스의 고정 IP 주소에 추가합니다. 이는 고정 IP 주소의 역방향 DNS 검색이 수행될 때 반환되는 하위 도메인입니다. 자세한 내용은 DNS 영역을 생성하여 도메인의 DNS 레코드 관리를 참조하세요.

> **Note**
>
> 도메인의 DNS 레코드 관리를 Lightsail에 이전하는 것이 좋습니다. 이렇게 하면 도메인을 비롯한 모든 리소스를 한 위치(Lightsail 콘솔)에서 관리할 수 있습니다. 자세한 내용은 DNS 영역을 생성하여 도메인의 DNS 레코드 관리를 참조하세요.

4. 인터넷의 DNS를 통해 변경 사항이 전파될 때까지 기다립니다. 그런 다음 AWS Support에 요청을 제출하여 역방향 DNS를 구성하는 단계로 계속 진행할 수 있습니다.

AWS Support에 요청을 제출하여 역방향 DNS 구성

보안상의 이유로 Lightsail에서는 기본적으로 포트 25를 통해 아웃바운드 메시지를 제한합니다. 그러나 계정에서 이러한 할당량을 제거하도록 AWS Support에 요청하고 고정 IP에 대한 역방향 DNS를 구성할 수 있습니다.

AWS Support에 요청을 제출하려면

1. AWS 계정 루트 사용자로 Lightsail 콘솔에 로그인합니다.

> **Important**
>
> 요청은 AWS 계정 루트 사용자를 사용하여 제출해야 합니다. AWS 계정 루트 사용자에 대한 자세한 내용은 AWS 계정 루트 사용자 단원을 참조하십시오.

2. 이메일 전송 제한 제거 요청 양식으로 이동하여 다음 필수 정보를 입력합니다.

> **Note**
>
> 양식은 EIP(탄력적 IP), EC2 인스턴스 등의 Amazon EC2(Elastic Compute Cloud) 리소스를 참조합니다. 그러나 고정 IP, Lightsail 인스턴스 등의 Lightsail 리소스에 대한 양식을 사용할 수도 있습니다.

- 이메일 주소 - 요청에 대한 회신을 수신할 수 있는 이메일 주소를 입력합니다. 계정 이메일 주소는 이 텍스트 상자에 미리 채워져 있습니다.
- 사용 사례 설명 - 이메일 할당량 제거 요청의 사유를 입력합니다.
- 탄력적 IP 주소 - 이 안내서의 초반부에 있는 사전 조건의 2단계에서 인스턴스에 연결한 고정 IP 주소를 입력합니다. 최대 2개의 고정 IP 주소를 입력할 수 있습니다.
- EIP에 대한 역방향 DNS 레코드 - 이 안내서의 초반부에 있는 사전 조건의 3단계에서 정의한 하위 도메인을 입력합니다. 이는 역방향 DNS 검색이 수행될 때 반환되는 도메인입니다.

3. 완료되면 Submit(제출)을 선택합니다.

AWS Support에서 요청이 완료되면 고정 IP 주소가 역방향 DNS 검색으로 정방향-확인(forward-confirmed)될 수 있습니다.

나중에 Lightsail 계정에서 고정 IP 주소를 삭제하려면 역방향 DNS 구성을 제거하도록 AWS Support에 요청을 제출해야 합니다. 역방향 DNS 구성이 제거되면 Lightsail 콘솔을 사용하여 Lightsail 계정에서 고정 IP 주소를 삭제할 수 있습니다. 자세한 내용은 고정 IP 삭제를 참조하세요.

Lightsail 객체 스토리지 버킷을 사용하여 데이터 저장 및 관리

Amazon Lightsail 객체 스토리지 서비스를 사용하여 인터넷상의 어디서나 언제든지 객체를 저장하고 검색할 수 있습니다. 이는 개발자가 웹 규모 컴퓨팅을 더 쉽게 할 수 있도록 설계되었으며 Amazon Simple Storage Service(Amazon S3)를 사용하여 구축되었습니다. Lightsail 객체 스토리지를 사용하면 Amazon이 자체 글로벌 웹 사이트 네트워크를 운영하는 데 사용하는 것과 동일한 확장성이 뛰어나고 안정적이며 빠르고 저렴한 데이터 스토리지 인프라에 액세스할 수 있습니다. 이 서비스는 규모의 이점을 극대화하고 이러한 이점을 고객에게 전달하는 것을 목표로 합니다.

객체 스토리지 개념

Lightsail 객체 스토리지에 적용되는 개념과 용어는 다음과 같습니다.

버킷

버킷은 Lightsail 객체 스토리지 서비스에 저장된 객체의 컨테이너입니다. 모든 객체는 자체 URL이 있는 버킷에 저장됩니다. 예를 들어, `media/sailbot.jpg`라는 이름의 객체가 미국 동부(버지니아 북부) 지역(`us-east-1`)의 `amzn-s3-demo-bucket` 버킷에 저장된 경우 `https://amzn-s3-demo-bucket.us-east-1.amazonaws.com/media/sailbot.jpg`와 유사한 URL을 사용하여 주소를 지정할 수 있습니다.

를 사용할 수 AWS 리전 있는에서 버킷을 생성할 수 Lightsail 있습니다. 에서 사용할 수 있는 AWS 리전 Lightsail에 대한 자세한 내용은 AWS 일반 참조의 리전 및 엔드포인트를 참조하세요.

버킷 스토리지 플랜

AWS API에서 번들이라고 하는 스토리지 플랜은 버킷의 월별 비용, 스토리지 공간 및 데이터 전송 할당량을 지정합니다. 버킷을 처음 생성할 때 스토리지 플랜을 선택해야 합니다. 버킷이 가동되어 실행되고 나서 나중에 변경할 수 있습니다.

버킷의 플랜은 월별 AWS 결제 주기 내에 한 번만 변경할 수 있습니다. 버킷이 스토리지 공간 또는 데이터 전송 할당량을 지속적으로 초과하거나 버킷의 사용량이 스토리지 공간이나 데이터 전송 할당량 범위의 낮은 수준에 머무는 경우 버킷의 플랜을 변경하세요. 버킷 사용량이 예기치 않게 변동될 수 있으므로 버킷의 플랜을 매달 비용을 절감하려는 단기적인 관점이 아닌 장기적인 전략을 바탕으로 변경하는 것이 좋습니다. 앞으로 버킷에 충분한 스토리지 공간과 데이터 전송 할당량을 제공할 수 있는 스토리지 플랜을 선택합니다.

객체

객체는 버킷에 저장되는 기본 객체입니다. 버킷에 업로드하는 파일은 저장하는 동안 객체라고 합니다. 객체는 객체 데이터와 메타데이터로 구성됩니다. 데이터 부분은 Lightsail 객체 스토리지 서비스에서 볼 수 없습니다. 메타데이터는 객체를 설명하는 이름-값 페어의 집합입니다. 여기에는 마지막으로 수정한 날짜와 같은 몇 가지 기본 메타데이터 및 콘텐츠 형식과 같은 표준 HTTP 메타데이터가 포함됩니다.

객체는 키 이름 및 버전 ID를 통해 버킷 내에서 고유하게 식별됩니다.

객체 키 이름

키 이름은 버킷 내 객체의 고유한 식별자입니다. 버킷 내 모든 객체는 정확히 하나의 키를 갖습니다. 버킷, 키 및 버전 ID의 조합은 각 객체를 고유하게 식별합니다. 따라서 Lightsail 객체 스토리지를 '버킷 + 키 + 버전'과 객체 자체 사이의 기본 데이터 맵으로 생각할 수 있습니다. Lightsail 객체 스토리지 내 모든 객체는 웹 서비스 엔드포인트, 버킷 이름, 키, 그리고 선택 사항인 버전의 조합을 통해 고유하게 주소를 지정할 수 있습니다. 예를 들어, `https://amzn-s3-demo-bucket.us-east-1.amazonaws.com/media/sailbot.jpg` URL에서 `amzn-s3-demo-bucket`은 버킷의 이름이고 `media/sailbot.jpg`는 객체 키 이름입니다.

객체 버전 관리

버전 관리는 객체의 여러 변형을 동일한 버킷에 보유할 수 있는 기능입니다. 버전 관리를 활성화하면 버킷에 저장된 모든 버전의 객체를 모두 보존, 검색 및 복원할 수 있습니다. 또한, 의도치 않은 사용자 작업 및 애플리케이션 장애로부터 더 쉽게 복구할 수 있습니다.

버킷을 생성하면 버전 관리가 기본적으로 비활성화됩니다. 버전 관리를 활성화하면 저장된 버전을 수동으로 삭제할 때까지 버킷에 저장하는 모든 객체의 버전이 유지됩니다. 예를 들어, `media/sailbot.jpg` 객체를 저장한 후 나중에 동일한 객체 키 이름으로 더 큰 파일을 저장하면 더 작은 원본 객체가 이전 버전으로 유지됩니다. 더 큰 새로운 객체가 현재 버전이 됩니다. 이전 버전의 객체가 필요하지 않다고 판단되면 객체를 삭제할 수 있습니다. 객체의 현재 버전을 삭제하면 저장된 이전 버전의 객체가 모두 삭제됩니다.

저장된 객체 버전은 저장된 현재 객체 버전과 동일한 방식으로 버킷의 스토리지 공간을 사용합니다. 버전 관리를 활성화한 후에는 객체를 일시 중단하여 객체 버전을 더 이상 저장하지 않을 수 있습니다. 이렇게 하면 새 객체 버전을 업로드할 때 더 적은 버킷의 스토리지 공간을 소비합니다. 버전 관리를 일시 중단하면 저장된 객체 버전은 유지되지만, 버전 관리가 일시 중단된 동안 업로드한 새 객체 버전은 유지되지 않습니다.

버킷 및 객체 액세스

기본적으로 모든 객체 스토리지 리소스(버킷 및 객체)는 프라이빗입니다. 즉, 버킷 소유자(버킷을 생성한 Lightsail 계정만 버킷과 해당 객체에 액세스할 수 있습니다. 버킷 소유자는 필요에 따라 다른 사용자

에게 액세스 권한을 부여할 수 있습니다. 이 작업은 모든 객체 또는 개별 객체를 퍼블릭으로 설정하여 수행할 수 있으며, 이를 통해 전 세계 모든 사용자가 읽을 수 있습니다. 버킷에 Lightsail 인스턴스를 연결하거나 버킷에 대한 액세스 키를 만들어 전체 프로그램 액세스를 허용할 수도 있습니다. 마지막으로 다른 AWS 계정에 프로그래밍 방식의 읽기 전용 액세스 권한을 버킷에 부여할 수 있습니다.

AWS 리전

를 사용할 수 있는 모든에서 Lightsail 객체 스토리지 버킷 AWS 리전 을 생성할 수 Lightsail 있습니다. 지연 시간 최적화, 비용 최소화, 규정 요구 사항 준수 등 다양한 필요에 따라 리전을 선택할 수 있습니다. 에 저장된 객체는 다른 리전으로 명시적으로 전송하지 않는 한 리전을 벗어나지 AWS 리전 않습니다. 예를 들어, 미국 서부(오레곤) 리전에 저장된 객체는 해당 리전을 벗어나지 않습니다.

버킷 및 객체 관리

Lightsail 객체 스토리지는 의도적으로 단순성 및 견고성에 초점을 두는 최소한의 기능 세트를 사용하여 구축되었습니다. 버킷 및 객체를 관리하는 몇 가지 요소는 다음과 같습니다.

- 버킷 생성 - 데이터를 저장하는 버킷을 생성합니다. 버킷은 Lightsail 객체 스토리지 서비스의 기본 컨테이너입니다. 자세한 내용은 버킷 생성을 참조하세요.
- 데이터 저장 - Lightsail 콘솔, AWS Command Line Interface (AWS CLI) 및 AWS APIs. 파일 업로드에 대한 자세한 내용은 버킷으로 파일 업로드를 참조하세요.
- 데이터 다운로드 - 저장된 객체를 언제든지 다운로드할 수 있습니다. 자세한 내용은 버킷의 객체 다운로드를 참조하세요.
- 액세스 권한 부여 - 데이터를 업로드하거나 버킷에 있는 데이터를 다운로드하려는 다른 사용자(예: 소프트웨어 또는 개인)의 액세스를 허용하거나 거부합니다. 인증 메커니즘을 사용하면 데이터가 무단으로 액세스되지 않도록 보호하는 데 도움이 될 수 있습니다. 자세한 내용은 버킷 권한을 참조하세요.
- 버전 관리 - 버전 관리를 사용하면 버킷에 저장된 모든 객체의 버전 전체를 유지할 수 있습니다. 자세한 내용은 버킷의 객체 버전 사용 설정 및 사용 중지를 참조하세요.
- 사용량 모니터링 - 버킷에 저장된 객체 수와 사용 중인 스토리지 공간의 양을 모니터링합니다. 자세한 내용은 버킷 지표 확인을 참조하세요.
- 스토리지 플랜 변경 - 버킷을 과다 이용하는 경우 버킷의 크기를 늘리고, 사용률이 저조한 경우 버킷의 크기를 줄입니다. 자세한 내용은 버킷 플랜 변경을 참조하세요.
- 버킷 연결 - 웹 사이트 이미지 및 첨부 파일을 저장하려면 Lightsail 버킷을 WordPress 웹 사이트에 연결합니다. Lightsail 콘텐츠 전송 네트워크(CDN) 배포의 원본으로 버킷을 지정할 수도 있습니다. 이를 통해 버킷의 객체를 전 세계 사용자에게 신속하게 전달할 수 있습니다. 자세한 내용은 자습서:

WordPress 인스턴스에 버킷 연결 및 자습서: 콘텐츠 전송 네트워크 배포와 함께 버킷 사용을 참조하세요.

- 버킷 삭제 - 더 이상 사용하지 않는 경우 버킷을 삭제합니다. 자세한 내용은 버킷 삭제를 참조하세요.

객체 스토리지용 Lightsail 버킷 생성

파일을 클라우드에 업로드할 준비가 되면 Amazon Lightsail 객체 스토리지 서비스에 버킷을 생성합니다. Lightsail 객체 스토리지 서비스에 업로드하는 모든 파일은 Lightsail 버킷에 저장됩니다. 버킷에 대한 자세한 내용은 객체 스토리지를 참조하세요.

버킷 생성

Lightsail 버킷을 생성하려면 다음 절차를 완료하세요.

1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 스토리지를 선택합니다.
3. 버킷 생성을 선택합니다.
4. AWS 리전변경을 선택하여 버킷을 생성할 리전을 선택합니다.

   버킷과 함께 사용할 리소스 AWS 리전 와 동일한에 버킷을 생성하는 것이 좋습니다. 버킷을 생성한 후에는 리전을 변경할 수 없습니다.
5. 버킷의 스토리지 플랜을 선택합니다.

   스토리지 플랜에 따라 버킷의 월별 비용, 스토리지 공간 할당량 및 데이터 전송 할당량이 지정됩니다.

   버킷의 플랜은 월별 AWS 결제 주기 내에 한 번만 변경할 수 있습니다. 버킷이 스토리지 공간 또는 데이터 전송 할당량을 지속적으로 초과하거나 버킷의 사용량이 스토리지 공간이나 데이터 전송 할당량 범위의 낮은 수준에 머무는 경우 버킷의 플랜을 변경하세요. 자세한 내용은 버킷 플랜 변경을 참조하세요.
6. 버킷 이름을 입력합니다.

   버킷 이름에 대한 자세한 내용은 Amazon Lightsail의 버킷 이름 지정 규칙을 참조하세요.
7. 버킷 생성을 선택합니다.

   새 버킷의 관리 페이지로 리디렉션됩니다. 버킷을 사용하고 관리하는 데 필요한 추가 문서를 보려면 이 가이드의 다음 단계 섹션을 계속 진행합니다.

버킷 및 객체 관리

다음은 Lightsail 객체 스토리지 버킷을 관리하는 일반적인 단계입니다.

1. Amazon Lightsail 객체 스토리지 서비스의 객체와 버킷에 대해 알아보세요. 자세한 내용은 Amazon Lightsail의 객체 스토리지를 참조하세요.
2. Amazon Lightsail에서 버킷에 지정할 수 있는 이름에 대해 알아봅니다. 자세한 내용은 Amazon Lightsail의 버킷 이름 지정 규칙을 참조하세요.
3. 버킷을 생성하여 Lightsail 객체 스토리지 서비스를 시작합니다. 자세한 내용은 Amazon Lightsail에서 버킷 생성을 참조하세요.
4. 버킷의 보안 모범 사례와 버킷에 구성할 수 있는 액세스 권한에 대해 알아봅니다. 버킷의 모든 객체를 공개 또는 비공개로 설정하거나 개별 객체를 공개로 설정하도록 선택할 수 있습니다. 액세스 키를 생성하고, 버킷에 인스턴스를 연결하고, 다른 AWS 계정에 액세스 권한을 부여하여 버킷에 대한 액세스 권한을 부여할 수도 있습니다. 자세한 내용은 Amazon Lightsail 객체 스토리지에 대한 보안 모범 사례 및 Amazon Lightsail의 버킷 권한 이해를 참조하세요.

   버킷 액세스 권한에 대해 알아본 후 다음 가이드를 참조하여 버킷에 대한 액세스 권한을 부여합니다.

   - Amazon Lightsail의 버킷에 대한 퍼블릭 액세스 차단
   - Amazon Lightsail에서 버킷 액세스 권한 구성
   - Amazon Lightsail에서 버킷의 개별 객체에 대한 액세스 권한 구성
   - Amazon Lightsail에서 버킷의 액세스 키 생성
   - Amazon Lightsail에서 버킷에 대한 리소스 액세스 구성
   - Amazon Lightsail에서 버킷의 크로스 계정 액세스 구성
5. 버킷에 대한 액세스 로깅을 활성화하는 방법과 액세스 로그를 사용하여 버킷의 보안을 감사하는 방법에 대해 알아봅니다. 자세한 내용은 다음 안내서를 참조하세요.
   - Amazon Lightsail 객체 스토리지 서비스의 버킷에 대한 액세스 로깅
   - Amazon Lightsail 객체 스토리지 서비스에서 버킷에 대한 액세스 로그 형식
   - Amazon Lightsail 객체 스토리지 서비스에서 버킷에 대한 액세스 로깅 활성화
   - Amazon Lightsail에서 버킷에 대한 액세스 로그 사용하여 요청을 식별
6. Lightsail에서 버킷을 관리할 수 있는 기능을 사용자에게 부여하는 IAM 정책을 생성합니다. 자세한 내용은 Amazon Lightsail에서 버킷을 관리하기 위한 IAM 정책을 참조하세요.
7. 버킷의 객체에 라벨을 지정하고 식별하는 방식에 대해 알아봅니다. 자세한 내용은 Amazon Lightsail에서 객체 키 이름 이해를 참조하세요.

8. 파일을 업로드하고 버킷의 객체를 관리하는 방법에 대해 알아봅니다. 자세한 내용은 다음 안내서를 참조하세요.
   - Amazon Lightsail에서 버킷으로 파일 업로드
   - 멀티파트 업로드를 사용하여 Amazon Lightsail에서 버킷으로 파일 업로드
   - Amazon Lightsail에서 버킷의 객체 확인
   - Amazon Lightsail에서 버킷의 객체 복사 또는 이동
   - Amazon Lightsail에서 버킷의 객체 다운로드
   - Amazon Lightsail에서 버킷의 객체 필터링
   - Amazon Lightsail에서 버킷의 객체 태그 지정
   - Amazon Lightsail에서 버킷의 객체 삭제
9. 객체 버전 관리를 활성화하면 버킷에 저장된 모든 버전의 객체를 모두 보존, 검색 및 복원할 수 있습니다. 자세한 내용은 Amazon Lightsail에서 버킷의 객체 버전 사용 설정 및 사용 중지를 참조하세요.
10. 객체 버전 관리를 활성화한 후 버킷에 있는 객체의 이전 버전을 복원할 수 있습니다. 자세한 내용은 Amazon Lightsail에서 버킷의 이전 객체 버전 복원을 참조하세요.
11. 버킷 사용률을 모니터링합니다. 자세한 내용은 Amazon Lightsail에서 버킷에 대한 지표 보기를 참조하세요.
12. 버킷 사용률이 임곗값을 초과할 때 알림을 받도록 버킷 지표에 대한 경보를 구성합니다. 자세한 내용은 Amazon Lightsail에서 버킷 지표 경보 생성을 참조하세요.
13. 스토리지 및 네트워크 전송이 부족한 경우 버킷의 스토리지 플랜을 변경합니다. 자세한 내용은 Amazon Lightsail에서 버킷 플랜 변경을 참조하세요.
14. 버킷을 다른 리소스에 연결하는 방법에 대해 알아봅니다. 자세한 내용은 다음 자습서를 참조하세요.
   - 자습서: WordPress 인스턴스를 Amazon Lightsail 버킷에 연결
   - 자습서: Lightsail 콘텐츠 전송 네트워크 배포와 함께 Amazon Lightsail 버킷 사용
15. 버킷을 더 이상 사용하지 않는 경우 삭제할 수 있습니다. 자세한 내용은 Amazon Lightsail에서 버킷 삭제를 참조하세요.

Lightsail 객체 스토리지 버킷 삭제

Amazon Lightsail 객체 스토리지 서비스의 버킷을 더 이상 사용하지 않는 경우 삭제할 수 있습니다. 버킷을 삭제하면 버킷에 속한 모든 객체(저장된 객체 버전 및 액세스 키 포함)가 영구적으로 삭제됩니다.

버킷에 대한 자세한 내용은 객체 스토리지를 참조하세요.

버킷 강제 삭제

다음 조건 중 하나를 충족하는 버킷을 삭제하려면 승인이 필요합니다.

- 버킷이 배포의 오리진입니다.
- 버킷에 연결된 인스턴스가 있습니다.
- 버킷에 객체가 있습니다.
- 버킷에 액세스 키가 있습니다.

버킷을 사용하는 기존 워크플로가 중단되지 않도록 하려면 삭제를 승인해야 합니다. 버킷에 미디어를 저장하는 WordPress 웹 사이트나 버킷에서 객체를 캐싱하고 제공하는 배포를 예로 들 수 있습니다.

상기 조건 중 하나를 충족하는 버킷 삭제를 승인하려면 버킷을 강제로 삭제해야 합니다. 버킷을 삭제하기 전에 Lightsail 서비스에서 버킷이 이 중 어떤 조건을 충족하는지 묻는 메시지가 표시됩니다. Lightsail 콘솔을 사용하여 버킷을 삭제하는 경우 버킷을 강제로 삭제할 수 있는 옵션이 표시됩니다. 를 사용하는 경우 `delete-bucket` 요청 시 `--force-delete` 플래그를 지정 AWS CLI해야 합니다. 이 가이드의 Lightsail 콘솔을 사용하여 버킷 삭제 및 AWS CLI를 사용하여 버킷 삭제 섹션에 모든 절차가 나와 있습니다.

Lightsail 콘솔을 사용하여 버킷 삭제

Lightsail 콘솔을 통해 버킷을 삭제하려면 다음 절차를 완료하세요.

1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 스토리지를 선택합니다.
3. 삭제할 버킷의 이름을 선택합니다.
4. 탭 메뉴에서 줄임표(⋮) 아이콘을 선택한 다음 삭제(Delete)를 선택합니다.
5. 버킷 삭제(Delete bucket)를 선택합니다.
6. 프롬프트가 표시되면 버킷이 다음 조건 중 하나를 충족하는지 확인합니다.

    - 객체 포함
    - 액세스 키 보유
    - 인스턴스에 연결됨
    - 배포의 오리진

    이러한 조건을 갖춘 경우 버킷을 강제로 삭제하도록 선택해야 합니다.

7. 다음 옵션 중 하나를 선택하세요.

   - 이 절차의 6단계에서 나열된 조건을 하나라도 갖춘 버킷을 삭제하려면 강제 삭제(Force delete)를 선택합니다.
   - 이 절차의 6단계에서 나열된 조건을 하나도 갖추지 않은 버킷을 삭제하려면 예, 삭제합니다(Yes, delete)를 선택합니다.
   - 삭제를 취소하려면 아니요, 취소합니다(No, cancel)를 선택합니다.

를 사용하여 버킷 삭제 AWS CLI

AWS Command Line Interface ()를 사용하여 버킷을 삭제하려면 다음 절차를 완료하세요AWS CLI. `delete-bucket` 명령을 사용하여 이 작업을 수행할 수 있습니다. 자세한 내용은 AWS CLI 명령 레퍼런스의 delete-bucket을 참조하세요.

> **Note**
>
> 이 절차를 계속하기 전에 AWS CLI 및 Amazon S3에 대해 Lightsail를 설치하고 구성해야 합니다. 자세한 내용은 에서 작동 AWS CLI 하도록 구성을 Lightsail참조하세요.

1. 명령 프롬프트 또는 터미널 창을 엽니다.
2. 이중 한 곳에 다음 명령 중 하나를 입력합니다.

   - 다음 명령을 입력하여 이 가이드의 버킷 강제 삭제 섹션에 나열된 조건을 갖추지 않은 버킷을 삭제합니다.

     ```
     aws lightsail delete-bucket --bucket-name BucketName
     ```

   - 다음 명령을 입력하여 이 가이드의 버킷 강제 삭제 섹션에 나열된 조건을 갖춘 버킷을 삭제합니다.

     ```
     aws lightsail delete-bucket --bucket-name BucketName --force-delete
     ```

   명령에서 *BucketName*을 삭제하려는 버킷의 이름으로 바꿉니다.

   예제:

```
aws lightsail delete-bucket --bucket-name amzn-s3-demo-bucket
```

다음 예와 비슷한 결과가 나타나야 합니다.

```
C:\>aws lightsail delete-bucket --bucket-name DOC-EXAMPLE-BUCKET
{
    "operations": [
        {
            "id": "6example-4d30-4442-ae9a-examplef4f52",
            "resourceName": "DOC-EXAMPLE-BUCKET",
            "resourceType": "Bucket",
            "createdAt": "2021-06-30T13:42:43.873000-07:00",
            "location": {
                "availabilityZone": "all",
                "regionName": "us-west-2"
            },
            "isTerminal": true,
            "operationDetails": "62example362/DOC-EXAMPLE-BUCKET/small_1_0",
            "operationType": "DeleteBucket",
            "status": "Succeeded",
            "statusChangedAt": "2021-06-30T13:42:43.873000-07:00",
            "errorCode": "",
            "errorDetails": ""
        }
    ]
}
```

버킷 및 객체 관리

다음은 Lightsail 객체 스토리지 버킷을 관리하는 일반적인 단계입니다.

1. Amazon Lightsail 객체 스토리지 서비스의 객체와 버킷에 대해 알아보세요. 자세한 내용은 Amazon Lightsail의 객체 스토리지를 참조하세요.
2. Amazon Lightsail에서 버킷에 지정할 수 있는 이름에 대해 알아봅니다. 자세한 내용은 Amazon Lightsail의 버킷 이름 지정 규칙을 참조하세요.
3. 버킷을 생성하여 Lightsail 객체 스토리지 서비스를 시작합니다. 자세한 내용은 Amazon Lightsail에서 버킷 생성을 참조하세요.
4. 버킷의 보안 모범 사례와 버킷에 구성할 수 있는 액세스 권한에 대해 알아봅니다. 버킷의 모든 객체를 공개 또는 비공개로 설정하거나 개별 객체를 공개로 설정하도록 선택할 수 있습니다. 액세스 키를 생성하고, 버킷에 인스턴스를 연결하고, 다른 AWS 계정에 액세스 권한을 부여하여 버킷에 대한

액세스 권한을 부여할 수도 있습니다. 자세한 내용은 Amazon Lightsail 객체 스토리지에 대한 보안 모범 사례 및 Amazon Lightsail의 버킷 권한 이해를 참조하세요.

버킷 액세스 권한에 대해 알아본 후 다음 가이드를 참조하여 버킷에 대한 액세스 권한을 부여합니다.

- Amazon Lightsail의 버킷에 대한 퍼블릭 액세스 차단
- Amazon Lightsail에서 버킷 액세스 권한 구성
- Amazon Lightsail에서 버킷의 개별 객체에 대한 액세스 권한 구성
- Amazon Lightsail에서 버킷의 액세스 키 생성
- Amazon Lightsail에서 버킷에 대한 리소스 액세스 구성
- Amazon Lightsail에서 버킷의 크로스 계정 액세스 구성

5. 버킷에 대한 액세스 로깅을 활성화하는 방법과 액세스 로그를 사용하여 버킷의 보안을 감사하는 방법에 대해 알아봅니다. 자세한 내용은 다음 안내서를 참조하세요.
   - Amazon Lightsail 객체 스토리지 서비스의 버킷에 대한 액세스 로깅
   - Amazon Lightsail 객체 스토리지 서비스에서 버킷에 대한 액세스 로그 형식
   - Amazon Lightsail 객체 스토리지 서비스에서 버킷에 대한 액세스 로깅 활성화
   - Amazon Lightsail에서 버킷에 대한 액세스 로그 사용하여 요청을 식별
6. Lightsail에서 버킷을 관리할 수 있는 기능을 사용자에게 부여하는 IAM 정책을 생성합니다. 자세한 내용은 Amazon Lightsail에서 버킷을 관리하기 위한 IAM 정책을 참조하세요.
7. 버킷의 객체에 라벨을 지정하고 식별하는 방식에 대해 알아봅니다. 자세한 내용은 Amazon Lightsail에서 객체 키 이름 이해를 참조하세요.
8. 파일을 업로드하고 버킷의 객체를 관리하는 방법에 대해 알아봅니다. 자세한 내용은 다음 안내서를 참조하세요.
   - Amazon Lightsail에서 버킷으로 파일 업로드
   - 멀티파트 업로드를 사용하여 Amazon Lightsail에서 버킷으로 파일 업로드
   - Amazon Lightsail에서 버킷의 객체 확인
   - Amazon Lightsail에서 버킷의 객체 복사 또는 이동
   - Amazon Lightsail에서 버킷의 객체 다운로드
   - Amazon Lightsail에서 버킷의 객체 필터링
   - Amazon Lightsail에서 버킷의 객체 태그 지정
   - Amazon Lightsail에서 버킷의 객체 삭제

9. 객체 버전 관리를 활성화하면 버킷에 저장된 모든 버전의 객체를 모두 보존, 검색 및 복원할 수 있습니다. 자세한 내용은 Amazon Lightsail에서 버킷의 객체 버전 사용 설정 및 사용 중지를 참조하세요.
10. 객체 버전 관리를 활성화한 후 버킷에 있는 객체의 이전 버전을 복원할 수 있습니다. 자세한 내용은 Amazon Lightsail에서 버킷의 이전 객체 버전 복원을 참조하세요.
11. 버킷 사용률을 모니터링합니다. 자세한 내용은 Amazon Lightsail에서 버킷에 대한 지표 보기를 참조하세요.
12. 버킷 사용률이 임곗값을 초과할 때 알림을 받도록 버킷 지표에 대한 경보를 구성합니다. 자세한 내용은 Amazon Lightsail에서 버킷 지표 경보 생성을 참조하세요.
13. 스토리지 및 네트워크 전송이 부족한 경우 버킷의 스토리지 플랜을 변경합니다. 자세한 내용은 Amazon Lightsail에서 버킷 플랜 변경을 참조하세요.
14. 버킷을 다른 리소스에 연결하는 방법에 대해 알아봅니다. 자세한 내용은 다음 자습서를 참조하세요.
    - 자습서: WordPress 인스턴스를 Amazon Lightsail 버킷에 연결
    - 자습서: Lightsail 콘텐츠 전송 네트워크 배포와 함께 Amazon Lightsail 버킷 사용
15. 버킷을 더 이상 사용하지 않는 경우 삭제할 수 있습니다. 자세한 내용은 Amazon Lightsail에서 버킷 삭제를 참조하세요.

Lightsail 객체 스토리지 버킷 액세스 키 생성

액세스 키를 사용하여 버킷 및 해당 객체에 대한 전체 액세스 권한을 부여하는 자격 증명 집합을 생성할 수 있습니다. 액세스 키는 액세스 키 ID와 시크릿 액세스 키가 한 세트로 구성됩니다. 시크릿 액세스 키는 암호 키를 생성할 때만 표시됩니다. 소프트웨어 또는 플러그인에서 액세스 키를 구성하면 AWS APIs 및 AWS SDKs. AWS CLI에서 액세스 키를 구성할 수도 있습니다.

> **Important**
>
> 버킷당 두 개의 액세스 키를 가질 수 있지만 한 번에 하나의 버킷 액세스 키만 생성할 것이 권장됩니다. 또한 주기적으로 키를 교체하고 기존 키의 인벤토리를 확인할 것이 권장됩니다. 시크릿 액세스 키가 복사, 손실 또는 유출된 경우 액세스 키를 삭제하고 새로 생성해야 합니다. 버킷 액세스 키 교체에 대한 모범 사례는 버킷 액세스 키 교체 섹션을 참조하세요.

권한 옵션에 대한 자세한 내용은 버킷 권한을 참조하세요. 버킷에 대한 자세한 내용은 객체 스토리지를 참조하세요.

버킷용 액세스 키 생성

버킷용 액세스 키를 생성하려면 다음 절차를 완료하세요.

1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 스토리지를 선택합니다.
3. 액세스 권한을 구성할 버킷 이름을 선택합니다.
4. 권한 탭을 선택합니다.

   페이지의 액세스 키(Access keys) 섹션에는 버킷의 기존 액세스 키가 표시됩니다(있는 경우).
5. 액세스 키 생성(Create access key)을 선택하여 버킷에 대한 새 액세스 키를 생성합니다.
6. 프롬프트가 표시되면 예, 생성합니다(Yes, create)를 선택하여 새 액세스 키를 생성할지 확인합니다. 생성하지 않으려면 아니요, 취소합니다(No, cancel)를 선택합니다.
7. 액세스 키를 생성했다는 프롬프트가 표시되면 액세스 키 ID를 기록해 둡니다.
8. 비밀 액세스 키 표시(Show secret access key)를 선택하여 비밀 액세스 키를 확인하고 기록해 둡니다. 비밀 액세스 키는 다시 표시되지 않습니다.

   > **Important**
   >
   > 비밀 액세스 키 ID와 비밀 액세스 키를 안전한 위치에 저장합니다. 유출된 경우 삭제하고 새 키를 생성해야 합니다. 자세한 내용은 Lightsail 객체 스토리지 버킷에 대한 액세스 키 삭제 단원을 참조하십시오.
9. 계속(Continue)을 선택하여 완료합니다.

   새 액세스 키는 페이지의 액세스 키(Access keys) 섹션에 나열됩니다. 액세스 키가 유출되거나 분실된 경우 키를 삭제하고 새 키를 생성합니다.

   > **Note**
   >
   > 각 액세스 키 옆에 표시되는 마지막 사용 열은 키가 마지막으로 사용된 시간을 나타냅니다. 사용된 적이 없는 키 옆에는 대시가 표시됩니다. 액세스 키 노드를 확장하여 서비스와 키가 마지막으로 사용된 AWS 리전 위치를 확인합니다.

Lightsail 객체 스토리지 버킷에 대한 액세스 키 삭제

액세스 키는 버킷 및 해당 객체에 대한 전체 액세스 권한을 부여하는 자격 증명 집합입니다. 액세스 키는 액세스 키 ID와 시크릿 액세스 키가 한 세트로 구성됩니다. 시크릿 액세스 키가 복사, 손실 또는 유출된 경우 액세스 키를 삭제해야 합니다.

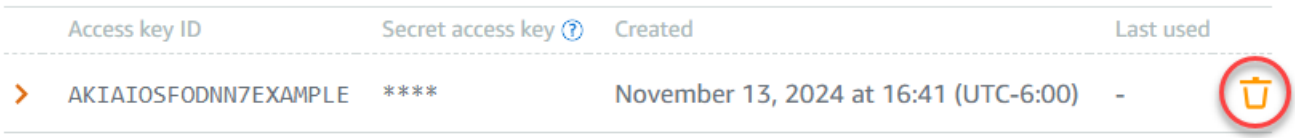
버킷에 대한 액세스 키 삭제

다음 절차를 사용하여 버킷 액세스 키를 삭제할 수 있습니다.

> **Warning**
> 액세스 키를 삭제하면 키가 영구 삭제되어 복원할 수 없습니다. 이 키는 새 액세스 키로만 교체할 수 있습니다.

기존 Lightsail 객체 스토리지 버킷 액세스 키를 삭제하려면

1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 스토리지를 선택합니다.
3. 액세스 키를 삭제할 버킷의 이름을 선택합니다.
4. 권한 탭을 선택합니다.
5. 액세스 키 아래에서 삭제하려는 액세스 키에 대한 제거 아이콘을 선택합니다.



6. 액세스 키 삭제를 계속하려면 예, 삭제를 선택합니다.

기존 키가 삭제되면 새 액세스 키를 생성하고 소프트웨어 또는 플러그인에 맞게 구성할 수 있습니다. 자세한 내용은 버킷 액세스 키 교체 단원을 참조하십시오.

Lightsail 버킷 및 객체에 대한 퍼블릭 액세스 제한

Amazon Simple Storage Service(S3)는 업계 최고의 확장성, 데이터 가용성, 보안 및 성능을 제공하는 객체 스토리지 서비스입니다. Amazon Lightsail 객체 스토리지 서비스는 Amazon S3 기술을 기반으로

구축됩니다. Amazon S3는 AWS 계정의 모든 S3 버킷에 대한 퍼블릭 액세스를 제한하는 데 사용할 수 있는 계정 수준 퍼블릭 액세스 차단 기능을 제공합니다. 계정 수준 퍼블릭 액세스 차단은 기존 개별 버킷 및 객체 권한에 관계없이 모든 S3 버킷을 AWS 계정 프라이빗으로 만들 수 있습니다.

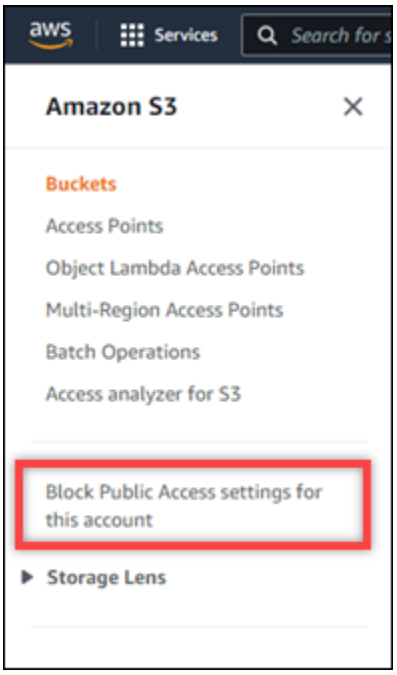
퍼블릭 액세스를 허용하거나 거부할 때 Lightsail 객체 스토리지 버킷에서 고려되는 사항은 다음과 같습니다.

- Lightsail 버킷 액세스 권한. 자세한 내용은 버킷 권한을 참조하세요.
- Amazon S3 계정 수준 퍼블릭 액세스 차단 구성(Lightsail 버킷 액세스 권한을 재정의함).

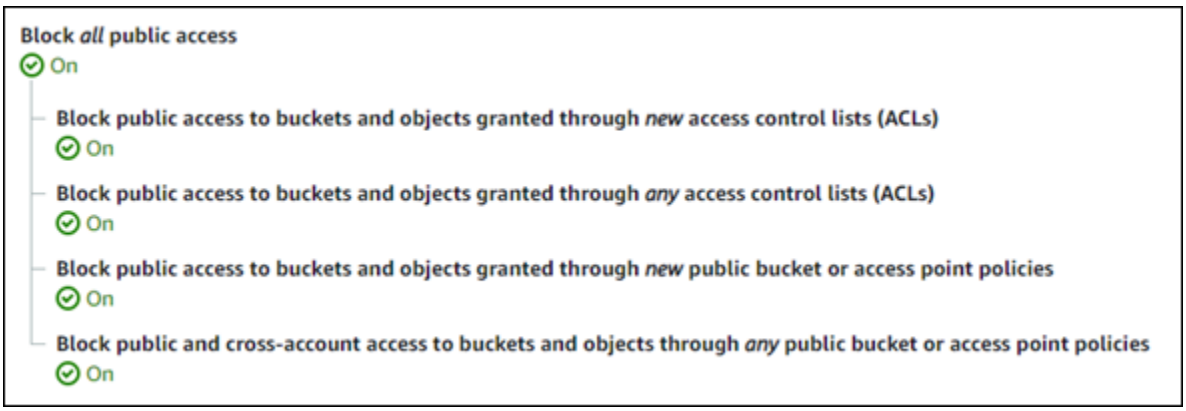
Amazon S3에서 계정 수준 모든 퍼블릭 액세스 차단을 켜면 퍼블릭 Lightsail 버킷과 객체가 프라이빗 상태가 되어 더 이상 공개적으로 액세스할 수 없게 됩니다.

계정에 대한 퍼블릭 액세스 차단 설정 구성

Amazon S3 콘솔, AWS Command Line Interface (AWS CLI), AWS SDKs 및 REST API를 사용하여 퍼블릭 액세스 차단 설정을 구성할 수 있습니다. 다음 예에서와 같이 Amazon S3 콘솔의 탐색 창에서 계정 수준 퍼블릭 액세스 차단 기능에 액세스할 수 있습니다.



Amazon S3 콘솔은 모든 퍼블릭 액세스를 차단하고, 신규 또는 임의의 액세스 제어 목록을 통해 부여된 퍼블릭 액세스를 차단하며, 신규 또는 모든 퍼블릭 버킷이나 액세스 포인트 정책을 통해 부여된 버킷 및 객체에 대한 퍼블릭 액세스를 차단하는 설정을 제공합니다.



Amazon S3 콘솔에서 각 설정을 켜기 또는 끄기로 설정할 수 있습니다. API에서는 TRUE(켜기) 또는 FALSE(끄기)에 해당하는 설정입니다. 다음 섹션에서는 S3 버킷 및 Lightsail 버킷에서의 각 설정의 영향을 설명합니다.

> **Note**
>
> 다음 섹션에서는 액세스 제어 목록(ACL)에 대해 설명합니다. ACL은 버킷 또는 개별 객체를 소유하거나 액세스할 수 있는 사용자를 정의합니다. 자세한 내용은 Amazon S3 사용 설명서의 액세스 제어 목록(ACL) 개요를 참조하세요.

- 모든 퍼블릭 액세스 차단 - S3 버킷, Lightsail 버킷 및 그에 해당하는 객체에 대한 모든 퍼블릭 액세스를 차단하려면 이 설정을 켭니다. 이 설정에는 다음 설정이 모두 통합되어 있습니다. 이 설정을 켜면 사용자(버킷 소유자)와 권한 있는 사용자만 버킷과 해당 객체에 액세스할 수 있습니다. 이 설정은 Amazon S3 콘솔에서만 켤 수 있습니다. AWS CLI, Amazon S3 API 또는 AWS SDKs에서는 사용할 수 없습니다.
  - 새로운 액세스 제어 목록(ACL)을 통해 권한이 부여된 버킷 및 객체에 대한 퍼블릭 액세스 차단 - 버킷과 객체에 퍼블릭 ACL을 넣는 것을 차단하려면 이 설정을 켭니다. 이 설정은 기존 ACL에는 영향을 주지 않습니다. 따라서 이미 퍼블릭 ACL이 있는 객체는 공개 상태로 유지됩니다. 또한 이 설정은 버킷 액세스 권한이 모든 객체는 퍼블릭이며 읽기 전용임(All objects are public and read-only)으로 설정되어 있기 때문에 퍼블릭 객체에도 영향을 주지 않습니다. 이 설정은 Amazon S3 API에서 `BlockPublicAcls`로 레이블이 지정되었습니다.

    > **Note**
    >
    > 이 설정을 켜면 Lightsail 버킷에 미디어를 삽입하는 Offload Media Lite 등의 WordPress 플러그 인은 작동이 중지될 수 있습니다. 이는 대부분의 WordPress 플러그 인이 객체에 대

한 퍼블릭 읽기 ACL을 구성하기 때문입니다. 객체 ACL을 토글하는 WordPress 플러그 인도 작동을 멈출 수 있습니다.

- 모든 액세스 제어 목록(ACL)을 통해 권한이 부여된 버킷 및 객체에 대한 퍼블릭 액세스 차단 - 퍼블릭 ACL을 무시하고 버킷과 객체에 대한 퍼블릭 액세스를 차단하려면 이 설정을 켭니다. 이 설정을 사용하면 버킷과 객체에 퍼블릭 ACL을 넣을 수 있지만 액세스 권한을 부여할 때는 이를 무시합니다. Lightsail 버킷의 경우 버킷의 액세스 권한 설정을 모든 객체는 퍼블릭이며 읽기 전용임(All objects are public and read-only)으로 설정하거나 개별 객체의 권한을 퍼블릭(읽기 전용)(Public (read-only))으로 설정하는 것은 둘 중 하나에 퍼블릭 ACL을 설정하는 것과 결과가 같습니다. 이 설정은 Amazon S3 API에서 `IgnorePublicAcls`로 레이블이 지정되었습니다.
- 새로운 퍼블릭 버킷 또는 액세스 포인트 정책을 통해 권한이 부여된 버킷 및 객체에 대한 퍼블릭 액세스 차단 - Lightsail 버킷에서 모든 객체는 퍼블릭이며 읽기 전용임 버킷 액세스 권한이 구성되는 것을 차단하려면 이 설정을 켭니다. 이 설정은 이미 모든 객체는 퍼블릭이며 읽기 전용임(All objects are public and read-only) 버킷 액세스 권한으로 구성된 버킷에는 영향을 주지 않습니다. 이 설정은 Amazon S3 API에서 `BlockPublicPolicy`로 레이블이 지정되었습니다.
- 모든 퍼블릭 버킷 또는 액세스 포인트 정책을 통해 권한이 부여된 버킷 및 객체에 대한 퍼블릭 및 크로스 계정 액세스 차단 - Lightsail 버킷을 모두 프라이빗 상태로 만들려면 이 설정을 켭니다. 이 설정은 모든 객체는 퍼블릭이며 읽기 전용임(All objects are public and read-only) 버킷 액세스 권한으로 구성된 버킷을 포함하여 모든 Lightsail 버킷을 프라이빗으로 만듭니다. 이 설정은 Amazon S3 API에서 `RestrictPublicBuckets`로 레이블이 지정되었습니다.

> **Important**
>
> 이 설정은 Lightsail에 모든 객체는 퍼블릭이며 읽기 전용임(All objects are public and read-only) 버킷 액세스 권한으로 구성된 Lightsail 버킷에 대한 크로스 계정 액세스도 차단합니다. 크로스 계정 액세스를 계속 허용하려면 Amazon S3의 모든 퍼블릭 버킷 또는 액세스 포인트 정책을 통해 권한이 부여된 버킷 및 객체에 대한 퍼블릭 및 크로스 계정 액세스 차단 설정을 켜기 전에 Lightsail에서 Lightsail 버킷을 모든 객체는 프라이빗임 버킷 액세스 권한으로 구성해야 합니다.

퍼블릭 액세스 차단과 이를 구성하는 방법에 대한 자세한 내용은 Amazon S3 사용 설명서에서 다음과 같은 리소스를 참조하세요.

- Amazon S3 스토리지에 대한 퍼블릭 액세스 차단
- 계정에 대한 퍼블릭 액세스 차단 설정 구성

Lightsail 콘솔, AWS CLI AWS SDKs 및 REST API를 사용하여 Lightsail 버킷에 대한 액세스 권한을 구성합니다. 자세한 내용은 버킷 권한을 참조하세요.

> **Note**
>
> Lightsail에서는 서비스 연결 역할(SLR)을 사용하여 현재 계정 수준 퍼블릭 액세스 차단 구성을 Amazon S3로부터 가져와 Lightsail 객체 스토리지 리소스에 적용합니다. Amazon S3에서 퍼블릭 액세스 차단을 구성한 후 Lightsail에 적용되려면 1시간 이상 기다려야 합니다. 자세한 내용은 서비스 연결 역할을 참조하세요.

버킷 및 객체 관리

다음은 Lightsail 객체 스토리지 버킷을 관리하는 일반적인 단계입니다.

1. Amazon Lightsail 객체 스토리지 서비스의 객체와 버킷에 대해 알아보세요. 자세한 내용은 Amazon Lightsail의 객체 스토리지를 참조하세요.
2. Amazon Lightsail에서 버킷에 지정할 수 있는 이름에 대해 알아봅니다. 자세한 내용은 Amazon Lightsail의 버킷 이름 지정 규칙을 참조하세요.
3. 버킷을 생성하여 Lightsail 객체 스토리지 서비스를 시작합니다. 자세한 내용은 Amazon Lightsail에서 버킷 생성을 참조하세요.
4. 버킷의 보안 모범 사례와 버킷에 구성할 수 있는 액세스 권한에 대해 알아봅니다. 버킷의 모든 객체를 공개 또는 비공개로 설정하거나 개별 객체를 공개로 설정하도록 선택할 수 있습니다. 액세스 키를 생성하고, 버킷에 인스턴스를 연결하고, 다른 AWS 계정에 액세스 권한을 부여하여 버킷에 대한 액세스 권한을 부여할 수도 있습니다. 자세한 내용은 Amazon Lightsail 객체 스토리지에 대한 보안 모범 사례 및 Amazon Lightsail의 버킷 권한 이해를 참조하세요.

   버킷 액세스 권한에 대해 알아본 후 다음 가이드를 참조하여 버킷에 대한 액세스 권한을 부여합니다.

   - Amazon Lightsail의 버킷에 대한 퍼블릭 액세스 차단
   - Amazon Lightsail에서 버킷 액세스 권한 구성
   - Amazon Lightsail에서 버킷의 개별 객체에 대한 액세스 권한 구성
   - Amazon Lightsail에서 버킷의 액세스 키 생성
   - Amazon Lightsail에서 버킷에 대한 리소스 액세스 구성
   - Amazon Lightsail에서 버킷의 크로스 계정 액세스 구성

5. 버킷에 대한 액세스 로깅을 활성화하는 방법과 액세스 로그를 사용하여 버킷의 보안을 감사하는 방법에 대해 알아봅니다. 자세한 내용은 다음 안내서를 참조하세요.
   - Amazon Lightsail 객체 스토리지 서비스의 버킷에 대한 액세스 로깅
   - Amazon Lightsail 객체 스토리지 서비스에서 버킷에 대한 액세스 로그 형식
   - Amazon Lightsail 객체 스토리지 서비스에서 버킷에 대한 액세스 로깅 활성화
   - Amazon Lightsail에서 버킷에 대한 액세스 로그 사용하여 요청을 식별
6. Lightsail에서 버킷을 관리할 수 있는 기능을 사용자에게 부여하는 IAM 정책을 생성합니다. 자세한 내용은 Amazon Lightsail에서 버킷을 관리하기 위한 IAM 정책을 참조하세요.
7. 버킷의 객체에 라벨을 지정하고 식별하는 방식에 대해 알아봅니다. 자세한 내용은 Amazon Lightsail에서 객체 키 이름 이해를 참조하세요.
8. 파일을 업로드하고 버킷의 객체를 관리하는 방법에 대해 알아봅니다. 자세한 내용은 다음 안내서를 참조하세요.
   - Amazon Lightsail에서 버킷으로 파일 업로드
   - 멀티파트 업로드를 사용하여 Amazon Lightsail에서 버킷으로 파일 업로드
   - Amazon Lightsail에서 버킷의 객체 확인
   - Amazon Lightsail에서 버킷의 객체 복사 또는 이동
   - Amazon Lightsail에서 버킷의 객체 다운로드
   - Amazon Lightsail에서 버킷의 객체 필터링
   - Amazon Lightsail에서 버킷의 객체 태그 지정
   - Amazon Lightsail에서 버킷의 객체 삭제
9. 객체 버전 관리를 활성화하면 버킷에 저장된 모든 버전의 객체를 모두 보존, 검색 및 복원할 수 있습니다. 자세한 내용은 Amazon Lightsail에서 버킷의 객체 버전 사용 설정 및 사용 중지를 참조하세요.
10. 객체 버전 관리를 활성화한 후 버킷에 있는 객체의 이전 버전을 복원할 수 있습니다. 자세한 내용은 Amazon Lightsail에서 버킷의 이전 객체 버전 복원을 참조하세요.
11. 버킷 사용률을 모니터링합니다. 자세한 내용은 Amazon Lightsail에서 버킷에 대한 지표 보기를 참조하세요.
12. 버킷 사용률이 임곗값을 초과할 때 알림을 받도록 버킷 지표에 대한 경보를 구성합니다. 자세한 내용은 Amazon Lightsail에서 버킷 지표 경보 생성을 참조하세요.
13. 스토리지 및 네트워크 전송이 부족한 경우 버킷의 스토리지 플랜을 변경합니다. 자세한 내용은 Amazon Lightsail에서 버킷 플랜 변경을 참조하세요.
14. 버킷을 다른 리소스에 연결하는 방법에 대해 알아봅니다. 자세한 내용은 다음 자습서를 참조하세요.
   - 자습서: WordPress 인스턴스를 Amazon Lightsail 버킷에 연결

- 자습서: Lightsail 콘텐츠 전송 네트워크 배포와 함께 Amazon Lightsail 버킷 사용

15.버킷을 더 이상 사용하지 않는 경우 삭제할 수 있습니다. 자세한 내용은 Amazon Lightsail에서 버킷 삭제를 참조하세요.

액세스 로그로 객체 스토리지 버킷 요청 추적

액세스 로깅은 Amazon Lightsail 객체 스토리지 서비스의 버킷에 대한 요청에 대한 자세한 기록을 제공합니다. 이 정보에는 요청 유형, 요청에 지정된 리소스, 요청이 처리된 시간과 날짜가 포함될 수 있습니다. 액세스 로그는 많은 애플리케이션에 유용합니다. 예를 들어 액세스 로그 정보는 보안 및 액세스 감사에 유용할 수 있습니다. 또한 고객 기반에 대해 알아보는 데 도움이 될 수 있습니다.

목차

- 로그 전송을 활성화하려면 필요한 사항
- 로그 객체 키 형식
- 로그 전송 방법
- 최선의 액세스 로그 전송
- 버킷 로깅 상태 변경 시 일정 기간에 걸쳐 단계적으로 반영됨

로그 전송을 활성화하려면 필요한 사항이 무엇입니까?

로그 전송을 활성화하기 전에 다음 사항을 고려하십시오. 자세한 내용은 버킷 액세스 로깅 활성화를 참조하세요.

1. 로그에 대한 대상 버킷을 식별합니다. 이 버킷은 Lightsail가 액세스 로그를 객체로 저장하려는 위치입니다. 원본 및 대상 버킷 모두 동일한 AWS 리전에 있어야 하며 동일한 계정에서 소유하고 있어야 합니다.

   소스 버킷 자체를 포함하여 소스 버킷과 동일한 리전에 있는 자신의 고유 버킷에 로그를 전달할 수 있습니다. 그러나 로그 관리를 간소화하기 위해서는 액세스 로그를 다른 버킷에 저장하는 것이 좋습니다.

   소스 버킷과 대상 버킷이 동일한 버킷이면 버킷에 작성되는 로그에 대해 추가 로그가 생성됩니다. 이러한 방식은 스토리지 사용이 약간 증가할 수 있으므로 이상적이지 않을 수 있습니다. 또한 로그에 대한 추가 로그로 인해 원하는 로그를 찾기가 힘들어질 수 있습니다. 소스 버킷에 액세스 로그를 저장하도록 선택한 경우 객체 이름이 공통 문자열로 시작하고 로그 객체를 더 쉽게 식별할 수 있도

록 로그 객체 키에 대한 접두사를 지정하는 것이 좋습니다. 키 접두사는 여러 버킷이 동일한 대상 버킷에 로깅할 때 원본 버킷을 서로 구별하는 데 유용합니다.

2. (선택 사항) 로그 객체 키에 대한 접두사를 식별합니다. 접두사를 사용하면 더 쉽게 로그 객체를 구분할 수 있습니다. 예를 들어 접두사 값을 `logs/`로 지정할 경우 Lightsail가 만드는 각 로그 객체의 키는 `logs/` 접두사로 시작합니다. 후행 슬래시 /는 접두사의 끝을 나타내는 데 필요합니다. 다음은 `logs/` 접두사가 있는 로그 객체 키의 예입니다.

```
logs/2021-11-31-21-32-16-E568B2907131C0C0
```

로그 객체 키 형식

Lightsail는 대상 버킷에 업로드한 로그 객체에 대해 다음과 같은 객체 키 형식을 사용합니다.

```
TargetPrefix/YYYY-mm-DD-HH-MM-SS-UniqueString
```

키에서 YYYY, mm, DD, HH, MM 및 SS는 각각 로그 파일이 전송된 연도와 월, 일, 시, 분, 초에 대한 숫자를 나타냅니다. 이러한 날짜 및 시간은 협정 세계시(UTC)로 표시됩니다.

로그 파일에는 해당 파일이 전송된 시간 이전에 기록된 레코드가 포함될 수 있습니다. 특정 기간의 모든 로그 레코드가 전송되었는지 여부는 확인할 수 없습니다.

키의 UniqueString 구성 요소는 파일의 덮어쓰기를 방지할 목적으로 사용되는 것으로, 특별한 의미가 없으므로 로그 처리 소프트웨어가 무시해도 됩니다.

로그 전송 방법

Lightsail는 주기적으로 액세스 로그 레코드를 수집하여 로그 파일에 레코드를 통합한 다음 대상 버킷의 로그 객체로 로그 파일을 업로드합니다. 동일한 대상 버킷에 전달하는 여러 소스 버킷에 대한 로깅을 활성화하면 대상 버킷은 해당 모든 소스 버킷에 대한 액세스 로그를 갖게 됩니다. 그러나 각 로그 객체는 특정 원본 버킷의 액세스 로그 레코드만 보고합니다.

최선의 액세스 로그 전송

액세스 로그 레코드는 최대한 전송하겠지만 항상 모든 레코드가 전송된다고 보장할 수는 없습니다. 버킷에 대해 적절히 로깅이 구성된 대부분의 요청은 로그 레코드가 전송됩니다. 대부분 기록된 지 몇 시간 내로 로그 레코드가 전송되지만 더 자주 전송될 수 있습니다.

모든 액세스 로깅이 제때 전송될 것이라고 보장할 수는 없습니다. 특정 요청에 대한 로그 레코드는 요청이 실제로 처리된 후에 오랫동안 전송되거나 전혀 전송되지 않을 수도 있습니다. 액세스 로그의 목적

은 버킷에 대한 트래픽의 특성에 대한 아이디어를 제공하는 것입니다. 실제로 로그 레코드가 누락되는 경우는 매우 드물지만 액세스 로깅 자체가 모든 요청을 완벽하게 기록할 목적으로 제공되는 것이 아닙니다.

버킷 로깅 상태 변경 시 일정 기간에 걸쳐 단계적으로 반영됨

버킷의 로깅 상태를 변경한 후 실제 로그 파일의 전송에 반영되려면 어느 정도 시간이 지나야 합니다. 예를 들어, 버킷에 대해 로깅을 사용 설정할 경우 이후 1시간 동안 이루어진 요청 중 일부는 기록되지만 일부는 기록되지 않을 수도 있습니다. 버킷 A에서 버킷 B로 로깅의 대상 버킷을 변경할 경우 이후 1시간 동안 일부 로그는 버킷 A로 계속 전송될 수 있지만, 다른 로그는 새로운 대상 버킷 B로 전송될 수 있습니다. 그러나 추가 작업을 수행하지 않아도 어느 정도 기간이 지나면 새 설정에 따라 모든 로그가 전송됩니다.

주제

- Lightsail 버킷 로그를 사용하여 객체 스토리지 액세스 분석
- Lightsail에서 버킷 액세스 로깅 활성화
- Lightsail에서 Amazon Athena를 사용하여 버킷 액세스 로그 분석

Lightsail 버킷 로그를 사용하여 객체 스토리지 액세스 분석

액세스 로깅은 Amazon Lightsail 객체 스토리지 서비스의 버킷에 대한 요청에 대한 자세한 기록을 제공합니다. 보안 및 액세스 감사를 위해 액세스 로그를 사용하거나 고객 기반에 대해 알아볼 수 있습니다. 이 섹션에서는 액세스 로그 파일에 대한 형식과 기타 세부 정보를 설명합니다. 로깅 기초 사항에 대한 자세한 내용은 버킷 액세스 로그를 참조하세요.

액세스 로그 파일은 줄 바꿈으로 구분되는 로그 레코드의 시퀀스로 구성됩니다. 각 로그 레코드는 하나의 요청을 표시하며 공백으로 구분된 필드로 구성됩니다.

다음은 5개 로그 레코드로 구성된 로그 예제입니다.

```
79a59df900b949e55d96a1e698fbacedfd6e09d98eacf8f8d5218e7cd47ef2be
 amzn-s3-demo-bucket [06/Feb/2019:00:00:38 +0000] 192.0.2.3
 79a59df900b949e55d96a1e698fbacedfd6e09d98eacf8f8d5218e7cd47ef2be 3E57427F3EXAMPLE
 REST.GET.VERSIONING - "GET /amzn-s3-demo-bucket?versioning HTTP/1.1" 200 - 113 - 7 -
 "-" "S3Console/0.4" - s9lzHYrFp76ZVxRcpX9+5cjAnEH2ROuNkd2BHfIa6UkFVdtjf5mKR3/eTPFvsiP/
XV/VLi31234= SigV2 ECDHE-RSA-AES128-GCM-SHA256 AuthHeader amzn-s3-demo-bucket.s3.us-
west-1.amazonaws.com TLSV1.1
```

```
79a59df900b949e55d96a1e698fbacedfd6e09d98eacf8f8d5218e7cd47ef2be
 amzn-s3-demo-bucket [06/Feb/2019:00:00:38 +0000] 192.0.2.3
 79a59df900b949e55d96a1e698fbacedfd6e09d98eacf8f8d5218e7cd47ef2be 891CE47D2EXAMPLE
 REST.GET.LOGGING_STATUS - "GET /amzn-s3-demo-bucket?logging HTTP/1.1" 200 -
 242 - 11 - "-" "S3Console/0.4" - 9vKBE6vMhrNiWHZmb2L0mXOcqPGzQOI5XLnCtZNPxev+Hf
+7tpT6sxDwDty4LHBUOZJG96N1234= SigV2 ECDHE-RSA-AES128-GCM-SHA256 AuthHeader amzn-s3-
demo-bucket.s3.us-west-1.amazonaws.com TLSV1.1
```

```
79a59df900b949e55d96a1e698fbacedfd6e09d98eacf8f8d5218e7cd47ef2be
 amzn-s3-demo-bucket [06/Feb/2019:00:00:38 +0000] 192.0.2.3
 79a59df900b949e55d96a1e698fbacedfd6e09d98eacf8f8d5218e7cd47ef2be A1206F460EXAMPLE
 REST.GET.BUCKETPOLICY - "GET /amzn-s3-demo-bucket?policy HTTP/1.1" 404
 NoSuchBucketPolicy 297 - 38 - "-" "S3Console/0.4" - BNaBsXZQQDbssi6xMBdBU2sLt
+Yf5kZDmeBUP35sFoKa3sLLeMC78iwEIWxs99CRUrbS4n11234= SigV2 ECDHE-RSA-AES128-GCM-SHA256
 AuthHeader amzn-s3-demo-bucket.s3.us-west-1.amazonaws.com TLSV1.1
```

```
79a59df900b949e55d96a1e698fbacedfd6e09d98eacf8f8d5218e7cd47ef2be
 amzn-s3-demo-bucket [06/Feb/2019:00:01:00 +0000] 192.0.2.3
 79a59df900b949e55d96a1e698fbacedfd6e09d98eacf8f8d5218e7cd47ef2be 7B4A0FABBEXAMPLE
 REST.GET.VERSIONING - "GET /amzn-s3-demo-bucket?versioning HTTP/1.1" 200 -
 113 - 33 - "-" "S3Console/0.4" - Ke1bUcazaN1jWuUlPJaxF64cQVpUEhoZKEG/hmy/gijN/
I1DeWqDfFvnpybfEseEME/u7ME1234= SigV2 ECDHE-RSA-AES128-GCM-SHA256 AuthHeader amzn-s3-
demo-bucket.s3.us-west-1.amazonaws.com TLSV1.1
```

```
79a59df900b949e55d96a1e698fbacedfd6e09d98eacf8f8d5218e7cd47ef2be
 amzn-s3-demo-bucket [06/Feb/2019:00:01:57 +0000] 192.0.2.3
 79a59df900b949e55d96a1e698fbacedfd6e09d98eacf8f8d5218e7cd47ef2be
 DD6CC733AEXAMPLE REST.PUT.OBJECT s3-dg.pdf "PUT /amzn-s3-demo-bucket/
s3-dg.pdf HTTP/1.1" 200 - - 4406583 41754 28 "-" "S3Console/0.4" -
 10S62Zv81kBW7BB6SX4XJ48o6kpcl6LPwEoizZQQxJd5qDSCTLX0TgS37kYUBKQW3+bPdrg1234= SigV4
 ECDHE-RSA-AES128-SHA AuthHeader amzn-s3-demo-bucket.s3.us-west-1.amazonaws.com TLSV1.1
```

> **Note**
>
> 모든 로그 레코드 필드를 –(대시)로 설정하여 데이터를 알 수 없거나 사용할 수 없거나 해당 필드가 요청에 적용될 수 없음을 나타낼 수 있습니다.

목차

- 로그 레코드 필드
- 복사 작업을 위한 추가 로깅
- 사용자 지정 액세스 로그 정보
- 확장 가능한 액세스 로그 형식에 대한 프로그래밍 고려 사항

로그 레코드 필드

다음 목록에서는 로그 레코드 필드에 대해 설명합니다.

액세스 포인트 ARN(Amazon 리소스 이름)

요청 액세스 포인트의 Amazon 리소스 이름(ARN) 입니다. 액세스 포인트 ARN의 형식이 잘못되었거나 이를 사용하지 않으면 필드에 '-'가 포함됩니다. 액세스 포인트에 대한 자세한 내용은 액세스 포인트 사용을 참조하십시오. ARN에 대한 자세한 내용은 AWS 일반 참조의 Amazon 리소스 이름(ARN)에 대한 주제를 참조하십시오.

입력 예

```
arn:aws:s3:us-east-1:123456789012:accesspoint/example-AP
```

버킷 소유자

원본 버킷의 정식 사용자 ID입니다. 정식 사용자 ID는 또 다른 형식의 AWS 계정 ID입니다. 정식 사용자 ID에 대한 자세한 내용은 AWS 일반 참조에서 AWS 계정 식별자를 참조하세요. 계정의 정식 사용자 ID를 찾는 방법에 대한 자세한 내용은 AWS 계정의 정식 사용자 ID 찾기를 참조하세요.

입력 예

```
79a59df900b949e55d96a1e698fbacedfd6e09d98eacf8f8d5218e7cd47ef2be
```

버킷

요청이 처리된 버킷의 이름. 시스템이 잘못된 양식의 요청을 수신하여 버킷을 결정할 수 없을 경우 해당 요청이 어떤 액세스 로그에도 표시되지 않습니다.

입력 예

```
amzn-s3-demo-bucket
```

Time

요청이 수신된 시간입니다. 이 날짜 및 시간은 협정 세계시(UTC)로 표시됩니다. *strftime()* 용어를 사용하는 형식은 *[%d/%b/%Y:%H:%M:%S %z]*와 같습니다.

입력 예

```
[06/Feb/2019:00:00:38 +0000]
```

원격 IP

요청자의 명백한 인터넷 주소입니다. 중간 프록시 및 방화벽이 요청 시스템의 실제 주소를 가릴 수 있습니다.

입력 예

```
192.0.2.3
```

요청자

요청자의 정식 사용자 ID 또는 인증되지 않은 요청의 -입니다. 요청자가 IAM 사용자인 경우 이 필드는 IAM 사용자가 속한 AWS 루트 계정과 함께 요청자의 IAM 사용자 이름을 반환합니다. 이 식별자는 액세스 제어 목적으로 사용되는 것과 동일합니다.

입력 예

```
79a59df900b949e55d96a1e698fbacedfd6e09d98eacf8f8d5218e7cd47ef2be
```

요청 ID

각 요청을 고유하게 식별하기 위해 Lightsail에서 생성한 문자열입니다.

입력 예

```
3E57427F33A59F07
```

작업

여기에 나열된 작업은 SOAP.*operation*, REST.*HTTP_method.resource_type*, WEBSITE.*HTTP_method.resource_type* 또는 BATCH.DELETE.OBJECT로 선언됩니다.

입력 예

```
REST.PUT.OBJECT
```

Key(키)

요청의 "키" 부분. URL로 인코딩되거나 작업에 키 파라미터가 없을 경우 "-".

입력 예

```
/photos/2019/08/puppy.jpg
```

요청-URII

HTTP 요청 메시지의 Request-URI 부분.

입력 예

```
"GET /amzn-s3-demo-bucket/photos/2019/08/puppy.jpg?x-foo=bar HTTP/1.1"
```

HTTP 상태

응답의 숫자 HTTP 상태 코드.

입력 예

```
200
```

오류 코드

Amazon S3 오류 코드 또는 오류가 발생하지 않은 경우 "-"입니다.

입력 예

```
NoSuchBucket
```

보낸 바이트

HTTP 프로토콜 오버헤드를 제외한 보낸 응답 바이트 수, 영일 경우 "-".

입력 예

```
2662992
```

객체 크기

해당 객체의 총 크기.

입력 예

```
3462992
```

총 시간

버킷의 관점에서 요청이 진행 중인 시간(밀리초)입니다. 이 값은 요청이 수신된 시간부터 응답의 마지막 바이트가 전송된 시간까지 측정됩니다. 클라이언트 관점의 측정값은 네트워크 지연 시간으로 인해 더 길 수 있습니다.

입력 예

```
70
```

반환 시간

Lightsail가 요청을 처리하는 데 소비한 시간(밀리초). 이 값은 요청의 마지막 바이트가 수신된 시간부터 응답의 첫 바이트가 전송된 시간까지 측정됩니다.

입력 예

```
10
```

참조자

HTTP Referrer 헤더의 값(있는 경우). HTTP 사용자 에이전트(예: 브라우저)는 일반적으로 이 헤더를 요청 시 연결 또는 포함 페이지의 URL로 설정합니다.

입력 예

```
"http://www.amazon.com/webservices"
```

사용자 에이전트

HTTP User-Agent 헤더의 값.

입력 예

```
"curl/7.15.1"
```

버전 ID

요청의 버전 ID 또는 작업이 `versionId` 파라미터를 사용하지 않는 경우 `-`입니다.

입력 예

```
3HL4kqtJvjVBH40Nrjfkd
```

호스트 Id

x-amz-id-2 또는 Lightsail 확장 요청 ID.

입력 예

```
s9lzHYrFp76ZVxRcpX9+5cjAnEH2ROuNkd2BHfIa6UkFVdtjf5mKR3/eTPFvsiP/XV/VLi31234=
```

서명 버전

요청을 인증하는 데 사용된 서명 버전, `SigV2` 또는 `SigV4` 또는 인증되지 않은 요청에 대한 `-`.

입력 예

```
SigV2
```

암호 그룹

HTTPS 요청에 대해 협상된 Secure Sockets Layer (SSL) 암호 또는 HTTP에 대해 협상된 `-` 암호.

입력 예

```
ECDHE-RSA-AES128-GCM-SHA256
```

인증 유형

사용된 요청 인증 유형, 인증 헤더의 경우 AuthHeader, 쿼리 문자열(사전 서명된 URL)에 대해서는 QueryString, 인증되지 않은 요청에 대해서는 -.

입력 예

```
AuthHeader
```

호스트 헤더

Lightsail에 연결하는 데 사용된 엔드포인트

입력 예

```
s3.us-west-2.amazonaws.com
```

TLS 버전

클라이언트가 협상한 TLS(전송 계층 보안) 버전. 값은 TLSv1, TLSv1.1, TLSv1.2 또는 -(TLS가 사용되지 않은 경우) 중 하나입니다.

입력 예

```
TLSv1.2
```

복사 작업을 위한 추가 로깅

복사 작업에는 GET 및 PUT이 관련됩니다. 그러므로 복사 작업을 수행할 때 2개의 레코드가 로그됩니다. 작업의 PUT 부분과 관련된 필드는 이전 표에 설명되어 있습니다. 아래 목록에서는 복사 작업의 GET 부분과 관련된 레코드의 필드에 대해 설명합니다.

버킷 소유자

복사 중인 객체가 저장되어 있는 버킷의 정식 사용자 ID입니다. 정식 사용자 ID는 또 다른 형식의 AWS 계정 ID입니다. 정식 사용자 ID에 대한 자세한 내용은 AWS 일반 참조에서 AWS 계정 식별자를 참조하세요. 계정의 정식 사용자 ID를 찾는 방법에 대한 자세한 내용은 AWS 계정의 정식 사용자 ID 찾기를 참조하세요.

입력 예

```
79a59df900b949e55d96a1e698fbacedfd6e09d98eacf8f8d5218e7cd47ef2be
```

버킷

복사 중인 객체가 저장되어 있는 버킷의 이름.

입력 예

```
amzn-s3-demo-bucket
```

Time

요청이 수신된 시간입니다. 이러한 날짜 및 시간은 협정 세계시(UTC)로 표시됩니다. `strftime()` 용어를 사용하는 형식은 `[%d/%B/%Y:%H:%M:%S %z]`입니다.

입력 예

```
[06/Feb/2019:00:00:38 +0000]
```

원격 IP

요청자의 명백한 인터넷 주소입니다. 중간 프록시 및 방화벽이 요청 시스템의 실제 주소를 가릴 수 있습니다.

입력 예

```
192.0.2.3
```

요청자

요청자의 정식 사용자 ID 또는 인증되지 않은 요청의 `-`입니다. 요청자가 IAM 사용자일 경우 IAM 사용자가 속한 AWS 루트 계정과 함께 요청자의 IAM 사용자 이름이 이 필드에 반환됩니다. 이 식별자는 액세스 제어 목적으로 사용되는 것과 동일합니다.

입력 예

```
79a59df900b949e55d96a1e698fbacedfd6e09d98eacf8f8d5218e7cd47ef2be
```

요청 ID

각 요청을 고유하게 식별하기 위해 Lightsail에서 생성한 문자열입니다.

입력 예

```
3E57427F33A59F07
```

작업

여기에 나열된 작업은 SOAP.*operation*, REST.*HTTP_method.resource_type*, WEBSITE.*HTTP_method.resource_type* 또는 BATCH.DELETE.OBJECT로 선언됩니다.

입력 예

```
REST.COPY.OBJECT_GET
```

Key(키)

복사 중인 객체의 "키", 작업에 키 파라미터가 없을 경우 "-".

입력 예

```
/photos/2019/08/puppy.jpg
```

요청-URII

HTTP 요청 메시지의 Request-URI 부분.

입력 예

```
"GET /amzn-s3-demo-bucket/photos/2019/08/puppy.jpg?x-foo=bar"
```

HTTP 상태

복사 작업 GET 부분의 숫자 HTTP 상태 코드.

입력 예

```
200
```

오류 코드

복사 작업 부분의 GET의 Amazon S3 오류 코드 또는 오류가 발생하지 않은 경우 -입니다.

입력 예

```
NoSuchBucket
```

보낸 바이트

HTTP 프로토콜 오버헤드를 제외한 보낸 응답 바이트 수, 영일 경우 "-".

입력 예

```
2662992
```

객체 크기

해당 객체의 총 크기.

입력 예

```
3462992
```

총 시간

버킷의 관점에서 요청이 진행 중인 시간(밀리초)입니다. 이 값은 요청이 수신된 시간부터 응답의 마지막 바이트가 전송된 시간까지 측정됩니다. 클라이언트 관점의 측정값은 네트워크 지연 시간으로 인해 더 길 수 있습니다.

입력 예

```
70
```

반환 시간

Lightsail가 요청을 처리하는 데 소비한 시간(밀리초). 이 값은 요청의 마지막 바이트가 수신된 시간부터 응답의 첫 바이트가 전송된 시간까지 측정됩니다.

입력 예

```
10
```

참조자

HTTP Referrer 헤더의 값(있는 경우). HTTP 사용자 에이전트(예: 브라우저)는 일반적으로 이 헤더를 요청 시 연결 또는 포함 페이지의 URL로 설정합니다.

입력 예

```
"http://www.amazon.com/webservices"
```

사용자 에이전트

HTTP User-Agent 헤더의 값.

입력 예

```
"curl/7.15.1"
```

버전 ID

복사 중인 객체의 버전 ID 또는 `x-amz-copy-source` 헤더가 복사 소스의 일부로 `versionId` 파라미터를 지정하지 않은 경우 `-`입니다.

입력 예

```
3HL4kqtJvjVBH40Nrjfkd
```

호스트 Id

x-amz-id-2 또는 Lightsail 확장 요청 ID.

입력 예

```
s9lzHYrFp76ZVxRcpX9+5cjAnEH2ROuNkd2BHfIa6UkFVdtjf5mKR3/eTPFvsiP/XV/VLi31234=
```

서명 버전

요청을 인증하는 데 사용된 서명 버전, `SigV2` 또는 `SigV4` 또는 인증되지 않은 요청에 대한 `-`.

입력 예

```
SigV2
```

암호 그룹

HTTPS 요청에 대해 협상된 Secure Sockets Layer (SSL) 암호 또는 HTTP에 대해 협상된 `-` 암호.

입력 예

```
ECDHE-RSA-AES128-GCM-SHA256
```

인증 유형

사용된 요청 인증 유형, 인증 헤더의 경우 AuthHeader이고, 쿼리 문자열(사전 서명된 URL)의 경우 QueryString이며 인증되지 않은 요청의 경우 -입니다.

입력 예

```
AuthHeader
```

호스트 헤더

Lightsail에 연결하는 데 사용된 엔드포인트

입력 예

```
s3.us-west-2.amazonaws.com
```

TLS 버전

클라이언트가 협상한 TLS(전송 계층 보안) 버전. 값은 TLSv1, TLSv1.1, TLSv1.2 또는 -(TLS가 사용되지 않은 경우) 중 하나입니다.

입력 예

```
TLSv1.2
```

사용자 지정 액세스 로그 정보

요청에 대한 액세스 로그 레코드에 저장할 사용자 지정 정보를 포함할 수 있습니다. 이렇게 하려면 요청에 대한 URL에 사용자 지정 쿼리 문자열 파라미터를 추가합니다. Lightsail는 "x-"로 시작하는 쿼리 문자열 파라미터를 무시하지만 요청에 대한 액세스 로그 레코드에 이러한 파라미터를 로그 레코드 Request-URI 필드의 일부로 포함합니다.

예를 들어 GET에 대한 "s3.amazonaws.com/amzn-s3-demo-bucket/photos/2019/08/puppy.jpg?x-user=johndoe" 요청은 관련 로그 레코드의 "s3.amazonaws.com/amzn-s3-

demo-bucket/photos/2019/08/puppy.jpg" 필드에 "x-user=johndoe" 문자열이 포함된다는 점을 제외하고 Request-URI에 대한 요청과 동일하게 작동합니다. 이 기능은 REST 인터페이스에서만 사용할 수 있습니다.

확장 가능한 액세스 로그 형식에 대한 프로그래밍 고려 사항

때때로 각 줄의 끝에 새로운 필드를 추가하여 액세스 로그 레코드 형식을 확장할 수 있습니다. 따라서 이해하지 못할 수 있는 후행 필드를 처리하기 위해 액세스 로그를 구문 분석하는 코드를 작성해야 합니다.

Lightsail에서 버킷 액세스 로깅 활성화

액세스 로깅은 Amazon Lightsail 객체 스토리지 서비스에서 버킷에 대한 요청에 대한 자세한 레코드를 제공합니다. 액세스 로그는 많은 애플리케이션에 유용합니다. 예를 들어 액세스 로그 정보는 보안 및 액세스 감사에 유용할 수 있습니다. 또한 고객 기반에 대해 알아보는 데 도움이 될 수 있습니다.

기본적으로 Lightsail는 버킷에 대한 액세스 로그를 수집하지 않습니다. 로깅을 활성화하면 Lightsail는 사용자가 선택한 대상 버킷에 소스 버킷에 대한 액세스 로그를 전달합니다. 소스 버킷과 대상 버킷은 모두 동일한에 AWS 리전 있어야 하며 동일한 계정이 소유해야 합니다.

액세스 로그 레코드에는 버킷에 대한 요청 내역이 자세히 나와 있습니다. 이 정보에는 요청 유형, 요청에 지정된 리소스, 요청이 처리된 시간 및 날짜가 포함될 수 있습니다. 이 가이드에서는 Lightsail API, AWS Command Line Interface (AWS CLI) 또는 AWS SDKs를 사용하여 버킷에 대한 액세스 로깅을 활성화하거나 비활성화하는 방법을 보여줍니다.

로깅 기초 사항에 대한 자세한 내용은 버킷 액세스 로그를 참조하세요.

목차

- 액세스 로깅 비용
- AWS CLI를 이용하여 액세스 로깅 활성화
- AWS CLI를 이용하여 액세스 로깅 비활성화

액세스 로깅 비용

버킷에 대한 액세스 로깅을 활성화하는 데는 추가 비용이 없습니다. 단, 시스템이 버킷에 전달하는 로그 파일은 스토리지 공간을 차지합니다. 로그 파일은 언제든지 삭제할 수 있습니다. 로그 버킷의 데이터 전송이 구성된 월별 허용 범위 내에 있을 경우 로그 파일 전달에 따른 데이터 전송 요금은 발생하지 않습니다.

대상 버킷은 액세스 로깅을 사용하도록 설정해서는 안 됩니다. 소스 버킷 자체를 포함하여 소스 버킷과 동일한 리전에 있는 자신의 고유 버킷에 로그를 전달할 수 있습니다. 그러나 로그 관리를 간소화하기 위해서는 액세스 로그를 다른 버킷에 저장하는 것이 좋습니다.

를 사용하여 액세스 로깅 활성화 AWS CLI

버킷에 대한 액세스 로깅을 활성화하려면 버킷이 AWS 리전 있는 각에 전용 로깅 버킷을 생성하는 것이 좋습니다. 그런 다음 해당 전용 로깅 버킷에 액세스 로그를 전달합니다.

AWS CLI를 사용하여 액세스 로깅을 활성화하려면 다음 절차를 완료합니다.

> **Note**
>
> 이 절차를 계속하기 Lightsail 전에 AWS CLI 를 설치하고에 대해 구성해야 합니다. 자세한 내용은 에서 작동 AWS CLI 하도록 구성을 Lightsail참조하세요.

1. 로컬 컴퓨터에서 명령 프롬프트 또는 터미널 창을 엽니다.
2. 액세스 로깅을 활성화하려면 다음 명령을 입력합니다.

```
aws lightsail update-bucket --bucket-name SourceBucketName --access-log-config
 "{\"enabled\": true, \"destination\": \"TargetBucketName\", \"prefix\":
 \"ObjectKeyNamePrefix/\"}"
```

명령에서 아래 예 텍스트를 사용자의 값으로 대체합니다.

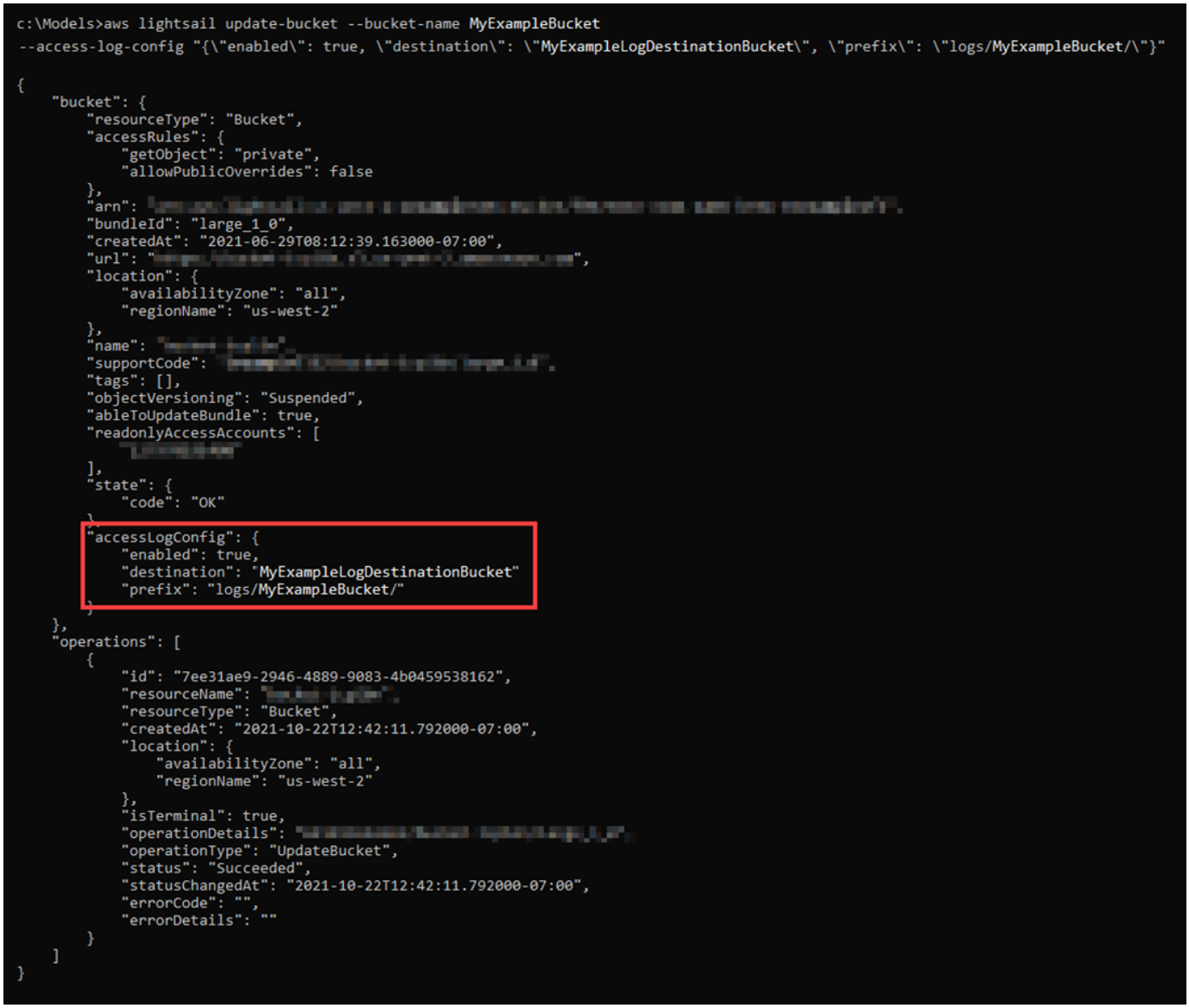
- *SourceBucketName* - 액세스 로그가 생성될 소스 버킷의 이름입니다.
- *TargetBucketName* — 액세스 로그가 저장될 대상 버킷의 이름입니다.
- *ObjectKeyNamePrefix/* - 액세스 로그에 대한 선택적 객체 키 이름 접두사입니다. 접두사는 슬래시(/)로 끝나야 합니다.

예제

```
aws lightsail update-bucket --bucket-name amzn-s3-demo-bucket1 --access-log-config
 "{\"enabled\": true, \"destination\": \"amzn-s3-demo-bucket2\", \"prefix\":
 \"logs/amzn-s3-demo-bucket1/\"}"
```

이 예에서 *amzn-s3-demo-bucket1*은 액세스 로그가 생성되는 소스 버킷이고, *amzn-s3-demo-bucket2*는 액세스 로그가 저장되는 대상 버킷이며, *logs/amzn-s3-demo-bucket1/*은 액세스 로그의 객체 키 이름 접두사입니다.

명령을 실행한 후 다음 예제와 유사한 결과가 표시되어야 합니다. 소스 버킷이 업데이트되고 액세스 로그가 생성되어 대상 버킷에 저장되기 시작합니다.



를 사용하여 액세스 로깅 비활성화 AWS CLI

AWS CLI를 사용하여 액세스 로깅을 비활성화하려면 다음 절차를 완료합니다.

> **Note**
>
> 이 절차를 계속하기 Lightsail 전에 AWS CLI 를 설치하고에 대해 구성해야 합니다. 자세한 내용은 에서 작동 AWS CLI 하도록 구성을 Lightsail참조하세요.

1. 로컬 컴퓨터에서 명령 프롬프트 또는 터미널 창을 엽니다.
2. 액세스 로깅을 비활성화하려면 다음 명령을 입력합니다.

```
aws lightsail update-bucket --bucket-name SourceBucketName --access-log-config
 "{\"enabled\": false}"
```

명령에서 *SourceBucketName*을 액세스 로깅을 비활성화할 소스 버킷의 이름으로 바꿉니다.

예제

```
aws lightsail update-bucket --bucket-name amzn-s3-demo-bucket --access-log-config
 "{\"enabled\": false}"
```

명령을 실행한 후 다음 예제와 유사한 결과가 표시되어야 합니다.

```
>aws lightsail update-bucket --bucket-name MyExampleBucket --access-log-config "{\"enabled\": false}"
{
    "bucket": {
        "resourceType": "Bucket",
        "accessRules": {
            "getObject": "private",
            "allowPublicOverrides": false
        },
        "arn": "
",
        "bundleId": "large_1_0",
        "createdAt": "2021-06-29T08:12:39.163000-07:00",
        "url":
        "location": {
            "availabilityZone": "all",
            "regionName": "us-west-2"
        },
        "name":
        "supportCode":
        "tags": [],
        "objectVersioning": "Suspended",
        "ableToUpdateBundle": true,
        "readonlyAccessAccounts": [
            "
"
        ],
        "state": {
            "code": "OK"
        },
        "accessLogConfig": {
            "enabled": false
        }
    },
    "operations": [
        {
            "id":
            "resourceName":
            "resourceType": "Bucket",
            "createdAt": "2021-10-22T13:24:36.881000-07:00",
            "location": {
                "availabilityZone": "all",
                "regionName": "us-west-2"
            },
            "isTerminal": true,
            "operationDetails":
            "operationType": "UpdateBucket",
            "status": "Succeeded",
            "statusChangedAt": "2021-10-22T13:24:36.881000-07:00",
            "errorCode": "",
            "errorDetails": ""
        }
    ]
}
```

Lightsail에서 Amazon Athena를 사용하여 버킷 액세스 로그 분석

이 안내서에서는 액세스 로그를 사용하여 버킷에 대한 요청을 식별하는 방법을 보여줍니다. 자세한 내용은 버킷 액세스 로그를 참조하세요.

목차

- Amazon Athena를 사용하여 요청에 대한 액세스 로그 쿼리
- Amazon S3 액세스 로그를 사용하여 객체 액세스 요청 식별

Amazon Athena를 사용하여 요청에 대한 액세스 로그 쿼리

Amazon Athena Athena를 사용하여 액세스 로그에서 버킷에 대한 요청을 쿼리하고 식별할 수 있습니다.

Lightsail에서는 액세스 로그를 Lightsail 버킷의 객체로 저장합니다. 로그를 분석할 수 있는 도구를 사용하는 것이 보통 더 쉽습니다. Athena는 객체의 분석을 지원하며 액세스 로그를 쿼리하는 데 사용할 수 있습니다.

예제

다음 예에서는 Amazon Athena에서 버킷 서버 액세스 로그를 쿼리할 수 있는 방법을 보여줍니다.

> **Note**
>
> Athena 쿼리에서 버킷 위치를 지정하려면 다음과 같이 로그가 S3 URI로 전달되는 대상 버킷 이름과 대상 접두사의 형식을 지정해야 합니다.`s3://amzn-s3-demo-bucket1-logs/prefix/`

1. https://console.aws.amazon.com/athena/에서 Athena 콘솔을 엽니다.
2. 쿼리 편집기에서 다음과 유사한 명령을 실행합니다.

```
create database bucket_access_logs_db
```

> **Note**
>
> S3 버킷 AWS 리전 과 동일한에 데이터베이스를 생성하는 것이 가장 좋습니다.

3. 쿼리 편집기에서 다음과 유사한 명령을 실행하여 2단계에서 생성한 데이터베이스에 테이블 스키마를 생성합니다. `STRING` 및 `BIGINT` 데이터 형식 값이 액세스 로그 속성입니다. Athena에서 이 속성을 쿼리할 수 있습니다. `LOCATION`의 경우 앞서 설명한 대로 버킷 및 접두사 경로를 입력하십시오.

```
CREATE EXTERNAL TABLE `s3_access_logs_db.amzn-s3-demo-bucket_logs`(
  `bucketowner` STRING,
  `bucket_name` STRING,
  `requestdatetime` STRING,
  `remoteip` STRING,
  `requester` STRING,
```

```
  `requestid` STRING,
  `operation` STRING,
  `key` STRING,
  `request_uri` STRING,
  `httpstatus` STRING,
  `errorcode` STRING,
  `bytessent` BIGINT,
  `objectsize` BIGINT,
  `totaltime` STRING,
  `turnaroundtime` STRING,
  `referrer` STRING,
  `useragent` STRING,
  `versionid` STRING,
  `hostid` STRING,
  `sigv` STRING,
  `ciphersuite` STRING,
  `authtype` STRING,
  `endpoint` STRING,
  `tlsversion` STRING)
ROW FORMAT SERDE
  'org.apache.hadoop.hive.serde2.RegexSerDe'
WITH SERDEPROPERTIES (
  'input.regex'='([^ ]*) ([^ ]*) \\[(.*?)\\] ([^ ]*) ([^ ]*) ([^ ]*) ([^ ]*)
 ([^ ]*) (\"[^\"]*\"|-) (-|[0-9]*) ([^ ]*) ([^ ]*) ([^ ]*) ([^ ]*) ([^ ]*) ([^ ]*)
 (\"[^\"]*\"|-) ([^ ]*)(?: ([^ ]*) ([^ ]*) ([^ ]*) ([^ ]*) ([^ ]*) ([^ ]*))?.*$')
STORED AS INPUTFORMAT
  'org.apache.hadoop.mapred.TextInputFormat'
OUTPUTFORMAT
  'org.apache.hadoop.hive.ql.io.HiveIgnoreKeyTextOutputFormat'
LOCATION
  's3://amzn-s3-demo-bucket1-logs/prefix/'
```

4. 탐색 창의 데이터베이스 아래에서 데이터베이스를 선택하십시오.
5. 테이블에서 테이블 이름 옆의 Preview table(테이블 미리보기)을 선택하십시오.

   결과 창에 bucketowner, bucket, requestdatetime 등 서버 액세스 로그의 데이터가 표시됩니다. 즉, Athena 테이블이 만들어졌다는 뜻입니다. 이제 버킷 서버 액세스 로그를 쿼리할 수 있습니다.

예 - 객체를 삭제한 사람과 시점 표시(타임스탬프, IP 주소 및 IAM 사용자)

```
SELECT RequestDateTime, RemoteIP, Requester, Key
```

```
FROM s3_access_logs_db.amzn-s3-demo-bucket_logs
WHERE key = 'images/picture.jpg' AND operation like '%DELETE%';
```

예 - IAM 사용자가 수행한 모든 작업 표시

```
SELECT *
FROM s3_access_logs_db.amzn-s3-demo-bucket_logs
WHERE requester='arn:aws:iam::123456789123:user/user_name';
```

예 - 특정 기간에 객체에 수행된 모든 작업 표시

```
SELECT *
FROM s3_access_logs_db.amzn-s3-demo-bucket_logs
WHERE Key='prefix/images/picture.jpg'
    AND parse_datetime(RequestDateTime,'dd/MMM/yyyy:HH:mm:ss Z')
    BETWEEN parse_datetime('2017-02-18:07:00:00','yyyy-MM-dd:HH:mm:ss')
    AND parse_datetime('2017-02-18:08:00:00','yyyy-MM-dd:HH:mm:ss');
```

예 - 특정 기간에 특정 IP 주소에서 전송한 데이터의 양 표시

```
SELECT SUM(bytessent) AS uploadTotal,
      SUM(objectsize) AS downloadTotal,
      SUM(bytessent + objectsize) AS Total
FROM s3_access_logs_db.amzn-s3-demo-bucket_logs
WHERE RemoteIP='1.2.3.4'
AND parse_datetime(RequestDateTime,'dd/MMM/yyyy:HH:mm:ss Z')
BETWEEN parse_datetime('2017-06-01','yyyy-MM-dd')
AND parse_datetime('2017-07-01','yyyy-MM-dd');
```

Amazon S3 액세스 로그를 사용하여 객체 액세스 요청 식별

액세스 로그에 대한 쿼리를 사용하여 GET, PUT 및 DELETE와 같은 작업에 대한 객체 액세스 요청을 식별하고 해당 요청에 대한 추가 정보를 검색할 수 있습니다.

다음 Amazon Athena 쿼리 예제에서는 서버 액세스 로그에서 버킷에 대한 모든 PUT 객체 요청을 가져오는 방법을 보여줍니다.

예 - 특정 기간에 PUT 객체 요청을 보내는 모든 요청자 표시

```
SELECT Bucket, Requester, RemoteIP, Key, HTTPStatus, ErrorCode, RequestDateTime
FROM s3_access_logs_db
```

```
WHERE Operation='REST.PUT.OBJECT' AND
parse_datetime(RequestDateTime,'dd/MMM/yyyy:HH:mm:ss Z')
BETWEEN parse_datetime('2019-07-01:00:42:42','yyyy-MM-dd:HH:mm:ss')
AND
parse_datetime('2019-07-02:00:42:42','yyyy-MM-dd:HH:mm:ss')
```

다음 Amazon Athena 쿼리 예제에서는 서버 액세스 로그에서 Amazon S3에 대한 모든 GET 객체 요청을 가져오는 방법을 보여 줍니다.

예 - 특정 기간에 GET 객체 요청을 보내는 모든 요청자 표시

```
SELECT Bucket, Requester, RemoteIP, Key, HTTPStatus, ErrorCode, RequestDateTime
FROM s3_access_logs_db
WHERE Operation='REST.GET.OBJECT' AND
parse_datetime(RequestDateTime,'dd/MMM/yyyy:HH:mm:ss Z')
BETWEEN parse_datetime('2019-07-01:00:42:42','yyyy-MM-dd:HH:mm:ss')
AND
parse_datetime('2019-07-02:00:42:42','yyyy-MM-dd:HH:mm:ss')
```

다음 Amazon Athena 쿼리 예제에서는 서버 액세스 로그에서 S3 버킷에 대한 모든 익명 요청을 가져오는 방법을 보여 줍니다.

예 - 특정 기간에 버킷에 요청하는 모든 익명 요청자 표시

```
SELECT Bucket, Requester, RemoteIP, Key, HTTPStatus, ErrorCode, RequestDateTime
FROM s3_access_logs_db.amzn-s3-demo-bucket_logs
WHERE Requester IS NULL AND
parse_datetime(RequestDateTime,'dd/MMM/yyyy:HH:mm:ss Z')
BETWEEN parse_datetime('2019-07-01:00:42:42','yyyy-MM-dd:HH:mm:ss')
AND
parse_datetime('2019-07-02:00:42:42','yyyy-MM-dd:HH:mm:ss')
```

Note

- 필요에 맞게 날짜 범위를 수정할 수 있습니다.
- 이 쿼리 예제는 보안 모니터링에도 유용할 수 있습니다. 예상치 못하거나 승인되지 않은 IP 주소/요청자의 `PutObject` 또는 `GetObject` 호출 결과를 검토하고 버킷에 대한 익명 요청을 식별할 수 있습니다.
- 이 쿼리는 로깅이 사용 설정된 시간부터의 정보만 검색합니다.

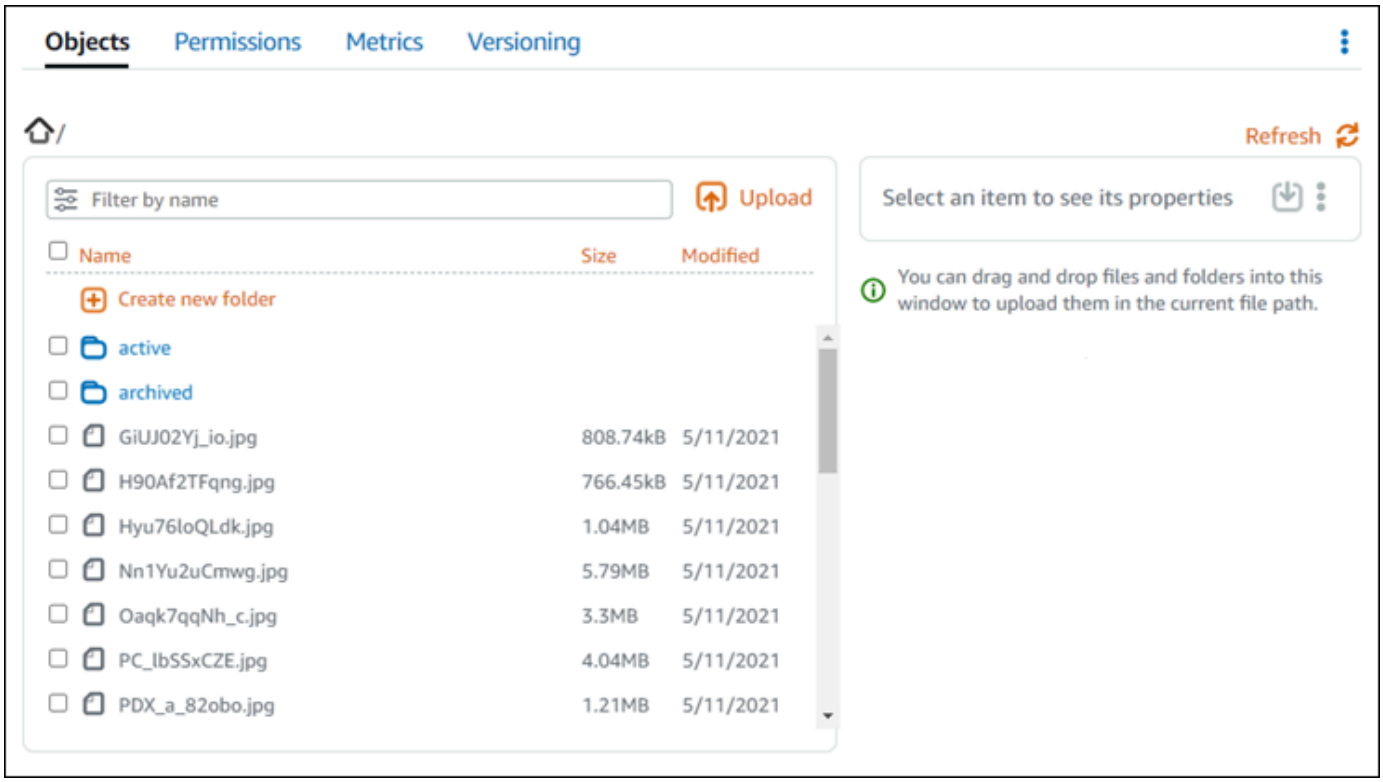
Lightsail 버킷의 파일 및 폴더 관리

Lightsail 콘솔을 사용하여 Amazon Lightsail 객체 스토리지 서비스의 버킷에 저장된 모든 객체를 볼 수 있습니다. AWS Command Line Interface (AWS CLI) 및 AWS SDKs를 사용하여 버킷의 객체 키를 나열할 수도 있습니다. 버킷에 대한 자세한 내용은 객체 스토리지를 참조하세요.

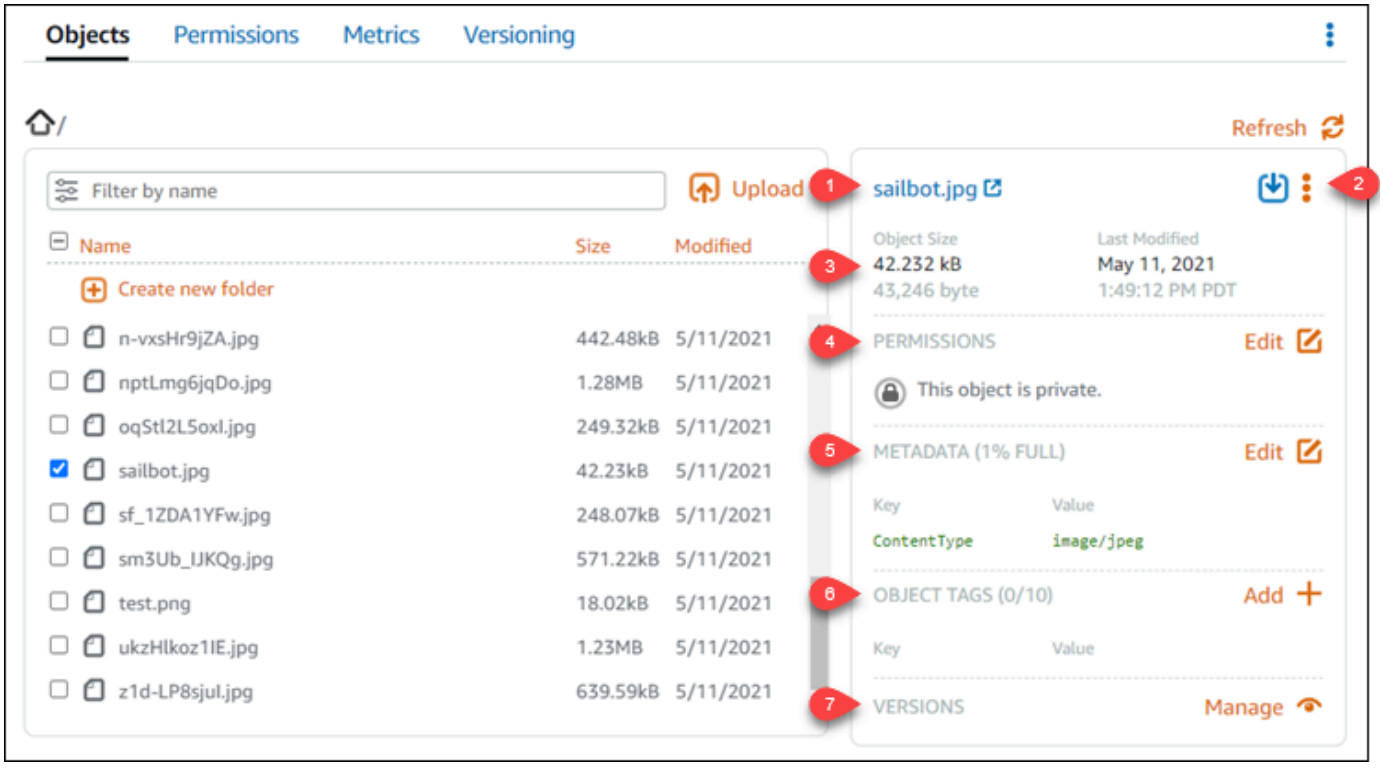
Lightsail 콘솔을 사용한 객체 필터링

Lightsail 콘솔을 사용하여 버킷에 저장된 객체를 보려면 다음 절차를 완료하세요.

1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 스토리지를 선택합니다.
3. 객체에 태그를 보려는 버킷의 이름을 선택합니다.
4. 객체(Objects) 탭의 객체(Objects) 브라우저 창에는 버킷에 저장된 객체와 폴더가 표시됩니다.



5. 속성을 보려는 객체의 위치를 찾습니다.
6. 속성을 보려는 객체 옆에 확인 표시를 추가합니다.
7. 페이지 오른쪽에 있는 객체 속성(Object properties) 창에서 객체에 대한 정보를 확인할 수 있습니다.



표시되는 정보는 다음과 같습니다.

1. 객체를 보고 다운로드할 수 있는 링크.
2. 객체를 복사하거나 삭제할 작업 메뉴(⋮). 객체 복사 및 삭제에 대한 자세한 내용은 Amazon Lightsail 버킷의 객체 복사 또는 이동 및 버킷 객체 삭제를 참조하세요.
3. 객체 크기 및 마지막으로 수정한 타임스탬프.
4. 비공개 또는 공개(읽기 전용)일 수 있는 개별 객체의 액세스 권한. 객체 권한에 대한 자세한 내용은 버킷 권한을 참조하세요.
5. 객체의 메타데이터. 콘텐츠 유형(`ContentType`) 키는 현재 Lightsail 객체 스토리지 서비스에서 지원하는 유일한 메타데이터입니다.
6. 객체 키 값 태그. 자세한 내용은 버킷 객체 태그 지정을 참조하세요.
7. 저장된 객체 버전을 관리하는 옵션. 자세한 내용은 버킷의 객체 버전 사용 설정 및 사용 중지를 참조하세요.

> **Note**
>
> 객체를 여러 개 선택하면 객체 속성(Object properties) 창에는 선택한 객체의 전체 크기만 표시됩니다.

를 사용하여 객체 보기 AWS CLI

AWS Command Line Interface (AWS CLI)를 사용하여 버킷의 객체 키를 나열하려면 다음 절차를 완료하세요. `list-objects-v2` 명령을 사용하여 이 작업을 수행할 수 있습니다. 자세한 내용은 AWS CLI 명령 레퍼런스에서 list-objects-v2를 참조하세요.

> **Note**
>
> 이 절차를 계속하기 전에 AWS CLI 및 Amazon S3에 대해 Lightsail를 설치하고 구성해야 합니다. 자세한 내용은 에서 작동 AWS Command Line Interface 하도록 구성을 Amazon Lightsail참조하세요.

1. 명령 프롬프트 또는 터미널 창을 엽니다.
2. 다음 명령 중 하나를 입력합니다.

   - 다음 명령을 입력하여 버킷의 객체 키를 나열합니다.

     ```
     aws s3api list-objects-v2 --bucket BucketName --query "Contents[].{Key: Key,
      Size: Size}"
     ```

     명령에서 *BucketName*을 모든 객체를 나열하려는 버킷의 이름으로 바꿉니다.

   - 다음 명령을 입력하여 특정 객체 키 이름 접두사로 시작하는 객체를 나열합니다.

     ```
     aws s3api list-objects-v2 --bucket BucketName --prefix ObjectKeyNamePrefix --
     query "Contents[].{Key: Key, Size: Size}"
     ```

     명령에서 아래 예 텍스트를 사용자의 값으로 대체합니다.

     - *BucketName* - 모든 객체를 나열할 버킷의 이름입니다.
     - *ObjectKeyNamePrefix* - 지정된 접두사로 시작하는 키로 응답을 제한할 객체 키 이름 접두사입니다.

     > **Note**
     >
     > 이 명령은 `--query` 파라미터를 사용하여 각 객체의 키 값과 크기에 대한 `list-objects-v2` 요청 응답을 필터링합니다.

예시:

버킷의 모든 객체 버전 나열:

```
aws s3api list-objects-v2 --bucket amzn-s3-demo-bucket --query "Contents[].{Key:
 Key, Size: Size}"
```

위의 명령은 다음 예와 비슷한 결과가 나타나는 것을 볼 수 있습니다.

```
C:\>aws s3api list-objects-v2 --bucket DOC-EXAMPLE-BUCKET --query "Contents[].{Key: Key, Size: Size}"
[
    {
        "Key": "GiUJ02Yj_io.jpg",
        "Size": 828150
    },
    {
        "Key": "H90Af2TFqng.jpg",
        "Size": 784846
    },
    {
        "Key": "Hyu76loQLdk.jpg",
        "Size": 1086363
    },
    {
        "Key": "Nn1Yu2uCmwg.jpg",
        "Size": 6075006
    },
    {
        "Key": "Oaqk7qqNh_c.jpg",
        "Size": 3458557
    },
    {
        "Key": "PC_lbSSxCZE.jpg",
        "Size": 4239636
    },
    {
        "Key": "PDX_a_82obo.jpg"
```

archived/ 객체 키 접두사로 시작하는 객체 키 나열:

```
aws s3api list-objects-v2 --bucket amzn-s3-demo-bucket --prefix archived/ --query
 "Contents[].{Key: Key, Size: Size}"
```

위의 명령은 다음 예와 비슷한 결과가 나타나는 것을 볼 수 있습니다.

```
C:\>aws s3api list-objects-v2  --bucket DOC-EXAMPLE-BUCKET --prefix archived/ --query "Contents[].{Key: Key, Size: Size}"
[
    {
        "Key": "archived/",
        "Size": 0
    },
    {
        "Key": "archived/1_CMoFsPfso.jpg",
        "Size": 2561865
    },
    {
        "Key": "archived/3y1zF4hIPCg.jpg",
        "Size": 6404907
    },
    {
        "Key": "archived/5IHz5WhosQE.jpg",
        "Size": 2377975
    },
    {
        "Key": "archived/sailbot.jpg",
        "Size": 43246
    }
]
```

버킷 및 객체 관리

다음은 Lightsail 객체 스토리지 버킷을 관리하는 일반적인 단계입니다.

1. Amazon Lightsail 객체 스토리지 서비스의 객체와 버킷에 대해 알아보세요. 자세한 내용은 Amazon Lightsail의 객체 스토리지를 참조하세요.
2. Amazon Lightsail에서 버킷에 지정할 수 있는 이름에 대해 알아봅니다. 자세한 내용은 Amazon Lightsail의 버킷 이름 지정 규칙을 참조하세요.
3. 버킷을 생성하여 Lightsail 객체 스토리지 서비스를 시작합니다. 자세한 내용은 Amazon Lightsail에서 버킷 생성을 참조하세요.
4. 버킷의 보안 모범 사례와 버킷에 구성할 수 있는 액세스 권한에 대해 알아봅니다. 버킷의 모든 객체를 공개 또는 비공개로 설정하거나 개별 객체를 공개로 설정하도록 선택할 수 있습니다. 액세스 키를 생성하고, 버킷에 인스턴스를 연결하고, 다른 AWS 계정에 액세스 권한을 부여하여 버킷에 대한 액세스 권한을 부여할 수도 있습니다. 자세한 내용은 Amazon Lightsail 객체 스토리지에 대한 보안 모범 사례 및 Amazon Lightsail의 버킷 권한 이해를 참조하세요.

   버킷 액세스 권한에 대해 알아본 후 다음 가이드를 참조하여 버킷에 대한 액세스 권한을 부여합니다.

   - Amazon Lightsail의 버킷에 대한 퍼블릭 액세스 차단
   - Amazon Lightsail에서 버킷 액세스 권한 구성
   - Amazon Lightsail에서 버킷의 개별 객체에 대한 액세스 권한 구성
   - Amazon Lightsail에서 버킷의 액세스 키 생성

- Amazon Lightsail에서 버킷에 대한 리소스 액세스 구성
- Amazon Lightsail에서 버킷의 크로스 계정 액세스 구성

5. 버킷에 대한 액세스 로깅을 활성화하는 방법과 액세스 로그를 사용하여 버킷의 보안을 감사하는 방법에 대해 알아봅니다. 자세한 내용은 다음 안내서를 참조하세요.
   - Amazon Lightsail 객체 스토리지 서비스의 버킷에 대한 액세스 로깅
   - Amazon Lightsail 객체 스토리지 서비스에서 버킷에 대한 액세스 로그 형식
   - Amazon Lightsail 객체 스토리지 서비스에서 버킷에 대한 액세스 로깅 활성화
   - Amazon Lightsail에서 버킷에 대한 액세스 로그 사용하여 요청을 식별
6. Lightsail에서 버킷을 관리할 수 있는 기능을 사용자에게 부여하는 IAM 정책을 생성합니다. 자세한 내용은 Amazon Lightsail에서 버킷을 관리하기 위한 IAM 정책을 참조하세요.
7. 버킷의 객체에 라벨을 지정하고 식별하는 방식에 대해 알아봅니다. 자세한 내용은 Amazon Lightsail에서 객체 키 이름 이해를 참조하세요.
8. 파일을 업로드하고 버킷의 객체를 관리하는 방법에 대해 알아봅니다. 자세한 내용은 다음 안내서를 참조하세요.
   - Amazon Lightsail에서 버킷으로 파일 업로드
   - 멀티파트 업로드를 사용하여 Amazon Lightsail에서 버킷으로 파일 업로드
   - Amazon Lightsail에서 버킷의 객체 확인
   - Amazon Lightsail에서 버킷의 객체 복사 또는 이동
   - Amazon Lightsail에서 버킷의 객체 다운로드
   - Amazon Lightsail에서 버킷의 객체 필터링
   - Amazon Lightsail에서 버킷의 객체 태그 지정
   - Amazon Lightsail에서 버킷의 객체 삭제
9. 객체 버전 관리를 활성화하면 버킷에 저장된 모든 버전의 객체를 모두 보존, 검색 및 복원할 수 있습니다. 자세한 내용은 Amazon Lightsail에서 버킷의 객체 버전 사용 설정 및 사용 중지를 참조하세요.
10. 객체 버전 관리를 활성화한 후 버킷에 있는 객체의 이전 버전을 복원할 수 있습니다. 자세한 내용은 Amazon Lightsail에서 버킷의 이전 객체 버전 복원을 참조하세요.
11. 버킷 사용률을 모니터링합니다. 자세한 내용은 Amazon Lightsail에서 버킷에 대한 지표 보기를 참조하세요.
12. 버킷 사용률이 임곗값을 초과할 때 알림을 받도록 버킷 지표에 대한 경보를 구성합니다. 자세한 내용은 Amazon Lightsail에서 버킷 지표 경보 생성을 참조하세요.
13. 스토리지 및 네트워크 전송이 부족한 경우 버킷의 스토리지 플랜을 변경합니다. 자세한 내용은 Amazon Lightsail에서 버킷 플랜 변경을 참조하세요.

14.버킷을 다른 리소스에 연결하는 방법에 대해 알아봅니다. 자세한 내용은 다음 자습서를 참조하세요.

- 자습서: WordPress 인스턴스를 Amazon Lightsail 버킷에 연결
- 자습서: Lightsail 콘텐츠 전송 네트워크 배포와 함께 Amazon Lightsail 버킷 사용

15.버킷을 더 이상 사용하지 않는 경우 삭제할 수 있습니다. 자세한 내용은 Amazon Lightsail에서 버킷 삭제를 참조하세요.

주제

- Lightsail 버킷 간에 객체 복사 및 이동
- 객체를 삭제하여 Lightsail 버킷 스토리지 지우기
- Lightsail 버킷의 객체 다운로드
- 이름 접두사를 기준으로 Lightsail 버킷의 객체 필터링
- Lightsail에서 객체 버전 관리 활성화 및 일시 중지
- Lightsail 버킷에서 이전 객체 버전 복구
- Lightsail 버킷의 객체에 태그 지정

Lightsail 버킷 간에 객체 복사 및 이동

Amazon Lightsail 객체 스토리지 서비스의 버킷에 이미 저장된 객체를 복사할 수 있습니다. 이 가이드에서는 Lightsail 콘솔과 AWS Command Line Interface ()를 사용하여 객체를 복사하는 방법을 보여줍니다AWS CLI. 버킷의 객체를 복사하여 객체의 복제본을 생성하거나, 객체의 이름을 바꾸거나, 객체 AWS 리전 를 Lightsail 여러 위치로 이동합니다(예: 사용 Lightsail 가능한 객체를 한 위치에서 다른 위치로 이동). AWS APIs, AWS SDKs 및 AWS Command Line Interface ()만 사용하여 위치 간에 객체를 복사할 수 있습니다AWS CLI.

버킷에 대한 자세한 내용은 객체 스토리지를 참조하세요.

객체 복사 제한 사항

Lightsail 콘솔을 사용하여 최대 2GB 크기의 객체 사본을 생성할 수 있습니다. AWS Command Line Interface (AWS CLI), AWS APIs 및 AWS SDKs. 크기가 5GB보다 큰 객체를 복사하려면 , AWS CLI AWS APIs 및 AWS SDKs의 멀티파트 업로드 작업을 사용해야 합니다. 자세한 내용은 멀티파트 업로드를 사용하여 버킷으로 파일 업로드를 참조하세요.

Lightsail 콘솔을 사용한 객체 복사

Lightsail 콘솔을 사용하여 버킷에 저장된 객체를 복사하려면 다음 절차를 완료하세요. 버킷의 객체를 이동하려면 객체를 새 위치로 복사하고 원본 객체를 삭제해야 합니다.

1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 스토리지를 선택합니다.
3. 객체를 복사하려는 버킷의 이름을 선택합니다.
4. 객체(Objects) 탭에서 객체 브라우저 창(Objects browser pane)을 사용하여 복사할 객체의 위치를 찾습니다.
5. 복사할 객체 옆에 확인 표시를 추가합니다.
6. 객체 정보(Object information) 창에서 작업(⋮) 메뉴를 선택한 후 다음으로 복사(Copy to)를 선택합니다.
7. 대상 선택(Select destination) 창이 표시되면 선택한 객체를 복사할 버킷의 위치를 찾습니다. 대상 경로(Destination path) 텍스트 상자에 폴더 이름을 입력하여 새 경로를 생성할 수도 있습니다.
8. 복사(Copy)를 선택하여 선택한 대상이나 지정한 대상에 객체를 복사합니다. 복사하지 않으려면 아니요, 취소합니다(No, cancel)를 선택합니다.

   객체가 성공적으로 복사되면 복사 완료(Copy complete) 메시지가 표시됩니다. 객체를 이동하려는 경우 원본 객체를 삭제해야 합니다. 자세한 내용은 버킷 객체 삭제를 참조하세요.

를 사용하여 객체 복사 AWS CLI

AWS Command Line Interface ()를 사용하여 버킷의 객체를 복사하려면 다음 절차를 완료합니다AWS CLI. `copy-object` 명령을 사용하여 이 작업을 수행할 수 있습니다. 자세한 내용은 AWS CLI 명령 레퍼런스의 copy-object를 참조하세요.

> **Note**
>
> 이 절차를 계속하기 전에 AWS CLI 및 Amazon S3에 대해 Lightsail를 설치하고 구성해야 합니다. 자세한 내용은 에서 작동 AWS CLI 하도록 구성을 Lightsail참조하세요.

1. 명령 프롬프트 또는 터미널 창을 엽니다.
2. 다음 명령을 입력하여 버킷의 객체를 복사합니다.

```
aws s3api copy-object --copy-source SourceBucketNameAndObjectKey --
key DestinationObjectKey --bucket DestinationBucketName --acl bucket-owner-full-
control
```

다음 명령에서 아래 텍스트를 사용자의 값으로 대체합니다.

- *SourceBucketNameAndObjectKey* - 현재 소스 객체가 존재하는 버킷의 이름 및 복사할 객체의 전체 객체 키입니다. 예를 들어, 버킷 amzn-s3-demo-bucket에서 images/sailbot.jpg 객체를 복사하려면 amzn-s3-demo-bucket/images/sailbot.jpg를 지정하면 됩니다.
- *DestinationObjectKey* - 새 객체 사본의 전체 객체 키입니다.
- *DestinationBucket* - 대상 버킷의 이름입니다.

예시:

- 버킷의 객체를 동일한 버킷에 복사:

```
aws s3api copy-object --copy-source amzn-s3-demo-bucket1/images/sailbot.jpg
 --key media/sailbot.jpg --bucket amzn-s3-demo-bucket --acl bucket-owner-full-
control
```

- 한 버킷에서 다른 버킷으로 객체 복사:

```
aws s3api copy-object --copy-source amzn-s3-demo-bucket1/images/sailbot.jpg --
key images/sailbot.jpg --bucket amzn-s3-demo-bucket2 --acl bucket-owner-full-
control
```

다음 예와 비슷한 결과가 나타나야 합니다.

```
C:\>aws s3api copy-object --copy-source DOC-EXAMPLE-BUCKET/images/sailbot.jpg --key images/archived/sailbot.jpg --bucket DOC-EXAMPLE-BUCKET
{
    "ServerSideEncryption": "AES256",
    "CopyObjectResult": {
        "ETag": "\"694d34example91d92d64f342aa234c3\"",
        "LastModified": "2021-05-10T05:35:42+00:00"
    }
}
```

버킷 및 객체 관리

다음은 Lightsail 객체 스토리지 버킷을 관리하는 일반적인 단계입니다.

1. Amazon Lightsail 객체 스토리지 서비스의 객체와 버킷에 대해 알아보세요. 자세한 내용은 Amazon Lightsail의 객체 스토리지를 참조하세요.
2. Amazon Lightsail에서 버킷에 지정할 수 있는 이름에 대해 알아봅니다. 자세한 내용은 Amazon Lightsail의 버킷 이름 지정 규칙을 참조하세요.
3. 버킷을 생성하여 Lightsail 객체 스토리지 서비스를 시작합니다. 자세한 내용은 Amazon Lightsail에서 버킷 생성을 참조하세요.
4. 버킷의 보안 모범 사례와 버킷에 구성할 수 있는 액세스 권한에 대해 알아봅니다. 버킷의 모든 객체를 공개 또는 비공개로 설정하거나 개별 객체를 공개로 설정하도록 선택할 수 있습니다. 액세스 키를 생성하고, 버킷에 인스턴스를 연결하고, 다른 AWS 계정에 액세스 권한을 부여하여 버킷에 대한 액세스 권한을 부여할 수도 있습니다. 자세한 내용은 Amazon Lightsail 객체 스토리지에 대한 보안 모범 사례 및 Amazon Lightsail의 버킷 권한 이해를 참조하세요.

   버킷 액세스 권한에 대해 알아본 후 다음 가이드를 참조하여 버킷에 대한 액세스 권한을 부여합니다.

   - Amazon Lightsail의 버킷에 대한 퍼블릭 액세스 차단
   - Amazon Lightsail에서 버킷 액세스 권한 구성
   - Amazon Lightsail에서 버킷의 개별 객체에 대한 액세스 권한 구성
   - Amazon Lightsail에서 버킷의 액세스 키 생성
   - Amazon Lightsail에서 버킷에 대한 리소스 액세스 구성
   - Amazon Lightsail에서 버킷의 크로스 계정 액세스 구성
5. 버킷에 대한 액세스 로깅을 활성화하는 방법과 액세스 로그를 사용하여 버킷의 보안을 감사하는 방법에 대해 알아봅니다. 자세한 내용은 다음 안내서를 참조하세요.

   - Amazon Lightsail 객체 스토리지 서비스의 버킷에 대한 액세스 로깅
   - Amazon Lightsail 객체 스토리지 서비스에서 버킷에 대한 액세스 로그 형식
   - Amazon Lightsail 객체 스토리지 서비스에서 버킷에 대한 액세스 로깅 활성화
   - Amazon Lightsail에서 버킷에 대한 액세스 로그 사용하여 요청을 식별
6. Lightsail에서 버킷을 관리할 수 있는 기능을 사용자에게 부여하는 IAM 정책을 생성합니다. 자세한 내용은 Amazon Lightsail에서 버킷을 관리하기 위한 IAM 정책을 참조하세요.
7. 버킷의 객체에 라벨을 지정하고 식별하는 방식에 대해 알아봅니다. 자세한 내용은 Amazon Lightsail에서 객체 키 이름 이해를 참조하세요.
8. 파일을 업로드하고 버킷의 객체를 관리하는 방법에 대해 알아봅니다. 자세한 내용은 다음 안내서를 참조하세요.

   - Amazon Lightsail에서 버킷으로 파일 업로드

- 멀티파트 업로드를 사용하여 Amazon Lightsail에서 버킷으로 파일 업로드
- Amazon Lightsail에서 버킷의 객체 확인
- Amazon Lightsail에서 버킷의 객체 복사 또는 이동
- Amazon Lightsail에서 버킷의 객체 다운로드
- Amazon Lightsail에서 버킷의 객체 필터링
- Amazon Lightsail에서 버킷의 객체 태그 지정
- Amazon Lightsail에서 버킷의 객체 삭제

9. 객체 버전 관리를 활성화하면 버킷에 저장된 모든 버전의 객체를 모두 보존, 검색 및 복원할 수 있습니다. 자세한 내용은 Amazon Lightsail에서 버킷의 객체 버전 사용 설정 및 사용 중지를 참조하세요.
10. 객체 버전 관리를 활성화한 후 버킷에 있는 객체의 이전 버전을 복원할 수 있습니다. 자세한 내용은 Amazon Lightsail에서 버킷의 이전 객체 버전 복원을 참조하세요.
11. 버킷 사용률을 모니터링합니다. 자세한 내용은 Amazon Lightsail에서 버킷에 대한 지표 보기를 참조하세요.
12. 버킷 사용률이 임곗값을 초과할 때 알림을 받도록 버킷 지표에 대한 경보를 구성합니다. 자세한 내용은 Amazon Lightsail에서 버킷 지표 경보 생성을 참조하세요.
13. 스토리지 및 네트워크 전송이 부족한 경우 버킷의 스토리지 플랜을 변경합니다. 자세한 내용은 Amazon Lightsail에서 버킷 플랜 변경을 참조하세요.
14. 버킷을 다른 리소스에 연결하는 방법에 대해 알아봅니다. 자세한 내용은 다음 자습서를 참조하세요.
    - 자습서: WordPress 인스턴스를 Amazon Lightsail 버킷에 연결
    - 자습서: Lightsail 콘텐츠 전송 네트워크 배포와 함께 Amazon Lightsail 버킷 사용
15. 버킷을 더 이상 사용하지 않는 경우 삭제할 수 있습니다. 자세한 내용은 Amazon Lightsail에서 버킷 삭제를 참조하세요.

객체를 삭제하여 Lightsail 버킷 스토리지 지우기

Amazon Lightsail 객체 스토리지 서비스에서 버킷의 객체를 삭제할 수 있습니다. 스토리지 공간을 확보하려면 더 이상 필요하지 않은 객체를 삭제하면 됩니다. 예를 들어, 로그 파일을 수집하는 경우 더 이상 필요하지 않은 로그 파일은 삭제하는 것이 좋습니다.

버킷에 대한 자세한 내용은 객체 스토리지를 참조하세요.

목차

- 버전 관리를 사용하는 버킷에서 객체 삭제

- Lightsail 콘솔을 사용하여 객체 삭제
- Lightsail 콘솔을 사용하여 객체 버전 삭제
- 를 사용하여 단일 객체 또는 객체 버전 삭제 AWS CLI
- 를 사용하여 여러 객체 또는 객체 버전 삭제 AWS CLI

버전 관리를 사용하는 버킷에서 객체 삭제

버킷에 버전 관리를 사용하도록 설정한 경우 동일한 객체의 여러 버전이 버킷에 존재할 수 있습니다. Lightsail 콘솔, AWS CLI AWS APIs 또는 AWS SDKS를 사용하여 객체의 모든 버전을 삭제할 수 있습니다. 그러나 다음 옵션을 고려해야 합니다.

Lightsail 콘솔을 사용하여 객체 및 객체 버전 삭제

Lightsail 콘솔의 객체(Objects) 탭에 있는 객체 브라우저(Objects browser) 창에서 객체의 현재 버전을 삭제하면 이전 버전의 객체도 모두 삭제됩니다. 객체의 특정 버전은 버전 관리(Manage versions) 창에서 삭제해야 합니다. 버전 관리(Manage versions) 창을 사용하여 객체의 현재 버전을 삭제하면 기존의 가장 최신 버전이 현재 버전으로 복원됩니다. 자세한 내용은 가이드 후반부의 Lightsail 콘솔을 사용하여 객체 버전 삭제를 참조하세요.

Lightsail API AWS CLI, 또는 AWS SDKs를 사용하여 객체 및 객체 버전 삭제

단일 객체와 객체에 저장된 모든 버전을 삭제하려면 삭제 요청에 객체의 키만 지정하면 됩니다. 객체의 특정 버전을 삭제하려면 객체 키와 버전 ID를 모두 지정해야 합니다. 자세한 내용은 가이드 후반부의 AWS CLI를 사용하여 단일 객체 또는 객체 버전 삭제를 참조하세요.

Lightsail 콘솔을 사용하여 객체 삭제

Lightsail 콘솔을 사용하여 저장된 이전 버전을 포함하여 객체를 삭제하려면 다음 절차를 완료하세요. Lightsail 콘솔을 사용하면 한 번에 하나의 객체만 삭제할 수 있습니다. 를 사용하여 여러 객체 AWS CLI 를 한 번에 삭제합니다. 자세한 내용은 가이드 후반부의 AWS CLI를 사용하여 여러 객체 또는 객체 버전 삭제를 참조하세요.

1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 스토리지를 선택합니다.
3. 객체를 삭제할 버킷의 이름을 선택합니다.
4. 객체(Objects) 탭에서 객체 브라우저(Objects browser) 창을 사용하여 삭제할 객체의 위치를 찾습니다.

5. 삭제하려는 객체 옆에 확인 표시를 추가합니다.
6. 객체 정보(Object information) 창에서 작업(⋮) 메뉴를 선택한 후 삭제(Delete)를 선택합니다.
7. 확인 창이 표시되면 예, 삭제합니다(Yes, delete)를 선택하여 객체를 영구적으로 삭제할지 확인합니다.

   폴더에 있는 유일한 객체를 삭제할 경우 폴더도 함께 삭제됩니다. 이 문제는 폴더가 객체 키 이름의 일부이기 때문에 발생하며, 버킷에 동일한 객체 접두사를 공유하는 다른 객체가 없는 경우 객체를 삭제하면 이전 폴더도 삭제됩니다. 자세한 내용은 객체 스토리지 버킷의 키 이름을 참조하세요.

Lightsail 콘솔을 사용하여 객체 버전 삭제

객체의 저장된 버전을 삭제하려면 다음 절차를 완료하세요. 이 절차는 버전을 사용하도록 설정한 버킷에서만 적용됩니다. 자세한 내용은 버킷의 객체 버전 사용 설정 및 사용 중지를 참조하세요.

1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 스토리지를 선택합니다.
3. 객체를 삭제할 버킷의 이름을 선택합니다.
4. 객체 브라우저(Objects browser) 창을 사용하여 삭제하려는 객체의 위치를 찾습니다.
5. 저장된 이전 버전을 삭제할 객체 옆에 확인 표시를 추가합니다.
6. 객체 정보(Object information) 창의 버전(Versions) 섹션에서 관리(Manage)를 선택한 후 관리(Manage)를 선택합니다.
7. 저장된 객체 버전 관리 창이 표시되면 삭제할 객체 버전 옆에 확인 표시를 추가합니다.

   객체의 현재 버전을 삭제하도록 선택할 수도 있습니다.
8. 선택한 항목 삭제(Delete selected)를 선택하여 선택한 버전을 삭제합니다.

   삭제한 경우 다음과 같이 진행됩니다.

   - 현재 객체 버전 - 기존의 가장 최신 객체 버전이 현재 버전으로 복원됩니다.
   - 유일한 객체 버전 - 객체가 버킷에서 삭제됩니다. 삭제한 버전이 현재 폴더에 있는 유일한 객체인 경우 폴더도 삭제됩니다. 이 문제는 폴더가 객체 키 이름의 일부이기 때문에 발생하며, 버킷에 동일한 객체 키 접두사를 공유하는 다른 객체가 없는 경우 객체를 삭제하면 이전 폴더도 삭제됩니다. 자세한 내용은 버킷의 객체 버전 사용 설정 및 사용 중지를 참조하세요.

를 사용하여 단일 객체 또는 객체 버전 삭제 AWS CLI

AWS Command Line Interface ()를 사용하여 버킷의 단일 객체 또는 객체 버전을 삭제하려면 다음 절차를 완료하세요AWS CLI. `delete-object` 명령을 사용하여 이 작업을 수행할 수 있습니다. 자세한 내용은 AWS CLI 명령 레퍼런스의 delete-object를 참조하세요.

> **Note**
>
> 이 절차를 계속하기 전에 AWS CLI 및 Amazon S3에 대해 Lightsail를 설치하고 구성해야 합니다. 자세한 내용은 에서 작동 AWS Command Line Interface 하도록 구성을 Amazon Lightsail참조하세요.

1. 명령 프롬프트 또는 터미널 창을 엽니다.
2. 다음 명령을 입력하여 버킷의 객체 또는 객체 버전을 삭제합니다.

   객체를 삭제하려면:

   ```
   aws s3api delete-object --bucket BucketName --key ObjectKey
   ```

   객체 버전을 삭제하려면:

   > **Note**
   >
   > 버전을 사용하도록 설정한 버킷에서만 객체 버전을 삭제할 수 있습니다. 자세한 내용은 버킷의 객체 버전 사용 설정 및 사용 중지를 참조하세요.

   ```
   aws s3api delete-object --bucket BucketName --key ObjectKey --version-id VersionID
   ```

   명령에서 아래 예 텍스트를 사용자의 값으로 대체합니다.

   - *BucketName* - 객체를 삭제할 버킷의 이름입니다.
   - *ObjectKey* - 삭제하려는 객체의 전체 객체 키입니다.
   - *VersionID* - 삭제하려는 객체 버전의 ID입니다.

   예시:

객체 삭제:

```
aws s3api delete-object --bucket amzn-s3-demo-bucket --key images/sailbot.jpg
```

객체 버전 삭제:

```
aws s3api delete-object --bucket amzn-s3-demo-bucket --key images/sailbot.jpg --
version-id YF0YMBlUvexampleO07l2vJi9hRz4ujX
```

다음 예와 비슷한 결과가 나타나야 합니다.

```
C:\Users\latino>aws s3api delete-object --bucket DOC-EXAMPLE-BUCKET  --key  images/sailbot.jpg  --version-id YF0YMBlUvexampleO07l2vJi9hRz4ujX
{
    "VersionId": "YF0YMBexampleY7PO07l2vJi9hRz4ujX"
}
```

를 사용하여 여러 객체 또는 객체 버전 삭제 AWS CLI

AWS Command Line Interface (AWS CLI)를 사용하여 버킷의 여러 객체를 삭제하려면 다음 절차를 완료하세요. `delete-objects` 명령을 사용하여 이 작업을 수행할 수 있습니다. 자세한 내용은 AWS CLI 명령 참조의 delete-objects를 참조하세요.

> **Note**
>
> 이 절차를 계속하기 전에 AWS CLI 및 Amazon S3에 대해 Lightsail를 설치하고 구성해야 합니다. 자세한 내용은 에서 작동 AWS Command Line Interface 하도록 구성을 Amazon Lightsail참조하세요.

1. 명령 프롬프트 또는 터미널 창을 엽니다.
2. 다음 명령을 입력하여 버킷에 있는 여러 객체 또는 여러 객체 버전을 삭제합니다.

```
aws s3api delete-objects --bucket BucketName --delete file://LocalDirectory
```

명령에서 아래 예 텍스트를 사용자의 값으로 대체합니다.

- *BucketName* - 여러 객체 또는 여러 객체 버전을 삭제할 버킷의 이름입니다.
- *LocalDirectory* - 삭제할 객체 또는 버전을 지정하는 .json 문서에 대한 컴퓨터의 디렉터리 경로입니다. .json 문서는 다음과 같이 형식을 지정할 수 있습니다.

객체를 삭제하려면 .json 파일에 다음 텍스트를 입력하고 *ObjectKey*를 삭제할 객체의 객체 키로 바꿉니다.

```
{
  "Objects": [
    {
      "Key": "ObjectKey1"
    },
    {
      "Key": "ObjectKey2"
    }
  ],
  "Quiet": false
}
```

객체 버전을 삭제하려면 .json 파일에 다음 텍스트를 입력합니다. *ObjectKey* 및 *VersionID*를 삭제하려는 객체 키 및 객체 버전 ID와 바꿉니다.

> **Note**
>
> 버전을 사용하도록 설정한 버킷에서만 객체 버전을 삭제할 수 있습니다. 자세한 내용은 버킷의 객체 버전 사용 설정 및 사용 중지를 참조하세요.

```
{
  "Objects": [
    {
      "Key": "ObjectKey1",
      "VersionId": "VersionID1"
    },
    {
      "Key": "ObjectKey2",
      "VersionId": "VersionID2"
    }
  ],
  "Quiet": false
}
```

예시:

- Linux 또는 Unix 컴퓨터:

```
aws s3api delete-objects --bucket amzn-s3-demo-bucket --delete file://home/user/
Documents/delete-objects.json
```

- Windows 컴퓨터:

```
aws s3api delete-objects --bucket amzn-s3-demo-bucket --delete file://C:\Users
\user\Documents\delete-objects.json
```

다음 예와 비슷한 결과가 나타나야 합니다.

```
C:\>aws s3api delete-objects --bucket DOC-EXAMPLE-BUCKET  --delete file://C:\Users\user\Documents\delete-objects.json
{
    "Deleted": [
        {
            "Key": "images/sailbot.jpg",
            "VersionId": "26sqexampleztRiT6TsGhMMz0FxQAEw."
        },
        {
            "Key": "images/sailbot.jpg",
            "VersionId": "QwDrexampleDJxJtZC1CrExbpNlEC5O4"
        }
    ]
}
```

버킷 및 객체 관리

다음은 Lightsail 객체 스토리지 버킷을 관리하는 일반적인 단계입니다.

1. Amazon Lightsail 객체 스토리지 서비스의 객체와 버킷에 대해 알아보세요. 자세한 내용은 Amazon Lightsail의 객체 스토리지를 참조하세요.
2. Amazon Lightsail에서 버킷에 지정할 수 있는 이름에 대해 알아봅니다. 자세한 내용은 Amazon Lightsail의 버킷 이름 지정 규칙을 참조하세요.
3. 버킷을 생성하여 Lightsail 객체 스토리지 서비스를 시작합니다. 자세한 내용은 Amazon Lightsail에서 버킷 생성을 참조하세요.
4. 버킷의 보안 모범 사례와 버킷에 구성할 수 있는 액세스 권한에 대해 알아봅니다. 버킷의 모든 객체를 공개 또는 비공개로 설정하거나 개별 객체를 공개로 설정하도록 선택할 수 있습니다. 액세스 키를 생성하고, 버킷에 인스턴스를 연결하고, 다른 AWS 계정에 액세스 권한을 부여하여 버킷에 대한 액세스 권한을 부여할 수도 있습니다. 자세한 내용은 Amazon Lightsail 객체 스토리지에 대한 보안 모범 사례 및 Amazon Lightsail의 버킷 권한 이해를 참조하세요.

버킷 액세스 권한에 대해 알아본 후 다음 가이드를 참조하여 버킷에 대한 액세스 권한을 부여합니다.

- Amazon Lightsail의 버킷에 대한 퍼블릭 액세스 차단
- Amazon Lightsail에서 버킷 액세스 권한 구성
- Amazon Lightsail에서 버킷의 개별 객체에 대한 액세스 권한 구성
- Amazon Lightsail에서 버킷의 액세스 키 생성
- Amazon Lightsail에서 버킷에 대한 리소스 액세스 구성
- Amazon Lightsail에서 버킷의 크로스 계정 액세스 구성

5. 버킷에 대한 액세스 로깅을 활성화하는 방법과 액세스 로그를 사용하여 버킷의 보안을 감사하는 방법에 대해 알아봅니다. 자세한 내용은 다음 안내서를 참조하세요.
   - Amazon Lightsail 객체 스토리지 서비스의 버킷에 대한 액세스 로깅
   - Amazon Lightsail 객체 스토리지 서비스에서 버킷에 대한 액세스 로그 형식
   - Amazon Lightsail 객체 스토리지 서비스에서 버킷에 대한 액세스 로깅 활성화
   - Amazon Lightsail에서 버킷에 대한 액세스 로그 사용하여 요청을 식별
6. Lightsail에서 버킷을 관리할 수 있는 기능을 사용자에게 부여하는 IAM 정책을 생성합니다. 자세한 내용은 Amazon Lightsail에서 버킷을 관리하기 위한 IAM 정책을 참조하세요.
7. 버킷의 객체에 라벨을 지정하고 식별하는 방식에 대해 알아봅니다. 자세한 내용은 Amazon Lightsail에서 객체 키 이름 이해를 참조하세요.
8. 파일을 업로드하고 버킷의 객체를 관리하는 방법에 대해 알아봅니다. 자세한 내용은 다음 안내서를 참조하세요.
   - Amazon Lightsail에서 버킷으로 파일 업로드
   - 멀티파트 업로드를 사용하여 Amazon Lightsail에서 버킷으로 파일 업로드
   - Amazon Lightsail에서 버킷의 객체 확인
   - Amazon Lightsail에서 버킷의 객체 복사 또는 이동
   - Amazon Lightsail에서 버킷의 객체 다운로드
   - Amazon Lightsail에서 버킷의 객체 필터링
   - Amazon Lightsail에서 버킷의 객체 태그 지정
   - Amazon Lightsail에서 버킷의 객체 삭제
9. 객체 버전 관리를 활성화하면 버킷에 저장된 모든 버전의 객체를 모두 보존, 검색 및 복원할 수 있습니다. 자세한 내용은 Amazon Lightsail에서 버킷의 객체 버전 사용 설정 및 사용 중지를 참조하세요.

10.객체 버전 관리를 활성화한 후 버킷에 있는 객체의 이전 버전을 복원할 수 있습니다. 자세한 내용은 Amazon Lightsail에서 버킷의 이전 객체 버전 복원을 참조하세요.

11.버킷 사용률을 모니터링합니다. 자세한 내용은 Amazon Lightsail에서 버킷에 대한 지표 보기를 참조하세요.

12.버킷 사용률이 임곗값을 초과할 때 알림을 받도록 버킷 지표에 대한 경보를 구성합니다. 자세한 내용은 Amazon Lightsail에서 버킷 지표 경보 생성을 참조하세요.

13.스토리지 및 네트워크 전송이 부족한 경우 버킷의 스토리지 플랜을 변경합니다. 자세한 내용은 Amazon Lightsail에서 버킷 플랜 변경을 참조하세요.

14.버킷을 다른 리소스에 연결하는 방법에 대해 알아봅니다. 자세한 내용은 다음 자습서를 참조하세요.

- 자습서: WordPress 인스턴스를 Amazon Lightsail 버킷에 연결
- 자습서: Lightsail 콘텐츠 전송 네트워크 배포와 함께 Amazon Lightsail 버킷 사용

15.버킷을 더 이상 사용하지 않는 경우 삭제할 수 있습니다. 자세한 내용은 Amazon Lightsail에서 버킷 삭제를 참조하세요.

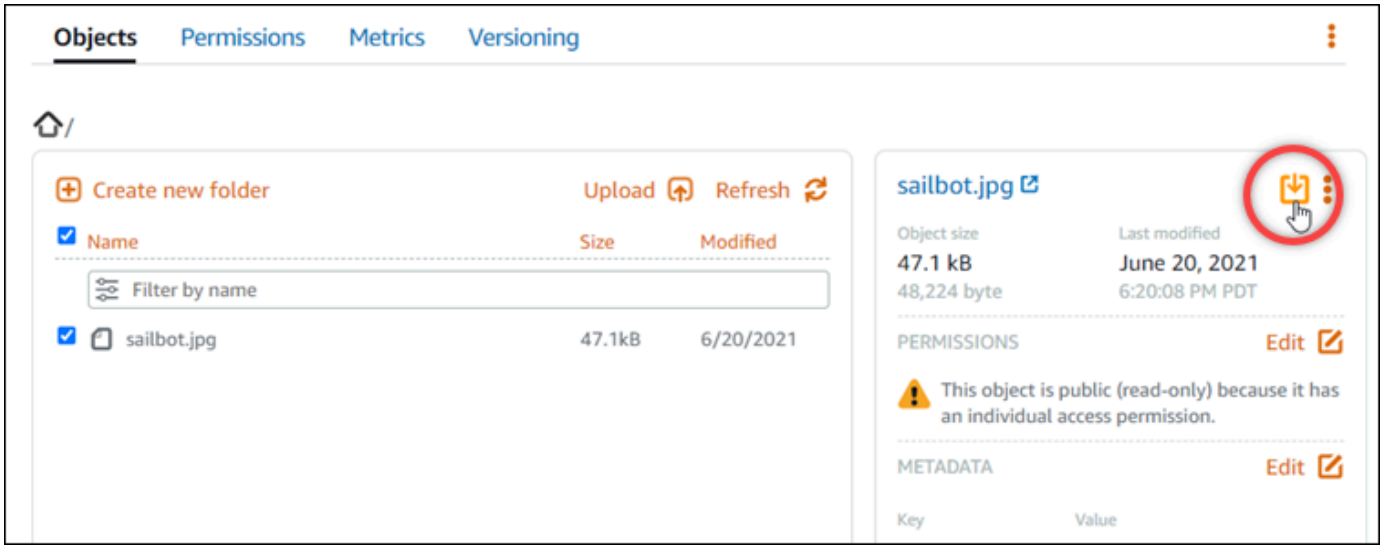
Lightsail 버킷의 객체 다운로드

Amazon Lightsail 객체 스토리지 서비스에서 공개(읽기 전용) 상태이거나 사용자에게 액세스 권한이 있는 버킷의 객체를 다운로드할 수 있습니다. Lightsail 콘솔을 사용하면 한 번에 하나의 객체를 다운로드할 수 있습니다. 한 요청으로 여러 객체를 다운로드하려면 AWS Command Line Interface (AWS CLI), AWS SDKs 또는 REST API를 사용합니다. 이 가이드에서는 Lightsail 콘솔 및 AWS CLI를 사용하여 객체를 다운로드하는 방법을 안내합니다. 버킷에 대한 자세한 내용은 객체 스토리지를 참조하세요.

Lightsail 콘솔을 사용한 객체 다운로드

Lightsail 콘솔을 사용하여 버킷의 객체를 다운로드하려면 다음 절차를 완료하세요.

1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 스토리지를 선택합니다.
3. 파일을 다운로드할 버킷의 이름을 선택합니다.
4. 객체(Objects) 탭에서 객체 브라우저 창(Objects browser pane)을 사용하여 다운로드할 객체의 위치를 찾습니다.
5. 다운로드하려는 객체 옆에 확인 표시를 추가합니다.
6. 객체 정보(Object information) 창에서 다운로드 아이콘을 선택합니다.



브라우저 구성에 따라 선택한 파일이 페이지에 표시되거나 컴퓨터에 다운로드됩니다. 파일이 페이지에 표시되면 마우스 오른쪽 버튼으로 클릭하고 다른 이름으로 저장(Save as)을 선택하여 컴퓨터에 저장할 수 있습니다.

를 사용하여 객체 다운로드 AWS CLI

AWS Command Line Interface (AWS CLI)를 사용하여 버킷에서 객체를 다운로드하려면 다음 절차를 완료하세요. `get-object` 명령을 사용하여 이 작업을 수행할 수 있습니다. 자세한 내용은 AWS CLI Command Reference의 get-object를 참조하세요.

> **Note**
>
> 이 절차를 계속하기 전에 AWS CLI 및 Amazon S3에 대해 Lightsail를 설치하고 구성해야 합니다. 자세한 내용은 에서 작동 AWS Command Line Interface 하도록 구성을 Amazon Lightsail참조하세요.

1. 명령 프롬프트 또는 터미널 창을 엽니다.
2. 다음 명령을 입력하여 버킷의 객체를 다운로드합니다.

```
aws s3api get-object --bucket BucketName --key ObjectKey LocalFilePath
```

명령에서 아래 예 텍스트를 사용자의 값으로 대체합니다.

- *BucketName* - 객체를 다운로드할 버킷의 이름입니다.

- *ObjectKey* - 다운로드하려는 객체의 전체 객체 키입니다.
- *LocalFilePath* - 다운로드한 파일을 저장할 컴퓨터의 전체 파일 경로입니다.

예제:

```
aws s3api get-object --bucket amzn-s3-demo-bucket --key images/sailbot.jpg C:\Users
\user\Pictures\sailbot.jpg
```

다음 예와 비슷한 결과가 나타나야 합니다.

```
C:\>aws s3api get-object --bucket DOC-EXAMPLE-BUCKET --key images/sailbot.jpg C:\Users\user\Pictures\sailbot.jpg
{
    "AcceptRanges": "bytes",
    "LastModified": "2021-05-10T05:09:31+00:00",
    "ContentLength": 48224,
    "ETag": "\"694d34example91d92d64f342aa234c3\"",
    "ContentType": "binary/octet-stream",
    "ServerSideEncryption": "AES256",
    "Metadata": {}
}
```

버킷 및 객체 관리

다음은 Lightsail 객체 스토리지 버킷을 관리하는 일반적인 단계입니다.

1. Amazon Lightsail 객체 스토리지 서비스의 객체와 버킷에 대해 알아보세요. 자세한 내용은 Amazon Lightsail의 객체 스토리지를 참조하세요.
2. Amazon Lightsail에서 버킷에 지정할 수 있는 이름에 대해 알아봅니다. 자세한 내용은 Amazon Lightsail의 버킷 이름 지정 규칙을 참조하세요.
3. 버킷을 생성하여 Lightsail 객체 스토리지 서비스를 시작합니다. 자세한 내용은 Amazon Lightsail에서 버킷 생성을 참조하세요.
4. 버킷의 보안 모범 사례와 버킷에 구성할 수 있는 액세스 권한에 대해 알아봅니다. 버킷의 모든 객체를 공개 또는 비공개로 설정하거나 개별 객체를 공개로 설정하도록 선택할 수 있습니다. 액세스 키를 생성하고, 버킷에 인스턴스를 연결하고, 다른 AWS 계정에 액세스 권한을 부여하여 버킷에 대한 액세스 권한을 부여할 수도 있습니다. 자세한 내용은 Amazon Lightsail 객체 스토리지에 대한 보안 모범 사례 및 Amazon Lightsail의 버킷 권한 이해를 참조하세요.

   버킷 액세스 권한에 대해 알아본 후 다음 가이드를 참조하여 버킷에 대한 액세스 권한을 부여합니다.

   - Amazon Lightsail의 버킷에 대한 퍼블릭 액세스 차단

- Amazon Lightsail에서 버킷 액세스 권한 구성
- Amazon Lightsail에서 버킷의 개별 객체에 대한 액세스 권한 구성
- Amazon Lightsail에서 버킷의 액세스 키 생성
- Amazon Lightsail에서 버킷에 대한 리소스 액세스 구성
- Amazon Lightsail에서 버킷의 크로스 계정 액세스 구성

5. 버킷에 대한 액세스 로깅을 활성화하는 방법과 액세스 로그를 사용하여 버킷의 보안을 감사하는 방법에 대해 알아봅니다. 자세한 내용은 다음 안내서를 참조하세요.
   - Amazon Lightsail 객체 스토리지 서비스의 버킷에 대한 액세스 로깅
   - Amazon Lightsail 객체 스토리지 서비스에서 버킷에 대한 액세스 로그 형식
   - Amazon Lightsail 객체 스토리지 서비스에서 버킷에 대한 액세스 로깅 활성화
   - Amazon Lightsail에서 버킷에 대한 액세스 로그 사용하여 요청을 식별
6. Lightsail에서 버킷을 관리할 수 있는 기능을 사용자에게 부여하는 IAM 정책을 생성합니다. 자세한 내용은 Amazon Lightsail에서 버킷을 관리하기 위한 IAM 정책을 참조하세요.
7. 버킷의 객체에 라벨을 지정하고 식별하는 방식에 대해 알아봅니다. 자세한 내용은 Amazon Lightsail에서 객체 키 이름 이해를 참조하세요.
8. 파일을 업로드하고 버킷의 객체를 관리하는 방법에 대해 알아봅니다. 자세한 내용은 다음 안내서를 참조하세요.
   - Amazon Lightsail에서 버킷으로 파일 업로드
   - 멀티파트 업로드를 사용하여 Amazon Lightsail에서 버킷으로 파일 업로드
   - Amazon Lightsail에서 버킷의 객체 확인
   - Amazon Lightsail에서 버킷의 객체 복사 또는 이동
   - Amazon Lightsail에서 버킷의 객체 다운로드
   - Amazon Lightsail에서 버킷의 객체 필터링
   - Amazon Lightsail에서 버킷의 객체 태그 지정
   - Amazon Lightsail에서 버킷의 객체 삭제
9. 객체 버전 관리를 활성화하면 버킷에 저장된 모든 버전의 객체를 모두 보존, 검색 및 복원할 수 있습니다. 자세한 내용은 Amazon Lightsail에서 버킷의 객체 버전 사용 설정 및 사용 중지를 참조하세요.
10. 객체 버전 관리를 활성화한 후 버킷에 있는 객체의 이전 버전을 복원할 수 있습니다. 자세한 내용은 Amazon Lightsail에서 버킷의 이전 객체 버전 복원을 참조하세요.
11. 버킷 사용률을 모니터링합니다. 자세한 내용은 Amazon Lightsail에서 버킷에 대한 지표 보기를 참조하세요.

12.버킷 사용률이 임곗값을 초과할 때 알림을 받도록 버킷 지표에 대한 경보를 구성합니다. 자세한 내용은 Amazon Lightsail에서 버킷 지표 경보 생성을 참조하세요.

13.스토리지 및 네트워크 전송이 부족한 경우 버킷의 스토리지 플랜을 변경합니다. 자세한 내용은 Amazon Lightsail에서 버킷 플랜 변경을 참조하세요.

14.버킷을 다른 리소스에 연결하는 방법에 대해 알아봅니다. 자세한 내용은 다음 자습서를 참조하세요.

- 자습서: WordPress 인스턴스를 Amazon Lightsail 버킷에 연결
- 자습서: Lightsail 콘텐츠 전송 네트워크 배포와 함께 Amazon Lightsail 버킷 사용

15.버킷을 더 이상 사용하지 않는 경우 삭제할 수 있습니다. 자세한 내용은 Amazon Lightsail에서 버킷 삭제를 참조하세요.

이름 접두사를 기준으로 Lightsail 버킷의 객체 필터링

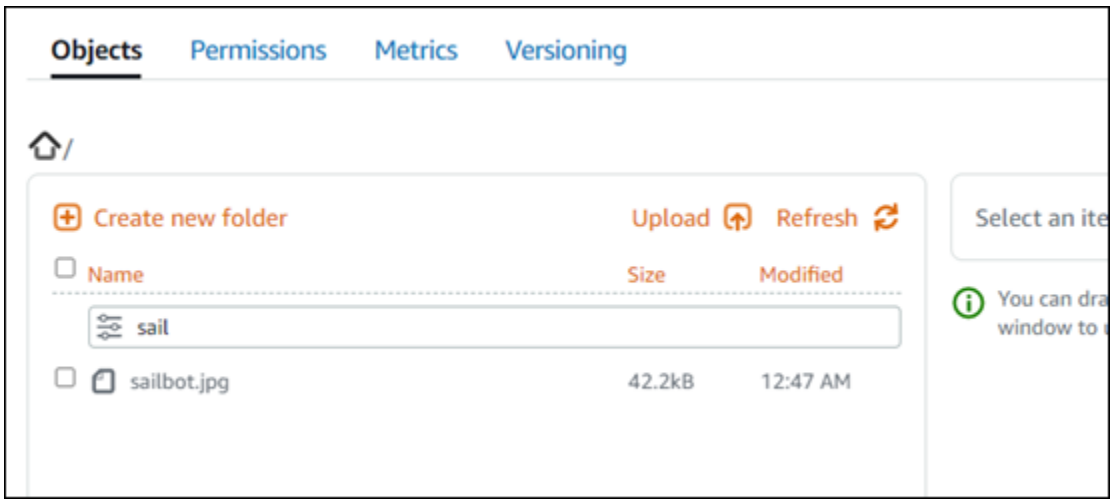
필터링을 사용하여 Amazon Lightsail 객체 스토리지 서비스에서 버킷의 객체를 찾을 수 있습니다. 이 설명서에서는 Lightsail 콘솔과 AWS Command Line Interface (AWS CLI)를 사용하여 객체를 필터링하는 방법을 안내합니다. 버킷에 대한 자세한 내용은 객체 스토리지를 참조하세요.

Lightsail 콘솔을 사용한 객체 필터링

Lightsail 콘솔을 사용하여 버킷의 객체를 필터링하려면 다음 절차를 완료하세요.

1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 스토리지를 선택합니다.
3. 객체를 찾으려는 버킷의 이름을 선택합니다.
4. 객체(Objects) 탭에서 이름으로 필터링(Filter by name) 텍스트 상자에 객체 접두사를 입력합니다.

   현재 보고 있는 폴더의 객체 목록이 입력한 텍스트와 일치하도록 필터링됩니다. 다음 예에서는 `sail`을 입력하면 페이지의 객체 목록이 `sail`로 시작하는 객체만 표시되도록 필터링됨을 보여줍니다.



다른 폴더의 객체 목록을 필터링하려면 해당 폴더로 이동합니다. 그런 다음 이름으로 필터링(Filter by name) 텍스트 상자에 객체 접두사를 입력합니다.

를 사용하여 객체 필터링 AWS CLI

AWS Command Line Interface (AWS CLI)를 사용하여 버킷의 객체를 필터링하려면 다음 절차를 완료하세요. `list-objects-v2` 명령을 사용하여 이 작업을 수행할 수 있습니다. 자세한 내용은 AWS CLI 명령 레퍼런스에서 list-objects-v2를 참조하세요.

> **Note**
>
> 이 절차를 계속하기 전에 AWS CLI 및 Amazon S3에 대해 Lightsail를 설치하고 구성해야 합니다. 자세한 내용은 에서 작동 AWS Command Line Interface 하도록 구성을 Amazon Lightsail참조하세요.

1. 명령 프롬프트 또는 터미널 창을 엽니다.
2. 다음 명령을 입력하여 특정 객체 키 이름 접두사로 시작하는 객체를 나열합니다.

```
aws s3api list-objects-v2 --bucket BucketName --prefix ObjectKeyNamePrefix --query
 "Contents[].{Key: Key, Size: Size}"
```

명령에서 아래 예 텍스트를 사용자의 값으로 대체합니다.

- *BucketName* - 모든 객체를 나열할 버킷의 이름입니다.

- *ObjectKeyNamePrefix* - 지정된 접두사로 시작하는 키로 응답을 제한할 객체 키 이름 접두사입니다.

> **Note**
>
> 이 명령은 `--query` 파라미터를 사용하여 각 객체의 키 값과 크기에 대한 `list-objects-v2` 요청 응답을 필터링합니다.

예제:

```
aws s3api list-objects-v2 --bucket amzn-s3-demo-bucket --prefix archived/ --query
 "Contents[].{Key: Key, Size: Size}"
```

다음 예제와 비슷한 결과가 나타나야 합니다.

버킷 및 객체 관리

다음은 Lightsail 객체 스토리지 버킷을 관리하는 일반적인 단계입니다.

1. Amazon Lightsail 객체 스토리지 서비스의 객체와 버킷에 대해 알아보세요. 자세한 내용은 Amazon Lightsail의 객체 스토리지를 참조하세요.
2. Amazon Lightsail에서 버킷에 지정할 수 있는 이름에 대해 알아봅니다. 자세한 내용은 Amazon Lightsail의 버킷 이름 지정 규칙을 참조하세요.

3. 버킷을 생성하여 Lightsail 객체 스토리지 서비스를 시작합니다. 자세한 내용은 Amazon Lightsail에서 버킷 생성을 참조하세요.
4. 버킷의 보안 모범 사례와 버킷에 구성할 수 있는 액세스 권한에 대해 알아봅니다. 버킷의 모든 객체를 공개 또는 비공개로 설정하거나 개별 객체를 공개로 설정하도록 선택할 수 있습니다. 액세스 키를 생성하고, 버킷에 인스턴스를 연결하고, 다른 AWS 계정에 액세스 권한을 부여하여 버킷에 대한 액세스 권한을 부여할 수도 있습니다. 자세한 내용은 Amazon Lightsail 객체 스토리지에 대한 보안 모범 사례 및 Amazon Lightsail의 버킷 권한 이해를 참조하세요.

   버킷 액세스 권한에 대해 알아본 후 다음 가이드를 참조하여 버킷에 대한 액세스 권한을 부여합니다.

   - Amazon Lightsail의 버킷에 대한 퍼블릭 액세스 차단
   - Amazon Lightsail에서 버킷 액세스 권한 구성
   - Amazon Lightsail에서 버킷의 개별 객체에 대한 액세스 권한 구성
   - Amazon Lightsail에서 버킷의 액세스 키 생성
   - Amazon Lightsail에서 버킷에 대한 리소스 액세스 구성
   - Amazon Lightsail에서 버킷의 크로스 계정 액세스 구성
5. 버킷에 대한 액세스 로깅을 활성화하는 방법과 액세스 로그를 사용하여 버킷의 보안을 감사하는 방법에 대해 알아봅니다. 자세한 내용은 다음 안내서를 참조하세요.
   - Amazon Lightsail 객체 스토리지 서비스의 버킷에 대한 액세스 로깅
   - Amazon Lightsail 객체 스토리지 서비스에서 버킷에 대한 액세스 로그 형식
   - Amazon Lightsail 객체 스토리지 서비스에서 버킷에 대한 액세스 로깅 활성화
   - Amazon Lightsail에서 버킷에 대한 액세스 로그 사용하여 요청을 식별
6. Lightsail에서 버킷을 관리할 수 있는 기능을 사용자에게 부여하는 IAM 정책을 생성합니다. 자세한 내용은 Amazon Lightsail에서 버킷을 관리하기 위한 IAM 정책을 참조하세요.
7. 버킷의 객체에 라벨을 지정하고 식별하는 방식에 대해 알아봅니다. 자세한 내용은 Amazon Lightsail에서 객체 키 이름 이해를 참조하세요.
8. 파일을 업로드하고 버킷의 객체를 관리하는 방법에 대해 알아봅니다. 자세한 내용은 다음 안내서를 참조하세요.
   - Amazon Lightsail에서 버킷으로 파일 업로드
   - 멀티파트 업로드를 사용하여 Amazon Lightsail에서 버킷으로 파일 업로드
   - Amazon Lightsail에서 버킷의 객체 확인
   - Amazon Lightsail에서 버킷의 객체 복사 또는 이동
   - Amazon Lightsail에서 버킷의 객체 다운로드

- Amazon Lightsail에서 버킷의 객체 필터링
- Amazon Lightsail에서 버킷의 객체 태그 지정
- Amazon Lightsail에서 버킷의 객체 삭제

9. 객체 버전 관리를 활성화하면 버킷에 저장된 모든 버전의 객체를 모두 보존, 검색 및 복원할 수 있습니다. 자세한 내용은 Amazon Lightsail에서 버킷의 객체 버전 사용 설정 및 사용 중지를 참조하세요.
10. 객체 버전 관리를 활성화한 후 버킷에 있는 객체의 이전 버전을 복원할 수 있습니다. 자세한 내용은 Amazon Lightsail에서 버킷의 이전 객체 버전 복원을 참조하세요.
11. 버킷 사용률을 모니터링합니다. 자세한 내용은 Amazon Lightsail에서 버킷에 대한 지표 보기를 참조하세요.
12. 버킷 사용률이 임곗값을 초과할 때 알림을 받도록 버킷 지표에 대한 경보를 구성합니다. 자세한 내용은 Amazon Lightsail에서 버킷 지표 경보 생성을 참조하세요.
13. 스토리지 및 네트워크 전송이 부족한 경우 버킷의 스토리지 플랜을 변경합니다. 자세한 내용은 Amazon Lightsail에서 버킷 플랜 변경을 참조하세요.
14. 버킷을 다른 리소스에 연결하는 방법에 대해 알아봅니다. 자세한 내용은 다음 자습서를 참조하세요.
    - 자습서: WordPress 인스턴스를 Amazon Lightsail 버킷에 연결
    - 자습서: Lightsail 콘텐츠 전송 네트워크 배포와 함께 Amazon Lightsail 버킷 사용
15. 버킷을 더 이상 사용하지 않는 경우 삭제할 수 있습니다. 자세한 내용은 Amazon Lightsail에서 버킷 삭제를 참조하세요.

Lightsail에서 객체 버전 관리 활성화 및 일시 중지

Amazon Lightsail 객체 스토리지 서비스의 버전 관리는 동일 버킷 내에 여러 개의 객체 변형을 보유하는 것을 의미합니다. 버전 관리를 사용하면 버킷에 저장된 모든 버전의 객체를 전부 보존, 검색 및 복원할 수 있습니다. 또한, 의도치 않은 사용자 작업 및 애플리케이션 장애가 발생해도 쉽게 복구할 수 있습니다. 버킷에 대해 버전 관리를 활성화하면 Lightsail 객체 스토리지 서비스가 동일한 객체에 대해 여러 쓰기 요청을 동시에 수신하는 경우에 모든 객체가 저장됩니다. Lightsail 객체 스토리지 서비스의 버킷에서는 버전 관리가 기본적으로 비활성화되므로, 명시적으로 활성화해야 합니다. 버킷에 대한 자세한 내용은 객체 스토리지를 참조하세요.

> **Important**
>
> 개별 객체 공개 가능(읽기 전용)(Individual objects can be made public (read-only)) 액세스 권한이 구성된 버킷에서 버전 관리를 활성화하거나 일시 중지하면 권한이 모든 객체 비공개(All objects are private)로 재설정됩니다. 개별 객체를 공개하는 옵션을 계속 사용하려면 버킷 액

세스 권한을 다시 개별 객체 공개 가능(읽기 전용)Individual objects can be made public (read-only)으로 직접 변경해야 합니다. 자세한 내용은 버킷 액세스 권한 구성을 참조하세요.

비활성화, 활성화 및 일시 중지된 버전의 버킷

버킷 버전 관리는 Lightsail 콘솔에서 다음 3가지 상태 중 하나로 설정됩니다.

- 비활성화됨(API 및 SDK에서 `NeverEnabled`)
- 활성화됨(API 및 SDK에서 `Enabled`)
- 일시 중지됨(API 및 SDK에서 `Suspended`)

버킷에서 버전 관리를 활성화한 후에는 비활성화 상태로 되돌릴 수 없습니다. 그러나 버전 관리를 일시 중지할 수는 있습니다. 버전 관리는 버킷 수준에서 사용 설정하고 일시 중지합니다.

버전 관리 상태는 해당 버킷의 모든 객체(일부 객체 제외)에 적용됩니다. 버킷에서 버전 관리를 사용 설정하면 모든 새 객체의 버전이 관리되고 고유한 버전 ID가 부여됩니다. 버전 관리를 활성화할 때 버킷에 이미 있던 객체는 항상 이후 버전으로 지정됩니다. 이후 요청을 통해 수정되는 경우 고유한 버전 ID를 부여받게 됩니다.

버전 ID

버킷에 대한 버전 관리를 활성화하면 Lightsail 객체 스토리지 서비스는 저장되는 객체에 대해 고유한 버전 ID를 자동으로 생성합니다. 예를 들어, `photo.gif`(버전 111111) 및 `photo.gif`(버전 121212)와 같이 하나의 버킷에서 키는 동일하지만 버전 ID가 다른 두 개의 객체를 보유할 수 있습니다.

버전 ID는 편집할 수 없습니다. 버전 ID는 유니코드, UTF-8 인코딩, URL 지원, 불투명 문자열이며 길이가 1,024바이트를 넘지 않습니다. 다음은 버전 ID의 예입니다.

```
3sL4kqtJlcpXroDTDmJ+rmSpXd3dIbrHY+MTRCxf3vjVBH40Nr8X8gdRQBpUMLUo
```

Lightsail 콘솔을 사용하여 객체 버전 관리 활성화 또는 일시 중지

Lightsail 콘솔을 사용하여 객체 버전 관리를 활성화하거나 일시 중지하려면 다음 절차를 완료하세요.

1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 스토리지를 선택합니다.
3. 버전 관리를 활성화하거나 일시 중지할 버킷의 이름을 선택합니다.
4. 버전 관리(Versioning) 탭을 선택합니다.
5. 버킷의 현재 버전 관리 상태에 따라 다음 작업 중 하나를 수행합니다.
   - 버전 관리가 현재 일시 중지되었거나 활성화되지 않은 경우 페이지의 객체 버전 관리(Object versioning) 섹션에서 토글 버튼을 선택하여 버전 관리를 활성화합니다.
   - 버전 관리가 현재 활성화된 경우 페이지의 객체 버전 관리(Object versioning) 섹션에서 토글 버튼을 선택하여 버전 관리를 일시 중지합니다.

를 사용하여 객체 버전 관리 활성화 또는 일시 중지 AWS CLI

AWS Command Line Interface (AWS CLI)를 사용하여 객체 버전 관리를 활성화하거나 일시 중지하려면 다음 절차를 완료하세요. `update-bucket` 명령을 사용하여 이 작업을 수행할 수 있습니다. 자세한 내용은 AWS CLI 명령 레퍼런스의 update-bucket을 참조하세요.

> **Note**
>
> 이 절차를 계속하기 전에 AWS CLI 및 Amazon S3에 대해 Lightsail를 설치하고 구성해야 합니다. 자세한 내용은 에서 작동 AWS CLI 하도록 구성을 Lightsail참조하세요.

1. 터미널(Terminal) 또는 명령 프롬프트(Command Prompt) 창을 엽니다.
2. 다음 명령을 입력하여 객체 버전 관리를 활성화하거나 일시 중지합니다.

   ```
   aws lightsail update-bucket --bucket-name BucketName --versioning VersioningState
   ```

   명령에서 아래 예 텍스트를 사용자의 값으로 대체합니다.

   - *BucketName* - 객체 버전 관리를 활성화할 버킷의 이름입니다.
   - *VersioningState* - 다음 중 하나입니다.
     - `Enabled` - 객체 버전 관리를 활성화합니다.

- Suspended - 이전에 활성화된 경우 객체 버전 관리를 일시 중지합니다.

예제:

```
aws lightsail update-bucket --bucket-name amzn-s3-demo-bucket --versioning Enabled
```

다음 예와 비슷한 결과가 나타나야 합니다.

```
C:\>aws lightsail update-bucket --bucket-name DOC-EXAMPLE-BUCKET --versioning Enabled
{
    "bucket": {
        "resourceType": "Bucket",
        "accessRules": {
            "getObject": "private",
            "allowPublicOverrides": false
        },
        "arn": "arn:aws:lightsail:us-west-2:1example7491:Bucket/f067383e-ee41-4485-b934-example2e2fd",
        "bundleId": "small_1_0",
        "createdAt": "2021-06-29T08:12:39.163000-07:00",
        "url": "https://DOC-EXAMPLE-BUCKET.s3.us-west-2.amazonaws.com",
        "location": {
            "availabilityZone": "all",
            "regionName": "us-west-2"
        },
        "name": "DOC-EXAMPLE-BUCKET",
        "supportCode": "621291663362/DOC-EXAMPLE-BUCKET/small_1_0",
        "tags": [],
        "objectVersioning": "Enabled",
        "ableToUpdateBundle": true
    },
    "operations": [
        {
            "id": "0d53d290-f4b2-43f0-89d2-example43448",
            "resourceName": "DOC-EXAMPLE-BUCKET",
            "resourceType": "Bucket",
            "createdAt": "2021-06-29T08:29:56.241000-07:00",
            "location": {
                "availabilityZone": "all",
                "regionName": "us-west-2"
            },
            "isTerminal": true,
            "operationDetails": "6example3362/DOC-EXAMPLE-BUCKET/small_1_0",
            "operationType": "UpdateBucket",
            "status": "Succeeded",
            "statusChangedAt": "2021-06-29T08:29:56.241000-07:00",
            "errorCode": "",
            "errorDetails": ""
        }
    ]
}
```

버킷 및 객체 관리

다음은 Lightsail 객체 스토리지 버킷을 관리하는 일반적인 단계입니다.

1. Amazon Lightsail 객체 스토리지 서비스의 객체와 버킷에 대해 알아보세요. 자세한 내용은 Amazon Lightsail의 객체 스토리지를 참조하세요.
2. Amazon Lightsail에서 버킷에 지정할 수 있는 이름에 대해 알아봅니다. 자세한 내용은 Amazon Lightsail의 버킷 이름 지정 규칙을 참조하세요.
3. 버킷을 생성하여 Lightsail 객체 스토리지 서비스를 시작합니다. 자세한 내용은 Amazon Lightsail에서 버킷 생성을 참조하세요.
4. 버킷의 보안 모범 사례와 버킷에 구성할 수 있는 액세스 권한에 대해 알아봅니다. 버킷의 모든 객체를 공개 또는 비공개로 설정하거나 개별 객체를 공개로 설정하도록 선택할 수 있습니다. 액세스 키를 생성하고, 버킷에 인스턴스를 연결하고, 다른 AWS 계정에 액세스 권한을 부여하여 버킷에 대한 액세스 권한을 부여할 수도 있습니다. 자세한 내용은 Amazon Lightsail 객체 스토리지에 대한 보안 모범 사례 및 Amazon Lightsail의 버킷 권한 이해를 참조하세요.

   버킷 액세스 권한에 대해 알아본 후 다음 가이드를 참조하여 버킷에 대한 액세스 권한을 부여합니다.

   - Amazon Lightsail의 버킷에 대한 퍼블릭 액세스 차단
   - Amazon Lightsail에서 버킷 액세스 권한 구성
   - Amazon Lightsail에서 버킷의 개별 객체에 대한 액세스 권한 구성
   - Amazon Lightsail에서 버킷의 액세스 키 생성
   - Amazon Lightsail에서 버킷에 대한 리소스 액세스 구성
   - Amazon Lightsail에서 버킷의 크로스 계정 액세스 구성
5. 버킷에 대한 액세스 로깅을 활성화하는 방법과 액세스 로그를 사용하여 버킷의 보안을 감사하는 방법에 대해 알아봅니다. 자세한 내용은 다음 안내서를 참조하세요.
   - Amazon Lightsail 객체 스토리지 서비스의 버킷에 대한 액세스 로깅
   - Amazon Lightsail 객체 스토리지 서비스에서 버킷에 대한 액세스 로그 형식
   - Amazon Lightsail 객체 스토리지 서비스에서 버킷에 대한 액세스 로깅 활성화
   - Amazon Lightsail에서 버킷에 대한 액세스 로그 사용하여 요청을 식별
6. Lightsail에서 버킷을 관리할 수 있는 기능을 사용자에게 부여하는 IAM 정책을 생성합니다. 자세한 내용은 Amazon Lightsail에서 버킷을 관리하기 위한 IAM 정책을 참조하세요.
7. 버킷의 객체에 라벨을 지정하고 식별하는 방식에 대해 알아봅니다. 자세한 내용은 Amazon Lightsail에서 객체 키 이름 이해를 참조하세요.
8. 파일을 업로드하고 버킷의 객체를 관리하는 방법에 대해 알아봅니다. 자세한 내용은 다음 안내서를 참조하세요.
   - Amazon Lightsail에서 버킷으로 파일 업로드

- 멀티파트 업로드를 사용하여 Amazon Lightsail에서 버킷으로 파일 업로드
- Amazon Lightsail에서 버킷의 객체 확인
- Amazon Lightsail에서 버킷의 객체 복사 또는 이동
- Amazon Lightsail에서 버킷의 객체 다운로드
- Amazon Lightsail에서 버킷의 객체 필터링
- Amazon Lightsail에서 버킷의 객체 태그 지정
- Amazon Lightsail에서 버킷의 객체 삭제

9. 객체 버전 관리를 활성화하면 버킷에 저장된 모든 버전의 객체를 모두 보존, 검색 및 복원할 수 있습니다. 자세한 내용은 Amazon Lightsail에서 버킷의 객체 버전 사용 설정 및 사용 중지를 참조하세요.
10. 객체 버전 관리를 활성화한 후 버킷에 있는 객체의 이전 버전을 복원할 수 있습니다. 자세한 내용은 Amazon Lightsail에서 버킷의 이전 객체 버전 복원을 참조하세요.
11. 버킷 사용률을 모니터링합니다. 자세한 내용은 Amazon Lightsail에서 버킷에 대한 지표 보기를 참조하세요.
12. 버킷 사용률이 임곗값을 초과할 때 알림을 받도록 버킷 지표에 대한 경보를 구성합니다. 자세한 내용은 Amazon Lightsail에서 버킷 지표 경보 생성을 참조하세요.
13. 스토리지 및 네트워크 전송이 부족한 경우 버킷의 스토리지 플랜을 변경합니다. 자세한 내용은 Amazon Lightsail에서 버킷 플랜 변경을 참조하세요.
14. 버킷을 다른 리소스에 연결하는 방법에 대해 알아봅니다. 자세한 내용은 다음 자습서를 참조하세요.
    - 자습서: WordPress 인스턴스를 Amazon Lightsail 버킷에 연결
    - 자습서: Lightsail 콘텐츠 전송 네트워크 배포와 함께 Amazon Lightsail 버킷 사용
15. 버킷을 더 이상 사용하지 않는 경우 삭제할 수 있습니다. 자세한 내용은 Amazon Lightsail에서 버킷 삭제를 참조하세요.

Lightsail 버킷에서 이전 객체 버전 복구

Amazon Lightsail 객체 스토리지 서비스의 버킷에 버전이 활성화된 경우 이전 객체 버전을 복원할 수 있습니다. 의도하지 않은 사용자 작업 또는 애플리케이션 장애로부터 복구된 이전 객체 버전을 복원합니다.

Lightsail 콘솔을 사용하여 객체의 이전 버전을 복원할 수 있습니다. AWS Command Line Interface (AWS CLI) 및 AWS SDKs를 사용하여 객체의 이전 버전을 복원할 수도 있습니다. 복원하려면 객체의 특정 버전을 동일한 버킷에 복사하고 같은 객체 키 이름을 사용하면 됩니다. 이렇게 하면 현재 버전이 이전 버전으로 대체되어 이전 버전이 현재 버전이 됩니다. 버전 관리에 대한 자세한 내용은 버킷의 객

체 버전 관리 사용 설정 및 사용 중지를 참조하세요. 버킷에 대한 자세한 내용은 객체 스토리지를 참조하세요.

Lightsail 콘솔을 사용하여 이전 객체 버전 복원

Lightsail 콘솔을 사용하여 이전 객체 버전을 복원하려면 다음 절차를 완료하세요.

1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 스토리지를 선택합니다.
3. 이전 객체 버전을 복원할 버킷 이름을 선택합니다.
4. 객체(Objects) 탭에서 객체 브라우저(Objects browser) 창을 사용하여 객체의 위치를 찾습니다.
5. 이전 버전을 복원할 객체 옆에 확인 표시를 추가합니다.
6. 객체 정보(Object information) 창의 버전(Versions) 섹션에서 관리(Manage)를 선택합니다.
7. 복원을 선택합니다.
8. 저장된 버전 창에 객체 복원(Restore object)이 표시되면 복원할 객체의 버전을 선택합니다.
9. 계속을 선택합니다.
10. 확인 프롬프트가 표시되면 예, 복원합니다(Yes, restore)를 선택하여 객체 버전을 복원합니다. 복원하지 않으려면 아니요, 취소합니다(No, cancel)를 선택합니다.

를 사용하여 객체의 이전 버전 복원 AWS CLI

AWS Command Line Interface (AWS CLI)를 사용하여 이전 객체 버전을 복원하려면 다음 절차를 완료하세요. `copy-object` 명령을 사용하여 다운로드할 수 있습니다. 동일한 객체 키를 사용하여 이전 객체 버전을 같은 버킷에 복사해야 합니다. 자세한 내용은 AWS CLI 명령 레퍼런스의 copy-object를 참조하세요.

> **Note**
>
> 이 절차를 계속하기 전에 AWS CLI 및 Amazon S3에 대해 Lightsail를 설치하고 구성해야 합니다. 자세한 내용은 에서 작동 AWS Command Line Interface 하도록 구성을 Amazon Lightsail참조하세요.

1. 명령 프롬프트 또는 터미널 창을 엽니다.
2. 다음 명령을 입력하여 이전 객체 버전을 복원합니다.

```
aws s3api copy-object --copy-source "BucketName/ObjectKey?versionId=VersionId" --
key ObjectKey --bucket BucketName
```

명령에서 아래 예 텍스트를 사용자의 값으로 대체합니다.

- *BucketName* - 이전 객체 버전을 복원할 버킷 이름입니다. `--copy-source` 및 `--bucket` 파라미터에 대해 동일한 버킷 이름을 지정해야 합니다.
- *ObjectKey* - 복원할 객체의 이름입니다. `--copy-source` 및 `--key` 파라미터에 대해 동일한 객체 키 이름을 지정해야 합니다.
- *VersionId* - 현재 버전으로 복원할 이전 객체 버전의 ID입니다. `list-object-versions` 명령을 사용하여 버킷에 있는 객체의 버전 ID 목록을 가져옵니다.

예제:

```
aws s3api copy-object --copy-source "amzn-s3-demo-bucket/sailbot.jpg?
versionId=GQWEexample87Mdl8Q_DKdVTiVMi_VyU" -key sailbot.jpg --bucket amzn-s3-demo-
bucket
```

다음 예와 비슷한 결과가 나타나야 합니다.

```
C:\>aws s3api copy-object --copy-source "DOC-EXAMPLE-BUCKET/sailbot.jpg?versionId=GQWEexample87Mdl8Q_DKdVTiVMi_VyU"
--key sailbot.jpg --bucket DOC-EXAMPLE-BUCKET
{
    "CopySourceVersionId": "GQWEcouyrfexampleQ_DKdVTiVMi_VyU",
    "VersionId": "hjL8anKzI1xcXYyexampleDvvqMXSLoi",
    "ServerSideEncryption": "AES256",
    "CopyObjectResult": {
        "ETag": "\"dc5afd388fb3example20cda3fe41c54\"",
        "LastModified": "2021-05-16T06:45:35+00:00"
    }
}
```

버킷 및 객체 관리

다음은 Lightsail 객체 스토리지 버킷을 관리하는 일반적인 단계입니다.

1. Amazon Lightsail 객체 스토리지 서비스의 객체와 버킷에 대해 알아보세요. 자세한 내용은 Amazon Lightsail의 객체 스토리지를 참조하세요.
2. Amazon Lightsail에서 버킷에 지정할 수 있는 이름에 대해 알아봅니다. 자세한 내용은 Amazon Lightsail의 버킷 이름 지정 규칙을 참조하세요.

3. 버킷을 생성하여 Lightsail 객체 스토리지 서비스를 시작합니다. 자세한 내용은 Amazon Lightsail에서 버킷 생성을 참조하세요.
4. 버킷의 보안 모범 사례와 버킷에 구성할 수 있는 액세스 권한에 대해 알아봅니다. 버킷의 모든 객체를 공개 또는 비공개로 설정하거나 개별 객체를 공개로 설정하도록 선택할 수 있습니다. 액세스 키를 생성하고, 버킷에 인스턴스를 연결하고, 다른 AWS 계정에 액세스 권한을 부여하여 버킷에 대한 액세스 권한을 부여할 수도 있습니다. 자세한 내용은 Amazon Lightsail 객체 스토리지에 대한 보안 모범 사례 및 Amazon Lightsail의 버킷 권한 이해를 참조하세요.

   버킷 액세스 권한에 대해 알아본 후 다음 가이드를 참조하여 버킷에 대한 액세스 권한을 부여합니다.

   - Amazon Lightsail의 버킷에 대한 퍼블릭 액세스 차단
   - Amazon Lightsail에서 버킷 액세스 권한 구성
   - Amazon Lightsail에서 버킷의 개별 객체에 대한 액세스 권한 구성
   - Amazon Lightsail에서 버킷의 액세스 키 생성
   - Amazon Lightsail에서 버킷에 대한 리소스 액세스 구성
   - Amazon Lightsail에서 버킷의 크로스 계정 액세스 구성
5. 버킷에 대한 액세스 로깅을 활성화하는 방법과 액세스 로그를 사용하여 버킷의 보안을 감사하는 방법에 대해 알아봅니다. 자세한 내용은 다음 안내서를 참조하세요.
   - Amazon Lightsail 객체 스토리지 서비스의 버킷에 대한 액세스 로깅
   - Amazon Lightsail 객체 스토리지 서비스에서 버킷에 대한 액세스 로그 형식
   - Amazon Lightsail 객체 스토리지 서비스에서 버킷에 대한 액세스 로깅 활성화
   - Amazon Lightsail에서 버킷에 대한 액세스 로그 사용하여 요청을 식별
6. Lightsail에서 버킷을 관리할 수 있는 기능을 사용자에게 부여하는 IAM 정책을 생성합니다. 자세한 내용은 Amazon Lightsail에서 버킷을 관리하기 위한 IAM 정책을 참조하세요.
7. 버킷의 객체에 라벨을 지정하고 식별하는 방식에 대해 알아봅니다. 자세한 내용은 Amazon Lightsail에서 객체 키 이름 이해를 참조하세요.
8. 파일을 업로드하고 버킷의 객체를 관리하는 방법에 대해 알아봅니다. 자세한 내용은 다음 안내서를 참조하세요.
   - Amazon Lightsail에서 버킷으로 파일 업로드
   - 멀티파트 업로드를 사용하여 Amazon Lightsail에서 버킷으로 파일 업로드
   - Amazon Lightsail에서 버킷의 객체 확인
   - Amazon Lightsail에서 버킷의 객체 복사 또는 이동
   - Amazon Lightsail에서 버킷의 객체 다운로드

- Amazon Lightsail에서 버킷의 객체 필터링
- Amazon Lightsail에서 버킷의 객체 태그 지정
- Amazon Lightsail에서 버킷의 객체 삭제

9. 객체 버전 관리를 활성화하면 버킷에 저장된 모든 버전의 객체를 모두 보존, 검색 및 복원할 수 있습니다. 자세한 내용은 Amazon Lightsail에서 버킷의 객체 버전 사용 설정 및 사용 중지를 참조하세요.
10. 객체 버전 관리를 활성화한 후 버킷에 있는 객체의 이전 버전을 복원할 수 있습니다. 자세한 내용은 Amazon Lightsail에서 버킷의 이전 객체 버전 복원을 참조하세요.
11. 버킷 사용률을 모니터링합니다. 자세한 내용은 Amazon Lightsail에서 버킷에 대한 지표 보기를 참조하세요.
12. 버킷 사용률이 임곗값을 초과할 때 알림을 받도록 버킷 지표에 대한 경보를 구성합니다. 자세한 내용은 Amazon Lightsail에서 버킷 지표 경보 생성을 참조하세요.
13. 스토리지 및 네트워크 전송이 부족한 경우 버킷의 스토리지 플랜을 변경합니다. 자세한 내용은 Amazon Lightsail에서 버킷 플랜 변경을 참조하세요.
14. 버킷을 다른 리소스에 연결하는 방법에 대해 알아봅니다. 자세한 내용은 다음 자습서를 참조하세요.
    - 자습서: WordPress 인스턴스를 Amazon Lightsail 버킷에 연결
    - 자습서: Lightsail 콘텐츠 전송 네트워크 배포와 함께 Amazon Lightsail 버킷 사용
15. 버킷을 더 이상 사용하지 않는 경우 삭제할 수 있습니다. 자세한 내용은 Amazon Lightsail에서 버킷 삭제를 참조하세요.

Lightsail 버킷의 객체에 태그 지정

버킷의 객체에 태그를 지정하여 용도, 소유자, 환경 또는 기타 기준에 따라 객체를 분류할 수 있습니다. 객체를 업로드할 때 또는 업로드한 후에 태그를 추가할 수 있습니다. 버킷에 대한 자세한 내용은 객체 스토리지를 참조하세요.

Lightsail 콘솔을 사용하여 객체 태그 추가 및 삭제

Lightsail 콘솔을 사용하여 버킷의 객체에 태그를 추가하거나 객체에서 태그를 삭제하려면 다음 절차를 완료하세요.

1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 스토리지를 선택합니다.
3. 객체에 태그를 지정하려는 버킷의 이름을 선택합니다.
4. 객체(Objects) 탭에서 객체 브라우저(Objects browser) 창을 사용하여 객체의 위치를 찾습니다.

5. 태그를 추가하거나 삭제할 객체 옆에 확인 표시를 추가합니다.
6. 객체 정보 창의 객체 태그(Object tags) 섹션에서

   - 추가(Add) 또는 편집(Edit)(태그가 이미 추가된 경우) 옵션 중 하나를 선택합니다. 키(Key) 텍스트 상자에 키를 입력하고 값(Value) 텍스트 상자에 값을 입력합니다. 그런 다음 저장(Save)을 선택하여 태그를 추가합니다. 그렇지 않은 경우 취소를 선택합니다.
   - 편집한 다음 삭제할 키 값 태그 옆의 X를 선택합니다. 태그 삭제를 완료했다면 저장(Save)을 선택하고 삭제하지 않으려면 취소(Cancel)를 선택합니다.

를 사용하여 객체에 대한 태그 추가 및 삭제 AWS CLI

AWS Command Line Interface ()를 사용하여 객체에 태그를 추가하거나 객체에서 태그를 삭제하려면 다음 절차를 완료하세요AWS CLI. `put-object-tagging` 및 `delete-object-tagging` 명령을 사용하여 이 작업을 수행할 수 있습니다. 자세한 내용은 AWS CLI 명령 레퍼런스에서 put-object-tagging 및 delete-object-tagging을 참조하세요.

> **Note**
>
> 이 절차를 계속하기 전에 AWS CLI 및 Amazon S3에 대해 Lightsail를 설치하고 구성해야 합니다. 자세한 내용은 에서 작동 AWS CLI 하도록 구성을 Lightsail참조하세요.

1. 명령 프롬프트 또는 터미널 창을 엽니다.
2. 다음 명령 중 하나를 입력합니다.

   - 객체에 태그를 추가하려면 다음 단계를 따르세요.

     ```
     aws s3api put-object-tagging --bucket BucketName --key ObjectKey --tagging
      "{\"TagSet\":[{ \"Key\": \"KeyTag\", \"Value\": \"ValueTag\" }]}"
     ```

     명령에서 아래 예 텍스트를 사용자의 값으로 대체합니다.

     - *BucketName* - 태그를 지정하려는 객체가 포함된 버킷의 이름입니다.
     - *ObjectKey* - 태그를 지정하려는 객체의 전체 객체 키입니다.
     - *KeyTag* - 태그의 키 값입니다.
     - *ValueTag* - 태그의 값입니다.
   - 객체에 태그를 추가하려면 다음 단계를 따르세요.

```
aws s3api put-object-tagging --bucket BucketName --key ObjectKey --tagging
 "{\"TagSet\":[{ \"Key\": \"KeyTag1\", \"Value\": \"ValueTag1\" }, { \"Key\":
 \"KeyTag2\", \"Value\": \"ValueTag2\" }]}"
```

명령에서 아래 예 텍스트를 사용자의 값으로 대체합니다.

- *BucketName* - 태그를 지정하려는 객체가 포함된 버킷의 이름입니다.
- *ObjectKey* - 태그를 지정하려는 객체의 전체 객체 키입니다.
- *KeyTag1* - 첫 번째 태그의 키 값입니다.
- *ValueTag1* - 첫 번째 태그의 값입니다.
- *KeyTag2* - 두 번째 태그의 키 값입니다.
- *ValueTag2* - 두 번째 태그의 값입니다.

- 객체에서 모든 태그를 삭제하려면 다음 단계를 따르세요.

```
aws s3api delete-object-tagging --bucket BucketName --key ObjectKey
```

명령에서 아래 예 텍스트를 사용자의 값으로 대체합니다.

- *BucketName* - 모든 태그를 삭제하려는 객체가 포함된 버킷의 이름입니다.
- *ObjectKey* - 태그를 지정하려는 객체의 전체 객체 키입니다.

예제:

```
aws s3api delete-object --bucket amzn-s3-demo-bucket --key nptLmg6jqDo.jpg --
tagging "{\"TagSet\":[{ \"Key\": \"Importance\", \"Value\": \"High\" }]}"
```

다음 예와 비슷한 결과가 나타나야 합니다.

```
C:\>aws s3api put-object-tagging --bucket DOC-EXAMPLE-BUCKET  --key nptLmg6jqDo.jpg
--tagging "{\"TagSet\":[{ \"Key\": \"Importance\", \"Value\": \"High\" }]}"
{
    "VersionId": "9nL2d41NuZdhdk4HS3kZIwOxJeS1kCkm"
}
```

버킷 및 객체 관리

다음은 Lightsail 객체 스토리지 버킷을 관리하는 일반적인 단계입니다.

1. Amazon Lightsail 객체 스토리지 서비스의 객체와 버킷에 대해 알아보세요. 자세한 내용은 Amazon Lightsail의 객체 스토리지를 참조하세요.
2. Amazon Lightsail에서 버킷에 지정할 수 있는 이름에 대해 알아봅니다. 자세한 내용은 Amazon Lightsail의 버킷 이름 지정 규칙을 참조하세요.
3. 버킷을 생성하여 Lightsail 객체 스토리지 서비스를 시작합니다. 자세한 내용은 Amazon Lightsail에서 버킷 생성을 참조하세요.
4. 버킷의 보안 모범 사례와 버킷에 구성할 수 있는 액세스 권한에 대해 알아봅니다. 버킷의 모든 객체를 공개 또는 비공개로 설정하거나 개별 객체를 공개로 설정하도록 선택할 수 있습니다. 액세스 키를 생성하고, 버킷에 인스턴스를 연결하고, 다른 AWS 계정에 액세스 권한을 부여하여 버킷에 대한 액세스 권한을 부여할 수도 있습니다. 자세한 내용은 Amazon Lightsail 객체 스토리지에 대한 보안 모범 사례 및 Amazon Lightsail의 버킷 권한 이해를 참조하세요.

   버킷 액세스 권한에 대해 알아본 후 다음 가이드를 참조하여 버킷에 대한 액세스 권한을 부여합니다.

   - Amazon Lightsail의 버킷에 대한 퍼블릭 액세스 차단
   - Amazon Lightsail에서 버킷 액세스 권한 구성
   - Amazon Lightsail에서 버킷의 개별 객체에 대한 액세스 권한 구성
   - Amazon Lightsail에서 버킷의 액세스 키 생성
   - Amazon Lightsail에서 버킷에 대한 리소스 액세스 구성
   - Amazon Lightsail에서 버킷의 크로스 계정 액세스 구성
5. 버킷에 대한 액세스 로깅을 활성화하는 방법과 액세스 로그를 사용하여 버킷의 보안을 감사하는 방법에 대해 알아봅니다. 자세한 내용은 다음 안내서를 참조하세요.

   - Amazon Lightsail 객체 스토리지 서비스의 버킷에 대한 액세스 로깅
   - Amazon Lightsail 객체 스토리지 서비스에서 버킷에 대한 액세스 로그 형식
   - Amazon Lightsail 객체 스토리지 서비스에서 버킷에 대한 액세스 로깅 활성화
   - Amazon Lightsail에서 버킷에 대한 액세스 로그 사용하여 요청을 식별
6. Lightsail에서 버킷을 관리할 수 있는 기능을 사용자에게 부여하는 IAM 정책을 생성합니다. 자세한 내용은 Amazon Lightsail에서 버킷을 관리하기 위한 IAM 정책을 참조하세요.
7. 버킷의 객체에 라벨을 지정하고 식별하는 방식에 대해 알아봅니다. 자세한 내용은 Amazon Lightsail에서 객체 키 이름 이해를 참조하세요.
8. 파일을 업로드하고 버킷의 객체를 관리하는 방법에 대해 알아봅니다. 자세한 내용은 다음 안내서를 참조하세요.

   - Amazon Lightsail에서 버킷으로 파일 업로드

- 멀티파트 업로드를 사용하여 Amazon Lightsail에서 버킷으로 파일 업로드
- Amazon Lightsail에서 버킷의 객체 확인
- Amazon Lightsail에서 버킷의 객체 복사 또는 이동
- Amazon Lightsail에서 버킷의 객체 다운로드
- Amazon Lightsail에서 버킷의 객체 필터링
- Amazon Lightsail에서 버킷의 객체 태그 지정
- Amazon Lightsail에서 버킷의 객체 삭제

9. 객체 버전 관리를 활성화하면 버킷에 저장된 모든 버전의 객체를 모두 보존, 검색 및 복원할 수 있습니다. 자세한 내용은 Amazon Lightsail에서 버킷의 객체 버전 사용 설정 및 사용 중지를 참조하세요.
10. 객체 버전 관리를 활성화한 후 버킷에 있는 객체의 이전 버전을 복원할 수 있습니다. 자세한 내용은 Amazon Lightsail에서 버킷의 이전 객체 버전 복원을 참조하세요.
11. 버킷 사용률을 모니터링합니다. 자세한 내용은 Amazon Lightsail에서 버킷에 대한 지표 보기를 참조하세요.
12. 버킷 사용률이 임곗값을 초과할 때 알림을 받도록 버킷 지표에 대한 경보를 구성합니다. 자세한 내용은 Amazon Lightsail에서 버킷 지표 경보 생성을 참조하세요.
13. 스토리지 및 네트워크 전송이 부족한 경우 버킷의 스토리지 플랜을 변경합니다. 자세한 내용은 Amazon Lightsail에서 버킷 플랜 변경을 참조하세요.
14. 버킷을 다른 리소스에 연결하는 방법에 대해 알아봅니다. 자세한 내용은 다음 자습서를 참조하세요.
    - 자습서: WordPress 인스턴스를 Amazon Lightsail 버킷에 연결
    - 자습서: Lightsail 콘텐츠 전송 네트워크 배포와 함께 Amazon Lightsail 버킷 사용
15. 버킷을 더 이상 사용하지 않는 경우 삭제할 수 있습니다. 자세한 내용은 Amazon Lightsail에서 버킷 삭제를 참조하세요.

인스턴스용 Lightsail 버킷에 대한 액세스 제어

Amazon Lightsail 인스턴스를 Lightsail 버킷에 연결하여 버킷과 해당 객체에 대한 프로그래밍 방식의 모든 액세스 권한을 제공할 수 있습니다. 인스턴스를 버킷에 연결하면 액세스 키와 같은 자격 증명을 관리할 필요가 없습니다. 연결하는 인스턴스와 버킷이 동일한 AWS 리전에 있어야 합니다. 다른 리전에 속하는 버킷에는 인스턴스를 연결할 수 없습니다.

리소스 액세스는 버킷에 파일을 직접 업로드하도록 인스턴스에 소프트웨어 또는 플러그 인을 구성하는 경우에 유용합니다. 예를 들어, 버킷에 미디어 파일을 저장하도록 WordPress 인스턴스를 구성할 수 있습니다. 자세한 내용은 자습서: WordPress 인스턴스에 버킷 연결을 참조하세요.

권한 옵션에 대한 자세한 내용은 버킷 권한을 참조하세요. 보안 모범 사례에 대한 자세한 내용은 객체 스토리지에 대한 보안 모범 사례를 참조하세요. 버킷에 대한 자세한 내용은 객체 스토리지를 참조하세요.

버킷에 대한 리소스 액세스 구성

버킷에 대한 리소스 액세스를 구성하려면 다음 절차를 완료하세요.

1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 스토리지를 선택합니다.
3. 리소스 액세스를 구성할 버킷 이름을 선택합니다.
4. 권한 탭을 선택합니다.

   페이지의 리소스 액세스(Resource access) 섹션에는 현재 버킷에 연결된 인스턴스가 표시됩니다(있는 경우).
5. 인스턴스 연결(Attach instance)을 선택하여 버킷에 인스턴스를 연결합니다.
6. 인스턴스 선택(Select an instance) 드롭다운 메뉴에서 버킷에 연결할 인스턴스를 선택합니다.

   > **Note**
   >
   > 실행 중이거나 중지된 상태인 인스턴스만 연결할 수 있습니다. 또한 버킷 AWS 리전 과 동일한에 있는 인스턴스만 연결할 수 있습니다.
7. 연결(Attach)을 선택하여 인스턴스에 연결합니다. 그렇지 않은 경우 취소를 선택합니다.

   인스턴스는 연결된 버킷과 객체에 대한 모든 액세스 권한을 갖게 됩니다. 인스턴스에서 소프트웨어 또는 플러그 인이 버킷에 파일을 프로그래밍 방식으로 업로드하고 액세스하도록 구성할 수 있습니다. 예를 들어, 버킷에 미디어 파일을 저장하도록 WordPress 인스턴스를 구성할 수 있습니다. 자세한 내용은 자습서: WordPress 인스턴스에 버킷 연결을 참조하세요.

사용량 변동에 따른 Lightsail 버킷 스토리지 플랜 조정

Amazon Lightsail 객체 스토리지 서비스를 사용하는 경우 버킷의 스토리지 플랜으로 월별 비용, 스토리지 공간 할당량 및 데이터 전송 할당량이 좌우됩니다. 월별 AWS 결제 주기 내에 버킷의 스토리지 플랜을 한 번만 업데이트할 수 있습니다. 버킷의 스토리지 플랜을 변경하면 스토리지 공간 및 네트워크 전송 할당량이 재설정됩니다. 그러나 이전 스토리지 플랜을 사용하여 발생했을 수 있는 초과 스토리지 공간 및 데이터 전송 요금은 재설정되지 않습니다.

버킷이 스토리지 공간 또는 데이터 전송 할당량을 지속적으로 초과하거나 버킷의 사용량이 이러한 할당량 범위의 낮은 수준에 머무는 경우 버킷의 스토리지 플랜을 업데이트하세요. 버킷 사용량이 예기치 않게 변동될 수 있으므로 버킷의 스토리지 플랜을 매달 비용을 절감하려는 단기적인 관점이 아닌 장기적인 전략을 바탕으로 변경하는 것이 좋습니다. 앞으로 버킷에 충분한 스토리지 공간과 데이터 전송 할당량을 제공할 수 있는 스토리지 플랜을 선택합니다.

버킷에 대한 자세한 내용은 객체 스토리지를 참조하세요.

Lightsail 콘솔을 사용하여 버킷의 스토리지 플랜 변경

Lightsail 콘솔을 통해 버킷의 스토리지 플랜을 변경하려면 다음 절차를 수행하세요.

1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 스토리지를 선택합니다.
3. 플랜을 변경하려는 버킷의 이름을 선택합니다.
4. 버킷 관리 페이지에서 지표(Metrics) 탭을 선택합니다.
5. 스토리지 플랜 변경(Change storage plan)을 선택합니다.
6. 표시되는 확인 프롬프트에서 예, 변경합니다(Yes, change)를 선택하여 계속해서 버킷 스토리지 플랜을 변경합니다. 변경하지 않으려면 아니요, 취소합니다(No, cancel)를 선택합니다.
7. 사용할 플랜을 선택한 다음 플랜 선택(Select plan)을 선택합니다.
8. 표시되는 확인 프롬프트에서 예, 적용합니다(Yes, apply)를 선택하여 변경 사항을 버킷에 적용합니다. 적용하지 않으려면 아니요, 돌아갑니다(No, go back)를 선택합니다.

를 사용하여 버킷의 스토리지 플랜 변경 AWS CLI

AWS Command Line Interface ()를 사용하여 버킷의 계획을 변경하려면 다음 절차를 완료하세요AWS CLI. `update-bucket-bundle` 명령을 사용하여 플랜을 변경할 수 있습니다. 버킷 스토리지 플랜은 API에서 버킷 번들이라고 합니다. 자세한 내용은 AWS CLI 명령 레퍼런스의 update-bucket-bundle을 참조하세요.

> **Note**
>
> 이 절차를 계속하기 전에 AWS CLI 및 Amazon S3에 대해 Lightsail를 설치하고 구성해야 합니다. 자세한 내용은 에서 작동 AWS CLI 하도록 구성을 Lightsail참조하세요.

1. 명령 프롬프트 또는 터미널 창을 엽니다.

2. 다음 명령을 입력하여 버킷의 플랜을 변경합니다.

```
aws lightsail update-bucket-bundle --bucket-name BucketName --bundle-id BundleID
```

다음 예 명령을 사용하려면 아래 텍스트를 사용자의 값으로 대체합니다.

- *BucketName* -스토리지 플랜을 업데이트할 버킷의 이름입니다.
- *BundleID* - 버킷에 적용할 새 버킷 번들의 ID입니다. `get-bucket-bundles` 명령을 실행하여 사용 가능한 버킷 번들 및 해당 ID 목록을 확인합니다. 자세한 내용은 AWS CLI 명령 참조의 get-bucket-bundles를 참조하세요.

예제:

```
aws lightsail update-bucket-bundle --bucket-name amzn-s3-demo-bucket --bundle-
id medium_1_0
```

다음 예와 비슷한 결과가 나타나야 합니다.

```
C:\>aws lightsail update-bucket-bundle --bucket-name DOC-EXAMPLE-BUCKET --bundle-id medium_1_0

{
    "operations": [
        {
            "id": "8example-8176-48bd-b1da-exampleb8404",
            "resourceName": "DOC-EXAMPLE-BUCKET",
            "resourceType": "Bucket",
            "createdAt": "2021-06-30T12:05:57.362000-07:00",
            "location": {
                "availabilityZone": "all",
                "regionName": "us-west-2"
            },
            "isTerminal": true,
            "operationDetails": "62example362/DOC-EXAMPLE-BUCKET/medium_1_0",
            "operationType": "UpdateBucketBundle",
            "status": "Succeeded",
            "statusChangedAt": "2021-06-30T12:05:57.362000-07:00",
            "errorCode": "",
            "errorDetails": ""
        }
    ]
}
```

향상된 보안을 위한 Lightsail 버킷 액세스 권한 관리

버킷에 있는 객체에 대한 인증되지 않은 공개 읽기 전용 액세스를 제어하려면 버킷 액세스 권한을 사용하면 됩니다. 버킷을 비공개하거나 공개(읽기 전용)할 수 있습니다. 개별 객체를 공개(읽기 전용)하는 옵션을 사용할 수도 있지만, 버킷을 비공개할 수도 있습니다.

> **Important**
>
> 버킷을 공개(읽기 전용)하면 버킷의 URL(예: `https://amzn-s3-demo-bucket.us-east-1.amazonaws.com/media/sailbot.jpg`)을 통해 인터넷상의 모든 사용자가 버킷에 있는 객체를 전부 읽을 수 있게 됩니다. 인터넷상의 다른 사용자가 객체에 액세스하지 못하게 하려면 버킷을 공개(읽기 전용)해서는 안 됩니다.

권한 옵션에 대한 자세한 내용은 버킷 권한을 참조하세요. 보안 모범 사례에 대한 자세한 내용은 객체 스토리지에 대한 보안 모범 사례를 참조하세요. 버킷에 대한 자세한 내용은 객체 스토리지를 참조하세요.

> **Important**
>
> Lightsail 객체 스토리지 리소스에서는 퍼블릭 액세스를 허용하거나 거부할 때 Lightsail 버킷 액세스 권한 및 Amazon S3 계정 수준 퍼블릭 액세스 차단 구성을 모두 고려합니다. 자세한 내용은 버킷에 대한 퍼블릭 액세스 차단을 참조하세요.

버킷 액세스 권한 구성

버킷에 대한 액세스 권한을 구성하려면 다음 절차를 완료하세요.

1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 스토리지를 선택합니다.
3. 액세스 권한을 구성할 버킷 이름을 선택합니다.
4. 권한 탭을 선택합니다.

   페이지의 버킷 액세스 권한(Bucket access permissions) 섹션에는 현재 버킷에 구성된 액세스 권한이 표시됩니다.
5. 권한 변경(Change permission)을 선택하여 버킷 액세스 권한을 변경합니다.

6. 다음 옵션 중 하나를 선택하세요.

   - 모든 객체 비공개(All objects are private) - 사용자 본인이나 사용자에게 액세스 권한을 부여받은 사람만 버킷에 있는 모든 객체를 읽을 수 있습니다.
   - 개별 객체 공개(읽기 전용) 가능(Individual objects can be made public (read-only)) – 개별 객체를 공개(읽기 전용)하지 않는 한 사용자 본인이나 사용자에게 액세스 권한을 부여받은 사람만 버킷에 있는 객체를 읽을 수 있습니다. 개별 객체 액세스 권한에 대한 자세한 내용은 버킷에 있는 개별 객체에 대한 액세스 권한 구성을 참조하세요.

     버킷에 있는 일부 객체만 공개(읽기 전용)하고 다른 객체는 모두 비공개하는 등 특별히 필요한 경우에만 개별 객체 공개(읽기 전용) 가능(Individual objects can be made public (read-only)) 옵션을 선택하는 것이 좋습니다. 예를 들어, 일부 WordPress 플러그 인의 경우 버킷에서 개별 객체를 공개하도록 허용해야 합니다. 자세한 내용은 자습서: WordPress 인스턴스에 버킷 연결 및 자습서: 콘텐츠 전송 네트워크 배포와 함께 버킷 사용을 참조하세요.
   - 모든 객체 공개(읽기 전용)(All objects are public (read-only)) – 인터넷상의 모든 사용자가 버킷에 있는 객체를 전부 읽을 수 있습니다.

     > **Important**
     >
     > 버킷을 공개(읽기 전용)하면 버킷의 URL(예: `https://amzn-s3-demo-bucket.us-east-1.amazonaws.com/media/sailbot.jpg`)을 통해 인터넷상의 모든 사용자가 버킷에 있는 객체를 전부 읽을 수 있게 됩니다. 인터넷상의 다른 사용자가 객체에 액세스하지 못하게 하려면 버킷을 공개(읽기 전용)해서는 안 됩니다.

7. 저장을 선택하여 변경 사항을 저장합니다. 그렇지 않은 경우 취소를 선택합니다.

   변경하는 버킷 액세스 권한에 따라 다음과 같은 변경 사항이 구현됩니다.

   - 모든 객체 비공개(All objects are private) - 이전에 공개(읽기 전용)(Public (read-only)) 개별 객체 액세스 권한으로 구성했던 경우에도 버킷의 모든 객체가 비공개로 설정됩니다.
   - 개별 객체 공개 가능(읽기 전용)(Individual objects can be made public (read-only)) - 이전에 공개(읽기 전용)(Public (read-only)) 개별 객체 액세스 권한으로 구성했던 객체가 공개됩니다. 이제 객체에 개별 객체 액세스 권한을 구성할 수 있습니다.
   - 모든 객체 공개(읽기 전용)(All objects are public (read-only)) - 이전에 비공개(Private) 개별 객체 액세스 권한으로 구성했던 경우에도 버킷의 모든 객체가 공개(읽기 전용)됩니다.

     개별 객체 액세스 권한에 대한 자세한 내용은 버킷에 있는 개별 객체에 대한 액세스 권한 구성을 참조하세요.

AWS 계정 간 Lightsail 버킷에 대한 읽기 전용 액세스 권한 부여

크로스 계정 액세스를 통해 다른 AWS 계정과 해당 계정의 사용자에게 버킷에 있는 모든 객체에 대한 읽기 전용 액세스 권한을 부여할 수 있습니다. 교차 계정 액세스는 다른 AWS 계정과 객체를 공유하려는 경우에 적합합니다. 다른 AWS 계정에 교차 계정 액세스 권한을 부여하면 해당 계정의 사용자는 버킷의 URL과 객체(예: `https://amzn-s3-demo-bucket.us-east-1.amazonaws.com/media/sailbot.jpg`)를 통해 버킷의 객체에 대한 읽기 전용 액세스 권한을 갖게 됩니다. 버킷에 최대 10개의 AWS 계정에 대한 액세스 권한을 부여할 수 있습니다.

권한 옵션에 대한 자세한 내용은 버킷 권한을 참조하세요. 보안 모범 사례에 대한 자세한 내용은 객체 스토리지에 대한 보안 모범 사례를 참조하세요. 버킷에 대한 자세한 내용은 객체 스토리지를 참조하세요.

버킷에 대한 교차 계정 액세스 구성

버킷에 대한 교차 계정 액세스를 구성하려면 다음 절차를 완료하세요.

1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 스토리지를 선택합니다.
3. 교차 계정 액세스 권한을 구성할 버킷 이름을 선택합니다.
4. 권한 탭을 선택합니다.

   페이지의 크로스 계정 액세스 섹션에는 현재 버킷에 액세스할 수 있도록 구성된 AWS 계정 ID(있는 경우)가 표시됩니다.
5. 교차 계정 액세스 추가를 선택하여 다른 AWS 계정의 버킷에 대한 액세스 권한을 부여합니다.
6. AWS 계정 ID 텍스트 상자에 액세스 권한을 부여하려는 계정의 ID를 입력합니다.
7. 저장(Save)을 선택하여 액세스 권한을 부여합니다. 그렇지 않은 경우 취소를 선택합니다.

   추가한 AWS 계정 ID는 페이지의 교차 계정 액세스 섹션에 나열됩니다. AWS 계정의 크로스 계정 액세스 권한을 제거하려면 제거할 AWS 계정 ID 옆에 있는 삭제(휴지통) 아이콘을 선택합니다.

Amazon Lightsail의 개별 버킷 객체에 대한 퍼블릭 액세스 권한 부여

버킷에 있는 개별 객체에 대한 인증되지 않은 공개 읽기 전용 액세스를 제어하려면 개별 객체 액세스 권한을 사용하면 됩니다. 버킷에 있는 개별 객체를 비공개하거나 공개(읽기 전용)할 수 있습니다.

> **Important**
>
> 개별 객체 액세스 권한은 버킷의 액세스 권한이 개별 객체 공개 가능(읽기 전용)(Individual objects can be made public (read-only))으로 설정된 경우에만 구성할 수 있습니다. 버킷 권한 옵션에 대한 자세한 내용은 버킷 권한을 참조하세요. 버킷에 대한 자세한 내용은 객체 스토리지를 참조하세요.

버킷의 일부 객체만 공개하고 다른 모든 객체는 비공개로 유지하는 등 특별한 경우에만 개별 객체 액세스 권한을 구성하는 것이 좋습니다. 예를 들어, 일부 WordPress 플러그 인의 경우 버킷에서 개별 객체를 공개하도록 허용해야 합니다. 자세한 내용은 자습서: WordPress 인스턴스에 버킷 연결 및 자습서: 콘텐츠 전송 네트워크 배포와 함께 버킷 사용을 참조하세요.

권한 옵션에 대한 자세한 내용은 버킷 권한을 참조하세요. 보안 모범 사례에 대한 자세한 내용은 객체 스토리지에 대한 보안 모범 사례를 참조하세요. 버킷에 대한 자세한 내용은 객체 스토리지를 참조하세요.

개별 객체 액세스 권한 구성

버킷의 개별 객체에 대한 액세스 권한을 구성하려면 다음 절차를 완료하세요. 사용자에게 Lightsail에서 버킷 관리 가능 권한을 부여하는 IAM 정책의 예는 버킷을 관리하기 위한 IAM 정책을 참조하세요.

1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 스토리지를 선택합니다.
3. 개별 객체에 대한 액세스 권한을 구성할 버킷 이름을 선택합니다.
4. 객체(Objects) 탭을 선택합니다.
5. 액세스 권한을 구성하려는 객체 옆에 확인 표시를 추가합니다.

   객체 정보 창에는 객체에 대한 현재 액세스 권한이 표시됩니다.
6. 객체에 대한 액세스 권한을 변경하려면 객체 정보 창의 권한(Permissions) 섹션에서 편집(Edit)을 선택합니다.

   > **Note**
   >
   > 편집 옵션을 사용할 수 없는 경우 버킷의 액세스 권한으로는 개별 객체 액세스 권한을 구성할 수 없습니다. 개별 객체 액세스 권한을 구성하려면 버킷 액세스 권한을 개별 객체 공

개 가능(읽기 전용)(Individual objects can be made public (read-only))으로 설정해야 합니다. 자세한 내용은 버킷 액세스 권한 구성을 참조하세요.

7. 권한 선택(Select a permission) 드롭다운 메뉴에서 다음 옵션 중 하나를 선택합니다.

   - 비공개(Private) - 사용자 본인이나 사용자에게 액세스 권한을 부여받은 사람만 객체를 읽을 수 있습니다.
   - 공개(읽기 전용)(Public (read-only)) - 전 세계 모든 사용자가 객체를 읽을 수 있습니다.

8. 저장을 선택하여 변경 사항을 저장합니다. 그렇지 않은 경우 취소를 선택합니다.

   버킷의 버킷 액세스 권한 설정은 개별 객체 액세스 권한에 다음과 같은 영향을 미칩니다.

   - 버킷 액세스 권한을 모든 객체 비공개(All objects are private)로 변경하면 버킷의 모든 객체가 공개(읽기 전용)(Public (read-only)) 개별 객체 액세스 권한으로 구성된 경우에도 비공개됩니다. 단, 구성된 개별 객체 액세스 권한은 그대로 유지됩니다. 예를 들어, 버킷 액세스 권한을 개별 객체 공개 가능(읽기 전용)(Individual objects can be made public (read-only))으로 다시 변경하면 공개(읽기 전용)(Public (read-only)) 개별 액세스 권한이 설정된 모든 객체가 공개적으로 읽을 수 있도록 설정됩니다.
   - 버킷 액세스 권한을 모든 객체 공개(읽기 전용)(All objects are public (read-only))로 변경하면 버킷에 있는 모든 객체가 비공개(Private) 개별 객체 액세스 권한으로 설정된 경우에도 공개(읽기 전용)됩니다.

     버킷 액세스 권한에 대한 자세한 내용은 버킷 액세스 권한 구성을 참조하세요.

멀티파트 업로드로 Lightsail 버킷에 파일 업로드

멀티파트 업로드를 사용하면 단일 파일을 버킷에 여러 파트의 세트로 업로드할 수 있습니다. 각 파트는 파일 데이터의 연속적인 부분입니다. 이러한 파일 파트는 독립적으로 그리고 임의의 순서로 업로드할 수 있습니다. 부분의 전송이 실패할 경우 다른 부분에 영향을 주지 않고도 해당 부분을 재전송할 수 있습니다. 파일의 모든 파트가 업로드되면 Amazon S3에서 이러한 파트를 결합하고 Amazon Lightsail의 버킷에 객체를 생성합니다. 일반적으로 객체 크기가 100MB에 근접할 경우, 단일 작업에서 객체를 업로드하는 대신 멀티파트 업로드 사용을 고려해 봐야 합니다. 버킷에 대한 자세한 내용은 객체 스토리지를 참조하세요.

멀티파트 업로드 사용은 다음 이점을 제공합니다.

- 개선된 처리량 - 파트를 병렬적으로 업로드하여 처리량을 개선할 수 있습니다.

- 네트워크 문제로부터 빠른 복구 - 더 작아진 파트 크기는 네트워크 오류로 인해 실패한 업로드 재시작의 영향을 최소화합니다.
- 시간 경과에 따라 업로드 - 시간 경과에 따라 파일 파트를 업로드할 수 있습니다. 멀티파트 업로드를 시작한 후 24시간 안에 이를 완료해야 합니다.
- 최종 파일 크기를 알기 전에 업로드 시작 – 파일을 생성하면서 업로드할 수 있습니다.

다음 방법으로 멀티파트 업로드를 사용하는 것이 좋습니다.

- 안정적인 고대역폭 네트워크를 통해 대용량 파일을 업로드하는 경우, 멀티파트 업로드는 멀티 스레드 성능을 위해 파일 파트를 병렬로 업로드하여 사용 가능한 대역폭 사용을 극대화합니다.
- 불규칙한 네트워크를 통해 업로드하는 경우, 멀티파트 업로드를 사용하여 업로드가 다시 시작되는 것을 방지하여 네트워크 오류에 대한 복원력을 높입니다. 멀티파트 업로드를 사용하는 경우 중단된 파트만 다시 업로드해야 합니다. 처음부터 시작하거나 전체 파일을 다시 업로드할 필요가 없습니다.

목차

- 멀티파트 업로드 프로세스
- 동시 멀티파트 업로드 작업
- 멀티파트 업로드 보존
- Amazon Simple Storage Service 멀티파트 업로드 제한
- 업로드할 파일 분할
- 를 사용하여 멀티파트 업로드 시작 AWS CLI
- 를 사용하여 파트 업로드 AWS CLI
- 를 사용하여 멀티파트 업로드의 부분 나열 AWS CLI
- 멀티파트 업로드 .json 파일 생성
- 를 사용하여 멀티파트 업로드 완료 AWS CLI
- 를 사용하여 버킷에 대한 멀티파트 업로드 나열 AWS CLI
- AWS CLI를 사용하여 멀티파트 업로드 중지

멀티파트 업로드 프로세스

멀티파트 업로드는 Amazon S3 작업을 사용하여 Lightsail의 버킷에 파일을 업로드하는 3단계 프로세스입니다.

1. CreateMultipartUpload 작업을 사용하여 멀티파트 업로드를 시작합니다.
2. UploadPart 작업을 사용하여 파일 파트를 업로드합니다.
3. CompleteMultipartUpload 작업을 사용하여 멀티파트 업로드를 완료합니다.

> **Note**
>
> AbortMultipartUpload 작업을 사용하여 시작한 멀티파트 업로드를 중지합니다.

멀티파트 업로드 요청이 완료되면 Amazon Simple Storage Service는 업로드된 파트에서 객체를 구성합니다. 그런 다음 버킷의 다른 객체에 액세스하는 것과 동일한 방식으로 해당 객체에 액세스할 수 있습니다.

진행 중인 모든 멀티파트 업로드 작업의 목록 또는 특정 멀티파트 업로드에 대해 업로드한 부분의 목록을 확인할 수 있습니다. 다음 섹션에서 이러한 각 작업에 대해 자세히 설명합니다.

멀티파트 업로드 시작

멀티파트 업로드를 시작하기 위한 요청을 보내면 Amazon Simple Storage Service는 업로드 ID와 함께 응답을 반환합니다. 이 응답이 멀티파트 업로드의 고유 식별자입니다. 파트 업로드, 파트 나열, 업로드 완료 또는 업로드 중단 시 항상 이 업로드 ID를 포함해야 합니다. 업로드 중인 객체를 설명하는 메타데이터를 제공하려면 요청에 메타데이터를 지정하여 멀티파트 업로드를 시작해야 합니다.

파트 업로드

부분을 업로드할 때 업로드 ID와 함께 부분 번호를 지정해야 합니다. 1부터 10,000까지 부분 번호를 지정할 수 있습니다. 부분 번호를 사용하여 업로드하는 객체에서 각 부분과 그 위치를 고유하게 식별합니다. 부분 번호는 굳이 연속 시퀀스로 선택할 필요가 없습니다(예를 들면 1, 5 및 14를 선택해도 됩니다). 이전에 업로드한 부분과 동일한 부분 번호로 새 부분을 업로드할 경우 이전에 업로드한 부분을 덮어쓰게 됩니다.

파트를 업로드할 때마다 Amazon Simple Storage Service는 응답으로 ETag 헤더를 반환합니다. 각 부분 업로드에 대해 부분 번호와 ETag 값을 기록해야 합니다. 이후 멀티파트 업로드를 완료하기 위한 요청에 이러한 값을 포함해야 하기 때문입니다.

> **Note**
>
> 멀티파트 업로드의 업로드된 모든 파트는 버킷에 저장됩니다. 이러한 파트는 사용자가 업로드를 완료하거나 업로드를 중지하거나 업로드 시간이 초과될 때까지 버킷의 스토리지 공간을 사용합니다. 자세한 내용은 가이드 후반부의 멀티파트 업로드 보존을 참조하세요.

멀티파트 업로드 완료

멀티파트 업로드를 완료하면 Amazon Simple Storage Service는 부분 번호를 바탕으로 오름차순으로 각 부분을 결합하여 객체를 완성합니다. 멀티파트 업로드 시작 요청에 객체 메타데이터가 제공된 경우 Amazon Simple Storage Service는 해당 메타데이터를 객체와 연결합니다. 성공적으로 완료 요청이 수행되면 파트는 더 이상 존재하지 않습니다.

멀티파트 업로드 완료 요청에는 업로드 ID와 각 파트 번호 및 해당 ETag 값의 목록이 포함되어야 합니다. Amazon Simple Storage Service 응답에는 결합된 객체 데이터를 고유하게 식별하는 ETag가 포함됩니다. 이 ETag가 반드시 객체 데이터의 MD5 해시여야 하는 것은 아닙니다.

필요할 경우 멀티파트 업로드를 중단할 수 있습니다. 멀티파트 업로드를 중단한 후에는 해당 업로드 ID로 다시 부분을 업로드할 수 없습니다. 취소된 멀티파트 업로드의 부분이 소비하는 모든 스토리지는 다시 비워집니다. 파트 업로드가 진행 중일 때 멀티파트 업로드를 중단하더라도 진행 중인 파트 업로드는 성공적으로 완료되거나, 혹은 오류로 멈출 수도 있습니다. 따라서 각 부분에서 사용하고 있는 스토리지를 모두 비우려면 모든 파트 업로드가 완료된 후에 멀티파트 업로드를 중단해야 합니다.

멀티파트 업로드 나열

특정 멀티파트 업로드 또는 진행 중인 모든 멀티파트 업로드에 대해 부분 목록을 확인할 수 있습니다. 부분 목록 조회 작업은 특정 멀티파트 업로드에 대해 업로드한 부분의 정보를 반환합니다. 각 부분 목록 조회 요청에 대해 Amazon Simple Storage Service는 특정 멀티파트 업로드에서 최대 1,000개의 부분에 대해 부분 정보를 반환합니다. 멀티파트 업로드에서 1,000개 이상의 부분이 있을 경우 모든 부분을 검색하려면 부분 목록 조회 요청을 여러 번 반복해야 합니다. 반환된 파트 목록에는 업로드 중인 파트가 포함되지 않습니다. 멀티파트 업로드 나열 작업을 사용하여 진행 중인 멀티파트 업로드의 목록을 확인할 수 있습니다.

진행 중인 멀티파트 업로드는 시작했지만 아직 완료 또는 중단하지 않은 업로드입니다. 각 요청은 최대 1,000개의 멀티파트 업로드를 반환합니다. 진행 중인 멀티파트 업로드가 1,000개 이상일 경우 남은 멀티파트 업로드를 모두 검색하려면 추가 요청을 전송해야 합니다. 반환된 목록은 확인을 위해서만 사용합니다. 전체 멀티파트 업로드 요청을 보낼 때 이 나열 결과를 사용해서는 안 됩니다. 그 대신, 부분

을 업로드할 때 지정한 부분 번호 및 Amazon Simple Storage Service가 반환한 해당 ETag 값의 목록을 보관해야 합니다.

동시 멀티파트 업로드 작업

분산 개발 환경에서는 애플리케이션에서 한 객체에 대해 동시에 여러 업데이트를 시작할 수 있습니다. 애플리케이션은 동일한 객체 키를 사용하여 여러 멀티파트 업로드를 시작할 수 있습니다. 이러한 각 업로드에 대해 애플리케이션은 부분을 업로드한 후 Amazon Simple Storage Service가 객체를 생성하도록 업로드 완료 요청을 전송할 수 있습니다. 버킷에서 버전 관리를 사용할 경우 멀티파트 업로드를 완료하면 항상 새 버전이 생성됩니다. 버킷에서 버전 관리를 활성화하지 않았으면 멀티파트 업로드가 시작된 후 완료되기 전에 수신되는 요청과 같은 다른 요청이 우선할 수 있습니다.

> **Note**
>
> 멀티파트 업로드를 시작한 후 완료하기 전에 수신되는 요청과 같은 다른 요청이 우선될 수 있습니다. 예를 들어, 한 키로 멀티파트 업로드를 시작한 후 이를 완료하기 전에 다른 작업을 수행하면 해당 키가 삭제될 수 있습니다. 이 경우 전체 멀티파트 업로드 응답은 객체를 보지 못한 상태에서도 객체를 성공적으로 생성했다고 나타낼 수 있습니다.

멀티파트 업로드 보존

멀티파트 업로드의 업로드된 모든 파트는 버킷에 저장됩니다. 이러한 파트는 사용자가 업로드를 완료하거나 업로드를 중지하거나 업로드 시간이 초과될 때까지 버킷의 스토리지 공간을 사용합니다. 멀티파트 업로드가 생성된 지 24시간이 지나면 멀티파트 업로드 시간이 초과되어 삭제됩니다. 멀티파트 업로드를 중지하거나 시간이 초과되면 업로드된 모든 파트가 삭제되고, 파트가 버킷에서 차지했던 스토리지 공간이 확보됩니다.

Amazon Simple Storage Service 멀티파트 업로드 제한

다음 표에 멀티파트 업로드의 주요 사양이 나와 있습니다.

- 최대 객체 크기: 50TB
- 업로드당 최대 파트 번호: 10,000
- 파트 번호: 1~10,000(포함)
- 파트 크기: 5MB(최소)~5GB(최대) 멀티파트 업로드의 마지막 파트에는 크기 제한이 없습니다.
- 파트 나열 요청에 대해 반환되는 최대 파트 번호: 1,000
- 멀티파트 업로드 나열 요청에서 반환되는 최대 멀티파트 업로드 번호: 1,000

업로드할 파일 분할

Linux 또는 Unix 운영 체제에서 `split` 명령을 사용하여 파일을 여러 파트로 분할한 다음 버킷에 업로드할 수 있습니다. Windows 운영 체제에서 파일을 분할하는 데 사용할 수 있는 유사한 프리웨어 애플리케이션이 있습니다. 파일을 여러 파트로 분할한 후 가이드의 멀티파트 업로드 시작 섹션을 이어서 진행합니다.

를 사용하여 멀티파트 업로드 시작 AWS CLI

AWS Command Line Interface (AWS CLI)를 사용하여 멀티파트 업로드를 시작하려면 다음 절차를 완료하세요. `create-multipart-upload` 명령을 사용하여 이 작업을 수행할 수 있습니다. 자세한 내용은 AWS CLI 명령 레퍼런스의 create-multipart-upload를 참조하세요.

> **Note**
>
> 이 절차를 계속하기 전에 AWS CLI 및 Amazon S3에 대해 Lightsail를 설치하고 구성해야 합니다. 자세한 내용은 에서 작동 AWS CLI 하도록 구성을 Lightsail참조하세요.

1. 명령 프롬프트 또는 터미널 창을 엽니다.
2. 다음 명령을 입력하여 버킷에 대한 멀티파트 업로드를 생성합니다.

```
aws s3api create-multipart-upload --bucket BucketName --key ObjectKey --acl bucket-owner-full-control
```

명령에서 아래 예 텍스트를 사용자의 값으로 대체합니다.

- *BucketName*#- 멀티파트 업로드를 생성하려는 버킷의 이름입니다.
- *ObjectKey*#- 업로드할 파일에 사용하려는 객체 키입니다.

예제:

```
aws s3api create-multipart-upload --bucket amzn-s3-demo-bucket --key sailbot.mp4 --acl bucket-owner-full-control
```

다음 예제와 비슷한 결과가 나타나야 합니다. 응답에는 파트를 업로드하고 이 객체에 대한 멀티파트 업로드를 완료할 수 있도록 후속 명령에 지정해야 하는 `UploadID`가 포함됩니다.

```
C:\ >aws s3api create-multipart-upload --bucket DOC-EXAMPLE-BUCKET --key sailbot.mp4
{
    "AbortDate": "2021-05-20T00:00:00+00:00",
    "AbortRuleId": "ExpireMultiPart",
    "ServerSideEncryption": "AES256",
    "Bucket": "DOC-EXAMPLE-BUCKET",
    "Key": "sailbot.mp4",
    "UploadId": "R4QU.mO.exampleiHWiLOeNw7JtXX7OotRhTLsXXCzF21CZdYlfj5lfjtiMnpzVw2WPj.exampleBTmL_N_.42.DlHYOTsITFsX.tO3XOUTTAHiCxY5VR8jWRGdkVkUG"
}
```

멀티파트 업로드에 대한 UploadID를 포함한 후 이 설명서의 다음 AWS CLI를 사용하여 파트 업로드 섹션을 이어서 진행하고 파트 업로드를 시작하세요.

를 사용하여 파트 업로드 AWS CLI

AWS Command Line Interface (AWS CLI)를 사용하여 멀티파트 업로드의 파트를 업로드하려면 다음 절차를 완료하세요. upload-part 명령을 사용하여 이 작업을 수행할 수 있습니다. 자세한 내용은 AWS CLI 명령 레퍼런스의 upload-part를 참조하세요.

> **Note**
>
> 이 절차를 계속하기 전에 AWS CLI 및 Amazon S3에 대해 Lightsail를 설치하고 구성해야 합니다. 자세한 내용은 에서 작동 AWS CLI 하도록 구성을 Lightsail참조하세요.

1. 명령 프롬프트 또는 터미널 창을 엽니다.
2. 다음 명령을 입력하여 버킷에 파트를 업로드합니다.

```
aws s3api upload-part --bucket BucketName --key ObjectKey --part-number Number --
body FilePart --upload-id "UploadID"  --acl bucket-owner-full-control
```

명령에서 아래 예 텍스트를 사용자의 값으로 대체합니다.

- *BucketName*#- 멀티파트 업로드를 생성하려는 버킷의 이름입니다.
- *ObjectKey*#- 업로드할 파일에 사용하려는 객체 키입니다.
- *Number* - 업로드하려는 파트의 파트 번호입니다. 부분 번호를 사용하여 업로드하는 객체에서 각 부분과 그 위치를 고유하게 식별합니다. 업로드하는 각 파트에 대해 --part-number 파라미터를 점진적으로 늘려야 합니다. 이렇게 하려면 멀티파트 업로드를 완료할 때 Amazon Simple Storage Service가 객체를 결합해야 하는 순서대로 번호를 매겨야 합니다.
- *FilePart* - 컴퓨터에서 업로드하려는 파트 파일입니다.
- *UploadID* - 가이드의 앞부분에서 생성한 멀티파트 업로드의 업로드 ID입니다.

예제:

```
aws s3api upload-part --bucket amzn-s3-demo-bucket --
key sailbot.mp4 --part-number 1 --body sailbot.mp4.001 --upload-id
 "R4QU.mO.exampleiHWiLOeNw7JtXX7OotRhTLsXXCzF21CZdYlfj5lfjtiMnpzVw2WPj.exampleBTmL_N_.42.Dl
 --acl bucket-owner-full-control
```

다음 예제와 비슷한 결과가 나타나야 합니다. 업로드한 각 파트에 대해 upload-part 명령을 반복합니다. 각 파트 업로드 요청의 응답에는 업로드한 파트의 ETag 값이 포함됩니다. 업로드한 각 파트의 ETag 값을 기록합니다. 멀티파트 업로드를 완료하려면 모든 ETag 값이 필요합니다. 이 값은 가이드의 후반부에서 다룹니다.

```
C:\ >aws s3api upload-part --bucket DOC-EXAMPLE-BUCKET  --key sailbot.mp4  --part-number 1 --body sailbot.mp4.001
 --upload-id "R4QU.mO.exampleiHWiLOeNw7JtXX7OotRhTLsXXCzF21CZdYlfj5lfjtiMnpzVw2WPj.exampleBTmL_N_.42.DlHYOTsITFsX.tO3XOUTTAHiCxY5VR8jWRGdkVkUG"
{
    "ServerSideEncryption": "AES256",
    "ETag": "\"4example7530246113e837a860a38bbb\""
}
```

AWS CLI를 사용하여 멀티파트 업로드의 파트 나열

AWS Command Line Interface (AWS CLI)를 사용하여 멀티파트 업로드의 파트를 나열하려면 다음 절차를 완료하세요. list-parts 명령을 사용하여 이 작업을 수행할 수 있습니다. 자세한 내용은 AWS CLI 명령 레퍼런스의 list-parts를 참조하세요.

이 절차를 완료하여 멀티파트 업로드에서 업로드된 모든 파트의 ETag 값을 가져옵니다. 이 값이 있어야 가이드 후반부에서 멀티파트 업로드를 완료할 수 있습니다. 단, 파트 업로드 응답의 ETag 값을 모두 기록한 경우 이 절차를 건너뛰고 가이드의 멀티파트 업로드 .json 파일 생성 섹션을 이어서 진행합니다.

> **Note**
>
> 이 절차를 계속하기 전에 AWS CLI 및 Amazon S3에 대해 Lightsail를 설치하고 구성해야 합니다. 자세한 내용은 에서 작동 AWS CLI 하도록 구성을 Lightsail참조하세요.

1. 명령 프롬프트 또는 터미널 창을 엽니다.
2. 다음 명령을 입력하여 버킷에 멀티파트 업로드 파트를 나열합니다.

```
aws s3api list-parts --bucket BucketName --key ObjectKey --upload-id "UploadID"
```

명령에서 아래 예 텍스트를 사용자의 값으로 대체합니다.

- *BucketName*#- 멀티파트 업로드의 파트를 나열하려는 버킷의 이름입니다.
- *ObjectKey*#- 멀티파트 업로드의 객체 키입니다.
- *UploadID* - 가이드의 앞부분에서 생성한 멀티파트 업로드의 업로드 ID입니다.

예제:

```
aws s3api list-parts --bucket amzn-s3-demo-bucket --key sailbot.mp4 --upload-id
 "R4QU.mO.exampleiHWiLOeNw7JtXX7OotRhTLsXXCzF21CZdYlfj5lfjtiMnpzVw2WPj.exampleBTmL_N_.42.Dl
```

다음 예제와 비슷한 결과가 나타나야 합니다. 응답에는 멀티파트 업로드에 업로드한 파트의 ETag 값과 모든 파트 번호가 나열됩니다. 이 값을 클립보드로 복사하고 가이드의 멀티파트 업로드 .json 생성 섹션을 이어서 진행합니다.

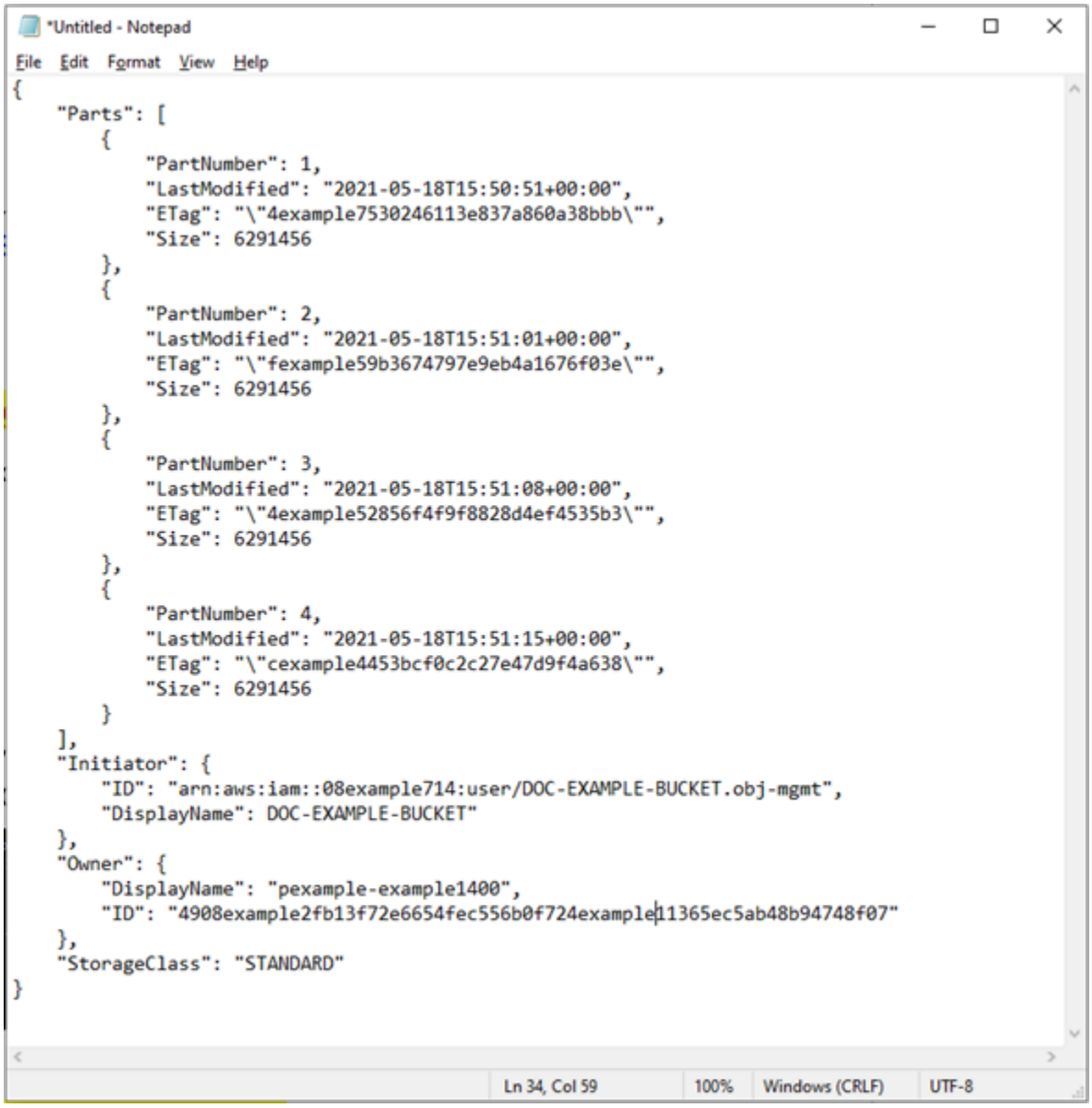
```
C:\>aws s3api list-parts --bucket DOC-EXAMPLE-BUCKET --key sailbot.mp4 --upload-id "R4QU.mO.exampleiHWiLOeNw7JtXX7OotR
hTLsXXCzF21CZdYlfj5lfjtiMnpzVw2WPj.exampleBTmL_N_.42.DlHYOTsITFsX.tO3XOUTTAHiCxY5VR8jWRGdkVkUG"
{
    "Parts": [
        {
            "PartNumber": 1,
            "LastModified": "2021-05-18T15:50:51+00:00",
            "ETag": "\"4example7530246113e837a860a38bbb\"",
            "Size": 6291456
        },
        {
            "PartNumber": 2,
            "LastModified": "2021-05-18T15:51:01+00:00",
            "ETag": "\"fexample59b3674797e9eb4a1676f03e\"",
            "Size": 6291456
        },
        {
            "PartNumber": 3,
            "LastModified": "2021-05-18T15:51:08+00:00",
            "ETag": "\"4example52856f4f9f8828d4ef4535b3\"",
            "Size": 6291456
        },
        {
            "PartNumber": 4,
            "LastModified": "2021-05-18T15:51:15+00:00",
            "ETag": "\"cexample4453bcf0c2c27e47d9f4a638\"",
            "Size": 6291456
        }
    ],
    "Initiator": {
        "ID": "arn:aws:iam::08example714:user/DOC-EXAMPLE-BUCKET.obj-mgmt",
        "DisplayName": "DOC-EXAMPLE-BUCKET"
    },
    "Owner": {
        "DisplayName": "pexample-example1400",
        "ID": "4908example2fb13f72e6654fec556b0f724example11365ec5ab48b94748f07"
    },
    "StorageClass": "STANDARD"
}
```

멀티파트 업로드 .json 파일 생성

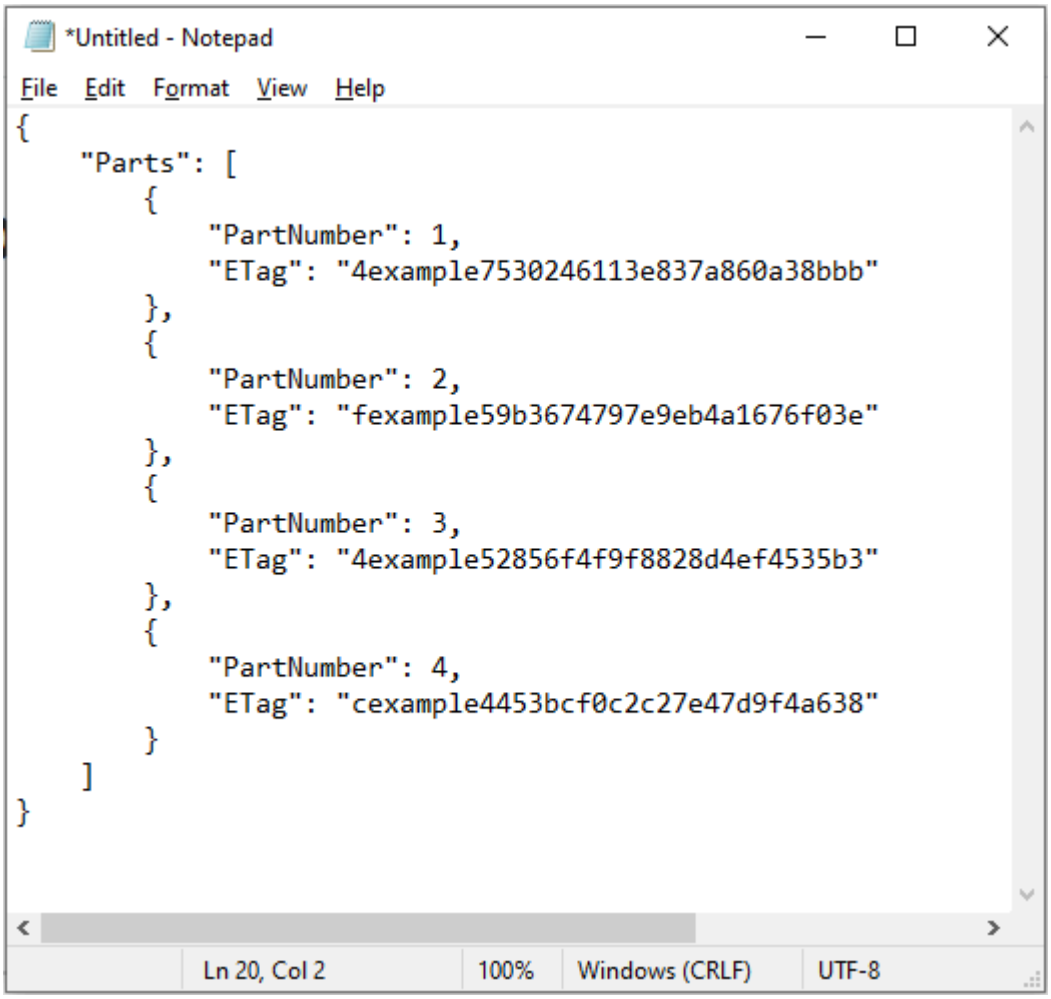
업로드한 모든 파트와 파트 ETag 값을 정의하는 멀티파트 업로드 .json 파일을 생성하려면 다음 절차를 완료하세요. 이 작업은 가이드의 후반부에서 수행해야 합니다.

1. 텍스트 편집기를 열고 가이드의 이전 섹션에서 요청한 `list-parts` 명령의 응답을 붙여넣습니다.

   결과는 다음 예제와 같아야 합니다.



2. 다음 예와 같이 텍스트 파일의 형식을 다시 지정합니다.



3. 텍스트 파일을 컴퓨터에 `mpstructure.json`으로 저장하고 가이드의 AWS CLI를 사용하여 멀티파트 업로드 완료 섹션을 이어서 진행합니다.

AWS CLI를 사용하여 멀티파트 업로드 완료

AWS Command Line Interface (AWS CLI)를 사용하여 멀티파트 업로드를 마치려면 다음 절차를 완료하세요. `complete-multipart-upload` 명령을 사용하여 이 작업을 수행할 수 있습니다. 자세한 내용은 AWS CLI 명령 레퍼런스의 complete-multipart-upload를 참조하세요.

> **Note**
>
> 이 절차를 계속하기 전에 AWS CLI 및 Amazon S3에 대해 Lightsail를 설치하고 구성해야 합니다. 자세한 내용은 에서 작동 AWS CLI 하도록 구성을 Lightsail참조하세요.

1. 명령 프롬프트 또는 터미널 창을 엽니다.

2. 다음 명령을 입력하여 버킷에 파트를 업로드합니다.

```
aws s3api complete-multipart-upload --multipart-upload file://JSONFileName --
bucket BucketName --key ObjectKey --upload-id "UploadID"  --acl bucket-owner-full-
control
```

명령에서 아래 예 텍스트를 사용자의 값으로 대체합니다.

- *JSONFileName*#- 가이드의 앞부분에서 생성한 .json 파일의 이름(예:mpstructure.json)입니다.
- *BucketName*#- 멀티파트 업로드를 완료하려는 버킷의 이름입니다.
- *ObjectKey*#- 멀티파트 업로드의 객체 키입니다.
- *UploadID* - 가이드의 앞부분에서 생성한 멀티파트 업로드의 업로드 ID입니다.

Example:

```
aws s3api complete-multipart-upload --multipart-upload file://mpstructure.json
 --bucket amzn-s3-demo-bucket --key sailbot.mp4 --upload-id
 "R4QU.mO.exampleiHWiLOeNw7JtXX7OotRhTLsXXCzF21CZdYlfj5lfjtiMnpzVw2WPj.exampleBTmL_N_.42.Dl
 --acl bucket-owner-full-control
```

다음 예와 비슷한 응답이 나타나는 것을 볼 수 있습니다. 이것으로 멀티파트 업로드가 완료되었는지 확인할 수 있습니다. 이제 객체가 결합되어 버킷에서 사용할 수 있게 됩니다.

```
C:\>aws s3api complete-multipart-upload --multipart-upload file://mpstructure.json --bucket DOC-EXAMPLE-BUCKET  --key sailbot.mp4
--upload-id "R4QU.mO.exampleiHWiLOeNw7JtXX7OotRhTLsXXCzF21CZdYlfj5lfjtiMnpzVw2WPj.exampleBTmL_N_.42.DlHYOTsITFsX.tO3XOUTTAHiCxY5VR8jWRGdkVkUG"
{
    "ServerSideEncryption": "AES256",
    "VersionId": "MexampleKMdfPQb.2YZHqOvE_T.vSDtY",
    "Location": "https://DOC-EXAMPLE-BUCKET.s3.us-west-2.amazonaws.com/sailbot.mp4",
    "Bucket": "DOC-EXAMPLE-BUCKET",
    "Key": "sailbot.mp4",
    "ETag": "\"1example5964e3115e5d3f3c9a731585-4\""
}
```

AWS CLI를 사용하여 버킷에 대한 멀티파트 업로드 나열

AWS Command Line Interface (AWS CLI)를 사용하여 버킷의 모든 멀티파트 업로드를 나열하려면 다음 절차를 완료하세요. `list-multipart-uploads` 명령을 사용하여 이 작업을 수행할 수 있습니다. 자세한 내용은 AWS CLI 명령 레퍼런스의 list-multipart-uploads를 참조하세요.

> **Note**
>
> 이 절차를 계속하기 전에 AWS CLI 및 Amazon S3에 대해 Lightsail를 설치하고 구성해야 합니다. 자세한 내용은 에서 작동 AWS CLI 하도록 구성을 Lightsail참조하세요.

1. 명령 프롬프트 또는 터미널 창을 엽니다.
2. 다음 명령을 입력하여 버킷에 파트를 업로드합니다.

```
aws s3api list-multipart-uploads --bucket BucketName
```

명령에서 *BucketName*을 모든 멀티파트 업로드를 나열하려는 버킷의 이름으로 바꿉니다.

예제:

```
aws s3api list-multipart-uploads --bucket amzn-s3-demo-bucket
```

다음 예와 비슷한 응답이 나타나는 것을 볼 수 있습니다.

AWS CLI를 사용하여 멀티파트 업로드 중지

AWS Command Line Interface ()를 사용하여 멀티파트 업로드를 중지하려면 다음 절차를 완료하세요 AWS CLI. 멀티파트 업로드를 시작했는데 더 이상 계속하지 않으려는 경우 이 작업을 수행하면 됩니다. `abort-multipart-upload` 명령을 사용하여 이 작업을 수행할 수 있습니다. 자세한 내용은 AWS CLI 명령 레퍼런스의 abort-multipart-upload를 참조하세요.

> **Note**
>
> 이 절차를 계속하기 전에 AWS CLI 및 Amazon S3에 대해 Lightsail를 설치하고 구성해야 합니다. 자세한 내용은 에서 작동 AWS CLI 하도록 구성을 Lightsail참조하세요.

1. 명령 프롬프트 또는 터미널 창을 엽니다.
2. 다음 명령을 입력하여 버킷에 파트를 업로드합니다.

```
aws s3api abort-multipart-upload --bucket BucketName --key ObjectKey --upload-id
 "UploadID" --acl bucket-owner-full-control
```

명령에서 아래 예 텍스트를 사용자의 값으로 대체합니다.

- *BucketName*#- 멀티파트 업로드를 중단하려는 버킷의 이름입니다.
- *ObjectKey*#- 멀티파트 업로드의 객체 키입니다.
- *UploadID* - 중지하려는 멀티파트 업로드의 업로드 ID입니다.

예제:

```
aws s3api abort-multipart-upload --bucket amzn-s3-demo-bucket --key sailbot.mp4 --
upload-id
 "R4QU.mO.exampleiHWiLOeNw7JtXX7OotRhTLsXXCzF21CZdYlfj5lfjtiMnpzVw2WPj.exampleBTmL_N_.42.Dl
 --acl bucket-owner-full-control
```

이 명령은 응답을 반환하지 않습니다. `list-multipart-uploads` 명령을 실행하여 멀티파트 업로드가 중지되었음을 확인할 수 있습니다.

Lightsail 객체 스토리지에 대한 버킷 이름 지정 요구 사항 준수

Amazon Lightsail 객체 스토리지 서비스에서 버킷을 생성할 때는 버킷에 이름을 지정해야 합니다. 버킷 이름은 고객이 버킷에 저장된 객체에 액세스할 때 사용할 URL의 일부입니다. 예를 들어 amzn-s3-demo-bucket에서 버킷의 이름을 지정하는 경우 버킷us-east-1 AWS 리전의 URL은 입니다amzn-s3-demo-bucket.s3.us-east-1.amazonaws.com. 버킷을 생성한 후에는 이름을 변경할 수 없습니다. 고객이 지정한 버킷 이름을 볼 수 있다는 점에 유의하세요. Lightsail 객체 스토리지 서비스에 대

한 자세한 내용은 객체 스토리지를 참조하세요. 버킷 생성에 대한 자세한 내용은 버킷 생성을 참조하세요.

버킷 이름은 DNS를 준수해야 합니다. 따라서 Lightsail에서 버킷 이름 지정에는 다음 규칙이 적용됩니다.

- 버킷 이름은 3~54자여야 합니다.
- 버킷 이름은 소문자, 숫자 및 하이픈(-)으로만 구성될 수 있습니다.
- 버킷 이름은 문자 또는 숫자로 시작하고 끝나야 합니다.
- 하이픈(-)은 단어를 구분할 수 있지만, 연속적으로 지정할 수는 없습니다. 예를 들어, doc-example-bucket은 허용되지만 doc--example--bucket은 허용되지 않습니다.
- 버킷 이름은 Amazon Simple Storage Service(S3)의 버킷을 포함하여 aws(표준 리전) 파티션 내에서 고유해야 합니다.
- 버킷 이름은 접두사 amzn-s3-demo-로 시작해서는 안됩니다.
- 버킷 이름은 접두사 sthree-로 시작해서는 안됩니다.
- 버킷 이름은 접두사 sthree-configurator로 시작해서는 안됩니다.
- 버킷 이름은 접미사 -s3alias로 끝나서는 안됩니다.

예제 버킷 이름

다음 예제 버킷 이름은 유효하며 권장 이름 지정 지침을 따릅니다.

- docexamplebucket1
- log-delivery-march-2020
- my-hosted-content

다음 예 버킷 이름은 허용되지 않습니다.

- doc.example.bucket(기간 포함)
- doc--example--bucket(2개의 연속 하이픈 포함)
- doc-example-bucket-(하이픈으로 끝남)

Lightsail 객체 스토리지 버킷의 키 이름

버킷에 업로드하는 파일은 Amazon Lightsail 객체 스토리지 서비스에 객체로 저장됩니다. 객체 키(또는 키 이름)는 버킷에 저장된 객체를 고유하게 식별합니다. 이 가이드에서는 Lightsail 콘솔을 통해 볼 수 있는 버킷의 폴더 구조를 구성하는 키 이름 및 키 이름 접두사의 개념을 안내합니다. 버킷에 대한 자세한 내용은 객체 스토리지를 참조하세요.

키 이름

Lightsail 객체 스토리지 서비스 데이터 모델은 파일 시스템에서 볼 수 있는 계층 구조 대신 단순한 구조를 사용합니다. 이 모델에는 폴더 및 하위 폴더의 계층이 없습니다. 그러나 키 이름 접두사와 구분 기호를 사용하여 논리적 계층을 유추할 수 있습니다. Lightsail 콘솔은 키 이름 접두사를 사용하여 객체를 폴더 구조로 표시합니다.

버킷에 다음과 같은 객체 키를 가진 4개의 객체가 있다고 가정해 보겠습니다.

- `Development/Projects.xls`
- `Finance/statement1.pdf`
- `Private/taxdocument.pdf`
- `to-dos.doc`

Lightsail 콘솔은 키 이름 접두사(`Development/`, `Finance/`, `Private/`) 및 구분 기호(`/`)를 사용하여 폴더 구조를 표현합니다. `to-dos.doc` 키 이름에는 접두사가 없으므로, 이 객체는 버킷의 루트 수준에 표시됩니다. Lightsail 콘솔에서 `Development/` 폴더를 찾으면 `Projects.xls` 객체가 표시됩니다. `Finance/` 폴더에는 `statement1.pdf` 객체가 표시되고, `Private/` 폴더에는 `taxdocument.pdf` 객체가 표시됩니다.

Lightsail 콘솔에서는 키 이름 접두사 및 구분 기호 값을 키 이름으로 사용하여 0바이트 객체를 만들어 폴더를 생성할 수 있습니다. 이러한 폴더 객체는 콘솔에 표시되지 않습니다. 그러나 다른 객체처럼 동작합니다. Amazon S3 API, AWS Command Line Interface (AWS CLI) 또는 AWS SDKs.

객체 키 명명 지침

객체 키 이름에 임의의 UTF-8 문자를 사용할 수 있습니다. 하지만 특정 문자는 키 이름에 사용하면 일부 애플리케이션 또는 프로토콜에 문제가 발생할 수도 있습니다. 다음 지침에 따르면 DNS, 웹 안전 문자, XML 파싱 프로그램 및 기타 API의 호환성을 극대화할 수 있습니다.

사용 가능 문자

다음 문자 집합은 일반적으로 키 이름으로 사용해도 문제가 되지 않습니다.

- 영숫자
  - 0~9
  - a-z
  - A~Z
- 특수 문자
  - 슬래시(/)
  - 느낌표(!)
  - 하이픈(-)
  - 밑줄(_)
  - 마침표(.)
  - 별표(*)
  - 작은 따옴표(')
  - 여는 괄호(()
  - 닫는 괄호())

다음은 유효한 객체 키 이름의 예입니다.

- `4my-organization`
- `my.great_photos-2014/jan/myvacation.jpg`
- `videos/2014/birthday/video1.wmv`

> **Important**
>
> 객체 키 이름이 마침표(.) 또는 2개의 마침표(..)로 끝나는 경우 Lightsail 콘솔을 사용하여 객체를 다운로드할 수 없습니다. 키 이름이 하나 또는 두 마침표로 끝나는 객체를 다운로드하려면 Amazon S3 API AWS CLI및 AWS SDKs 사용해야 합니다. 자세한 내용은 버킷의 객체 다운로드를 참조하세요.

특별한 처리가 필요한 문자

키 이름에서 다음 문자는 추가 코드 처리가 필요할 수도 있으며 URL 인코딩되거나 HEX로 참조해야 할 수 있습니다. 이러한 문자 중 일부는 인쇄가 되지 않으며 브라우저에서 처리하지 못할 수 있으므로 특별한 처리가 필요합니다.

- 앰퍼샌드(‘&‘)
- 달러(‘$‘)
- ASCII 문자 범위 00-1F(16진수, 10진수: 0~31) 및 7F(10진수: 127)
- '@' 기호(‘@‘)
- 등호(‘=‘)
- 세미콜론(‘;")
- 콜론(‘:‘)
- 더하기(‘+‘)
- 공백 – 경우에 따라 중요한 의미가 있는 공백의 순서가 사라질 수 있음(특히 공백이 여러 개 있는 경우)
- 쉼표(‘,‘)
- 물음표(‘?‘)

피해야 하는 문자

모든 애플리케이션 간 일관성을 유지하기 위해 상당한 특수 처리가 필요하므로 다음과 같은 문자는 키 이름에서 사용하지 않는 것이 좋습니다.

- 백슬래시(‘\‘)
- 왼쪽 중괄호(‘{‘)
- 인쇄되지 않는 ASCII 문자(128~255 사이의 10진수)
- 캐럿(‘^‘)
- 오른쪽 중괄호(‘}‘)
- 백분율 문자(‘%‘)
- 억음 악센트 기호(‘`‘)
- 오른쪽 대괄호(‘]‘)

- 인용 부호
- '보다 큼' 기호('>')
- 왼쪽 대괄호('[')
- 물결표('~')
- '보다 작음' 기호('<')
- '파운드' 문자('#')
- 세로 막대/파이프('|')

XML 관련 객체 키 제한 사항

줄 끝 처리의 XML 표준에서 지정한 대로 모든 XML 텍스트는 단일 캐리지 리턴(ASCII 코드 13) 및 줄 바꿈 바로 뒤에 오는 캐리지 리턴(ASCII 코드 10)이 단일 줄 바꿈 문자로 대체되도록 정규화됩니다. XML 요청에서 객체 키를 올바르게 구문 분석하려면 캐리지 리턴 및 기타 특수 문자가 XML 태그 내에 삽입될 때 해당 XML 엔터티 코드로 대체되어야 합니다. 다음은 이러한 특수 문자 및 대응하는 엔터티 코드의 목록입니다.

- ' 문자: `'`
- " 문자: `"`
- & 문자: `&`
- < 문자: `<`
- < 문자: `>`
- `\r` 문자: `
` 또는 `
`
- `\n` 문자: `
` 또는 `
`

다음 예에서는 캐리지 리턴의 대체물로 XML 엔터티 코드를 사용하는 방법을 보여줍니다. `DeleteObjects` 요청은 `/some/prefix/objectwith\rcarriagereturn` 키 파라미터가 있는 객체를 삭제합니다. 여기서 \r은 캐리지 리턴입니다.

```
<Delete xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
      <Object>
        <Key>/some/prefix/objectwith&#13;carriagereturn</Key>
      </Object>
    </Delete>
```

Lightsail 객체 스토리지 버킷 보호

Amazon Lightsail 객체 스트리지는 자체 보안 정책을 개발하고 구현할 때 고려해야 할 여러 보안 기능을 제공합니다. 다음 모범 사례는 일반적인 지침이며 완벽한 보안 솔루션을 나타내지는 않습니다. 이러한 모범 사례는 사용자의 환경에 적절하지 않거나 충분하지 않을 수 있으므로 규정이 아닌 참고용으로만 사용하세요.

목차

- 예방적 보안 모범 사례
  - 최소 권한 액세스 구현
  - Lightsail 버킷에 공개적으로 액세스할 수 없음을 확인
  - Amazon S3에서 퍼블릭 액세스 차단 사용
  - 버킷에 인스턴스를 연결하여 전체 프로그래밍 방식 액세스 권한 부여
  - 버킷 액세스 키 교체
  - 교차 계정 액세스를 사용하여 다른 AWS 계정에 버킷의 객체에 대한 액세스 권한 부여
  - 데이터의 암호화
  - 버전 관리 사용
- 모니터링 및 감사 모범 사례
  - 액세스 로깅 활성화 및 주기적인 보안 및 액세스 감사 수행
  - Lightsail 버킷 식별, 태그 지정 및 감사
  - 모니터링 도구를 사용하여 AWS 모니터링 구현
  - 사용 AWS CloudTrail
  - AWS 보안 권고 모니터링

예방적 보안 모범 사례

다음 모범 사례는 Lightsail 버킷과 관련된 보안 사고를 방지하는 데 도움이 될 수 있습니다.

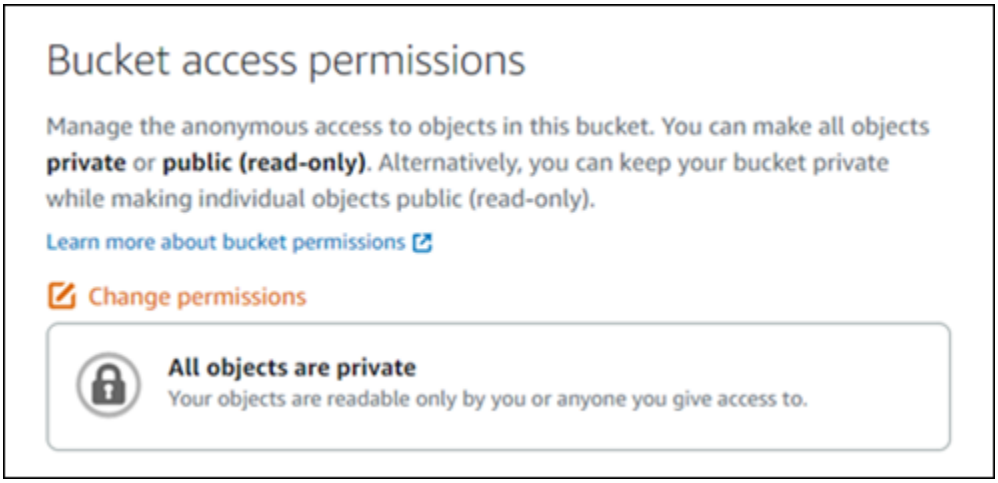
최소 권한 액세스 구현

권한을 부여할 때 누가 어떤 Lightsail 리소스에 대해 어떤 권한을 갖는지 결정합니다. 해당 리소스에서 허용할 작업을 사용 설정합니다. 따라서 작업을 수행하는 데 필요한 권한만 부여해야 합니다. 최소 권한 액세스를 구현하는 것이 오류 또는 악의적인 의도로 인해 발생할 수 있는 보안 위험과 영향을 최소화할 수 있는 근본적인 방법입니다.

버킷을 관리하기 위한 IAM 정책 생성에 대한 자세한 내용은 버킷을 관리하는 IAM 정책을 참조하세요. Lightsail 버킷에서 지원하는 Amazon S3 작업에 대한 자세한 내용은 Amazon Lightsail API 레퍼런스에서 객체 스토리지에 대한 작업을 참조하세요.

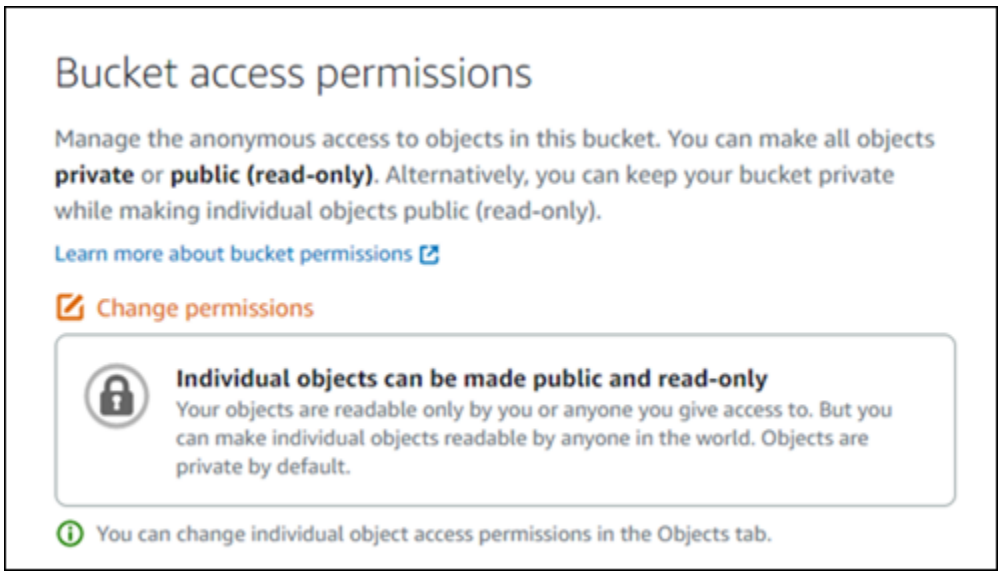
Lightsail 버킷에 공개적으로 액세스할 수 없음을 확인

기본적으로 버킷 및 객체는 프라이빗입니다. 버킷 액세스 권한을 모든 객체 프라이빗(All objects are private)으로 설정하여 버킷을 프라이빗으로 유지합니다. 대부분의 사용 사례의 경우 버킷이나 개별 객체를 퍼블릭으로 설정할 필요가 없습니다. 자세한 내용은 버킷에 있는 개별 객체에 대한 액세스 권한 구성을 참조하세요.

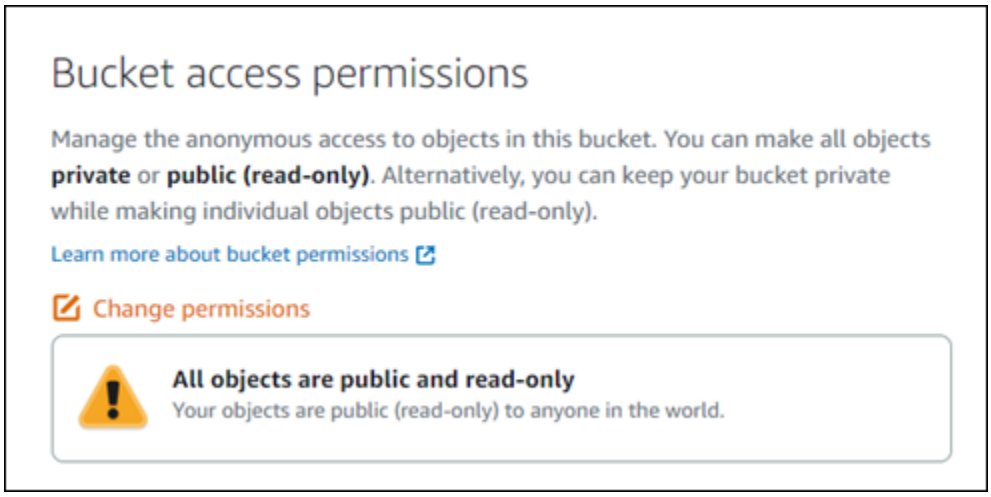


그러나 버킷을 사용하여 웹 사이트 또는 애플리케이션의 미디어를 호스팅하는 경우 특정 시나리오에서 버킷이나 개별 객체를 퍼블릭으로 설정해야 할 수 있습니다. 다음 옵션 중 하나를 구성하여 버킷 또는 개별 객체를 퍼블릭으로 설정할 수 있습니다.

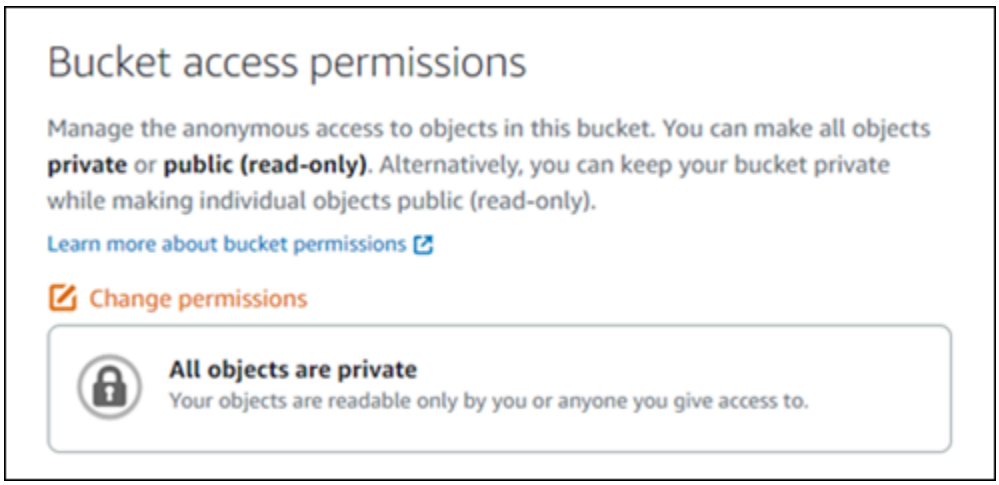
- 버킷의 일부 객체만 인터넷의 모든 사용자에게 퍼블릭(읽기 전용)이 되어야 하는 경우 버킷 액세스 권한을 개별 객체는 퍼블릭 및 읽기 전용으로 설정 가능(Individual objects can be made public and read-only)으로 변경하고 퍼블릭이 되어야 하는 객체만 퍼블릭(읽기 전용)(Public (read-only))으로 변경합니다. 이 옵션은 버킷을 프라이빗으로 유지하지만 개별 객체를 퍼블릭으로 설정할 수 있는 옵션을 제공합니다. 공개적으로 액세스하지 않으려는 민감한 정보나 기밀 정보가 포함된 경우 개별 객체를 퍼블릭으로 설정하지 마세요. 개별 객체를 퍼블릭으로 설정하는 경우 각 개별 객체의 퍼블릭 액세스 가능성을 주기적으로 확인해야 합니다.



- 버킷의 모든 객체가 인터넷상의 모든 사용자에게 퍼블릭(읽기 전용)이 되어야 하는 경우 버킷 액세스 권한을 모든 객체는 퍼블릭이며 읽기 전용(All objects are public and read-only)으로 변경합니다. 버킷의 객체에 민감한 정보나 기밀 정보가 포함된 경우 이 옵션을 사용하지 마세요.



- 이전에 버킷을 퍼블릭으로 변경했거나 개별 객체를 퍼블릭으로 변경한 경우 버킷 액세스 권한을 모든 객체는 프라이빗(All objects are private)으로 변경하여 버킷과 모든 해당 객체를 프라이빗으로 빠르게 변경할 수 있습니다.

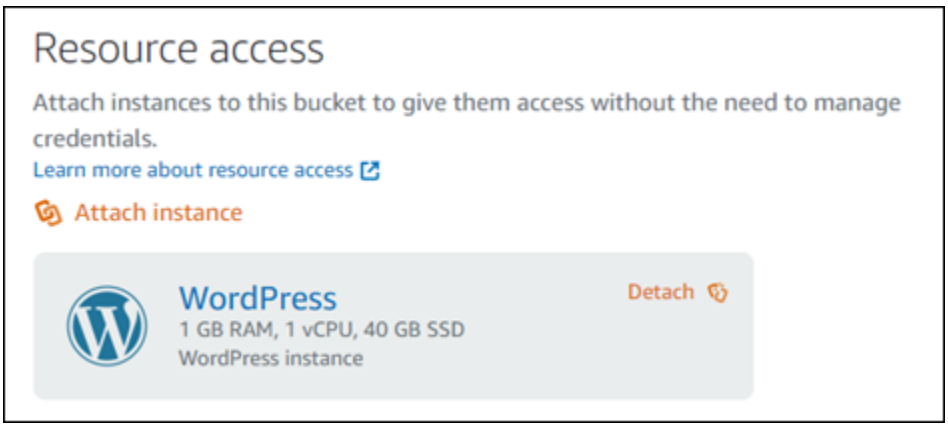


Amazon S3에서 퍼블릭 액세스 차단 사용

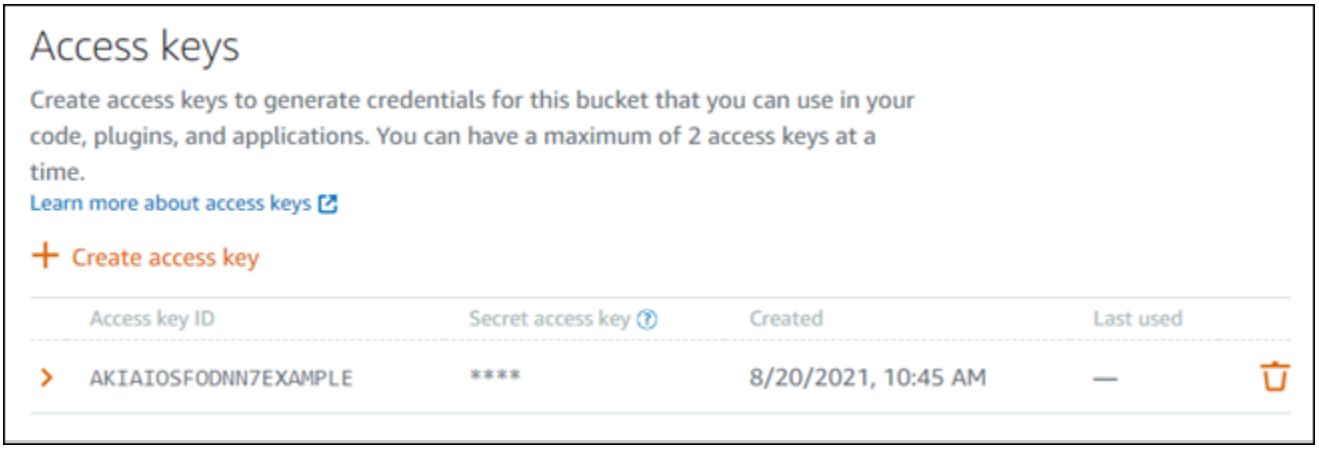
Lightsail 객체 스토리지 리소스에서는 퍼블릭 액세스를 허용하거나 거부할 때 Lightsail 버킷 액세스 권한 및 Amazon S3 계정 수준 퍼블릭 액세스 차단 구성을 모두 고려합니다. Amazon S3 계정 수준 퍼블릭 액세스 차단을 통해 계정 관리자와 버킷 소유자가 Amazon S3 버킷과 Lightsail 버킷에 대한 퍼블릭 액세스를 중앙에서 제한할 수 있습니다. 리소스가 생성되는 방식과 관계없고 구성되었을 수도 있는 개별 버킷 및 객체 권한과 관계없이 퍼블릭 액세스 차단을 통해 모든 Amazon S3 버킷과 Lightsail 버킷을 프라이빗으로 만들 수 있습니다. 자세한 내용은 버킷에 대한 퍼블릭 액세스 차단을 참조하세요.

버킷에 인스턴스를 연결하여 전체 프로그래밍 방식 액세스 권한 부여

Lightsail 객체 스토리지 버킷에 인스턴스를 연결하는 것이 버킷에 대한 액세스를 제공하는 가장 안전한 방법입니다. 인스턴스를 버킷에 연결하는 방식인 리소스 액세스(Resource access) 기능은 인스턴스에 버킷에 대한 전체 프로그래밍 방식 액세스 권한을 부여합니다. 이 방법을 사용하면 인스턴스 또는 애플리케이션에 직접 버킷 자격 증명을 저장하지 않아도 되며 자격 증명을 주기적으로 교체하지 않아도 됩니다. 예를 들어, 일부 WordPress 플러그 인은 인스턴스가 액세스할 수 있는 버킷에 액세스할 수 있습니다. 자세한 내용은 버킷에 대한 리소스 액세스 구성 및 자습서: WordPress 인스턴스에 버킷 연결을 참조하세요.



그러나 애플리케이션이 Lightsail 인스턴스에 없는 경우 버킷 액세스 키를 생성하고 구성할 수 있습니다. 버킷 액세스 키는 자동으로 교체되지 않는 장기 자격 증명입니다. 자세한 내용은 Lightsail 객체 스토리지 버킷 액세스 키 생성 단원을 참조하십시오.



버킷 액세스 키 교체

버킷당 최대 2개의 액세스 키를 사용할 수 있습니다. 동시에 두 개의 서로 다른 액세스 키를 가질 수 있지만 키 교체 시간 외에는 버킷에 대해 한 번에 하나의 액세스 키만 생성하는 것이 좋습니다. 이 접근 방식을 사용하면 버킷 액세스 키를 사용할 가능성 없이 언제든지 새 버킷 액세스 키를 생성할 수 있습니다. 예를 들어 교체를 위해 두 번째 액세스 키를 생성하는 것은 기존 시크릿 액세스 키가 복사, 손실 또는 손상되어 기존 액세스 키를 교체해야 하는 경우에 유용합니다.

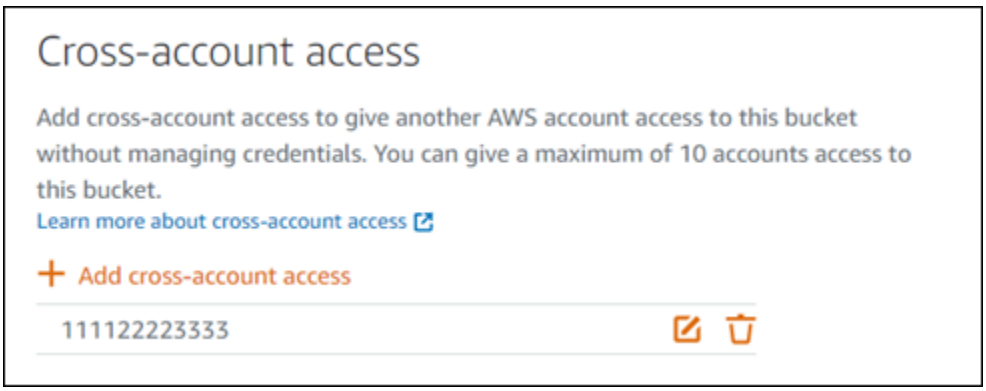
버킷에 액세스 키를 사용하는 경우 주기적으로 키를 교체하고 기존 키의 목록을 만들어야 합니다. 액세스 키가 마지막으로 사용된 날짜와 해당 키가 사용된 AWS 리전 이 키 사용 방법에 대한 기대치와 일치하는지 확인합니다. 액세스 키가 마지막으로 사용된 날짜가 Lightsail 콘솔에서 버킷 관리 페이지의 권한 탭에 있는 액세스 키 섹션에 표시됩니다. 사용되지 않는 액세스 키를 삭제합니다.

액세스 키를 교체하려면 새 액세스 키를 생성하고 소프트웨어에서 구성하여 테스트한 다음 이전에 사용된 애겟스 키를 삭제합니다. 액세스 키를 삭제하면 키가 영구 삭제되어 복원할 수 없습니다. 이 키는

새 액세스 키로만 교체할 수 있습니다. 자세한 내용은 Lightsail 객체 스토리지 버킷 액세스 키 생성 및 Lightsail 객체 스토리지 버킷에 대한 액세스 키 삭제 섹션을 참조하세요.

교차 계정 액세스를 사용하여 다른 AWS 계정에 버킷의 객체에 대한 액세스 권한 부여

교차 계정 액세스를 사용하여 버킷과 해당 객체를 공개하지 않고도 AWS 계정이 있는 특정 개인이 버킷의 객체에 액세스할 수 있도록 할 수 있습니다. 교차 계정 액세스를 구성한 경우 나열된 계정 ID가 버킷의 객체에 대한 액세스 권한을 부여하려는 올바른 계정인지 확인합니다. 자세한 내용은 버킷에 대한 크로스 계정 액세스 구성을 참조하세요.



데이터의 암호화

Lightsail은 Amazon 관리 키로 서버 측 암호화를 수행하고 HTTPS(TLS)를 시행하여 전송 중인 데이터를 암호화합니다. 서버 측 암호화를 사용하면 별도의 서비스에 저장되는 키로 데이터를 암호화하여 데이터에 대한 위험을 줄일 수 있습니다. 또한 전송 중인 데이터의 암호화를 사용하면 잠재적 공격자가 중간자 공격 또는 유사한 공격을 사용하여 네트워크 트래픽을 도청하거나 조작하는 것을 방지할 수 있습니다.

버전 관리 사용

버전 관리는 동일 버킷 내에 여러 개의 객체 변형을 보유하는 것을 의미합니다. 버전 관리를 사용하여 Lightsail 버킷에 저장된 모든 버전의 객체를 모두 보존, 검색 및 복원할 수 있습니다. 또한 의도치 않은 사용자 작업 및 애플리케이션 장애로부터 쉽게 복구할 수 있습니다. 자세한 내용은 버킷의 객체 버전 사용 설정 및 사용 중지를 참조하세요.

모니터링 및 감사 모범 사례

다음 모범 사례는 Lightsail 버킷에 대한 잠재적인 보안 약점 및 사고를 감지하는 데 도움이 될 수 있습니다.

액세스 로깅 활성화 및 주기적인 보안 및 액세스 감사 수행

액세스 로깅은 버킷에 대한 요청에 대한 자세한 레코드를 제공합니다. 이 정보에는 요청 유형(GET, PUT), 요청에 지정된 리소스 및 요청이 처리된 시간 및 날짜가 포함될 수 있습니다. 버킷에 대한 액세스 로깅을 활성화하고 주기적으로 보안 및 액세스 감사를 수행하여 버킷에 액세스하는 엔터티를 식별합니다. 기본적으로 Lightsail는 버킷에 대한 액세스 로그를 수집하지 않습니다. 액세스 로깅을 수동으로 활성화해야 합니다. 자세한 내용은 버킷 액세스 로그 및 버킷 액세스 로깅 활성화를 참조하세요.

Lightsail 버킷 식별, 태그 지정 및 감사

IT 자산 식별은 거버넌스와 보안의 중요한 측면입니다. 보안 태세를 평가하고 잠재적인 취약 영역에 대한 조치를 취하려면 모든 Lightsail 버킷에 대한 가시성을 확보해야 합니다.

태그 지정을 사용하여 보안에 민감한 리소스 또는 감사에 민감한 리소스를 식별한 다음 이러한 리소스를 검색해야 할 때 해당 태그를 사용합니다. 자세한 내용은 태그를 참조하세요.

모니터링 도구를 사용하여 AWS 모니터링 구현

모니터링은 Lightsail 버킷 및 기타 리소스의 안정성, 보안, 가용성 및 성능을 유지 관리하는 데 중요한 부분입니다. Lightsail에서 버킷 크기(BucketSizeBytes) 및 Number of objects(NumberOfObjects) 버킷 메트릭에 대한 알림 경보를 모니터링하고 생성할 수 있습니다. 예를 들어, 버킷의 크기가 특정 크기로 증가 또는 축소될 때 또는 버킷의 객체 수가 특정 숫자까지 올라가거나 줄어들 때 알림을 받을 수 있습니다. 자세한 내용은 버킷 지표 경보 생성을 참조하세요.

사용 AWS CloudTrail

AWS CloudTrail 는에서 사용자, 역할 또는 AWS 서비스가 수행한 작업의 레코드를 제공합니다 Lightsail. CloudTrail에서 수집한 정보를 사용하여 Lightsail에 전송된 요청, 요청이 발신된 IP 주소, 요청을 전송한 사용자, 요청이 전송된 시간 및 추가 세부 정보를 확인할 수 있습니다. 예를 들어 데이터 액세스에 영향을 주는 작업(특히 CreateBucketAccessKey, GetBucketAccessKeys, DeleteBucketAccessKey, SetResourceAccessForBucket 및 UpdateBucket)의 CloudTrail 항목을 식별할 수 있습니다. AWS 계정을 설정하면 CloudTrail이 기본적으로 활성화됩니다. CloudTrail 콘솔에서 최근 이벤트를 볼 수 있습니다. Lightsail 버킷에 대한 활동 및 이벤트에 대한 지속적인 레코드를 생성하려면 CloudTrail 콘솔에서 추적을 생성할 수 있습니다. 자세한 내용은 AWS CloudTrail 사용 설명서의 추적을 위해 데이터 이벤트 로깅을 참조하세요.

AWS 보안 권고 모니터링

AWS 계정에 등록된 기본 이메일 주소를 적극적으로 모니터링합니다. AWS 는이 이메일 주소를 사용하여 사용자에게 영향을 미칠 수 있는 새로운 보안 문제에 대해 연락을 드릴 것입니다.

AWS 광범위한 영향을 미치는 운영 문제는 AWS 서비스 상태 대시보드에 게시됩니다. Personal Health Dashboard를 통해 개별 계정에도 운영 문제가 게시됩니다. 자세한 내용은 AWS Health 설명서를 참조하세요.

Lightsail 버킷 및 객체에 대한 액세스 제어

기본적으로 모든 Amazon Lightsail 객체 스토리지 리소스(버킷 및 객체)는 프라이빗입니다. 즉, 버킷 소유자(버킷을 생성한 Lightsail 계정)만 버킷과 해당 객체에 액세스할 수 있습니다. 버킷 소유자는 필요에 따라 다른 사용자에게 액세스 권한을 부여할 수 있습니다. 버킷 및 해당 객체에 액세스할 권한을 부여하는 방법은 다음과 같습니다.

- 읽기 전용 액세스 – 다음 옵션은 버킷의 URL(예: `https://amzn-s3-demo-bucket.us-east-1.amazonaws.com/media/sailbot.jpg`)을 통해 버킷 및 해당 객체에 대한 읽기 전용 액세스를 제어합니다.
  - 버킷 액세스 권한 – 버킷 액세스 권한을 사용하여 인터넷상의 누구나 버킷의 모든 객체에 액세스할 권한을 부여합니다. 자세한 내용은 가이드 후반부의 버킷 액세스 권한을 참조하세요.
  - 개별 객체 액세스 권한 – 개별 객체 액세스 권한을 사용하여 인터넷상의 누구나 버킷의 개별 객체에 액세스할 권한을 부여합니다. 자세한 내용은 가이드 후반부의 개별 객체 액세스 권한을 참조하세요.
  - 교차 계정 액세스 - 교차 계정 액세스를 사용하여 다른 AWS 계정의 버킷에 있는 모든 객체에 대한 액세스 권한을 부여합니다. 자세한 내용은 가이드 후반부의 교차 계정 액세스를 참조하세요.
- 읽기 및 쓰기 액세스 – 다음 옵션은 버킷과 버킷 객체에 대한 전체 읽기 및 쓰기 액세스를 제어합니다. (AWS CLI), AWS APIs 및 AWS SDKs와 AWS Command Line Interface 함께 이러한 옵션을 사용합니다.
  - 액세스 키 – 액세스 키를 사용하여 애플리케이션 또는 플러그 인에 액세스할 권한을 부여합니다. 자세한 내용은 가이드 후반부의 액세스 키를 참조하세요.
  - 리소스 액세스 – 리소스 액세스를 사용하여 Lightsail 인스턴스에 액세스할 권한을 부여합니다. 자세한 내용은 가이드 후반부의 리소스 액세스를 참조하세요.
- Amazon Simple Storage Service 퍼블릭 액세스 차단 – Amazon Simple Storage Service(S3) 계정 수준 퍼블릭 액세스 차단 기능을 사용하여 Amazon S3 및 Lightsail의 버킷에 대한 퍼블릭 액세스를 중앙에서 제한합니다. 구성되었을 수도 있는 개별 버킷 및 객체 권한과 관계없이 퍼블릭 액세스 차단을 통해 모든 Amazon S3 버킷과 Lightsail 버킷을 프라이빗으로 만들 수 있습니다. 자세한 내용은 이 설명서 후반부의 Amazon S3 퍼블릭 액세스 차단을 참조하세요.

버킷에 대한 자세한 내용은 객체 스토리지를 참조하세요. 보안 모범 사례에 대한 자세한 내용은 객체 스토리지에 대한 보안 모범 사례를 참조하세요.

버킷 액세스 권한

버킷에 있는 객체에 대한 인증되지 않은 공개 읽기 전용 액세스를 제어하려면 버킷 액세스 권한을 사용하면 됩니다. 버킷 액세스 권한을 구성할 때 다음 옵션 중 하나를 선택할 수 있습니다.

- 모든 객체 비공개(All objects are private) - 사용자 본인이나 사용자에게 액세스 권한을 부여받은 사람만 버킷에 있는 모든 객체를 읽을 수 있습니다. 이 옵션을 사용하면 개별 객체를 공개(읽기 전용)할 수 없습니다.
- 개별 객체 공개 가능(읽기 전용)(Individual objects can be made public (read-only)) – 개별 객체를 공개(읽기 전용)로 지정하지 않는 한 사용자 본인이나 사용자에게 액세스 권한을 부여받은 사람만 버킷에 있는 객체를 읽을 수 있습니다. 이 옵션을 사용하면 개별 객체를 공개(읽기 전용)할 수 있습니다. 자세한 내용은 가이드 후반부의 개별 객체 액세스 권한을 참조하세요.
- 모든 객체 공개(읽기 전용)(All objects are public (read-only)) – 인터넷상의 모든 사용자가 버킷에 있는 객체를 전부 읽을 수 있습니다. 이 옵션을 선택하면 버킷의 URL(예: `https://amzn-s3-demo-bucket.us-east-1.amazonaws.com/media/sailbot.jpg`)을 통해 인터넷상의 누구나 버킷의 모든 객체를 읽을 수 있습니다.

버킷 액세스 권한을 구성하는 방법에 대한 자세한 내용은 버킷 액세스 권한 구성을 참조하세요.

개별 객체 액세스 권한

버킷에 있는 개별 객체에 대한 인증되지 않은 공개 읽기 전용 액세스를 제어하려면 개별 객체 액세스 권한을 사용하면 됩니다. 개별 객체 액세스 권한은 버킷의 버킷 액세스 권한을 통해 개별 객체를 공개(읽기 전용)할 수 있는 경우에만 구성할 수 있습니다. 개별 객체에 대한 액세스 권한을 구성할 때 다음 옵션 중 하나를 선택할 수 있습니다.

- 비공개(Private) - 사용자 본인이나 사용자에게 액세스 권한을 부여받은 사람만 객체를 읽을 수 있습니다.
- 공개(읽기 전용)(Public (read-only)) - 인터넷에서 누구나 객체를 읽을 수 있습니다. 버킷의 URL(예: `https://amzn-s3-demo-bucket.us-east-1.amazonaws.com/media/sailbot.jpg`)을 통해 인터넷상의 모든 사용자가 개별 객체를 읽을 수 있게 됩니다.

개별 객체 액세스 권한을 구성하는 방법에 대한 자세한 내용은 버킷에 있는 개별 객체에 대한 액세스 권한 구성을 참조하세요.

크로스 계정 액세스

교차 계정 액세스를 사용하여 다른 AWS 계정 및 해당 사용자의 버킷에 있는 모든 객체에 인증된 읽기 전용 액세스 권한을 부여합니다. 교차 계정 액세스는 다른 AWS 계정과 객체를 공유하려는 경우에 적합합니다. 다른 AWS 계정에 크로스 계정 액세스 권한을 부여하면 해당 계정의 사용자는 버킷(예: `https://amzn-s3-demo-bucket.us-east-1.amazonaws.com/media/sailbot.jpg`)의 URL을 통해 버킷에 있는 객체에 읽기 전용 권한으로 액세스할 수 있게 됩니다. 최대 10개의 AWS 계정에 대한 액세스 권한을 부여할 수 있습니다.

크로스 계정 액세스 권한을 구성하는 방법에 대한 자세한 내용은 버킷의 크로스 계정 액세스 권한 구성을 참조하세요.

액세스 키

액세스 키를 사용하여 버킷 및 해당 객체에 대한 전체 읽기 및 쓰기 액세스 권한을 부여하는 자격 증명 집합을 생성할 수 있습니다. 액세스 키는 액세스 키 ID와 비밀 액세스 키가 한 세트로 구성됩니다. 버킷당 최대 2개의 액세스 키를 사용할 수 있습니다. AWS APIs 및 AWS SDKs. AWS CLI에서 액세스 키를 구성할 수도 있습니다.

액세스 키를 생성하는 방법에 대한 자세한 내용은 버킷의 액세스 키 생성을 참조하세요.

리소스 액세스

리소스 액세스를 사용하여 Lightsail 인스턴스의 버킷과 객체에 전체 읽기 및 쓰기 액세스 권한을 부여할 수 있습니다. 리소스 액세스를 사용하면 액세스 키와 같은 자격 증명을 관리할 필요가 없습니다. 인스턴스에 액세스 권한을 부여하려면 인스턴스를 동일한 AWS 리전의 버킷에 연결합니다. 액세스를 거부하려면 버킷에서 인스턴스를 분리합니다. 인스턴스의 애플리케이션이 버킷의 파일을 프로그래밍 방식으로 업로드하고 액세스하도록 구성하는 경우 리소스 액세스 권한을 사용하면 좋습니다. 이러한 사용 사례 중 하나는 미디어 파일을 버킷에 저장하도록 WordPress 인스턴스를 구성하는 것입니다. 자세한 내용은 자습서: WordPress 인스턴스에 버킷 연결 및 자습서: 콘텐츠 전송 네트워크 배포와 함께 버킷 사용을 참조하세요.

리소스 액세스 권한을 구성하는 방법에 대한 자세한 내용은 버킷의 리소스 액세스 권한 구성을 참조하세요.

Amazon S3 퍼블릭 액세스 차단

Amazon S3 퍼블릭 액세스 차단 기능을 사용하여 Amazon S3 및 Lightsail의 버킷에 대한 퍼블릭 액세스를 중앙에서 제한합니다. 구성되었을 수도 있는 개별 버킷 및 객체 권한과 관계없이 퍼블릭 액세스 차단을 통해 모든 Amazon S3 버킷과 Lightsail 버킷을 프라이빗으로 만들 수 있습니다. Amazon S3 콘

솔, AWS CLI, AWS SDKs 및 REST API를 사용하여 Lightsail 객체 스토리지 서비스의 버킷을 포함하여 계정의 모든 버킷에 대한 퍼블릭 액세스 차단 설정을 구성할 수 있습니다. 자세한 내용은 버킷에 대한 퍼블릭 액세스 차단을 참조하세요.

Lightsail 객체 스토리지 버킷에 파일을 업로드합니다.

Amazon Lightsail 객체 스토리지 서비스의 버킷에 파일을 업로드하면 해당 파일이 객체로 저장됩니다. 객체는 파일 데이터 및 객체를 설명하는 메타데이터로 구성됩니다. 버킷에 저장할 수 있는 객체 수에는 제한이 없습니다.

이미지, 백업, 데이터, 동영상 등 모든 유형의 파일을 버킷에 업로드할 수 있습니다. Lightsail 콘솔을 사용하여 업로드할 수 있는 최대 파일 크기는 2GB입니다. 더 큰 파일을 업로드하려면 Lightsail API, AWS Command Line Interface (AWS CLI) 또는 AWS SDKs 사용합니다.

Lightsail에서는 업로드할 파일의 크기에 따라 다음 옵션을 제공합니다.

- Lightsail 콘솔을 사용하여 최대 2GB 크기의 객체 업로드 - Lightsail 콘솔로 최대 2GB 크기의 단일 객체를 업로드할 수 있습니다. 자세한 내용은 가이드 후반부의 Lightsail 콘솔을 사용하여 버킷에 파일 업로드를 참조하세요.
- AWS SDKs, REST API 또는를 사용하는 단일 작업으로 최대 5GB 크기의 객체 업로드 AWS CLI - 단일 PUT 작업으로 최대 5GB 크기의 단일 객체를 업로드할 수 있습니다. 자세한 내용은 가이드 후반부의 AWS CLI를 사용하여 버킷에 파일 업로드를 참조하세요.
- AWS SDKs, REST API 또는 - 멀티파트 업로드 API를 사용하여 AWS CLI5MB~50TB 크기의 큰 객체 하나를 업로드할 수 있습니다. 멀티파트 업로드 API는 대용량 객체의 업로드 경험을 개선하기 위해 디자인되었습니다. 객체를 부분별로 업로드할 수 있습니다. 이러한 객체 부분은 임의의 순서로 독립적으로, 그리고 병렬적으로 업로드할 수 있습니다. 자세한 내용은 멀티파트 업로드를 사용하여 버킷으로 파일 업로드를 참조하세요.

버킷에 대한 자세한 내용은 객체 스토리지를 참조하세요.

객체 키 이름 및 버전 관리

Lightsail 콘솔을 사용하여 파일을 업로드하면 파일 이름이 객체 키 이름으로 사용됩니다. 객체 키(또는 키 이름)는 버킷에 저장된 객체를 고유하게 식별합니다. 파일이 업로드되는 폴더(있는 경우)가 키 이름 접두사로 사용됩니다. 예를 들어, `sailbot.jpg` 파일을 `images` 버킷의 폴더에 업로드하면 전체 객체 키 이름과 접두사가 `images/sailbot.jpg`가 됩니다. 그러나 콘솔에는 해당 객체가 `sailbot.jpg` 폴더의 `images`로 표시됩니다. 객체 키 이름에 대한 자세한 내용은 객체 스토리지 버킷의 키 이름을 참조하세요.

Lightsail 콘솔을 사용하여 디렉터리를 업로드하면 디렉터리의 모든 파일과 하위 폴더가 버킷에 업로드됩니다. 그런 다음 Lightsail이 업로드된 각 파일 이름과 폴더 이름의 조합인 객체 키 이름을 할당합니다. 예를 들어, `sample1.jpg` 및 `sample2.jpg`라는 2개의 파일이 들어 있는 `images` 폴더를 업로드하는 경우 Lightsail에서 파일을 업로드한 다음 해당 키 이름인 `images/sample1.jpg` 및 `images/sample2.jpg`를 할당합니다. 객체는 콘솔에 `images` 폴더의 `sample1.jpg` 및 `sample2.jpg`로 표시됩니다.

이미 존재하는 키 이름으로 파일을 업로드하고 버킷에 버전 관리를 활성화하지 않으면 새로 업로드된 객체가 이전 객체를 대체합니다. 그러나 버킷에서 버전 관리를 활성화한 경우 Lightsail에서는 기존 객체를 교체하지 않고 새 버전의 객체를 생성합니다. 자세한 내용은 버킷의 객체 버전 사용 설정 및 사용 중지를 참조하세요.

Lightsail 콘솔을 사용하여 버킷에 파일 업로드

Lightsail 콘솔을 사용하여 파일 및 디렉터리를 업로드하려면 다음 절차를 완료하세요.

1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 스토리지를 선택합니다.
3. 파일 및 폴더를 업로드하려는 버킷의 이름을 선택합니다.
4. 객체(Objects) 탭에서 다음 작업 중 하나를 수행합니다.

   - 파일 및 폴더를 객체(Objects) 페이지로 끌어다 놓습니다.
   - 업로드(Upload)와 파일(File)을 차례로 선택하여 개별 파일을 업로드하거나 디렉터리(Directory)를 선택하여 폴더와 폴더의 모든 콘텐츠를 업로드합니다.

     > **Note**
     >
     > 새 폴더 생성(Create new folder)을 선택하여 폴더를 생성할 수도 있습니다. 그런 다음 새 폴더를 찾아 파일을 업로드할 수 있습니다.

   업로드가 완료되면 업로드 성공 메시지가 표시됩니다.

를 사용하여 버킷에 파일 업로드 AWS CLI

AWS Command Line Interface (AWS CLI)를 사용하여 버킷에 파일과 폴더를 업로드하려면 다음 절차를 완료하세요. `put-object` 명령을 사용하여 이 작업을 수행할 수 있습니다. 자세한 내용은 AWS CLI 명령 레퍼런스의 put-object를 참조하세요.

> **Note**
>
> 이 절차를 계속하기 전에 AWS CLI 및 Amazon S3에 대해 Lightsail를 설치하고 구성해야 합니다. 자세한 내용은 에서 작동 AWS CLI 하도록 구성을 Lightsail참조하세요.

1. 터미널 또는 명령 프롬프트 창을 엽니다.
2. 다음 명령을 입력하여 버킷에 파일을 업로드합니다.

```
aws s3api put-object --bucket BucketName --key ObjectKey --body LocalDirectory --
acl bucket-owner-full-control
```

명령에서 아래 예 텍스트를 사용자의 값으로 대체합니다.

- *BucketName*을 파일을 업로드하려는 버킷의 이름으로 바꿉니다.
- *ObjectKey*를 버킷에 있는 객체의 전체 객체 키로 바꿉니다.
- *LocalDirectory*를 업로드할 파일의 컴퓨터에 있는 로컬 디렉터리 폴더 경로로 바꿉니다.

예제:

- Linux 또는 Unix 컴퓨터:

```
aws s3api put-object --bucket amzn-s3-demo-bucket --key images/sailbot.jpg --
body home/user/Pictures/sailbot.jpg --acl bucket-owner-full-control
```

- Windows 컴퓨터:

```
aws s3api put-object --bucket amzn-s3-demo-bucket --key images/sailbot.jpg --
body "C:\Users\user\Pictures\sailbot.jpg" --acl bucket-owner-full-control
```

다음 예와 비슷한 결과가 나타나야 합니다.

```
C:\>aws s3api put-object --bucket DOC-EXAMPLE-BUCKET --key images/sailbot.jpg --body "C:\Users\user\Pictures\sailbot.jpg"
{
    "ETag": "\"694d34edexampled92d64f342aa234c3\""
}
```

IPv6 전용 요청에 대한 AWS CLI 구성

Amazon S3는 IPv6를 통한 버킷 액세스를 지원합니다. 듀얼 스택 엔드포인트를 사용하여 IPv6을 통해 Amazon S3 API 호출로 요청합니다. 이 섹션에서는 IPv6를 넘어 듀얼 스택 엔드포인트에 요청하는 데 사용되는 방법을 보여 줍니다. 자세한 내용은 Amazon S3 사용 설명서의 Amazon S3 듀얼 스택 엔드포인트 사용을 참조하세요. 설정 지침은 에서 작동 AWS Command Line Interface 하도록 구성을 Amazon Lightsail AWS CLI참조하세요.

> **Important**
>
> 버킷에 액세스하는 클라이언트와 네트워크가 IPv6을 사용하도록 설정되어 있어야 합니다. 자세한 내용은 IPv6 도달 가능성을 참조하세요.

IPv6 전용 인스턴스에서 S3를 요청하는 방법에는 두 가지가 있습니다. 모든 Amazon S3 요청을 지정된의 듀얼 스택 엔드포인트로 전달 AWS CLI 하도록를 구성할 수 있습니다 AWS 리전. 또는 지정된 AWS CLI 명령에만 듀얼 스택 엔드포인트를 사용하려는 경우(모든 명령이 아님) 모든 명령에 S3 듀얼 스택 엔드포인트를 추가할 수 있습니다.

구성 AWS CLI

AWS Config 파일의 프로파일에서 구성 값을 `use_dualstack_endpoint`로 `true` 설정하여 Amazon S3 및 s3api AWS CLI 명령으로 이루어진 모든 Amazon S3 요청을 지정된 리전의 듀얼 스택 엔드포인트로 보냅니다. config 파일 또는 AWS CLI --region 옵션을 사용하여 명령에서 리전을 지정합니다.

다음 명령을 입력하여 AWS CLI를 구성합니다.

```
aws configure set default.s3.use_dualstack_endpoint true
```

```
aws configure set default.s3.addressing_style virtual
```

특정 명령에 이중 스택 엔드포인트 추가

아무 s3 또는 s3api 명령에 대해 `--endpoint-url` 파라미터를 `https://s3.dualstack.`*`aws-region`*`.amazonaws.com` 또는 `http://s3.dualstack.`*`aws-region`*`.amazonaws.com`으로 설정하여 명령별로 이중 스택 엔드포인트를 사용할 수 있습니다. 아래 예제에서 *`bucketname`*과 *`aws-region`*을 버킷 및 AWS 리전의 이름으로 바꿉니다.

```
aws s3api list-objects --bucket bucketname --endpoint-url https://s3.dualstack.aws-region.amazonaws.com
```

Lightsail에서 버킷 및 객체 관리

다음은 Lightsail 객체 스토리지 버킷을 관리하는 일반적인 단계입니다.

1. Amazon Lightsail 객체 스토리지 서비스의 객체와 버킷에 대해 알아보세요. 자세한 내용은 Amazon Lightsail의 객체 스토리지를 참조하세요.
2. Amazon Lightsail에서 버킷에 지정할 수 있는 이름에 대해 알아봅니다. 자세한 내용은 Amazon Lightsail의 버킷 이름 지정 규칙을 참조하세요.
3. 버킷을 생성하여 Lightsail 객체 스토리지 서비스를 시작합니다. 자세한 내용은 Amazon Lightsail에서 버킷 생성을 참조하세요.
4. 버킷의 보안 모범 사례와 버킷에 구성할 수 있는 액세스 권한에 대해 알아봅니다. 버킷의 모든 객체를 공개 또는 비공개로 설정하거나 개별 객체를 공개로 설정하도록 선택할 수 있습니다. 액세스 키를 생성하고, 버킷에 인스턴스를 연결하고, 다른 AWS 계정에 액세스 권한을 부여하여 버킷에 대한 액세스 권한을 부여할 수도 있습니다. 자세한 내용은 Amazon Lightsail 객체 스토리지에 대한 보안 모범 사례 및 Amazon Lightsail의 버킷 권한 이해를 참조하세요.

   버킷 액세스 권한에 대해 알아본 후 다음 가이드를 참조하여 버킷에 대한 액세스 권한을 부여합니다.

   - Amazon Lightsail의 버킷에 대한 퍼블릭 액세스 차단
   - Amazon Lightsail에서 버킷 액세스 권한 구성
   - Amazon Lightsail에서 버킷의 개별 객체에 대한 액세스 권한 구성
   - Amazon Lightsail에서 버킷의 액세스 키 생성
   - Amazon Lightsail에서 버킷에 대한 리소스 액세스 구성
   - Amazon Lightsail에서 버킷의 크로스 계정 액세스 구성
5. 버킷에 대한 액세스 로깅을 활성화하는 방법과 액세스 로그를 사용하여 버킷의 보안을 감사하는 방법에 대해 알아봅니다. 자세한 내용은 다음 안내서를 참조하세요.

   - Amazon Lightsail 객체 스토리지 서비스의 버킷에 대한 액세스 로깅
   - Amazon Lightsail 객체 스토리지 서비스에서 버킷에 대한 액세스 로그 형식
   - Amazon Lightsail 객체 스토리지 서비스에서 버킷에 대한 액세스 로깅 활성화
   - Amazon Lightsail에서 버킷에 대한 액세스 로그 사용하여 요청을 식별

6. Lightsail에서 버킷을 관리할 수 있는 기능을 사용자에게 부여하는 IAM 정책을 생성합니다. 자세한 내용은 Amazon Lightsail에서 버킷을 관리하기 위한 IAM 정책을 참조하세요.
7. 버킷의 객체에 라벨을 지정하고 식별하는 방식에 대해 알아봅니다. 자세한 내용은 Amazon Lightsail에서 객체 키 이름 이해를 참조하세요.
8. 파일을 업로드하고 버킷의 객체를 관리하는 방법에 대해 알아봅니다. 자세한 내용은 다음 안내서를 참조하세요.
   - Amazon Lightsail에서 버킷으로 파일 업로드
   - 멀티파트 업로드를 사용하여 Amazon Lightsail에서 버킷으로 파일 업로드
   - Amazon Lightsail에서 버킷의 객체 확인
   - Amazon Lightsail에서 버킷의 객체 복사 또는 이동
   - Amazon Lightsail에서 버킷의 객체 다운로드
   - Amazon Lightsail에서 버킷의 객체 필터링
   - Amazon Lightsail에서 버킷의 객체 태그 지정
   - Amazon Lightsail에서 버킷의 객체 삭제
9. 객체 버전 관리를 활성화하면 버킷에 저장된 모든 버전의 객체를 모두 보존, 검색 및 복원할 수 있습니다. 자세한 내용은 Amazon Lightsail에서 버킷의 객체 버전 사용 설정 및 사용 중지를 참조하세요.
10. 객체 버전 관리를 활성화한 후 버킷에 있는 객체의 이전 버전을 복원할 수 있습니다. 자세한 내용은 Amazon Lightsail에서 버킷의 이전 객체 버전 복원을 참조하세요.
11. 버킷 사용률을 모니터링합니다. 자세한 내용은 Amazon Lightsail에서 버킷에 대한 지표 보기를 참조하세요.
12. 버킷 사용률이 임곗값을 초과할 때 알림을 받도록 버킷 지표에 대한 경보를 구성합니다. 자세한 내용은 Amazon Lightsail에서 버킷 지표 경보 생성을 참조하세요.
13. 스토리지 및 네트워크 전송이 부족한 경우 버킷의 스토리지 플랜을 변경합니다. 자세한 내용은 Amazon Lightsail에서 버킷 플랜 변경을 참조하세요.
14. 버킷을 다른 리소스에 연결하는 방법에 대해 알아봅니다. 자세한 내용은 다음 자습서를 참조하세요.
   - 자습서: WordPress 인스턴스를 Amazon Lightsail 버킷에 연결
   - 자습서: Lightsail 콘텐츠 전송 네트워크 배포와 함께 Amazon Lightsail 버킷 사용
15. 버킷을 더 이상 사용하지 않는 경우 삭제할 수 있습니다. 자세한 내용은 Amazon Lightsail에서 버킷 삭제를 참조하세요.

Lightsail의 교차 오리진 리소스 공유(CORS)

교차 오리진 리소스 공유(CORS) 한 도메인에서 로드되어 다른 도메인에 있는 리소스와 상호 작용하는 클라이언트 웹 애플리케이션에 대한 방법을 정의합니다. CORS 지원을 통해 Lightsail 객체 스토리지로 강력한 클라이언트 측 웹 애플리케이션을 구축하고, 버킷 리소스에 대한 교차 오리진 액세스를 선택적으로 허용할 수 있습니다. CORS에 대한 자세한 내용은 CORS란 무엇인가요?를 참조하세요.

이 섹션은 AWS Command Line Interface(AWS CLI)를 사용하여 Lightsail 버킷에 대해 CORS를 구성하는 방법을 보여줍니다. 교차 오리진 요청을 허용하도록 버킷을 구성하려면, 버킷에 대한 액세스를 허용할 오리진, 각 오리진에 대해 지원할 작업(HTTP 메서드) 및 기타 작업 특정 정보를 식별하는 규칙을 정의하는 JSON 문서를 사용하여 버킷에 CORS 구성을 추가합니다.

주제

- CORS 사용 사례
- Lightsail이 CORS 구성을 평가하는 방법
- AWS CLI를 사용하여 CORS 구성
- CORS 문제 해결

CORS 사용 사례

다음 예제 시나리오는 Lightsail 버킷에 CORS를 구성할 필요가 있는 방법을 설명합니다.

시나리오: 웹 글꼴 호스팅

Lightsail 버킷에서 웹 글꼴을 호스팅한다고 가정해 보겠습니다. 브라우저는 웹 글꼴을 불러오기 위해 CORS 검사(사전 검사라고도 함)가 필요합니다. 웹 글꼴을 호스팅하는 버킷이 이러한 요청을 하는 오리진을 허용하도록 구성합니다.

Lightsail이 CORS 구성을 평가하는 방법

Lightsail 객체 스토리지는 브라우저에서 사전 요청을 받으면 버킷에 대한 CORS 구성을 평가하여, 수신된 브라우저 요청과 일치하는 첫 번째 CORS 규칙을 사용하여 교차 오리진 요청을 허용합니다. 규칙이 일치하려면 다음 조건이 충족되어야 합니다.

- 요청의 `Origin` 헤더는 `AllowedOrigins` 요소의 오리진과 일치해야 합니다.
- `Access-Control-Request-Method` 헤더에 지정된 HTTP 메서드는 `AllowedMethods` 요소의 메서드와 일치해야 합니다.

- `Access-Control-Request-Headers` 헤더에 나열된 헤더는 `AllowedHeaders` 요소의 헤더와 일치해야 합니다.

> **Note**
>
> 버킷에 CORS를 활성화하면 버킷 권한이 계속 적용됩니다. CORS 구성은 브라우저에서 교차 오리진 요청이 진행되도록 허용할지 여부만 결정합니다. 자세한 내용은 Lightsail 버킷 및 객체에 대한 액세스 제어 섹션을 참조하세요.

CORS 구성의 요소

CORS 구성은 CORS 규칙 배열을 포함하는 JSON 문서입니다. 각 규칙은 버킷에 대한 액세스가 허용되는 오리진, 허용되는 HTTP 메서드 및 기타 구성 옵션을 정의합니다.

다음 요소를 CORS 규칙에 포함할 수 있습니다.

allowedOrigins

버킷에 대한 액세스가 허용되는 오리진을 지정합니다. 와일드카드(*)를 사용하여 모든 오리진을 허용하거나 `https://example.com`과 같은 특정 도메인을 지정할 수 있습니다.

allowedMethods

지정된 오리진에 대해 허용되는 HTTP 메서드를 지정합니다. 유효한 값에는 GET, PUT, POST, DELETE, HEAD가 포함됩니다.

allowedHeaders

Access-Control-Request-Headers 헤더를 통해 사전 OPTIONS 요청에 허용되는 헤더를 지정합니다.

exposeHeaders

클라이언트 애플리케이션에서 액세스할 수 있는 응답 헤더를 지정합니다.

id

CORS 규칙의 고유 식별자입니다.

maxAgeSeconds

브라우저가 사전 요청에 대한 응답을 캐싱할 수 있는 시간을 초 단위로 지정합니다.

이러한 파라미터에 대한 자세한 내용은 Amazon Lightsail API 레퍼런스의 BucketCorsRule을 참조하세요.

AWS CLI를 사용하여 CORS 구성

`--cors` 파라미터와 함께 AWS CLI를 사용하여 Lightsail 버킷에 대한 CORS를 구성할 수 있습니다. 이 파라미터는 CORS 구성이 포함된 JSON 파일을 수락합니다. CORS 구성의 요소에 대한 자세한 내용은 CORS 구성의 요소 섹션을 참조하세요.

주제

- CORS 구성 적용
- 예제 CORS 구성
- CORS 구성 제거

CORS 구성 적용

다음 절차는 JSON 파일을 지정하여 버킷에 CORS 구성을 적용할 수 있는 방법을 보여줍니다. 더 많은 예제 구성은 예제 CORS 구성 섹션을 참조하세요.

AWS CLI를 사용하여 버킷에 대한 CORS를 구성하려면

1. CORS 구성이 포함된 JSON 파일을 생성합니다. 예를 들어, 다음 콘텐츠가 포함된 `cors-config.json`이라는 파일을 생성합니다.

```
{
  "CORSRules": [
    {
      "AllowedOrigins": ["https://example.com"],
      "AllowedMethods": ["GET", "PUT", "POST"],
      "AllowedHeaders": ["*"],
      "MaxAgeSeconds": 3000
    }
  ]
}
```

2. AWS CLI를 사용하여 버킷에 CORS 구성을 적용합니다.

```
aws lightsail update-bucket --bucket-name amzn-s3-demo-bucket --cors file://cors-config.json
```

3. CORS 구성이 성공적으로 적용되었는지 확인합니다.

```
aws lightsail get-buckets --bucket-name amzn-s3-demo-bucket --include-cors
```

> **Note**
>
> *amzn-s3-demo-bucket*을 Lightsail 버킷 이름으로 바꿉니다.

예제 CORS 구성

다음 예제는 다양한 사용 사례에 대한 일반적인 CORS 구성을 보여줍니다.

예제 1: 모든 오리진 및 메서드 허용

이 구성을 사용하면 모든 오리진이 HTTP 메서드를 사용하여 버킷에 액세스할 수 있습니다.

```
{
    "CORSRules": [
      {
        "AllowedOrigins": ["*"],
        "AllowedMethods": ["GET", "PUT", "POST", "DELETE", "HEAD"],
        "AllowedHeaders": ["*"],
        "MaxAgeSeconds": 3000
      }
    ]
  }
```

예제 2: 특정 도메인으로 제한

이 구성은 `https://mywebsite.com`의 요청만 허용합니다.

```
{
    "CORSRules": [
      {
        "AllowedOrigins": ["https://mywebsite.com"],
        "AllowedMethods": ["GET", "PUT"],
        "AllowedHeaders": ["Authorization", "Content-Type"],
        "ExposeHeaders": ["ETag"],
        "MaxAgeSeconds": 3600
```

```
      }
    ]
  }
```

예제 3: 서로 다른 오리진에 대한 다중 규칙

이 구성은 서로 다른 오리진에 대해 서로 다른 규칙을 정의합니다.

```
{
    "CORSRules": [
      {
        "AllowedOrigins": ["https://mywebsite.com"],
        "AllowedMethods": ["GET", "PUT", "POST"],
        "AllowedHeaders": ["*"],
        "MaxAgeSeconds": 3600
      },
      {
        "AllowedOrigins": ["https://cdn.mywebsite.com"],
        "AllowedMethods": ["GET"],
        "AllowedHeaders": ["Authorization"],
        "MaxAgeSeconds": 86400
      }
    ]
  }
```

CORS 구성 제거

버킷에서 CORS 구성을 제거하려면 다음 AWS CLI 명령을 사용합니다.

```
aws lightsail update-bucket --bucket-name amzn-s3-demo-bucket --cors '{"rules":[]}'
```

> **Note**
>
> *amzn-s3-demo-bucket*을 Lightsail 버킷의 이름으로 바꿉니다.

CORS 구성을 제거한 후에는 브라우저에서 버킷에 대한 교차 오리진 요청을 차단합니다.

CORS 문제 해결

CORS에 문제가 있는 경우 다음을 확인하세요.

- CORS 구성 확인 - CORS 구성이 올바른 형식의 JSON이고 사용 사례에 필요한 규칙이 포함되어 있는지 확인합니다.
- 오리진 일치 확인 - 요청의 오리진은 `AllowedOrigins` 목록의 항목과 정확하게 일치해야 합니다. 프로토콜(http/https), 하위 도메인 및 포트가 정확하게 일치해야 합니다.
- HTTP 메서드 확인 - 사용하는 HTTP 메서드가 일치하는 규칙의 `AllowedMethods`에 나열되어 있는지 확인합니다.
- 브라우저 개발자 도구 확인 - 브라우저의 개발자 도구를 사용하여 사전 OPTIONS 요청 및 응답을 검사하여 CORS 관련 오류를 식별합니다.
- 버킷 권한 검증 - 버킷에 CORS 외에도 적절한 권한이 구성되어 있는지 확인합니다. CORS는 브라우저 기반 교차 오리진 액세스만 제어하며 버킷 수준 권한은 제어하지 않습니다.

문제를 해결하는 동안 버킷에서 CORS 구성을 제거해야 하는 경우 CORS 구성 제거 섹션을 참조하세요.

Amazon Lightsail에서 컨테이너 배포 및 관리

Amazon Lightsail 컨테이너 서비스는 컨테이너를 배포, 실행 및 관리할 수 있는 확장성이 뛰어난 컴퓨팅 및 네트워킹 리소스입니다. 컨테이너는 애플리케이션이 한 컴퓨팅 환경에서 다른 컴퓨팅 환경으로 빠르고 안정적으로 실행되도록 코드와 종속성을 함께 패키징하는 표준 소프트웨어 단위입니다.

Lightsail 컨테이너 서비스는 로컬 시스템에서 생성하고 서비스로 푸시하는 이미지 또는 Amazon ECR Public Gallery와 같은 온라인 리포지토리의 이미지를 사용하여 AWS 인프라에서 컨테이너를 실행할 수 있는 컴퓨팅 환경이라고 생각할 수 있습니다.

Docker와 같은 소프트웨어를 설치하여 로컬 시스템에서 로컬로 컨테이너를 실행할 수도 있습니다. Amazon Elastic Container Service(Amazon ECS) 및 Amazon Elastic Compute Cloud(Amazon EC2)는 컨테이너를 실행할 수 있는 AWS 인프라 내의 다른 리소스입니다. 자세한 내용은 Amazon ECS 개발자 가이드를 참조하세요.

목차

- 컨테이너
- Lightsail 컨테이너 서비스 요소
  - Lightsail 컨테이너 서비스
  - 컨테이너 서비스 용량(규모 및 성능)
  - 요금
  - 배포
  - 배포 버전
  - 컨테이너 이미지 소스
  - 컨테이너 서비스 ARN
  - 퍼블릭 엔드포인트 및 기본 도메인
  - 사용자 지정 도메인 및 SSL/TLS 인증서
  - 컨테이너 로그
  - Metrics
- Lightsail 컨테이너 서비스 사용

컨테이너

컨테이너는 애플리케이션이 한 컴퓨팅 환경에서 다른 컴퓨팅 환경으로 빠르고 안정적으로 실행되도록 코드와 종속성을 함께 패키징하는 표준 소프트웨어 단위입니다. 개발 환경에서 컨테이너를 실행하고 사전 프로덕션 환경에 배포한 다음, 프로덕션 환경에 배포할 수 있습니다. 개발 환경이 로컬 시스템인지, 사전 프로덕션 환경이 데이터 센터의 물리적 서버인지, 프로덕션 환경이 클라우드의 가상 프라이빗 서버인지에 관계없이 컨테이너가 안정적으로 실행됩니다.

컨테이너 이미지는 코드, 런타임, 시스템 도구, 시스템 라이브러리 및 설정과 같은 애플리케이션을 실행하는 데 필요한 모든 것이 포함된 가벼운 독립 실행형 소프트웨어 패키지입니다. 컨테이너 이미지는 런타임 시 컨테이너가 됩니다. 애플리케이션과 종속성을 컨테이너화하면 소프트웨어가 배포된 운영 체제 및 인프라에서 제대로 실행되는지를 더 이상 걱정할 필요가 없어 코드에 더 많은 시간을 할애할 수 있습니다.

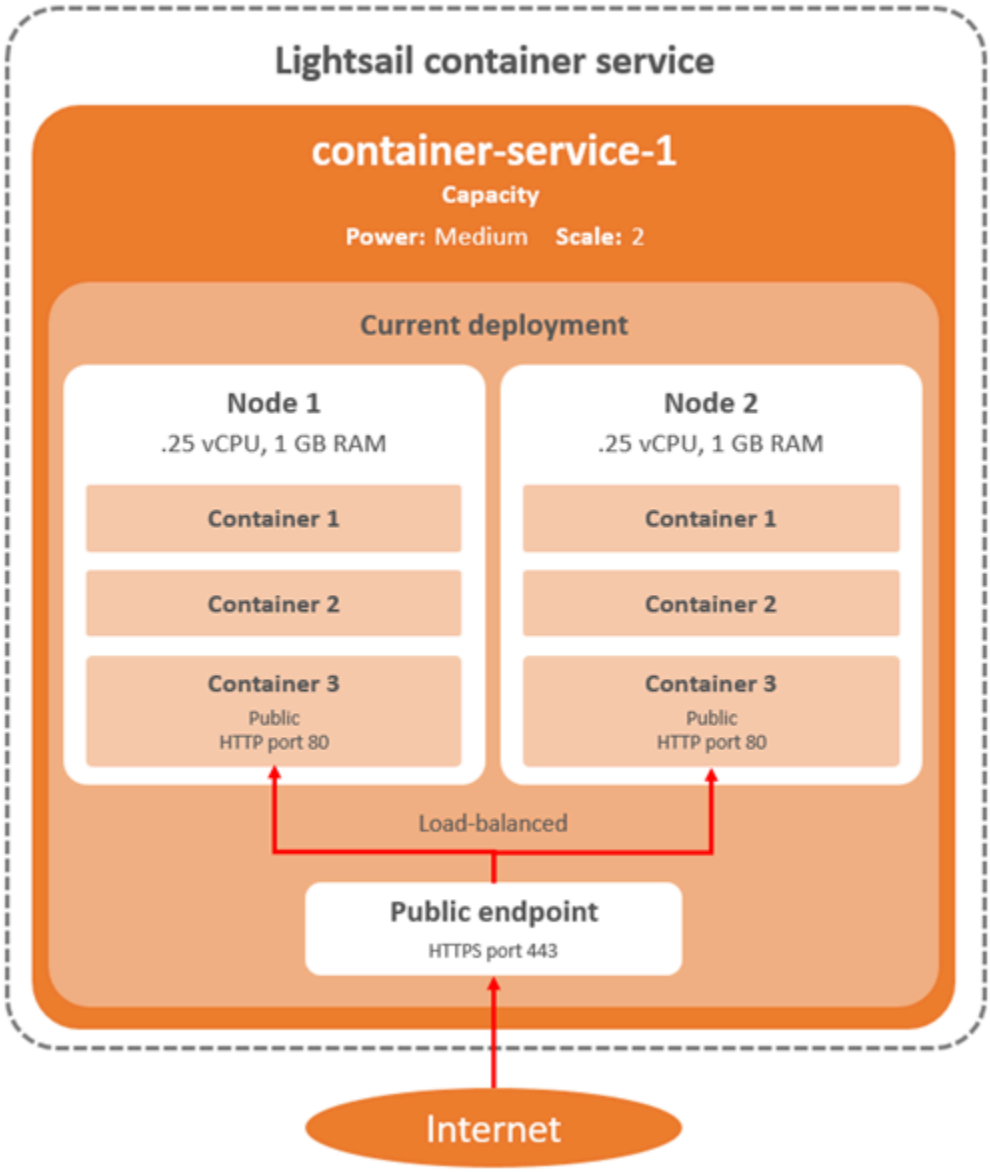
컨테이너 및 컨테이너 이미지에 대한 자세한 내용은 Docker 문서의 컨테이너란?을 참조하세요.

Lightsail 컨테이너 서비스 요소

Lightsail 컨테이너 서비스를 시작하려면 먼저 다음과 같은 핵심 요소를 이해해야 합니다.

Lightsail 컨테이너 서비스

컨테이너 서비스는 Lightsail이 지원되는 모든 AWS 리전 에서 생성할 수 있는 Lightsail 컴퓨팅 리소스입니다. 언제든지 컨테이너 서비스를 만들고 삭제할 수 있습니다. 자세한 내용은 Lightsail 컨테이너 서비스 생성 및 Lightsail 컨테이너 서비스 삭제를 참조하세요.



컨테이너 서비스 용량(규모 및 성능)

컨테이너 서비스를 처음 생성할 때는 다음과 같은 용량 파라미터를 선택해야 합니다.

- 규모(Scale) - 컨테이너 워크로드를 실행할 컴퓨팅 노드의 수입니다. 컨테이너 워크로드는 서비스의 컴퓨팅 노드 전체에 복사됩니다. 컨테이너 서비스에 대해 최대 20개의 컴퓨팅 노드를 지정할 수 있습니다. 가용성을 개선하고 용량을 늘리려면 서비스 성능을 좌우하는 노드 수에 따라 규모를 선택하면 됩니다. 컨테이너에 대한 트래픽은 모든 노드에서 로드 밸런싱됩니다.
- 성능(Power) - 컨테이너 서비스에 있는 각 노드의 메모리 및 vCPU입니다. 선택할 수 있는 성능은 나노(Na), 마이크로(Mi), 스몰(Sm), 미디엄(Md), 라지(Lg), 엑스라지(Xl)이며, 뒤로 갈수록 메모리와 vCPU 용량이 점차 더 커집니다.

컨테이너 서비스 규모를 1보다 크게 지정하면 컨테이너 워크로드가 서비스의 여러 컴퓨팅 노드에 걸쳐 복사됩니다. 예를 들어, 서비스 규모가 3이고 성능이 나노인 경우 각각 512MB의 RAM과 0.25개의 vCPU를 갖춘 3개의 컴퓨팅 리소스에서 컨테이너 워크로드 사본 3개가 실행됩니다. 수신 트래픽은 3개의 리소스 간에 로드 밸런싱됩니다. 컨테이너 서비스에 더 큰 용량을 지정할수록 처리할 수 있는 트래픽의 수가 늘어납니다.

프로비저닝이 부족해지면 언제든지 다운타임 없이 컨테이너 서비스의 성능 및 규모를 동적으로 늘리고, 과다 프로비저닝될 경우 축소할 수 있습니다. Lightsail에서는 현재 배포와 함께 용량 변경을 자동으로 관리합니다. 자세한 내용은 컨테이너 서비스의 용량 변경을 참조하세요.

가격 책정

컨테이너 서비스의 월별 요금은 성능 요금과 컴퓨팅 노드 수(서비스 규모)를 곱하여 계산됩니다. 예를 들어, 가격이 40 USD이고 컴퓨팅 노드가 3개 규모인 미디엄 성능 서비스를 사용하려면 월 120 USD를 부담해야 합니다. 컨테이너 서비스 요금은 컨테이너 서비스의 사용 및 배포 여부에 관계없이 부과됩니다. 요금이 더 이상 청구되지 않도록 하려면 컨테이너 서비스를 삭제해야 합니다.

각 컨테이너 서비스에는 구성된 용량에 관계없이 매월 500GB의 데이터 전송 할당량이 포함됩니다. 데이터 전송 할당량은 서비스에 대해 어떤 성능 및 규모를 선택하더라도 변함이 없습니다. 할당량을 초과하여 인터넷으로 데이터를 전송하면 GB당 0.09 USD부터 AWS 리전 시작하는에 따라 초과 요금이 발생합니다. 할당량을 초과하여 인터넷에서 전송된 데이터에는 초과 요금이 부과되지 않습니다. 자세한 내용은 Lightsail 요금 페이지를 참조하세요.

배포

Lightsail 컨테이너 서비스에서 배포를 생성할 수 있습니다. 배포는 서비스에서 시작하려는 컨테이너 워크로드의 사양 집합입니다.

배포의 각 컨테이너 항목에는 다음과 같은 파라미터를 지정할 수 있습니다.

- 시작할 컨테이너의 이름
- 컨테이너에 사용할 소스 컨테이너 이미지
- 컨테이너를 시작할 때 실행할 명령
- 컨테이너에 적용할 환경 변수
- 컨테이너에서 열 네트워크 포트
- 컨테이너 서비스의 기본 도메인을 통해 공개적으로 액세스할 수 있도록 지원하는 배포의 컨테이너

> **Note**
>
> 배포에 있는 컨테이너 하나만 각 컨테이너 서비스에 공개적으로 액세스할 수 있습니다.

다음 상태 확인 파라미터는 배포가 시작된 후 배포의 퍼블릭 엔드포인트에 적용됩니다.

- 상태 확인을 수행할 디렉터리 경로입니다.
- 간격 초, 시간 초과 초, 성공 코드, 정상 임계값 및 비정상 임계값과 같은 고급 상태 확인 설정입니다.

컨테이너 서비스에는 활성 배포가 동시에 하나만 존재할 수 있으며, 배포에는 최대 10개의 컨테이너 항목이 포함될 수 있습니다. 컨테이너 서비스를 생성하면서 배포를 생성하거나 서비스를 설정하여 실행하고 난 후에 생성할 수 있습니다. 자세한 내용은 컨테이너 서비스 배포 생성 및 관리를 참조하세요.

배포 버전

컨테이너 서비스에서 생성하는 모든 배포는 배포 버전으로 저장됩니다. 기존 배포의 파라미터를 수정하면 컨테이너가 서비스에 다시 배포되고 수정된 배포로 인해 새 배포 버전이 만들어집니다. 각 컨테이너 서비스에 대한 최신 배포 버전 50개가 저장됩니다. 50개의 배포 버전 중 하나를 사용하여 동일한 컨테이너 서비스에서 새 배포를 생성할 수 있습니다. 자세한 내용은 컨테이너 서비스 배포 생성 및 관리를 참조하세요.

컨테이너 이미지 소스

배포를 생성할 때 배포의 각 컨테이너 항목에 소스 컨테이너 이미지를 지정해야 합니다. 배포를 생성한 직후 컨테이너 서비스는 사용자가 지정한 소스에서 이미지를 가져와 컨테이너를 생성하는 데 사용합니다.

이미지는

- Amazon ECR Public Gallery 또는 기타 퍼블릭 컨테이너 이미지 레지스트리와 같은 퍼블릭 레지스트리. Amazon ECR Public에 대한 자세한 내용은 Amazon ECR Public 사용 설명서의 Amazon Elastic Container Registry Public이란 무엇입니까?를 참조하세요.
- 로컬 시스템에서 컨테이너 서비스로 푸시된 이미지. 로컬 시스템에 컨테이너 이미지를 생성하는 경우 해당 이미지를 컨테이너 서비스에 푸시하여 배포를 생성할 때 사용할 수 있습니다. 자세한 내용은 컨테이너 서비스 이미지 생성 및 컨테이너 이미지 푸시 및 관리를 참조하세요.

Lightsail 컨테이너 서비스는 Linux 기반 컨테이너 이미지를 지원합니다. Windows 기반 컨테이너 이미지는 현재 지원되지 않지만 Windows에서 Docker, AWS Command Line Interface (AWS CLI) 및 Lightsail Control (lightsailctl) 플러그인을 실행하여 Linux 기반 이미지를 빌드하고 Lightsail 컨테이너 서비스에 푸시할 수 있습니다.

컨테이너 서비스 ARN

Amazon 리소스 이름(ARNs AWS 리소스를 고유하게 식별합니다. IAM 정책 및 API 호출 AWS과 같이 모든에서 리소스를 명확하게 지정해야 하는 경우 ARN이 필요합니다.

컨테이너 서비스의 ARN을 가져오려면 `GetContainerServices` Lightsail API 작업을 사용하고 `serviceName` 파라미터를 사용하여 컨테이너 서비스의 이름을 지정합니다. 다음 예와 같이 작업 결과에 컨테이너 서비스 ARN이 나열됩니다. 자세한 내용은 Amazon LightsailAPI 레퍼런스의 GetContainerServices를 참조하세요.

출력은 다음과 비슷합니다.

```
{
    "containerServices": [
        {
            "containerServiceName": "container-service-1",
            "arn": "arn:aws:lightsail: :111122223333:ContainerService/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
            "createdAt": "2024-01-01T00:00:00+00:00",
            "location": {
                "availabilityZone": "all",
                "regionName": "us-west-2"
        },
        .....
}
```

퍼블릭 엔드포인트 및 기본 도메인

배포를 생성할 때 배포에서 컨테이너 서비스의 퍼블릭 엔드포인트로 사용할 컨테이너 항목을 지정할 수 있습니다. 퍼블릭 엔드포인트 컨테이너의 애플리케이션은 임의로 생성된 컨테이너 서비스의 기본 도메인을 통해 인터넷에서 공개적으로 액세스할 수 있습니다. 기본 도메인의 형식은 입니다. `https://<ServiceName>.<RandomGUID>.<AWSRegion>.cs.amazonlightsail.com`여기서 *<ServiceName>*은 컨테이너 서비스의 이름이고, *<RandomGUID>*는 Lightsail 계정의 AWS 리전 에서 컨테이너 서비스의 전역 고유 식별자이며, *<AWSRegion>*은 컨테이너 서비스가 생성된 AWS 리전 입니다. Lightsail 컨테이너 서비스의 퍼블릭 엔드포인트는 HTTPS만 지원하며, TCP 또는 UDP 트래픽은

지원하지 않습니다. 하나의 컨테이너만 서비스의 퍼블릭 엔드포인트가 될 수 있습니다. 따라서 나머지 컨테이너가 내부적으로 액세스할 수 있는 동안 애플리케이션의 프런트 엔드를 호스팅하는 컨테이너를 퍼블릭 엔드포인트로 선택해야 합니다.

컨테이너 서비스의 기본 도메인을 사용하거나 자체 사용자 지정 도메인(등록된 도메인 이름)을 사용할 수 있습니다. 컨테이너 서비스와 함께 사용자 지정 도메인을 사용하는 방법에 대한 자세한 내용은 컨테이너 서비스용 사용자 지정 도메인 사용 설정 및 관리를 참조하세요.

프라이빗 도메인

또한, 모든 컨테이너 서비스에는 *<ServiceName>*.service.local로 형식이 지정된 프라이빗 도메인이 있으며, 여기서 *<ServiceName>*은 컨테이너 서비스의 이름입니다. 프라이빗 도메인을 사용하여 서비스와 동일한 AWS 리전에 있는 다른 Lightsail 리소스에서 컨테이너 서비스에 액세스할 수 있습니다. 프라이빗 도메인은 서비스 배포에서 퍼블릭 엔드포인트를 지정하지 않은 경우 컨테이너 서비스에 액세스할 수 있는 유일한 방법입니다. 퍼블릭 엔드포인트를 지정하지 않더라도 컨테이너 서비스에 대해 기본 도메인이 생성되지만, 해당 도메인을 찾아보려고 하면 `404 No Such Service` 오류 메시지가 표시됩니다.

컨테이너 서비스의 프라이빗 도메인을 사용하여 특정 컨테이너에 액세스하려면 연결 요청을 수락할 컨테이너의 개방 포트를 지정해야 합니다. 이렇게 하려면 요청의 도메인 형식을 *<ServiceName>*.service.local:*<PortNumber>*로 지정하면 됩니다. 여기서 *<ServiceName>*은 컨테이너 서비스의 이름이고 *<PortNumber>*는 연결하려는 컨테이너의 개방 포트입니다. 예를 들어, `container-service-1`이라는 컨테이너 서비스에 배포를 생성하고 포트 6379가 열려 있는 Redis 컨테이너를 지정하는 경우 요청 도메인 형식을 *container-service-1*.service.local:*6379*로 지정해야 합니다.

사용자 지정 도메인 및 SSL/TLS 인증서

기본 도메인을 사용하는 대신 컨테이너 서비스와 함께 최대 4개의 사용자 지정 도메인을 사용할 수 있습니다. 예를 들어, 사용자 지정 도메인에 대한 트래픽(예: `example.com`)을 퍼블릭 엔드포인트로 레이블이 지정된 배포의 컨테이너로 연결할 수 있습니다.

서비스와 함께 사용자 지정 도메인을 사용하려면 먼저 사용할 도메인에 대한 SSL/TLS 인증서를 요청해야 합니다. 그런 다음 도메인의 DNS에 CNAME 레코드 집합을 추가하여 SSL/TLS 인증서를 검증해야 합니다. SSL/TLS 인증서가 검증되면 유효한 SSL/TLS 인증서를 서비스에 연결하여 컨테이너 서비스에서 사용자 지정 도메인을 사용하도록 설정합니다. 자세한 내용은 Lightsail 컨테이너 서비스용 SSL/TLS 인증서 생성, Lightsail 컨테이너 서비스용 SSL/TLS 인증서 검증 및 Lightsail 컨테이너 서비스용 사용자 지정 도메인 사용 설정 및 관리를 참조하세요.

컨테이너 로그

컨테이너 서비스의 모든 컨테이너는 액세스할 수 있는 로그를 생성하여 컨테이너의 운영을 진단합니다. 이 로그가 컨테이너 내부에서 실행되는 프로세스의 stdout 및 stderr 스트림을 제공합니다. 자세한 내용은 컨테이너 서비스 지표 확인을 참조하세요.

Metrics

컨테이너 서비스의 지표를 모니터링하여 과도한 활용으로 인해 발생할 수 있는 문제를 진단할 수 있습니다. 또한, 지표를 모니터링하여 서비스가 과소 또는 과다 프로비저닝되었는지 확인할 수도 있습니다. 자세한 내용은 컨테이너 서비스 지표 확인을 참조하세요.

Lightsail 컨테이너 서비스 사용

다음은 Lightsail 컨테이너 서비스를 관리하고, 로컬 시스템에서 서비스로 이미지를 푸시하거나 퍼블릭 레지스트리의 컨테이너 이미지를 사용하기 위한 일반 단계입니다.

배포에서 Lightsail 컨테이너 서비스를 관리하고 컨테이너 이미지를 사용하려면

1. Lightsail 계정에 컨테이너 서비스를 생성합니다. 자세한 내용은 Lightsail 컨테이너 서비스 생성을 참조하세요.
2. Lightsail 컨테이너 서비스와 함께 컨테이너 이미지를 사용하려면 다음 옵션 중 하나를 사용합니다.
   - 로컬 시스템에서 컨테이너 이미지 사용 - 로컬 시스템에 소프트웨어를 설치하여 자체 컨테이너 이미지를 생성한 다음 이를 Lightsail 컨테이너 서비스로 푸시합니다. 자세한 내용은 다음 안내서를 참조하세요.
     - Lightsail 컨테이너 서비스용 컨테이너 이미지를 관리하기 위한 소프트웨어 설치
     - Lightsail 컨테이너 서비스용 컨테이너 이미지 생성
     - Lightsail 컨테이너 서비스에서 컨테이너 이미지 푸시 및 관리
   - 퍼블릭 레지스트리의 컨테이너 이미지 사용 - Amazon ECR 퍼블릭 갤러리와 같은 퍼블릭 레지스트리에서 Lightsail 컨테이너 서비스에 대한 컨테이너 이미지를 찾아 사용할 수 있습니다. Amazon ECR 퍼블릭 갤러리에 대한 자세한 내용은 Amazon ECR Public 사용 설명서의 Amazon Elastic Container Registry Public이란 무엇입니까?를 참조하세요.
3. Lightsail 컨테이너 서비스용 컨테이너 이미지를 관리하기 위한 소프트웨어 설치.
4. Lightsail 컨테이너 서비스용 컨테이너 이미지 생성.
5. Lightsail 컨테이너 서비스에서 컨테이너 이미지 푸시 및 관리.

6. 컨테이너를 구성하고 시작하는 컨테이너 서비스에서 배포를 생성합니다. 자세한 내용은 Lightsail 컨테이너 서비스용 배포 생성 및 관리를 참조하세요.
7. 컨테이너 서비스의 이전 배포를 확인합니다. 이전 배포 버전을 사용하여 새 배포를 생성할 수 있습니다. 자세한 내용은 Lightsail 컨테이너 서비스의 배포 버전 확인 및 관리를 참조하세요.
8. 컨테이너 서비스의 컨테이너 로그를 확인합니다. 자세한 내용은 Lightsail 컨테이너 서비스의 컨테이너 로그 확인을 참조하세요.
9. 컨테이너와 함께 사용하려는 도메인에 대한 SSL/TLS 인증서를 생성합니다. 자세한 내용은 Lightsail 컨테이너 서비스용 SSL/TLS 인증서 생성을 참조하세요.
10. 도메인의 DNS에 레코드를 추가하여 SSL/TLS 인증서를 검증합니다. 자세한 내용은 Lightsail 컨테이너 서비스용 SSL/TLS 인증서 검증을 참조하세요.
11. 컨테이너 서비스에 유효한 SSL/TLS 인증서를 연결하여 사용자 지정 도메인을 사용하도록 설정합니다. 자세한 내용은 Lightsail 컨테이너 서비스용 사용자 지정 도메인 사용 설정 및 관리를 참조하세요.
12. 컨테이너 서비스의 사용률 지표를 모니터링합니다. 자세한 내용은 컨테이너 서비스 지표 확인을 참조하세요.
13. (선택 사항) 성능 사양을 높여 컨테이너 서비스의 용량을 수직적으로, 규모 사양을 늘려 용량을 수평적으로 확장할 수 있습니다. 자세한 내용은 Lightsail 컨테이너 서비스의 용량 변경을 참조하세요.
14. 컨테이너 서비스를 사용하지 않는 경우 월별 요금이 발생하지 않도록 이를 삭제합니다. 자세한 내용은 Lightsail 컨테이너 서비스 삭제를 참조하세요.

Lightsail을 사용하여 고가용성 컨테이너 서비스 생성

이 가이드에서는 Lightsail 콘솔을 사용하여 Amazon Lightsail 컨테이너 서비스를 생성하는 방법을 안내하고 구성할 수 있는 컨테이너 서비스 설정에 대해 설명합니다.

시작하기 전에 Lightsail 컨테이너 서비스를 구성하는 요소를 숙지하는 것이 좋습니다. 자세한 내용은 컨테이너 서비스를 참조하세요.

컨테이너 서비스 용량(규모 및 성능)

컨테이너 서비스를 처음 생성할 때는 컨테이너 서비스 용량을 선택해야 합니다. 용량은 다음 파라미터의 조합으로 구성됩니다.

- 규모(Scale) - 컨테이너 워크로드를 실행할 컴퓨팅 노드의 수입니다. 컨테이너 워크로드는 서비스의 컴퓨팅 노드 전체에 복사됩니다. 컨테이너 서비스에 대해 최대 20개의 컴퓨팅 노드를 지정할 수 있습

니다. 가용성을 개선하고 용량을 늘리려면 서비스 성능을 좌우하는 노드 수에 따라 규모를 선택하면 됩니다. 컨테이너에 대한 트래픽은 모든 노드에서 로드 밸런싱됩니다.

- 성능(Power) - 컨테이너 서비스에 있는 각 노드의 메모리 및 vCPU입니다. 선택할 수 있는 성능은 나노(Na), 마이크로(Mi), 스몰(Sm), 미디엄(Md), 라지(Lg), 엑스라지(Xl)이며, 뒤로 갈수록 메모리와 vCPU 용량이 점차 더 커집니다.

수신 트래픽은 컨테이너 서비스의 규모(컴퓨팅 노드 수)에 걸쳐 로드 밸런싱됩니다. 예를 들어, 규모가 3인 나노 성능 서비스를 사용하면 컨테이너 워크로드 사본을 3개 실행할 수 있습니다. 각 노드에는 512MB의 RAM과 0.25개의 vCPU가 있습니다. 수신 트래픽은 3개 노드에 걸쳐 로드 밸런싱됩니다. 컨테이너 서비스에 더 큰 용량을 선택할수록 처리할 수 있는 트래픽의 수가 늘어납니다.

프로비저닝이 부족해지면 언제든지 다운타임 없이 컨테이너 서비스의 성능 및 규모를 동적으로 늘리고, 과다 프로비저닝될 경우 축소할 수 있습니다. Lightsail에서는 현재 배포와 함께 용량 변경을 자동으로 관리합니다. 자세한 내용은 Lightsail 컨테이너 서비스의 용량 변경을 참조하세요.

가격 책정

컨테이너 서비스의 월별 요금은 기본 성능 요금과 규모(컴퓨팅 노드 수)를 곱하여 계산됩니다. 예를 들어, 규모가 3인 40 USD의 미디엄 성능 서비스를 사용하면 월 120 USD를 부담해야 합니다.

각 컨테이너 서비스에는 구성된 용량에 관계없이 매월 500GB의 데이터 전송 할당량이 포함됩니다. 데이터 전송 할당량은 서비스에 대해 어떤 성능 및 규모를 선택하더라도 변함이 없습니다. 할당량을 초과하여 인터넷으로 데이터를 내보낼 경우 초과 요금은 AWS 리전에 따라 다르며, GB당 0.09 USD부터 시작합니다. 할당량을 초과하여 인터넷에서 전송된 데이터에는 초과 요금이 부과되지 않습니다. 자세한 내용은 Lightsail 요금 페이지를 참조하세요.

컨테이너 서비스 요금은 컨테이너 서비스의 사용 및 배포 여부에 관계없이 부과됩니다. 요금이 더 이상 청구되지 않도록 하려면 컨테이너 서비스를 삭제해야 합니다. 자세한 내용은 Lightsail 컨테이너 서비스 삭제를 참조하세요.

컨테이너 서비스 상태

컨테이너 서비스는 다음 상태 중 하나일 수 있습니다.

- 보류 중(Pending) - 컨테이너 서비스가 생성 중입니다.
- 준비(Ready) - 컨테이너 서비스가 실행 중이지만, 활성 컨테이너 배포가 없습니다.
- 배포 중(Deploying) - 컨테이너 서비스에 배포가 시작됩니다.
- 실행 중(Running) - 컨테이너 서비스가 실행 중이며 활성 컨테이너 배포가 있습니다.

- 업데이트 중(Updating) - 컨테이너 서비스 용량 또는 사용자 지정 도메인이 업데이트되고 있습니다.
- 삭제 중(Deleting) - 컨테이너 서비스가 삭제 중입니다. 컨테이너 서비스를 삭제하기로 선택하면 컨테이너 서비스가 한동안 이 상태로 유지됩니다.
- 사용 중지(Disabled) - 컨테이너 서비스 사용이 중지되고, 활성 배포와 컨테이너가 있는 경우 종료됩니다.

컨테이너 서비스 하위 상태

컨테이너 서비스가 배포 중 또는 업데이트 중 상태이면 다음과 같은 추가 하위 상태 중 하나가 컨테이너 서비스 상태 아래에 표시됩니다.

- 시스템 리소스 생성 중(Creating system resources) - 컨테이너 서비스의 시스템 리소스를 생성하는 중입니다.
- 네트워크 인프라 생성 중(Creating network infrastructure) - 컨테이너 서비스의 네트워크 인프라를 생성하는 중입니다.
- 인증서 프로비저닝(Provisioning certificate) - 컨테이너 서비스용 SSL/TLS 인증서를 생성하는 중입니다.
- 서비스 프로비저닝(Provisioning service) - 컨테이너 서비스를 프로비저닝하는 중입니다.
- 배포 생성 중(Creating deployment) - 컨테이너 서비스에서 배포를 생성하는 중입니다.
- 상태 확인 평가 중(Evaluating health check) - 배포 상태를 평가하는 중입니다.
- 배포 활성화 중(Activating deployment) - 배포를 활성화하는 중입니다.

컨테이너 서비스가 보류 중(Pending) 상태이면 다음과 같은 추가 하위 상태 중 하나가 컨테이너 서비스 상태 아래에 표시됩니다.

- 인증서 제한 초과(Certificate limit exceeded) - 컨테이너 서비스에 필요한 SSL/TLS 인증서가 계정에 허용된 최대 인증서 수를 초과합니다.
- 알 수 없는 오류(Unknown error) - 컨테이너 서비스를 생성하는 중에 오류가 발생했습니다.

컨테이너 서비스 생성

Lightsail 컨테이너 서비스를 생성하려면 다음 절차를 완료하세요.

1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창의 이미지에서 컨테이너를 선택합니다.

3. 컨테이너 서비스 생성(Create container service)을 선택합니다.
4. 컨테이너 서비스 생성 페이지에서 변경을 AWS 리전선택한 다음 컨테이너 서비스에 AWS 리전 대한를 선택합니다.
5. 컨테이너 서비스의 용량을 선택합니다. 자세한 내용은 가이드의 컨테이너 서비스 용량(규모 및 성능) 섹션을 참조하세요.
6. 컨테이너 서비스를 생성하면서 동시에 시작할 배포를 생성하려면 다음 단계를 완료하세요. 그렇지 않으면 7단계로 건너뛰고 배포 없이 컨테이너 서비스를 생성합니다.

   퍼블릭 레지스트리에서 컨테이너 이미지를 사용하려는 경우 배포와 함께 컨테이너 서비스를 생성합니다. 아니면 로컬 시스템에 있는 컨테이너 이미지를 사용하려는 경우 배포 없이 서비스를 생성합니다. 서비스를 설정하고 실행한 후 컨테이너 이미지를 로컬 시스템에서 컨테이너 서비스로 푸시할 수 있습니다. 그런 다음 컨테이너 서비스에 등록되어 있는 푸시된 컨테이너 이미지를 사용하여 배포를 생성할 수 있습니다.

   a. 배포 생성(Create a deployment)을 선택합니다.
   b. 다음 옵션 중 하나를 선택하세요.

      - 예 배포 선택(Choose an example deployment) – Lightsail 팀이 미리 구성된 배포 파라미터로 큐레이팅한 컨테이너 이미지를 사용하여 배포를 생성하려면 이 옵션을 선택합니다. 이 옵션을 이용하면 널리 사용되는 컨테이너를 컨테이너 서비스에서 가장 빠르고 쉽게 설치하고 실행할 수 있습니다.
      - 사용자 지정 배포 지정(Specify a custom deployment) - 선택한 컨테이너를 지정하여 배포를 생성하려면 이 옵션을 선택합니다.

      새 배포 파라미터를 입력할 수 있는 배포 양식 보기가 열립니다.

   c. 배포의 파라미터를 입력합니다. 지정할 수 있는 배포 파라미터에 대한 자세한 내용은 Lightsail 컨테이너 서비스용 배포 생성 및 관리 가이드의 배포 파라미터 섹션을 참조하세요.
   d. 컨테이너 항목 추가(Add container entry)를 선택하여 둘 이상의 컨테이너 항목을 배포에 추가합니다. 배포에 컨테이너 항목을 10개까지 포함할 수 있습니다.
   e. 배포 파라미터를 입력했으면 저장 및 배포(Save and deploy)를 선택하여 컨테이너 서비스에 배포를 생성합니다.
7. 컨테이너 서비스 이름을 입력합니다.

   컨테이너 서비스 이름은 다음과 같아야 합니다.

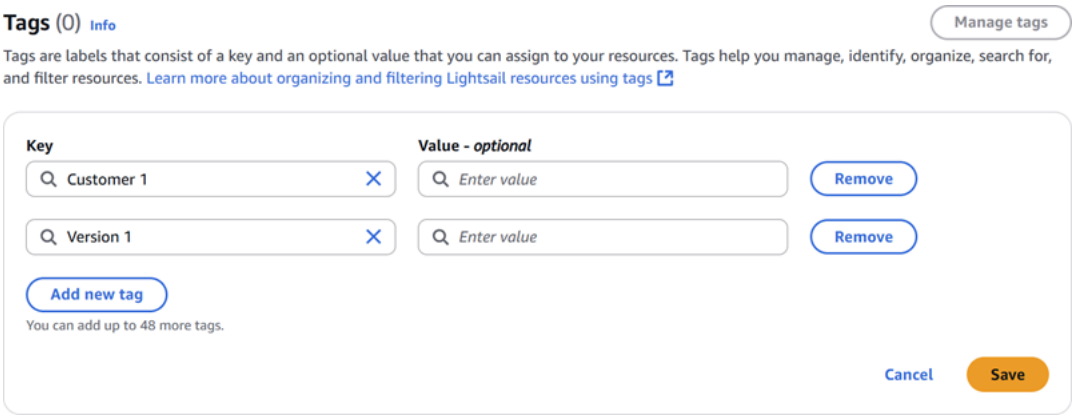
   - Lightsail 계정 AWS 리전 의 각 내에서 고유해야 합니다.

- 2~63자의 문자로 구성되어야 합니다.
- 영숫자 및 하이픈으로만 구성되어야 합니다.
- 하이픈(-)으로 단어를 구분할 수 있지만, 이름의 시작이나 끝에 사용할 수는 없습니다.

> **Note**
>
> 지정한 이름은 컨테이너 서비스의 기본 도메인 이름 일부가 되며 전체 공개됩니다.

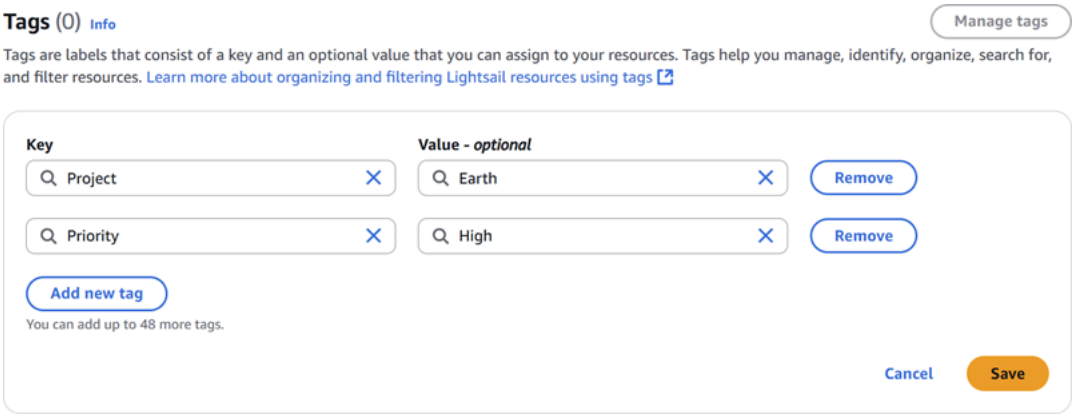
8. 다음 옵션 중 하나를 선택하여 컨테이너 서비스에 태그를 추가합니다.

- 키 전용 태그를 추가하거나 태그를 관리(이미 태그를 추가한 경우)합니다. 새 태그를 태그 키 텍스트 상자에 입력하고 Enter를 누릅니다. 태그를 추가하려면 태그 입력이 완료될 때 저장을 선택하고, 태그를 추가하지 않으려면 취소를 선택합니다.



- 키-값 태그를 생성한 다음 키를 키 텍스트 상자에 입력하고, 값을 값 텍스트 상자에 입력합니다. 태그 입력이 완료되면 저장을 선택하고, 태그를 추가하지 않으려면 취소를 선택합니다.

  키-값 태그는 저장 전에 한 번에 하나씩만 추가할 수 있습니다. 둘 이상의 키-값 태그를 추가하려면 이전 단계를 반복하세요.



> **Note**
>
> 키 전용 태그 및 키-값 태그에 대한 자세한 내용은 태그를 참조하세요.

9. 컨테이너 서비스 생성(Create container service)을 선택합니다.

   새 컨테이너 서비스의 관리 페이지로 리디렉션됩니다. 새 컨테이너 서비스를 생성하는 동안 상태는 보류 중(Pending)입니다. 잠시 후 현재 배포가 없으면 서비스 상태가 준비(Ready)로 변경되거나 배포를 생성했으면 실행 중(Running)으로 변경됩니다.

Lightsail 컨테이너 서비스를 위한 Docker 이미지 빌드 및 테스트

Docker를 사용하면 컨테이너를 기반으로 하는 분산 애플리케이션을 빌드, 실행, 테스트 및 배포할 수 있습니다. Amazon Lightsail 컨테이너 서비스는 배포에서 Docker 컨테이너 이미지를 사용하여 컨테이너를 시작합니다.

이 가이드에서는 Dockerfile을 사용하여 로컬 시스템에 컨테이너 이미지를 생성하는 방법을 안내합니다. 이미지가 생성되면 이미지를 Lightsail 컨테이너 서비스에 푸시하여 배포할 수 있습니다.

이 가이드의 절차를 완료하려면 기본적으로 Docker가 무엇이고 어떻게 작동하는지를 이해해야 합니다. Docker에 대한 자세한 내용은 Docker란 무엇인가? 및 Docker 개요를 참조하세요.

목차

- 1단계: 필수 구성 요소 완성
- 2단계: Dockerfile 생성 및 컨테이너 이미지 구축
- 3단계: 새 컨테이너 이미지 실행
- (선택 사항) 4단계: 로컬 시스템에서 실행 중인 컨테이너 정리
- 컨테이너 이미지를 생성한 후의 다음 단계

1단계: 필수 구성 요소 완성

시작하기 전에 컨테이너를 생성하는 데 필요한 소프트웨어를 설치한 다음 Lightsail 컨테이너 서비스에 푸시해야 합니다. 예를 들어, Lightsail 컨테이너 서비스와 함께 사용할 수 있는 컨테이너 이미지를 생성하고 구축하려면 Docker를 설치하고 사용해야 합니다. 자세한 내용은 Amazon Lightsail 컨테이너 서비스용 컨테이너 이미지를 관리하기 위한 소프트웨어 설치를 참조하세요.

2단계: Dockerfile 생성 및 컨테이너 이미지 구축

Dockerfile을 생성하고 Dockerfile에서 `mystaticwebsite` Docker 컨테이너 이미지를 구축하려면 다음 절차를 완료하세요. 컨테이너 이미지는 Ubuntu의 Apache 웹 서버에서 호스팅되는 간단한 정적 웹 사이트에 사용됩니다.

1. 로컬 시스템에 Dockerfile을 저장할 `mystaticwebsite` 폴더를 생성합니다.
2. 방금 만든 폴더에 Dockerfile을 생성합니다.

   Dockerfile은 `.TXT`와 같은 파일 확장자를 사용하지 않습니다. 전체 파일 이름은 `Dockerfile`입니다.
3. 컨테이너 이미지를 구성하는 방법에 따라 다음 코드 블록 중 하나를 복사하여 Dockerfile에 붙여넣습니다.

   - 간단한 정적 웹 사이트 컨테이너 이미지를 Hello World 메시지와 함께 생성하려면 다음 코드 블록을 복사하여 Dockerfile에 붙여넣습니다. 이 코드 샘플은 Ubuntu 18.04 이미지를 사용합니다. `RUN` 지침은 패키지 캐시를 업데이트하고 Apache를 설치 및 구성한 후 Hello World 메시지를 웹 서버의 문서 루트로 인쇄합니다. `EXPOSE` 지침은 컨테이너에 포트 80을 노출하고 `CMD` 지침은 웹 서버를 시작합니다.

     ```
     FROM ubuntu:18.04

     # Install dependencies
     RUN apt-get update && \
      apt-get -y install apache2

     # Write hello world message
     RUN echo 'Hello World!' > /var/www/html/index.html

     # Open port 80
     EXPOSE 80

     # Start Apache service
     CMD ["/usr/sbin/apache2ctl", "-D", "FOREGROUND"]
     ```

   - 정적 웹 사이트 컨테이너 이미지에 사용자 고유의 HTML 파일 집합을 사용하려면 Dockerfile을 저장하는 폴더와 동일한 폴더에 `html` 폴더를 생성합니다. 그런 다음 HTML 파일을 해당 폴더에 넣습니다.

HTML 파일이 html 폴더에 있으면 다음 코드 블록을 복사하여 Dockerfile에 붙여넣습니다. 이 코드 샘플은 Ubuntu 18.04 이미지를 사용합니다. RUN 지침은 패키지 캐시를 업데이트하고 Apache를 설치 및 구성합니다. COPY 지침은 html 폴더의 내용을 웹 서버의 문서 루트에 복사합니다. EXPOSE 지침은 컨테이너에 포트 80을 노출하고 CMD 지침은 웹 서버를 시작합니다.

```
FROM ubuntu:18.04

# Install dependencies
RUN apt-get update && \
 apt-get -y install apache2

# Copy html directory files
COPY html /var/www/html/

# Open port 80
EXPOSE 80

CMD ["/usr/sbin/apache2ctl", "-D", "FOREGROUND"]
```

4. 명령 프롬프트 또는 터미널 창을 열고 Dockerfile을 저장할 폴더로 디렉터리를 변경합니다.
5. 다음 명령을 입력하여 폴더의 Dockerfile을 사용하여 컨테이너 이미지를 구축합니다. 이 명령은 이름이 mystaticwebsite인 새 Docker 컨테이너 이미지를 구축합니다.

```
docker build -t mystaticwebsite .
```

이미지가 성공적으로 구축되었음을 확인하는 메시지가 표시됩니다.

6. 다음 명령을 입력하여 로컬 시스템의 컨테이너 이미지를 확인합니다.

```
docker images --filter reference=mystaticwebsite
```

다음 예와 유사한 결과가 나타나 생성된 새 컨테이너 이미지를 보여줍니다.

```
C:\Users\[redacted]\Documents\Docker\Dockerfiles\mystaticwebsite>docker images --filter reference=mystaticwebsite
REPOSITORY          TAG                 IMAGE ID            CREATED             SIZE
mystaticwebsite     latest              8f7ffd1013e0        8 minutes ago       199MB
```

새로 구축한 컨테이너 이미지를 사용하여 테스트하고 로컬 시스템에서 새 컨테이너를 실행할 준비를 마쳤습니다. 이 가이드의 다음 3단계: 새 컨테이너 이미지 실행 섹션을 이어서 진행하세요.

3단계: 새 컨테이너 이미지 실행

생성한 새 컨테이너 이미지를 실행하려면 다음 단계를 완료하세요.

1. 명령 프롬프트 또는 터미널 창에서 다음 명령을 입력하여 가이드의 이전 2단계: Dockerfile 생성 및 컨테이너 이미지 구축 섹션에서 구축한 컨테이너 이미지를 실행합니다. `-p 8080:80` 옵션은 컨테이너의 노출된 포트 80을 로컬 시스템의 포트 8080에 매핑합니다. `-d` 옵션은 컨테이너가 분리 모드에서 실행되도록 지정합니다.

   ```
   docker container run -d -p 8080:80 --name mystaticwebsite mystaticwebsite:latest
   ```

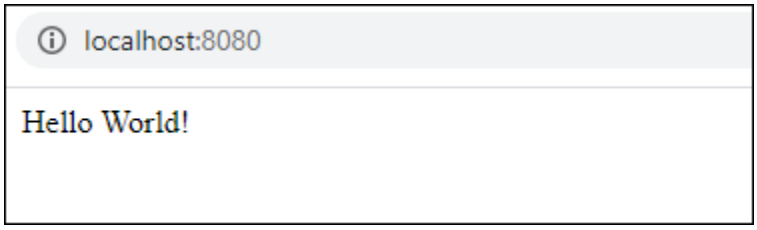
2. 다음 명령을 입력하여 실행 중인 컨테이너를 확인합니다.

   ```
   docker container ls -a
   ```

   다음 예와 유사한 결과가 나타나 실행 중인 새 컨테이너를 보여줍니다.

   ```
   C:\Users\[redacted]\Documents\Docker\Dockerfiles\mystaticwebsite>docker container ls -a
   CONTAINER ID        IMAGE                   COMMAND                  CREATED             STATUS              PORTS                   NAMES
   62382081e06b        mystaticwebsite:latest   "/bin/sh -c /root/ru…"   6 minutes ago       Up 6 minutes        0.0.0.0:8080->80/tcp    mystaticwebsite
   ```

3. 컨테이너가 설정되고 실행 중인지 확인하려면 새 브라우저 창을 열고 `http://localhost:8080`을 찾습니다. 다음 예와 비슷한 메시지가 나타나는 것을 볼 수 있습니다. 이렇게 하면 컨테이너가 로컬 시스템에서 설정되어 실행 중인지 확인할 수 있습니다.

   새로 구축한 컨테이너 이미지를 Lightsail 컨테이너 서비스에 배포할 수 있도록 Lightsail 계정으로 푸시할 준비를 마쳤습니다. 자세한 내용은 Amazon Lightsail 컨테이너 서비스에서 컨테이너 이미지 푸시 및 관리를 참조하세요.

(선택 사항) 4단계: 로컬 시스템에서 실행 중인 컨테이너 정리

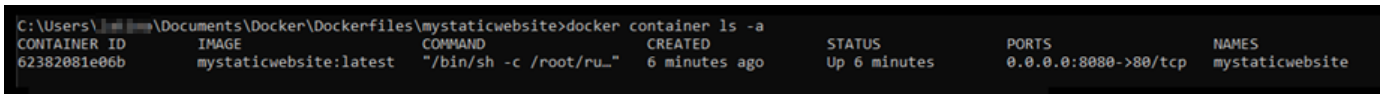
Lightsail 컨테이너 서비스에 푸시할 수 있는 컨테이너 이미지를 생성했으니, 가이드의 절차에 따른 결과로 로컬 시스템에서 실행 중인 컨테이너를 정리해야 합니다.

로컬 시스템에서 실행 중인 컨테이너를 정리하려면 다음 단계를 완료하세요.

1. 다음 명령을 실행하여 로컬 시스템에서 실행 중인 컨테이너를 확인합니다.

```
docker container ls -a
```

다음과 유사한 결과가 나타나 로컬 시스템에서 실행 중인 컨테이너의 이름을 나열합니다.



2. 다음 명령을 실행하여 가이드의 앞부분에서 생성하여 실행 중인 컨테이너를 제거합니다. 이렇게 하면 컨테이너가 강제로 중지되고 영구적으로 삭제됩니다.

```
docker container rm <ContainerName> --force
```

명령에서 <ContainerName>을 중지할 컨테이너 이름으로 바꾸고 삭제합니다.

예제:

```
docker container rm mystaticwebsite --force
```

이제 가이드를 따라 생성한 컨테이너가 삭제됩니다.

컨테이너 이미지를 생성한 후의 다음 단계

컨테이너 이미지를 생성한 후 배포할 준비가 되면 해당 이미지를 Lightsail 컨테이너 서비스에 푸시합니다. 자세한 내용은 Lightsail 컨테이너 서비스 이미지 관리를 참조하세요.

주제

- 컨테이너 서비스에 대한 Lightsail 컨테이너 이미지 푸시, 보기 및 삭제
- Docker AWS CLI및 컨테이너용 Lightsail 제어 플러그인 설치
- Amazon ECR 프라이빗 리포지토리에 대한 액세스 권한을 Lightsail 컨테이너 서비스에 부여

컨테이너 서비스에 대한 Lightsail 컨테이너 이미지 푸시, 보기 및 삭제

Amazon Lightsail 컨테이너 서비스에서 배포를 생성할 때는 각 컨테이너 항목에 소스 컨테이너 이미지를 지정해야 합니다. Amazon ECR Public Gallery와 같은 퍼블릭 레지스트리의 이미지를 사용하거나 로컬 시스템에서 생성한 이미지를 사용할 수 있습니다. 이 설명서에서는 로컬 시스템에서 Lightsail 컨

테이너 서비스로 컨테이너 이미지를 푸시하는 방법을 안내합니다. 컨테이너 이미지를 생성하는 방법에 대한 자세한 내용은 컨테이너 서비스 이미지 생성을 참조하세요.

목차

- 사전 조건
- 로컬 시스템에서 컨테이너 서비스로 컨테이너 이미지 푸시
- 컨테이너 서비스에 저장된 컨테이너 이미지 보기
- 컨테이너 서비스에 저장된 컨테이너 이미지 삭제

사전 조건

컨테이너 이미지를 컨테이너 서비스로 푸시하기 전에 다음 사전 조건을 완료하세요.

- Lightsail 계정에 컨테이너 서비스를 생성합니다. 자세한 내용은 Amazon Lightsail 컨테이너 서비스 생성을 참조하세요.
- 로컬 시스템에 자체 컨테이너 이미지를 생성하는 데 필요한 소프트웨어를 설치하고 이를 Lightsail 컨테이너 서비스로 푸시합니다. 자세한 내용은 Amazon Lightsail 컨테이너 서비스용 컨테이너 이미지를 관리하기 위한 소프트웨어 설치를 참조하세요.
- 로컬 시스템에서 Lightsail 컨테이너 서비스에 푸시할 컨테이너 이미지를 생성합니다. 자세한 내용은 Amazon Lightsail 컨테이너 서비스의 컨테이너 이미지 생성을 참조하세요.

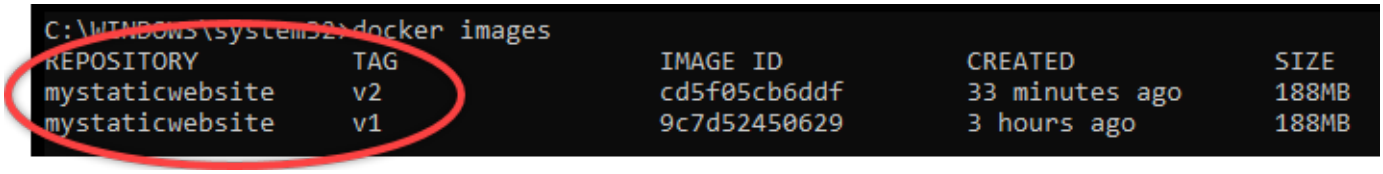
로컬 시스템에서 컨테이너 서비스로 컨테이너 이미지 푸시

컨테이너 이미지를 컨테이너 서비스로 푸시하려면 다음 절차를 완료하세요.

1. 명령 프롬프트 또는 터미널 창을 엽니다.
2. 명령 프롬프트 또는 터미널 창에 다음 명령을 입력하여 현재 로컬 시스템에 있는 Docker 이미지를 확인합니다.

   ```
   docker images
   ```

3. 결과에서 컨테이너 서비스에 푸시할 컨테이너 이미지의 이름(리포지토리 이름)과 태그를 찾습니다. 이후 단계에서 사용해야 하므로 이름과 태그를 기록해 둡니다.



4. 다음 명령을 입력하여 로컬 시스템의 컨테이너 이미지를 컨테이너 서비스로 푸시합니다.

```
aws lightsail push-container-image --region <Region> --service-
name <ContainerServiceName> --label <ContainerImageLabel> --
image <LocalContainerImageName>:<ImageTag>
```

명령에서 다음과 같이 바꿉니다.

- *<Region>*을 컨테이너 서비스가 생성된 AWS 리전으로 바꿉니다.
- *<ContainerServiceName>*을 컨테이너 서비스의 이름으로 바꿉니다.
- *<ContainerImageLabel>*을 컨테이너 서비스에 저장할 때 컨테이너 이미지에 지정하려는 레이블로 바꿉니다. 등록된 컨테이너 이미지의 여러 버전을 추적하는 데 사용할 수 있는 설명 레이블을 지정합니다.

  이 레이블이 컨테이너 서비스에서 생성한 컨테이너 이미지 이름의 일부가 됩니다. 예를 들어, 컨테이너 서비스 이름이 `container-service-1`이고, 컨테이너 이미지 레이블이 `mystaticsite`이며, 푸시하는 컨테이너 이미지의 첫 번째 버전인 경우 컨테이너 서비스에서 생성한 이미지 이름은 `:container-service-1.mystaticsite.1`입니다.
- *<LocalContainerImageName>*을 컨테이너 서비스로 푸시하려는 컨테이너 이미지의 이름으로 바꿉니다. 이 절차의 이전 단계에서 컨테이너 이미지 이름을 지정했습니다.
- *<ImageTag>*를 컨테이너 서비스로 푸시하려는 컨테이너 이미지의 태그로 바꿉니다. 이 절차의 이전 단계에서 컨테이너 이미지 태그를 지정했습니다.

예제:

```
aws lightsail push-container-image --region us-west-2 --service-name myservice --
label mystaticwebsite --image mystaticwebsite:v2
```

컨테이너 이미지가 컨테이너 서비스로 푸시되었음을 확인하는 다음 예와 유사한 결과가 나타납니다.

```
C:\WINDOWS\system32>aws lightsail push-container-image --service-name myservice --label mystaticwebsite
--image mystaticwebsite:v2

[1B5a355b95: Preparing
[1B0994b087: Preparing
[1B0c904ff3: Preparing
[1B370aa736: Preparing
[1Bf192bbc8: Preparing
[1Bbc0bd923: Preparing
[7BDigest: sha256:3a585ca39bba342e390b39f2fea00bbc20f492c0cda7b923dd766abe31918f3bB/1.96kB
Image "mystaticwebsite:v2" registered.
Refer to this image as ":myservice.mystaticwebsite.2" in deployments.
```

Lightsail 콘솔의 컨테이너 서비스에서 푸시된 컨테이너 이미지를 확인하려면 이 설명서의 컨테이너 서비스에 저장된 컨테이너 이미지 보기 섹션을 참조하세요.

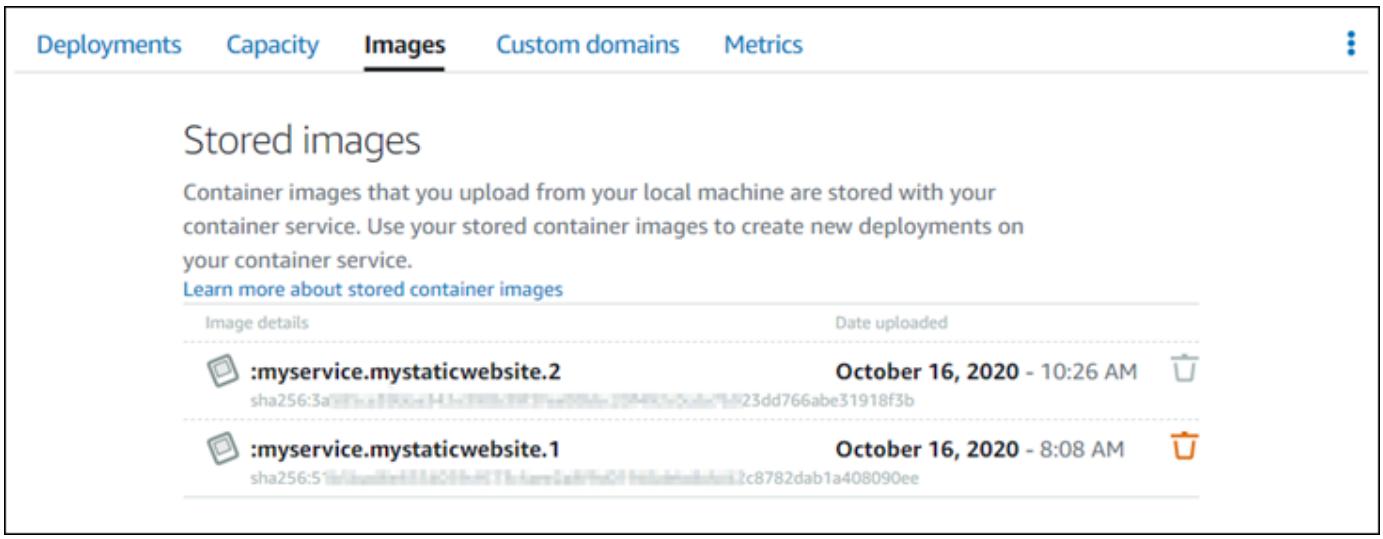
컨테이너 서비스에 저장된 컨테이너 이미지 보기

컨테이너 서비스에 푸시되어 저장된 컨테이너 이미지를 보려면 다음 절차를 완료하세요.

1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창의 이미지에서 컨테이너를 선택합니다.
3. 저장된 컨테이너 이미지를 보려는 컨테이너 서비스의 이름을 선택합니다.
4. 컨테이너 서비스 관리 페이지에서 이미지(Images) 탭을 선택합니다.

> **Note**
>
> 컨테이너 서비스에 이미지를 푸시하지 않은 경우 이미지(Images) 탭이 표시되지 않습니다. 컨테이너 서비스에 대한 이미지 탭을 표시하려면 먼저 컨테이너 이미지를 서비스로 푸시해야 합니다.

이미지(Images) 페이지에는 컨테이너 서비스로 푸시되어 현재 서비스에 저장된 컨테이너 이미지가 나열되어 있습니다. 현재 배포에서 사용 중인 컨테이너 이미지는 삭제할 수 없으며, 회색으로 표시된 삭제 아이콘과 함께 나열됩니다.



서비스에 저장된 컨테이너 이미지를 사용하여 배포를 생성할 수 있습니다. 자세한 내용은 Amazon Lightsail 컨테이너 서비스용 배포 생성 및 관리를 참조하세요.

컨테이너 서비스에 저장된 컨테이너 이미지 삭제

컨테이너 서비스에 푸시되어 저장된 컨테이너 이미지를 삭제하려면 다음 절차를 완료하세요.

1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창의 이미지에서 컨테이너를 선택합니다.
3. 현재 배포를 확인하려는 컨테이너 서비스의 이름을 선택합니다.
4. 컨테이너 서비스 관리 페이지에서 이미지(Images) 탭을 선택합니다.

> **Note**
>
> 컨테이너 서비스에 이미지를 푸시하지 않은 경우 이미지(Images) 탭이 표시되지 않습니다. 컨테이너 서비스에 대한 이미지 탭을 표시하려면 먼저 컨테이너 이미지를 서비스로 푸시해야 합니다.

5. 삭제할 컨테이너 이미지를 찾은 후 삭제(휴지통) 아이콘을 선택합니다.

> **Note**
>
> 현재 배포에서 사용 중인 컨테이너 이미지는 삭제할 수 없으며, 해당 삭제 아이콘은 회색으로 표시됩니다.

6. 확인 프롬프트가 표시되면 예, 삭제합니다(Yes, delete)를 선택하여 저장된 이미지를 영구적으로 삭제할지 확인합니다.

   저장된 컨테이너 이미지가 컨테이너 서비스에서 즉시 삭제됩니다.

Docker AWS CLI및 컨테이너용 Lightsail 제어 플러그인 설치

Amazon Lightsail 콘솔을 사용하여 Lightsail 컨테이너 서비스를 생성하고, Amazon ECR Public Gallery와 같은 온라인 퍼블릭 레지스트리의 컨테이너 이미지를 사용하여 배포를 생성할 수 있습니다. 사용자의 자체 컨테이너 이미지를 생성하고 컨테이너 서비스에 푸시하려면, 컨테이너 이미지를 생성하려는 동일한 컴퓨터에 다음과 같은 추가 소프트웨어를 설치해야 합니다.

- Docker - Lightsail 컨테이너 서비스에서 사용할 수 있는 자체 컨테이너 이미지를 실행, 테스트 및 생성합니다.
- AWS Command Line Interface (AWS CLI) - 생성한 컨테이너 이미지의 파라미터를 지정한 다음 Lightsail 컨테이너 서비스로 푸시합니다. 버전 2.1.1 이상은 Lightsail Control 플러그 인과 함께 사용할 수 있습니다.
- Lightsail 제어(lightsailctl) 플러그인 -가 로컬 시스템에 있는 컨테이너 이미지에 액세스할 수 AWS CLI 있도록 합니다.

이 가이드의 다음 섹션에서는 이러한 소프트웨어 패키지를 다운로드할 수 있는 위치와 설치 방법을 안내합니다. 컨테이너 서비스에 대한 자세한 내용은 컨테이너 서비스를 참조하세요.

목차

- Docker 설치
- 설치 AWS CLI
- Lightsail Control 플러그 인 설치
  - Windows에 lightsailctl 플러그인 설치
  - macOS에 lightsailctl 플러그인 설치
  - Linux에 lightsailctl 플러그인 설치

Docker 설치

Docker는 Linux 컨테이너를 기반으로 하는 분산 애플리케이션을 구축, 실행, 테스트 및 배포할 수 있는 기술입니다. Lightsail 컨테이너 서비스와 함께 사용할 수 있는 자체 컨테이너 이미지를 생성하려면

Docker 소프트웨어를 설치하고 사용해야 합니다. 자세한 내용은 Lightsail 컨테이너 서비스의 컨테이너 이미지 생성을 참조하세요.

Docker는 최신 Linux 배포 버전(Ubuntu 등)을 비롯하여 macOS 및 Windows 등 다양한 운영 체제에서 사용할 수 있습니다. 특정 운영 체제에 Docker를 설치하는 방법에 대한 자세한 내용은 Docker 설치 가이드를 참조하세요.

> **Note**
>
> 항상 최신 Docker 버전을 설치해야 합니다. 이전 버전의 Docker는이 가이드의 뒷부분에 설명된 AWS CLI 및 Lightsail 제어(lightsailctl) 플러그인과 함께 사용할 수 없습니다.

설치 AWS CLI

AWS CLI 는 명령줄 셸의 명령을 Lightsail사용하여와 같은 AWS 서비스와 상호 작용할 수 있는 오픈 소스 도구입니다. AWS CLI 를 설치하고 사용하여 로컬 시스템에서 생성된 컨테이너 이미지를 Lightsail 컨테이너 서비스로 푸시해야 합니다.

AWS CLI 는 다음 버전에서 사용할 수 있습니다.

- 버전 2.x – 일반적으로 사용 가능한 최신 버전의 AWS CLI입니다. 이 버전은의 최신 메이저 버전 AWS CLI 이며 컨테이너 이미지를 Lightsail 컨테이너 서비스로 푸시하는 기능을 포함하여 모든 최신 기능을 지원합니다. 버전 2.1.1 이상은 Lightsail Control 플러그 인과 함께 사용할 수 있습니다.
- 버전 1.x - 이전 버전과의 호환성을 위해 사용할 수 AWS CLI 있는의 이전 버전입니다. 이 버전은 컨테이너 이미지를 Lightsail 컨테이너 서비스에 푸시하는 기능을 지원하지 않습니다. 따라서 버전 AWS CLI 2를 대신 설치해야 합니다.

AWS CLI 버전 2는 Linux, macOS 및 Windows 운영 체제에서 사용할 수 있습니다. 이러한 운영 체제에를 설치하는 방법에 AWS CLI 대한 지침은 AWS CLI 사용 설명서의 AWS CLI 버전 2 설치를 참조하세요.

Lightsail Control 플러그 인 설치

Lightsail Control(lightsailctl) 플러그인은가 로컬 시스템에서 생성한 컨테이너 이미지에 AWS CLI 액세스할 수 있도록 하는 경량 애플리케이션입니다. 이 플러그 인을 사용하면 컨테이너 이미지를 Lightsail 컨테이너 서비스로 푸시하여 서비스에 배포할 수 있습니다.

시스템 요구 사항

- 64비트를 지원하는 Windows, macOS 또는 Linux 운영 체제가 필요합니다.
- AWS CLI lightsailctl 플러그인을 사용하려면 버전 2를 로컬 시스템에 설치해야 합니다. 자세한 내용은 이 설명서 앞부분의 AWS CLI설치 섹션을 참조하세요.

최신 버전의 lightsailctl 플러그인 사용

lightsailctl 플러그인은 때때로 향상된 기능으로 업데이트됩니다. lightsailctl 플러그인을 사용할 때마다 최신 버전을 사용하고 있는지 확인합니다. 새 버전이 지원되는 경우, 최신 기능을 사용하려면 최신 버전으로 업데이트하라는 메시지가 표시됩니다. 업데이트된 버전을 사용할 수 있는 경우 설치 프로세스를 반복하여 최신 버전의 lightsailctl 플러그인을 가져와야 합니다.

다음은 lightsailctl 플러그인의 모든 릴리스와 각 버전에 포함된 기능 및 개선 사항을 나열합니다.

- v1.0.0(2020년 11월 12일 릴리스) - 최초 릴리스에서는 AWS CLI 버전 2가 컨테이너 이미지를 Lightsail 컨테이너 서비스로 푸시하는 기능을 추가합니다.

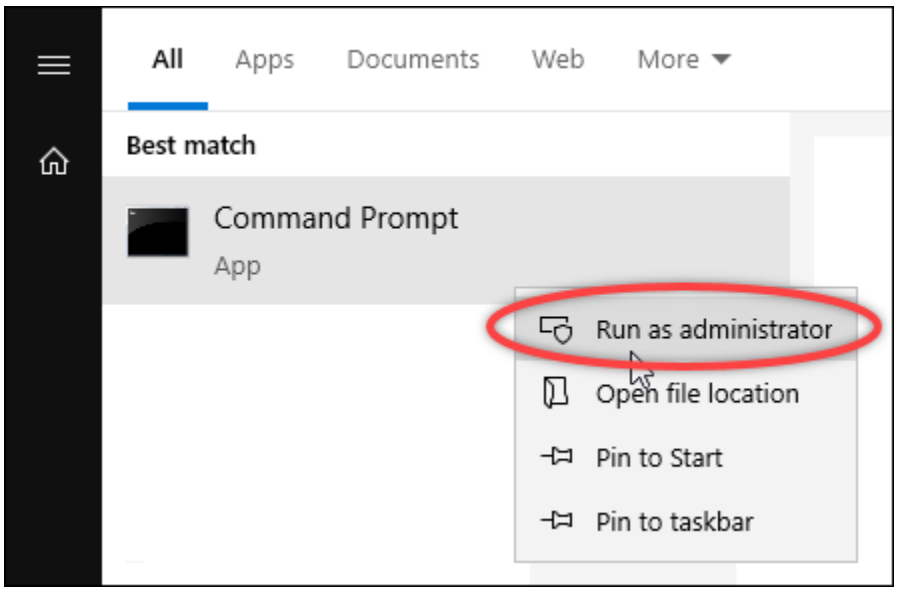
Windows에 lightsailctl 플러그인 설치

Windows에 lightsailctl 플러그인을 설치하려면 다음 절차를 완료하세요.

1. 다음 URL에서 실행 파일을 다운로드하고 `C:\Temp\lightsailctl\` 디렉터리에 저장합니다.

   ```
   https://s3.us-west-2.amazonaws.com/lightsailctl/latest/windows-amd64/
   lightsailctl.exe
   ```

2. Windows 시작(Windows Start) 버튼을 선택한 다음 cmd를 검색합니다.
3. 검색 결과에서 명령 프롬프트(Command Prompt) 애플리케이션을 마우스 오른쪽 버튼으로 클릭하고 관리자 권한으로 실행(Run as administrator)을 선택합니다.



> **Note**
>
> 명령 프롬프트(Command Prompt)에서 디바이스를 변경할 수 있도록 허용할지 묻는 메시지가 나타날 수 있습니다. 설치를 계속하려면 예(Yes)를 선택해야 합니다.

4. 다음 명령을 입력하여 lightsailctl 플러그인을 저장한 `C:\Temp\lightsailctl\` 디렉터리를 가리키는 경로 환경 변수를 설정합니다.

```
setx PATH "%PATH%;C:\Temp\lightsailctl" /M
```

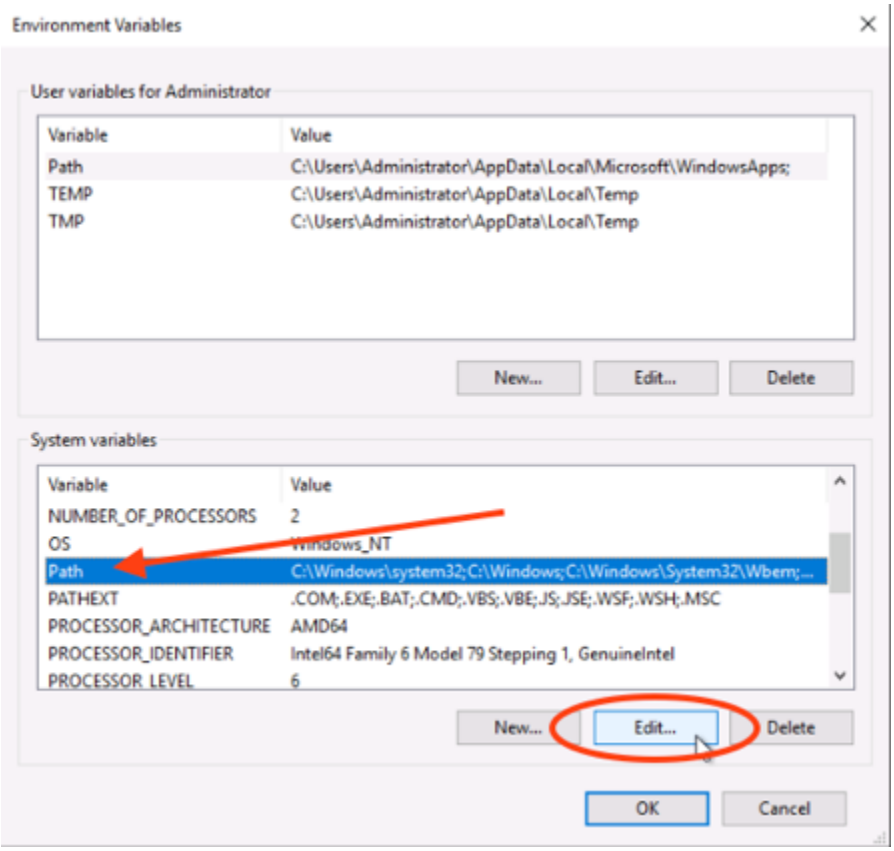
다음 예제와 비슷한 결과가 나타나야 합니다.

```
C:\WINDOWS\system32>setx PATH "%PATH%;C:\Temp\lightsailctl\" /M

SUCCESS: Specified value was saved.
```

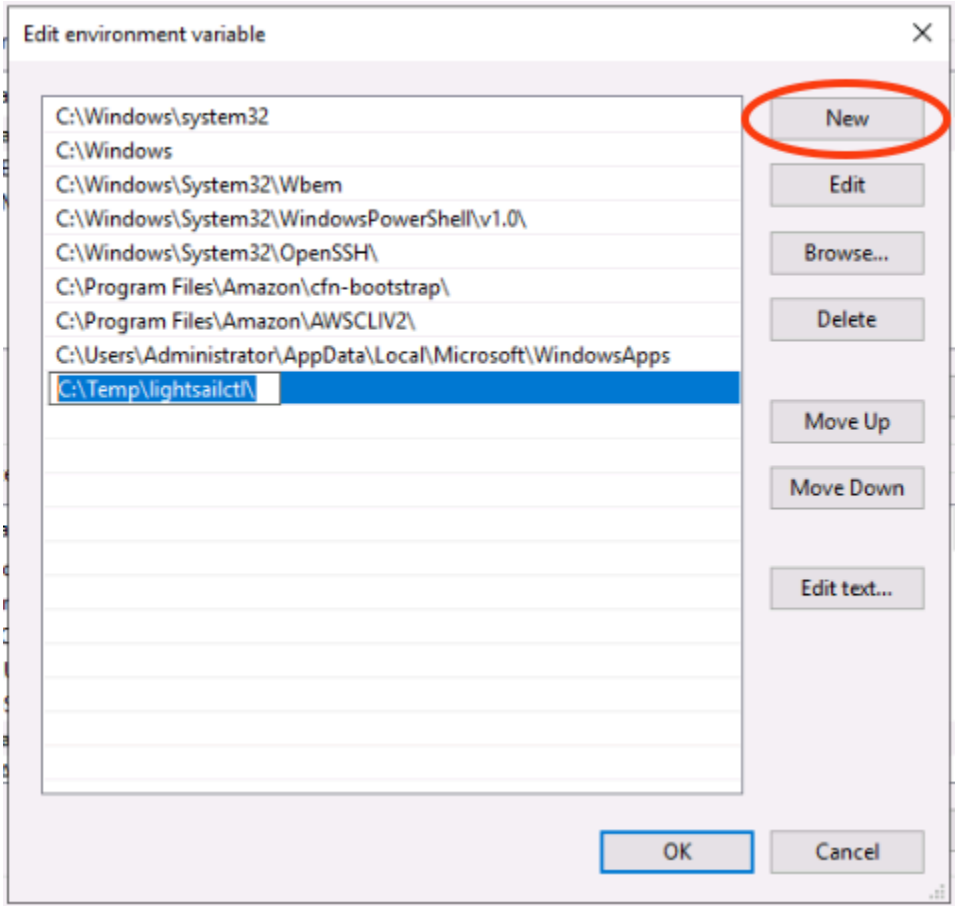
이`setx` 명령은 1024자를 초과하여 잘립니다. PATH에 이미 여러 변수가 설정되어 있는 경우 다음 절차를 사용하여 경로 환경 변수를 수동으로 설정하세요.

1. 시작(Start) 메뉴에서 제어판(Control Panel)을 클릭합니다.
2. 시스템 및 보안(System and Security)을 선택한 다음 시스템(System)을 선택합니다.
3. 고급 시스템 설정을 선택합니다.

4. 시스템 속성(System Properties) 대화 상자에서 고급(Advanced) 탭을 선택한 다음 [환경 변수(Environment Variables)를 선택합니다.
5. 환경 변수(Environment Variables) 대화 상자의 시스템 변수(System Variables) 상자에서 경로(Path)를 선택합니다.
6. 시스템 변수(System Variables) 대화 상자 아래에 있는 편집(Edit) 버튼을 선택합니다.



7. 신규(New)를 선택한 후 다음 경로를 입력합니다.`C:\Temp\lightsailctl\`



8. 세 개의 연속된 대화 상자에서 확인(OK)을 선택한 다음 시스템(System) 대화 상자를 닫습니다.

이제 AWS Command Line Interface (AWS CLI)를 사용하여 컨테이너 이미지를 Lightsail 컨테이너 서비스로 푸시할 준비가 되었습니다. 자세한 내용은 컨테이너 이미지 푸시 및 관리를 참조하세요.

macOS에 lightsailctl 플러그인 설치

다음 절차 중 하나를 완료하여 macOS에 lightsailctl 플러그인을 다운로드하고 설치합니다.

Homebrew 다운로드 및 설치

1. 터미널 창을 엽니다.
2. 다음 명령을 입력하여 lightsailctl 플러그인을 다운로드하고 설치합니다.

```
brew install aws/tap/lightsailctl
```

> **Note**
>
> Homebrew에 대한 자세한 내용은 Homebrew 웹 사이트를 참조하세요.

수동 다운로드 및 설치

1. 터미널 창을 엽니다.
2. 다음 명령을 입력하여 lightsailctl 플러그인을 다운로드하고 bin 폴더에 복사합니다.

```
curl "https://s3.us-west-2.amazonaws.com/lightsailctl/latest/darwin-amd64/
lightsailctl" -o "/usr/local/bin/lightsailctl"
```

3. 다음 명령을 입력하여 플러그 인을 실행할 수 있도록 합니다.

```
chmod +x /usr/local/bin/lightsailctl
```

4. 다음 명령을 입력하여 플러그 인의 확장 속성을 지웁니다.

```
xattr -c /usr/local/bin/lightsailctl
```

이제 AWS CLI 를 사용하여 컨테이너 이미지를 Lightsail 컨테이너 서비스로 푸시할 준비가 되었습니다. 자세한 내용은 컨테이너 이미지 푸시 및 관리를 참조하세요.

Linux에 lightsailctl 플러그인 설치

Linux에 Lightsail 컨테이너 서비스 플러그 인을 설치하려면 다음 절차를 완료하세요.

1. 터미널 창을 엽니다.
2. 다음 명령을 입력하여 lightsailctl 플러그인을 다운로드합니다.

   - AMD 64비트 아키텍처 버전의 플러그 인:

   ```
   curl "https://s3.us-west-2.amazonaws.com/lightsailctl/latest/linux-amd64/
   lightsailctl" -o "/usr/local/bin/lightsailctl"
   ```

   - ARM 64비트 아키텍처 버전의 플러그 인:

```
curl "https://s3.us-west-2.amazonaws.com/lightsailctl/latest/linux-arm64/
lightsailctl" -o "/usr/local/bin/lightsailctl"
```

3. 다음 명령을 입력하여 플러그 인을 실행할 수 있도록 합니다.

```
sudo chmod +x /usr/local/bin/lightsailctl
```

이제 AWS CLI 를 사용하여 컨테이너 이미지를 Lightsail 컨테이너 서비스로 푸시할 준비가 되었습니다. 자세한 내용은 컨테이너 이미지 푸시 및 관리를 참조하세요.

Amazon ECR 프라이빗 리포지토리에 대한 액세스 권한을 Lightsail 컨테이너 서비스에 부여

Amazon Elastic Container Registry(Amazon ECR)는 AWS Identity and Access Management (IAM)를 사용하는 리소스 기반 권한이 있는 프라이빗 리포지토리를 지원하는 AWS 관리형 컨테이너 이미지 레지스트리 서비스입니다. Amazon ECR 프라이빗 리포지토리 AWS 리전에 대한 액세스 권한을 Amazon Lightsail 컨테이너 서비스에 부여할 수 있습니다. 그런 다음 프라이빗 리포지토리에서 컨테이너 서비스로 이미지를 배포할 수 있습니다.

Lightsail 콘솔 또는 AWS Command Line Interface ()를 사용하여 Lightsail 컨테이너 서비스 및 Amazon ECR 프라이빗 리포지토리에 대한 액세스를 관리할 수 있습니다AWS CLI. 그러나 과정을 단순화하는 Lightsail 콘솔을 사용하는 것이 좋습니다.

컨테이너 서비스에 대한 자세한 내용은 컨테이너 서비스를 참조하세요. Amazon ECR에 대한 자세한 내용은 Amazon ECR 사용 설명서를 참조하세요.

목차

- 필수 권한
- Lightsail 콘솔을 사용하여 프라이빗 리포지토리에 대한 액세스 관리
- AWS CLI 를 사용하여 프라이빗 리포지토리에 대한 액세스 관리
  - Amazon ECR 이미지 풀러 IAM 역할 활성화 또는 비활성화
  - Amazon ECR 프라이빗 리포지토리에 정책 문이 있는지 확인
    - 정책 문이 없는 프라이빗 리포지토리에 정책 추가
    - 정책 문이 있는 프라이빗 리포지토리에 정책 추가

필수 권한

Amazon ECR 프라이빗 리포지토리에 대한 Lightsail 컨테이너 서비스 액세스를 관리할 사용자는 IAM에서 다음 권한 정책 중 하나를 가지고 있어야 합니다. 자세한 내용은 AWS Identity and Access Management 사용 설명서의 IAM 자격 증명 권한 추가 및 제거 섹션을 참조하세요.

모든 Amazon ECR 프라이빗 리포지토리에 대한 액세스 권한 부여

다음 권한 정책은 모든 Amazon ECR 프라이빗 리포지토리에 대한 액세스를 구성할 수 있는 권한을 사용자에게 부여합니다.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "ManageEcrPrivateRepositoriesAccess",
            "Effect": "Allow",
            "Action": [
                "ecr:SetRepositoryPolicy",
                "ecr:DescribeRepositories",
                "ecr:DeleteRepositoryPolicy",
                "ecr:GetRepositoryPolicy"
            ],
            "Resource": "arn:aws:ecr:*:111122223333:repository/*"
        }
    ]
}
```

정책에서 *AwsAccountId*를 AWS 계정 ID 번호로 바꿉니다.

특정 Amazon ECR 프라이빗 리포지토리에 대한 액세스 권한 부여

다음 권한 정책은 특정 AWS 리전의 특정 Amazon ECR 프라이빗 리포지토리에 대한 액세스를 구성할 수 있는 권한을 사용자에게 부여합니다.

정책에서 아래 예제 텍스트를 사용자의 값으로 바꿉니다.

- *AwsRegion* - 프라이빗 리포지토리의 AWS 리전 코드(예: `us-east-1`)입니다. Lightsail 컨테이너 서비스는 액세스하려는 프라이빗 리포지토리 AWS 리전 와 동일한에 있어야 합니다.

- *AwsAccountId* - AWS 계정 ID 번호입니다.
- *RepositoryName* - 액세스를 관리하려는 프라이빗 리포지토리의 이름입니다.

다음은 예제 값으로 채워진 권한 정책의 예제입니다.

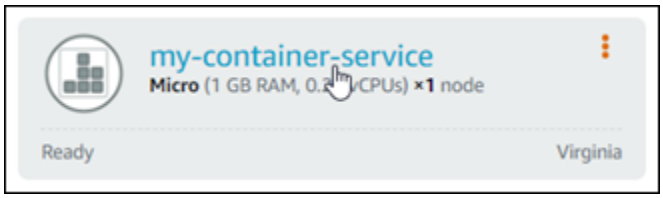
JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "ManageEcrPrivateRepositoriesAccess",
            "Effect": "Allow",
            "Action": [
                "ecr:SetRepositoryPolicy",
                "ecr:DescribeRepositories",
                "ecr:DeleteRepositoryPolicy",
                "ecr:GetRepositoryPolicy"
            ],
            "Resource": "arn:aws:ecr:us-east-1:111122223333:repository/my-
private-repo"
        }
    ]
}
```

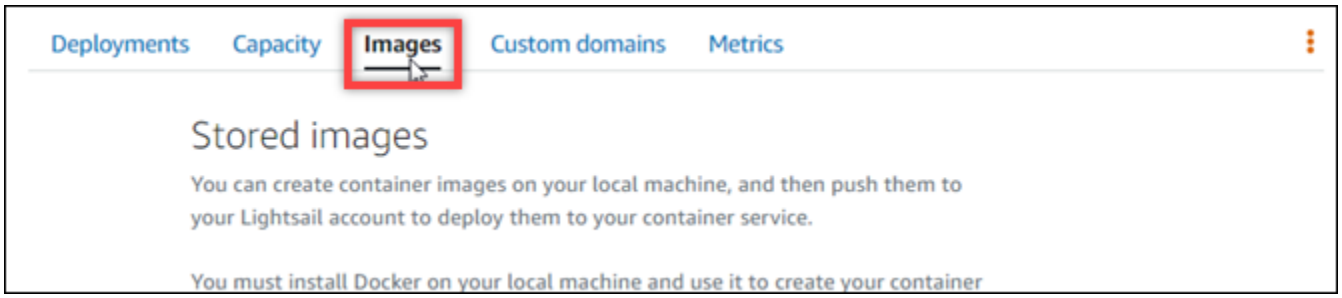
Lightsail 콘솔을 사용하여 프라이빗 리포지토리에 대한 액세스 관리

Lightsail 콘솔을 사용하여 Amazon ECR 프라이빗 리포지토리에 대한 Lightsail 컨테이너 서비스의 액세스를 관리하려면 다음 절차를 완료하세요.

1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창의 이미지에서 컨테이너를 선택합니다.
3. Amazon ECR 프라이빗 리포지토리에 대한 액세스를 구성하려는 컨테이너 서비스의 이름을 선택합니다.

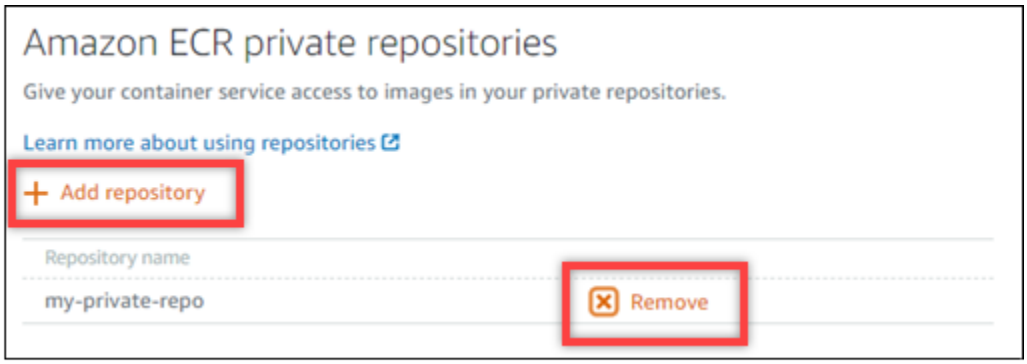


4. 이미지(Images) 탭을 선택합니다.

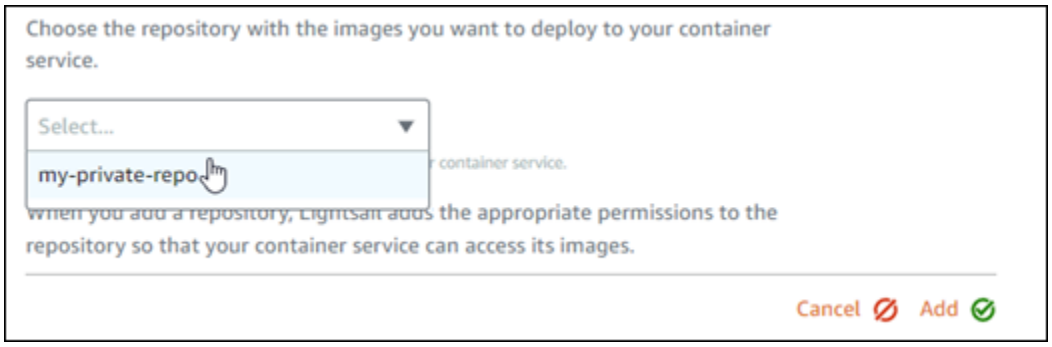


5. Amazon ECR 프라이빗 리포지토리에 컨테이너 서비스에 대한 액세스 권한을 부여하려면 리포지토리 추가를 선택합니다.

> **Note**
>
> 이전에 추가한 Amazon ECR 프라이빗 리포지토리에서 컨테이너 서비스에 대한 액세스를 제거하려면 제거를 선택할 수 있습니다.

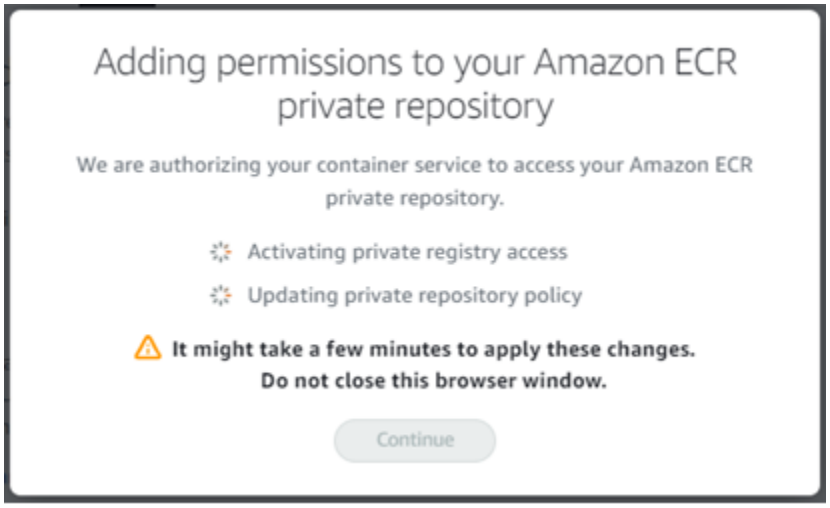


6. 표시되는 드롭다운에서 액세스하려는 프라이빗 리포지토리를 선택한 다음 추가(Add)를 선택합니다.



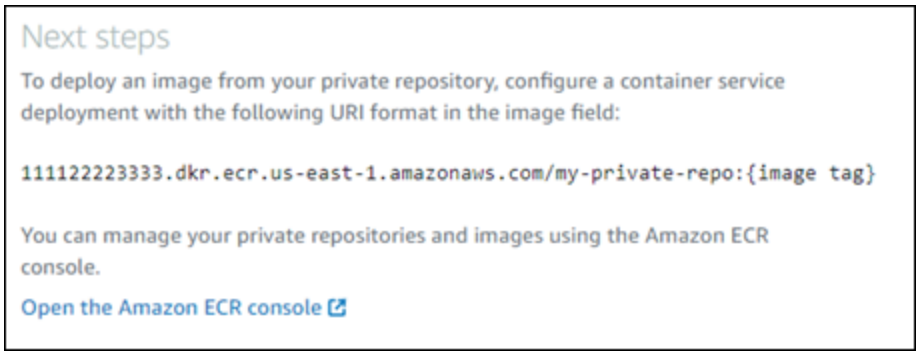
Lightsail이 보안 주체 Amazon 리소스 이름(ARN)을 포함하는 컨테이너 서비스에 대한 Amazon ECR 이미지 풀러 IAM 역할을 활성화하는 데 몇 분 정도 걸립니다. 그런 다음 Lightsail은 선택한

Amazon ECR 프라이빗 리포지토리의 권한 정책에 IAM 역할 보안 주체 ARN을 자동으로 추가합니다. 그러면 컨테이너 서비스에 프라이빗 리포지토리와 해당 이미지에 대한 액세스 권한이 부여됩니다. 프로세스가 완료되고 계속(Continue)을 선택할 수 있음을 나타내는 모달이 표시될 때까지 브라우저 창을 닫지 마세요.



7. 활성화가 완료되면 계속(Continue)을 선택합니다.

선택한 Amazon ECR 프라이빗 리포지토리가 추가되면 페이지의 Amazon ECR 프라이빗 리포지토리 섹션에 나열됩니다. 이 페이지에는 프라이빗 리포지토리에서 Lightsail컨테이너 서비스로 이미지를 배포하는 방법에 대한 지침이 포함되어 있습니다. 프라이빗 리포지토리에서 이미지를 사용하려면 컨테이너 서비스 배포를 생성할 때 이미지(Image) 값으로 페이지에 표시되는 URI 형식을 지정합니다. 지정한 URI에서 예시 *{image tag}*를 배포하려는 이미지의 태그로 바꿉니다. 자세한 내용은 컨테이너 서비스 배포 생성 및 관리를 참조하세요.



AWS CLI 를 사용하여 프라이빗 리포지토리에 대한 액세스 관리

AWS Command Line Interface (AWS CLI)를 사용하여 Amazon ECR 프라이빗 리포지토리에 대한 Lightsail 컨테이너 서비스 액세스를 관리하려면 다음 단계가 필요합니다.

> **Important**
>
> 과정을 단순화하기 때문에 Lightsail 콘솔을 사용하여 Amazon ECR 프라이빗 리포지토리에 대한 Lightsail 컨테이너 서비스 액세스를 관리하는 것이 좋습니다. 자세한 내용을 알아보려면 이 가이드 앞부분의 Lightsail 콘솔을 사용하여 프라이빗 리포지토리에 대한 액세스 관리를 참조하세요.

1. Amazon ECR 이미지 풀러 IAM 역할 활성화 또는 비활성화 -에 AWS CLI `update-container-service` 대한 명령을 사용하여 Amazon ECR 이미지 풀러 IAM 역할을 Lightsail 활성화 또는 비활성화합니다. 활성화하면 Amazon ECR 이미지 풀러 IAM 역할에 대한 보안 주체 Amazon 리소스 이름(ARN)이 생성됩니다. 자세한 내용은, 이 설명서의 Amazon ECR 이미지 풀러 IAM 역할 활성화 또는 비활성화 섹션을 참조하세요.
2. Amazon ECR 프라이빗 리포지토리에 정책 문이 있는지 확인 - Amazon ECR 이미지 풀러 IAM 역할을 활성화한 후 컨테이너 서비스로 액세스하려는 Amazon ECR 프라이빗 리포지토리에 기존 정책 문이 있는지 확인해야 합니다. 자세한 내용은, 이 설명서 뒷부분의 Amazon ECR 프라이빗 리포지토리에 정책 문이 있는지 확인을 참조하세요.

   리포지토리에 기존 정책 문이 있는지에 따라 다음 방법 중 하나를 사용하여 IAM 역할 보안 주체 ARN을 리포지토리에 추가합니다.

   a. 정책 문이 없는 프라이빗 리포지토리에 정책 추가 - Amazon ECR용 `set-repository-policy` 명령을 사용하여 AWS CLI 컨테이너 서비스의 Amazon ECR 이미지 풀러 역할 보안 주체 ARN을 기존 정책이 있는 프라이빗 리포지토리에 추가합니다. 자세한 내용을 알아보려면, 이 가이드 뒷부분의 정책 문이 없는 프라이빗 리포지토리에 정책 추가를 참조하세요.

   b. 정책 문이 있는 프라이빗 리포지토리에 정책 추가 - Amazon ECR용 `set-repository-policy` 명령을 사용하여 AWS CLI 컨테이너 서비스에 대한 Amazon ECR 이미지 풀러 역할을 기존 정책이 없는 프라이빗 리포지토리에 추가합니다. 자세한 내용을 알아보려면 이 가이드 뒷부분의 정책 문이 있는 프라이빗 리포지토리에 정책 추가를 참조하세요.

Amazon ECR 이미지 풀러 IAM 역할 활성화 또는 비활성화

Lightsail 컨테이너 서비스에 대한 Amazon ECR 이미지 풀러 IAM 역할을 활성화하거나 비활성화하려면 다음 절차를 완료하세요. 에 대한 `update-container-service` 명령을 사용하여 AWS CLI Amazon ECR 이미지 풀러 IAM 역할을 활성화하거나 비활성화할 수 있습니다Lightsail. 자세한 내용은 AWS CLI 명령 레퍼런스의 update-container-service를 참조하세요.

> **Note**
>
> 이 절차를 계속하려면 Lightsail 먼저 AWS CLI 를 설치하고에 맞게 구성해야 합니다. 자세한 내용은 AWS CLI 가 Lightsail과 함께 작동하도록 구성을 참조하세요.

1. 명령 프롬프트 또는 터미널 창을 엽니다.
2. 다음 명령을 입력하여 컨테이너 서비스를 업데이트하고 Amazon ECR 이미지 풀러 IAM 역할을 활성화 또는 비활성화합니다.

```
aws lightsail update-container-service --service-name ContainerServiceName --
private-registry-access ecrImagePullerRole={isActive=RoleActivationState} --
region AwsRegionCode
```

명령에서 아래 예 텍스트를 사용자의 값으로 대체합니다.

- *ContainerServiceName* - Amazon ECR 이미지 풀러 IAM 역할을 활성화하거나 비활성화할 컨테이너 서비스의 이름입니다.
- *RoleActivationState* - Amazon ECR 이미지 풀러 IAM 역할의 활성화 상태입니다. `true`를 지정하여 역할을 활성화하거나 `false`를 지정하여 비활성화합니다.
- *AwsRegionCode* - 컨테이너 서비스의 AWS 리전 코드(예: `us-east-1`)입니다.

예시:

- Amazon ECR 이미지 풀러 IAM 역할 활성화

```
aws lightsail update-container-service --service-name my-container-service --
private-registry-access ecrImagePullerRole={isActive=true} --region us-east-1
```

- Amazon ECR mazon ECR 이미지 풀러 IAM 역할 비활성화

```
aws lightsail update-container-service --service-name my-container-service --
private-registry-access ecrImagePullerRole={isActive=false} --region us-east-1
```

3. 경우에 따라 다음 작업을 수행합니다.

- Amazon ECR 이미지 풀러 역할을 활성화한 경우 - 이전 응답을 받은 후 30초 이상 기다립니다. 그리고 다음 단계로 계속 진행하여 컨테이너 서비스에 대한 Amazon ECR 이미지 풀러 IAM 역할의 보안 주체 ARN을 가져옵니다.
- Amazon ECR 이미지 풀러 역할 비활성화한 경우 - 이전에 Amazon ECR 이미지 풀러 IAM 역할 보안 주체 ARN을 Amazon ECR 프라이빗 리포지토리의 권한 정책에 추가한 경우 리포지토리에서 해당 권한 정책을 제거해야 합니다. 자세한 내용은 Amazon ECR 사용 설명서의 프라이빗 리포지토리 정책 문 삭제를 참조하세요.

4. 다음 명령을 입력하여 컨테이너 서비스에 대한 Amazon ECR 이미지 풀러 IAM 역할의 보안 주체 ARN을 가져옵니다.

```
aws lightsail get-container-services --service-name ContainerServiceName --
region AwsRegionCode
```

명령에서 아래 예 텍스트를 사용자의 값으로 대체합니다.

- *ContainerServiceName* - Amazon ECR 이미지 풀러 IAM 역할 보안 주체 ARN을 가져올 컨테이너 서비스의 이름입니다.
- *AwsRegionCode* - 컨테이너 서비스의 AWS 리전 코드(예: `us-east-1`)입니다.

예제:

```
aws lightsail get-container-services --service-name my-container-service --
region us-east-1
```

응답에서 ECR 이미지 풀러 IAM 역할 보안 주체 ARN을 찾습니다. 역할이 나열되면 복사하거나 기록해 둡니다. 이 가이드의 다음 섹션에 필요합니다. 다음으로, 컨테이너 서비스로 액세스하려는 Amazon ECR 프라이빗 리포지토리에 기존 정책 문이 있는지 확인해야 합니다. 이 설명서의 Amazon ECR 프라이빗 리포지토리에 정책 문이 있는지 확인 섹션으로 계속 진행합니다.

Amazon ECR 프라이빗 리포지토리에 정책 문이 있는지 확인

다음 절차를 사용하여 Amazon ECR 프라이빗 리포지토리에 정책 문이 있는지 확인합니다. Amazon ECR에 AWS CLI `get-repository-policy` 명령을 사용할 수 있습니다. 자세한 내용은 AWS CLI 명령 레퍼런스의 update-container-service를 참조하세요.

> **Note**
>
> 이 절차를 계속하려면 먼저를 설치하고 Amazon ECR에 맞게 AWS CLI 구성해야 합니다. 자세한 내용은 Amazon ECR 사용 설명서의 Amazon ECR을 사용하여 설정을 참조하세요.

1. 명령 프롬프트 또는 터미널 창을 엽니다.
2. 다음 명령을 입력하여 특정 프라이빗 리포지토리에 대한 정책 문을 가져옵니다.

```
aws ecr get-repository-policy --repository-name RepositoryName --
region AwsRegionCode
```

명령에서 아래 예 텍스트를 사용자의 값으로 대체합니다.

- *RepositoryName* - Lightsail 컨테이너 서비스에 대한 액세스를 구성하려는 프라이빗 리포지토리의 이름을 선택합니다.
- *AwsRegionCode* - 프라이빗 리포지토리의 AWS 리전 코드(예: us-east-1)입니다.

예제:

```
aws ecr get-repository-policy --repository-name my-private-repo --region us-east-1
```

다음과 같은 응답 중 하나가 표시됩니다.

- RepositoryPolicyNotFoundException - 프라이빗 리포지토리에 정책 문이 없습니다. 리포지토리에 정책 문이 없으면 이 가이드 뒷부분의 정책 문이 없는 프라이빗 리포지토리에 정책 추가 섹션의 단계를 따릅니다.

```
C:\>aws ecr get-repository-policy --repository-name my-private-repo

An error occurred (RepositoryPolicyNotFoundException) when calling the GetRepositoryPolicy operation: Repository policy
does not exist for the repository with name 'my-private-repo' in the registry with id '[redacted]'
```

- 리포지토리 정책이 발견됨 - 프라이빗 리포지토리에 정책 문이 있으며 요청 응답에 표시됩니다. 리포지토리에 정책 문이 있으면 기존 정책을 복사한 다음 이 가이드 뒷부분의 정책 문이 없는 프라이빗 리포지토리에 정책 추가 섹션의 단계를 따릅니다.

```
C:\>aws ecr get-repository-policy --repository-name my-private-repo
{
    "registryId": "[redacted]",
    "repositoryName": "my-private-repo",
    "policyText": "{\n  \"Version\" : \"2012-10-17\",\n  \"Statement\" : [ {\n    \"Sid\" : \"AllowUserPushPull\",\n
\"Effect\" : \"Allow\",\n    \"Principal\" : {\n      \"AWS\" : \"arn:aws:iam::[redacted]:user/example-user\"\n    },\
n    \"Action\" : [ \"ecr:BatchGetImage\", \"ecr:BatchCheckLayerAvailability\", \"ecr:CompleteLayerUpload\", \"ecr:GetDo
wnloadUrlForLayer\", \"ecr:InitiateLayerUpload\", \"ecr:PutImage\", \"ecr:UploadLayerPart\" ]\n  } ]\n}"
}
```

정책 문이 없는 프라이빗 리포지토리에 정책 추가

정책 문이 없는 Amazon ECR 프라이빗 리포지토리에 정책을 추가하려면 다음 절차를 완료하세요. 추가하는 정책에는 Lightsail 컨테이너 서비스의 Amazon ECR 이미지 풀러 IAM 역할 보안 주체 ARN이 포함되어야 합니다. 그러면 컨테이너 서비스가 프라이빗 리포지토리에서 이미지를 배포할 수 있는 액세스 권한이 부여됩니다.

> **Important**
>
> Lightsail은 사용자가 Lightsail 콘솔을 사용하여 액세스를 구성할 때 Amazon ECR 이미지 풀러 역할을 Amazon ECR 프라이빗 리포지토리에 자동으로 추가합니다. 이 경우 이 섹션의 절차를 사용하여 Amazon ECR 이미지 풀러 역할을 프라이빗 리포지토리에 수동으로 추가할 필요가 없습니다. 자세한 내용을 알아보려면 이 가이드 앞부분의 Lightsail 콘솔을 사용하여 프라이빗 리포지토리에 대한 액세스 관리를 참조하세요.

AWS CLI를 사용하여 프라이빗 리포지토리에 정책을 추가할 수 있습니다. 정책이 포함된 JSON 파일을 생성한 다음 Amazon ECR에 대한 `set-repository-policy` 명령으로 해당 파일을 참조하면 됩니다. 자세한 내용은 AWS CLI 명령 레퍼런스의 set-repository-policy를 참조하세요.

> **Note**
>
> 이 절차를 계속하기 전에를 설치하고 Amazon ECR에 맞게 AWS CLI 구성해야 합니다. 자세한 내용은 Amazon ECR 사용 설명서의 Amazon ECR을 사용하여 설정을 참조하세요.

1. 텍스트 편집기를 열고 다음 정책 문을 새 텍스트 파일에 붙여 넣습니다.

   JSON

```
{
```

```
  "Version":"2012-10-17",
  "Statement": [
  {
      "Sid": "AllowLightsailPull-ecr-private-repo-demo",
      "Effect": "Allow",
      "Principal": {
        "AWS": "IamRolePrincipalArn"
      },
      "Action": [
        "ecr:BatchGetImage",
        "ecr:GetDownloadUrlForLayer"
      ]
    }
  ]
}
```

텍스트에서 *IamRolePrincipalArn*을 이 설명서의 앞부분에서 얻은 컨테이너 서비스의 Amazon ECR 이미지 풀러 IAM 역할 보안 주체 ARN으로 바꿉니다.

2. 파일을 컴퓨터의 액세스 가능한 위치에 `ecr-policy.json`으로 저장합니다(예: Windows의 경우 `C:\Temp\ecr-policy.json`, macOS 또는 Linux의 경우 `/tmp/ecr-policy.json`).
3. 생성된 `ecr-policy.json` 파일의 경로 위치를 기록해 둡니다. 이 절차의 뒷부분에 나오는 명령에서 이 위치를 지정합니다.
4. 명령 프롬프트 또는 터미널 창을 엽니다.
5. 다음 명령을 입력하여 컨테이너 서비스로 액세스하려는 프라이빗 리포지토리에 대한 정책 문을 설정합니다.

```
aws ecr set-repository-policy --repository-name RepositoryName --policy-text
 file://path/to/ecr-policy.json --region AwsRegionCode
```

명령에서 아래 예 텍스트를 사용자의 값으로 대체합니다.

- *RepositoryName* - 정책을 추가하려는 프라이빗 리포지토리의 이름입니다.
- *path/to/* - 이 설명서의 앞부분에서 생성한 컴퓨터의 `ecr-policy.json` 파일 경로입니다.
- *AwsRegionCode* - 프라이빗 리포지토리의 AWS 리전 코드(예: `us-east-1`)입니다.

예시:

- Windows의 경우:

```
aws ecr set-repository-policy --repository-name my-private-repo --policy-text
 file://C:\Temp\ecr-policy.json --region us-east-1
```

- macOS 또는 Linux에서

```
aws ecr set-repository-policy --repository-name my-private-repo --policy-text
 file:///tmp/ecr-policy.json --region us-east-1
```

이제 컨테이너 서비스에서 프라이빗 리포지토리와 해당 이미지에 액세스할 수 있습니다. 리포지토리의 이미지를 사용하려면 컨테이너 서비스 배포에 대한 이미지(Image) 값으로 다음 URI를 지정합니다. URI에서 예제 ##를 배포하려는 이미지의 태그로 바꿉니다. 자세한 내용은 컨테이너 서비스 배포 생성 및 관리를 참조하세요.

```
AwsAccountId.dkr.ecr.AwsRegionCode.amazonaws.com/RepositoryName:ImageTag
```

URI에서 아래 예제 텍스트를 사용자의 값으로 바꿉니다.

- *AwsAccountId* - AWS 계정 ID 번호입니다.
- *AwsRegionCode* - 프라이빗 리포지토리의 AWS 리전 코드(예: us-east-1)입니다.
- *RepositoryName* - 컨테이너 이미지를 배포할 프라이빗 리포지토리의 이름입니다.
- *ImageTag* - 컨테이너 서비스에 배포할 프라이빗 리포지토리의 컨테이너 이미지 태그입니다.

예제:

```
111122223333.dkr.ecr.us-east-1.amazonaws.com/my-private-repo:myappimage
```

정책 문이 있는 프라이빗 리포지토리에 정책 추가

정책 문이 있는 Amazon ECR 프라이빗 리포지토리에 정책을 추가하려면 다음 절차를 완료하세요. 추가하는 정책에는 기존 정책과 Lightsail 컨테이너 서비스의 Amazon ECR 이미지 풀러 IAM 역할 보안 주체 ARN이 들어 있는 새 정책이 포함되어야 합니다. 이렇게 하면 프라이빗 리포지토리에 대한 기존 권한이 유지되는 동시에 컨테이너 서비스가 프라이빗 리포지토리의 이미지를 배포할 수 있는 액세스 권한도 부여됩니다.

> **Important**
>
> Lightsail은 사용자가 Lightsail 콘솔을 사용하여 액세스를 구성할 때 Amazon ECR 이미지 풀러 역할을 Amazon ECR 프라이빗 리포지토리에 자동으로 추가합니다. 이 경우 이 섹션의 절차를 사용하여 Amazon ECR 이미지 풀러 역할을 프라이빗 리포지토리에 수동으로 추가할 필요가 없습니다. 자세한 내용을 알아보려면 이 가이드 앞부분의 Lightsail 콘솔을 사용하여 프라이빗 리포지토리에 대한 액세스 관리를 참조하세요.

AWS CLI를 사용하여 프라이빗 리포지토리에 정책을 추가할 수 있습니다. 기존 정책과 새 정책이 포함된 JSON 파일을 만들어 이를 수행합니다. 그런 다음 Amazon ECR에 대한 `set-repository-policy` 명령을 사용하여 해당 파일을 참조합니다. 자세한 내용은 AWS CLI 명령 레퍼런스의 set-repository-policy를 참조하세요.

> **Note**
>
> 이 절차를 계속하려면 먼저를 설치하고 Amazon ECR에 맞게 AWS CLI 구성해야 합니다. 자세한 내용은 Amazon ECR 사용 설명서의 Amazon ECR을 사용하여 설정을 참조하세요.

1. 명령 프롬프트 또는 터미널 창을 엽니다.
2. 다음 명령을 입력하여 특정 프라이빗 리포지토리에 대한 정책 문을 가져옵니다.

   ```
   aws ecr get-repository-policy --repository-name RepositoryName --
   region AwsRegionCode
   ```

   명령에서 아래 예 텍스트를 사용자의 값으로 대체합니다.

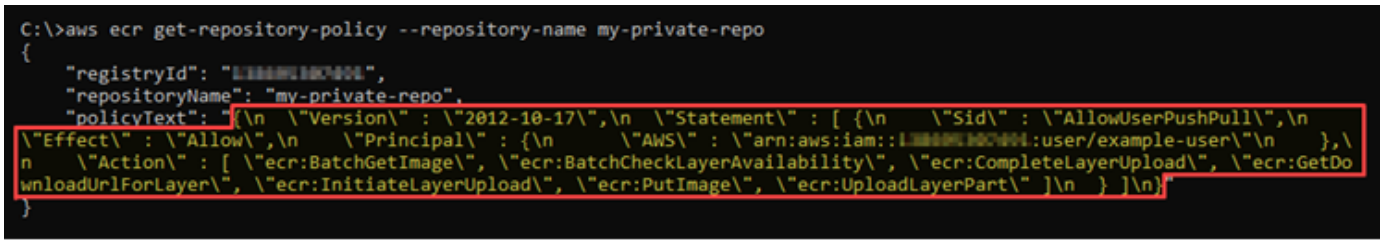
   - *RepositoryName* - Lightsail 컨테이너 서비스에 대한 액세스를 구성하려는 프라이빗 리포지토리의 이름을 선택합니다.
   - *AwsRegionCode* - 프라이빗 리포지토리의 AWS 리전 코드(예: `us-east-1`)입니다.

   예제:

   ```
   aws ecr get-repository-policy --repository-name my-private-repo --region us-east-1
   ```

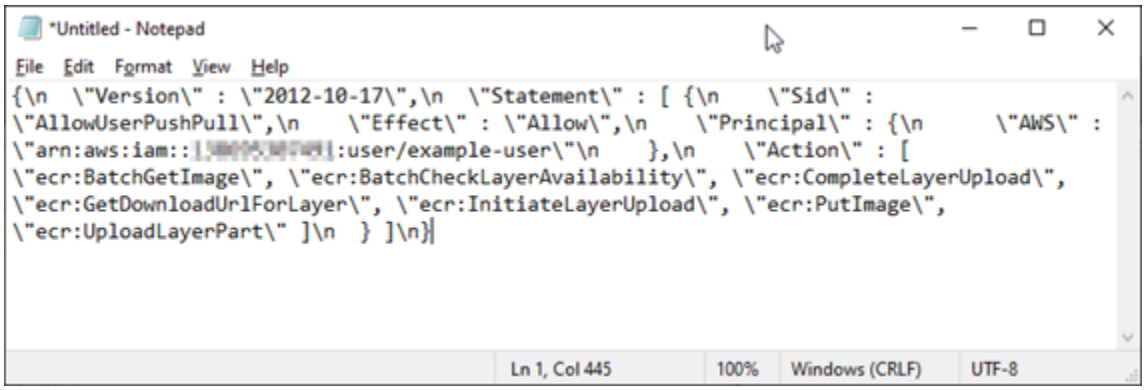
3. 응답에서 기존 정책을 복사하고 다음 단계로 계속 진행합니다.

다음 예에서 강조 표시된 것처럼 큰따옴표로 묶인 `policyText`의 내용만 복사해야 합니다.



4. 텍스트 편집기를 열고 이전 단계에서 복사한 프라이빗 리포지토리의 기존 정책을 붙여 넣습니다.

   결과는 다음 예제와 같아야 합니다.



5. 붙여 넣은 텍스트에서 `\n`을 줄 바꿈으로 바꾸고 나머지 `\`를 삭제합니다.

   결과는 다음 예제와 같아야 합니다.

6. 텍스트 파일 끝에 다음 정책 문을 붙여 넣습니다.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "AllowLightsailPull-ecr-private-repo-demo",
            "Effect": "Allow",
            "Principal": {
                "AWS": "IamRolePrincipalArn"
            },
            "Action": [
                "ecr:BatchGetImage",
                "ecr:GetDownloadUrlForLayer"
            ]
        }
    ]
}
```

7. 텍스트에서 *IamRolePrincipalArn*을 이 설명서의 앞부분에서 얻은 컨테이너 서비스의 Amazon ECR 이미지 풀러 IAM 역할 보안 주체 ARN으로 바꿉니다.

   결과는 다음 예제와 같아야 합니다.

8. 파일을 컴퓨터의 액세스 가능한 위치에 `ecr-policy.json`으로 저장합니다(예: Windows의 경우 `C:\Temp\ecr-policy.json`, macOS 또는 Linux의 경우 `/tmp/ecr-policy.json`).
9. `ecr-policy.json` 파일의 경로 위치를 기록해 둡니다. 이 절차의 뒷부분에 나오는 명령에서 이 위치를 지정합니다.
10. 명령 프롬프트 또는 터미널 창을 엽니다.

11. 다음 명령을 입력하여 컨테이너 서비스로 액세스하려는 프라이빗 리포지토리에 대한 정책 문을 설정합니다.

```
aws ecr set-repository-policy --repository-name RepositoryName --policy-text
 file://path/to/ecr-policy.json --region AwsRegionCode
```

명령에서 아래 예 텍스트를 사용자의 값으로 대체합니다.

- *RepositoryName* - 정책을 추가하려는 프라이빗 리포지토리의 이름입니다.
- *path/to/* - 이 설명서의 앞부분에서 생성한 컴퓨터의 `ecr-policy.json` 파일 경로입니다.
- *AwsRegionCode* - 프라이빗 리포지토리의 AWS 리전 코드(예: `us-east-1`)입니다.

예시:

- Windows의 경우:

```
aws ecr set-repository-policy --repository-name my-private-repo --policy-text
 file://C:\Temp\ecr-policy.json --region us-east-1
```

- macOS 또는 Linux에서

```
aws ecr set-repository-policy --repository-name my-private-repo --policy-text
 file:///tmp/ecr-policy.json --region us-east-1
```

다음 예와 비슷한 응답이 나타나는 것을 볼 수 있습니다.

```
C:\>aws ecr set-repository-policy --repository-name my-private-repo --policy-text file://C:\Temp\ecr-policy.json --regio
n us-west-2
{
    "registryId": "[redacted]",
    "repositoryName": "my-private-repo",
    "policyText": "{\n  \"Version\" : \"2012-10-17\",\n  \"Statement\" : [ {\n    \"Sid\" : \"AllowLightsailPull-my-cont
ainer-service\",\n    \"Effect\" : \"Allow\",\n    \"Principal\" : {\n      \"AWS\" : \"arn:aws:iam::[redacted]:role/a
mazon/lightsail/[redacted]/containers/my-container-service/private-repo-access/[redacted]\"\n    },\n    \"Action\" : [ \"ecr:BatchGetImage\", \"ecr:GetDownloadUrlForLayer\" ]\n  }, {\n    \"Sid\" :
\"AllowUserPushPull\",\n    \"Effect\" : \"Allow\",\n    \"Principal\" : {\n      \"AWS\" : \"arn:aws:iam::[redacted]:
user/example-user\"\n    },\n    \"Action\" : [ \"ecr:BatchCheckLayerAvailability\", \"ecr:BatchGetImage\", \"ecr:Comple
teLayerUpload\", \"ecr:GetDownloadUrlForLayer\", \"ecr:InitiateLayerUpload\", \"ecr:PutImage\", \"ecr:UploadLayerPart\"
]\n  } ]\n}"
}
```

`get-repository-policy` 명령을 다시 실행하면 프라이빗 리포지토리에 새로운 추가 정책 문이 표시됩니다. 이제 컨테이너 서비스에서 프라이빗 리포지토리와 해당 이미지에 액세스할 수 있습니다. 리포지토리의 이미지를 사용하려면 컨테이너 서비스 배포에 대한 이미지(Image) 값으로

다음 URI를 지정합니다. URI에서 예제 *##*를 배포하려는 이미지의 태그로 바꿉니다. 자세한 내용은 컨테이너 서비스 배포 생성 및 관리를 참조하세요.

```
AwsAccountId.dkr.ecr.AwsRegionCode.amazonaws.com/RepositoryName:ImageTag
```

URI에서 아래 예제 텍스트를 사용자의 값으로 바꿉니다.

- *AwsAccountId* - AWS 계정 ID 번호입니다.
- *AwsRegionCode* - 프라이빗 리포지토리의 AWS 리전 코드(예: us-east-1)입니다.
- *RepositoryName* - 컨테이너 이미지를 배포할 프라이빗 리포지토리의 이름입니다.
- *ImageTag* - 컨테이너 서비스에 배포할 프라이빗 리포지토리의 컨테이너 이미지 태그입니다.

예제:

```
111122223333.dkr.ecr.us-east-1.amazonaws.com/my-private-repo:myappimage
```

Lightsail에서 컨테이너 서비스 배포 생성 및 관리

Amazon Lightsail 컨테이너 서비스에서 컨테이너를 시작할 준비가 되면 배포를 생성합니다. 배포는 서비스에서 시작하려는 컨테이너에 대한 사양 집합입니다. 컨테이너 서비스에는 실행 중인 배포가 동시에 하나만 존재할 수 있으며, 배포에는 최대 10개의 컨테이너 항목이 포함될 수 있습니다. 컨테이너 서비스를 생성하면서 배포를 생성하거나 서비스를 설정하여 실행하고 난 후에 생성할 수 있습니다.

> **Note**
>
> 새 배포를 만들면 컨테이너 서비스의 기존 사용률 지표가 사라지고, 새로운 현재 배포에 대한 지표만 표시됩니다.

컨테이너 서비스에 대한 자세한 내용은 Amazon Lightsail의 컨테이너 서비스를 참조하세요.

목차

- 사전 조건
- 배포 파라미터
  - 컨테이너 항목 파라미터

  - 퍼블릭 엔드포인트 파라미터
- 컨테이너 간 통신
- 컨테이너 로그
- 배포 버전
- 배포 상태
- 배포 실패
- 현재 컨테이너 서비스 배포 확인
- 컨테이너 서비스 배포 생성 또는 수정

사전 조건

컨테이너 서비스에서 배포를 생성하기 전에 다음 사전 조건을 완료하세요.

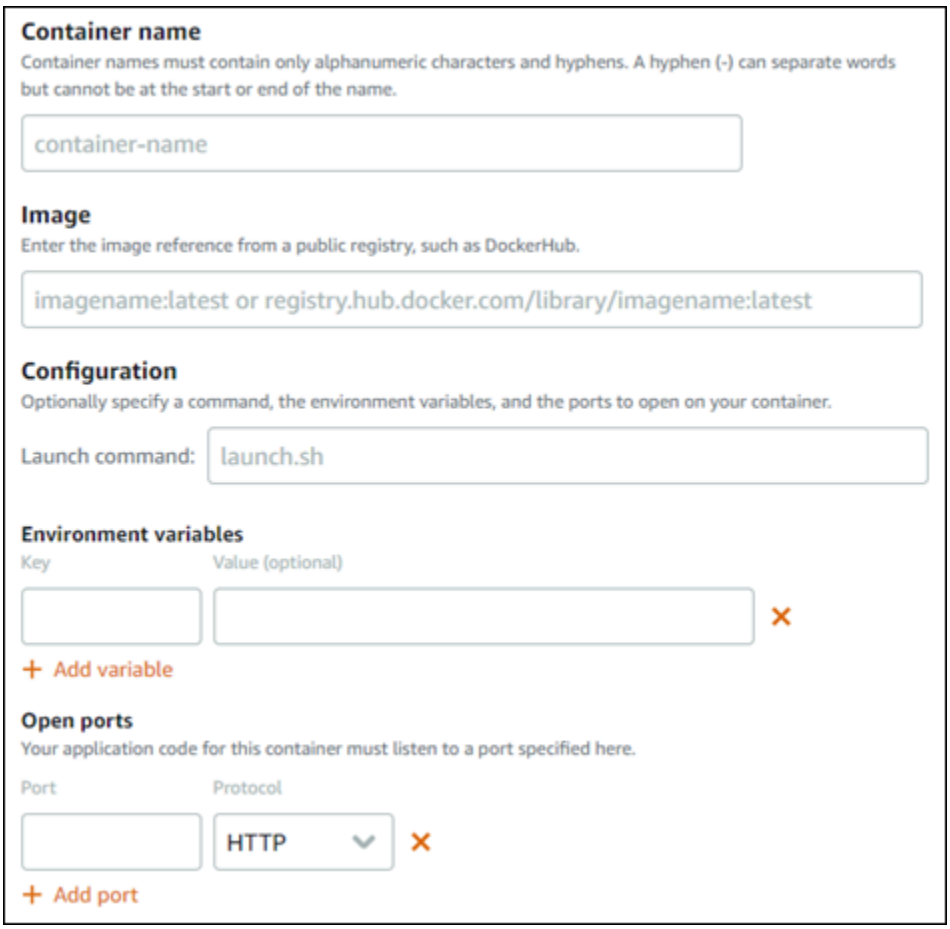
- Lightsail 계정에 컨테이너 서비스를 생성합니다. 자세한 내용은 Amazon Lightsail 컨테이너 서비스 생성을 참조하세요.
- 컨테이너 서비스에서 컨테이너를 시작할 때 사용할 컨테이너 이미지를 식별합니다.
  - Amazon ECR Public Gallery와 같은 퍼블릭 레지스트리에서 컨테이너 이미지를 찾습니다. 자세한 내용은 Amazon ECR 퍼블릭 사용 설명서의 Amazon ECR Public Gallery를 참조하세요.
  - 로컬 시스템에서 컨테이너 이미지를 생성하고 Lightsail 컨테이너 서비스로 푸시합니다. 자세한 내용은 다음 안내서를 참조하세요.
    - Amazon Lightsail 컨테이너 서비스용 컨테이너 이미지를 관리하기 위한 소프트웨어 설치
    - 컨테이너 서비스 이미지 생성
    - 컨테이너 이미지 푸시 및 관리

배포 파라미터

이 섹션에서는 배포의 컨테이너 항목 및 퍼블릭 엔드포인트에 대해 지정할 수 있는 파라미터를 설명합니다.

컨테이너 항목 파라미터

배포에 컨테이너 항목을 10개까지 추가할 수 있습니다. 각 컨테이너 항목에는 다음과 같은 파라미터를 지정할 수 있습니다.



- 컨테이너 이름(Container name) - 컨테이너 이름을 입력합니다. 배포 내의 모든 컨테이너에는 고유한 이름이 있어야 하며, 영숫자 및 하이픈만 포함되어야 합니다. 하이픈으로 단어를 구분할 수 있지만, 이름의 시작이나 끝에 사용할 수는 없습니다.
- 소스 이미지(Source image) - 컨테이너의 소스 컨테이너 이미지를 지정합니다. 퍼블릭 레지스트리 소스
  - Amazon ECR Public Gallery 또는 기타 퍼블릭 컨테이너 이미지 레지스트리와 같은 퍼블릭 레지스트리.

    Amazon ECR Public에 대한 자세한 내용은 Amazon ECR Public 사용 설명서의 Amazon Elastic Container Registry Public이란 무엇입니까?를 참조하세요.
  - 로컬 시스템에서 컨테이너 서비스로 푸시된 이미지. 저장된 이미지를 지정하려면 저장된 이미지 선택(Choose stored images)을 선택한 다음 사용할 이미지를 선택하면 됩니다.

    로컬 시스템에 컨테이너 이미지를 생성하는 경우 해당 이미지를 컨테이너 서비스에 푸시하여 배포를 생성할 때 사용할 수 있습니다. 자세한 내용은 Amazon Lightsail 컨테이너 서비스용 컨테이너

이미지 생성 및 Amazon Lightsail 컨테이너 서비스에서 컨테이너 이미지 푸시 및 관리를 참조하세요.

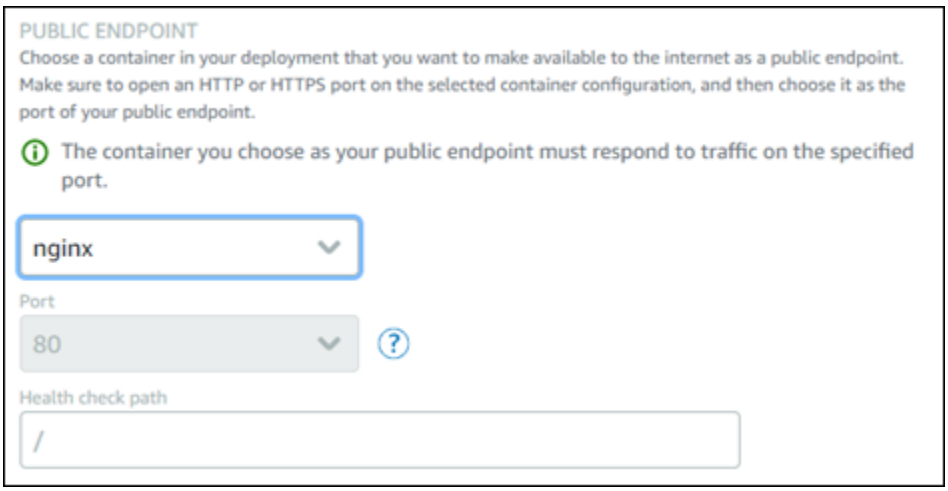
- 시작 명령(Launch command) - 컨테이너가 생성될 때 셸 스크립트 또는 bash 스크립트를 실행하는 실행 명령을 지정합니다. 실행 명령을 통해 소프트웨어 추가, 소프트웨어 업데이트와 같은 작업을 수행하거나 다른 방식으로 컨테이너를 구성할 수 있습니다.
- 환경 변수(Environment variables) - 컨테이너에서 실행되는 애플리케이션 또는 스크립트의 동적 구성을 제공하는 키 값 파라미터인 환경 변수를 지정합니다.
- 개방 포트(Open ports)- 컨테이너에서 열 포트와 프로토콜을 지정합니다. HTTP, HTTPS, TCP 및 UDP를 통해 포트를 열도록 지정할 수 있습니다. 컨테이너 서비스의 퍼블릭 엔드포인트로 사용할 컨테이너의 HTTP 또는 HTTPS 포트를 열어야 합니다. 자세한 내용은 이 가이드의 다음 섹션을 참조하세요.

퍼블릭 엔드포인트 파라미터

배포에서 컨테이너 서비스의 퍼블릭 엔드포인트로 사용할 컨테이너 항목을 지정할 수 있습니다. 퍼블릭 엔드포인트 컨테이너의 애플리케이션은 임의로 생성된 컨테이너 서비스의 기본 도메인을 통해 인터넷에서 공개적으로 액세스할 수 있습니다. 기본 도메인의 형식은 `https://<ServiceName>.<RandomGUID>.<AWSRegion>.cs.amazonlightsail.com`으로 지정됩니다. 여기서 *<ServiceName>*은 컨테이너 서비스의 이름이고, *<RandomGUID>*는 Lightsail 계정에 대해 AWS 리전에서 임의로 생성된 컨테이너 서비스의 전역 고유 식별자이며, *<AWSRegion>*은 컨테이너 서비스가 생성된 AWS 리전입니다. Lightsail 컨테이너 서비스의 퍼블릭 엔드포인트는 HTTPS만 지원하며, TCP 또는 UDP 트래픽은 지원하지 않습니다. 하나의 컨테이너만 서비스의 퍼블릭 엔드포인트가 될 수 있습니다. 따라서 나머지 컨테이너가 내부적으로 액세스할 수 있는 동안 애플리케이션의 프런트 엔드를 호스팅하는 컨테이너를 퍼블릭 엔드포인트로 선택해야 합니다.

> **Note**
>
> 컨테이너 서비스에서 사용자 지정 도메인 이름을 사용할 수 있습니다. 자세한 내용은 Amazon Lightsail 컨테이너 서비스용 사용자 지정 도메인 사용 설정 및 관리를 참조하세요.

배포의 퍼블릭 엔드포인트와 컨테이너 서비스에는 다음과 같은 파라미터를 지정할 수 있습니다.



- 엔드포인트 컨테이너(Endpoint container) - 배포에서 컨테이너 서비스의 퍼블릭 엔드포인트 역할을 할 컨테이너의 이름을 선택합니다. 배포에서 HTTP 또는 HTTPS 포트가 열려 있는 컨테이너만 드롭다운 메뉴에 나열됩니다.
- 포트(Port) - 퍼블릭 엔드포인트에 사용할 HTTP 또는 HTTPS 포트를 선택합니다. 선택한 컨테이너에 열려 있는 HTTP 및 HTTPS 포트만 드롭다운 메뉴에 나열됩니다. 선택한 컨테이너가 처음 시작될 때 HTTPS 연결을 지원하도록 구성되지 않은 경우 HTTP 포트를 선택합니다.

> **Note**
>
> HTTP 포트를 퍼블릭 엔드포인트 포트로 선택하더라도 컨테이너 서비스의 기본 도메인은 기본적으로 HTTPS를 사용합니다. 이는 컨테이너 서비스의 로드 밸런서가 기본적으로 HTTPS용으로 구성되어 있지만, HTTP를 사용하여 컨테이너와의 연결을 설정하기 때문입니다. 컨테이너 서비스의 로드 밸런서는 HTTP를 사용하여 컨테이너에 연결되지만, HTTPS를 사용하여 사용자에게 콘텐츠를 제공합니다.

- 상태 확인 경로(Health check path) - 선택한 퍼블릭 엔드포인트 컨테이너에서 컨테이너 서비스의 로드 밸런서 상태가 정상인지 정기적으로 검사할 경로를 지정합니다.
- 고급 상태 확인 설정— 선택한 퍼블릭 엔드포인트 컨테이너에 대해 다음과 같은 상태 확인 설정을 구성할 수 있습니다.
  - 상태 확인 제한 시간 초 – 응답을 기다릴 시간(초)입니다. 이 시간 동안 응답이 수신되지 않으면 상태 확인이 실패합니다. 2~60초를 지정할 수 있습니다.
  - 상태 확인 간격 초 - 컨테이너의 상태 확인 사이의 대략적인 간격(초)입니다. 5~300초를 지정할 수 있습니다.

- 상태 확인 성공 코드 - 컨테이너로부터 응답 성공을 확인할 때 사용하는 HTTP 코드입니다. 200에서 499 사이의 값을 지정할 수 있습니다. 여러 값(예: 200,202) 또는 값 범위(예: 200~299)를 지정할 수 있습니다.
- 상태 확인 정상 임계값 - 컨테이너를 정상 상태로 이동하기 전에 필요한 연속 상태 확인 성공 횟수입니다.
- 상태 확인 비정상 임계값 - 컨테이너를 비정상 상태로 이동하기 전에 필요한 연속 상태 확인 실패 횟수입니다.

프라이빗 도메인

또한, 모든 컨테이너 서비스에는 *<ServiceName>*.service.local로 형식이 지정된 프라이빗 도메인이 있으며, 여기서 *<ServiceName>*은 컨테이너 서비스의 이름입니다. 프라이빗 도메인을 사용하여 서비스와 동일한 AWS 리전에 있는 다른 Lightsail 리소스에서 컨테이너 서비스에 액세스할 수 있습니다. 프라이빗 도메인은 서비스 배포에서 퍼블릭 엔드포인트를 지정하지 않은 경우 컨테이너 서비스에 액세스할 수 있는 유일한 방법입니다. 퍼블릭 엔드포인트를 지정하지 않더라도 컨테이너 서비스에 대해 기본 도메인이 생성되지만, 해당 도메인을 찾아보려고 하면 `404 No Such Service` 오류 메시지가 표시됩니다.

컨테이너 서비스의 프라이빗 도메인을 사용하여 특정 컨테이너에 액세스하려면 연결 요청을 수락할 컨테이너의 개방 포트를 지정해야 합니다. 이렇게 하려면 요청의 도메인 형식을 *<ServiceName>*.service.local:*<PortNumber>*로 지정하면 됩니다. 여기서 *<ServiceName>*은 컨테이너 서비스의 이름이고 *<PortNumber>*는 연결하려는 컨테이너의 개방 포트입니다. 예를 들어, container-service-1이라는 컨테이너 서비스에 배포를 생성하고 포트 6379가 열려 있는 Redis 컨테이너를 지정하는 경우 요청 도메인 형식을 *container-service-1*.service.local:*6379*로 지정해야 합니다.

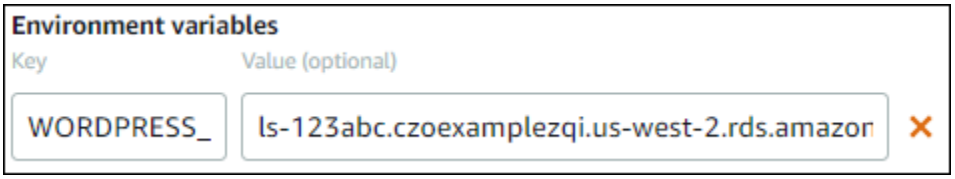
컨테이너 간 통신

환경 변수를 사용하면 동일한 컨테이너 서비스 내의 컨테이너 간, 다른 컨테이너 서비스 내의 컨테이너 간 또는 컨테이너와 다른 리소스 간(예: 컨테이너와 관리형 데이터베이스)에 통신을 열 수 있습니다.

동일한 컨테이너 서비스 내의 컨테이너 간에 통신을 열려면 다음 예제에 표시된 것과 같이 localhost를 참조하는 환경 변수를 컨테이너 배포에 추가합니다.

서로 다른 컨테이너 서비스에 있는 컨테이너 간에 통신을 열려면 다음 예제에 표시된 것과 같이 다른 컨테이너 서비스의 프라이빗 도메인(예: `container-service-1.service.local`)을 참조하는 환경 변수를 컨테이너 배포에 추가합니다.

컨테이너와 다른 리소스 간에 통신을 열려면 리소스의 퍼블릭 엔드포인트 URL을 참조하는 환경 변수를 컨테이너 배포에 추가합니다. 예를 들어, Lightsail 관리형 데이터베이스의 퍼블릭 엔드포인트는 일반적으로 `ls-123abc.czoexamplezqi.us-west-2.rds.amazonaws.com`입니다. 따라서 다음 예제에 표시된 것과 같이 환경 변수에서 이를 참조해야 합니다.



컨테이너 로그

배포의 모든 컨테이너는 로그를 생성합니다. 컨테이너 로그는 컨테이너 내부에서 실행되는 프로세스의 stdout 및 stderr 스트림을 제공합니다. 컨테이너의 로그에 주기적으로 액세스하여 작업을 진단할 수 있습니다. 자세한 내용은 Amazon Lightsail 컨테이너 서비스의 컨테이너 로그 확인을 참조하세요.

배포 버전

컨테이너 서비스에서 생성하는 모든 배포는 배포 버전으로 저장됩니다. 기존 배포의 파라미터를 수정하면 컨테이너가 서비스에 다시 배포되고 수정된 배포로 인해 새 배포 버전이 만들어집니다. 각 컨테이너 서비스에 대한 최신 배포 버전 50개가 저장됩니다. 50개의 배포 버전 중 하나를 사용하여 동일한 컨테이너 서비스에서 새 배포를 생성할 수 있습니다. 자세한 내용은 Amazon Lightsail 컨테이너 서비스의 배포 버전 확인 및 관리를 참조하세요.

배포 상태

배포는 생성된 후 다음 상태 중 하나가 될 수 있습니다.

- 활성화 중 - 배포가 활성화되는 중이며 컨테이너가 생성 중입니다.
- 활성 - 배포가 성공적으로 생성되었으며 현재 컨테이너 서비스에서 실행 중입니다.
- 비활성 - 이전에 성공적으로 생성한 배포가 더 이상 컨테이너에서 실행되지 않습니다.

- 실패 - 배포에 지정된 하나 이상의 컨테이너를 시작하지 못했기 때문에 배포에 실패했습니다.

배포 실패

배포에서 하나 이상의 컨테이너가 시작되지 않으면 배포가 실패합니다. 배포에 실패하고 컨테이너 서비스에서 이전 배포가 실행 중인 경우 컨테이너 서비스는 이전 배포를 활성 배포로 유지합니다. 이전 배포가 없으면 컨테이너 서비스가 현재 활성 배포가 없는 준비 상태로 변경됩니다.

실패한 배포의 컨테이너 로그를 보며 잘못된 부분을 진단하고 문제를 해결할 수 있습니다. 자세한 내용은 Amazon Lightsail 컨테이너 서비스의 컨테이너 로그 확인을 참조하세요.

현재 컨테이너 서비스 배포 확인

Lightsail 컨테이너 서비스의 현재 배포를 보려면 다음 절차를 완료하세요.

1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창의 이미지에서 컨테이너를 선택합니다.
3. 현재 배포를 보려는 컨테이너 서비스의 이름을 선택합니다.
4. 컨테이너 서비스 관리 페이지에서 배포(Deployments) 탭을 선택합니다.

   이 배포(Deployments) 페이지에는 현재 배포 및 배포 버전이 나열됩니다. 컨테이너 서비스에서 배포를 생성하지 않은 경우 페이지의 두 섹션은 모두 비어 있습니다.

컨테이너 서비스 배포 생성 또는 수정

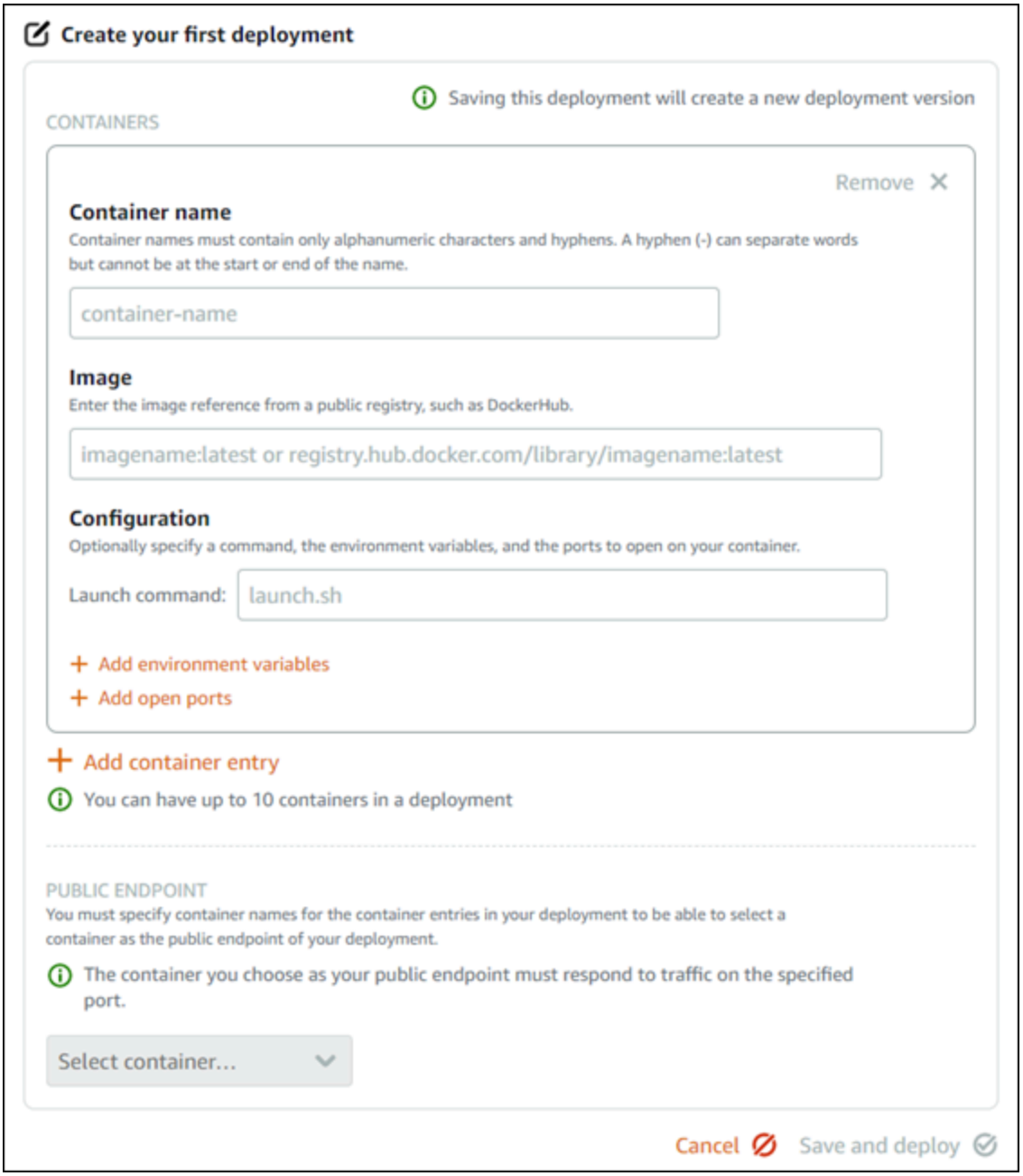
Lightsail 컨테이너 서비스에서 배포를 생성하거나 수정하려면 다음 절차를 완료하세요. 새 배포를 생성하든 기존 배포를 수정하든 컨테이너 서비스는 모든 배포를 새 배포 버전으로 저장합니다. 자세한 내용은 Amazon Lightsail 컨테이너 서비스의 배포 버전 확인 및 관리를 참조하세요.

1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창의 이미지에서 컨테이너를 선택합니다.
3. 컨테이너 서비스 배포를 생성하거나 수정할 컨테이너 서비스의 이름을 선택합니다.
4. 컨테이너 서비스 관리 페이지에서 배포(Deployments) 탭을 선택합니다.

   이 배포(Deployments) 페이지에는 현재 배포 및 배포 버전(있는 경우)이 나열됩니다.
5. 다음 옵션 중 하나를 선택하세요.

- 컨테이너 서비스에 기존 배포가 있는 경우 배포 수정(Modify your deployment)을 선택합니다.
- 컨테이너 서비스에 배포가 없는 경우 배포 생성(Create a deployment)을 선택합니다.

기존 배포 파라미터를 편집하거나 새 배포 파라미터를 입력할 수 있는 배포 양식이 열립니다.



6. 배포의 파라미터를 입력합니다. 지정할 수 있는 배포 파라미터에 대한 자세한 내용은 가이드 앞 부분에 나와 있는 배포 파라미터 섹션을 참조하세요.
7. 컨테이너 항목 추가(Add container entry)를 선택하여 둘 이상의 컨테이너 항목을 배포에 추가합니다. 배포에 컨테이너 항목을 10개까지 포함할 수 있습니다.

8. 퍼블릭 엔드포인트 컨테이너 서비스로 사용할 배포의 컨테이너 항목을 선택합니다. 여기에는 HTTP 또는 HTTPS 포트, 선택한 컨테이너 항목의 상태 확인 경로 및 고급 상태 확인 설정 지정이 포함됩니다. 자세한 내용은 이 가이드의 앞부분의 퍼블릭 엔드포인트 파라미터를 참조하세요.
9. 배포 파라미터를 입력했으면 저장 및 배포(Save and deploy)를 선택하여 컨테이너 서비스에 배포를 생성합니다.

   배포가 생성되는 동안 컨테이너 서비스 상태가 배포 중으로 변경됩니다. 잠시 후 배포 상태에 따라 컨테이너 서비스 상태가 다음 중 하나로 변경됩니다.

   - 배포에 성공하면 컨테이너 서비스 상태가 실행 중으로 변경되고 배포 상태가 활성으로 변경됩니다. 배포에서 퍼블릭 엔드포인트를 구성한 경우 퍼블릭 엔드포인트로 선택한 컨테이너를 컨테이너 서비스의 기본 도메인을 통해 사용할 수 있습니다.
   - 배포에 실패하고 컨테이너 서비스에서 이전 배포가 실행 중인 경우 컨테이너 서비스 상태가 실행 중으로 변경되고 컨테이너 서비스는 이전 배포를 활성 배포로 유지합니다. 이전 배포가 없으면 컨테이너 서비스 상태가 현재 활성 배포가 없는 준비로 변경됩니다. 실패한 배포의 컨테이너 로그를 보며 잘못된 부분을 진단하고 문제를 해결할 수 있습니다. 자세한 내용은 Amazon Lightsail 컨테이너 서비스의 컨테이너 로그 확인을 참조하세요.

주제

- Lightsail 컨테이너 서비스의 용량 크기 조정
- Lightsail 컨테이너 서비스의 배포 버전 확인 및 관리
- Lightsail 컨테이너 서비스 로그 분석

Lightsail 컨테이너 서비스의 용량 크기 조정

Amazon Lightsail 컨테이너 서비스의 용량은 규모와 성능으로 구성됩니다. 규모는 컨테이너 서비스의 컴퓨팅 노드 수를 지정하고, 성능은 서비스 내 각 노드의 메모리와 vCPU를 결정합니다. 가용성을 개선하고 용량을 늘리려면 서비스 성능을 좌우하는 노드 수에 따라 규모를 선택하면 됩니다.

이 가이드의 절차에 따라 프로비저닝이 부족해지면 언제든지 다운타임 없이 컨테이너 서비스의 성능 및 규모를 동적으로 늘리고, 과다 프로비저닝될 경우 축소할 수 있습니다. Lightsail에서는 현재 배포와 함께 용량 변경을 자동으로 관리합니다.

> **Note**
>
> 새 배포를 만들면 컨테이너 서비스의 기존 사용률 지표가 사라지고, 새로운 현재 배포에 대한 지표만 표시됩니다.

컨테이너 서비스에 대한 자세한 내용은 컨테이너 서비스를 참조하세요.

컨테이너 서비스의 용량 변경

Lightsail 컨테이너 서비스 용량을 변경하려면 다음 절차를 수행하세요.

1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창의 이미지에서 컨테이너를 선택합니다.
3. 용량을 변경할 컨테이너 서비스의 이름을 선택합니다.
4. 컨테이너 서비스 관리 페이지에서 용량(Capacity) 탭을 선택합니다.

   용량(Capacity) 페이지에는 컨테이너 서비스의 현재 성능, 규모 및 월별 요금이 나와 있습니다.
5. 용량 변경(Change capacity)을 선택하여 성능 및 규모를 변경합니다.
6. 표시되는 확인 프롬프트에서 예, 계속합니다(Yes, continue)를 선택하여 컨테이너 서비스 용량을 변경하면 현재 배포가 다시 배포될 것임을 확인합니다.
7. 컨테이너 서비스의 성능 및 규모를 새로 선택합니다.
8. 예, 적용합니다(Yes, apply)를 선택하여 컨테이너 서비스에 새 용량을 적용합니다.

   컨테이너 서비스의 상태가 업데이트 중(Updating)으로 변경됩니다. 잠시 후 서비스 상태가 사용(Enabled)으로 변경되며, 컨테이너가 새로운 용량으로 운영되기 시작합니다.

Lightsail 컨테이너 서비스의 배포 버전 확인 및 관리

Amazon Lightsail 컨테이너 서비스에서 생성하는 모든 배포는 배포 버전으로 저장됩니다. 기존 배포의 파라미터를 수정하면 컨테이너가 서비스에 다시 배포되고 수정된 배포로 인해 새 배포 버전이 만들어집니다. 각 컨테이너 서비스에 대한 최신 배포 버전 50개가 저장됩니다. 50개의 배포 버전 중 하나를 사용하여 동일한 컨테이너 서비스에서 새 배포를 생성할 수 있습니다. 이 가이드에서는 컨테이너 서비스의 배포 버전을 확인하고 관리하는 방법을 안내합니다.

컨테이너 서비스에 대한 자세한 내용은 컨테이너 서비스를 참조하세요.

배포 버전 상태

각 배포 버전은 생성된 후 다음 상태 중 하나가 될 수 있습니다.

- 배포 중(활성화) - 배포가 시작되는 중입니다.
- 활성 - 배포가 성공적으로 생성되었으며 현재 컨테이너 서비스에서 실행 중입니다. 한 번에 하나의 활성 배포만 컨테이너 서비스에 존재할 수 있습니다.
- 비활성 - 이전에 성공적으로 생성한 배포가 더 이상 컨테이너에서 실행되지 않습니다.
- 실패 - 배포에 지정된 하나 이상의 컨테이너를 시작하지 못했기 때문에 배포에 실패했습니다.

사전 조건

시작하기 전에 Lightsail 컨테이너 서비스를 생성해야 합니다. 자세한 내용은 컨테이너 서비스 생성을 참조하세요.

컨테이너를 구성하고 시작하는 컨테이너 서비스에서 배포를 생성해야 합니다. 자세한 내용은 Amazon Lightsail 컨테이너 서비스용 배포 생성 및 관리를 참조하세요.

컨테이너 서비스의 배포 버전 확인

Lightsail 컨테이너 서비스의 배포 버전을 보려면 다음 절차를 완료하세요.

1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창의 이미지에서 컨테이너를 선택합니다.
3. 배포 버전을 보려는 컨테이너 서비스의 이름을 선택합니다.
4. 컨테이너 서비스 관리 페이지에서 배포(Deployments) 탭을 선택합니다.

   이 배포(Deployments) 페이지에는 현재 배포 및 배포 버전(있는 경우)이 나열됩니다.
5. 컨테이너 서비스의 배포 버전은 페이지의 배포 버전(Deployment versions) 섹션 아래에 나열됩니다.

   각 배포에는 생성된 날짜, 상태 및 작업 메뉴가 있습니다.
6. 배포 버전의 작업 메뉴를 통해 다음 옵션 중 하나를 선택합니다.

   - 새 배포 생성(Create new deployment) - 선택한 배포 버전에서 새 배포를 생성하려면 이 옵션을 선택합니다. 배포를 생성하는 방법에 대한 자세한 내용은 컨테이너 서비스 배포 생성 또는 수정을 참조하세요.

> **Note**
>
> 실패 상태인 버전에서 새 배포를 생성하도록 선택한 경우 배포를 생성하기 전에 실패의 원인을 수정해야 합니다. 그렇지 않으면 배포가 다시 실패합니다.

- 세부 정보 보기(View details) - 선택한 배포 버전의 컨테이너 항목 및 퍼블릭 엔드포인트 파라미터를 보려면 이 옵션을 선택합니다. 실패한 배포를 진단해야 할 경우 배포의 컨테이너 로그를 볼 수도 있습니다. 자세한 내용은 컨테이너 서비스 지표 확인을 참조하세요.

Lightsail 컨테이너 서비스 로그 분석

Amazon Lightsail 컨테이너 서비스 배포에 포함된 모든 컨테이너는 로그를 생성합니다. 컨테이너 로그는 컨테이너 내부에서 실행되는 프로세스의 stdout 및 stderr 스트림을 제공합니다. 컨테이너의 로그에 주기적으로 액세스하여 작업을 진단할 수 있습니다. 최근 3일 동안의 로그 항목이 저장되며, 그 후 가장 오래된 로그 항목이 최신 항목으로 대체됩니다.

컨테이너 로그 필터링

컨테이너 로그에는 하루에 수백 개의 항목이 포함될 수 있습니다. 로그 창에 표시되는 항목 수를 줄이고 원하는 항목을 쉽게 찾으려면 필터링 옵션을 사용하면 됩니다. 시작 및 종료 날짜(현지 시간)와 특정 조건별로 컨테이너 로그를 필터링할 수 있습니다. 조건별로 필터링할 때 지정한 조건에 대한 로그 항목을 포함하거나 제외하도록 선택할 수 있습니다.

include 또는 exclude 필터 조건은 대소문자를 구분하는 정확한 일치 항목을 찾습니다. 예를 들어, 메시지에 `HTTP`가 있는 로그 이벤트만 포함하도록 지정할 경우 메시지에 `HTTP`가 포함된 모든 로그 이벤트가 표시됩니다. 단, 메시지에 `http`가 포함된 로그 이벤트는 표시되지 않습니다. `Error`를 제외하도록 지정하면 메시지에 `Error`가 포함되지 않은 모든 로그 이벤트와 함께 `ERROR`가 포함된 로그 이벤트도 표시됩니다.

사전 조건

시작하기 전에 Lightsail 컨테이너 서비스를 생성해야 합니다. 자세한 내용은 Amazon Lightsail 컨테이너 서비스 생성을 참조하세요.

컨테이너를 구성하고 시작하는 컨테이너 서비스에서 배포를 생성해야 합니다. 자세한 내용은 Amazon Lightsail 컨테이너 서비스용 배포 생성 및 관리를 참조하세요.

컨테이너 로그 확인

Lightsail 컨테이너 서비스의 컨테이너 로그를 보려면 다음 절차를 완료하세요.

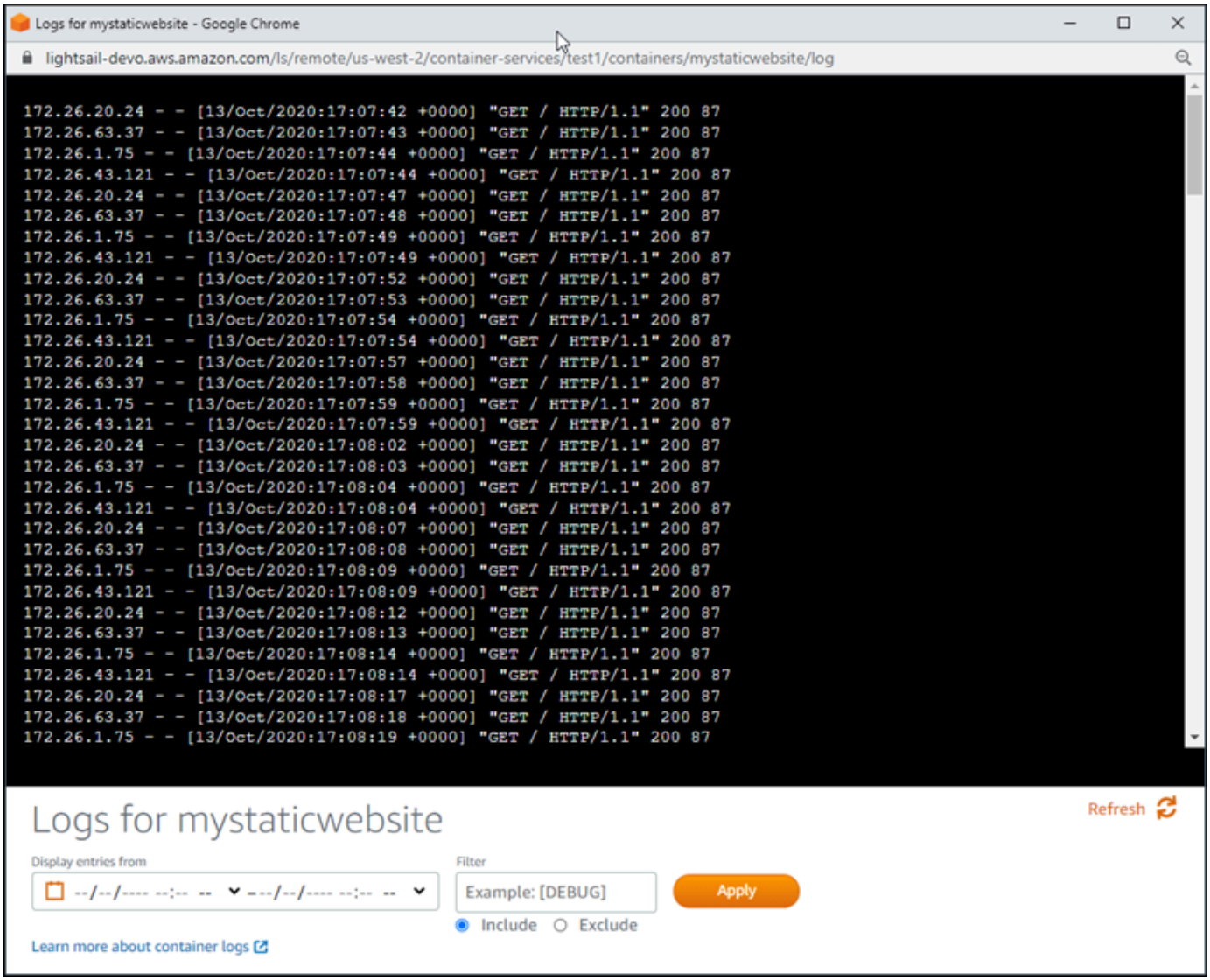
1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창의 이미지에서 컨테이너를 선택합니다.
3. 컨테이너 로그를 보려는 컨테이너 서비스의 이름을 선택합니다.
4. 컨테이너 서비스 관리 페이지에서 배포(Deployments) 탭을 선택합니다.

   이 배포(Deployments) 페이지에는 현재 배포 및 배포 버전(있는 경우)이 나열됩니다.
5. 다음 옵션 중 하나를 선택하여 컨테이너 로그를 확인합니다.

   - 현재 배포의 컨테이너 로그에 액세스하려면 페이지의 현재 배포(Current deployment) 섹션에서 컨테이너 항목의 로그 열기(Open log)를 선택합니다.
   - 이전 배포의 컨테이너 로그에 액세스하려면 페이지의 배포 버전(Deployment versions) 섹션에서 이전 배포의 작업 메뉴 아이콘(⋮)을 선택한 다음, 세부 정보 표시(Show details)를 선택합니다. 버전 세부 정보(Version details) 페이지가 표시되면 나열된 컨테이너 항목에 대해 로그 열기(Open log)를 선택합니다.

   새 브라우저 창에 컨테이너 로그가 열립니다. 아래로 스크롤하여 더 많은 로그 항목을 보고 페이지를 새로 고쳐 최신 항목 집합을 로드할 수 있습니다. 페이지 아래쪽에 필터링 옵션이 표시됩니다.

   > **Note**
   >
   > 로그 항목은 협정 세계시(UTC)를 기준으로 오름차순으로 표시됩니다. 즉, 가장 오래된 로그 항목이 맨 위에 나타나며 새로운 로그 항목을 보려면 아래로 스크롤해야 합니다.



Lightsail의 사용자 지정 도메인을 사용하여 보안 웹 액세스 활성화

등록된 도메인 이름을 서비스와 함께 사용할 수 있도록 Amazon Lightsail 컨테이너 서비스에 대한 사용자 지정 도메인을 활성화합니다. 사용자 지정 도메인을 사용하도록 설정하기 전에 컨테이너 서비스는 서비스를 처음 생성할 때 서비스에 연결된 기본 도메인(예: `containerservicename.123456abcdef.us-west-2.cs.amazonlightsail.com`)의 트래픽만 수락합니다. 사용자 정의 도메인을 사용하는 경우 컨테이너 서비스와 함께 사용할 도메인에 대해 생성한 Lightsail SSL/TLS 인증서를 선택한 다음, 해당 인증서에서 사용할 도메인을 선택합니다. 사용자 지정 도메인을 사용하도록 설정하면 컨테이너 서비스가 선택한 인증서와 연결된 모든 도메인의 트래픽을 수락합니다.

> **Important**
>
> Lightsail 컨테이너 서비스를 배포의 오리진으로 선택하는 경우, Lightsail은 배포의 기본 도메인 이름을 컨테이너 서비스에 사용자 지정 도메인으로 자동으로 추가합니다. 이렇게 하면 트래픽을 배포와 컨테이너 서비스 간에 라우팅할 수 있습니다. 그러나 컨테이너 서비스에 배포의 기본 도메인 이름을 수동으로 추가해야 하는 경우도 있습니다. 자세한 내용은 배포의 기본 도메인을 컨테이너 서비스에 추가를 참조하세요.

목차

- 컨테이너 서비스 사용자 지정 도메인 제한
- 사전 조건
- 컨테이너 서비스용 사용자 지정 도메인 확인
- 컨테이너 서비스용 사용자 지정 도메인 사용
- 컨테이너 서비스용 사용자 지정 도메인 사용 중지

컨테이너 서비스 사용자 지정 도메인 제한

컨테이너 서비스 사용자 지정 도메인에는 다음과 같은 제한이 적용됩니다.

- 각 Lightsail 컨테이너 서비스에서 최대 4개의 사용자 지정 도메인을 사용할 수 있으며, 2개 이상의 서비스에서 동일한 도메인을 사용할 수 없습니다.
- Lightsail DNS 영역을 사용하여 도메인의 DNS를 관리하는 경우 도메인의 Apex(예: `example.com`)와 하위 도메인(예: `www.example.com`)의 트래픽을 컨테이너 서비스로 라우팅할 수 있습니다.

사전 조건

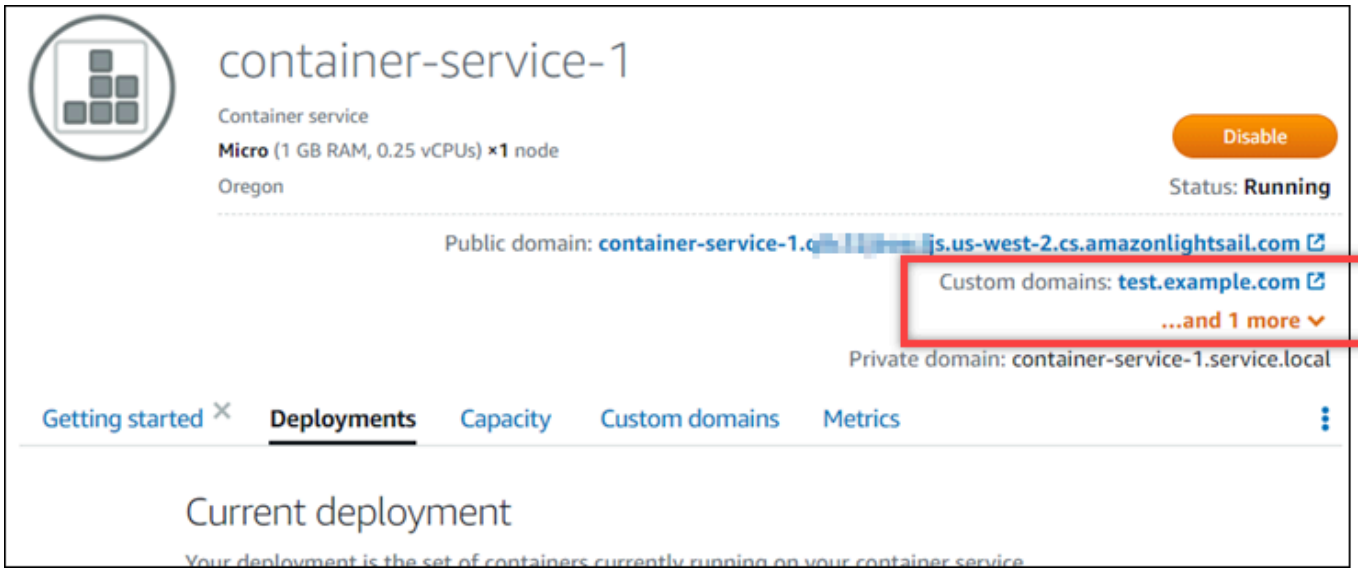
시작하기 전에 Lightsail 컨테이너 서비스를 생성해야 합니다. 자세한 내용은 Amazon Lightsail 컨테이너 서비스 생성을 참조하세요.

컨테이너 서비스에 대한 SSL/TLS 인증서를 생성하고 검증해야 합니다. 자세한 내용은 컨테이너 서비스 SSL/TLS 인증서 생성 및 컨테이너 서비스 SSL/TLS 인증서 검증을 참조하세요.

컨테이너 서비스용 사용자 지정 도메인 확인

현재 컨테이너 서비스에서 사용하는 사용자 지정 도메인을 확인하려면 다음 절차를 완료하세요.

1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창의 이미지에서 컨테이너를 선택합니다.
3. 사용 중인 사용자 지정 도메인을 확인하려는 컨테이너 서비스의 이름을 선택합니다.
4. 다음 예와 같이 컨테이너 서비스 관리 페이지의 머리글에서 사용자 지정 도메인 값을 찾습니다. 이 도메인 값이 컨테이너 서비스에서 현재 사용 중인 사용자 지정 도메인입니다.



5. 컨테이너 서비스 관리 페이지에서 사용자 지정 도메인(Custom domains) 탭을 선택합니다.

   연결된 각 인증서 아래에서 사용 중인 사용자 지정 도메인이 페이지의 Custom domain SSL/TLS certificates(사용자 지정 도메인 SSL/TLS 인증서) 섹션에 나열됩니다. 현재 컨테이너 서비스에 연결된 인증서는 Attached certificates(연결된 인증서) 섹션에 나열됩니다.

컨테이너 서비스용 사용자 지정 도메인 사용

인증서를 서비스에 연결하여 Lightsail 컨테이너 서비스에서 사용자 지정 도메인을 사용하려면 다음 절차를 완료하세요.

1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창의 이미지에서 컨테이너를 선택합니다.
3. 사용자 지정 도메인을 사용하려는 컨테이너 서비스의 이름을 선택합니다.
4. 컨테이너 서비스 관리 페이지에서 사용자 지정 도메인(Custom domains) 탭을 선택합니다.

   사용자 지정 도메인(Custom domains) 페이지에 현재 컨테이너 서비스에 연결된 SSL/TLS 인증서가 표시됩니다(있는 경우).

5. 인증서 연결(Attach certificate)을 선택합니다.

   인증서가 없는 경우 먼저 도메인에 대한 SSL/TLS 인증서를 생성하고 검증한 후 컨테이너 서비스에 인증서를 연결해야 합니다. 자세한 내용은 컨테이너 서비스 SSL/TLS 인증서 생성을 참조하세요.

6. 드롭다운 메뉴가 표시되면 컨테이너 서비스와 함께 사용할 도메인에 대해 유효한 인증서를 선택합니다.
7. 인증서 정보가 올바른지 확인한 다음 Attach(연결)를 선택합니다.
8. 컨테이너 서비스의 Status(상태)가 Updating(업데이트 중)으로 변경됩니다. 상태가 Ready(준비됨)로 변경된 후 인증서의 Custom domains(사용자 지정 도메인) 섹션에 표시됩니다.
9. Add domain assignment(도메인 할당 추가)를 선택하여 도메인을 컨테이너 서비스에 연결합니다.
10. 인증서와 DNS 정보가 올바른지 확인한 다음 Add assignment(할당 추가)를 선택합니다. 잠시 후 컨테이너 서비스에서 선택한 도메인의 트래픽을 수락하기 시작합니다.
11. 도메인 할당을 추가한 후 새 브라우저 창을 열고 컨테이너 서비스에서 사용하는 사용자 지정 도메인을 찾습니다. 컨테이너 서비스에서 실행 중인 애플리케이션(있는 경우)이 로드되어야 합니다.

컨테이너 서비스용 사용자 지정 도메인 사용 중지

인증서를 서비스에서 분리하거나 이전에 선택한 도메인을 선택 취소하여 Lightsail 컨테이너 서비스에서 사용자 지정 도메인을 사용하지 않으려면 다음 절차를 완료하세요.

1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창의 이미지에서 컨테이너를 선택합니다.
3. 사용자 지정 도메인을 사용 중지하려는 컨테이너 서비스의 이름을 선택합니다.
4. 컨테이너 서비스 관리 페이지에서 사용자 지정 도메인(Custom domains) 탭을 선택합니다.

   사용자 지정 도메인(Custom domains) 페이지에 현재 컨테이너 서비스에 연결된 SSL/TLS 인증서가 표시됩니다(있는 경우).

5. 다음 옵션 중 하나를 선택하세요.

   1. 이전에 선택한 도메인을 선택 취소하거나 컨테이너 서비스와 연결된 도메인을 더 선택하려면 Configure container service domains(컨테이너 서비스 도메인 구성)를 선택합니다.
   2. 분리를 선택하면 컨테이너 서비스에서 인증서가 분리되고, 컨테이너에서 연결된 모든 도메인이 제거됩니다.

> **Important**
>
> 도메인의 DNS 레코드를 수정하지 않은 경우 수정하여 트래픽 경로가 더 이상 컨테이너 서비스로 라우팅되지 않고, 대신 다른 리소스로 라우팅되도록 설정합니다.

주제

- 도메인 트래픽을 Lightsail 컨테이너 서비스로 라우팅
- Route 53를 사용하여 도메인의 트래픽을 Lightsail 컨테이너 서비스로 라우팅

도메인 트래픽을 Lightsail 컨테이너 서비스로 라우팅

서비스에 사용자 지정 도메인을 활성화한 후에는 등록된 도메인 이름이 Amazon Lightsail 컨테이너 서비스를 연결해야 합니다. 컨테이너 서비스와 함께 사용하는 인증서에 지정된 각 도메인의 DNS 영역에 별칭 레코드를 추가하면 됩니다. 추가하는 모든 레코드는 컨테이너 서비스의 기본 도메인(예: `https://<ServiceName>.<RandomGUID>.<AWSRegion>.cs.amazonlightsail.com`)을 가리켜야 합니다.

이 설명서에서는 Lightsail DNS 영역을 사용하여 도메인이 컨테이너 서비스를 가리키도록 하는 절차를 안내합니다. Lightsail DNS 영역에 대한 자세한 내용은 DNS in Amazon Lightsail을 참조하세요.

컨테이너 서비스에 대한 자세한 내용은 컨테이너 서비스를 참조하세요.

> **Note**
>
> Route 53를 도메인의 DNS를 호스팅하는 데 사용할 경우 Route 53에서 도메인의 호스팅 영역에 별칭 레코드를 추가해야 합니다. 자세한 내용은 Route 53의 도메인에 대한 트래픽을 Amazon Lightsail 컨테이너 서비스로 라우팅을 참조하세요.

사전 조건

시작하기 전에 Lightsail 컨테이너 서비스에 사용자 지정 도메인을 활성화해야 합니다. 자세한 내용은 Amazon Lightsail 컨테이너 서비스용 사용자 지정 도메인 사용 설정 및 관리를 참조하세요.

컨테이너 서비스의 기본 도메인 가져오기

별칭 레코드를 도메인의 DNS에 추가할 때 지정하는 컨테이너 서비스의 기본 도메인 이름을 가져오려면 다음 절차를 완료하세요.

1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창의 이미지에서 컨테이너를 선택합니다.
3. 기본 도메인 이름을 가져올 컨테이너 서비스의 이름을 선택합니다.
4. 컨테이너 서비스 관리 페이지의 머리글 섹션에 있는 기본 도메인 이름을 기록합니다. 컨테이너 서비스 기본 도메인 이름은 *<ServiceName>*.*<RandomGUID>*.*<AWSRegion>*.cs.amazonlightsail.com과 유사합니다.

   이 값은 도메인의 DNS에서 정규 이름(CNAME) 레코드의 일부로 추가해야 합니다. 나중에 참조할 수 있도록 이 값을 복사하여 텍스트 파일에 붙여넣는 것이 좋습니다. 자세한 내용은 이 가이드의 도메인의 DNS 영역에 CNAME 레코드 추가 섹션을 참조하세요.

도메인의 DNS 영역에 레코드 추가

주소(IPv4의 경우 A 또는 IPv6의 경우 AAAA) 레코드 또는 표준(CNAME) 레코드를 도메인의 DNS 영역에 추가하려면 다음 절차를 완료하세요.

1. 왼쪽 탐색 창에서 도메인 및 DNS를 선택합니다.
2. 페이지의 DNS 영역(DNS zones) 섹션에서 도메인의 트래픽을 컨테이너 서비스로 연결할 레코드를 추가하려는 도메인 이름을 선택합니다.
3. DNS records(DNS 레코드) 탭을 선택합니다.
4. DNS 영역의 현재 상태에 따라 다음 단계 중 하나를 완료합니다.

   - A, AAAA, 또는 CNAME 레코드를 추가하지 않은 경우 레코드 추가(Add record)를 선택합니다.
   - 이전에 A, AAAA 또는 CNAME 레코드를 추가한 경우 페이지에 나열된 기존 A, AAAA 또는 CNAME 레코드 옆에 있는 편집 아이콘을 선택한 다음, 이 절차의 5단계로 건너뜁니다.
5. 레코드 유형 드롭다운 메뉴에서 레코드, AAAA 레코드 또는 CNAME 레코드를 선택합니다.

   - A 레코드를 추가하여 IPv4 네트워크에서 도메인의 정점(예: `example.com`) 또는 하위 도메인(예: `www.example.com`)을 컨테이너 서비스에 매핑합니다.
   - AAAA 레코드를 추가하여 IPv6 네트워크에서 도메인의 정점(예: `example.com`) 또는 하위 도메인(예: `www.example.com`)을 컨테이너 서비스에 매핑합니다.

- CNAME 레코드를 추가하여 컨테이너 서비스의 퍼블릭 도메인(기본 DNS)에 하위 도메인(예:`www.example.com`)을 매핑합니다.

6. Record name(레코드 이름) 텍스트 상자에 다음 옵션 중 하나를 입력합니다.

   - A 레코드 또는 AAAA 레코드의 경우 @를 입력하여 도메인의 정점(예: `example.com`)에 대한 트래픽을 컨테이너 서비스로 라우팅하거나 하위 도메인(예: `www`)을 입력하여 하위 도메인(예: `www.example.com`)의 트래픽을 컨테이너 서비스로 라우팅합니다.
   - CNAME 레코드의 경우 하위 도메인(예: `www`)을 입력하여 하위 도메인(예: `www.example.com`)의 트래픽을 컨테이너 서비스로 라우팅합니다.

7. 추가하는 레코드에 따라 다음 단계 중 하나를 완료합니다.

   - A 레코드 또는 AAAA 레코드의 경우 확인(Resolves to) 텍스트 상자에서 컨테이너 서비스 이름을 선택합니다.
   - CNAME 레코드의 경우 다음으로 매핑(Maps to) 텍스트 상자에 컨테이너 서비스의 기본 도메인 이름을 입력합니다.

8. DNS 영역에 레코드를 저장하려면 저장 아이콘을 선택합니다.

   이 단계를 반복하여 컨테이너 서비스와 함께 사용 중인 인증서의 도메인에 대한 별도의 DNS 레코드를 추가합니다. 인터넷의 DNS를 통해 변경 사항이 전파될 때까지 기다립니다. 몇 분 후 도메인이 컨테이너 서비스를 가리키고 있는지 확인합니다.

Route 53를 사용하여 도메인의 트래픽을 Lightsail 컨테이너 서비스로 라우팅

`example.com`와 같이 등록된 도메인에 대한 트래픽을 Amazon Lightsail 컨테이너 서비스에서 실행되는 애플리케이션으로 라우팅할 수 있습니다. Lightsail 컨테이너 서비스의 기본 도메인을 가리키는 도메인의 호스팅 영역에 별칭 레코드를 추가하면 됩니다.

이 자습서에서는 Lightsail 컨테이너 서비스에 대한 별칭 레코드를 Route 53의 호스팅 영역에 추가하는 방법을 보여줍니다. ()를 사용해야만이 작업을 수행할 수 있습니다 AWS Command Line Interface AWS CLI. Route 53 콘솔을 사용하여 수행할 수 없습니다.

> **Note**
>
> Lightsail를 도메인의 DNS를 호스팅하는 데 사용할 경우 Lightsail에서 도메인의 DNS 영역에 별칭 레코드를 추가해야 합니다. 자세한 내용은 Amazon Lightsail에서 도메인의 트래픽을 Lightsail 컨테이너 서비스로 라우팅을 참조하세요.

목차

- 1단계: 필수 구성 요소 완성
- 2단계: Lightsail 컨테이너 서비스에 대한 호스팅 영역 ID 가져오기
- 3단계: 레코드 세트 JSON 파일 생성
- 4단계: Route 53의 도메인의 호스팅 영역에 레코드 추가

1단계: 필수 구성 요소 완성

아직 수행하지 않은 경우 다음 사전 조건을 완료합니다.

- Route 53에 도메인 이름을 등록하거나 Route 53를 등록된(기존) 도메인 이름에 대한 DNS 서비스로 만듭니다. 자세한 내용은 Amazon Route 53 개발자 안내서의 Amazon Route 53를 사용하여 도메인 이름 등록하기 또는 Amazon Route 53를 기존 도메인에 대한 DNS 서비스로 만들기를 참조하세요.
- Lightsail 컨테이너 서비스에 애플리케이션을 배포합니다. 자세한 내용은 컨테이너 서비스 배포 생성 및 관리를 참조하세요.
- Lightsail 컨테이너 서비스에 등록된 도메인 이름을 활성화합니다. 자세한 내용은 사용자 지정 도메인 활성화 및 관리를 참조하세요.
- 계정으로 AWS CLI 를 구성합니다. 자세한 내용은 에서 작동 AWS CLI 하도록 구성을 Lightsail참조하세요.

2단계: Lightsail 컨테이너 서비스에 대한 호스팅 영역 ID 가져오기

Route 53의 호스팅 영역에 별칭 레코드를 추가할 때 Lightsail 컨테이너 서비스에 대한 호스팅 영역 ID를 지정해야 합니다. 예를 들어 Lightsail 컨테이너 서비스가 미국 서부(오레곤)(us-west-2)에 있는 AWS 리전경우 Route 53의 호스팅 영역에 Lightsail 컨테이너 서비스의 별칭 레코드를 추가할 `Z0959753D43BBB908BAV` 때 호스팅 영역 ID를 지정해야 합니다.

다음은 Lightsail 컨테이너 서비스를 생성할 수 있는 각 AWS 리전의 호스팅 영역 ID입니다.

EU(런던)(eu-west-2): Z0624918ZXDYQZLOXA66

미국 동부(버지니아 북부)(us-east-1): Z06246771KYU0IRHI74W4

아시아 태평양(싱가포르)(ap-southeast-1): Z0625921354DRJH4EY9V0

EU(아일랜드)(eu-west-1): Z0624732FELAMMKW3Y21

아시아 태평양(도쿄)(ap-northeast-1): Z0626125UAU4JWQ9JSKN

아시아 태평양(서울)(ap-northeast-2): Z06260262XZM84B2WPLHH

아시아 태평양(자카르타)(ap-southeast-3): Z03072883T5HFTY4T7CDL

아시아 태평양(뭄바이)(ap-south-1): Z10460781IQMISS0I0VVY

아시아 태평양(시드니)(ap-southeast-2): Z09597943PQQZATPFE96E

캐나다(중부)(ca-central-1): Z10450993RIRIJJUUMA5W

유럽(프랑크푸르트)(eu-central-1): Z06137433FV04OY4EC6L0

유럽(스톡홀름)(eu-north-1): Z016970523TDG2TZMUXKK

유럽(파리)(eu-west-3): Z09594631DSW2QUR7CFGO

미국 동부(오하이오)(us-east-2): Z10362273VJ548563IY84

미국 서부(오레곤)(us-west-2): Z0959753D43BBB908BAV

3단계: 레코드 세트 JSON 파일 생성

를 사용하여 Route 53에서 도메인의 호스팅 영역에 DNS 레코드를 추가할 때는 레코드에 대한 구성 파라미터 세트를 지정 AWS CLI해야 합니다. 가장 쉬운 방법은 모든 파라미터가 포함된 JSON(.json) 파일을 생성한 다음 AWS CLI 요청에서 JSON 파일을 참조하는 것입니다.

별칭 레코드에 대한 레코드 세트 파라미터가 있는 JSON 파일을 생성하려면 다음 절차를 완료합니다.

1. 텍스트 편집기(예: Windows의 경우 메모장 또는 Linux의 경우 Nano)를 엽니다.
2. 다음 텍스트를 복사하여 텍스트 편집기에 붙여 넣습니다.

```
{
  "Comment": "Comment",
  "Changes": [
    {
      "Action": "CREATE",
      "ResourceRecordSet": {
        "Name": "Domain.",
        "Type": "A",
        "AliasTarget": {
          "HostedZoneId": "LightsailContainerServiceHostedZoneID",
          "DNSName": " LightsailContainerServiceAddress.",
          "EvaluateTargetHealth": true
        }
```

```
      }
    }
  ]
}
```

파일에서 아래 예시 텍스트를 사용자의 텍스트로 대체합니다.

- 레코드 세트에 대한 개인 메모 또는 의견을 *##*로 남깁니다.
- Lightsail 컨테이너 서비스와 함께 사용하려는 등록된 도메인 이름이 있는 *###*(예: `example.com` 또는 `www.example.com`). Lightsail 컨테이너 서비스와 함께 도메인의 루트를 사용하려면 도메인의 하위 도메인 공간에 `@` 기호를 지정해야 합니다(예: `@.example.com`).
- *#### #### ### AWS ### ### ## ID# ## LightsailContainerServiceHostedZoneID*입니다. Lightsail 자세한 내용은 이 가이드 앞부분의 2단계: Lightsail 컨테이너 서비스에 대한 호스팅 영역 ID 가져오기를 참조하세요.
- *#### #### ### ### ### ## LightsailContainerServiceAddress*. Lightsail Lightsail 콘솔에 로그인하고 컨테이너 서비스를 탐색한 다음 컨테이너 서비스 관리 페이지의 헤더 섹션에 나열된 퍼블릭 도메인(예: `container-service-1.q8cexampleljs.us-west-2.cs.amazonlightsail.com`)을 복사하여 이를 가져올 수 있습니다.

예제:

```
{
  "Comment": "Alias record for Lightsail container service",
  "Changes": [
    {
      "Action": "CREATE",
      "ResourceRecordSet": {
        "Name": "@.example.com.",
        "Type": "A",
        "AliasTarget": {
          "HostedZoneId": "Z0959753D43BBB908BAV",
          "DNSName": "container-service-1.q8cexampleljs.us-
west-2.cs.amazonlightsail.com.",
          "EvaluateTargetHealth": true
        }
      }
    }
  ]
}
```

3. 파일을 change-resource-record-sets.json로 로컬 디렉터리에 저장합니다.

4단계: Route 53의 도메인의 호스팅 영역에 레코드 추가

AWS CLI를 사용하여 Route 53의 도메인 호스팅 영역에 레코드를 추가하려면 다음 절차를 완료하세요. change-resource-record-sets 명령을 사용하여 이 작업을 수행합니다. 자세한 내용은 AWS CLI 명령 레퍼런스의 change-resource-record-sets를 참조하세요.

> **Note**
>
> 이 절차를 계속하기 전에 AWS CLI 및 Lightsail Route 53에 대해를 설치하고 구성해야 합니다. 자세한 내용은 에서 작동 AWS CLI 하도록 구성을 Lightsail참조하세요.

1. 명령 프롬프트 또는 터미널 창을 엽니다.
2. 다음 명령을 입력하여 Route 53에서 도메인의 호스팅 영역에 레코드를 추가합니다.

```
aws route53 change-resource-record-sets --hosted-zone-id HostedZoneID --change-
batch PathToJsonFile
```

명령에서 아래 예 텍스트를 사용자의 값으로 대체합니다.

- Route 53에 등록된 도메인의 호스팅 영역 ID가 있는 *HostedZoneID*입니다. list-hosted-zones 명령을 사용하여 Route 53 계정의 호스팅 영역에 대한 ID 목록을 가져옵니다.
- 레코드 파라미터가 포함된 .json 파일의 컴퓨터에 있는 로컬 디렉토리 폴더 경로가 있는 *PathToJsonFile*입니다. 자세한 내용은 이 가이드 앞부분의 3단계: 레코드 집합 JSON 파일 만들기 섹션을 참조하세요.

예시:

Linux 또는 Unix 컴퓨터:

```
aws route53 change-resource-record-sets --hosted-zone-id Z123456789ABCDEFGHIJ --
change-batch home/user/awscli/route53/change-resource-record-sets.json
```

Windows 컴퓨터:

```
aws route53 change-resource-record-sets --hosted-zone-id Z123456789ABCDEFGHIJ --
change-batch file://C:\awscli\route53\change-resource-record-sets.json
```

다음 예와 비슷한 결과가 나타나야 합니다.

```
H:\>aws route53 change-resource-record-sets --hosted-zone-id Z123456789ABCDEFGHIJ
--change-batch file://C:\awscli\route53\change-resource-record-sets.json
{
    "ChangeInfo": {
        "Id": "/change/C05953EXAMPLEZ4V4LOAC",
        "Status": "PENDING",
        "SubmittedAt": "2021-08-11T20:58:30.960000+00:00",
        "Comment": "Alias record for Lightsail container service"
    }
}
```

인터넷의 DNS를 통해 변경 사항이 전파될 때까지 기다립니다. 몇 시간 정도 걸릴 수 있습니다. 완료되면 Route 53에 등록된 도메인에 대한 인터넷 트래픽이 Lightsail 컨테이너 서비스로 라우팅을 시작해야 합니다.

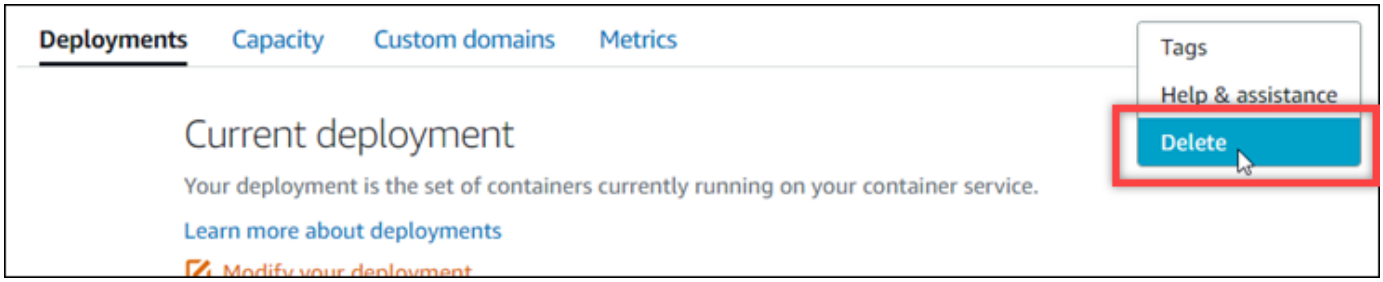
Lightsail 컨테이너 서비스 삭제

Amazon Lightsail 컨테이너 서비스를 더 이상 사용하지 않을 경우 언제든지 삭제할 수 있습니다. 컨테이너 서비스를 삭제하면 해당 서비스와 관련된 모든 배포 및 등록된 컨테이너 이미지가 영구적으로 삭제됩니다. 단, 생성한 SSL/TLS 인증서 및 도메인은 Lightsail 계정에 남아 있으므로 다른 리소스에서 사용할 수 있습니다. 컨테이너 서비스에 대한 자세한 내용은 Amazon Lightsail의 컨테이너 서비스를 참조하세요.

컨테이너 서비스 삭제

컨테이너 서비스를 삭제하려면 다음 절차를 완료하세요.

1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창의 이미지에서 컨테이너를 선택합니다.
3. 삭제할 컨테이너 서비스의 이름을 선택합니다.
4. 탭 메뉴에서 줄임표 아이콘을 선택한 다음 삭제(Delete)를 선택합니다.



5. 컨테이너 서비스 삭제(Delete container service)를 선택하여 서비스를 삭제합니다.
6. 프롬프트가 표시되면 예, 삭제합니다(Yes, delete)를 선택하여 영구적으로 삭제된다는 점을 확인합니다.

   컨테이너 서비스가 잠시 후 삭제됩니다.

Amazon Lightsail의 보안

AWS는 클라우드 보안을 가장 중요하게 생각합니다. AWS 고객으로서 여러분은 가장 높은 보안 요구 사항을 충족하기 위해 설계된 데이터 센터 및 네트워크 아키텍처의 혜택을 받게 됩니다.

보안은 AWS과 사용자의 공동 책임입니다. 공동 책임 모델은 이 사항을 클라우드 내 보안 및 클라우드의 보안으로 설명합니다.

- 클라우드의 보안 - AWS는 AWS 클라우드 내에서 AWS 서비스를 실행하는 인프라를 보호할 책임이 있습니다. AWS 또한 안전하게 사용할 수 있는 서비스를 제공합니다. 규정 준수 프로그램과 적용되는 서비스에 대한 자세한 내용은 규정 준수 프로그램 제공 범위 내 AWS 서비스를 참조하세요.
- 클라우드의 보안 – 귀하의 책임은 귀하가 사용하는 AWS서비스에 의해 결정됩니다. 또한 귀하는 귀사의 데이터 민감도, 귀사의 요구 사항, 관련 법률 및 규정을 비롯한 기타 요소에 대해서도 책임이 있습니다.

이 설명서는 Amazon Lightsail 사용 시 공동 책임 모델을 적용하는 방법을 이해하는 데 도움이 됩니다. 다음 주제에서는 보안 및 규정 준수 목표를 충족하도록 Amazon Lightsail을(를) 구성하는 방법을 보여 줍니다. 또한 Amazon Lightsail 리소스를 모니터링하고 보호하는 데 도움이 되는 다른 AWS 서비스를 사용하는 방법을 배우게 됩니다.

Amazon Lightsail에서 인프라 보안

관리형 서비스인 Amazon Lightsail은(는) AWS 글로벌 네트워크 보안으로 보호됩니다. AWS 보안 서비스와 AWS의 인프라 보호 방법에 대한 자세한 내용은 AWS 클라우드 보안을 참조하세요. 인프라 보안에 대한 모범 사례를 사용하여 AWS 환경을 설계하려면 보안 원칙 AWS Well-Architected Framework의 인프라 보호를 참조하세요.

AWS에서 게시한 API 직접 호출을 사용하여 네트워크를 통해 Lightsail에 액세스합니다. 고객은 다음을 지원해야 합니다.

- Transport Layer Security(TLS) TLS 1.2는 필수이며 TLS 1.3을 권장합니다.
- DHE(Ephemeral Diffie-Hellman) 또는 ECDHE(Elliptic Curve Ephemeral Diffie-Hellman)와 같은 완전 전송 보안(PFS)이 포함된 암호 제품군 Java 7 이상의 최신 시스템은 대부분 이러한 모드를 지원합니다.

또한 요청은 액세스 키 ID 및 IAM 위탁자와 관련된 시크릿 액세스 키를 사용하여 서명해야 합니다. 또는 AWS Security Token Service(AWS STS)를 사용하여 임시 자격 증명을 생성하여 요청에 서명할 수 있습니다.

Amazon Lightsail의 복원성

AWS 글로벌 인프라는 AWS 리전 및 가용 영역을 중심으로 구축됩니다. AWS 리전에서는 물리적으로 분리되고 격리된 다수의 가용 영역을 제공하며 이러한 가용 영역은 짧은 대기 시간, 높은 처리량 및 높은 중복성을 갖춘 네트워크에 연결되어 있습니다. 가용 영역을 사용하면 중단 없이 영역 간에 자동으로 장애 극복 조치가 이루어지는 애플리케이션 및 데이터베이스를 설계하고 운영할 수 있습니다. 가용 영역은 기존의 단일 또는 다중 데이터 센터 인프라보다 가용성, 내결함성, 확장성이 뛰어납니다.

AWS 리전 및 가용 영역에 대한 자세한 정보는 AWS 글로벌 인프라를 참조하십시오.

AWS 글로벌 인프라뿐만 아니라 Amazon Lightsail도 데이터 복원력과 백업 요구 사항을 지원하는 다양한 기능을 제공합니다.

- 리전 간에 인스턴스 및 디스크 스냅샷 복사 자세한 내용은 스냅샷을 참조하세요.
- 인스턴스 및 디스크 스냅샷 자동화. 자세한 내용은 스냅샷을 참조하세요.
- 로드 밸런서를 사용하여 단일 가용 영역 또는 여러 가용 영역의 여러 인스턴스 간에 수신 트래픽 분산 자세한 내용은 로드 밸런서를 참조하세요.

Amazon Lightsail의 I자격 증명 및 액세스 관리

대상

AWS Identity and Access Management (IAM)를 사용하는 방법은에서 수행하는 작업에 따라 다릅니다 Amazon Lightsail.

서비스 사용자 - Amazon Lightsail 서비스를 사용하여 작업을 수행하는 경우 필요한 보안 인증과 권한을 관리자가 제공합니다. 더 많은 Amazon Lightsail 기능을 사용하여 작업을 수행하게 되면 추가 권한이 필요할 수 있습니다. 액세스 권한 관리 방법을 이해하면 관리자에게 올바른 권한을 요청하는 데 도움이 됩니다. Amazon Lightsail에서 기능에 액세스할 수 없는 경우 Identity and Access Management(IAM) 문제 해결을 참조하세요.

서비스 관리자 - 회사에서 Amazon Lightsail 리소스를 책임지고 있는 경우 Amazon Lightsail에 대한 전체 액세스 권한을 가지고 있을 것입니다. 서비스 관리자는 직원이 액세스해야 하는 Amazon Lightsail

기능과 리소스를 결정합니다. 그런 다음 IAM 관리자에게 요청을 제출하여 서비스 사용자의 권한을 변경해야 합니다. 이 페이지의 정보를 검토하여 IAM의 기본 개념을 이해하세요. 회사가 Amazon Lightsail에서 IAM을 사용하는 방법에 대한 자세한 내용은 Amazon Lightsail에서 IAM을 사용하는 방식을 참조하세요.

IAM 관리자 - IAM 관리자라면 Amazon Lightsail에 대한 액세스 권한 관리 정책 작성 방법을 자세히 알고 싶을 것입니다. IAM에서 사용할 수 있는 Amazon Lightsail ID 기반 정책의 예를 보려면 Amazon Lightsail ID 기반 정책 예를 참조하세요.

자격 증명을 통한 인증

인증은 자격 증명 자격 증명을 AWS 사용하여에 로그인하는 방법입니다. 를 사용하여 로그인하는 방법에 대한 자세한 내용은 IAM 사용 설명서의 IAM 콘솔 및 로그인 페이지를 AWS Management Console 참조하세요.

AWS 계정 루트 사용자, IAM 사용자 또는 IAM 역할을 수임하여 인증(로그인 AWS)되어야 합니다. 회사의 싱글 사인온(SSO) 인증을 사용하거나 Google 또는 Facebook을 사용하여 로그인할 수도 있습니다. 이러한 경우 관리자는 이전에 IAM 역할을 사용하여 자격 증명 연동을 설정한 것입니다. 다른 회사의 자격 증명을 AWS 사용하여에 액세스하는 경우 역할을 간접적으로 수임하는 것입니다.

AWS Management Console에 직접 로그인하려면 루트 사용자 이메일 또는 IAM 사용자 이름과 암호를 사용하세요. 루트 사용자 또는 IAM 사용자 액세스 키를 사용하여 AWS 프로그래밍 방식으로에 액세스할 수 있습니다. AWS 는 자격 증명을 사용하여 요청에 암호화 방식으로 서명할 수 있는 SDK 및 명령줄 도구를 제공합니다. AWS 도구를 사용하지 않는 경우 요청에 직접 서명해야 합니다. 이렇게 하려면 인바운드 API 요청을 인증하기 위한 프로토콜인 서명 버전 4를 사용합니다. 요청 인증에 대한 자세한 내용은 https://docs.aws.amazon.com/general/latest/gr/signature-version-4.html의AWS 일반 참조서명 버전 4 서명 프로세스를 참조하세요.

사용하는 인증 방법에 상관 없이 추가 보안 정보를 제공해야 할 수도 있습니다. 예를 들어, 다중 인증(MFA)을 사용하여 계정의 보안을 강화하는 것이 AWS 좋습니다. 자세한 내용은 IAM 사용 설명서의 AWS에서 다중 인증(MFA) 사용을 참조하세요.

AWS 계정 루트 사용자

를 생성할 때 모든 AWS 서비스 및 리소스에 대한 완전한 액세스 권한이 있는 AWS 계정 theroot 사용자라는 하나의 로그인 자격 증명으로 AWS 계정시작합니다. 일상적인 태스크에 루트 사용자를 사용하지 않을 것을 강력히 권장합니다. 루트 사용자가 필요한 작업 목록은 IAM 사용자 설명서의 루트 사용자 자격 증명이 필요한 작업을 참조하세요.

IAM 사용자 및 그룹

IAM 사용자는 단일 개인 또는 애플리케이션에 대한 특정 권한을 가진 ID입니다. 장기 자격 증명이 있는 IAM 사용자 대신 임시 자격 증명을 사용하는 것이 좋습니다. 자세한 내용은 IAM 사용 설명서의 자격 증명 공급자와의 페더레이션을 사용하여 IAM 사용 설명서의 임시 자격 증명을 AWS 사용하여에 액세스하도록 인간 사용자에게 요구를 참조하세요.

IAM 그룹은 IAM 사용자 모음을 지정하고 대규모 사용자 집합에 대한 관리 권한을 더 쉽게 만듭니다. 자세한 내용은 IAM 사용 설명서의 IAM 사용자 사용 사례 섹션을 참조하세요.

IAM 역할

IAM 역할은 임시 자격 증명을 제공하는 특정 권한이 있는 자격 증명입니다. 사용자에서 IAM 역할(콘솔)로 전환하거나 또는 API 작업을 호출하여 역할을 수임할 수 있습니다. AWS CLI AWS 자세한 내용은 IAM 사용 설명서의 역할 수임 방법을 참조하세요.

IAM 역할은 페더레이션 사용자 액세스, 임시 IAM 사용자 권한, 교차 계정 액세스, 교차 서비스 액세스 및 Amazon EC2에서 실행되는 애플리케이션에 유용합니다. 자세한 내용은 IAM 사용 설명서의 교차 계정 리소스 액세스를 참조하세요.

임시 자격 증명이 있는 IAM 역할은 다음과 같은 상황에서 유용합니다.

- 임시 IAM 사용자 권한 - IAM 사용자는 IAM 역할을 수임하여 특정 작업에 대한 다양한 권한을 임시로 받을 수 있습니다.
- 페더레이션 사용자 액세스 - 페더레이션 ID에 권한을 부여하려면 역할을 생성하고 해당 역할의 권한을 정의합니다. 페더레이션 ID가 인증되면 역할이 연결되고 역할에 정의된 권한이 부여됩니다. 페더레이션 관련 역할에 대한 자세한 내용은 IAM 사용 설명서의 Create a role for a third-party identity provider (federation)를 참조하세요. IAM Identity Center를 사용하는 경우, 권한 집합을 구성합니다. 인증 후 ID가 액세스할 수 있는 항목을 제어하기 위해 IAM Identity Center는 권한 집합을 IAM의 역할과 연관짓습니다. 권한 세트에 대한 자세한 내용은 AWS IAM Identity Center 사용 설명서의 권한 세트를 참조하세요.
- 교차 계정 액세스 - IAM 역할을 사용하여 다른 계정의 사용자(신뢰할 수 있는 보안 주체)가 내 계정의 리소스에 액세스하도록 허용할 수 있습니다. 역할은 교차 계정 액세스를 부여하는 기본적인 방법입니다. 그러나 일부 에서는 정책을 리소스에 직접 연결할 AWS 서비스수 있습니다(역할을 프록시로 사용하는 대신). 크로스 계정 액세스를 위한 역할과 리소스 기반 정책의 차이점을 알아보려면 IAM 사용 설명서의 IAM 역할과 리소스 기반 정책의 차이를 참조하십시오.
- 교차 서비스 액세스 - 일부는 다른에서 기능을 AWS 서비스 사용합니다 AWS 서비스. 예를 들어, 서비스에서 호출하면 일반적으로 해당 서비스는 Amazon EC2에서 애플리케이션을 실행하거나

Amazon S3에 객체를 저장합니다. 서비스는 직접적으로 호출하는 위탁자의 권한을 사용하거나, 서비스 역할을 사용하거나, 또는 서비스 연결 역할을 사용하여 이 작업을 수행할 수 있습니다.

- 전달 액세스 세션(FAS) - IAM 사용자 또는 역할을 사용하여에서 작업을 수행하는 경우 AWS보안 주체로 간주됩니다. 정책은 보안 주체에게 권한을 부여합니다. 일부 서비스를 사용할 때는 다른 서비스에서 다른 작업을 트리거하는 작업을 수행할 수 있습니다. 이 경우 두 작업을 모두 수행할 수 있는 권한이 있어야 합니다. 작업에 정책의 추가 종속 작업이 필요한지 여부를 확인하려면 서비스 권한 부여 참조의 Amazon Lightsail에 대한 작업, 리소스 및 조건 키를 참조하세요.
- 서비스 역할 - 서비스 역할은 서비스가 사용자를 대신하여 작업을 수행하기 위해 수임하는 IAM 역할입니다. IAM 관리자는 IAM 내에서 서비스 역할을 생성, 수정 및 삭제할 수 있습니다. 자세한 내용은 IAM 사용 설명서의 AWS 서비스 AWS에 권한을 위임할 역할 생성을 참조하세요.
- 서비스 연결 역할 - 서비스 연결 역할은에 연결된 서비스 역할의 한 유형입니다 AWS 서비스. 서비스는 사용자를 대신하여 작업을 수행하기 위해 역할을 수임할 수 있습니다. 서비스 연결 역할은에 나타나 AWS 계정 며 서비스가 소유합니다. IAM 관리자는 서비스 연결 역할의 권한을 볼 수 있지만 편집은 할 수 없습니다.

- Amazon EC2에서 실행되는 애플리케이션 - IAM 역할을 사용하여 EC2 인스턴스에서 실행되고 AWS CLI 또는 AWS API 요청을 수행하는 애플리케이션의 임시 자격 증명을 관리할 수 있습니다. 이것은 EC2 인스턴스 내에 액세스 키를 저장하는 경우에 바람직한 방법입니다. EC2 인스턴스에 AWS 역할을 할당하고 모든 애플리케이션에서 사용할 수 있도록 하려면 인스턴스에 연결된 인스턴스 프로파일을 생성합니다. 인스턴스 프로필에는 역할이 포함되어 있으며 EC2 인스턴스에서 실행되는 프로그램이 임시 보안 인증을 얻을 수 있습니다. 자세한 정보는 IAM 사용 설명서의 IAM 역할을 사용하여 Amazon EC2 인스턴스에서 실행되는 애플리케이션에 권한 부여를 참조하세요.

IAM 역할을 사용할지 또는 IAM 사용자를 사용할지를 알아보려면 IAM 사용 설명서의 IAM 역할(사용자 대신)을 생성하는 경우,를 참조하세요.

정책을 사용하여 액세스 관리

정책을 AWS 생성하고 자격 증명 또는 리소스에 연결하여 AWS 에서 액세스를 제어합니다. 정책은 자격 증명 또는 리소스와 연결될 때 권한을 정의합니다.는 보안 주체가 요청할 때 이러한 정책을 AWS 평가합니다. 대부분의 정책은에 JSON 문서 AWS 로 저장됩니다. JSON 정책 문서에 대한 자세한 내용은 IAM 사용 설명서의 JSON 정책 개요 섹션을 참조하세요.

정책을 사용하여 관리자는 어떤 보안 주체가 어떤 리소스에 대해 어떤 조건에서 작업을 수행할 수 있는지 정의하여 누가 무엇을 액세스할 수 있는지 지정합니다.

기본적으로 사용자 및 역할에는 어떠한 권한도 없습니다. IAM 관리자는 IAM 정책을 생성하고 사용자가 수임할 수 있는 역할에 추가합니다. IAM 정책은 작업을 수행하기 위해 사용하는 방법과 관계없이 작업에 대한 권한을 정의합니다.

관리자는 AWS JSON 정책을 사용하여 누가 무엇에 액세스할 수 있는지 지정할 수 있습니다. 즉, 어떤 보안 주체가 어떤 리소스와 어떤 조건에서 작업을 수행할 수 있는지를 지정할 수 있습니다.

모든 IAM 개체(사용자 또는 역할)는 처음에는 권한이 없습니다. 다시 말해, 기본적으로 사용자는 아무 작업도 수행할 수 없으며, 자신의 암호를 변경할 수도 없습니다. 사용자에게 태스크를 수행할 권한을 부여하기 위해 관리자는 사용자에게 권한 정책을 연결해야 합니다. 또한 관리자는 의도한 권한을 가지고 있는 그룹에 사용자를 추가할 수 있습니다. 관리자가 그룹에 권한을 부여하면 그룹의 모든 사용자가 해당 권한을 받습니다.

IAM 정책은 태스크를 수행하기 위해 사용하는 방법과 상관없이 작업에 대한 권한을 정의합니다. 예를 들어, `iam:GetRole` 작업을 허용하는 정책이 있다고 가정합니다. 해당 정책이 있는 사용자는 AWS Management Console AWS CLI, 또는 AWS API에서 역할 정보를 가져올 수 있습니다.

자격 증명 기반 정책

ID 기반 정책은 ID(사용자, 사용자 그룹 또는 역할)에 연결하는 JSON 권한 정책 문서입니다. 이러한 정책은 자격 증명이 수행할 수 있는 작업, 대상 리소스 및 이에 관한 조건을 제어합니다. ID 기반 정책을 생성하는 방법을 알아보려면 IAM 사용 설명서에서 고객 관리형 정책으로 사용자 지정 IAM 권한 정의를 참조하세요.

ID 기반 정책은 인라인 정책(단일 ID에 직접 포함) 또는 관리형 정책(여러 ID에 연결된 독립 실행형 정책)일 수 있습니다. 관리형 정책 또는 인라인 정책을 선택하는 방법을 알아보려면 IAM 사용 설명서의 관리형 정책 및 인라인 정책 중에서 선택 섹션을 참조하세요.

ID 기반 정책은 IAM 사용자, 사용자 그룹 또는 역할과 같은 ID에 연결할 수 있는 JSON 권한 정책 문서입니다. 이러한 정책은 사용자 및 역할이 어떤 리소스와 어떤 조건에서 어떤 작업을 수행할 수 있는지를 제어합니다. 자격 증명 기반 정책을 생성하는 방법을 알아보려면 IAM 사용 설명서에서 고객 관리형 정책으로 사용자 지정 IAM 권한 정의를 참조하세요.

리소스 기반 정책

리소스 기반 정책은 리소스에 연결하는 JSON 정책 설명서입니다. 예를 들어 IAM 역할 신뢰 정책 및 Amazon S3 버킷 정책이 있습니다. 리소스 기반 정책을 지원하는 서비스에서 서비스 관리자는 이러한 정책을 사용하여 특정 리소스에 대한 액세스를 통제할 수 있습니다. 리소스 기반 정책에서 보안 주체를 지정해야 합니다.

리소스 기반 정책은 해당 서비스에 있는 인라인 정책입니다. 리소스 기반 정책에서는 IAM의 AWS 관리형 정책을 사용할 수 없습니다.

리소스 기반 정책은 리소스에 연결하는 JSON 정책 설명서입니다. 리소스 기반 정책의 예제는 IAM 역할 신뢰 정책과 Amazon S3 버킷 정책입니다. 리소스 기반 정책을 지원하는 서비스에서 서비스 관리자는 이러한 정책을 사용하여 특정 리소스에 대한 액세스를 통제할 수 있습니다. 정책이 연결된 리소스의 경우 정책은 지정된 보안 주체가 해당 리소스와 어떤 조건에서 어떤 작업을 수행할 수 있는지를 정의합니다. 리소스 기반 정책에서 보안 주체를 지정해야 합니다. 보안 주체에는 계정, 사용자, 역할, 페더레이션 사용자 또는이 포함될 수 있습니다 AWS 서비스.

액세스 제어 목록(ACL)

액세스 제어 목록(ACL)은 어떤 위탁자(계정 멤버, 사용자 또는 역할)가 리소스에 액세스할 수 있는 권한을 가지고 있는지를 제어합니다. ACL은 JSON 정책 문서 형식을 사용하지 않지만 리소스 기반 정책과 유사합니다.

Amazon S3 AWS WAF및 Amazon VPC는 ACLs. ACL에 관한 자세한 내용은 Amazon Simple Storage Service 개발자 가이드의 액세스 제어 목록(ACL) 개요를 참조하세요.

액세스 제어 목록(ACL)은 어떤 보안 주체(계정 멤버, 사용자 또는 역할)가 리소스에 액세스할 수 있는 권한을 가지고 있는지를 제어합니다. ACL은 JSON 정책 문서 형식을 사용하지 않지만 리소스 기반 정책과 유사합니다.

기타 정책 유형

AWS 는 보다 일반적인 정책 유형에서 부여한 최대 권한을 설정할 수 있는 추가 정책 유형을 지원합니다.

- 권한 경계 - ID 기반 정책에서 IAM 엔터티에 부여할 수 있는 최대 권한을 설정합니다. 자세한 정보는 IAM 사용 설명서의 IAM 엔터티의 권한 범위를 참조하세요.
- 서비스 제어 정책(SCP) - AWS Organizations내 조직 또는 조직 단위에 대한 최대 권한을 지정합니다. 자세한 내용은AWS Organizations 사용 설명서의 서비스 제어 정책을 참조하세요.
- 리소스 제어 정책(RCP) – 계정의 리소스에 사용할 수 있는 최대 권한을 설정합니다. 자세한 내용은 AWS Organizations 사용 설명서의 리소스 제어 정책(RCP)을 참조하세요.
- 세션 정책 – 역할 또는 페더레이션 사용자에 대해 임시 세션을 프로그래밍 방식으로 생성할 때 파라미터로 전달하는 고급 정책입니다. 자세한 내용은 IAM 사용 설명서의 세션 정책을 참조하세요.

- 권한 경계 – 권한 경계는 ID 기반 정책에 따라 IAM 엔터티(IAM 사용자 또는 역할)에 부여할 수 있는 최대 권한을 설정하는 고급 기능입니다. 엔터티에 대한 권한 경계를 설정할 수 있습니다. 그 결과로

얻는 권한은 개체의 자격 증명 기반 정책과 그 권한 경계의 교집합입니다. `Principal` 필드에서 사용자나 역할을 보안 주체로 지정하는 리소스 기반 정책은 권한 경계를 통해 제한되지 않습니다. 이러한 정책 중 하나에 포함된 명시적 거부는 허용을 재정의합니다. 권한 경계에 대한 자세한 정보는 IAM 사용 설명서의 IAM 엔티티에 대한 권한 경계를 참조하세요.

- 서비스 제어 정책(SCPs) - SCPs는의 조직 또는 조직 단위(OU)에 대한 최대 권한을 지정하는 JSON 정책입니다 AWS Organizations. AWS Organizations 는 AWS 계정 기업이 소유한 여러을 그룹화하고 중앙에서 관리하기 위한 서비스입니다. 조직에서 모든 기능을 활성화할 경우, 서비스 제어 정책(SCP)을 임의의 또는 모든 계정에 적용할 수 있습니다. SCP는 각 AWS 계정 루트 사용자를 포함하여 멤버 계정의 엔터티에 대한 권한을 제한합니다. 조직 및 SCP에 대한 자세한 정보는 AWS Organizations 사용 설명서의 SCP 작동 방식을 참조하십시오.
- 세션 정책 – 세션 정책은 역할 또는 페더레이션 사용자에 대해 임시 세션을 프로그래밍 방식으로 생성할 때 파라미터로 전달하는 고급 정책입니다. 결과적으로 얻는 세션의 권한은 사용자 또는 역할의 ID 기반 정책의 교차와 세션 정책입니다. 또한 권한을 리소스 기반 정책에서 가져올 수도 있습니다. 이러한 정책 중 하나에 포함된 명시적 거부는 허용을 재정의합니다. 자세한 내용은 IAM 사용 설명서의 세션 정책을 참조하세요.

여러 정책 유형

여러 정책 유형이 요청에 적용되는 경우, 결과 권한은 이해하기가 더 복잡합니다. 에서 여러 정책 유형이 관련될 때 요청을 허용할지 여부를 AWS 결정하는 방법을 알아보려면 IAM 사용 설명서의 정책 평가 로직을 참조하세요.

주제

- AWS 에 대한 관리형 정책 Amazon Lightsail
- Amazon Lightsail에서 IAM을 사용하는 방식
- IAM 사용자에게 Lightsail 액세스 권한 부여

AWS 에 대한 관리형 정책 Amazon Lightsail

사용자, 그룹 및 역할에 권한을 추가하려면 직접 정책을 작성하는 것보다 AWS 관리형 정책을 사용하는 것이 더 쉽습니다. 팀에 필요한 권한만 제공하는 IAM 고객 관리형 정책을 생성하기 위해서는 시간과 전문 지식이 필요합니다. 빠르게 시작하려면 AWS 관리형 정책을 사용할 수 있습니다. 이 정책은 일반적인 사용 사례를 다루며 사용자의 AWS 계정에서 사용할 수 있습니다. AWS 관리형 정책에 대한 자세한 내용은 IAM 사용 설명서의 AWS 관리형 정책을 참조하세요.

AWS 서비스는 AWS 관리형 정책을 유지 관리하고 업데이트합니다. AWS 관리형 정책에서는 권한을 변경할 수 없습니다. 서비스는 때때로 AWS 관리형 정책에 추가 권한을 추가하여 새 기능을 지원합니다. 이 유형의 업데이트는 정책이 연결된 모든 ID(사용자, 그룹 및 역할)에 적용됩니다. 서비스는 새 기능이 시작되거나 새 작업을 사용할 수 있게 될 때 AWS 관리형 정책을 업데이트할 가능성이 높습니다. 서비스는 AWS 관리형 정책에서 권한을 제거하지 않으므로 정책 업데이트로 인해 기존 권한이 손상되지 않습니다.

또한는 여러 서비스에 걸쳐 있는 직무에 대한 관리형 정책을 AWS 지원합니다. 예를 들어 ReadOnlyAccess AWS 관리형 정책은 모든 AWS 서비스 및 리소스에 대한 읽기 전용 액세스를 제공합니다. 서비스가 새 기능을 시작하면는 새 작업 및 리소스에 대한 읽기 전용 권한을 AWS 추가합니다. 직무 정책의 목록과 설명은 IAM 사용 설명서의 직무에 관한AWS 관리형 정책을 참조하세요.

AWS 관리형 정책: LightsailExportAccess

IAM 엔터티에는 LightsailExportAccess를 연결할 수 없습니다. 이 정책은 Lightsail이(가) 사용자를 대신하여 작업을 수행할 수 있도록 서비스 연결 역할에 연결됩니다. 자세한 내용은 서비스 연결 역할을 참조하세요.

이 정책은 Lightsail이 인스턴스 및 디스크 스냅샷을 Amazon Elastic Compute Cloud로 내보내고 Amazon Simple Storage Service(S3)에서 현재 계정 수준 퍼블릭 액세스 차단 구성을 가져오도록 허용하는 권한을 부여합니다.

권한 세부 정보

이 정책에는 다음 권한이 포함되어 있습니다.

- `ec2` - 인스턴스 이미지 및 디스크 스냅샷을 나열 및 복사할 수 있는 액세스를 허용합니다.
- `iam` - 서비스 연결 역할을 삭제하고 서비스 연결 역할의 삭제 상태를 검색할 수 있는 액세스를 허용합니다.
- `s3` - AWS 계정에 대한 `PublicAccessBlock` 구성을 검색할 수 있는 액세스를 허용합니다.

JSON

```
{
 "Version":"2012-10-17",
 "Statement": [
  {
   "Effect": "Allow",
```

```
    "Action": [
     "iam:DeleteServiceLinkedRole",
     "iam:GetServiceLinkedRoleDeletionStatus"
    ],
    "Resource": "arn:aws:iam::*:role/aws-service-role/lightsail.amazonaws.com/
AWSServiceRoleForLightsail*"
   },
   {
    "Effect": "Allow",
    "Action": [
     "ec2:CopySnapshot",
     "ec2:DescribeSnapshots",
     "ec2:CopyImage",
     "ec2:DescribeImages"
    ],
    "Resource": "*"
   },
   {
    "Effect": "Allow",
    "Action": [
     "s3:GetAccountPublicAccessBlock"
    ],
    "Resource": "*"
   }
  ]
}
```

Lightsail AWS 관리형 정책에 대한 업데이트

- `LightsailExportAccess` 관리형 정책에 대한 편집

  `LightsailExportAccess` 관리형 정책에 `s3:GetAccountPublicAccessBlock` 작업을 추가했습니다. 이것은 Lightsail이 Amazon S3에서 블록 퍼블릭 액세스 구성을 가져오도록 하용합니다.

  2022년 1월 14일

- Lightsail에서 변경 사항 추적 시작

  Lightsail가 AWS 관리형 정책에 대한 변경 내용 추적을 시작했습니다.

  2022년 1월 14일

Amazon Lightsail에서 IAM을 사용하는 방식

IAM을 사용하여 Lightsail에 대한 액세스를 관리하려면 먼저 어떤 IAM 기능을 Lightsail에 사용할 수 있는지를 이해해야 합니다. Lightsail 및 기타 AWS 서비스에서 IAM을 사용하는 방법을 전체적으로 알아보려면 IAM 사용 설명서의 AWS IAM으로 작업하는 서비스를 참조하세요.

Lightsail ID 기반 정책

IAM ID 기반 정책을 사용하면 허용되거나 거부되는 작업과 리소스 및 작업이 허용되거나 거부되는 조건을 지정할 수 있습니다. Lightsail는 특정 작업, 리소스 및 조건 키를 지원합니다. JSON 정책에서 사용하는 모든 요소에 대해 알고 싶다면 IAM 사용 설명서의 IAM JSON 정책 요소 참조를 참조하세요.

작업

관리자는 AWS JSON 정책을 사용하여 누가 무엇에 액세스할 수 있는지 지정할 수 있습니다. 즉, 어떤 보안 주체가 어떤 리소스와 어떤 조건에서 작업을 수행할 수 있는지를 지정할 수 있습니다.

JSON 정책의 `Action`요소는 정책에서 액세스를 허용하거나 거부하는 데 사용할 수 있는 작업을 설명합니다. 연결된 작업을 수행할 수 있는 권한을 부여하기 위한 정책에 작업을 포함하세요.

Lightsail의 정책 작업은 작업 앞에 `lightsail:` 접두사를 사용합니다. 예를 들어 누군가에게 Lightsail `CreateInstances` API 작업을 통해 Lightsail 인스턴스를 실행하는 권한을 부여하려면 해당 정책에 `lightsail:CreateInstances` 작업을 포함시킵니다. 정책 문에는 `Action` 또는 `NotAction` 요소가 포함되어야 합니다. Lightsail는 이 서비스로 수행할 수 있는 작업을 설명하는 고유한 작업 집합을 정의합니다.

명령문 하나에 여러 태스크를 지정하려면 다음과 같이 쉼표로 구분합니다.

```
"Action": [
      "lightsail:action1",
      "lightsail:action2"
```

와일드카드(*)를 사용하여 여러 작업을 지정할 수 있습니다. 예를 들어, `Create`라는 단어로 시작하는 모든 작업을 지정하려면 다음 작업을 포함합니다.

```
"Action": "lightsail:Create*"
```

Lightsail 작업 목록을 보려면 IAM 사용 설명서의 Amazon Lightsail에서 정의한 작업을 참조하세요.

리소스

관리자는 AWS JSON 정책을 사용하여 누가 무엇에 액세스할 수 있는지 지정할 수 있습니다. 즉, 어떤 보안 주체가 어떤 리소스와 어떤 조건에서 작업을 수행할 수 있는지를 지정할 수 있습니다.

Resource JSON 정책 요소는 작업이 적용되는 하나 이상의 객체를 지정합니다. 모범 사례에 따라 Amazon 리소스 이름(ARN)을 사용하여 리소스를 지정합니다. 리소스 수준 권한을 지원하지 않는 작업의 경우, 와일드카드(*)를 사용하여 해당 문이 모든 리소스에 적용됨을 나타냅니다.

```
"Resource": "*"
```

> **Important**
>
> Lightsail에서는 일부 API 작업에 대해 리소스 수준 권한을 지원하지 않습니다. 자세한 내용은 리소스 수준 권한 및 태그 기반 권한 부여에 대한 지원을 참조하세요.

Lightsail 인스턴스 리소스에는 다음 ARN이 있습니다.

```
arn:${Partition}:lightsail:${Region}:${Account}:Instance/${InstanceId}
```

ARN 형식에 대한 자세한 내용은 Amazon 리소스 이름(ARNs) 및 AWS 서비스 네임스페이스를 참조하세요.

예를 들어 문에서 ea123456-e6b9-4f1d-b518-3ad1234567e6 인스턴스를 지정하려면 다음 ARN을 사용합니다.

```
"Resource": "arn:aws:lightsail:us-east-1:123456789012:Instance/ea123456-e6b9-4f1d-
b518-3ad1234567e6"
```

특정 계정에 속하는 모든 인스턴스를 지정하려면 와일드카드(*)를 사용합니다.

```
"Resource": "arn:aws:lightsail:us-east-1:123456789012:Instance/*"
```

리소스를 생성하기 위한 작업과 같은 일부 Lightsail 작업은 특정 리소스에서 수행할 수 없습니다. 이러한 경우, 와일드카드(*)를 사용해야 합니다.

```
"Resource": "*"
```

다양한 Lightsail API 작업에는 여러 리소스가 관여합니다. 예를 들어 `AttachDisk`는 Lightsail 블록 스토리지 디스크를 인스턴스에 연결하므로 IAM 사용자에게 디스크 사용 권한과 인스턴스 사용 권한이 있어야 합니다. 단일 문에서 여러 리소스를 지정하려면 ARN을 쉼표로 구분합니다.

```
"Resource": [
      "resource1",
      "resource2"
```

Lightsail 리소스 유형 및 해당 ARN의 목록을 보려면 IAM 사용 설명서의 Amazon Lightsail에서 정의된 리소스를 참조하세요. 각 리소스의 ARN을 지정할 수 있는 작업을 알아보려면 Amazon Lightsail가 정의한 작업을 참조하세요.

조건 키

관리자는 AWS JSON 정책을 사용하여 누가 무엇에 액세스할 수 있는지 지정할 수 있습니다. 즉, 어떤 보안 주체가 어떤 리소스와 어떤 조건에서 작업을 수행할 수 있는지를 지정할 수 있습니다.

`Condition` 요소는 정의된 기준에 따라 문이 실행되는 시기를 지정합니다. 같음(equals) 또는 미만(less than)과 같은 조건 연산자를 사용하여 정책의 조건을 요청의 값과 일치시키는 조건식을 생성할 수 있습니다. 모든 AWS 전역 조건 키를 보려면 IAM 사용 설명서의 AWS 전역 조건 컨텍스트 키를 참조하세요.

Lightsail는 서비스별 조건 키를 제공하지 않지만, 일부 전역 조건 키 사용을 지원합니다. 모든 AWS 전역 조건 키를 보려면 IAM 사용 설명서의 AWS 전역 조건 컨텍스트 키를 참조하세요.

Lightsail 조건 키 목록을 보려면 IAM 사용 설명서의 Amazon Lightsail에 대한 조건 키를 참조하세요. 조건 키를 사용할 수 있는 작업과 리소스를 알아보려면 Amazon Lightsail가 정의한 작업 단원을 참조하세요.

예제

Lightsail 자격 증명 기반 정책의 예를 보려면 Amazon Lightsail 자격 증명 기반 정책 예제를 참조하세요.

Lightsail 리소스 기반 정책

Lightsail는 리소스 기반 정책을 지원하지 않습니다.

액세스 제어 목록(ACL)

Lightsail은 ACL(액세스 제어 목록)을 지원하지 않습니다.

Lightsail 태그 기반 권한 부여

태그를 Lightsail 리소스에 연결하거나 요청을 통해 태그를 Lightsail에 전달할 수 있습니다. 태그를 기반으로 액세스를 통제하려면 `lightsail:ResourceTag/key-name`, `aws:RequestTag/key-name` 또는 `aws:TagKeys` 조건 키를 사용하여 정책의 조건 요소에 태그 정보를 제공합니다.

> **Important**
>
> Lightsail에서는 일부 API 작업에 대한 태그 기반 권한 부여를 지원하지 않습니다. 자세한 내용은 리소스 수준 권한 및 태그 기반 권한 부여에 대한 지원을 참조하세요.

Lightsail 리소스 태그 지정에 대한 자세한 내용은 태그를 참조하세요.

해당 리소스의 태그를 기반으로 리소스에 대한 액세스를 제한하는 자격 증명 기반 정책 예제를 보려면 태그를 기반으로 Lightsail 리소스 생성 및 삭제 허용을 참조하세요.

Lightsail IAM 역할

IAM 역할은 특정 권한을 가지고 있는 AWS 계정 내 엔터티입니다.

Lightsail에서 임시 보안 인증 사용

임시 보안 인증을 사용하여 페더레이션을 통해 로그인하거나, IAM 역할을 맡거나, 교차 계정 역할을 맡을 수 있습니다. AssumeRole 또는 GetFederationToken 같은 AWS STS API 작업을 호출하여 임시 보안 인증을 가져옵니다.

Lightsail는 임시 자격 증명 사용을 지원합니다.

서비스 연결 역할

서비스 연결 역할을 사용하면 AWS 서비스가 다른 서비스의 리소스에 액세스하여 사용자를 대신하여 작업을 완료할 수 있습니다. 서비스 연결 역할은 IAM 계정에 나타나고 서비스가 소유합니다. IAM 관리자는 서비스 연결 역할의 권한을 볼 수 있지만 편집할 수 없습니다.

Lightsail에서는 서비스 연결 역할을 지원합니다. Lightsail 서비스 연결 역할을 생성하고 관리하는 방법에 대한 자세한 내용은 서비스 연결 역할을 참조하세요.

서비스 역할

Lightsail은 서비스 역할을 지원하지 않습니다.

주제

- Lightsail에서 IAM 자격 증명 정책을 사용하여 최소 권한 부여
- IAM 정책을 사용하여 특정 Lightsail 리소스에 대한 액세스 권한 부여
- Amazon Lightsail에 대한 서비스 연결 역할 사용
- IAM 정책을 사용하여 Lightsail 버킷 관리

Lightsail에서 IAM 자격 증명 정책을 사용하여 최소 권한 부여

기본적으로 IAM 사용자 및 역할은 Lightsail 리소스를 생성하거나 수정할 수 있는 권한이 없습니다. 또한 AWS Management Console AWS CLI또는 AWS API를 사용하여 작업을 수행할 수 없습니다. IAM 관리자는 지정된 리소스에서 특정 API 작업을 수행할 수 있는 권한을 사용자와 역할에게 부여하는 IAM 정책을 생성해야 합니다. 그런 다음 관리자는 해당 권한이 필요한 IAM 사용자 또는 그룹에 이러한 정책을 연결해야 합니다.

이러한 예제 JSON 정책 문서를 사용하여 IAM ID 기반 정책을 생성하는 방법을 알아보려면 IAM 사용 설명서의 JSON 탭에서 정책 생성을 참조하세요.

정책 모범 사례

자격 증명 기반 정책에 따라 계정에서 사용자가 Amazon Lightsail 리소스를 생성, 액세스 또는 삭제할 수 있는지 여부가 결정됩니다. 이 작업으로 인해 AWS 계정에 비용이 발생할 수 있습니다. ID 기반 정책을 생성하거나 편집할 때는 다음 지침과 권장 사항을 따르세요.

- AWS 관리형 정책을 시작하고 최소 권한으로 전환 - 사용자 및 워크로드에 권한 부여를 시작하려면 많은 일반적인 사용 사례에 대한 권한을 부여하는 AWS 관리형 정책을 사용합니다. 에서 사용할 수 있습니다 AWS 계정. 사용 사례에 맞는 AWS 고객 관리형 정책을 정의하여 권한을 추가로 줄이는 것이 좋습니다. 자세한 내용은 IAM 사용 설명서의 AWS 관리형 정책 또는 AWS 직무에 대한 관리형 정책을 참조하세요.
- 최소 권한 적용 – IAM 정책을 사용하여 권한을 설정하는 경우, 작업을 수행하는 데 필요한 권한만 부여합니다. 이렇게 하려면 최소 권한으로 알려진 특정 조건에서 특정 리소스에 대해 수행할 수 있는 작업을 정의합니다. IAM을 사용하여 권한을 적용하는 방법에 대한 자세한 정보는 IAM 사용 설명서에 있는 IAM의 정책 및 권한을 참조하세요.
- IAM 정책의 조건을 사용하여 액세스 추가 제한 – 정책에 조건을 추가하여 작업 및 리소스에 대한 액세스를 제한할 수 있습니다. 예를 들어, SSL을 사용하여 모든 요청을 전송해야 한다고 지정하는 정책 조건을 작성할 수 있습니다. AWS 서비스와 같은 특정를 통해 사용되는 경우 조건을 사용하여 서비스 작업에 대한 액세스 권한을 부여할 수도 있습니다 CloudFormation. 자세한 내용은 IAM 사용 설명서의 IAM JSON 정책 요소: 조건을 참조하세요.

- IAM Access Analyzer를 통해 IAM 정책을 확인하여 안전하고 기능적인 권한 보장 - IAM Access Analyzer에서는 IAM 정책 언어(JSON)와 모범 사례가 정책에서 준수되도록 새로운 및 기존 정책을 확인합니다. IAM Access Analyzer는 100개 이상의 정책 확인 항목과 실행 가능한 추천을 제공하여 안전하고 기능적인 정책을 작성하도록 돕습니다. 자세한 내용은 IAM 사용 설명서의 IAM Access Analyzer에서 정책 검증을 참조하세요.
- 다중 인증(MFA) 필요 -에서 IAM 사용자 또는 루트 사용자가 필요한 시나리오가 있는 경우 추가 보안을 위해 MFA를 AWS 계정켭니다. API 작업을 직접적으로 호출할 때 MFA가 필요하면 정책에 MFA 조건을 추가합니다. 자세한 내용은 IAM 사용 설명서의 MFA를 통한 보안 API 액세스를 참조하세요.

IAM의 모범 사례에 대한 자세한 내용은 IAM 사용 설명서의 IAM의 보안 모범 사례를 참조하세요.

Lightsail 콘솔 사용

Amazon Lightsail 콘솔에 액세스하려면 모든 Lightsail 작업 및 리소스에 대한 전체 액세스 권한이 있어야 합니다. 이러한 권한은 AWS 계정의 Lightsail 리소스에 대한 세부 정보를 나열하고 볼 수 있도록 허용해야 합니다. 최소 필수 권한(즉, 전체 액세스 권한이 아님)보다 더 제한적인 자격 증명 기반 정책을 만들면 콘솔이 해당 정책에 연결된 객체(IAM 사용자 또는 역할)에 대해 의도대로 작동하지 않습니다.

해당 객체가 Lightsail 콘솔을 사용할 수 있도록 하려면 다음과 같은 정책을 객체에 연결합니다. 자세한 내용은 IAM 사용 설명서의 사용자에게 권한 추가를 참조하세요.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "lightsail:*"
            ],
            "Resource": "*"
        }
    ]
}
```

AWS CLI 또는 AWS API만 호출하는 사용자에게는 최소 콘솔 권한을 허용할 필요가 없습니다. 그 대신, 수행하려는 API 작업과 일치하는 작업에만 액세스할 수 있도록 합니다.

사용자가 자신이 권한을 볼 수 있도록 허용

이 예제는 IAM 사용자가 자신의 사용자 ID에 연결된 인라인 및 관리형 정책을 볼 수 있도록 허용하는 정책을 생성하는 방법을 보여줍니다. 이 정책에는 콘솔에서 또는 AWS CLI 또는 AWS API를 사용하여 프로그래밍 방식으로이 작업을 완료할 수 있는 권한이 포함됩니다.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "ViewOwnUserInfo",
            "Effect": "Allow",
            "Action": [
                "iam:GetUserPolicy",
                "iam:ListGroupsForUser",
                "iam:ListAttachedUserPolicies",
                "iam:ListUserPolicies",
                "iam:GetUser"
            ],
            "Resource": ["arn:aws:iam::*:user/${aws:username}"]
        },
        {
            "Sid": "NavigateInConsole",
            "Effect": "Allow",
            "Action": [
                "iam:GetGroupPolicy",
                "iam:GetPolicyVersion",
                "iam:GetPolicy",
                "iam:ListAttachedGroupPolicies",
                "iam:ListGroupPolicies",
                "iam:ListPolicyVersions",
                "iam:ListPolicies",
                "iam:ListUsers"
            ],
            "Resource": "*"
        }
    ]
}
```

태그를 기반으로 Lightsail 리소스 생성 및 삭제 허용

자격 증명 기반 정책의 조건을 사용하여 태그를 기반으로 Lightsail 리소스에 대한 액세스를 제어할 수 있습니다. 이 예제에서는 생성 요청에서 `allow`의 키 태그와 `true`의 값이 정의되지 않은 경우 사용자

가 새 Lightsail 리소스를 생성하지 못하도록 제한하는 정책을 만드는 방법을 보여 줍니다. 또한 이 정책은 `allow/true` 키-값 태그가 없는 한 사용자가 리소스를 삭제하지 않도록 제한합니다.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "lightsail:Create*",
                "lightsail:TagResource",
                "lightsail:UntagResource"
            ],
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "aws:RequestTag/allow": "true"
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "lightsail:Delete*",
                "lightsail:TagResource",
                "lightsail:UntagResource"
            ],
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "aws:ResourceTag/allow": "true"
                }
            }
        }
    ]
}
```

다음 예제에서는 사용자가 `allow/false`가 아닌 키-값 태그를 가진 리소스에 대한 태그를 변경하지 못하도록 제한합니다.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Deny",
            "Action": [
                "lightsail:TagResource"
            ],
            "Resource": "*",
            "Condition": {
                "StringNotEquals": {
                    "aws:ResourceTag/allow": "false"
                }
            }
        }
    ]
}
```

이러한 정책을 계정의 IAM 사용자에게 연결할 수 있습니다. 자세한 내용은 IAM 사용 설명서의 IAM JSON 정책 요소: 조건을 참조하세요.

IAM 정책을 사용하여 특정 Lightsail 리소스에 대한 액세스 권한 부여

리소스 수준 권한은 사용자가 작업을 수행할 리소스를 지정할 수 있는 기능입니다. Amazon Lightsail은 리소스 수준 권한을 지원합니다. 즉, 필요 조건을 지정하거나 사용 또는 편집 가능한 특정 리소스를 지정하여 사용자가 특정 Lightsail 작업을 사용할 수 있는지 여부를 제어할 수 있습니다. 예를 들어, 사용자에게 특정 Amazon 리소스 이름(ARN)을 사용하여 인스턴스 또는 데이터베이스를 관리하는 권한을 부여할 수 있습니다.

> **Important**
>
> Lightsail에서는 일부 API 작업에 대해 리소스 수준 권한을 지원하지 않습니다. 자세한 내용은 리소스 수준 권한 및 태그 기반 권한 부여에 대한 지원을 참조하세요.

Lightsail 작업으로 생성 또는 수정되는 리소스 및 IAM 정책문에 사용할 수 있는 Lightsail 조건 키와 ARN에 대한 자세한 내용은 IAM 사용 설명서의 Amazon Lightsail에 사용되는 작업, 리소스 및 조건 키를 참조하세요.

특정 인스턴스 관리 허용

다음 정책은 인스턴스 재부팅/시작/중지, 인스턴스 포트 관리 및 특정 인스턴스의 스냅샷 생성을 위한 액세스 권한을 부여합니다. 또한 Lightsail 계정의 다른 인스턴스 관련 정보 및 리소스에 대한 읽기 전용 액세스 권한을 제공합니다. 정책에서 *InstanceARN*을 인스턴스의 Amazon 리소스 이름(ARN)으로 대체합니다.

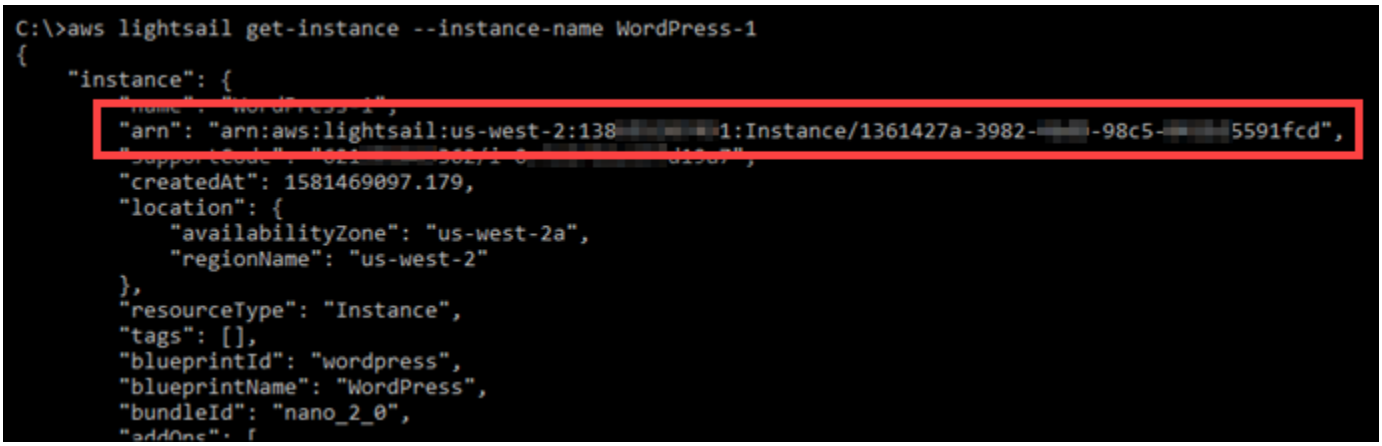
JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "VisualEditor0",
            "Effect": "Allow",
            "Action": [
                "lightsail:GetActiveNames",
                "lightsail:GetAlarms",
                "lightsail:GetAutoSnapshots",
                "lightsail:GetBlueprints",
                "lightsail:GetBundles",
                "lightsail:GetCertificates",
                "lightsail:GetCloudFormationStackRecords",
                "lightsail:GetContactMethods",
                "lightsail:GetDisk",
                "lightsail:GetDisks",
                "lightsail:GetDiskSnapshot",
                "lightsail:GetDiskSnapshots",
                "lightsail:GetDistributionBundles",
                "lightsail:GetDistributionLatestCacheReset",
                "lightsail:GetDistributionMetricData",
                "lightsail:GetDistributions",
                "lightsail:GetDomain",
                "lightsail:GetDomains",
                "lightsail:GetExportSnapshotRecords",
                "lightsail:GetInstance",
                "lightsail:GetInstanceAccessDetails",
                "lightsail:GetInstanceMetricData",
```

```
                "lightsail:GetInstancePortStates",
                "lightsail:GetInstances",
                "lightsail:GetInstanceSnapshot",
                "lightsail:GetInstanceSnapshots",
                "lightsail:GetInstanceState",
                "lightsail:GetKeyPair",
                "lightsail:GetKeyPairs",
                "lightsail:GetLoadBalancer",
                "lightsail:GetLoadBalancerMetricData",
                "lightsail:GetLoadBalancers",
                "lightsail:GetLoadBalancerTlsCertificates",
                "lightsail:GetOperation",
                "lightsail:GetOperations",
                "lightsail:GetOperationsForResource",
                "lightsail:GetRegions",
                "lightsail:GetRelationalDatabase",
                "lightsail:GetRelationalDatabaseBlueprints",
                "lightsail:GetRelationalDatabaseBundles",
                "lightsail:GetRelationalDatabaseEvents",
                "lightsail:GetRelationalDatabaseLogEvents",
                "lightsail:GetRelationalDatabaseLogStreams",
                "lightsail:GetRelationalDatabaseMetricData",
                "lightsail:GetRelationalDatabaseParameters",
                "lightsail:GetRelationalDatabases",
                "lightsail:GetRelationalDatabaseSnapshot",
                "lightsail:GetRelationalDatabaseSnapshots",
                "lightsail:GetStaticIp",
                "lightsail:GetStaticIps",
                "lightsail:IsVpcPeered"
            ],
            "Resource": "*"
        },
        {
            "Sid": "VisualEditor2",
            "Effect": "Allow",
            "Action": [
                "lightsail:CloseInstancePublicPorts",
                "lightsail:CreateInstanceSnapshot",
                "lightsail:OpenInstancePublicPorts",
                "lightsail:PutInstancePublicPorts",
                "lightsail:RebootInstance",
                "lightsail:StartInstance",
                "lightsail:StopInstance"
            ],
```

```
            "Resource": "arn:aws:lightsail:us-
east-2:123456789012:Instance/244ad76f-8aad-4741-809f-12345EXAMPLE"
        }
    ]
}
```

인스턴스의 ARN을 가져오려면 `GetInstance` Lightsail API 작업을 사용하고 `instanceName` 파라미터로 인스턴스의 이름을 지정합니다. 다음 예와 같이 작업 결과에 인스턴스 ARN이 나열됩니다. 자세한 내용은 Amazon Lightsail API 레퍼런스의 GetInstance을 참조하세요.



특정 데이터베이스 관리 허용

다음 정책은 특정 데이터베이스의 재부팅/시작/중지 및 업데이트를 위한 액세스 권한을 부여합니다. 또한 Lightsail 계정의 다른 데이터베이스 관련 정보 및 리소스에 대한 읽기 전용 액세스 권한을 제공합니다. 정책에서 *DatabaseARN*을 데이터베이스의 Amazon 리소스 이름(ARN)으로 대체합니다.

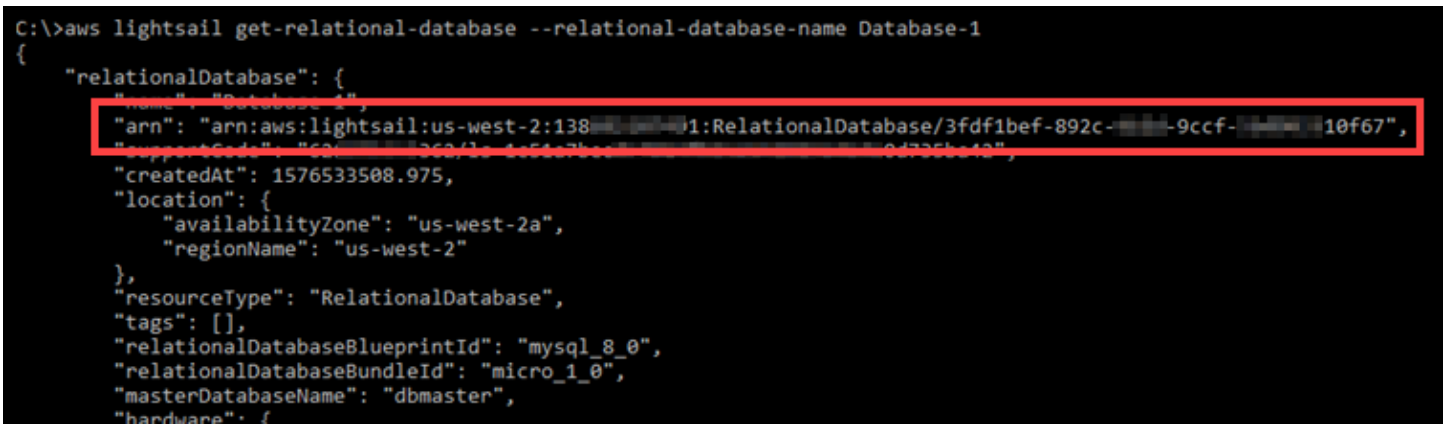
JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "VisualEditor0",
            "Effect": "Allow",
            "Action": [
                "lightsail:GetActiveNames",
                "lightsail:GetAlarms",
                "lightsail:GetAutoSnapshots",
                "lightsail:GetBlueprints",
                "lightsail:GetBundles",
```

```
                "lightsail:GetCertificates",
                "lightsail:GetCloudFormationStackRecords",
                "lightsail:GetContactMethods",
                "lightsail:GetDisk",
                "lightsail:GetDisks",
                "lightsail:GetDiskSnapshot",
                "lightsail:GetDiskSnapshots",
                "lightsail:GetDistributionBundles",
                "lightsail:GetDistributionLatestCacheReset",
                "lightsail:GetDistributionMetricData",
                "lightsail:GetDistributions",
                "lightsail:GetDomain",
                "lightsail:GetDomains",
                "lightsail:GetExportSnapshotRecords",
                "lightsail:GetInstance",
                "lightsail:GetInstanceAccessDetails",
                "lightsail:GetInstanceMetricData",
                "lightsail:GetInstancePortStates",
                "lightsail:GetInstances",
                "lightsail:GetInstanceSnapshot",
                "lightsail:GetInstanceSnapshots",
                "lightsail:GetInstanceState",
                "lightsail:GetKeyPair",
                "lightsail:GetKeyPairs",
                "lightsail:GetLoadBalancer",
                "lightsail:GetLoadBalancerMetricData",
                "lightsail:GetLoadBalancers",
                "lightsail:GetLoadBalancerTlsCertificates",
                "lightsail:GetOperation",
                "lightsail:GetOperations",
                "lightsail:GetOperationsForResource",
                "lightsail:GetRegions",
                "lightsail:GetRelationalDatabase",
                "lightsail:GetRelationalDatabaseBlueprints",
                "lightsail:GetRelationalDatabaseBundles",
                "lightsail:GetRelationalDatabaseEvents",
                "lightsail:GetRelationalDatabaseLogEvents",
                "lightsail:GetRelationalDatabaseLogStreams",
                "lightsail:GetRelationalDatabaseMetricData",
                "lightsail:GetRelationalDatabaseParameters",
                "lightsail:GetRelationalDatabases",
                "lightsail:GetRelationalDatabaseSnapshot",
                "lightsail:GetRelationalDatabaseSnapshots",
                "lightsail:GetStaticIp",
```

```
                "lightsail:GetStaticIps",
                "lightsail:IsVpcPeered"
            ],
            "Resource": "*"
        },
        {
            "Sid": "VisualEditor2",
            "Effect": "Allow",
            "Action": [
                "lightsail:RebootRelationalDatabase",
                "lightsail:StartRelationalDatabase",
                "lightsail:StopRelationalDatabase",
                "lightsail:UpdateRelationalDatabase"
            ],
            "Resource": "arn:aws:lightsail:us-
east-2:123456789012:RelationalDatabase/244ad76f-8aad-4741-809f-12345EXAMPLE"
        }
    ]
}
```

데이터베이스의 ARN을 가져오려면 `GetRelationalDatabase` Lightsail API 작업을 사용하고 `relationalDatabaseName` 파라미터를 사용하여 데이터베이스의 이름을 지정합니다. 다음 예와 같이 작업 결과에 데이터베이스 ARN이 나열됩니다. 자세한 내용은 Amazon Lightsail API 레퍼런스의 GetRelationalDatabase를 참조하세요.



Amazon Lightsail에 대한 서비스 연결 역할 사용

Amazon Lightsail는 AWS Identity and Access Management (IAM) 서비스 연결 역할을 사용합니다. 서비스 연결 역할은 Amazon Lightsail에 직접 연결된 고유한 유형의 IAM 역할입니다. 서비스 연결 역할은

에서 사전 정의Amazon Lightsail하며가 사용자를 대신하여 다른 AWS 서비스를 호출하는 데 Lightsail 필요한 모든 권한을 포함합니다.

서비스 연결 역할을 통해 Amazon Lightsail 설정이 쉬워지는데 필요한 권한을 수동으로 추가할 필요가 없기 때문입니다. Amazon Lightsail에서 서비스 연결 역할 권한을 정의하므로, 달리 정의되지 않은 한 Amazon Lightsail에서만 해당 역할을 맡을 수 있습니다. 정의된 권한에는 신뢰 정책과 권한 정책이 포함되며, 권한 정책은 다른 IAM 엔터티에 연결할 수 없습니다.

먼저 관련 리소스를 삭제한 후에만 서비스 연결 역할을 삭제할 수 있습니다. 이렇게 하면 리소스에 대한 액세스 권한을 부주의로 삭제할 수 없기 때문에 Amazon Lightsail 리소스가 보호됩니다.

서비스 연결 역할을 지원하는 기타 서비스에 대한 자세한 내용은 IAM으로 작업하는 AWS 서비스를 참조하고 서비스 연결 역할 열에 예가 있는 서비스를 찾습니다. 해당 서비스에 대한 서비스 연결 역할 설명서를 보려면 예 링크를 선택합니다.

Amazon Lightsail에 대한 서비스 연결 역할 권한

Amazon Lightsail는 AWSServiceRoleForLightsail - 역할이라는 서비스 연결 역할을 사용하여 Lightsail 인스턴스 및 블록 스토리지 디스크 스냅샷을 Amazon Elastic Compute Cloud(Amazon EC2)로 내보내고 Amazon Simple Storage Service(Amazon S3)에서 현재 계정 수준 퍼블릭 액세스 차단 구성을 가져옵니다.

AWSServiceRoleForLightsail 서비스 연결 역할은 역할을 수임하기 위해 다음 서비스를 신뢰합니다.

- `lightsail.amazonaws.com`

역할 권한 정책은 Amazon Lightsail가 지정된 리소스에서 다음 작업을 완료하도록 허용합니다.

- 작업: 모든 AWS 리소스`ec2:CopySnapshot`에 대해.
- 작업: 모든 AWS 리소스`ec2:DescribeSnapshots`에 대해.
- 작업: 모든 AWS 리소스`ec2:CopyImage`에 대해.
- 작업: 모든 AWS 리소스`ec2:DescribeImages`에 대해.
- 작업: `cloudformation:DescribeStacks` 모든 AWS CloudFormation 스택에서.
- 작업: 모든 AWS 리소스`s3:GetAccountPublicAccessBlock`에 대해.

서비스 연결 역할 권한

IAM 엔터티(사용자, 그룹, 역할 등)가 서비스 연결 역할의 설명을 작성하거나 편집할 수 있도록 권한을 구성할 수 있습니다.

IAM 엔터티가 특정 서비스 연결 역할을 만들 수 있도록 허용하려면

서비스 연결 역할을 생성해야 하는 IAM 엔터티에 다음 정책을 추가합니다.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "iam:CreateServiceLinkedRole",
            "Resource": "arn:aws:iam::*:role/aws-service-role/
lightsail.amazonaws.com/AWSServiceRoleForLightsail*",
            "Condition": {"StringLike": {"iam:AWSServiceName":
 "lightsail.amazonaws.com"}}
        },
        {
            "Effect": "Allow",
            "Action": "iam:PutRolePolicy",
            "Resource": "arn:aws:iam::*:role/aws-service-role/
lightsail.amazonaws.com/AWSServiceRoleForLightsail*"
        }
    ]
}
```

IAM 엔터티가 서비스 연결 역할을 생성할 수 있도록 허용하려면

서비스 연결 역할 또는 필요한 정책을 포함해야 하는 모든 서비스 역할을 생성해야 하는 IAM 엔터티의 권한 정책에 다음 명령문을 추가합니다. 이 정책은 역할에 정책을 연결합니다.

```
{
    "Effect": "Allow",
    "Action": "iam:CreateServiceLinkedRole",
    "Resource": "arn:aws:iam::*:role/aws-service-role/*"
}
```

IAM 엔터티가 서비스 역할의 설명을 편집할 수 있도록 허용하려면

서비스 연결 역할 또는 서비스 역할의 설명을 편집해야 하는 IAM 엔터티의 권한 정책에 다음 명령문을 추가합니다.

```
{
    "Effect": "Allow",
    "Action": "iam:UpdateRoleDescription",
    "Resource": "arn:aws:iam::*:role/aws-service-role/*"
}
```

IAM 엔터티가 특정 서비스 연결 역할을 삭제하도록 허용하려면

서비스 연결 역할을 삭제해야 하는 IAM 개엔터티의 권한 정책에 다음 명령문을 추가합니다.

```
{
    "Effect": "Allow",
    "Action": [
        "iam:DeleteServiceLinkedRole",
        "iam:GetServiceLinkedRoleDeletionStatus"
    ],
    "Resource": "arn:aws:iam::*:role/aws-service-role/lightsail.amazonaws.com/
AWSServiceRoleForLightsail*"
}
```

IAM 엔터티가 모든 서비스 역할을 삭제하도록 허용하려면

서비스 연결 역할 또는 서비스 역할을 삭제해야 하는 IAM 엔터티의 권한 정책에 다음 명령문을 추가합니다.

```
{
    "Effect": "Allow",
    "Action": [
        "iam:DeleteServiceLinkedRole",
        "iam:GetServiceLinkedRoleDeletionStatus"
    ],
```

```
    "Resource": "arn:aws:iam::*:role/aws-service-role/*"
}
```

또는 AWS 관리형 정책을 사용하여 서비스에 대한 전체 액세스를 제공할 수 있습니다.

Amazon Lightsail에 대한 서비스 연결 역할 생성

서비스 연결 역할은 수동으로 생성할 필요가 없습니다. Lightsail 인스턴스 또는 블록 스토리지 디스크 스냅샷을 Amazon EC2로 내보내거나 AWS AWS Management Console, 또는 AWS API에서 Lightsail 버킷을 생성 AWS CLI하거나 업데이트하면가 서비스 연결 역할을 Amazon Lightsail 생성합니다.

이 서비스 연결 역할을 삭제한 다음 다시 생성해야 하는 경우 동일한 프로세스를 사용하여 계정에서 역할을 다시 생성할 수 있습니다. Lightsail 인스턴스 또는 블록 스토리지 디스크 스냅샷을 Amazon EC2로 내보내거나 Lightsail 버킷을 생성 또는 업데이트하면 Amazon Lightsail이 서비스 연결 역할을 다시 생성합니다.

> **Important**
>
> Amazon Lightsail이 서비스 연결 역할을 만들 수 있도록 IAM 권한을 구성해야 합니다. 이렇게 하려면 다음 서비스 연결 역할 권한 단원에 있는 단계를 완료하십시오.

Amazon Lightsail에 대한 서비스 연결 역할 편집

Amazon Lightsail 에서는 AWSServiceRoleForLightsail 서비스 연결 역할을 편집할 수 없습니다. 서비스 연결 역할을 생성한 후에는 다양한 개체가 역할을 참조할 수 있기 때문에 역할 이름을 변경할 수 없습니다. 하지만 IAM을 사용하여 역할의 설명을 편집할 수 있습니다. 자세한 내용은 IAM 사용 설명서의 서비스 링크 역할 편집을 참조하세요.

Amazon Lightsail에 대한 서비스 연결 역할 삭제

서비스 연결 역할이 필요한 기능 또는 서비스가 더 이상 필요 없는 경우에는 해당 역할을 삭제하는 것이 좋습니다. 따라서 적극적으로 모니터링하거나 유지하지 않는 미사용 엔터티가 없도록 합니다. 그러나 AWSServiceRoleForLightsail 서비스 연결 역할을 삭제하려면 먼저 보류 중인 복사 상태의 Amazon Lightsail 인스턴스 또는 디스크 스냅샷이 없는지 확인해야 합니다. 자세한 내용은 스냅샷을 Amazon EC2로 내보내기를 참조하세요.

IAM을 사용하여 수동으로 서비스 연결 역할을 삭제하려면 다음을 수행하세요.

IAM 콘솔 AWS CLI, 또는 AWS API를 사용하여 AWSServiceRoleForLightsail 서비스 연결 역할을 삭제합니다. 자세한 내용은 IAM 사용 설명서의 서비스 연결 역할 삭제를 참조하세요.

Amazon Lightsail 서비스 연결 역할을 지원하는 리전

Amazon Lightsail는 서비스가 제공되는 모든 리전에서 서비스 연결 역할을 사용하도록 지원합니다. Lightsail을 사용할 수 있는 리전에 대한 자세한 내용은 Amazon Lightsail 리전을 참조하세요.

IAM 정책을 사용하여 Lightsail 버킷 관리

다음 정책은 Amazon Lightsail 객체 스토리지 서비스의 특정 버킷을 관리할 수 있는 사용자 액세스 권한을 부여합니다. 이 정책은 Lightsail 콘솔, AWS Command Line Interface (AWS CLI), AWS API 및 AWS SDKs를 통해 버킷에 대한 액세스 권한을 부여합니다. 이 정책에서 *<BucketName>*을 관리할 버킷 이름으로 대체합니다. IAM 정책에 대한 자세한 내용은 AWS Identity and Access Management 사용 설명서의 IAM 정책 생성을 참조하세요. IAM 사용자 및 사용자 그룹 생성에 대한 자세한 내용은 AWS Identity and Access Management 사용 설명서의 첫 번째 IAM 위임 사용자 및 사용자 그룹 생성을 참조하세요.

> **Important**
>
> 이 정책을 보유하지 않은 사용자는 Lightsail 콘솔에서 버킷 관리 페이지의 객체(Objects) 탭을 볼 때 오류가 발생합니다.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "LightsailAccess",
            "Effect": "Allow",
            "Action": "lightsail:*",
            "Resource": "*"
        },
        {
            "Sid": "S3BucketAccess",
            "Effect": "Allow",
            "Action": "s3:*",
            "Resource": [
                "arn:aws:s3:::<BucketName>/*",
```

```
                "arn:aws:s3:::<BucketName>"
            ]
        }
    ]
}
```

버킷 및 객체 관리

다음은 Lightsail 객체 스토리지 버킷을 관리하는 일반적인 단계입니다.

1. Amazon Lightsail 객체 스토리지 서비스의 객체와 버킷에 대해 알아보세요. 자세한 내용은 Amazon Lightsail의 객체 스토리지를 참조하세요.
2. Amazon Lightsail에서 버킷에 지정할 수 있는 이름에 대해 알아봅니다. 자세한 내용은 Amazon Lightsail의 버킷 이름 지정 규칙을 참조하세요.
3. 버킷을 생성하여 Lightsail 객체 스토리지 서비스를 시작합니다. 자세한 내용은 Amazon Lightsail에서 버킷 생성을 참조하세요.
4. 버킷의 보안 모범 사례와 버킷에 구성할 수 있는 액세스 권한에 대해 알아봅니다. 버킷의 모든 객체를 공개 또는 비공개로 설정하거나 개별 객체를 공개로 설정하도록 선택할 수 있습니다. 액세스 키를 생성하고, 버킷에 인스턴스를 연결하고, 다른 AWS 계정에 액세스 권한을 부여하여 버킷에 대한 액세스 권한을 부여할 수도 있습니다. 자세한 내용은 Amazon Lightsail 객체 스토리지에 대한 보안 모범 사례 및 Amazon Lightsail의 버킷 권한 이해를 참조하세요.

   버킷 액세스 권한에 대해 알아본 후 다음 가이드를 참조하여 버킷에 대한 액세스 권한을 부여합니다.

   - Amazon Lightsail의 버킷에 대한 퍼블릭 액세스 차단
   - Amazon Lightsail에서 버킷 액세스 권한 구성
   - Amazon Lightsail에서 버킷의 개별 객체에 대한 액세스 권한 구성
   - Amazon Lightsail에서 버킷의 액세스 키 생성
   - Amazon Lightsail에서 버킷에 대한 리소스 액세스 구성
   - Amazon Lightsail에서 버킷의 크로스 계정 액세스 구성
5. 버킷에 대한 액세스 로깅을 활성화하는 방법과 액세스 로그를 사용하여 버킷의 보안을 감사하는 방법에 대해 알아봅니다. 자세한 내용은 다음 안내서를 참조하세요.
   - Amazon Lightsail 객체 스토리지 서비스의 버킷에 대한 액세스 로깅
   - Amazon Lightsail 객체 스토리지 서비스에서 버킷에 대한 액세스 로그 형식
   - Amazon Lightsail 객체 스토리지 서비스에서 버킷에 대한 액세스 로깅 활성화

- Amazon Lightsail에서 버킷에 대한 액세스 로그 사용하여 요청을 식별

6. Lightsail에서 버킷을 관리할 수 있는 기능을 사용자에게 부여하는 IAM 정책을 생성합니다. 자세한 내용은 Amazon Lightsail에서 버킷을 관리하기 위한 IAM 정책을 참조하세요.
7. 버킷의 객체에 라벨을 지정하고 식별하는 방식에 대해 알아봅니다. 자세한 내용은 Amazon Lightsail에서 객체 키 이름 이해를 참조하세요.
8. 파일을 업로드하고 버킷의 객체를 관리하는 방법에 대해 알아봅니다. 자세한 내용은 다음 안내서를 참조하세요.
   - Amazon Lightsail에서 버킷으로 파일 업로드
   - 멀티파트 업로드를 사용하여 Amazon Lightsail에서 버킷으로 파일 업로드
   - Amazon Lightsail에서 버킷의 객체 확인
   - Amazon Lightsail에서 버킷의 객체 복사 또는 이동
   - Amazon Lightsail에서 버킷의 객체 다운로드
   - Amazon Lightsail에서 버킷의 객체 필터링
   - Amazon Lightsail에서 버킷의 객체 태그 지정
   - Amazon Lightsail에서 버킷의 객체 삭제
9. 객체 버전 관리를 활성화하면 버킷에 저장된 모든 버전의 객체를 모두 보존, 검색 및 복원할 수 있습니다. 자세한 내용은 Amazon Lightsail에서 버킷의 객체 버전 사용 설정 및 사용 중지를 참조하세요.
10. 객체 버전 관리를 활성화한 후 버킷에 있는 객체의 이전 버전을 복원할 수 있습니다. 자세한 내용은 Amazon Lightsail에서 버킷의 이전 객체 버전 복원을 참조하세요.
11. 버킷 사용률을 모니터링합니다. 자세한 내용은 Amazon Lightsail에서 버킷에 대한 지표 보기를 참조하세요.
12. 버킷 사용률이 임곗값을 초과할 때 알림을 받도록 버킷 지표에 대한 경보를 구성합니다. 자세한 내용은 Amazon Lightsail에서 버킷 지표 경보 생성을 참조하세요.
13. 스토리지 및 네트워크 전송이 부족한 경우 버킷의 스토리지 플랜을 변경합니다. 자세한 내용은 Amazon Lightsail에서 버킷 플랜 변경을 참조하세요.
14. 버킷을 다른 리소스에 연결하는 방법에 대해 알아봅니다. 자세한 내용은 다음 자습서를 참조하세요.
    - 자습서: WordPress 인스턴스를 Amazon Lightsail 버킷에 연결
    - 자습서: Lightsail 콘텐츠 전송 네트워크 배포와 함께 Amazon Lightsail 버킷 사용
15. 버킷을 더 이상 사용하지 않는 경우 삭제할 수 있습니다. 자세한 내용은 Amazon Lightsail에서 버킷 삭제를 참조하세요.

IAM 사용자에게 Lightsail 액세스 권한 부여

AWS 계정 루트 사용자 또는 관리자 액세스 권한이 있는 AWS Identity and Access Management (IAM) 사용자는 AWS 계정에 하나 이상의 IAM 사용자를 생성할 수 있으며, 해당 사용자는에서 제공하는 서비스에 대한 다양한 수준의 액세스 권한으로 구성될 수 있습니다 AWS.

Amazon Lightsail의 경우 Lightsail 서비스에만 액세스할 수 있는 IAM 사용자를 생성하려고 할 수 있습니다. Lightsail 리소스 보기, 생성, 편집 또는 삭제에 대한 액세스 권한이 필요하지만에서 제공하는 다른 서비스에 대한 액세스 권한이 필요하지 않은 팀원이 팀에 합류하면이 작업을 수행할 수 있습니다 AWS. 이를 구성하려면 먼저 Lightsail에 대한 액세스를 허용한 IAM 정책을 생성한 다음 IAM 그룹을 생성하고 해당 정책을 연결해야 합니다. 그런 다음 IAM 사용자를 생성하여 그룹의 멤버로 만들면 Lightsail에 대한 액세스 권한을 부여할 수 있습니다.

다른 사람이 팀에서 나가면 Lightsail 액세스 그룹에서 해당 사용자를 제거하여 Lightsail에 대한 액세스 권한을 취소할 수 있습니다(예: 팀에서 나가지만 여전히 회사에서 근무하는 경우). 또는 예를 들어 해당 사용자가 퇴사하여 다시 액세스할 필요가 없는 경우 IAM에서 사용자를 삭제할 수 있습니다.

> **Warning**
>
> 이 시나리오에서는 프로그래밍 방식 액세스 권한과 장기 보안 인증이 있는 IAM 사용자가 필요하며 이는 보안 위험을 내포합니다. 이 위험을 줄이려면 이러한 사용자에게 작업을 수행하는 데 필요한 권한만 제공하고 더 이상 필요하지 않을 경우 이러한 사용자를 제거하는 것이 좋습니다. 필요한 경우 액세스 키를 업데이트할 수 있습니다. 자세한 내용은 IAM 사용 설명서의 액세스 키 업데이트를 참조하세요.

목차

- Lightsail 액세스를 위한 IAM 정책 생성
- Lightsail 액세스에 대한 IAM 그룹 생성 및 Lightsail 액세스 정책 연결
- IAM 사용자를 생성한 후 Lightsail 액세스 그룹에 추가

Lightsail 액세스를 위한 IAM 정책 생성

다음 단계에 따라 Lightsail 액세스에 대한 IAM 정책을 생성합니다. 자세한 내용은 IAM 설명서의 IAM 정책 생성을 참조하십시오.

1. IAM 콘솔에 로그인합니다.

2. 왼쪽 탐색 창에서 Policies(정책)를 선택합니다.
3. 정책 생성을 선택하세요.
4. Create Policy(정책 생성) 페이지에서 JSON 탭을 선택합니다.

5. 텍스트 상자의 내용을 강조 표시한 후 다음 정책 구성 텍스트를 복사하여 붙여넣습니다.

   JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "lightsail:*"
            ],
            "Resource": "*"
        }
    ]
}
```

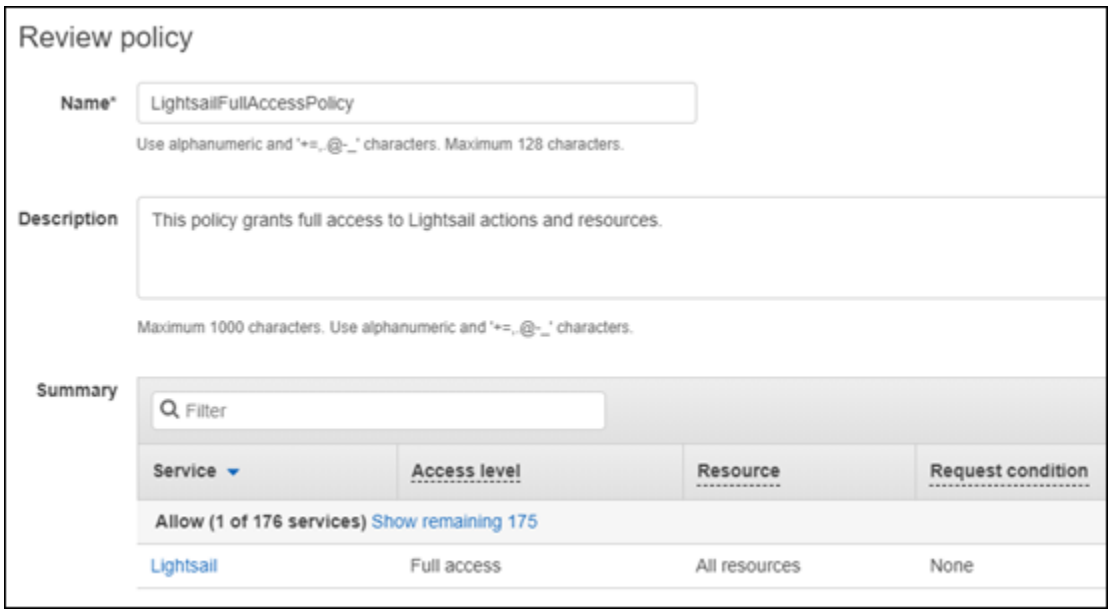
   결과는 다음 예제와 같아야 합니다.

이렇게 하면 모든 Lightsail 작업과 리소스에 대한 액세스가 허용됩니다. VPC 피어링 활성화 AWS, Amazon EC2로 Lightsail 스냅샷 내보내기,를 사용하여 Amazon EC2 리소스 생성 등에서 제공하는 다른 서비스에 액세스해야 하는 작업에는이 정책에 포함되지 않은 추가 권한이 Lightsail필요합니다. 자세한 내용은 다음 안내서를 참조하세요.

- 외부 AWS 리소스와 함께 작동하도록 Amazon VPC 피어링 설정 Amazon Lightsail
- Amazon EC2로 Amazon Lightsail 스냅샷 내보내기
- Lightsail의 내보낸 스냅샷에서 Amazon EC2 인스턴스 생성

작업별 및 리소스별로 부여할 수 있는 권한의 예는 Amazon Lightsail 리소스 수준 권한 정책 예를 참조하세요.

6. 정책 검토를 선택합니다.
7. Review Policy(정책 검토) 페이지에서 정책의 이름을 지정합니다. `LightsailFullAccessPolicy`처럼 설명하는 이름을 지정합니다.
8. 설명을 추가하고 정책 설정을 검토합니다. 변경해야 하는 경우 Previous(이전)를 선택하여 정책을 변경합니다.



9. 정책 설정이 올바른지 확인한 후 Create Policy(정책 생성)를 선택합니다.

   이제 정책이 생성되어 기존 IAM 그룹에 추가하거나 이 설명서의 다음 섹션에 있는 단계를 사용하여 새 IAM 그룹을 생성할 수 있습니다.

Lightsail 액세스에 대한 IAM 그룹을 생성하고 Lightsail 액세스 정책을 연결합니다.

다음 단계에 따라 Lightsail 액세스에 대한 IAM 그룹을 생성한 다음 이 설명서의 이전 섹션에서 생성한 Lightsail 액세스 정책을 연결합니다. 자세한 내용은 IAM 설명서의 IAM 그룹 생성 및 IAM 그룹에 정책 연결을 참조하세요.

1. IAM 콘솔의 왼쪽 탐색 창에서 그룹을 선택합니다.
2. 새 그룹 생성을 선택합니다.
3. Set Group Name(그룹 이름 설정) 페이지에서 그룹 이름을 지정합니다. `LightsailFullAccessGroup`처럼 설명하는 이름을 지정합니다.
4. Attach Policy(정책 연결) 페이지에서 이 안내서의 앞부분에서 생성한 Lightsail 정책을 검색합니다(예: `LightsailFullAccessPolicy`).
5. 정책 옆에 확인 표시를 추가하고 Next step(다음 단계)을 선택합니다.
6. 그룹 설정을 검토합니다. 변경해야 하는 경우 Previous(이전)를 선택하여 그룹 정책을 변경합니다.
7. 그룹 설정이 올바른지 확인한 후 Create Group(그룹 생성)을 선택합니다.

이제 그룹이 생성되고 그룹에 추가된 사용자는 Lightsail 작업 및 리소스에 액세스할 수 있습니다. 기존 IAM 사용자를 그룹에 추가하거나 이 설명서의 다음 섹션에 있는 단계를 사용하여 새 IAM 사용자를 생성할 수 있습니다.

IAM 사용자를 생성한 후 Lightsail 액세스 그룹에 추가

다음 단계에 따라 IAM 사용자를 생성한 후 Lightsail 액세스 그룹에 사용자를 추가합니다. 자세한 내용은 IAM 설명서의 AWS 계정의 IAM 사용자 생성 및 IAM 그룹에서 사용자 추가 및 제거를 참조하세요.

1. IAM 콘솔의 왼쪽 탐색 창에서 사용자를 선택합니다.
2. 사용자 추가를 선택합니다.
3. 페이지의 Set user details(사용자 세부 정보 설정) 섹션에서 사용자의 이름을 지정합니다.
4. 페이지의 AWS 액세스 유형 선택 섹션에서 다음 옵션 중에서 선택합니다.

   a. 프로그래밍 방식 액세스를 선택하여 Lightsail 작업 및 리소스에 사용할 수 있는 AWS API, CLI, SDK 및 기타 개발 도구에 대한 액세스 키 ID와 보안 액세스 키를 활성화합니다. 자세한 내용은 에서 작동 AWS CLI 하도록 구성을 Lightsail참조하세요.

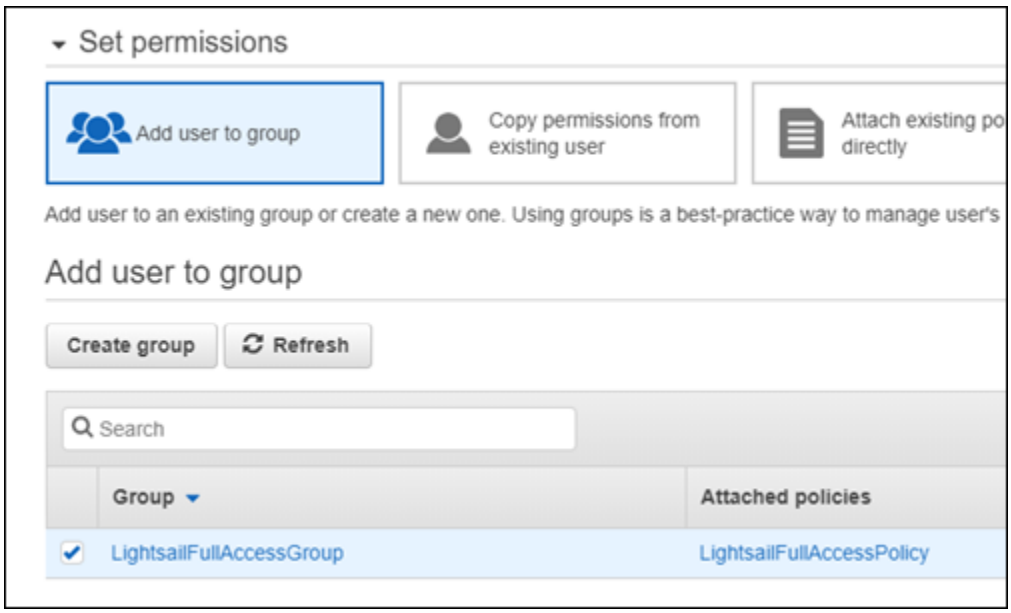
   b. AWS 관리 콘솔 액세스를 선택하여 사용자가 AWS 관리 콘솔에 로그인하여 Lightsail 콘솔에 로그인할 수 있는 암호를 활성화합니다. 이 옵션을 선택하면 다음과 같은 암호 옵션이 나타납니다.

      i. 자동 생성 암호를 선택하여 IAM에서 암호를 생성하게 하거나 사용자 지정 암호를 선택하여 자체 암호를 입력합니다.

      ii. Require password reset(암호 재설정 필요)을 선택하여 다음 로그인 시 사용자가 새 암호를 생성하도록 합니다(암호 재설정).

   > **Note**
   >
   > 프로그래밍 방식 액세스 옵션만 선택하면 사용자가 AWS 콘솔 및 Lightsail 콘솔에 로그인할 수 없습니다.

5. 다음: 권한을 선택합니다.
6. 페이지의 Set permissions(권한 설정) 섹션에서 Add user to group(사용자를 그룹에 추가)을 선택한 다음 이 안내서 앞부분에서 생성한 Lightsail 액세스 그룹을 선택합니다(예: `LightsailFullAccessGroup`).



7. 다음: 태그를 선택합니다.
8. (선택 사항) 태그를 키 값 페어로 연결하여 메타데이터를 사용자에게 추가합니다. IAM에서 태그 사용에 대한 자세한 내용은 IAM 엔터티 태그 지정을 참조하세요.
9. 다음: 검토를 선택합니다.
10. 사용자 설정을 검토합니다. 변경해야 하는 경우 Previous(이전)를 선택하여 사용자의 그룹 또는 정책을 변경합니다.
11. 사용자 설정이 올바른지 확인한 후 Create user(사용자 생성)를 선택합니다.

    사용자가 생성되며 이제 Lightsail에 액세스할 수 있습니다. 사용자의 Lightsail 액세스 권한을 취소하려면 Lightsail 액세스 그룹에서 사용자를 제거합니다. 자세한 내용은 IAM 설명서의 IAM 그룹에 사용자 추가 및 제거를 참조하세요.
12. 사용자의 자격 증명을 가져오려면 다음 옵션을 선택합니다.

    a. .csv 다운로드를 선택하여 사용자 이름, 암호, 액세스 키 ID, 보안 액세스 키 및 계정의 AWS 콘솔 로그인 링크가 포함된 파일을 다운로드합니다.

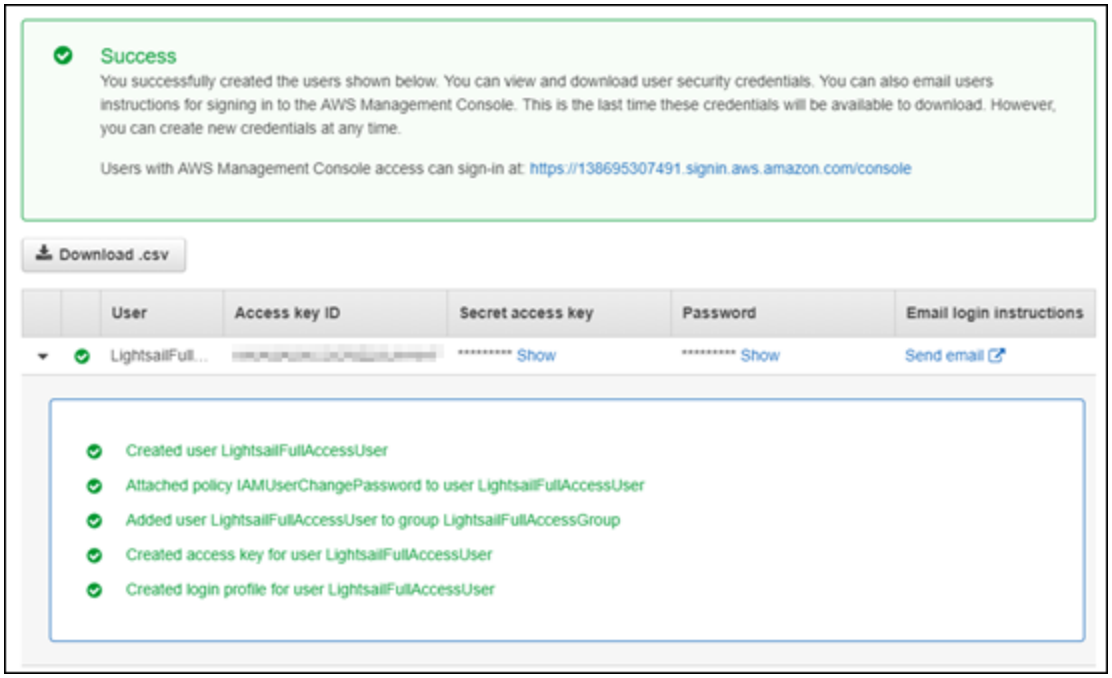
    b. 비밀 액세스 키 아래에서 표시를 선택하여 AWS API, CLI, SDK, 기타 개발자 도구를 통해 Lightsail에 프로그래밍 방식으로 액세스하는 데 사용할 수 있는 액세스 키를 확인합니다.

    > **Important**
    >
    > 이는 보안 액세스 키를 보거나 다운로드할 수 있는 유일한 기회이며 사용자가 AWS API를 사용하기 전에이 정보를 제공해야 합니다. 사용자의 새 액세스 키 ID와 비밀 액

세스 키를 안전한 장소에 보관하세요. 이 단계 이후에는 보안 키에 다시 액세스할 수 없습니다.

c. 암호 아래에서 표시를 선택하여 IAM에서 생성한 사용자 암호를 확인합니다. 사용자가 처음 로그인할 수 있도록 암호를 제공해야 합니다.

d. Send email(이메일 전송)을 선택하여 사용자에게 이메일을 보내 이제 Lightsail에 대한 액세스 권한이 있다고 알려 줍니다.



업데이트 관리를 통해 Lightsail 인스턴스 및 컨테이너 보안 유지

Amazon Web Services(AWS), Amazon Lightsail및 타사 애플리케이션 공급업체는에서 사용할 수 있는 인스턴스 이미지(블루프린트라고도 함)를 주기적으로 업데이트하고 패치합니다Lightsail. AWS 또한 인스턴스를 생성한 후에는 인스턴스에서 운영 체제 또는 애플리케이션을 업데이트하거나 패치하지 Lightsail 않습니다. Lightsail 또한는 Lightsail 컨테이너 서비스에서 구성한 운영 체제 및 소프트웨어를 업데이트하거나 패치하지 않습니다. 따라서 Amazon Lightsail 인스턴스 및 컨테이너 서비스의 운영 체제와 애플리케이션을 정기적으로 패치, 업데이트 및 보호하는 것이 좋습니다. 자세한 내용은 AWS 공동 책임 모델을 참조하세요.

인스턴스 블루프린트 소프트웨어 지원

다음 Amazon Lightsail 플랫폼 및 청사진 목록은 각 공급업체의 지원 페이지로 연결됩니다. 여기에서 사용 방법 가이드, 운영 체제 및 애플리케이션을 최신 상태로 유지하는 등의 정보를 볼 수 있습니다. 자

동 업데이트 서비스나 애플리케이션 공급업체에서 제공하는 업데이트 설치 권장 프로세스를 사용할 수 있습니다.

Windows

- Windows Server 2022, Windows Server 2019 및 Windows Server 2016
- Microsoft SQL Server

Linux 및 Unix - 운영 체제만

- Amazon Linux 2023
- Amazon Linux 2
- Ubuntu
- Debian
- FreeBSD
- openSUSE
- CentOS

Linux 및 Unix - 운영 체제 및 애플리케이션

- Ubuntu의 Plesk 호스팅 스택
- 리눅스용 cPanel & WHM
- WordPress
- WordPress 멀티사이트
- LAMP(PHP 8)
- Node.js
- Joomla!
- Magento
- MEAN
- Drupal
- GitLab CE
- Redmine
- Nginx

- Ghost
- Django
- PrestaShop

Amazon Lightsail 리소스에 대한 규정 준수 검증

AWS 는 규정 준수에 도움이 되는 다음 리소스를 제공합니다.

- 보안 및 규정 준수 빠른 시작 안내서 -이 배포 안내서에서는 아키텍처 고려 사항에 대해 설명하고 보안 및 규정 준수 중심 기준 환경을 배포하기 위한 단계를 제공합니다 AWS.
- AWS 규정 준수 리소스 -이 워크북 및 가이드 모음은 업계 및 위치에 적용될 수 있습니다.
- AWS Config 개발자 안내서의 규칙을 사용하여 리소스 평가 -이 AWS Config 서비스는 리소스 구성이 내부 관행, 업계 지침 및 규정을 얼마나 잘 준수하는지 평가합니다.
- AWS Security Hub CSPM -이 AWS 서비스는 보안 업계 표준 및 모범 사례 준수 여부를 확인하는 데 도움이 AWS 되는 내 보안 상태에 대한 포괄적인 보기를 제공합니다.

인터페이스 엔드포인트(AWS PrivateLink)를 사용하여 Amazon Lightsail에 액세스

AWS PrivateLink를 사용하여 VPC와 Amazon Lightsail 사이에 프라이빗 연결을 생성할 수 있습니다. 인터넷 게이트웨이, NAT 디바이스, VPN 연결 또는 Direct Connect 연결을 사용하지 않고 VPC에 있는 것처럼 Amazon Lightsail에 액세스할 수 있습니다. VPC의 인스턴스에서 Amazon Lightsail API에 액세스하는 데는 퍼블릭 IP 주소가 필요하지 않습니다.

AWS PrivateLink에서 제공되는 인터페이스 엔드포인트를 생성하여 이 프라이빗 연결을 설정합니다. 인터페이스 엔드포인트에 대해 사용 설정하는 각 서브넷에서 엔드포인트 네트워크 인터페이스를 생성합니다. 이는 Amazon Lightsail로 향하는 트래픽의 진입점 역할을 하는 요청자 관리형 네트워크 인터페이스입니다.

자세한 내용은 AWS PrivateLink 가이드의 AWS PrivateLink를 통해 AWS 서비스에 액세스를 참조하세요.

Amazon Lightsail에 대한 고려 사항

Amazon Lightsail에 대한 인터페이스 엔드포인트를 설정하기 전에 먼저 가상 프라이빗 클라우드(VPC)를 생성해야 합니다. 자세한 내용은 Amazon Virtual Private Cloud 사용 설명서의 VPC 생성을 참조하세요. 또한 AWS PrivateLink 안내서의 고려 사항을 검토하세요.

Amazon Lightsail에서는 인터페이스 엔드포인트를 통해 모든 API 작업에 대한 호출 수행을 지원합니다. Lightsail에 사용 가능한 API 작업에 대한 자세한 내용은 Amazon Lightsail API 레퍼런스를 참조하세요.

Amazon Lightsail용 인터페이스 엔드포인트 생성

Amazon VPC 콘솔 또는 AWS Command Line Interface(AWS CLI)를 사용하여 Amazon Lightsail에 대한 인터페이스 엔드포인트를 생성할 수 있습니다. 자세한 내용은 AWS PrivateLink 설명서의 인터페이스 엔드포인트 생성을 참조하세요.

다음 서비스 이름을 사용하여 Amazon Lightsail에 대한 엔드포인트를 생성합니다.

```
com.amazonaws.region.lightsail
```

인터페이스 엔드포인트에 프라이빗 DNS를 사용하도록 설정하는 경우, 리전에 대한 기본 DNS 이름(예: Amazon Lightsail)을 사용하여 에 API 요청을 할 수 있습니다. 예를 들어 `lightsail.us-east-1.amazonaws.com`입니다. 사용할 수 있는 리전 코드는 Lightsail의 리전 및 가용 영역 섹션을 참조하세요.

AWS CLI 예제

인터페이스 엔드포인트를 사용하여 Lightsail에 액세스하려면 AWS CLI 명령에 `--region` 및 `--endpoint-url` 파라미터를 사용합니다. Lightsail에서 수행할 수 있는 작업의 목록은 Amazon Lightsail API 레퍼런스의 작업을 참조하세요.

다음 예시에서 AWS 리전 *us-east-1* 및 VPC 엔드포인트 ID의 DNS 이름(*vpce-1a2b3c4d-5e6f.s3.us-east-1.vpce.amazonaws.com*)을 실제 정보로 바꿉니다.

예: 엔드포인트 URL을 사용하여 Lightsail 인스턴스 나열

다음 예시에서는 인터페이스 엔드포인트를 사용하여 인스턴스를 나열합니다.

```
aws lightsail get-instances --region us-east-1 --endpoint-url
 https://vpce-1a2b3c4d-5e6f.lightsail.us-east-1.vpce.amazonaws.com
```

예: 엔드포인트 URL을 사용하여 Lightsail 디스크 나열

다음 예시에서는 인터페이스 엔드포인트를 사용하여 디스크를 나열합니다.

```
aws lightsail get-disks --region us-east-1 --endpoint-url
 https://vpce-1a2b3c4d-5e6f.lightsail.us-east-1.vpce.amazonaws.com
```

엔드포인트의 엔드포인트 정책 생성

엔드포인트 정책은 인터페이스 엔드포인트에 연결할 수 있는 IAM 리소스입니다. 기본 엔드포인트 정책을 사용하면 인터페이스 엔드포인트를 통해 Amazon LightsailI에 대한 전체 액세스를 허용합니다. VPC에서 Amazon Lightsail에 허용되는 액세스를 제어하려면 사용자 지정 엔드포인트 정책을 인터페이스 엔드포인트에 연결합니다.

엔드포인트 정책은 다음 정보를 지정합니다.

- 작업을 수행할 수 있는 보안 주체(AWS 계정, IAM 사용자, IAM 역할)
- 수행할 수 있는 작업.
- 작업을 수행할 수 있는 리소스.

자세한 정보는 AWS PrivateLink 가이드의 엔드포인트 정책을 사용하여 서비스에 대한 액세스 제어를 참조하세요.

예제: Amazon Lightsail 작업에 대한 VPC 엔드포인트 정책

다음은 사용자 지정 엔드포인트 정책의 예입니다. 이 정책을 인터페이스 엔드포인트에 연결하면 Lightsail에서 모든 사용자에게 엔드포인트를 통해 블록 스토리지 디스크를 삭제할 수 있는 권한을 거부하고 모든 사용자에게 다른 모든 Lightsail 작업을 수행할 수 있는 권한을 부여합니다.

```
{
  "Statement": [
    {
      "Action": "lightsail:*",
      "Effect": "Allow",
      "Principal": "*",
      "Resource": "*"
    },
    {
      "Action": "lightsail:DeleteDisk",
      "Effect": "Deny",
```

```
      "Principal": "*",
      "Resource": "*"
    }
  ]
}
```

Lightsail 리소스 지표 모니터링

지표 데이터를 확인하고 수집하여 Amazon Lightsail에서 인스턴스, 데이터베이스, 배포, 로드 밸런서, 컨테이너 서비스 및 버킷의 성능을 모니터링할 수 있습니다. 시간 경과에 따른 기준을 설정하여 경보를 구성하면 리소스 성능과 관련된 이상 및 문제를 보다 쉽게 탐지할 수 있습니다.

Amazon Lightsail에서는 인스턴스, 데이터베이스, 콘텐츠 전송 네트워크(CDN) 배포, 로드 밸런서, 컨테이너 서비스 및 버킷에 대한 지표 데이터를 보고합니다. Lightsail 콘솔에서 이 데이터를 보고 모니터링할 수 있습니다. 모니터링은 리소스의 안정성, 가용성 및 성능을 유지하는 데 중요한 부분입니다. 리소스에서 지표 데이터를 정기적으로 모니터링하고 수집하면 다중 지점 오류가 발생할 경우 보다 쉽게 디버깅할 수 있습니다.

목차

- 효과적으로 리소스 모니터링
- 지표 개념 및 용어
- Lightsail에서 사용할 수 있는 지표

효과적으로 리소스 모니터링

사용자 환경에서 정상적인 리소스 성능에 대한 기준을 설정해야 합니다. 다양한 시간과 다양한 부하 조건에서 성능을 측정합니다. 리소스를 모니터링할 때 시간 경과에 따른 리소스 성능 기록을 작성해 두어야 합니다. 리소스의 현재 성능을 수집한 기록 데이터와 비교합니다. 이를 통해 정상적인 성능 패턴과 성능 이상을 식별하고 해결 방법을 고안할 수 있습니다.

예를 들어 인스턴스에 대해 CPU 사용률, 네트워크 사용률 및 상태 확인을 모니터링할 수 있습니다. 설정한 기준 이하로 성능이 떨어지면 인스턴스를 재구성하거나 최적화하여 CPU 사용률을 줄이거나 네트워크 트래픽을 줄일 수 있습니다. 인스턴스가 지속적으로 CPU 사용률 임계값을 초과하여 작동하는 경우 인스턴스 플랜을 더 큰 플랜으로 전환하는 것이 좋을 수 있습니다(7 USD/월 플랜 대신 5 USD/월 플랜 사용). 인스턴스의 새 스냅샷을 생성한 다음 더 큰 플랜을 사용해 스냅샷에서 새 인스턴스를 생성하여 더 큰 플랜으로 전환할 수 있습니다.

기준을 설정한 후에는 리소스가 지정된 임계값을 벗어날 때 알림을 받도록 Lightsail 콘솔에서 경보를 구성할 수 있습니다. 자세한 내용은 알림 및 경보를 참조하세요.

지표 개념 및 용어

다음 용어와 개념은 Lightsail에서 지표의 사용을 보다 정확하게 이해하는 데 도움이 됩니다.

Metrics

지표는 시간 순서별 데이터 포인트 집합을 나타냅니다. 지표를 모니터링할 변수로 생각하면 데이터 포인트는 시간에 따른 변수의 값을 나타냅니다. 지표는 이름으로 고유하게 정의됩니다. 예를 들어 Lightsail에서 제공하는 일부 인스턴스 지표에는 CPU 사용률(`CPUUtilization`), 수신 네트워크 트래픽(`NetworkIn`) 및 송신 네트워크 트래픽(`NetworkOut`)이 포함됩니다. Lightsail에서 사용 가능한 모든 리소스 지표에 대한 자세한 내용은 Lightsail에서 사용할 수 있는 지표를 참조하십시오.

지표 보존 기간

기간이 60초(1분 분해능)로 설정된 데이터 포인트는 15일 동안 사용이 가능합니다. 기간이 300초(5분 분해능)로 설정된 데이터 포인트는 63일 동안 사용이 가능합니다. 기간이 3,600초(1시간 분해능)로 설정된 데이터 포인트는 455일(15개월) 동안 사용이 가능합니다.

원래 더 짧은 기간 동안 사용 가능한 데이터 포인트는 장기 보관을 위해 집계됩니다. 예를 들어 세부 수준이 1분인 데이터 포인트는 1분 분해능으로 15일 동안 사용할 수 있습니다. 15일 이후에는 이 데이터를 계속 사용할 수 있지만 데이터가 5분 분해능으로 집계됩니다. 63일 이후에는 이 데이터가 추가로 집계되어 1시간 분해능으로 제공됩니다. 이러한 기간보다 긴 지표 가용성이 필요한 경우 Lightsail API, AWS Command Line Interface (AWS CLI) 및 SDKs를 사용하여 오프라인 또는 다른 스토리지의 데이터 포인트를 검색할 수 있습니다.

자세한 내용은 Lightsail API 참조의 GetInstanceMetricData, GetBucketMetricData, GetLoadBalancerMetricData, GetDistributionMetricData, GetRelationalDatabaseMetricData를 참조하세요.

통계

지표 통계는 일정 기간에 걸쳐 데이터를 집계하는 데 사용됩니다. 통계의 예로는 `Average`, `Sum` 및 `Maximum`이 있습니다. 예를 들어, `Average` 통계를 사용하여 인스턴스 CPU 사용률 지표 데이터의 평균을 구할 수 있고, `Sum` 통계를 사용하여 데이터베이스 연결을 추가할 수 있으며, `Maximum` 통계를 사용하여 최대 로드 밸런서 응답 시간을 검색할 수 있습니다.

사용 가능한 지표 통계 목록은 Lightsail API 참조의 GetInstanceMetricData 통계, GetBucketMetricData 통계, GetLoadBalancerMetricData 통계, GetDistributionMetricData 통계, GetRelationalDatabaseMetricData 통계를 참조하세요.

단위

각각의 통계는 측정 단위를 가지고 있습니다. 단위로는 `Bytes`, `Seconds`, `Count` 및 `Percent`가 있습니다. 전체 단위 목록을 알아보려면 Lightsail API 참조의 GetInstanceMetricData의 단위, GetLoadBalancerMetricData의 단위, GetDistributionMetricData의 단위 및 GetRelationalDatabaseMetricData의 단위를 참조하세요.

기간

기간은 특정 데이터 포인트와 관련된 시간의 길이로, 반환된 데이터 포인트의 세밀도 수준을 나타냅니다. 각 데이터 포인트는 지정한 기간에 걸쳐 수집된 지표 데이터의 집계를 나타냅니다. 기간은 초 단위로 정의되며 기간의 유효한 값은 60초(1분) 및 300초(5분)의 배수입니다.

Lightsail API를 사용하여 데이터 포인트를 검색하는 경우 기간, 시작 시간 및 종료 시간을 지정할 수 있습니다. 이러한 파라미터는 데이터 포인트와 관련된 전체 시간을 결정합니다. Lightsail는 지표 데이터를 1분 또는 5분 단위로 보고합니다. 따라서 기간을 60초와 300초의 배수로 지정해야 합니다. 시작 시간과 종료 시간에 값을 지정하여 Lightsail에서 값이 반환되는 기간의 수를 결정할 수 있습니다. 10분 단위로 집계된 통계를 선호할 경우에는 기간을 600으로 지정합니다. 전체 시간 동안 통계를 집계하고 싶은 경우에는 기간을 3,600 등으로 지정합니다.

Lightsail 경보와 관련된 기간도 중요합니다. Lightsail는 경보 여부를 판단하고자 5분마다 데이터 포인트를 평가하며, 경보용 각 데이터 포인트는 5분간 집계된 데이터를 나타냅니다. 특정 지표를 모니터링하도록 경보를 생성하면 Lightsail이 해당 지표를 지정된 임계값과 비교하게 됩니다. Lightsail에서 비교하는 방식을 광범위하게 제어할 수 있습니다. 비교 작업이 수행되는 기간을 지정할 수 있을 뿐 아니라, 결론에 도달하기까지 사용되는 평가 기간의 수를 지정할 수 있습니다. 자세한 내용은 경보 섹션을 참조하세요.

경보

경보는 지정된 기간 동안 단일 지표를 감시하고 지표가 지정한 임계값을 초과할 때 사용자에게 알립니다. 알림은 Lightsail 콘솔에 표시되는 배너, 지정한 이메일 주소로 전송되는 이메일 및 지정한 휴대폰 번호로 전송되는 SMS 문자 메시지 형태로 사용할 수 있습니다. 자세한 내용은 경보 섹션을 참조하세요.

Lightsail에서 사용할 수 있는 지표

인스턴스 지표

다음과 같은 인스턴스 지표를 사용할 수 있습니다. 자세한 내용은 Amazon Lightsail의 인스턴스 지표 보기를 참조하세요.

- CPU 사용률(**CPUUtilization**) - 인스턴스에서 현재 사용 중인 할당된 컴퓨팅 유닛의 비율(%)입니다. 이 지표는 인스턴스에서 애플리케이션을 실행하는 처리 능력을 식별합니다. 인스턴스에 전체 프로세스 코어가 할당되지 않았을 때 운영 체제의 도구에서 비율이 Lightsail보다 낮게 표시할 수 있습니다.

  Lightsail 콘솔에서 인스턴스의 CPU 사용률 지표 그래프를 보면 지속 가능 영역과 버스트 가능 영역이 있습니다. 이러한 영역의 의미에 대한 자세한 내용은 CPU 사용률 지속 가능 및 버스트 가능 영역을 참조하십시오.
- 버스트 용량(분)(**BurstCapacityTime**) 및 백분율(**BurstCapacityPercentage**) - 버스트 용량(분)은 인스턴스가 100% CPU 사용률로 버스트할 수 있는 시간을 나타냅니다. 버스트 용량 백분율은 인스턴스에 사용할 수 있는 CPU 성능의 백분율입니다. 인스턴스는 지속적으로 버스트 용량을 소비하고 누적합니다. 버스트 용량(분)은 인스턴스가 100% CPU 사용률로 작동하는 경우에만 전체 비율로 소비됩니다. 인스턴스 버스트 용량에 대한 자세한 내용은 Amazon Lightsail에서 인스턴스 버스트 용량 보기를 참조하십시오.
- 수신 네트워크 트래픽(**NetworkIn**) - 인스턴스가 모든 네트워크 인터페이스에서 수신한 바이트 수입니다. 이 지표는 단일 인스턴스로 들어오는 네트워크 트래픽의 볼륨을 식별합니다. 보고된 숫자는 해당 기간에 수신된 바이트 수입니다. 이 지표는 5분 간격으로 보고되므로 보고된 숫자를 300으로 나누어 초당 바이트 수를 구합니다.
- 송신 네트워크 트래픽(**NetworkOut**) - 인스턴스가 모든 네트워크 인터페이스에서 보낸 바이트 수입니다. 이 지표는 단일 인스턴스에서 나가는 네트워크 트래픽의 볼륨을 식별합니다. 보고된 숫자는 해당 기간에 전송된 바이트 수입니다. 이 지표는 5분 간격으로 보고되므로 보고된 숫자를 300으로 나누어 초당 바이트 수를 구합니다.
- 상태 확인 실패(**StatusCheckFailed**) - 인스턴스가 인스턴스 상태 확인 및 시스템 상태 확인을 통과했는지 여부를 보고합니다. 이 지표는 0(통과) 또는 1(실패)이 될 수 있습니다. 이 지표는 1분 간격으로 제공됩니다.
- 인스턴스 상태 확인 실패(**StatusCheckFailed_Instance**) - 인스턴스가 인스턴스 상태 확인을 통과했는지 여부를 보고합니다. 이 지표는 0(통과) 또는 1(실패)이 될 수 있습니다. 이 지표는 1분 간격으로 제공됩니다.

- 시스템 상태 확인 실패(**StatusCheckFailed_System**) - 인스턴스가 시스템 상태 확인을 통과했는지 여부를 보고합니다. 이 지표는 0(통과) 또는 1(실패)이 될 수 있습니다. 이 지표는 1분 간격으로 제공됩니다.
- 토큰 메타데이터 요청 없음(**MetadataNoToken**) - 토큰 없이 인스턴스 메타데이터 서비스에 성공적으로 액세스한 횟수입니다. 이 메트릭은 토큰을 사용하지 않는 인스턴스 메타데이터 서비스 버전 1을 사용하여 인스턴스 메타데이터에 액세스하는 프로세스가 있는지 확인합니다. 모든 요청이 인스턴스 메타데이터 서비스 버전 2와 같은 토큰 지원 세션을 사용하는 경우 값은 0입니다. 자세한 내용은 Amazon Lightsail의 인스턴스 메타데이터 및 사용자 데이터를 참조하세요.

데이터베이스 지표

다음과 같은 데이터베이스 지표를 사용할 수 있습니다. 자세한 내용은 Amazon Lightsail의 데이터베이스 지표 보기를 참조하십시오.

- CPU 사용률(**CPUUtilization**) - 데이터베이스에서 현재 사용 중인 CPU 사용률입니다.
- 데이터베이스 연결(**DatabaseConnections**) - 사용 중인 데이터베이스 연결 수입니다.
- 디스크 대기열 깊이(**DiskQueueDepth**) - 디스크에 액세스하기 위해 대기 중인 IO(읽기/쓰기 요청) 수입니다.
- 여유 스토리지 공간(**FreeStorageSpace**) - 사용 가능한 스토리지 공간 크기입니다.
- 네트워크 수신 처리량(**NetworkReceiveThroughput**) - 모니터링 및 복제에 사용되는 고객 데이터베이스 트래픽과 AWS 트래픽을 모두 포함하여 데이터베이스에 수신되는 네트워크 트래픽입니다.
- 네트워크 송신 처리량(**NetworkTransmitThroughput**) - 모니터링 및 복제에 사용되는 고객 데이터베이스 트래픽과 AWS 트래픽을 모두 포함하여 데이터베이스에서 송신되는 네트워크 트래픽입니다.

배포 지표

사용할 수 있는 배포 지표는 아래와 같습니다. 자세한 내용은 Amazon Lightsail에서 배포 지표 확인을 참조하세요.

- 요청(**Requests**) - 모든 HTTP 메소드 및 HTTP와 HTTPS 요청 모두에 대해 배포가 받은 총 뷰어 요청 수입니다.
- 업로드한 바이트(**BytesUploaded**) - POST 및 PUT 요청을 사용하여 배포가 오리진으로 업로드한 바이트 수입니다.

- 다운로드한 바이트(**BytesDownloaded**) - GET, HEAD 및 OPTIONS 요청에 대해 뷰어가 다운로드한 바이트 수입니다.
- 총 오류율(**TotalErrorRate**) - 응답의 HTTP 상태 코드가 4xx 또는 5xx인 모든 뷰어 요청의 백분율입니다.
- HTTP 4xx 오류율(**4xxErrorRate**) - 응답의 HTTP 상태 코드가 4xx인 모든 뷰어 요청의 백분율입니다. 이러한 경우 클라이언트 또는 클라이언트 뷰어가 오류를 일으켰을 수 있습니다. 예를 들어, 404 상태 코드(찾을 수 없음)는 클라이언트가 찾을 수 없는 객체를 요청했음을 의미합니다.
- HTTP 5xx 오류율(**5xxErrorRate**) - 응답의 HTTP 상태 코드가 5xx인 모든 뷰어 요청의 백분율입니다. 이러한 경우 원본 서버가 요청을 충족하지 못한 것입니다. 예를 들어, 503 상태 코드(서비스를 사용할 수 없음)는 원본 서버를 현재 사용할 수 없음을 의미합니다.

로드 밸런서 지표

다음과 같은 로드 밸런서 지표를 사용할 수 있습니다. 자세한 내용은 Amazon Lightsail의 로드 밸런서 지표 보기를 참조하십시오.

- 정상 호스트 수(**HealthyHostCount**) - 정상으로 간주되는 대상 인스턴스 수입니다.
- 비정상 호스트 수(**UnhealthyHostCount**) - 비정상으로 간주되는 대상 인스턴스 수입니다.
- 로드 밸런서 HTTP 4XX(**HTTPCode_LB_4XX_Count**) - 로드 밸런서에서 생성된 HTTP 4XX 클라이언트 오류 코드 수입니다. 클라이언트 오류는 요청 형식이 잘못되었거나 불완전할 때 생성됩니다. 이러한 요청은 대상 인스턴스에서 수신되지 않습니다. 대상 인스턴스에서 생성된 응답 코드는 이 숫자에 포함되지 않습니다.
- 로드 밸런서 HTTP 5XX(**HTTPCode_LB_5XX_Count**) - 로드 밸런서에서 생성된 HTTP 5XX 서버 오류 코드 수입니다. 대상 인스턴스에서 생성된 응답 코드는 여기에 포함되지 않습니다. 이 지표는 로드 밸런서에 정상 인스턴스가 연결되어 있지 않거나 요청 속도가 인스턴스 용량을 초과하거나(스필오버) 또는 로드 밸런서 용량을 초과하는 경우에 보고됩니다.
- 인스턴스 HTTP 2XX(**HTTPCode_Instance_2XX_Count**) - 대상 인스턴스에서 생성된 HTTP 2XX 응답 코드 수입니다. 단, 로드 밸런서에서 생성된 응답 코드 수는 여기에 포함되지 않습니다.
- 인스턴스 HTTP 3XX(**HTTPCode_Instance_3XX_Count**) - 대상 인스턴스에서 생성된 HTTP 3XX 응답 코드 수입니다. 단, 로드 밸런서에서 생성된 응답 코드 수는 여기에 포함되지 않습니다.
- 인스턴스HTTP 4XX(**HTTPCode_Instance_4XX_Count**) - 대상 인스턴스에서 생성된 HTTP 4XX 응답 코드 수입니다. 단, 로드 밸런서에서 생성된 응답 코드 수는 여기에 포함되지 않습니다.
- 인스턴스 HTTP 5XX(**HTTPCode_Instance_5XX_Count**) - 대상 인스턴스에서 생성된 HTTP 5XX 응답 코드 수입니다. 단, 로드 밸런서에서 생성된 응답 코드 수는 여기에 포함되지 않습니다.

- 인스턴스 응답 시간(**InstanceResponseTime**) - 로드 밸런서에서 요청을 보낸 후 대상 인스턴스로부터 응답 신호를 받을 때까지의 경과 시간(초)입니다.
- 클라이언트 TLS 협상 오류 수(**ClientTLSNegotiationErrorCount**) - 로드 밸런서에서 생성된 TLS 오류로 인해 로드 밸런서에서 세션을 설정하지 않은 클라이언트에서 시작된 TLS 연결 수입니다. 가능한 원인으로는 암호 또는 프로토콜 불일치가 있습니다.
- 요청 수(**RequestCount**) - IPv4를 통해 처리된 요청 수입니다. 로드 밸런서의 대상 인스턴스에서 응답을 생성한 요청만 이 개수에 포함됩니다.
- 거부된 연결 수(**RejectedConnectionCount**) - 로드 밸런서가 최대 연결 수에 도달하여 거부된 연결 수입니다.

컨테이너 서비스 지표

사용할 수 있는 컨테이너 서비스 지표는 다음과 같습니다. 자세한 내용은 컨테이너 서비스 지표 확인을 참조하세요.

- CPU 사용률(**CPUUtilization**) - 컨테이너 서비스의 모든 노드에서 현재 사용 중인 컴퓨팅 유닛의 평균 백분율입니다. 이 지표는 컨테이너 서비스에서 컨테이너를 실행하는 데 필요한 처리 능력을 나타냅니다.
- 메모리 사용률(**MemoryUtilization**) - 컨테이너 서비스의 모든 노드에서 현재 사용 중인 메모리의 평균 백분율입니다. 이 지표는 컨테이너 서비스에서 컨테이너를 실행하는 데 필요한 메모리를 나타냅니다.

버킷 지표

사용할 수 있는 버킷 지표는 아래와 같습니다. 자세한 내용은 Amazon Lightsail에서 버킷 지표 확인을 참조하세요.

- 버킷 크기(**BucketSizeBytes**) - 버킷에 저장된 데이터의 양입니다. 이 값은 버킷에 대한 모든 불완전 멀티파트 업로드의 모든 파트 크기를 포함하여 버킷(현재 객체 및 현재가 아닌 객체 모두)의 모든 객체 크기를 합산하여 계산됩니다.
- 객체 수(**NumberOfObjects**) - 버킷에 저장된 총 객체 수입니다. 이 값은 버킷(현재 객체 및 현재가 아닌 객체 모두)의 모든 객체 수와 버킷에 대한 모든 불완전 멀티파트 업로드의 총 파트 수를 합산하여 계산됩니다.

> **Note**
>
> 버킷이 비어 있으면 버킷 지표 데이터가 보고되지 않습니다.

상태 지표를 사용하여 Lightsail 리소스 모니터링

다양한 기간에 걸쳐 다음 Amazon Lightsail 리소스 지표를 볼 수 있습니다. Lightsail의 리소스 지표에 대한 자세한 내용은 리소스 지표를 참조하세요.

인스턴스 지표

다음과 같은 인스턴스 지표를 사용할 수 있습니다. 자세한 내용은 Amazon Lightsail의 인스턴스 지표 보기를 참조하세요.

- CPU 사용률(**CPUUtilization**) - 인스턴스에서 현재 사용 중인 할당된 컴퓨팅 유닛의 비율(%)입니다. 이 지표는 인스턴스에서 애플리케이션을 실행하는 처리 능력을 식별합니다. 인스턴스에 전체 프로세스 코어가 할당되지 않았을 때 운영 체제의 도구에서 비율이 Lightsail보다 낮게 표시할 수 있습니다.

  Lightsail 콘솔에서 인스턴스의 CPU 사용률 지표 그래프를 보면 지속 가능 영역과 버스트 가능 영역이 있습니다. 이러한 영역의 의미에 대한 자세한 내용은 CPU 사용률 지속 가능 및 버스트 가능 영역을 참조하십시오.
- 버스트 용량(분)(**BurstCapacityTime**) 및 백분율(**BurstCapacityPercentage**) - 버스트 용량(분)은 인스턴스가 100% CPU 사용률로 버스트할 수 있는 시간을 나타냅니다. 버스트 용량 백분율은 인스턴스에 사용할 수 있는 CPU 성능의 백분율입니다. 인스턴스는 지속적으로 버스트 용량을 소비하고 누적합니다. 버스트 용량(분)은 인스턴스가 100% CPU 사용률로 작동하는 경우에만 전체 비율로 소비됩니다. 인스턴스 버스트 용량에 대한 자세한 내용은 인스턴스 버스트 용량 보기를 참조하세요.
- 수신 네트워크 트래픽(**NetworkIn**) - 인스턴스가 모든 네트워크 인터페이스에서 수신한 바이트 수입니다. 이 지표는 단일 인스턴스로 들어오는 네트워크 트래픽의 볼륨을 식별합니다. 보고된 숫자는 해당 기간에 수신된 바이트 수입니다. 이 지표는 5분 간격으로 보고되므로 보고된 숫자를 300으로 나누어 초당 바이트 수를 구합니다.
- 송신 네트워크 트래픽(**NetworkOut**) - 인스턴스가 모든 네트워크 인터페이스에서 보낸 바이트 수입니다. 이 지표는 단일 인스턴스에서 나가는 네트워크 트래픽의 볼륨을 식별합니다. 보고된 숫자는 해당 기간에 전송된 바이트 수입니다. 이 지표는 5분 간격으로 보고되므로 보고된 숫자를 300으로 나누어 초당 바이트 수를 구합니다.

- 상태 확인 실패(**StatusCheckFailed**) - 인스턴스가 인스턴스 상태 확인 및 시스템 상태 확인을 통과했는지 여부를 보고합니다. 이 지표는 0(통과) 또는 1(실패)이 될 수 있습니다. 이 지표는 1분 간격으로 제공됩니다.
- 인스턴스 상태 확인 실패(**StatusCheckFailed_Instance**) - 인스턴스가 인스턴스 상태 확인을 통과했는지 여부를 보고합니다. 이 지표는 0(통과) 또는 1(실패)이 될 수 있습니다. 이 지표는 1분 간격으로 제공됩니다.
- 시스템 상태 확인 실패(**StatusCheckFailed_System**) - 인스턴스가 시스템 상태 확인을 통과했는지 여부를 보고합니다. 이 지표는 0(통과) 또는 1(실패)이 될 수 있습니다. 이 지표는 1분 간격으로 제공됩니다.
- 시스템 상태 확인 실패(**StatusCheckFailed_System**) - 인스턴스가 시스템 상태 확인을 통과했는지 여부를 보고합니다. 이 지표는 0(통과) 또는 1(실패)이 될 수 있습니다. 이 지표는 1분 간격으로 제공됩니다.
- 토큰 메타데이터 요청 없음(**MetadataNoToken**) - 토큰 없이 인스턴스 메타데이터 서비스에 성공적으로 액세스한 횟수입니다. 이 메트릭은 토큰을 사용하지 않는 인스턴스 메타데이터 서비스 버전 1을 사용하여 인스턴스 메타데이터에 액세스하는 프로세스가 있는지 확인합니다. 모든 요청이 인스턴스 메타데이터 서비스 버전 2와 같은 토큰 지원 세션을 사용하는 경우 값은 0입니다. 자세한 내용은 인스턴스 메타데이터 및 사용자 데이터를 참조하십시오.

데이터베이스 지표

다음과 같은 데이터베이스 지표를 사용할 수 있습니다. 자세한 내용은 데이터베이스 지표 보기를 참조하세요.

- CPU 사용률(**CPUUtilization**) - 데이터베이스에서 현재 사용 중인 CPU 사용률입니다.
- 데이터베이스 연결(**DatabaseConnections**) - 사용 중인 데이터베이스 연결 수입니다.
- 디스크 대기열 깊이(**DiskQueueDepth**) - 디스크에 액세스하기 위해 대기 중인 IO(읽기/쓰기 요청) 수입니다.
- 여유 스토리지 공간(**FreeStorageSpace**) - 사용 가능한 스토리지 공간 크기입니다.
- 네트워크 수신 처리량(**NetworkReceiveThroughput**) - 모니터링 및 복제에 사용되는 고객 데이터베이스 트래픽과 AWS 트래픽을 모두 포함하여 데이터베이스에 수신되는 네트워크 트래픽입니다.
- 네트워크 송신 처리량(**NetworkTransmitThroughput**) - 모니터링 및 복제에 사용되는 고객 데이터베이스 트래픽과 AWS 트래픽을 모두 포함하여 데이터베이스에서 송신되는 네트워크 트래픽입니다.

배포 지표

사용할 수 있는 배포 지표는 아래와 같습니다. 자세한 내용은 Amazon Lightsail에서 배포 지표 확인을 참조하세요.

- 요청 - 모든 HTTP 메소드와 HTTP 및 HTTPS 요청 둘 다에 대해 배포에서 수신한 최종 사용자 요청의 총 수입니다.
- 업로드된 바이트 - POST 및 PUT 요청을 사용하여 배포에서 오리진으로 업로드한 바이트 수입니다.
- 다운로드된 바이트 - GET, HEAD 및 OPTIONS 요청에 대해 최종 사용자가 다운로드한 바이트 수입니다.
- 총 오류 발생률 - 응답의 HTTP 상태 코드가 4xx 또는 5xx인 모든 최종 사용자 요청의 백분율입니다.
- HTTP 4xx 오류 발생률 - 응답의 HTTP 상태 코드가 4xx인 모든 최종 사용자 요청의 백분율입니다. 이러한 경우 클라이언트 또는 클라이언트 뷰어가 오류를 일으켰을 수 있습니다. 예를 들어, 404 상태 코드(찾을 수 없음)는 클라이언트가 찾을 수 없는 객체를 요청했음을 의미합니다.
- HTTP 5xx 오류 발생률 - 응답의 HTTP 상태 코드가 5xx인 모든 최종 사용자 요청의 백분율입니다. 이러한 경우 원본 서버가 요청을 충족하지 못한 것입니다. 예를 들어, 503 상태 코드(서비스를 사용할 수 없음)는 원본 서버를 현재 사용할 수 없음을 의미합니다.

로드 밸런서 지표

다음과 같은 로드 밸런서 지표를 사용할 수 있습니다. 자세한 내용은 로드 밸런서 지표 보기를 참조하세요.

- 정상 호스트 수(**HealthyHostCount**) - 정상으로 간주되는 대상 인스턴스 수입니다.
- 비정상 호스트 수(**UnhealthyHostCount**) - 비정상으로 간주되는 대상 인스턴스 수입니다.
- 로드 밸런서 HTTP 4XX(**HTTPCode_LB_4XX_Count**) - 로드 밸런서에서 생성된 HTTP 4XX 클라이언트 오류 코드 수입니다. 클라이언트 오류는 요청 형식이 잘못되었거나 불완전할 때 생성됩니다. 이러한 요청은 대상 인스턴스에서 수신되지 않습니다. 대상 인스턴스에서 생성된 응답 코드는 이 숫자에 포함되지 않습니다.
- 로드 밸런서 HTTP 5XX(**HTTPCode_LB_5XX_Count**) - 로드 밸런서에서 생성된 HTTP 5XX 서버 오류 코드 수입니다. 대상 인스턴스에서 생성된 응답 코드는 여기에 포함되지 않습니다. 이 지표는 로드 밸런서에 정상 인스턴스가 연결되어 있지 않거나 요청 속도가 인스턴스 용량을 초과하거나(스필오버) 또는 로드 밸런서 용량을 초과하는 경우에 보고됩니다.
- 인스턴스 HTTP 2XX(**HTTPCode_Instance_2XX_Count**) - 대상 인스턴스에서 생성된 HTTP 2XX 응답 코드 수입니다. 단, 로드 밸런서에서 생성된 응답 코드 수는 여기에 포함되지 않습니다.

- 인스턴스 HTTP 3XX(**HTTPCode_Instance_3XX_Count**) - 대상 인스턴스에서 생성된 HTTP 3XX 응답 코드 수입니다. 단, 로드 밸런서에서 생성된 응답 코드 수는 여기에 포함되지 않습니다.
- 인스턴스HTTP 4XX(**HTTPCode_Instance_4XX_Count**) - 대상 인스턴스에서 생성된 HTTP 4XX 응답 코드 수입니다. 단, 로드 밸런서에서 생성된 응답 코드 수는 여기에 포함되지 않습니다.
- 인스턴스 HTTP 5XX(**HTTPCode_Instance_5XX_Count**) - 대상 인스턴스에서 생성된 HTTP 5XX 응답 코드 수입니다. 단, 로드 밸런서에서 생성된 응답 코드 수는 여기에 포함되지 않습니다.
- 인스턴스 응답 시간(**InstanceResponseTime**) - 로드 밸런서에서 요청을 보낸 후 대상 인스턴스로부터 응답 신호를 받을 때까지의 경과 시간(초)입니다.
- 요청 수(**RequestCount**) - IPv4를 통해 처리된 요청 수입니다. 로드 밸런서의 대상 인스턴스에서 응답을 생성한 요청만 이 개수에 포함됩니다.
- 클라이언트 TLS 협상 오류 수(**ClientTLSNegotiationErrorCount**) - 로드 밸런서에서 생성된 TLS 오류로 인해 로드 밸런서에서 세션을 설정하지 않은 클라이언트에서 시작된 TLS 연결 수입니다. 가능한 원인으로는 암호 또는 프로토콜 불일치가 있습니다.
- 거부된 연결 수(**RejectedConnectionCount**) - 로드 밸런서가 최대 연결 수에 도달하여 거부된 연결 수입니다.

컨테이너 서비스 지표

사용할 수 있는 컨테이너 서비스 지표는 다음과 같습니다. 자세한 내용은 컨테이너 서비스 지표 확인을 참조하세요.

- CPU 사용률 - 컨테이너 서비스의 모든 노드에서 현재 사용 중인 컴퓨팅 단위의 평균 백분율입니다. 이 지표는 컨테이너 서비스에서 컨테이너를 실행하는 데 필요한 처리 능력을 나타냅니다.
- 메모리 사용률 - 컨테이너 서비스의 모든 노드에서 현재 사용 중인 메모리의 평균 백분율입니다. 이 지표는 컨테이너 서비스에서 컨테이너를 실행하는 데 필요한 메모리를 나타냅니다.

버킷 지표

사용할 수 있는 버킷 지표는 아래와 같습니다. 자세한 내용은 버킷 지표 확인을 참조하세요.

- 버킷 크기 - 버킷에 저장된 데이터의 양입니다. 이 값은 버킷에 대한 모든 불완전 멀티파트 업로드의 모든 파트 크기를 포함하여 버킷(현재 객체 및 현재가 아닌 객체 모두)의 모든 객체 크기를 합산하여 계산됩니다.
- 객체 수 - 버킷에 저장된 총 객체 수입니다. 이 값은 버킷(현재 객체 및 현재가 아닌 객체 모두)의 모든 객체 수와 버킷에 대한 모든 불완전 멀티파트 업로드의 총 파트 수를 합산하여 계산됩니다.

> **Note**
>
> 버킷이 비어 있으면 버킷 지표 데이터가 보고되지 않습니다.

주제

- Lightsail 리소스에 대한 지표 알림 구성
- 지표를 사용하여 Lightsail 인스턴스 성능 모니터링
- Lightsail에서 지표 알람 수신
- Lightsail 인스턴스 지표 경보 생성
- Lightsail 지표 경보 삭제 또는 비활성화

Lightsail 리소스에 대한 지표 알림 구성

인스턴스, 데이터베이스, 로드 밸런서 또는 콘텐츠 전송 네트워크(CDN) 배포의 지표가 지정된 임계값을 초과하면 알림을 받도록 Lightsail을 구성할 수 있습니다. 알림은 Lightsail 콘솔에 표시되는 배너, 지정한 주소로 전송되는 이메일 또는 지정한 휴대폰 번호로 전송되는 SMS 문자 메시지 형태로 사용할 수 있습니다. 알림 확인 보류 중인 연락처를 검토하는 방법에 대한 자세한 내용은 확인 보류 중인 이메일 연락처 검토 섹션을 참조하세요.

알림을 받으려면 리소스 중 하나의 지표를 모니터링하는 경보를 구성해야 합니다. 예를 들어, 지정된 시간 동안 인스턴스의 송신 네트워크 트래픽이 500KB를 초과할 때 알려주는 경보를 구성할 수 있습니다. 자세한 내용은 지표 경보를 참조하세요.

경보가 트리거되면 알림 배너가 Lightsail 콘솔에 표시됩니다. 이메일 및 SMS 문자 메시지로 알림을 받으려면 리소스를 모니터링 AWS 리전 하려는 각에서 이메일 주소와 휴대폰 번호를 알림 연락처로 추가해야 합니다. 자세한 내용은 알림 연락처 추가를 참조하세요.

> **Note**
>
> SMS 문자 메시지는 Lightsail 리소스를 생성할 수 있는 모든 AWS 리전에서 지원되지 않으며, 전 세계 일부 국가 및 리전으로 문자 메시지를 전송할 수 없습니다. 자세한 내용은 알림 연락처 추가를 참조하세요.

알림을 받아야 할 때 알림이 수신되지 않는 경우 몇 가지 사항을 점검하여 알림 연락처가 올바르게 구성되었는지 확인해야 합니다. 자세한 내용은 알림 문제 해결을 참조하세요.

알림 수신을 중지하려면 Lightsail에서 이메일과 휴대폰 번호를 삭제합니다. 자세한 내용은 지표 경보 삭제 또는 비활성화를 참조하세요. 또한 경보를 비활성화하거나 삭제하여 특정 경보에 대한 알림 수신을 중지할 수도 있습니다. 자세한 내용은 지표 경보 삭제 또는 비활성화를 참조하세요.

지표를 사용하여 Lightsail 인스턴스 성능 모니터링

Amazon Lightsail에서 인스턴스를 시작한 후 인스턴스 관리 페이지의 지표 탭에서 해당 지표 그래프를 볼 수 있습니다. 지표 모니터링은 리소스의 안정성, 가용성 및 성능을 유지하는 데 중요한 부분입니다. 리소스에서 지표 데이터를 정기적으로 모니터링하고 수집하면 다중 지점 오류가 발생할 경우 보다 쉽게 디버깅할 수 있습니다. 지표에 대한 자세한 내용은 Amazon Lightsail의 지표를 참조하십시오.

리소스를 모니터링할 때는 사용자 환경에서 정상적인 리소스 성능에 대한 기준을 설정해야 합니다. 그런 다음 리소스가 지정된 임계값을 벗어날 때 알림을 받도록 Lightsail 콘솔에서 경보를 구성할 수 있습니다. 자세한 내용은 알림 및 경보를 참조하세요.

목차

- Lightsail에서 사용할 수 있는 인스턴스 지표
- CPU 사용률 지속 가능 및 버스트 가능 영역
- Lightsail 콘솔에서 인스턴스 지표 확인
- 인스턴스 지표 확인 후의 다음 단계

사용 가능한 인스턴스 지표

다음과 같은 인스턴스 지표를 사용할 수 있습니다.

- CPU 사용률(**CPUUtilization**) - 인스턴스에서 현재 사용 중인 할당된 컴퓨팅 유닛의 비율(%)입니다. 이 지표는 인스턴스에서 애플리케이션을 실행하는 처리 능력을 식별합니다. 인스턴스에 전체 프로세스 코어가 할당되지 않았을 때 운영 체제의 도구에서 비율이 Lightsail보다 낮게 표시할 수 있습니다.

  Lightsail 콘솔에서 인스턴스의 CPU 사용률 지표 그래프를 보면 지속 가능 영역과 버스트 가능 영역이 있습니다. 이러한 영역의 의미에 대한 자세한 내용은 CPU 사용률 지속 가능 및 버스트 가능 영역을 참조하십시오.
- 버스트 용량(분)(**BurstCapacityTime**) 및 백분율(**BurstCapacityPercentage**) - 버스트 용량(분)은 인스턴스가 100% CPU 사용률로 버스트할 수 있는 시간을 나타냅니다. 버스트 용량 백분율은 인스턴스에 사용할 수 있는 CPU 성능의 백분율입니다. 인스턴스는 지속적으로 버스트 용량을 소비

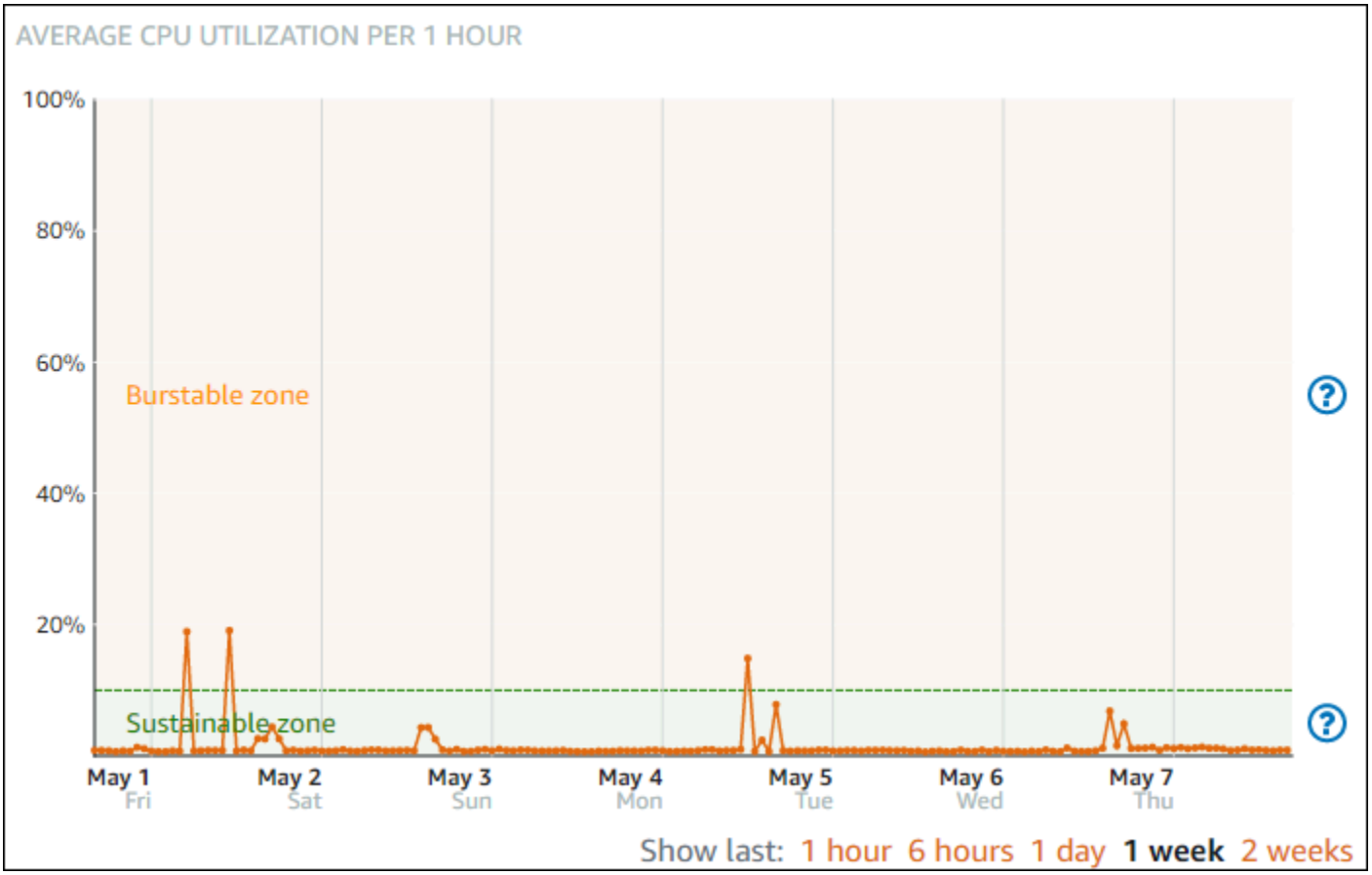
하고 누적합니다. 버스트 용량(분)은 인스턴스가 100% CPU 사용률로 작동하는 경우에만 전체 비율로 소비됩니다. 인스턴스 버스트 용량에 대한 자세한 내용은 인스턴스 버스트 용량 보기를 참조하세요.

- 수신 네트워크 트래픽(**NetworkIn**) - 인스턴스가 모든 네트워크 인터페이스에서 수신한 바이트 수입니다. 이 지표는 단일 인스턴스로 들어오는 네트워크 트래픽의 볼륨을 식별합니다. 보고된 숫자는 해당 기간에 수신된 바이트 수입니다. 이 지표는 5분 간격으로 보고되므로 보고된 숫자를 300으로 나누어 초당 바이트 수를 구합니다.
- 송신 네트워크 트래픽(**NetworkOut**) - 인스턴스가 모든 네트워크 인터페이스에서 보낸 바이트 수입니다. 이 지표는 단일 인스턴스에서 나가는 네트워크 트래픽의 볼륨을 식별합니다. 보고된 숫자는 해당 기간에 전송된 바이트 수입니다. 이 지표는 5분 간격으로 보고되므로 보고된 숫자를 300으로 나누어 초당 바이트 수를 구합니다.
- 상태 확인 실패(**StatusCheckFailed**) - 인스턴스가 인스턴스 상태 확인 및 시스템 상태 확인을 통과했는지 여부를 보고합니다. 이 지표는 0(통과) 또는 1(실패)이 될 수 있습니다. 이 지표는 1분 간격으로 제공됩니다.
- 인스턴스 상태 확인 실패(**StatusCheckFailed_Instance**) - 인스턴스가 인스턴스 상태 확인을 통과했는지 여부를 보고합니다. 이 지표는 0(통과) 또는 1(실패)이 될 수 있습니다. 이 지표는 1분 간격으로 제공됩니다.
- 시스템 상태 확인 실패(**StatusCheckFailed_System**) - 인스턴스가 시스템 상태 확인을 통과했는지 여부를 보고합니다. 이 지표는 0(통과) 또는 1(실패)이 될 수 있습니다. 이 지표는 1분 간격으로 제공됩니다.
- 토큰 메타데이터 요청 없음(**MetadataNoToken**) - 토큰 없이 인스턴스 메타데이터 서비스에 성공적으로 액세스한 횟수입니다. 이 메트릭은 토큰을 사용하지 않는 인스턴스 메타데이터 서비스 버전 1을 사용하여 인스턴스 메타데이터에 액세스하는 프로세스가 있는지 확인합니다. 모든 요청이 인스턴스 메타데이터 서비스 버전 2와 같은 토큰 지원 세션을 사용하는 경우 값은 0입니다. 자세한 내용은 인스턴스 메타데이터 및 사용자 데이터를 참조하십시오.

CPU 사용률 지속 가능 및 버스트 가능 영역

Lightsail은 기준 CPU 성능을 제공하는 버스트 가능 인스턴스를 사용하지만, 필요에 따라 기준 이상의 추가 CPU 성능을 일시적으로 제공할 수도 있습니다. 이를 버스팅이라고 합니다. 버스트 가능 인스턴스를 사용하면 가끔 발생하는 필요 성능 급증을 처리하고자 인스턴스를 과도하게 프로비저닝할 필요가 없기 때문에, 사용하지 않는 용량의 비용을 지출하는 일이 없습니다.

인스턴스의 CPU 사용률 지표 그래프에는 지속 가능 영역과 버스트 가능 영역이 표시됩니다. Lightsail 인스턴스는 시스템 운영에 영향을 주지 않고 지속 가능 영역에서 무기한으로 작동할 수 있습니다.



코드 컴파일, 새 소프트웨어 설치, 배치 작업 실행 또는 최대 로드 요청 처리 등으로 부하가 큰 경우 버스트 가능 영역에서 인스턴스가 작동할 수 있습니다. 버스트 가능 영역에서 작동하는 동안에는 인스턴스가 상대적으로 더 많은 CPU 주기를 사용합니다. 따라서 제한된 기간 동안만 이 영역에서 작동할 수 있습니다.

인스턴스가 버스트 가능 영역에서 작동할 수 있는 기간은 인스턴스가 버스트 가능 영역 내의 어느 지점에 있는지에 따라 달라집니다. 버스트 가능 영역의 하단에서 작동하는 인스턴스는 버스트 가능 영역의 상단에서 작동하는 인스턴스보다 오랜 기간 동안 버스트될 수 있습니다. 그러나 장기간 버스트 가능 영역에 있는 인스턴스는 지속 가능 영역에서 다시 작동할 때까지 결국 모든 CPU 용량을 소모하게 됩니다.

인스턴스의 CPU 사용률 지표를 모니터링하여 지속 가능 영역과 버스트 가능 영역 간에 성능이 어떻게 분포되는지 확인합니다. 시스템이 가끔씩 버스트 가능 영역으로 이동하는 경우 실행 중인 인스턴스를 계속 사용해도 좋습니다. 그러나 인스턴스가 버스트 가능 영역에 상당 시간 있는 경우 인스턴스 플랜을 더 큰 플랜으로 전환하는 것이 좋을 수 있습니다(12 USD/월 플랜 대신 5 USD/월 플랜 사용). 인스턴스의 새 스냅샷을 생성한 다음 스냅샷에서 새 인스턴스를 생성하여 더 큰 플랜으로 전환할 수 있습니다.

Lightsail 콘솔에서 인스턴스 지표 확인

Lightsail 콘솔에서 인스턴스 지표를 보려면 다음 단계를 수행하십시오.

1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 인스턴스를 선택합니다.
3. 지표를 보려는 인스턴스의 이름을 선택합니다.
4. 인스턴스 관리 페이지에서 지표 탭을 선택합니다.
5. 지표 그래프 머리글 아래의 드롭다운 메뉴에서 보려는 지표를 선택합니다.

   선택한 지표에 대한 데이터 포인트가 그래프에 시각적으로 표시됩니다.

   > **Note**
   >
   > Lightsail 콘솔에서 인스턴스의 CPU 사용률 지표 그래프를 보면 지속 가능 영역과 버스트 가능 영역이 있습니다. 이러한 영역에 대한 자세한 내용은 CPU 사용률 지속 가능 및 버스트 가능 영역을 참조하십시오.

6. 지표 그래프에 대해 다음 작업을 수행할 수 있습니다.

   - 1시간, 6시간, 1일, 1주 및 2주 동안의 데이터를 표시하도록 그래프 보기를 변경합니다.
   - 데이터 포인트에 커서를 놓으면 해당 데이터 포인트에 대한 자세한 정보를 볼 수 있습니다.
   - 선택한 지표가 지정한 임계값을 초과할 때 알림을 받도록 지표에 대한 경보를 추가합니다. 자세한 내용은 경보 및 인스턴스 지표 경보 생성을 참조하세요.

다음 단계

인스턴스 지표에 대해 수행할 수 있는 몇 가지 추가 작업이 있습니다.

- 선택한 지표가 지정한 임계값을 초과할 때 알림을 받도록 지표에 대한 경보를 추가합니다. 자세한 내용은 지표 경보 및 인스턴스 지표 경보 생성을 참조하세요.
- 경보가 트리거되면 알림 배너가 Lightsail 콘솔에 표시됩니다. 이메일 및 SMS 문자 메시지로 알림을 받으려면 리소스를 모니터링 AWS 리전 하려는 각에 이메일 주소와 휴대폰 번호를 알림 연락처로 추가해야 합니다. 자세한 내용은 알림 연락처 추가를 참조하세요.
- 알림 수신을 중지하려면 Lightsail에서 이메일과 휴대폰 번호를 삭제합니다. 자세한 내용은 지표 경보 삭제 또는 비활성화를 참조하세요. 또한 경보를 비활성화하거나 삭제하여 특정 경보에 대한 알림 수신을 중지할 수도 있습니다. 자세한 내용은 지표 경보 삭제 또는 비활성화를 참조하세요.

Lightsail에서 지표 알람 수신

Amazon Lightsail에서 인스턴스, 데이터베이스, 로드 밸런서 및 콘텐츠 전송 네트워크(CDN) 배포에 대한 단일 지표를 감시하는 경보를 만들 수 있습니다. 지정한 임계값에 상대적인 지표 값을 기준으로 경보 알림을 구성할 수 있습니다. 알림은 Lightsail 콘솔에 표시되는 배너, 이메일 주소로 전송되는 이메일, 휴대폰 번호로 전송되는 SMS 문자 메시지 형태로 사용할 수 있습니다. 이 안내서에서는 구성할 수 있는 경보 조건 및 설정을 설명합니다. 모든 Lightsail 리소스에 대한 활성 경보를 검토하는 방법에 대한 자세한 내용은 활성 경보에 대한 경보 알림 검토 섹션을 참조하세요.

내용

- 경보 구성
- 경보 상태
- 경보 예
- 경보가 누락 데이터를 처리하는 방법 구성
- 데이터가 누락되었을 때 경보 상태 평가 방법
- 그래프 예의 누락 데이터
- 경보에 대한 자세한 정보

경보 구성

Lightsail 콘솔에서 경보를 추가하려면 인스턴스, 데이터베이스, 로드 밸런서 또는 CDN 배포의 지표(Metrics) 탭으로 이동합니다. 그런 다음 모니터링할 지표를 선택하고 Add alarm(경보 추가)을 선택합니다. 지표당 두 개의 경보를 추가할 수 있습니다. 지표에 대한 자세한 내용은 리소스 지표를 참조하세요.

경보를 구성하려면 먼저 임계값을 지정합니다. 임계값은 경보의 상태가 변경되는(예: `OK` 상태에서 `ALARM` 상태로 또는 그 반대로 변경) 지점의 지표 값입니다. 자세한 내용은 경보 상태를 참조하세요. 지표를 임계값과 비교하는 데 사용할 비교 연산자를 선택합니다. 사용 가능한 연산자는 greater than or equal to(크거나 같음), greater than(보다 큼), less than(보다 작음) 및 less than or equal to(작거나 같음)입니다.

그런 다음 경보 상태가 변경되는 기준인 임계값 초과 횟수와 지표를 평가하는 기간을 지정합니다. Lightsail은 경보 여부를 판단하고자 5분마다 데이터 포인트를 평가하며, 각 데이터 포인트는 5분간 집계된 데이터를 나타냅니다. 예를 들어 임계값을 2회 초과하면 경보가 트리거되도록 지정한 경우 평가 기간은 지난 10분 이상(최대 24시간)이어야 합니다. 임계값을 10회 초과하면 경보가 트리거되도록 지정한 경우 평가 기간은 지난 50분 이상(최대 24시간)이어야 합니다.

경보 조건을 구성한 후 알림을 받는 방법을 구성할 수 있습니다. 경보가 Lightsail 상태에서 OK 상태로 바뀌면 항상 ALARM 콘솔에 알림 배너가 표시됩니다. 이메일 및 SMS 문자 메시지로 알림을 받도록 선택할 수도 있지만 알림 연락처를 구성해야 합니다. 자세한 내용은 지표 알림을 참조하세요. 이메일 및/또는 SMS 문자 메시지로 알림을 받기로 선택한 경우 경보 상태가 ALARM 상태에서 OK 상태로 변경될 때에도 알림을 받도록 설정할 수 있습니다. 이 알림은 모두 지우기 알림입니다.

경보에 대한 고급 설정에서 Lightsail이 누락된 지표 데이터를 처리하는 방법을 선택할 수 있습니다. 자세한 내용은 경보가 누락 데이터를 처리하는 방법 구성을 참조하세요.

경보 상태

경보는 항상 다음 상태 중 하나입니다.

- ALARM - 지표가 정의된 임곗값을 벗어났습니다.

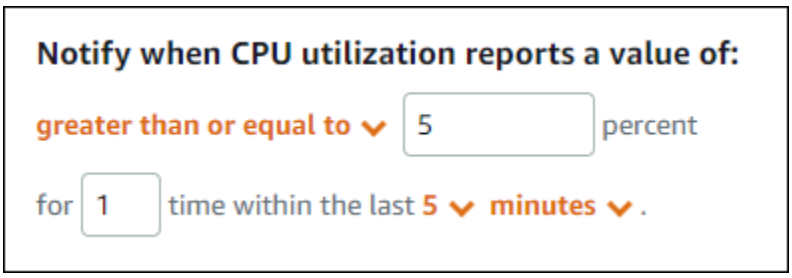
  예를 들어, greater than(보다 큼) 비교 연산자를 선택하면 지표가 지정된 임계값보다 클 때 경보가 ALARM 상태가 됩니다. less than(보다 작음) 비교 연산자를 선택하면 지표가 지정된 임계값보다 작을 때 경보가 ALARM 상태가 됩니다.
- OK - 측정치가 정의된 임곗값 내에 있습니다.

  예를 들어, greater than(보다 큼) 비교 연산자를 선택하면 지표가 지정된 임계값보다 작을 때 경보가 OK 상태가 됩니다. less than(보다 작음) 비교 연산자를 선택하면 지표가 지정된 임계값보다 클 때 경보가 OK 상태가 됩니다.
- INSUFFICIENT_DATA - 경보가 방금 시작되었거나, 지표를 사용할 수 없거나, 지표를 통해 경보 상태를 결정하는 데 사용할 충분한 데이터가 없습니다.

경보는 상태가 변경되는 경우에만 트리거되며, 경보는 단순히 특정 상태이기 때문에 트리거되는 것이 아니며, 상태가 변경되어야만 트리거됩니다. 경보가 트리거되면 Lightsail 콘솔에 배너가 표시됩니다. 이메일 및 SMS 문자 메시지로 알림을 받도록 경보를 구성할 수도 있습니다.

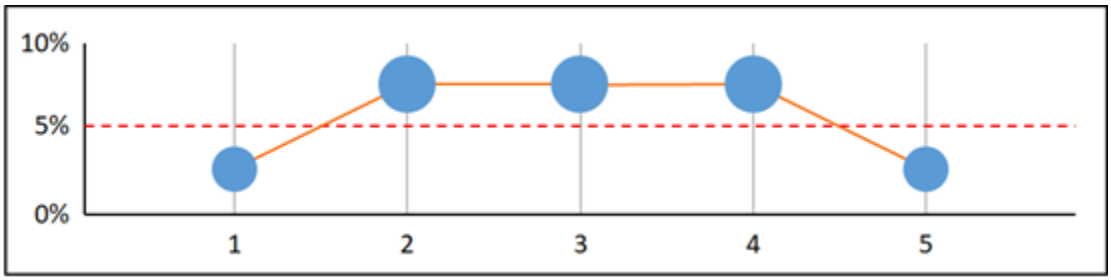
경보 예

앞에서 설명한 경보 조건을 염두에 두고 인스턴스의 CPU 사용률이 5분 동안 1회 5% 이상일 때 ALARM 상태가 되는 경보를 구성할 수 있습니다. 다음 예에서는 Lightsail 콘솔에서 이 경보를 설정하는 것을 보여 줍니다.



이 예에서 인스턴스의 CPU 사용률 지표가 단 하나의 데이터 포인트에서 5% 이상의 사용률로 보고되면 경보가 OK 상태에서 ALARM 상태로 변경됩니다. 사용률이 5% 이상으로 보고된 이후의 각 데이터 포인트에서는 경보가 ALARM 상태로 유지됩니다. 인스턴스의 CPU 사용률 지표가 단 하나의 데이터 포인트에서 4.9% 이하의 사용률로 보고되면 경보가 ALARM 상태에서 OK 상태로 변경됩니다.

다음 그래프는 이 경보를 더 자세히 보여줍니다. 빨간색 점선은 5% CPU 사용률 임계값을 나타내고 파란색 점은 지표 데이터 포인트를 나타냅니다. 첫 번째 데이터 포인트에서는 경보가 OK 상태입니다. 두 번째 데이터 포인트에서는 데이터 포인트가 임계값보다 크기 때문에 경보가 ALARM 상태로 변경됩니다. 세 번째 및 네 번째 데이터 포인트에서는 계속 데이터 포인트가 임계값보다 크기 때문에 ALARM 상태가 유지됩니다. 다섯 번째 데이터 포인트에서는 데이터 포인트가 임계값보다 작기 때문에 경보가 OK 상태로 변경됩니다.



경보가 누락 데이터를 처리하는 방법 구성

종종 특정 지표에 대한 경보 데이터 포인트 가운데 일부가 보고되지 않는 경우도 있습니다. 예를 들어 연결이 끊어지거나 서버가 중단되면 이러한 문제가 발생할 수 있습니다.

Lightsail에서는 경보를 구성할 때 누락 데이터 포인트를 처리하는 방법을 지정할 수 있습니다. 이렇게 하면 모니터링 중인 데이터 유형에 적합한 ALARM 상태로 경보를 구성할 수 있습니다. 누락된 데이터에 문제가 없는 경우의 거짓 긍정을 피할 수 있습니다.

각 경보가 항상 세 가지 상태 중 하나인 것과 마찬가지로, 보고된 각각의 특정 데이터 포인트는 세 가지 범주 중 하나에 속합니다.

- 위반하지 않음 - 데이터 포인트가 임곗값 내에 있습니다.

예를 들어, greater than(보다 큼) 비교 연산자를 선택하면 데이터 포인트가 지정된 임계값보다 작을 때 `Not breaching` 범주가 됩니다. less than(보다 작음) 비교 연산자를 선택하면 데이터 포인트가 지정된 임계값보다 클 때 `Not breaching` 범주가 됩니다.

- 위반 - 데이터 포인트가 임곗값을 벗어났습니다.

  예를 들어, greater than(보다 큼) 비교 연산자를 선택하면 데이터 포인트가 지정된 임계값보다 클 때 `Breaching` 범주가 됩니다. less than(보다 작음) 비교 연산자를 선택하면 데이터 포인트가 지정된 임계값보다 작을 때 `Breaching` 범주가 됩니다.
- 누락 - 데이터 포인트 누락에 대한 동작은 `treat missing data` 파라미터로 지정됩니다.

각 경보에 대해 다음 중 하나로 누락된 데이터 포인트가 처리되도록 Lightsail를 지정할 수 있습니다.

- 위반하지 않음 - 누락 데이터 포인트를 '양호'하고 임곗값 내에 있는 것으로 처리합니다.
- 위반 - 누락 데이터 포인트를 '불량'하고 임곗값을 위반한 것으로 처리합니다.
- 무시 - 현재 경보 상태를 유지합니다.
- 누락 - 경보가 상태 변경 여부를 평가할 때 누락 데이터 포인트를 고려하지 않습니다. 이는 경보의 기본 동작입니다.

최고의 옵션은 지표 유형에 따라 다릅니다. 인스턴스의 CPU 사용률과 같은 지표의 경우 누락 데이터 포인트를 위반으로 처리하는 것이 좋을 수 있습니다. 누락 데이터 포인트가 문제를 나타낼 수 있기 때문입니다. 그러나 로드 밸런서의 HTTP 500 서버 오류 수와 같이 오류가 발생한 경우에만 데이터 포인트를 생성하는 지표의 경우 누락 데이터를 위반하지 않음으로 처리하는 것이 좋을 수 있습니다.

경보에 대한 최상의 옵션을 선택하면 불필요하고 오해의 소지가 있는 경보 조건 변경을 막을 수 있으며, 시스템 상태를 보다 정확하게 나타낼 수 있습니다.

데이터가 누락되었을 때 경보 상태 평가 방법

Lightsail은 경보 상태 변경 여부를 평가할 때 누락 데이터를 처리하는 방법에 대해 설정한 값에 상관없이 Evaluation Periods(평가 기간)에서 지정한 것보다 더 큰 데이터 포인트 수를 검색하려 시도합니다. 검색하려고 시도하는 데이터 포인트 수는 경보 기간의 길이에 따라 다릅니다. 검색을 시도하는 데이터 포인트의 기간이 평가 범위입니다.

Lightsail이 이런 데이터 포인트를 검색한 후에는 다음이 진행됩니다.

- 평가 범위 동안 누락된 데이터 포인트가 없는 경우 Lightsail는 가장 최근 수집한 데이터 포인트에 따라 경보를 평가합니다.

- 평가 범위 동안 일부 데이터 포인트가 누락되었지만 수집한 기존 데이터 포인트의 수가 경보 Evaluation Periods(평가 기간) 수보다 많거나 같은 경우, Lightsail은 성공적으로 수집한 가장 최근의 기존 데이터 포인트에 따라 경보 상태를 평가합니다. 이 경우 누락 데이터 처리 방법에 대한 값이 필요 없으며, 이를 무시합니다.
- 평가 범위 동안 일부 데이터 포인트가 누락되었으며 수집한 기존 데이터 포인트의 수가 경보 Evaluation Periods(평가 기간) 수보다 적은 경우, Lightsail은 사용자가 누락 데이터 처리 방법에 대해 지정한 값으로 누락 데이터 포인트를 채운 다음 경보를 평가합니다. 하지만 보고 시기에 상관없이 평가 범위 동안의 실제 데이터 포인트는 모두 평가에 포함시킵니다. Lightsail는 가능한 몇 회만 누락 데이터 포인트를 사용합니다.

이 모든 상황에서 평가된 데이터 포인트의 수는 Evaluation periods(평가 기간)의 값과 동일합니다. Datapoints to alarm(경보에 대한 데이터 포인트)의 값보다 작은 값만 위반된 경우 경보 상태는 OK로 설정됩니다. 나머지 경우는 경보로 설정됩니다.

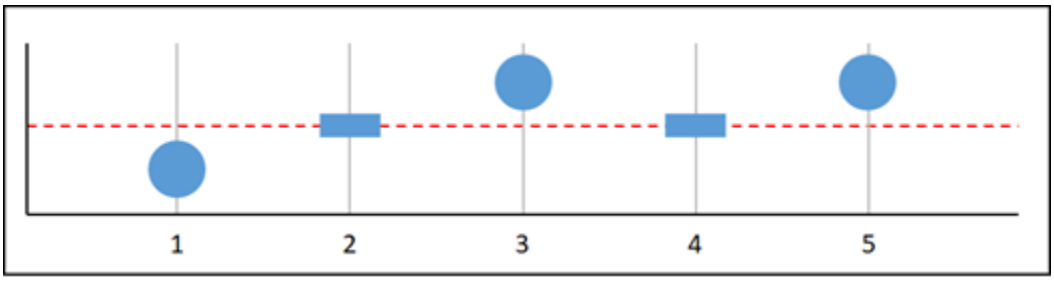
> **Note**
>
> 이 동작은 특별한 경우 지표 흐름이 멈춘 후에도 Lightsail 경보 기능이 일정 시간 동안 최종 데이터 포인트 세트를 계속해서 다시 평가할 수 있습니다. 이 재평가를 통해 지표 스트림 중지 직전에 상태가 변한 경우 경보가 상태를 변경하고 작업을 다시 실행할 수 있습니다. 이 동작을 완화하려면 더 짧은 기간을 사용하세요.

그래프 예의 누락 데이터

이 단원의 다음 그래프는 경보 평가 동작의 예를 보여 줍니다. 그래프 A, B, C, D 및 E에서 경보에 대한 위반 데이터 포인트 수와 평가 기간은 모두 3입니다. 빨간색 점선은 임계값을 나타내고, 파란색 점은 유효한 데이터 포인트를 나타내며, 파선은 누락 데이터를 나타냅니다. 임계값 선 위의 데이터 포인트는 위반이고 임계값 아래 데이터 포인트는 위반이 아닙니다. 가장 최근의 세 데이터 포인트 중 일부가 누락된 경우 Lightsail은 유효한 추가 데이터 포인트를 검색하려고 시도합니다.

> **Note**
>
> 경보를 생성한 직후에 데이터 포인트가 누락되었으며 경보를 생성하기 전에 Lightsail에 지표가 보고된 경우, Lightsail는 경보가 생성되기 전 가장 최근의 데이터 포인트를 검색해 경보를 평가합니다.

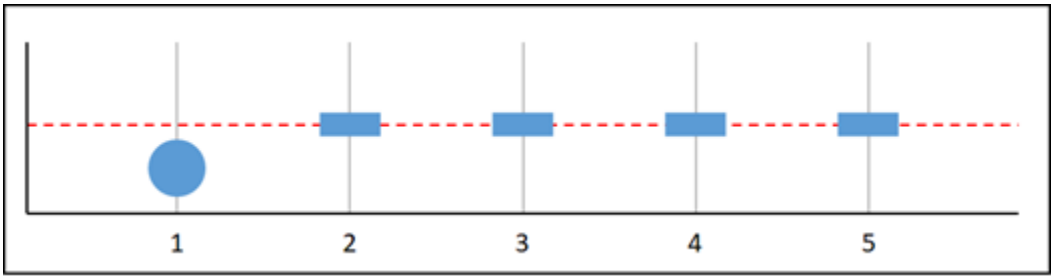
그래프 A



앞의 그래프로 표시된 지표에서 데이터 포인트 1은 임계값 내에 있고, 데이터 포인트 2는 누락되었고, 데이터 포인트 3은 위반이고, 데이터 포인트 4는 누락되었고, 데이터 포인트 5는 위반입니다. 평가 범위에 유효한 데이터 포인트가 3개 있으므로 이 지표에는 누락 데이터 포인트가 없습니다. 누락 데이터 포인트를 다음과 같이 처리하도록 경보를 구성한 경우:

- 위반하지 않음 - 경보가 OK 상태입니다.
- 위반 - 경보가 OK 상태입니다.
- 무시 - 경보가 OK 상태입니다.
- 누락 - 경보가 OK 상태입니다.

그래프 B

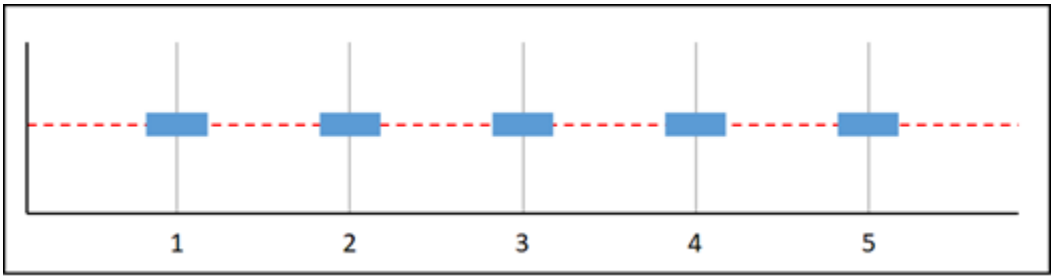


앞의 그래프로 표시된 지표에서 데이터 포인트 1은 임계값 내에 있고 데이터 포인트 2~5는 누락되었습니다. 평가 범위에 데이터 포인트가 하나만 있으므로 이 지표에는 누락 데이터 포인트가 두 개 있습니다. 누락 데이터 포인트를 다음과 같이 처리하도록 경보를 구성한 경우:

- 위반하지 않음 - 경보가 OK 상태입니다.
- 위반 - 경보가 OK 상태입니다.
- 무시 - 경보가 OK 상태입니다.
- 누락 - 경보가 OK 상태입니다.

이 시나리오에서는 누락 데이터가 위반으로 처리되더라도 경보가 OK 상태로 유지됩니다. 이는 하나의 기존 데이터 포인트가 위반되지 않고 위반으로 처리되는 두 개의 누락 데이터 포인트와 함께 평가되기 때문입니다. 다음에 이 경보를 평가할 때 데이터가 여전히 누락된 경우 ALARM 상태로 변경됩니다. 이는 검색된 최근 데이터 포인트 5개에 위반하지 않는 데이터 포인트가 더 이상 없기 때문입니다.

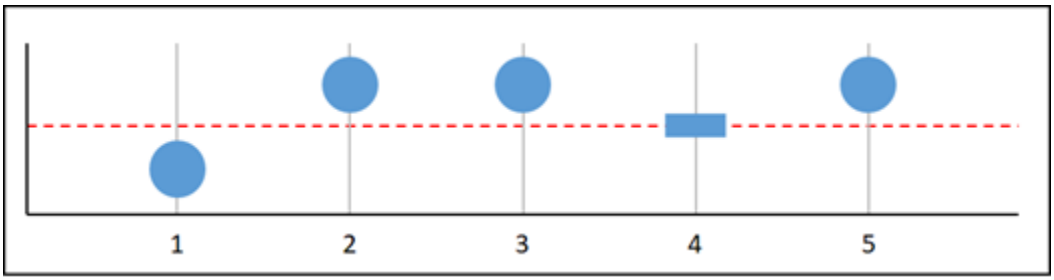
그래프 C



앞의 그래프로 표시된 지표에서 모든 데이터 포인트가 누락되었습니다. 평가 범위에서 모든 데이터 포인트가 누락되었으므로 이 지표에는 누락 데이터 포인트가 3개 있습니다. 누락 데이터 포인트를 다음과 같이 처리하도록 경보를 구성한 경우:

- 위반하지 않음 - 경보가 OK 상태입니다.
- 위반 - 경보가 ALARM 상태입니다.
- 무시 - 경보가 현재 상태를 유지합니다.
- 누락 - 경보가 INSUFFICIENT_DATA 상태입니다.

그래프 D



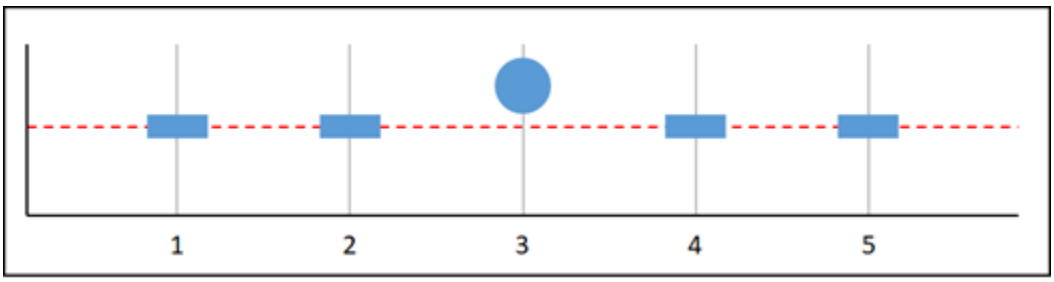
앞의 그래프로 표시된 지표에서 데이터 포인트 1은 임계값 내에 있고, 데이터 포인트 2는 위반이고, 데이터 포인트 3은 위반이고, 데이터 포인트 4는 누락되었고, 데이터 포인트 5는 위반입니다. 평가 범위에 유효한 데이터 포인트가 4개 있으므로 이 지표에는 누락 데이터 포인트가 없습니다. 누락 데이터 포인트를 다음과 같이 처리하도록 경보를 구성한 경우:

- 위반하지 않음 - 경보가 ALARM 상태입니다.
- 위반 - 경보가 ALARM 상태입니다.

- 무시 - 경보가 ALARM 상태입니다.
- 누락 - 경보가 ALARM 상태입니다.

이 시나리오에서는 경보가 모든 경우에 ALARM 상태가 됩니다. 실제 데이터 포인트가 충분해 누락 데이터 처리 방법에 대한 설정이 필요 없어 무시되기 때문입니다.

그래프 E

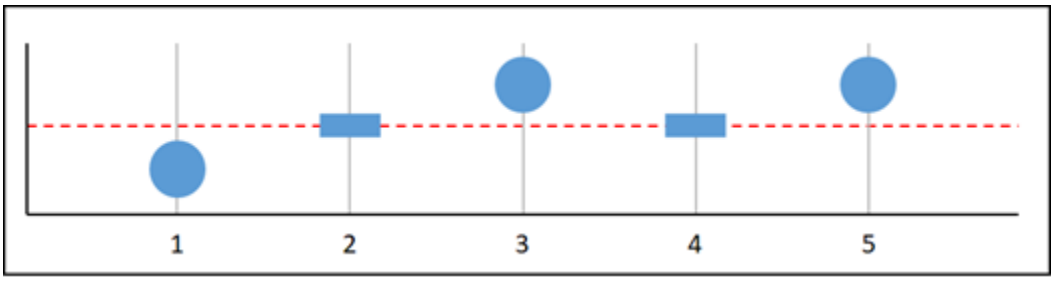


앞의 그래프로 표시된 지표에서 데이터 포인트 1과 2는 누락되었고, 데이터 포인트 3은 위반이고, 데이터 포인트 4와 5는 누락되었습니다. 평가 범위에 데이터 포인트가 하나만 있으므로 이 지표에는 누락 데이터 포인트가 두 개 있습니다. 누락 데이터 포인트를 다음과 같이 처리하도록 경보를 구성한 경우:

- 위반하지 않음 - 경보가 OK 상태입니다.
- 위반 - 경보가 ALARM 상태입니다.
- 무시 - 경보가 현재 상태를 유지합니다.
- 누락 - 경보가 ALARM 상태입니다.

그래프 F, G, H, I 및 J에서 Datapoints to alarm(경보에 대한 데이터 포인트)는 2이고 Evaluation periods(평가 기간)는 3입니다. 3중 2이며, N 경보 중 M입니다. 5는 경보의 평가 범위입니다.

그래프 F

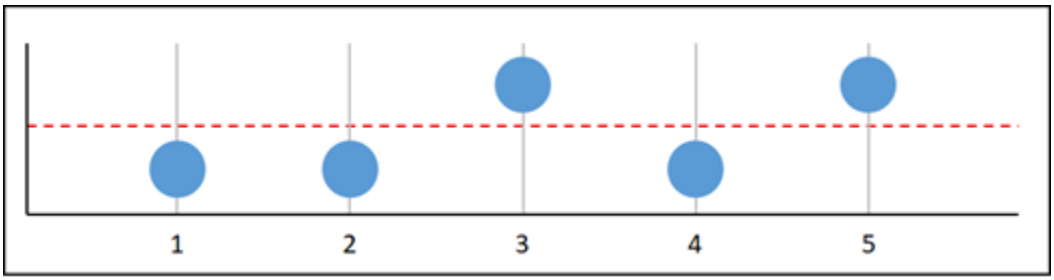


앞의 그래프로 표시된 지표에서 데이터 포인트 1은 임계값 내에 있고, 데이터 포인트 2는 누락되었고, 데이터 포인트 3은 위반이고, 데이터 포인트 4는 누락되었고, 데이터 포인트 5는 위반입니다. 평가 범

위에 데이터 포인트가 3개 있으므로 이 지표에는 누락 데이터 포인트가 없습니다. 누락 데이터 포인트를 다음과 같이 처리하도록 경보를 구성한 경우:

- 위반하지 않음 - 경보가 ALARM 상태입니다.
- 위반 - 경보가 ALARM 상태입니다.
- 무시 - 경보가 ALARM 상태입니다.
- 누락 - 경보가 ALARM 상태입니다.

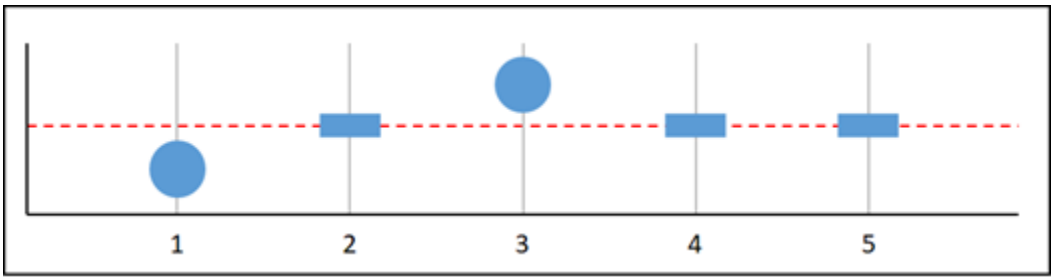
그래프 G



앞의 그래프로 표시된 지표에서 데이터 포인트 1과 2는 임계값 내에 있고, 데이터 포인트 3은 위반이고, 데이터 포인트 4는 임계값 내에 있고, 데이터 포인트 5는 위반입니다. 평가 범위에 데이터 포인트가 5개 있으므로 이 지표에는 누락 데이터 포인트가 없습니다. 누락 데이터 포인트를 다음과 같이 처리하도록 경보를 구성한 경우:

- 위반하지 않음 - 경보가 ALARM 상태입니다.
- 위반 - 경보가 ALARM 상태입니다.
- 무시 - 경보가 ALARM 상태입니다.
- 누락 - 경보가 ALARM 상태입니다.

그래프 H

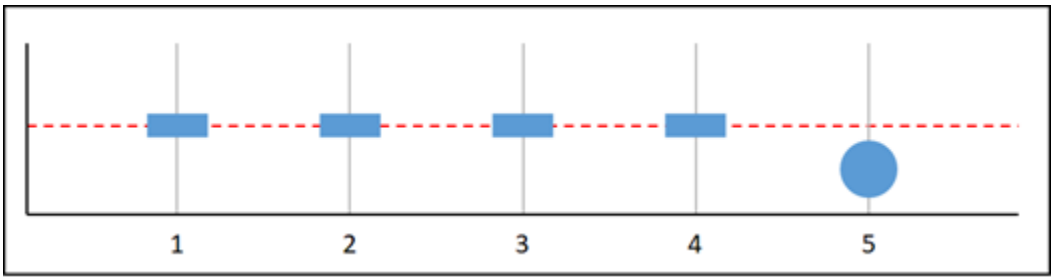


앞의 그래프로 표시된 지표에서 데이터 포인트 1은 임계값 내에 있고, 데이터 포인트 2는 누락되었고, 데이터 포인트 3은 위반이고, 데이터 포인트 4와 5는 누락되었습니다. 평가 범위에 데이터 포인트가 2

개 있으므로 이 지표에는 누락 데이터 포인트가 하나 있습니다. 누락 데이터 포인트를 다음과 같이 처리하도록 경보를 구성한 경우:

- 위반하지 않음 - 경보가 OK 상태입니다.
- 위반 - 경보가 ALARM 상태입니다.
- 무시 - 경보가 OK 상태입니다.
- 누락 - 경보가 OK 상태입니다.

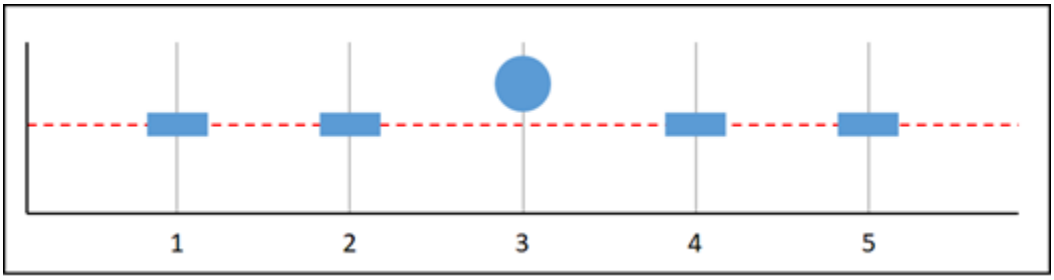
그래프 I



앞의 그래프로 표시된 지표에서 데이터 포인트 1~4는 누락되었고 데이터 포인트 5는 임계값 내에 있습니다. 평가 범위에 데이터 포인트가 하나 있으므로 이 지표에는 누락 데이터 포인트가 두 개 있습니다. 누락 데이터 포인트를 다음과 같이 처리하도록 경보를 구성한 경우:

- 위반하지 않음 - 경보가 OK 상태입니다.
- 위반 - 경보가 ALARM 상태입니다.
- 무시 - 경보가 OK 상태입니다.
- 누락 - 경보가 OK 상태입니다.

그래프 J



앞의 그래프로 표시된 지표에서 데이터 포인트 1과 2는 누락되었고, 데이터 포인트 3은 위반이고, 데이터 포인트 4와 5는 누락되었습니다. 평가 범위에 데이터 포인트가 하나 있으므로 이 지표에는 누락 데이터 포인트가 두 개 있습니다. 누락 데이터 포인트를 다음과 같이 처리하도록 경보를 구성한 경우:

- 위반하지 않음 - 경보가 OK 상태입니다.
- 위반 - 경보가 ALARM 상태입니다.
- 무시 - 경보가 현재 상태를 유지합니다.
- 누락 - 경보가 ALARM 상태입니다.

경보에 대한 자세한 정보

다음은 Lightsail에서 경보를 관리하는 데 도움을 주는 몇 가지 항목입니다.

- 인스턴스 지표 경보 생성
- 데이터베이스 지표 경보 생성
- 로드 밸런서 지표 경보 생성
- 배포 지표 경보 생성
- 지표 경보 삭제 또는 비활성화

Lightsail 인스턴스 지표 경보 생성

단일 인스턴스 지표를 감시하는 Amazon Lightsail 경보를 생성할 수 있습니다. 지정한 임계값에 상대적인 지표 값을 기준으로 경보 알림을 구성할 수 있습니다. 알림은 Lightsail 콘솔에 표시되는 배너, 이메일 주소로 전송되는 이메일, 휴대폰 번호로 전송되는 SMS 문자 메시지 형태로 사용할 수 있습니다. 경보에 대한 자세한 내용은 경보를 참조하세요.

목차

- 인스턴스 경보 제한
- 인스턴스 경보 구성 모범 사례
- 기본 경보 설정
- Lightsail 콘솔을 사용하여 인스턴스 지표 경보 생성
- Lightsail 콘솔을 사용하여 인스턴스 지표 경보 테스트
- 인스턴스 경보 생성 후의 다음 단계

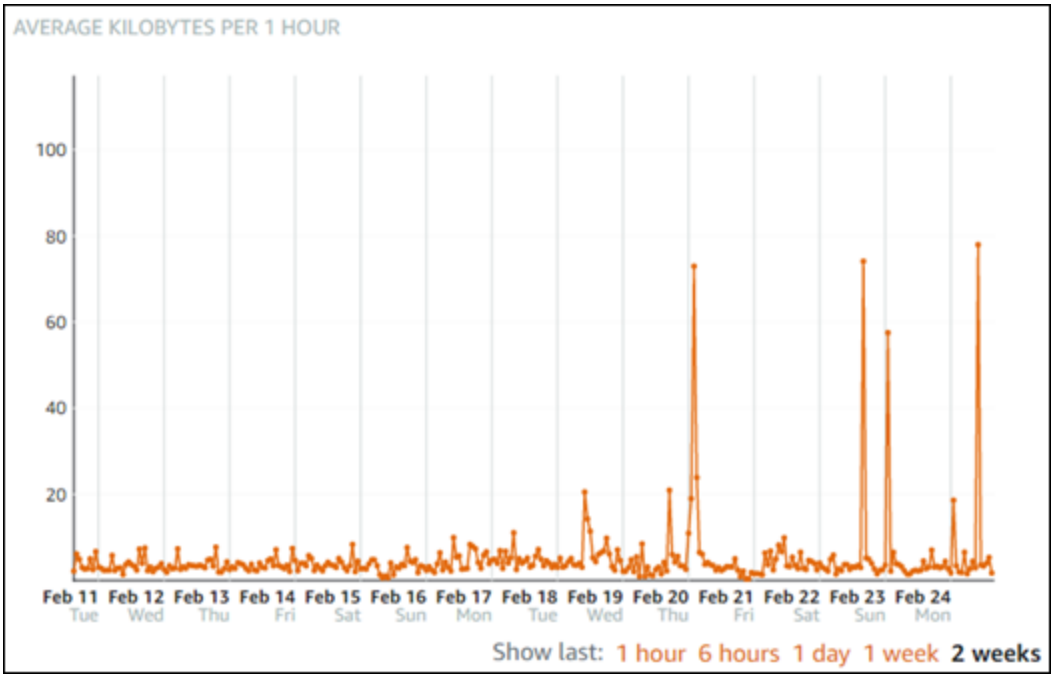
인스턴스 경보 제한

경보에는 다음과 같은 제한이 적용됩니다.

- 지표당 두 개의 경보를 구성할 수 있습니다.
- 경보는 5분 간격으로 평가되며 경보용 각 데이터 포인트는 5분간 집계된 지표 데이터를 나타냅니다.
- 경보 알림에 이메일 및/또는 SMS 문자 메시지를 사용하도록 구성한 경우에만 경보 상태가 OK로 변경될 때 알림을 받도록 경보를 구성할 수 있습니다.
- 경보 알림에 이메일 및/또는 SMS 문자 메시지를 사용하도록 구성한 경우에만 OK 경보 알림을 테스트할 수 있습니다.
- 이메일 및/또는 SMS 문자 메시지로 알리도록 경보를 구성하고 누락된 데이터 포인트에 대해 누락된 데이터 평가 안 함(Do not evaluate the missing data) 옵션을 선택한 경우에만 경보 상태가 INSUFFICIENT_DATA로 변경될 때 알리도록 경보를 구성할 수 있습니다.
- 경보가 OK 상태인 경우에만 알림을 테스트할 수 있습니다.

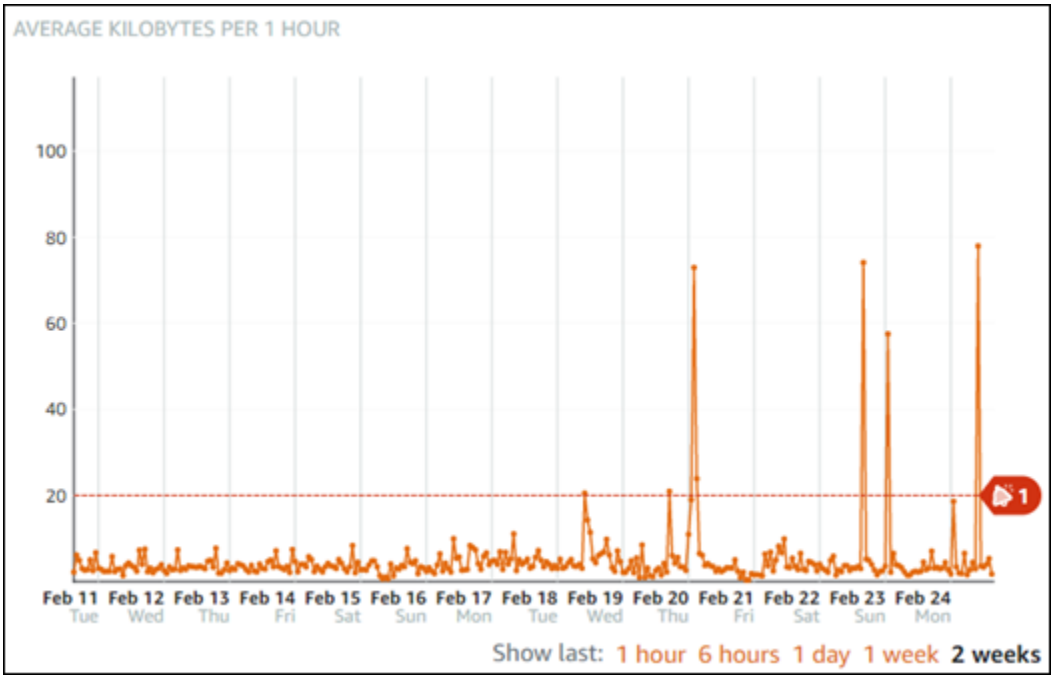
인스턴스 경보 구성 모범 사례

인스턴스에 대한 지표 경보를 구성하기 전에 지표의 기록 데이터를 확인해야 합니다. 지난 2주 동안 지표의 하위 수준, 중간 수준 및 상위 수준을 확인합니다. 다음 송신 네트워크 트래픽(NetworkOut) 지표 그래프 예에서 하위 수준은 시간당 0~10KB이고, 중간 수준은 시간당 10~20KB이며, 상위 수준은 시간당 20~80KB입니다.



경보 임계값을 하위 수준 범위(예: 시간당 5KB)보다 크거나 같게 구성하면 경보 알림이 더 자주 발생하고 불필요한 경보 알림을 받을 수 있습니다. 경보 임계값을 상위 수준 범위(예: 시간당 20KB)보다 크거나 같게 구성하면 경보 알림이 덜 자주 발생하지만 보다 중요한 경보 알림을 받을 수 있습니다. 경보를

구성하고 활성화하면 다음 예와 같이 임계값을 나타내는 경보 선이 그래프에 나타납니다. 1로 레이블이 지정된 경보 선은 경보 1의 임계값을 나타내고, 2로 레이블이 지정된 경보 선은 경보 2의 임계값을 나타냅니다.



기본 경보 설정

기본 경보 설정은 Lightsail 콘솔에 새 경보를 추가할 때 미리 채워집니다. 이것이 선택한 지표에 권장되는 경보 구성입니다. 그러나 기본 경보 구성이 리소스에 적합한지 확인해야 합니다. 예를 들어 인스턴스 송신 네트워크 트래픽(`NetworkOut`) 지표에 대한 기본 경보 임계값은 지난 10분간 2회 0바이트 이하입니다. 그러나 트래픽이 많은 경우 알림을 받고 싶다면 지난 10분간 2회 50KB 이상으로 경보 임곗값을 수정하거나, 트래픽이 없거나 많을 때 알림을 받도록 이러한 설정을 사용하여 두 번째 경보를 추가하는 것이 좋을 수 있습니다. 지정한 임계값은 이 설명서의 인스턴스 경보 구성 모범 사례 단원에 설명된 대로 지표 상위 수준 및 하위 수준에 맞추어 조정해야 합니다.

Lightsail 콘솔을 사용하여 인스턴스 지표 경보 생성

Lightsail 콘솔을 사용하여 인스턴스 지표 경보를 생성하려면 다음 단계를 수행하십시오.

1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 인스턴스를 선택합니다.
3. 경보를 생성하려는 인스턴스의 이름을 선택합니다.
4. 인스턴스 관리 페이지에서 지표 탭을 선택합니다.

5. Metrics Graphs(지표 그래프) 머리글 아래의 드롭다운 메뉴에서 경보를 생성하려는 지표를 선택합니다. 자세한 내용은 리소스 지표를 참조하세요.
6. 페이지의 경보(Alarms) 섹션에서 경보 추가(Add alarm)를 선택합니다.
7. 드롭다운 메뉴에서 비교 연산자 값을 선택합니다. greater than or equal to(크거나 같음), greater than(보다 큼), less than(보다 작음), less than or equal to(작거나 같음) 값이 있습니다.
8. 경보에 대한 임계값을 입력합니다.
9. 경보용 데이터 포인트를 입력합니다.
10. 평가 기간을 선택합니다. 기간은 5분에서 24시간까지 5분 단위로 지정할 수 있습니다.
11. 다음 알림 방법 중 하나를 선택합니다.

    - Email(이메일) - 경보 상태가 ALARM으로 변경되면 이메일로 알림을 받습니다.
    - SMS text message(SMS 문자 메시지) - 경보 상태가 ALARM으로 변경되면 SMS 문자 메시지로 알림을 받습니다. SMS 메시징은 Lightsail 리소스를 생성할 수 있는 AWS 리전 중 일부에서 지원되지 않으며, 일부 국가/지역에는 SMS 문자 메시지를 전송할 수 없습니다. 자세한 내용은 SMS 문자 메시지 지원을 참조하세요.

    > **Note**
    >
    > 이메일 또는 SMS로 알림을 받도록 선택했지만 해당 리소스의 AWS 리전에서 알림 연락처를 아직 구성하지 않은 경우 이메일 주소 또는 휴대폰 번호를 추가해야 합니다. 자세한 내용은 지표 알림을 참조하세요.

12. (선택 사항) 경보 상태가 OK로 변경될 때 알림을 받으려면 Send me a notification when the alarm state change to OK(경보 상태가 OK로 변경될 때 알림 받기)를 선택합니다. 이 옵션은 이메일 또는 SMS 문자 메시지로 알림을 받도록 선택한 경우에만 사용할 수 있습니다.
13. (선택 사항) 고급 설정을 선택하고 다음 옵션 중 하나를 선택합니다.

    - 경보가 누락된 데이터를 처리하는 방법을 선택합니다. 다음 옵션을 사용할 수 있습니다.
      - Assume it's not within the threshold (Breaching threshold)(임계값을 벗어난 것으로 간주(임계값 위반)) - 누락 데이터 포인트가 “불량”이고 임계값을 위반하는 것으로 처리됩니다.
      - Assume it's within the threshold (Not breaching threshold)(임계값 내에 있는 것으로 간주(임계값을 위반하지 않음)) - 누락 데이터 포인트가 “양호”이고 임계값 내에 있는 것으로 처리됩니다.
      - 마지막으로 양호한 데이터 포인트 값 사용(현재 경보 상태를 무시하고 유지 - 현재 경보 상태가 유지됩니다.

- Do not evaluate it (Treat missing data as missing)(평가 안 함(누락 데이터를 누락으로 처리)) - 상태 변경 여부를 평가할 때 경보에서 누락 데이터 포인트를 고려하지 않습니다.

- 경보 상태가 INSUFFICIENT_DATA로 변경될 때 알림을 받으려면 Send a notification if there is insufficient data(데이터가 불충분할 경우 알림 받기)를 선택합니다. 이 옵션은 이메일 또는 SMS 문자 메시지로 알림을 받도록 선택한 경우에만 사용할 수 있습니다.

14. 생성을 선택하여 경보를 추가합니다.

    나중에 경보를 편집하려면 편집할 경보 옆에 있는 줄임표 아이콘(:)을 선택하고 경보 편집을 선택합니다.

Lightsail 콘솔을 사용하여 인스턴스 지표 경보 테스트

Lightsail 콘솔을 사용하여 경보를 테스트하려면 다음 단계를 수행하십시오. 경보를 테스트하여 경보가 트리거될 때 이메일 또는 SMS 문자 메시지가 수신되는지를 확인하는 등 구성된 알림 옵션이 작동하는지 확인할 수 있습니다.

1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 인스턴스를 선택합니다.
3. 경보를 테스트하려는 인스턴스의 이름을 선택합니다.
4. 인스턴스 관리 페이지에서 지표 탭을 선택합니다.
5. Metrics Graphs(지표 그래프) 머리글 아래의 드롭다운 메뉴에서 경보를 테스트하려는 지표를 선택합니다.
6. 페이지의 경보 섹션까지 아래로 스크롤하고 테스트할 경보 옆에 있는 줄임표 아이콘(:)을 선택합니다.
7. 다음 옵션 중 하나를 선택하세요.

    - 경보 알림 테스트 - 경보 상태가 ALARM으로 변경될 때의 알림을 테스트하려면 이 옵션을 선택합니다.
    - OK 알림 테스트 - 경보 상태가 OK로 변경될 때의 알림을 테스트하려면 이 옵션을 선택합니다.

> **Note**
>
> 이러한 옵션을 사용할 수 없는 경우 경보에 대한 알림 옵션을 구성하지 않았거나 경보가 현재 ALARM 상태일 수 있습니다. 자세한 내용은 인스턴스 경보 제한을 참조하십시오.

선택한 테스트 옵션에 따라 경보가 일시적으로 ALARM 또는 OK 상태로 변경되며 경보에 구성한 알림 방법에 따라 이메일 및/또는 SMS 문자 메시지가 전송됩니다. 알림 배너는 ALARM 알림을 테스트하도록 선택한 경우에만 Lightsail 콘솔에 표시되며, OK 알림을 테스트하도록 선택한 경우 표시되지 않습니다. 몇 초 후에 경보가 실제 상태로 돌아옵니다.

다음 단계

인스턴스 경보에 대해 수행할 수 있는 몇 가지 추가 작업이 있습니다.

- 알림 수신을 중지하려면 Lightsail에서 이메일과 휴대폰 번호를 삭제합니다. 자세한 내용은 알림 연락처 삭제를 참조하세요. 또한 경보를 비활성화하거나 삭제하여 특정 경보에 대한 알림 수신을 중지할 수도 있습니다. 자세한 내용은 지표 경보 삭제 또는 비활성화를 참조하세요.

Lightsail 지표 경보 삭제 또는 비활성화

Amazon Lightsail 경보를 삭제하여 경보로 모니터링되는 지표가 임계값을 초과할 때 받는 알림을 중지할 수 있습니다. 또한 경보를 비활성화하여 알림 수신을 중지할 수도 있습니다. 자세한 내용은 경보 섹션을 참조하세요.

목차

- Lightsail 콘솔을 사용하여 지표 경보 삭제
- Lightsail 콘솔을 사용하여 지표 경보 비활성화 및 활성화

Lightsail 콘솔을 사용하여 지표 경보 삭제

Lightsail 콘솔을 사용하여 지표 경보를 삭제하려면 다음 단계를 수행하십시오.

1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 인스턴스, 데이터베이스 또는 네트워킹을 선택합니다.
3. 경보를 삭제하려는 리소스(인스턴스, 데이터베이스 또는 로드 밸런서)의 이름을 선택합니다.
4. 리소스 관리 페이지에서 지표 탭을 선택합니다.
5. Metrics Graphs(지표 그래프) 머리글 아래의 드롭다운 메뉴에서 경보를 삭제하려는 지표를 선택합니다.

6. 페이지의 경보 섹션까지 아래로 스크롤하고 삭제할 경보 옆에 있는 줄임표 아이콘(:)을 선택합니다.
7. 삭제를 선택합니다.
8. 프롬프트에서 삭제를 선택하여 경보 삭제를 확인합니다.

Lightsail 콘솔을 사용하여 지표 경보 비활성화 및 활성화

Lightsail 콘솔을 사용하여 지표 경보를 비활성화하려면 다음 단계를 수행하십시오.

1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 인스턴스, 데이터베이스 또는 네트워킹을 선택합니다.
3. 경보를 비활성화하려는 리소스(인스턴스, 데이터베이스 또는 로드 밸런서)의 이름을 선택합니다.
4. 리소스 관리 페이지에서 지표 탭을 선택합니다.
5. Metrics Graphs(지표 그래프) 머리글 아래의 드롭다운 메뉴에서 경보를 비활성화하려는 지표를 선택합니다.
6. 페이지의 Alarms(경보) 섹션까지 아래로 스크롤하고 비활성화할 경보를 찾아 토글을 선택하여 비활성화합니다. 마찬가지로, 비활성화된 경우 토글을 선택하여 활성화합니다.

Lightsail 버킷 성능 및 사용량 모니터링

Amazon Lightsail 객체 스토리지 서비스에서 버킷을 생성한 후 버킷 관리 페이지의 지표 탭에서 지표 그래프를 볼 수 있습니다. 지표 모니터링은 버킷의 가용성 및 성능을 유지하는 데 중요한 부분입니다. 버킷에서 지표 데이터를 정기적으로 모니터링하고 수집하면 필요할 때 버킷의 스토리지 공간과 네트워크 전송 할당량을 확대하거나 축소할 수 있습니다. 지표에 대한 자세한 내용은 리소스 지표를 참조하세요.

리소스를 모니터링할 때는 사용자 환경에서 정상적인 리소스 성능에 대한 기준을 설정해야 합니다. 그런 다음 리소스가 지정된 임계값을 벗어날 때 알림을 받도록 Lightsail 콘솔에서 경보를 구성할 수 있습니다. 자세한 내용은 알림 및 경보를 참조하세요.

버킷 지표

사용할 수 있는 버킷 지표는 아래와 같습니다.

- 버킷 크기 - 버킷에 저장된 데이터의 양입니다. 이 값은 버킷에 대한 모든 불완전 멀티파트 업로드의 모든 파트 크기를 포함하여 버킷(현재 객체 및 현재가 아닌 객체 모두)의 모든 객체 크기를 합산하여 계산됩니다.
- 객체 수 - 버킷에 저장된 총 객체 수입니다. 이 값은 버킷(현재 객체 및 현재가 아닌 객체 모두)의 모든 객체 수와 버킷에 대한 모든 불완전 멀티파트 업로드의 총 파트 수를 합산하여 계산됩니다.

> **Note**
>
> 버킷이 비어 있으면 버킷 지표 데이터가 보고되지 않습니다.

Lightsail 콘솔에서 버킷 지표 보기

Lightsail 콘솔에서 버킷 지표를 보려면 다음 절차를 완료하세요.

1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 스토리지를 선택합니다.
3. 지표를 보려는 버킷의 이름을 선택합니다.
4. 버킷 관리 페이지에서 지표(Metrics) 탭을 선택합니다.
5. 지표 그래프(Metrics graphs) 머리글 아래의 드롭다운 메뉴에서 보려는 지표를 선택합니다.

   선택한 지표에 대한 데이터 포인트가 그래프에 시각적으로 표시됩니다.

   #### TBD

   지표 그래프에 대해 다음 작업을 수행할 수 있습니다.

   - 1시간, 6시간, 1일, 1주 및 2주 동안의 데이터를 표시하도록 그래프 보기를 변경합니다.
   - 데이터 포인트에 커서를 놓으면 해당 데이터 포인트에 대한 자세한 정보를 볼 수 있습니다.
   - 선택한 지표가 지정한 임계값을 초과할 때 알림을 받도록 지표에 대한 경보를 추가합니다. 자세한 내용은 경보 및 버킷 지표 경보 생성을 참조하세요.

버킷 및 객체 관리

다음은 Lightsail 객체 스토리지 버킷을 관리하는 일반적인 단계입니다.

1. Amazon Lightsail 객체 스토리지 서비스의 객체와 버킷에 대해 알아보세요. 자세한 내용은 Amazon Lightsail의 객체 스토리지를 참조하세요.
2. Amazon Lightsail에서 버킷에 지정할 수 있는 이름에 대해 알아봅니다. 자세한 내용은 Amazon Lightsail의 버킷 이름 지정 규칙을 참조하세요.
3. 버킷을 생성하여 Lightsail 객체 스토리지 서비스를 시작합니다. 자세한 내용은 Amazon Lightsail에서 버킷 생성을 참조하세요.
4. 버킷의 보안 모범 사례와 버킷에 구성할 수 있는 액세스 권한에 대해 알아봅니다. 버킷의 모든 객체를 공개 또는 비공개로 설정하거나 개별 객체를 공개로 설정하도록 선택할 수 있습니다. 액세스 키를 생성하고, 버킷에 인스턴스를 연결하고, 다른 AWS 계정에 액세스 권한을 부여하여 버킷에 대한 액세스 권한을 부여할 수도 있습니다. 자세한 내용은 Amazon Lightsail 객체 스토리지에 대한 보안 모범 사례 및 Amazon Lightsail의 버킷 권한 이해를 참조하세요.

   버킷 액세스 권한에 대해 알아본 후 다음 가이드를 참조하여 버킷에 대한 액세스 권한을 부여합니다.

   - Amazon Lightsail의 버킷에 대한 퍼블릭 액세스 차단
   - Amazon Lightsail에서 버킷 액세스 권한 구성
   - Amazon Lightsail에서 버킷의 개별 객체에 대한 액세스 권한 구성
   - Amazon Lightsail에서 버킷의 액세스 키 생성
   - Amazon Lightsail에서 버킷에 대한 리소스 액세스 구성
   - Amazon Lightsail에서 버킷의 크로스 계정 액세스 구성
5. 버킷에 대한 액세스 로깅을 활성화하는 방법과 액세스 로그를 사용하여 버킷의 보안을 감사하는 방법에 대해 알아봅니다. 자세한 내용은 다음 안내서를 참조하세요.
   - Amazon Lightsail 객체 스토리지 서비스의 버킷에 대한 액세스 로깅
   - Amazon Lightsail 객체 스토리지 서비스에서 버킷에 대한 액세스 로그 형식
   - Amazon Lightsail 객체 스토리지 서비스에서 버킷에 대한 액세스 로깅 활성화
   - Amazon Lightsail에서 버킷에 대한 액세스 로그 사용하여 요청을 식별
6. Lightsail에서 버킷을 관리할 수 있는 기능을 사용자에게 부여하는 IAM 정책을 생성합니다. 자세한 내용은 Amazon Lightsail에서 버킷을 관리하기 위한 IAM 정책을 참조하세요.
7. 버킷의 객체에 라벨을 지정하고 식별하는 방식에 대해 알아봅니다. 자세한 내용은 Amazon Lightsail에서 객체 키 이름 이해를 참조하세요.
8. 파일을 업로드하고 버킷의 객체를 관리하는 방법에 대해 알아봅니다. 자세한 내용은 다음 안내서를 참조하세요.
   - Amazon Lightsail에서 버킷으로 파일 업로드

- 멀티파트 업로드를 사용하여 Amazon Lightsail에서 버킷으로 파일 업로드
- Amazon Lightsail에서 버킷의 객체 확인
- Amazon Lightsail에서 버킷의 객체 복사 또는 이동
- Amazon Lightsail에서 버킷의 객체 다운로드
- Amazon Lightsail에서 버킷의 객체 필터링
- Amazon Lightsail에서 버킷의 객체 태그 지정
- Amazon Lightsail에서 버킷의 객체 삭제

9. 객체 버전 관리를 활성화하면 버킷에 저장된 모든 버전의 객체를 모두 보존, 검색 및 복원할 수 있습니다. 자세한 내용은 Amazon Lightsail에서 버킷의 객체 버전 사용 설정 및 사용 중지를 참조하세요.
10. 객체 버전 관리를 활성화한 후 버킷에 있는 객체의 이전 버전을 복원할 수 있습니다. 자세한 내용은 Amazon Lightsail에서 버킷의 이전 객체 버전 복원을 참조하세요.
11. 버킷 사용률을 모니터링합니다. 자세한 내용은 Amazon Lightsail에서 버킷에 대한 지표 보기를 참조하세요.
12. 버킷 사용률이 임곗값을 초과할 때 알림을 받도록 버킷 지표에 대한 경보를 구성합니다. 자세한 내용은 Amazon Lightsail에서 버킷 지표 경보 생성을 참조하세요.
13. 스토리지 및 네트워크 전송이 부족한 경우 버킷의 스토리지 플랜을 변경합니다. 자세한 내용은 Amazon Lightsail에서 버킷 플랜 변경을 참조하세요.
14. 버킷을 다른 리소스에 연결하는 방법에 대해 알아봅니다. 자세한 내용은 다음 자습서를 참조하세요.
    - 자습서: WordPress 인스턴스를 Amazon Lightsail 버킷에 연결
    - 자습서: Lightsail 콘텐츠 전송 네트워크 배포와 함께 Amazon Lightsail 버킷 사용
15. 버킷을 더 이상 사용하지 않는 경우 삭제할 수 있습니다. 자세한 내용은 Amazon Lightsail에서 버킷 삭제를 참조하세요.

주제

- 지표 경보로 Lightsail 버킷 스토리지 모니터링

지표 경보로 Lightsail 버킷 스토리지 모니터링

단일 버킷 지표를 감시하는 Amazon Lightsail 경보를 생성할 수 있습니다. 지정한 임계값에 상대적인 지표 값을 기준으로 경보 알림을 구성할 수 있습니다. 알림은 Lightsail 콘솔에 표시되는 배너, 이메일 주소로 전송되는 이메일, 휴대폰 번호로 전송되는 SMS 문자 메시지 형태로 사용할 수 있습니다. 경보에 대한 자세한 내용은 경보를 참조하세요.

목차

- 버킷 경보 제한
- 버킷 경보 구성 모범 사례
- 기본 경보 설정
- Lightsail 콘솔을 사용하여 버킷 지표 경보 생성
- Lightsail 콘솔을 사용하여 버킷 지표 경보 테스트
- 버킷 경보 생성 후의 다음 단계

버킷 경보 제한

경보에는 다음과 같은 제한이 적용됩니다.

- 지표당 두 개의 경보를 구성할 수 있습니다.
- 경보는 5분 간격으로 평가되며 경보용 각 데이터 포인트는 5분간 집계된 지표 데이터를 나타냅니다.
- 경보 알림에 이메일 및/또는 SMS 문자 메시지를 사용하도록 구성한 경우에만 경보 상태가 OK로 변경될 때 알림을 받도록 경보를 구성할 수 있습니다.
- 경보 알림에 이메일 및/또는 SMS 문자 메시지를 사용하도록 구성한 경우에만 OK 경보 알림을 테스트할 수 있습니다.
- 이메일 및/또는 SMS 문자 메시지로 알리도록 경보를 구성하고 누락된 데이터 포인트에 대해 누락된 데이터 평가 안 함(Do not evaluate the missing data) 옵션을 선택한 경우에만 경보 상태가 `INSUFFICIENT_DATA`로 변경될 때 알리도록 경보를 구성할 수 있습니다.
- 경보가 OK 상태인 경우에만 알림을 테스트할 수 있습니다.

버킷 경보 구성 모범 사례

버킷에 대한 지표 경보를 구성하기 전에 알림을 받을 대상을 결정해야 합니다. 예를 들어, 버킷 크기 지표를 염두에 두고 버킷이 거의 가득 찼을 때 알림을 받을 수 있습니다. 현재 버킷 플랜에 5GB의 스토리지 저장 공간이 포함되어 있는 경우 버킷 크기가 4.5GB에 도달할 때에 맞춰 지표 경보를 구성할 수 있습니다. 이 경우 버킷 플랜의 사이즈를 확장할 수 있는 충분한 시간이 필요하다는 알림을 받게 됩니다.

기본 경보 설정

기본 경보 설정은 Lightsail 콘솔에 새 경보를 추가할 때 미리 채워집니다. 이것이 선택한 지표에 권장되는 경보 구성입니다. 그러나 기본 경보 구성이 리소스에 적합한지 확인해야 합니다. 예를 들어, 버킷 크기 바이트 지표의 기본 경보 임곗값은 75GB보다 크거나 같습니다. 그러나 5GB의 스토리지 저장 공간

만 포함되도록 구성된 경우 버킷에 비해 요청 임계값이 너무 높을 수 있습니다. 이 경우 경보 임계값을 4.5GB보다 크거나 같게 수정할 수 있습니다.

Lightsail 콘솔을 사용하여 버킷 지표 경보 생성

Lightsail 콘솔을 사용하여 버킷 지표 경보를 생성하려면 다음 단계를 완료합니다.

1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 스토리지를 선택합니다.
3. 경보를 생성하려는 버킷 이름을 선택합니다.
4. 버킷 관리 페이지에서 지표(Metrics) 탭을 선택합니다.
5. Metrics Graphs(지표 그래프) 머리글 아래의 드롭다운 메뉴에서 경보를 생성하려는 지표를 선택합니다. 자세한 내용은 리소스 지표를 참조하세요.
6. 페이지의 경보(Alarms) 섹션에서 경보 추가(Add alarm)를 선택합니다.
7. 드롭다운 메뉴에서 비교 연산자 값을 선택합니다. greater than or equal to(크거나 같음), greater than(보다 큼), less than(보다 작음), less than or equal to(작거나 같음) 값이 있습니다.
8. 경보에 대한 임계값을 입력합니다.
9. 경보용 데이터 포인트를 입력합니다.
10. 평가 기간을 선택합니다. 기간은 5분에서 24시간까지 5분 단위로 지정할 수 있습니다.
11. 다음 알림 방법 중 하나를 선택합니다.

    - Email(이메일) - 경보 상태가 ALARM으로 변경되면 이메일로 알림을 받습니다.
    - SMS text message(SMS 문자 메시지) - 경보 상태가 ALARM으로 변경되면 SMS 문자 메시지로 알림을 받습니다. SMS 메시징은 AWS 리전중 일부에서 지원되지 않으며, 일부 국가/리전에는 SMS 문자 메시지를 전송할 수 없습니다. 자세한 내용은 SMS 문자 메시지 지원을 참조하세요.

    > **Note**
    >
    > 이메일 또는 SMS로 알림을 받도록 선택했지만 해당 리소스의 AWS 리전에서 알림 연락처를 아직 구성하지 않은 경우 이메일 주소 또는 휴대폰 번호를 추가해야 합니다. 자세한 내용은 알림을 참조하세요.

12. (선택 사항) 경보 상태가 OK로 변경될 때 알림을 받으려면 Send me a notification when the alarm state change to OK(경보 상태가 OK로 변경될 때 알림 받기)를 선택합니다. 이 옵션은 이메일 또는 SMS 문자 메시지로 알림을 받도록 선택한 경우에만 사용할 수 있습니다.

13. (선택 사항) 고급 설정을 선택하고 다음 옵션 중 하나를 선택합니다.

    - 경보가 누락 데이터를 처리하는 방식을 선택합니다. 다음 옵션을 사용할 수 있습니다.
      - Assume it's not within the threshold (Breaching threshold)(임계값을 벗어난 것으로 간주(임계값 위반)) - 누락 데이터 포인트가 “불량”이고 임계값을 위반하는 것으로 처리됩니다.
      - Assume it's within the threshold (Not breaching threshold)(임계값 내에 있는 것으로 간주(임계값을 위반하지 않음)) - 누락 데이터 포인트가 “양호”이고 임계값 내에 있는 것으로 처리됩니다.
      - 마지막으로 양호한 데이터 포인트 값 사용(현재 경보 상태를 무시하고 유지 - 현재 경보 상태가 유지됩니다.
      - Do not evaluate it (Treat missing data as missing)(평가 안 함(누락 데이터를 누락으로 처리)) - 상태 변경 여부를 평가할 때 경보에서 누락 데이터 포인트를 고려하지 않습니다.
    - 경보 상태가 INSUFFICIENT_DATA로 변경될 때 알림을 받으려면 Send a notification if there is insufficient data(데이터가 불충분할 경우 알림 받기)를 선택합니다. 이 옵션은 이메일 또는 SMS 문자 메시지로 알림을 받도록 선택한 경우에만 사용할 수 있습니다.

14. 생성을 선택하여 경보를 추가합니다.

    나중에 경보를 편집하려면 편집할 경보 옆에 있는 줄임표 아이콘(⋮)을 선택하고 경보 편집을 선택합니다.

Lightsail 콘솔을 사용하여 버킷 지표 경보 테스트

Lightsail 콘솔을 사용하여 경보를 테스트하려면 다음 단계를 수행하십시오. 경보를 테스트하여 경보가 트리거될 때 이메일 또는 SMS 문자 메시지가 수신되는지를 확인하는 등 구성된 알림 옵션이 작동하는지 확인할 수 있습니다.

1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 스토리지를 선택합니다.
3. 경보를 테스트하려는 버킷의 이름을 선택합니다.
4. 버킷 관리 페이지에서 지표(Metrics) 탭을 선택합니다.
5. Metrics Graphs(지표 그래프) 머리글 아래의 드롭다운 메뉴에서 경보를 테스트하려는 지표를 선택합니다.
6. 페이지의 경보 섹션까지 아래로 스크롤하고 테스트할 경보 옆에 있는 줄임표 아이콘(⋮)을 선택합니다.
7. 다음 옵션 중 하나를 선택하세요.

- 경보 알림 테스트 - 경보 상태가 ALARM으로 변경될 때의 알림을 테스트하려면 이 옵션을 선택합니다.
- OK 알림 테스트 - 경보 상태가 OK로 변경될 때의 알림을 테스트하려면 이 옵션을 선택합니다.

> **Note**
>
> 이러한 옵션을 사용할 수 없는 경우 경보에 대한 알림 옵션을 구성하지 않았거나 경보가 현재 ALARM 상태일 수 있습니다. 자세한 내용은 버킷 경보 제한을 참조하세요.

선택한 테스트 옵션에 따라 경보가 일시적으로 ALARM 또는 OK 상태로 변경되며 경보에 구성한 알림 방법에 따라 이메일 및/또는 SMS 문자 메시지가 전송됩니다. 알림 배너는 ALARM 알림을 테스트하도록 선택한 경우에만 Lightsail 콘솔에 표시되며, OK 알림을 테스트하도록 선택한 경우 표시되지 않습니다. 몇 초 후에 경보가 실제 상태로 돌아옵니다.

버킷 경보 생성 후의 다음 단계

버킷 경보에 대해 수행할 수 있는 몇 가지 추가 작업이 있습니다.

- 알림 수신을 중지하려면 Lightsail에서 이메일과 휴대폰 번호를 삭제합니다. 자세한 내용은 알림 연락처 삭제를 참조하세요. 또한 경보를 비활성화하거나 삭제하여 특정 경보에 대한 알림 수신을 중지할 수도 있습니다. 자세한 내용은 지표 경보 삭제 또는 비활성화를 참조하세요.

Lightsail 컨테이너 서비스 리소스 사용률 모니터링

Amazon Lightsail 컨테이너 서비스를 생성한 후에는 서비스 관리 페이지의 지표 탭에서 지표 그래프를 볼 수 있습니다. 지표 모니터링은 리소스의 안정성, 가용성 및 성능을 유지하는 데 중요한 부분입니다. 리소스에서 지표 데이터를 정기적으로 모니터링하고 수집하면 다중 지점 오류가 발생할 경우 보다 쉽게 디버깅할 수 있습니다. 지표에 대한 자세한 내용은 Amazon Lightsail의 지표를 참조하세요.

리소스를 모니터링할 때는 사용자 환경에서 정상적인 리소스 성능에 대한 기준을 설정해야 합니다.

> **Note**
>
> 현재 컨테이너 서비스 지표에 대한 경보 및 알림은 지원되지 않습니다.

컨테이너 서비스 지표

사용할 수 있는 컨테이너 서비스 지표는 다음과 같습니다.

- CPU 사용률 - 컨테이너 서비스의 모든 노드에서 현재 사용 중인 컴퓨팅 단위의 평균 백분율입니다. 이 지표는 컨테이너 서비스에서 컨테이너를 실행하는 데 필요한 처리 능력을 나타냅니다.
- 메모리 사용률 - 컨테이너 서비스의 모든 노드에서 현재 사용 중인 메모리의 평균 백분율입니다. 이 지표는 컨테이너 서비스에서 컨테이너를 실행하는 데 필요한 메모리를 나타냅니다.

> **Note**
>
> 새 배포를 만들면 컨테이너 서비스의 기존 사용률 지표가 사라지고, 새로운 현재 배포에 대한 지표만 표시됩니다.

Lightsail 콘솔에서 컨테이너 서비스 지표 확인

Lightsail 콘솔에서 컨테이너 서비스 지표를 보려면 다음 절차를 완료하세요.

1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창의 이미지에서 컨테이너를 선택합니다.
3. 지표를 보려는 컨테이너 이름을 선택합니다.
4. 컨테이너 서비스 관리 페이지에서 지표(Metrics) 탭을 선택합니다.
5. 지표 그래프(Metrics graphs) 머리글 아래의 드롭다운 메뉴에서 보려는 지표를 선택합니다.

   선택한 지표에 대한 데이터 포인트가 그래프에 시각적으로 표시됩니다.
6. 지표 그래프에 대해 다음 작업을 수행할 수 있습니다.
   - 1시간, 6시간, 1일, 1주 및 2주 동안의 데이터를 표시하도록 그래프 보기를 변경합니다.
   - 데이터 포인트에 커서를 놓으면 해당 데이터 포인트에 대한 자세한 정보를 볼 수 있습니다.

   > **Note**
   >
   > 현재 컨테이너 서비스 지표에 대한 경보 및 알림은 지원되지 않습니다.

Lightsail 데이터베이스 성능 지표 모니터링

Amazon Lightsail에서 데이터베이스를 시작한 후 데이터베이스 관리 페이지의 지표 탭에서 해당 지표 그래프를 볼 수 있습니다. 지표 모니터링은 리소스의 안정성, 가용성 및 성능을 유지하는 데 중요한 부분입니다. 리소스에서 지표 데이터를 정기적으로 모니터링하고 수집하면 다중 지점 오류가 발생할 경우 보다 쉽게 디버깅할 수 있습니다. 지표에 대한 자세한 내용은 지표를 참조하세요.

리소스를 모니터링할 때는 사용자 환경에서 정상적인 리소스 성능에 대한 기준을 설정해야 합니다. 기준을 설정한 후에는 리소스가 지정된 임계값을 벗어날 때 알림을 받도록 Lightsail 콘솔에서 경보를 구성할 수 있습니다. 자세한 내용은 알림 및 경보를 참조하세요.

목차

- 데이터베이스 지표
- 데이터베이스 지표 보기
- 데이터베이스 지표 확인 후의 다음 단계

데이터베이스 지표

다음과 같은 데이터베이스 지표를 사용할 수 있습니다.

- CPU 사용률(**CPUUtilization**) - 데이터베이스에서 현재 사용 중인 CPU 사용률입니다.
- 데이터베이스 연결(**DatabaseConnections**) - 사용 중인 데이터베이스 연결 수입니다.
- 디스크 대기열 깊이(**DiskQueueDepth**) - 디스크에 액세스하기 위해 대기 중인 IO(읽기/쓰기 요청) 수입니다.
- 여유 스토리지 공간(**FreeStorageSpace**) - 사용 가능한 스토리지 공간 크기입니다.
- 네트워크 수신 처리량(**NetworkReceiveThroughput**) - 모니터링 및 복제에 사용되는 고객 데이터베이스 트래픽과 AWS 트래픽을 모두 포함하여 데이터베이스에 수신되는 네트워크 트래픽입니다.
- 네트워크 송신 처리량(**NetworkTransmitThroughput**) - 모니터링 및 복제에 사용되는 고객 데이터베이스 트래픽과 AWS 트래픽을 모두 포함하여 데이터베이스에서 송신되는 네트워크 트래픽입니다.

Lightsail 콘솔에서 데이터베이스 지표 보기

Lightsail 콘솔에서 데이터베이스 지표를 보려면 다음 단계를 수행하십시오.

1. Lightsail 콘솔에 로그인합니다.

2. 왼쪽 탐색 창에서 데이터베이스를 선택합니다.
3. 지표를 보려는 데이터베이스의 이름을 선택합니다.
4. 데이터베이스 관리 페이지에서 지표 탭을 선택합니다.
5. 지표 그래프 머리글 아래의 드롭다운 메뉴에서 보려는 지표를 선택합니다.

   선택한 지표에 대한 데이터 포인트가 그래프에 시각적으로 표시됩니다.
6. 지표 그래프에 대해 다음 작업을 수행할 수 있습니다.

   - 1시간, 6시간, 1일, 1주 및 2주 동안의 데이터를 표시하도록 그래프 보기를 변경합니다.
   - 데이터 포인트에 커서를 놓으면 해당 데이터 포인트에 대한 자세한 정보를 볼 수 있습니다.
   - 선택한 지표가 지정한 임계값을 초과할 때 알림을 받도록 지표에 대한 경보를 추가합니다. 자세한 내용은 경보 및 데이터베이스 지표 경보 생성을 참조하세요.

데이터베이스 지표 확인 후의 다음 단계

데이터베이스 지표에 대해 수행할 수 있는 몇 가지 추가 작업이 있습니다.

- 선택한 지표가 지정한 임계값을 초과할 때 알림을 받도록 지표에 대한 경보를 추가합니다. 자세한 내용은 경보 및 데이터베이스 지표 경보 생성을 참조하세요.
- 경보가 트리거되면 알림 배너가 Lightsail 콘솔에 표시됩니다. 이메일 및 SMS 문자 메시지로 알림을 받으려면 리소스를 모니터링 AWS 리전 하려는 각에서 이메일 주소와 휴대폰 번호를 알림 연락처로 추가해야 합니다. 자세한 내용은 알림 연락처 추가를 참조하세요.
- 알림 수신을 중지하려면 Lightsail에서 이메일과 휴대폰 번호를 삭제합니다. 자세한 내용은 지표 경보 삭제 또는 비활성화를 참조하세요. 또한 경보를 비활성화하거나 삭제하여 특정 경보에 대한 알림 수신을 중지할 수도 있습니다. 자세한 내용은 지표 경보 삭제 또는 비활성화를 참조하세요.

주제

- 지표 경보로 Lightsail 데이터베이스 상태 모니터링

지표 경보로 Lightsail 데이터베이스 상태 모니터링

단일 데이터베이스 지표를 감시하는 Amazon Lightsail 경보를 생성할 수 있습니다. 지정한 임계값에 상대적인 지표 값을 기준으로 경보 알림을 구성할 수 있습니다. 알림은 Lightsail 콘솔에 표시되는 배너, 이메일 주소로 전송되는 이메일, 휴대폰 번호로 전송되는 SMS 문자 메시지 형태로 사용할 수 있습니다. 경보에 대한 자세한 내용은 경보를 참조하세요.

목차

- 데이터베이스 경보 제한
- 데이터베이스 경보 구성 모범 사례
- 기본 경보 설정
- Lightsail 콘솔을 사용하여 데이터베이스 지표 경보 생성
- Lightsail 콘솔을 사용하여 데이터베이스 지표 경보 테스트
- 데이터베이스 경보 생성 후의 다음 단계

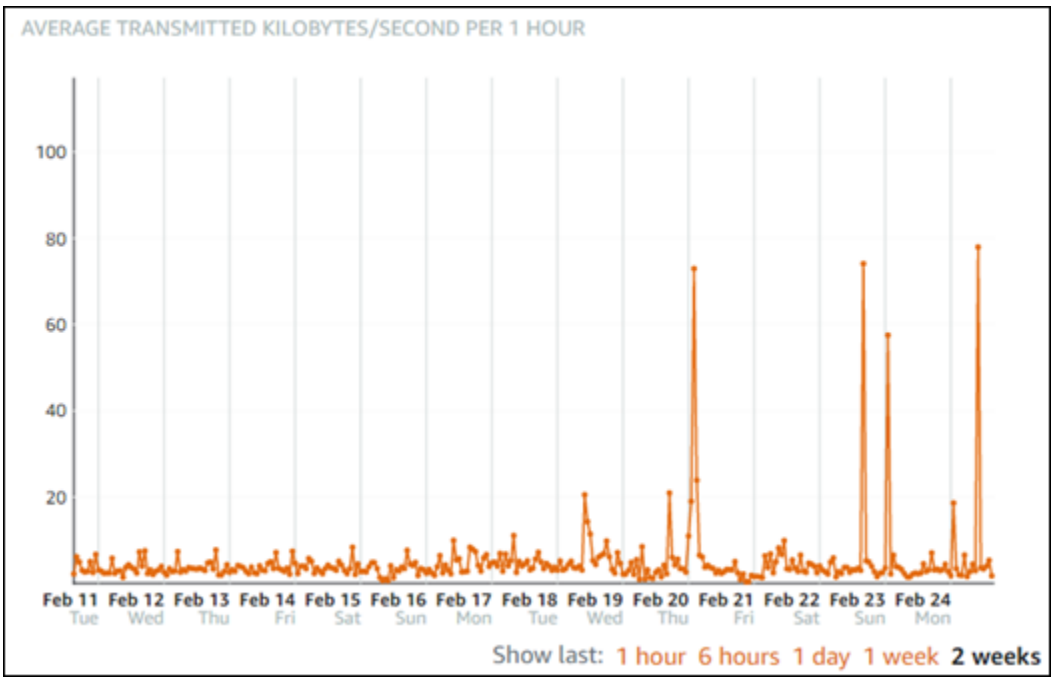
데이터베이스 경보 제한

경보에는 다음과 같은 제한이 적용됩니다.

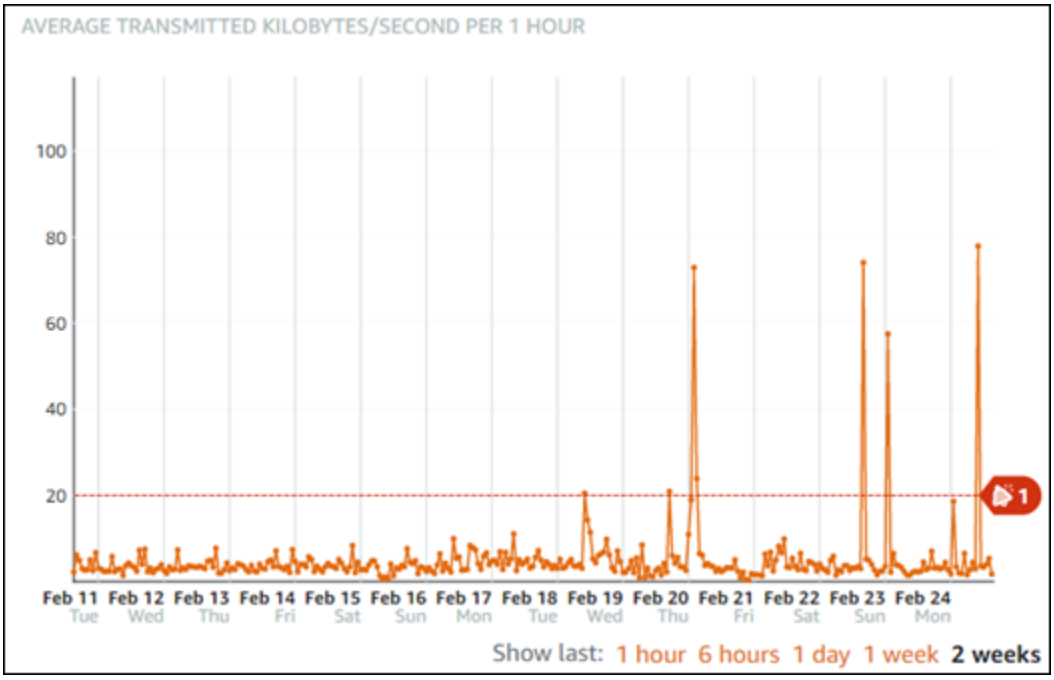
- 지표당 두 개의 경보를 구성할 수 있습니다.
- 경보는 5분 간격으로 평가되며 경보용 각 데이터 포인트는 5분간 집계된 지표 데이터를 나타냅니다.
- 경보 알림에 이메일 및/또는 SMS 문자 메시지를 사용하도록 구성한 경우에만 경보 상태가 OK로 변경될 때 알림을 받도록 경보를 구성할 수 있습니다.
- 경보 알림에 이메일 및/또는 SMS 문자 메시지를 사용하도록 구성한 경우에만 OK 경보 알림을 테스트할 수 있습니다.
- 이메일 및/또는 SMS 문자 메시지로 알리도록 경보를 구성하고 누락된 데이터 포인트에 대해 누락된 데이터 평가 안 함(Do not evaluate the missing data) 옵션을 선택한 경우에만 경보 상태가 `INSUFFICIENT_DATA`로 변경될 때 알리도록 경보를 구성할 수 있습니다.
- 경보가 OK 상태인 경우에만 알림을 테스트할 수 있습니다.

데이터베이스 경보 구성 모범 사례

데이터베이스에 대한 지표 경보를 구성하기 전에 지표의 기록 데이터를 확인해야 합니다. 지난 2주 동안 지표의 하위 수준, 중간 수준 및 상위 수준을 확인합니다. 다음 네트워크 송신 처리량(`NetworkTransmitThroughput`) 지표 그래프 예에서 하위 수준은 시간당 0~10KB/s이고, 중간 수준은 시간당 10~20KB/s이며, 상위 수준은 시간당 20~80KB/s입니다.



경보 임계값을 하위 수준 범위(예: 시간당 5KB/s)보다 크거나 같게 구성하면 경보 알림이 더 자주 발생하고 불필요한 경보 알림을 받을 수 있습니다. 경보 임계값을 상위 수준 범위(예: 시간당 20KB)보다 크거나 같게 구성하면 경보 알림이 덜 자주 발생하지만 보다 중요한 경보 알림을 받을 수 있습니다. 경보를 구성하고 활성화하면 다음 예와 같이 임계값을 나타내는 경보 선이 그래프에 나타납니다. 1로 레이블이 지정된 경보 선은 경보 1의 임계값을 나타내고, 2로 레이블이 지정된 경보 선은 경보 2의 임계값을 나타냅니다.



기본 경보 설정

기본 경보 설정은 Lightsail 콘솔에 새 경보를 추가할 때 미리 채워집니다. 이것이 선택한 지표에 권장되는 경보 구성입니다. 그러나 기본 경보 구성이 리소스에 적합한지 확인해야 합니다. 예를 들어 여유 스토리지 공간(`FreeStorageSpace`) 지표에 대한 기본 경보 임계값은 지난 5분간 1회 5바이트 미만입니다. 그러나 이 여유 스토리지 공간 임계값은 사용 중인 데이터베이스에서 너무 낮을 수 있습니다. 이 경우 지난 5분간 1회 4GB 미만으로 경보 임계값을 수정하는 것이 좋을 수 있습니다.

Lightsail 콘솔을 사용하여 데이터베이스 지표 경보 생성

Lightsail 콘솔을 사용하여 데이터베이스 지표 경보를 생성하려면 다음 단계를 수행하십시오.

1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 데이터베이스를 선택합니다.
3. 경보를 생성하려는 데이터베이스의 이름을 선택합니다.
4. 데이터베이스 관리 페이지에서 지표 탭을 선택합니다.
5. Metrics Graphs(지표 그래프) 머리글 아래의 드롭다운 메뉴에서 경보를 생성하려는 지표를 선택합니다. 자세한 내용은 리소스 지표를 참조하세요.
6. 페이지의 경보(Alarms) 섹션에서 경보 추가(Add alarm)를 선택합니다.
7. 드롭다운 메뉴에서 비교 연산자 값을 선택합니다. greater than or equal to(크거나 같음), greater than(보다 큼), less than(보다 작음), less than or equal to(작거나 같음) 값이 있습니다.
8. 경보에 대한 임계값을 입력합니다.
9. 경보용 데이터 포인트를 입력합니다.
10. 평가 기간을 선택합니다. 기간은 5분에서 24시간까지 5분 단위로 지정할 수 있습니다.
11. 다음 알림 방법 중 하나를 선택합니다.

    - Email(이메일) - 경보 상태가 ALARM으로 변경되면 이메일로 알림을 받습니다.
    - SMS text message(SMS 문자 메시지) - 경보 상태가 ALARM으로 변경되면 SMS 문자 메시지로 알림을 받습니다. SMS 메시징은 Lightsail 리소스를 생성할 수 있는 AWS 리전 중 일부에서 지원되지 않으며, 일부 국가/지역에는 SMS 문자 메시지를 전송할 수 없습니다. 자세한 내용은 SMS 문자 메시지 지원을 참조하세요.

> **Note**
>
> 이메일 또는 SMS로 알림을 받도록 선택했지만 해당 리소스의 AWS 리전에서 알림 연락처를 아직 구성하지 않은 경우 이메일 주소 또는 휴대폰 번호를 추가해야 합니다. 자세한 내용은 알림을 참조하세요.

12. (선택 사항) 경보 상태가 OK로 변경될 때 알림을 받으려면 Send me a notification when the alarm state change to OK(경보 상태가 OK로 변경될 때 알림 받기)를 선택합니다. 이 옵션은 이메일 또는 SMS 문자 메시지로 알림을 받도록 선택한 경우에만 사용할 수 있습니다.
13. (선택 사항) 고급 설정을 선택하고 다음 옵션 중 하나를 선택합니다.
    - 경보가 누락 데이터를 처리하는 방식을 선택합니다. 다음 옵션을 사용할 수 있습니다.
      - Assume it's not within the threshold (Breaching threshold)(임계값을 벗어난 것으로 간주(임계값 위반)) - 누락 데이터 포인트가 “불량”이고 임계값을 위반하는 것으로 처리됩니다.
      - Assume it's within the threshold (Not breaching threshold)(임계값 내에 있는 것으로 간주(임계값을 위반하지 않음)) - 누락 데이터 포인트가 “양호”이고 임계값 내에 있는 것으로 처리됩니다.
      - 마지막으로 양호한 데이터 포인트 값 사용(현재 경보 상태를 무시하고 유지 - 현재 경보 상태가 유지됩니다.
      - Do not evaluate it (Treat missing data as missing)(평가 안 함(누락 데이터를 누락으로 처리)) - 상태 변경 여부를 평가할 때 경보에서 누락 데이터 포인트를 고려하지 않습니다.
    - 경보 상태가 INSUFFICIENT_DATA로 변경될 때 알림을 받으려면 Send a notification if there is insufficient data(데이터가 불충분할 경우 알림 받기)를 선택합니다. 이 옵션은 이메일 또는 SMS 문자 메시지로 알림을 받도록 선택한 경우에만 사용할 수 있습니다.
14. 생성을 선택하여 경보를 추가합니다.

    나중에 경보를 편집하려면 편집할 경보 옆에 있는 줄임표 아이콘(⋮)을 선택하고 경보 편집을 선택합니다.

Lightsail 콘솔을 사용하여 데이터베이스 지표 경보 테스트

Lightsail 콘솔을 사용하여 경보를 테스트하려면 다음 단계를 수행하십시오. 경보를 테스트하여 경보가 트리거될 때 이메일 또는 SMS 문자 메시지가 수신되는지를 확인하는 등 구성된 알림 옵션이 작동하는지 확인할 수 있습니다.

1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 데이터베이스를 선택합니다.
3. 경보를 테스트하려는 데이터베이스의 이름을 선택합니다.
4. 데이터베이스 관리 페이지에서 지표 탭을 선택합니다.
5. Metrics Graphs(지표 그래프) 머리글 아래의 드롭다운 메뉴에서 경보를 테스트하려는 지표를 선택합니다.
6. 페이지의 경보 섹션까지 아래로 스크롤하고 테스트할 경보 옆에 있는 줄임표 아이콘(:)을 선택합니다.
7. 다음 옵션 중 하나를 선택하세요.

   - 경보 알림 테스트 - 경보 상태가 ALARM으로 변경될 때의 알림을 테스트하려면 이 옵션을 선택합니다.
   - OK 알림 테스트 - 경보 상태가 OK로 변경될 때의 알림을 테스트하려면 이 옵션을 선택합니다.

   > **Note**
   >
   > 이러한 옵션을 사용할 수 없는 경우 경보에 대한 알림 옵션을 구성하지 않았거나 경보가 현재 ALARM 상태일 수 있습니다. 자세한 내용은 데이터베이스 경보 제한을 참조하십시오.

   선택한 테스트 옵션에 따라 경보가 일시적으로 ALARM 또는 OK 상태로 변경되며 경보에 구성한 알림 방법에 따라 이메일 및/또는 SMS 문자 메시지가 전송됩니다. 알림 배너는 ALARM 알림을 테스트하도록 선택한 경우에만 Lightsail 콘솔에 표시되며, OK 알림을 테스트하도록 선택한 경우 표시되지 않습니다. 몇 초 후에 경보가 실제 상태로 돌아옵니다.

데이터베이스 경보 생성 후의 다음 단계

데이터베이스 경보에 대해 수행할 수 있는 몇 가지 추가 작업이 있습니다.

- 알림 수신을 중지하려면 Lightsail에서 이메일과 휴대폰 번호를 삭제합니다. 자세한 내용은 알림 연락처 삭제를 참조하세요. 또한 경보를 비활성화하거나 삭제하여 특정 경보에 대한 알림 수신을 중지할 수도 있습니다. 자세한 내용은 지표 경보 삭제 또는 비활성화를 참조하세요.

Lightsail 배포 성능 지표 모니터링

Amazon Lightsail에서 배포를 생성한 후에는 배포 관리 페이지의 지표(Metrics) 탭에서 지표 그래프를 볼 수 있습니다. 지표 모니터링은 리소스의 안정성, 가용성 및 성능을 유지하는 데 중요한 부분입니다. 리소스에서 지표 데이터를 정기적으로 모니터링하고 수집하면 다중 지점 오류가 발생할 경우 보다 쉽게 디버깅할 수 있습니다. 지표에 대한 자세한 내용은 지표를 참조하세요.

리소스를 모니터링할 때는 사용자 환경에서 정상적인 리소스 성능에 대한 기준을 설정해야 합니다. 그런 다음 리소스가 지정된 임계값을 벗어날 때 알림을 받도록 Lightsail 콘솔에서 경보를 구성할 수 있습니다. 자세한 내용은 알림 및 경보를 참조하세요.

목차

- 배포 지표
- Lightsail 콘솔에서 배포용 지표 확인
- 데이터베이스 지표 확인 후의 다음 단계

배포 지표

사용할 수 있는 배포 지표는 아래와 같습니다.

- 요청 - 모든 HTTP 메소드와 HTTP 및 HTTPS 요청 둘 다에 대해 배포에서 수신한 최종 사용자 요청의 총 수입니다.
- 업로드된 바이트 - POST 및 PUT 요청을 사용하여 배포에서 오리진으로 업로드한 바이트 수입니다.
- 다운로드된 바이트 - GET, HEAD 및 OPTIONS 요청에 대해 최종 사용자가 다운로드한 바이트 수입니다.
- 총 오류 발생률 - 응답의 HTTP 상태 코드가 4xx 또는 5xx인 모든 최종 사용자 요청의 백분율입니다.
- HTTP 4xx 오류 발생률 - 응답의 HTTP 상태 코드가 4xx인 모든 최종 사용자 요청의 백분율입니다. 이러한 경우 클라이언트 또는 클라이언트 뷰어가 오류를 일으켰을 수 있습니다. 예를 들어, 404 상태 코드(찾을 수 없음)는 클라이언트가 찾을 수 없는 객체를 요청했음을 의미합니다.
- HTTP 5xx 오류 발생률 - 응답의 HTTP 상태 코드가 5xx인 모든 최종 사용자 요청의 백분율입니다. 이러한 경우 원본 서버가 요청을 충족하지 못한 것입니다. 예를 들어, 503 상태 코드(서비스를 사용할 수 없음)는 원본 서버를 현재 사용할 수 없음을 의미합니다.

Lightsail 콘솔에서 배포용 지표 확인

Lightsail 콘솔에서 배포 지표를 보려면 다음 절차를 완료하세요.

1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 네트워킹을 선택합니다.
3. 지표를 보려는 배포의 이름을 선택합니다.
4. 배포 관리 페이지에서 지표(Metrics) 탭을 선택합니다.
5. 지표 그래프 머리글 아래의 드롭다운 메뉴에서 보려는 지표를 선택합니다.

   선택한 지표에 대한 데이터 포인트가 그래프에 시각적으로 표시됩니다.
6. 지표 그래프에 대해 다음 작업을 수행할 수 있습니다.

   - 1시간, 6시간, 1일, 1주 및 2주 동안의 데이터를 표시하도록 그래프 보기를 변경합니다.
   - 데이터 포인트에 커서를 놓으면 해당 데이터 포인트에 대한 자세한 정보를 볼 수 있습니다.
   - 선택한 지표가 지정한 임계값을 초과할 때 알림을 받도록 지표에 대한 경보를 추가합니다. 자세한 내용은 경보 및 인스턴스 지표 경보 생성을 참조하세요.

배포 지표 확인 후의 다음 단계

배포 알림에 대해 수행할 수 있는 몇 가지 추가 작업이 있습니다.

- 선택한 지표가 지정한 임계값을 초과할 때 알림을 받도록 지표에 대한 경보를 추가합니다. 자세한 내용은 경보 및 배포 지표 경보 생성을 참조하세요.
- 경보가 트리거되면 알림 배너가 Lightsail 콘솔에 표시됩니다. 이메일 및 SMS 문자 메시지로 알림을 받으려면 리소스를 모니터링 AWS 리전 하려는 각에서 이메일 주소와 휴대폰 번호를 알림 연락처로 추가해야 합니다. 자세한 내용은 알림 연락처 추가를 참조하세요.
- 알림 수신을 중지하려면 Lightsail에서 이메일과 휴대폰 번호를 삭제합니다. 자세한 내용은 지표 경보 삭제 또는 비활성화를 참조하세요. 또한 경보를 비활성화하거나 삭제하여 특정 경보에 대한 알림 수신을 중지할 수도 있습니다. 자세한 내용은 지표 경보 삭제 또는 비활성화를 참조하세요.

주제

- 지표 경보로 Lightsail 배포 상태 모니터링

지표 경보로 Lightsail 배포 상태 모니터링

단일 배포 지표를 감시하는 Amazon Lightsail 경보를 생성할 수 있습니다. 지정한 임계값에 상대적인 지표 값을 기준으로 경보 알림을 구성할 수 있습니다. 알림은 Lightsail 콘솔에 표시되는 배너, 이메일 주소로 전송되는 이메일, 휴대폰 번호로 전송되는 SMS 문자 메시지 형태로 사용할 수 있습니다. 경보에 대한 자세한 내용은 경보를 참조하세요.

목차

- 배포 경보 제한
- 배포 경보 구성 모범 사례
- 기본 경보 설정
- Lightsail 콘솔을 사용하여 배포 지표 경보 생성
- 배포 지표 경보 테스트
- 배포 경보 생성 후의 다음 단계

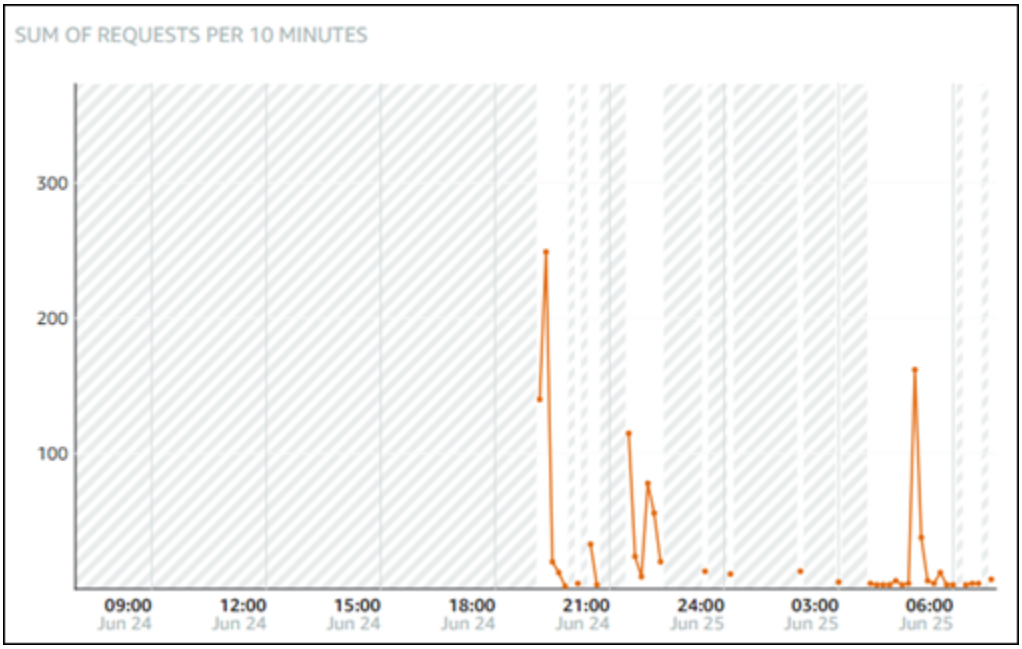
배포 경보 제한

경보에는 다음과 같은 제한이 적용됩니다.

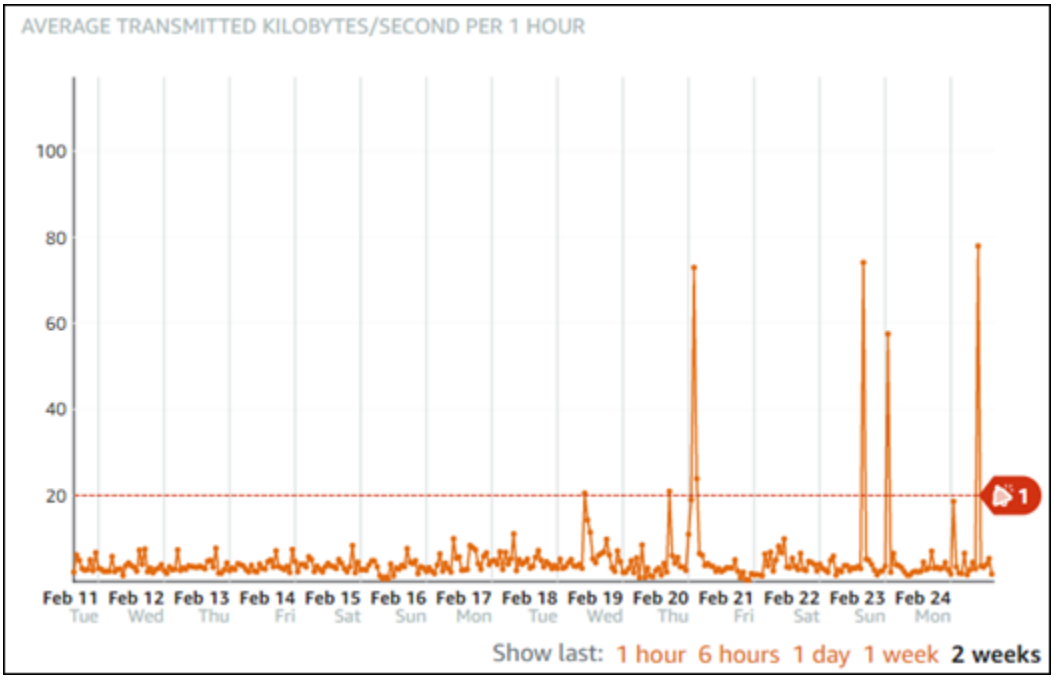
- 지표당 두 개의 경보를 구성할 수 있습니다.
- 경보는 5분 간격으로 평가되며 경보용 각 데이터 포인트는 5분간 집계된 지표 데이터를 나타냅니다.
- 경보 알림에 이메일 및/또는 SMS 문자 메시지를 사용하도록 구성한 경우에만 경보 상태가 OK로 변경될 때 알림을 받도록 경보를 구성할 수 있습니다.
- 경보 알림에 이메일 및/또는 SMS 문자 메시지를 사용하도록 구성한 경우에만 OK 경보 알림을 테스트할 수 있습니다.
- 이메일 및/또는 SMS 문자 메시지로 알리도록 경보를 구성하고 누락된 데이터 포인트에 대해 누락된 데이터 평가 안 함(Do not evaluate the missing data) 옵션을 선택한 경우에만 경보 상태가 `INSUFFICIENT_DATA`로 변경될 때 알리도록 경보를 구성할 수 있습니다.
- 경보가 OK 상태인 경우에만 알림을 테스트할 수 있습니다.

배포 경보 구성 모범 사례

배포에 대한 지표 배포 경보를 구성하기 전에 지표의 기록 데이터를 확인해야 합니다. 지난 2주 동안 지표의 하위 수준, 중간 수준 및 상위 수준을 확인합니다. 다음 요청 지표 그래프 예시에서 하위 수준은 0~10개 요청, 중간 수준은 10~50개 요청, 상위 수준은 50~250개 요청입니다.



경보 임계값을 하위 수준 범위(예: 5개 요청)보다 크거나 같게 구성하면 경보 알림이 더 자주 발생하고 불필요한 경보 알림을 받을 수 있습니다. 경보 임계값을 상위 수준 범위(예: 150개 요청)보다 크거나 같게 구성하면 경보 알림 빈도는 줄어들지만, 조사해야 할 중요한 경보 알림 위주로 받을 수 있습니다. 경보를 구성하고 활성화하면 다음 예와 같이 임계값을 나타내는 경보 선이 그래프에 나타납니다. 1로 레이블이 지정된 경보 선은 경보 1의 임계값을 나타내고, 2로 레이블이 지정된 경보 선은 경보 2의 임계값을 나타냅니다.



기본 경보 설정

기본 경보 설정은 Lightsail 콘솔에 새 경보를 추가할 때 미리 채워집니다. 이것이 선택한 지표에 권장되는 경보 구성입니다. 그러나 기본 경보 구성이 리소스에 적합한지 확인해야 합니다. 예를 들어, 요청 지표의 기본 경보 임곗값은 지난 15분간 3회 45개 요청보다 큽니다. 그러나 이 요청 임계값은 배포에 비해 너무 낮을 수 있습니다. 이 경우 지난 15분간 3회 150개 요청 이상으로 경보 임계값을 수정하는 것이 좋을 수 있습니다.

Lightsail 콘솔을 사용하여 배포 지표 경보 생성

Lightsail 콘솔을 사용하여 배포 지표 경보를 생성하려면 다음 단계를 수행하세요.

1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 네트워킹을 선택합니다.
3. 경보를 생성하려는 배포 이름을 선택합니다.
4. 배포 관리 페이지에서 지표(Metrics) 탭을 선택합니다.
5. Metrics Graphs(지표 그래프) 머리글 아래의 드롭다운 메뉴에서 경보를 생성하려는 지표를 선택합니다. 자세한 내용은 리소스 지표를 참조하세요.
6. 페이지의 경보(Alarms) 섹션에서 경보 추가(Add alarm)를 선택합니다.
7. 드롭다운 메뉴에서 비교 연산자 값을 선택합니다. greater than or equal to(크거나 같음), greater than(보다 큼), less than(보다 작음), less than or equal to(작거나 같음) 값이 있습니다.
8. 경보에 대한 임계값을 입력합니다.
9. 경보용 데이터 포인트를 입력합니다.
10. 평가 기간을 선택합니다. 기간은 5분에서 24시간까지 5분 단위로 지정할 수 있습니다.
11. 다음 알림 방법 중 하나를 선택합니다.

    - Email(이메일) - 경보 상태가 ALARM으로 변경되면 이메일로 알림을 받습니다.
    - SMS text message(SMS 문자 메시지) - 경보 상태가 ALARM으로 변경되면 SMS 문자 메시지로 알림을 받습니다. SMS 메시징은 Lightsail 리소스를 생성할 수 있는 AWS 리전 중 일부에서 지원되지 않으며, 일부 국가/지역에는 SMS 문자 메시지를 전송할 수 없습니다. 자세한 내용은 SMS 문자 메시지 지원을 참조하세요.

> **Note**
>
> 이메일 또는 SMS로 알림을 받도록 선택했지만 해당 리소스의 AWS 리전에서 알림 연락처를 아직 구성하지 않은 경우 이메일 주소 또는 휴대폰 번호를 추가해야 합니다. 자세한 내용은 알림을 참조하세요.

12. (선택 사항) 경보 상태가 OK로 변경될 때 알림을 받으려면 Send me a notification when the alarm state change to OK(경보 상태가 OK로 변경될 때 알림 받기)를 선택합니다. 이 옵션은 이메일 또는 SMS 문자 메시지로 알림을 받도록 선택한 경우에만 사용할 수 있습니다.
13. (선택 사항) 고급 설정을 선택하고 다음 옵션 중 하나를 선택합니다.
    - 경보가 누락 데이터를 처리하는 방식을 선택합니다. 다음 옵션을 사용할 수 있습니다.
      - Assume it's not within the threshold (Breaching threshold)(임계값을 벗어난 것으로 간주(임계값 위반)) - 누락 데이터 포인트가 “불량”이고 임계값을 위반하는 것으로 처리됩니다.
      - Assume it's within the threshold (Not breaching threshold)(임계값 내에 있는 것으로 간주(임계값을 위반하지 않음)) - 누락 데이터 포인트가 “양호”이고 임계값 내에 있는 것으로 처리됩니다.
      - Use the value of the last good datapoint (Ignore and maintain the current alarm state)(마지막으로 양호한 데이터 포인트 값 사용(현재 경보 상태를 무시하고 유지)) - 현재 경보 상태가 유지됩니다.
      - Do not evaluate it (Treat missing data as missing)(평가 안 함(누락 데이터를 누락으로 처리)) - 상태 변경 여부를 평가할 때 경보에서 누락 데이터 포인트를 고려하지 않습니다.
    - 경보 상태가 INSUFFICIENT_DATA로 변경될 때 알림을 받으려면 Send a notification if there is insufficient data(데이터가 불충분할 경우 알림 받기)를 선택합니다. 이 옵션은 이메일 또는 SMS 문자 메시지로 알림을 받도록 선택한 경우에만 사용할 수 있습니다.
14. 생성을 선택하여 경보를 추가합니다.

    나중에 경보를 편집하려면 편집할 경보 옆에 있는 줄임표 아이콘(⋮)을 선택하고 경보 편집을 선택합니다.

배포 지표 경보 테스트

Lightsail 콘솔을 사용하여 경보를 테스트하려면 다음 단계를 수행하십시오. 경보를 테스트하여 경보가 트리거될 때 이메일 또는 SMS 문자 메시지가 수신되는지를 확인하는 등 구성된 알림 옵션이 작동하는지 확인할 수 있습니다.

1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 네트워킹을 선택합니다.
3. 경보를 테스트하려는 배포 이름을 선택합니다.
4. 배포 관리 페이지에서 지표(Metrics) 탭을 선택합니다.
5. Metrics Graphs(지표 그래프) 머리글 아래의 드롭다운 메뉴에서 경보를 테스트하려는 지표를 선택합니다.
6. 페이지의 경보 섹션까지 아래로 스크롤하고 테스트할 경보 옆에 있는 줄임표 아이콘(:)을 선택합니다.
7. 다음 옵션 중 하나를 선택하세요.

   - 경보 알림 테스트 - 경보 상태가 ALARM으로 변경될 때의 알림을 테스트하려면 이 옵션을 선택합니다.
   - OK 알림 테스트 - 경보 상태가 OK로 변경될 때의 알림을 테스트하려면 이 옵션을 선택합니다.

   > **Note**
   >
   > 이러한 옵션을 사용할 수 없는 경우 경보에 대한 알림 옵션을 구성하지 않았거나 경보가 현재 ALARM 상태일 수 있습니다. 자세한 내용은 배포 경보 제한을 참조하세요.

   선택한 테스트 옵션에 따라 경보가 일시적으로 ALARM 또는 OK 상태로 변경되며 경보에 구성한 알림 방법에 따라 이메일 및/또는 SMS 문자 메시지가 전송됩니다. 알림 배너는 ALARM 알림을 테스트하도록 선택한 경우에만 Lightsail 콘솔에 표시되며, OK 알림을 테스트하도록 선택한 경우 표시되지 않습니다. 몇 초 후에 경보가 실제 상태로 돌아옵니다.

배포 경보 생성 후의 다음 단계

배포 경보에 대해 수행할 수 있는 몇 가지 추가 작업이 있습니다.

- 알림 수신을 중지하려면 Lightsail에서 이메일과 휴대폰 번호를 삭제합니다. 자세한 내용은 알림 연락처 삭제를 참조하세요. 또한 경보를 비활성화하거나 삭제하여 특정 경보에 대한 알림 수신을 중지할 수도 있습니다. 자세한 내용은 지표 경보 삭제 또는 비활성화를 참조하세요.

Lightsail 로드 밸런서 상태 지표 모니터링

Amazon Lightsail에서 로드 밸런서를 생성하고 인스턴스를 로드 밸런서에 연결한 후에는 로드 밸런서 관리 페이지의 지표 탭에서 해당 지표 그래프를 볼 수 있습니다. 지표 모니터링은 리소스의 안정성, 가용성 및 성능을 유지하는 데 중요한 부분입니다. 리소스에서 지표 데이터를 정기적으로 모니터링하고 수집하면 다중 지점 오류가 발생할 경우 보다 쉽게 디버깅할 수 있습니다. 지표에 대한 자세한 내용은 지표를 참조하세요.

리소스를 모니터링할 때는 사용자 환경에서 정상적인 리소스 성능에 대한 기준을 설정해야 합니다. 기준을 설정한 후에는 리소스가 지정된 임계값을 벗어날 때 알림을 받도록 Lightsail 콘솔에서 경보를 구성할 수 있습니다. 자세한 내용은 알림 및 경보를 참조하세요.

목차

- 로드 밸런서 지표
- 로드 밸런서 지표 확인
- 다음 단계

로드 밸런서 지표

다음과 같은 로드 밸런서 지표를 사용할 수 있습니다.

- 정상 호스트 수(**HealthyHostCount**) - 정상으로 간주되는 대상 인스턴스 수입니다.
- 비정상 호스트 수(**UnhealthyHostCount**) - 비정상으로 간주되는 대상 인스턴스 수입니다.
- 로드 밸런서 HTTP 4XX(**HTTPCode_LB_4XX_Count**) - 로드 밸런서에서 생성된 HTTP 4XX 클라이언트 오류 코드 수입니다. 클라이언트 오류는 요청 형식이 잘못되었거나 불완전할 때 생성됩니다. 이러한 요청은 대상 인스턴스에서 수신되지 않습니다. 대상 인스턴스에서 생성된 응답 코드는 이 숫자에 포함되지 않습니다.
- 로드 밸런서 HTTP 5XX(**HTTPCode_LB_5XX_Count**) - 로드 밸런서에서 생성된 HTTP 5XX 서버 오류 코드 수입니다. 대상 인스턴스에서 생성된 응답 코드는 여기에 포함되지 않습니다. 이 지표는 로드 밸런서에 정상 인스턴스가 연결되어 있지 않거나 요청 속도가 인스턴스 용량을 초과하거나(스필오버) 또는 로드 밸런서 용량을 초과하는 경우에 보고됩니다.
- 인스턴스 HTTP 2XX(**HTTPCode_Instance_2XX_Count**) - 대상 인스턴스에서 생성된 HTTP 2XX 응답 코드 수입니다. 단, 로드 밸런서에서 생성된 응답 코드 수는 여기에 포함되지 않습니다.
- 인스턴스 HTTP 3XX(**HTTPCode_Instance_3XX_Count**) - 대상 인스턴스에서 생성된 HTTP 3XX 응답 코드 수입니다. 단, 로드 밸런서에서 생성된 응답 코드 수는 여기에 포함되지 않습니다.

- 인스턴스HTTP 4XX(**HTTPCode_Instance_4XX_Count**) - 대상 인스턴스에서 생성된 HTTP 4XX 응답 코드 수입니다. 단, 로드 밸런서에서 생성된 응답 코드 수는 여기에 포함되지 않습니다.
- 인스턴스 HTTP 5XX(**HTTPCode_Instance_5XX_Count**) - 대상 인스턴스에서 생성된 HTTP 5XX 응답 코드 수입니다. 단, 로드 밸런서에서 생성된 응답 코드 수는 여기에 포함되지 않습니다.
- 인스턴스 응답 시간(**InstanceResponseTime**) - 로드 밸런서에서 요청을 보낸 후 대상 인스턴스로부터 응답 신호를 받을 때까지의 경과 시간(초)입니다.
- 클라이언트 TLS 협상 오류 수(**ClientTLSNegotiationErrorCount**) - 로드 밸런서에서 생성된 TLS 오류로 인해 로드 밸런서에서 세션을 설정하지 않은 클라이언트에서 시작된 TLS 연결 수입니다. 가능한 원인으로는 암호 또는 프로토콜 불일치가 있습니다.
- 요청 수(**RequestCount**) - IPv4를 통해 처리된 요청 수입니다. 로드 밸런서의 대상 인스턴스에서 응답을 생성한 요청만 이 개수에 포함됩니다.
- 거부된 연결 수(**RejectedConnectionCount**) - 로드 밸런서가 최대 연결 수에 도달하여 거부된 연결 수입니다.

로드 밸런서 지표 확인

Lightsail 콘솔에서 로드 밸런서 지표를 보려면 다음 단계를 수행하세요.

1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 네트워킹을 선택합니다.
3. 지표를 보려는 로드 밸런서의 이름을 선택합니다.
4. 로드 밸런서 관리 페이지에서 지표 탭을 선택합니다.
5. 지표 그래프 머리글 아래의 드롭다운 메뉴에서 보려는 지표를 선택합니다.

   선택한 지표에 대한 데이터 포인트가 그래프에 시각적으로 표시됩니다.
6. 지표 그래프에 대해 다음 작업을 수행할 수 있습니다.

   - 1시간, 6시간, 1일, 1주 및 2주 동안의 데이터를 표시하도록 그래프 보기를 변경합니다.
   - 데이터 포인트에 커서를 놓으면 해당 데이터 포인트에 대한 자세한 정보를 볼 수 있습니다.
   - 선택한 지표가 지정한 임계값을 초과할 때 알림을 받도록 지표에 대한 경보를 추가합니다. 자세한 내용은 경보 및 로드 밸런서 지표 경보 생성을 참조하세요.

다음 단계

로드 밸런서 지표에 대해 수행할 수 있는 몇 가지 추가 작업이 있습니다.

- 선택한 지표가 지정한 임계값을 초과할 때 알림을 받도록 지표에 대한 경보를 추가합니다. 자세한 내용은 경보 및 로드 밸런서 지표 경보 생성을 참조하세요.
- 경보가 트리거되면 알림 배너가 Lightsail 콘솔에 표시됩니다. 이메일 및 SMS 문자 메시지로 알림을 받으려면 리소스를 모니터링 AWS 리전 하려는 각에서 이메일 주소와 휴대폰 번호를 알림 연락처로 추가해야 합니다. 자세한 내용은 알림 연락처 추가를 참조하세요.
- 알림 수신을 중지하려면 Lightsail에서 이메일과 휴대폰 번호를 삭제합니다. 자세한 내용은 지표 경보 삭제 또는 비활성화를 참조하세요. 또한 경보를 비활성화하거나 삭제하여 특정 경보에 대한 알림 수신을 중지할 수도 있습니다. 자세한 내용은 지표 경보 삭제 또는 비활성화를 참조하세요.

주제

- 경보를 사용하여 Lightsail 로드 밸런서 지표 모니터링

경보를 사용하여 Lightsail 로드 밸런서 지표 모니터링

단일 로드 밸런서 지표를 감시하는 Amazon Lightsail 경보를 생성할 수 있습니다. 지정한 임계값에 상대적인 지표 값을 기준으로 경보 알림을 구성할 수 있습니다. 알림은 Lightsail 콘솔에 표시되는 배너, 이메일 주소로 전송되는 이메일, 휴대폰 번호로 전송되는 SMS 문자 메시지 형태로 사용할 수 있습니다. 경보에 대한 자세한 내용은 경보를 참조하세요.

목차

- 로드 밸런서 경보 제한
- 로드 밸런서 경보 구성 모범 사례
- 기본 경보 설정
- Lightsail 콘솔을 사용하여 로드 밸런서 지표 경보 생성
- Lightsail 콘솔을 사용하여 로드 밸런서 지표 경보 테스트
- 다음 단계

로드 밸런서 경보 제한

경보에는 다음과 같은 제한이 적용됩니다.

- 지표당 두 개의 경보를 구성할 수 있습니다.
- 경보는 5분 간격으로 평가되며 경보용 각 데이터 포인트는 5분간 집계된 지표 데이터를 나타냅니다.

- 경보 알림에 이메일 및/또는 SMS 문자 메시지를 사용하도록 구성한 경우에만 경보 상태가 OK로 변경될 때 알림을 받도록 경보를 구성할 수 있습니다.
- 경보 알림에 이메일 및/또는 SMS 문자 메시지를 사용하도록 구성한 경우에만 OK 경보 알림을 테스트할 수 있습니다.
- 이메일 및/또는 SMS 문자 메시지로 알리도록 경보를 구성하고 누락된 데이터 포인트에 대해 누락된 데이터 평가 안 함(Do not evaluate the missing data) 옵션을 선택한 경우에만 경보 상태가 INSUFFICIENT_DATA로 변경될 때 알리도록 경보를 구성할 수 있습니다.
- 경보가 OK 상태인 경우에만 알림을 테스트할 수 있습니다.

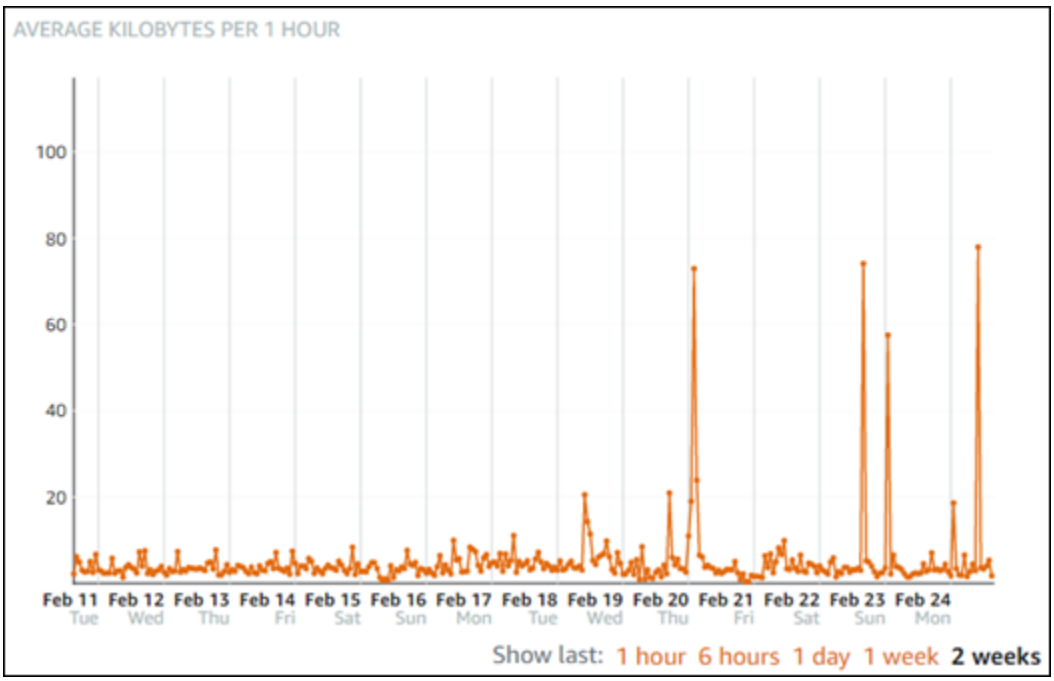
로드 밸런서 경보 구성 모범 사례

경보에는 다음과 같은 제한이 적용됩니다.

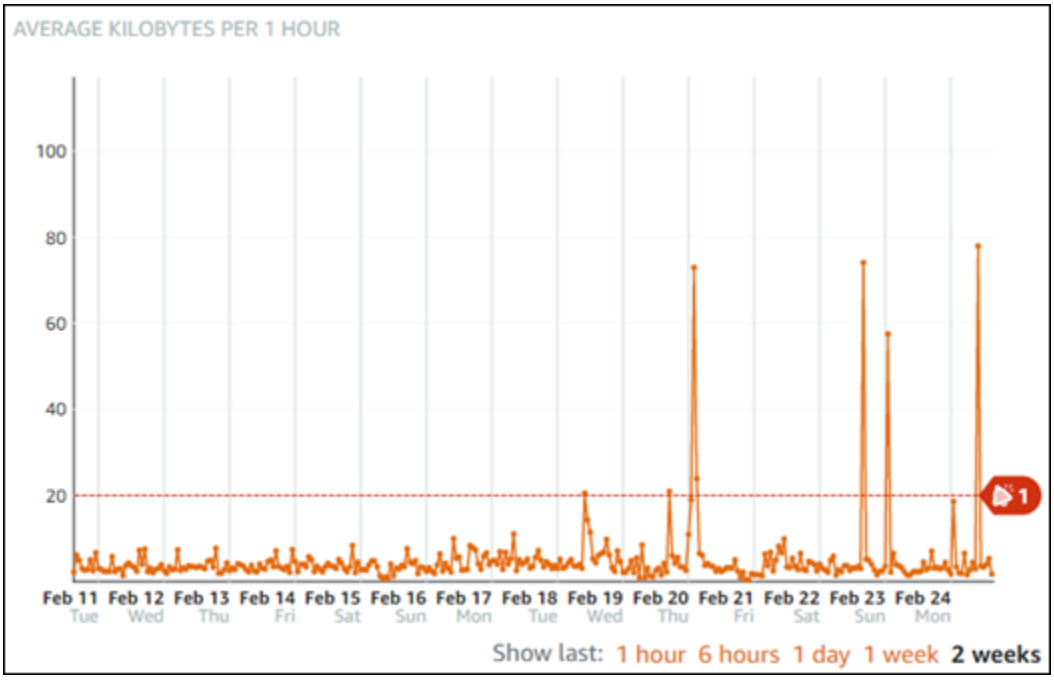
- 지표당 두 개의 경보를 구성할 수 있습니다.
- 경보는 5분 간격으로 평가되며 경보용 각 데이터 포인트는 5분간 집계된 지표 데이터를 나타냅니다.
- 경보 알림에 이메일 및/또는 SMS 문자 메시지를 사용하도록 구성한 경우에만 경보 상태가 OK로 변경될 때 알림을 받도록 경보를 구성할 수 있습니다.
- 경보 알림에 이메일 및/또는 SMS 문자 메시지를 사용하도록 구성한 경우에만 OK 경보 알림을 테스트할 수 있습니다.
- 이메일 및/또는 SMS 문자 메시지로 알리도록 경보를 구성하고 누락된 데이터 포인트에 대해 누락된 데이터 평가 안 함(Do not evaluate the missing data) 옵션을 선택한 경우에만 경보 상태가 INSUFFICIENT_DATA로 변경될 때 알리도록 경보를 구성할 수 있습니다.
- 경보가 OK 상태인 경우에만 알림을 테스트할 수 있습니다.

기본 경보 설정

지표 경보를 구성하기 전에 지표의 기록 데이터를 확인해야 합니다. 지난 2주 동안 지표의 하위 수준, 중간 수준 및 상위 수준을 확인합니다. 다음 인스턴스 송신 네트워크 트래픽(NetworkOut) 지표 그래프 예에서 하위 수준은 시간당 0~10KB이고, 중간 수준은 시간당 10~20KB이며, 상위 수준은 시간당 20~80KB입니다.



경보 임계값을 하위 수준 범위(예: 시간당 5KB)보다 크거나 같게 구성하면 경보 알림이 더 자주 발생하고 불필요한 경보 알림을 받을 수 있습니다. 경보 임계값을 상위 수준 범위(예: 시간당 20KB)보다 크거나 같게 구성하면 경보 알림이 덜 자주 발생하지만 보다 중요한 경보 알림을 받을 수 있습니다. 경보를 구성하고 활성화하면 다음 예와 같이 임계값을 나타내는 경보 선이 그래프에 나타납니다. 1로 레이블이 지정된 경보 선은 경보 1의 임계값을 나타내고, 2로 레이블이 지정된 경보 선은 경보 2의 임계값을 나타냅니다.



Lightsail 콘솔을 사용하여 로드 밸런서 지표 경보 생성

Lightsail 콘솔을 사용하여 로드 밸런서 지표 경보를 생성하려면 다음 단계를 수행하십시오.

1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 네트워킹을 선택합니다.
3. 경보를 생성하려는 로드 밸런서의 이름을 선택합니다.
4. 로드 밸런서 관리 페이지에서 지표 탭을 선택합니다.
5. Metrics Graphs(지표 그래프) 머리글 아래의 드롭다운 메뉴에서 경보를 생성하려는 지표를 선택합니다. 자세한 내용은 리소스 지표를 참조하세요.
6. 페이지의 경보(Alarms) 섹션에서 경보 추가(Add alarm)를 선택합니다.
7. 드롭다운 메뉴에서 비교 연산자 값을 선택합니다. greater than or equal to(크거나 같음), greater than(보다 큼), less than(보다 작음), less than or equal to(작거나 같음) 값이 있습니다.
8. 경보에 대한 임계값을 입력합니다.
9. 경보용 데이터 포인트를 입력합니다.
10. 평가 기간을 선택합니다. 기간은 5분에서 24시간까지 5분 단위로 지정할 수 있습니다.
11. 다음 알림 방법 중 하나를 선택합니다.

    - Email(이메일) - 경보 상태가 ALARM으로 변경되면 이메일로 알림을 받습니다.
    - SMS text message(SMS 문자 메시지) - 경보 상태가 ALARM으로 변경되면 SMS 문자 메시지로 알림을 받습니다. SMS 메시징은 Lightsail 리소스를 생성할 수 있는 AWS 리전 중 일부에서 지원되지 않으며, 일부 국가/지역에는 SMS 문자 메시지를 전송할 수 없습니다. 자세한 내용은 SMS 문자 메시지 지원을 참조하세요.

    > **Note**
    >
    > 이메일 또는 SMS로 알림을 받도록 선택했지만 해당 리소스의 AWS 리전에서 알림 연락처를 아직 구성하지 않은 경우 이메일 주소 또는 휴대폰 번호를 추가해야 합니다. 자세한 내용은 알림을 참조하세요.

12. (선택 사항) 경보 상태가 OK로 변경될 때 알림을 받으려면 Send me a notification when the alarm state change to OK(경보 상태가 OK로 변경될 때 알림 받기)를 선택합니다. 이 옵션은 이메일 또는 SMS 문자 메시지로 알림을 받도록 선택한 경우에만 사용할 수 있습니다.
13. (선택 사항) 고급 설정을 선택하고 다음 옵션 중 하나를 선택합니다.

- 경보가 누락 데이터를 처리하는 방식을 선택합니다. 다음 옵션을 사용할 수 있습니다.
  - Assume it's not within the threshold (Breaching threshold)(임계값을 벗어난 것으로 간주(임계값 위반)) - 누락 데이터 포인트가 “불량”이고 임계값을 위반하는 것으로 처리됩니다.
  - Assume it's within the threshold (Not breaching threshold)(임계값 내에 있는 것으로 간주(임계값을 위반하지 않음)) - 누락 데이터 포인트가 “양호”이고 임계값 내에 있는 것으로 처리됩니다.
  - 마지막으로 양호한 데이터 포인트 값 사용(현재 경보 상태를 무시하고 유지 - 현재 경보 상태가 유지됩니다.
  - Do not evaluate it (Treat missing data as missing)(평가 안 함(누락 데이터를 누락으로 처리)) - 상태 변경 여부를 평가할 때 경보에서 누락 데이터 포인트를 고려하지 않습니다.
- 경보 상태가 INSUFFICIENT_DATA로 변경될 때 알림을 받으려면 Send a notification if there is insufficient data(데이터가 불충분할 경우 알림 받기)를 선택합니다. 이 옵션은 이메일 또는 SMS 문자 메시지로 알림을 받도록 선택한 경우에만 사용할 수 있습니다.

14. 생성을 선택하여 경보를 추가합니다.

    나중에 경보를 편집하려면 편집할 경보 옆에 있는 줄임표 아이콘(⋮)을 선택하고 경보 편집을 선택합니다.

Lightsail 콘솔을 사용하여 로드 밸런서 지표 경보 테스트

Lightsail 콘솔을 사용하여 경보를 테스트하려면 다음 단계를 수행하십시오. 경보를 테스트하여 경보가 트리거될 때 이메일 또는 SMS 문자 메시지가 수신되는지를 확인하는 등 구성된 알림 옵션이 작동하는지 확인할 수 있습니다.

1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 네트워킹을 선택합니다.
3. 경보를 테스트하려는 로드 밸런서의 이름을 선택합니다.
4. 로드 밸런서 관리 페이지에서 지표 탭을 선택합니다.
5. Metrics Graphs(지표 그래프) 머리글 아래의 드롭다운 메뉴에서 경보를 테스트하려는 지표를 선택합니다.
6. 페이지의 경보 섹션까지 아래로 스크롤하고 테스트할 경보 옆에 있는 줄임표 아이콘(⋮)을 선택합니다.
7. 다음 옵션 중 하나를 선택하세요.

- 경보 알림 테스트 - 경보 상태가 ALARM으로 변경될 때의 알림을 테스트하려면 이 옵션을 선택합니다.
- OK 알림 테스트 - 경보 상태가 OK로 변경될 때의 알림을 테스트하려면 이 옵션을 선택합니다.

> **Note**
>
> 이러한 옵션을 사용할 수 없는 경우 경보에 대한 알림 옵션을 구성하지 않았거나 경보가 현재 ALARM 상태일 수 있습니다. 자세한 내용은 로드 밸런서 경보 제한을 참조하십시오.

선택한 테스트 옵션에 따라 경보가 일시적으로 ALARM 또는 OK 상태로 변경되며 경보에 구성한 알림 방법에 따라 이메일 및/또는 SMS 문자 메시지가 전송됩니다. 알림 배너는 ALARM 알림을 테스트하도록 선택한 경우에만 Lightsail 콘솔에 표시되며, OK 알림을 테스트하도록 선택한 경우 표시되지 않습니다. 몇 초 후에 경보가 실제 상태로 돌아옵니다.

로드 밸런서 경보 생성 후의 다음 단계

로드 밸런서 경보에 대해 수행할 수 있는 몇 가지 추가 작업이 있습니다.

- 알림 수신을 중지하려면 Lightsail에서 이메일과 휴대폰 번호를 삭제합니다. 자세한 내용은 알림 연락처 삭제를 참조하세요. 또한 경보를 비활성화하거나 삭제하여 특정 경보에 대한 알림 수신을 중지할 수도 있습니다. 자세한 내용은 지표 경보 삭제 또는 비활성화를 참조하세요.

Lightsail 모니터링을 위한 알림 연락처 설정

인스턴스, 데이터베이스, 로드 밸런서 또는 콘텐츠 전송 네트워크(CDN) 배포의 지표가 지정된 임계값을 초과하면 알림을 받도록 Amazon Lightsail을 구성할 수 있습니다. 알림은 Lightsail 콘솔에 표시되는 배너, 지정한 주소로 전송되는 이메일 또는 지정한 휴대폰 번호로 전송되는 SMS 문자 메시지 형태로 사용할 수 있습니다. 이메일 및 SMS 문자 메시지로 알림을 받으려면 리소스를 모니터링 AWS 리전 하려는 각에서 이메일 주소와 휴대폰 번호를 알림 연락처로 추가해야 합니다. 알림에 대한 자세한 내용은 알림을 참조하세요.

> **Important**
>
> SMS 문자 메시지 기능은 일시적으로 비활성화되었으며 Lightsail 리소스를 생성할 수 있는 AWS 리전 에서는 현재 지원되지 않습니다. 자세한 내용은 SMS 문자 메시지 지원을 참조하세요.

목차

- 리전별 알림 연락처 제한
- SMS 문자 메시지 지원
- 이메일 연락처 확인
- Lightsail 콘솔을 사용하여 알림 연락처 추가
- 를 사용하여 알림 연락처 추가 AWS CLI
- 알림 연락처 추가 후의 다음 단계

리전별 알림 연락처 제한

각각 하나의 이메일 주소와 하나의 휴대폰 번호만 추가할 수 있습니다 AWS 리전. 이메일 주소 또는 휴대폰 번호를 이미 추가한 리전에서 더 추가하려고 하면 기존 알림 연락처를 새 연락처로 바꿀 것인지 묻는 메시지가 표시됩니다.

에 여러 이메일 수신자가 필요한 경우 AWS 리전가 여러 수신자에게 전달하는 배포 목록을 구성하고 배포 목록의 이메일 주소를 알림 연락처로 추가할 수 있습니다.

SMS 문자 메시지 지원

> **Important**
>
> SMS 문자 메시지 기능은 일시적으로 비활성화되었으며 Lightsail 리소스를 생성할 수 있는 AWS 리전 에서는 현재 지원되지 않습니다. 또는 이메일 메시지를 구성하거나 Lightsail 콘솔에 표시된 알림 배너를 사용할 수 있습니다.
> SMS 문자 메시지 지원에 대한 다음 정보는 기능을 비활성화하기 전에 SMS 문자 메시지를 구성한 고객을 위해 게시됩니다.

Lightsail 리소스를 생성할 수 있는 일부 AWS 리전에서는 SMS 문자 메시징이 지원되지 않습니다. 일부 국가 및 지역에는 문자 메시지를 전송할 수 없습니다. SMS 메시징이 지원되지 않는 AWS 리전의 경우 이메일 알림 연락처만 구성할 수 있습니다.

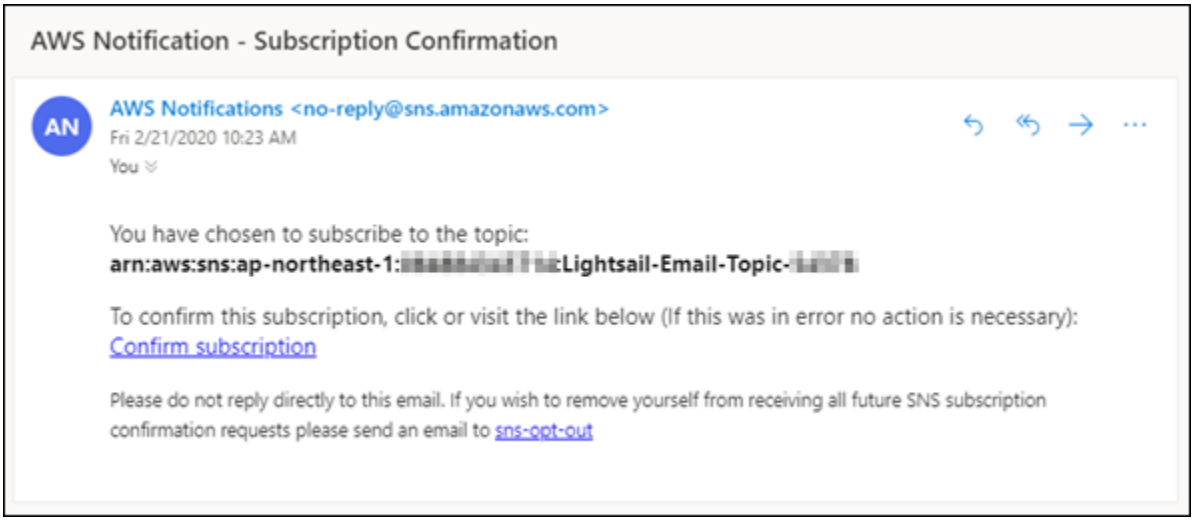
SMS 메시징은 다음 AWS 리전에서 지원됩니다. 이러한 리전에서 SMS 문자 메시징은 Lightsail에서 알림을 보내는 데 사용되는 Amazon Simple Notification Service(SNS)를 통해 지원됩니다.

- 미국 동부(버지니아 북부)(us-east-1)
- 미국 서부(오레곤)(us-west-2)
- 아시아 태평양(싱가포르)(ap-southeast-1)
- 아시아 태평양(시드니)(ap-southeast-2)
- 아시아 태평양(도쿄)(ap-northeast-1)
- 유럽(아일랜드)(eu-west-1)

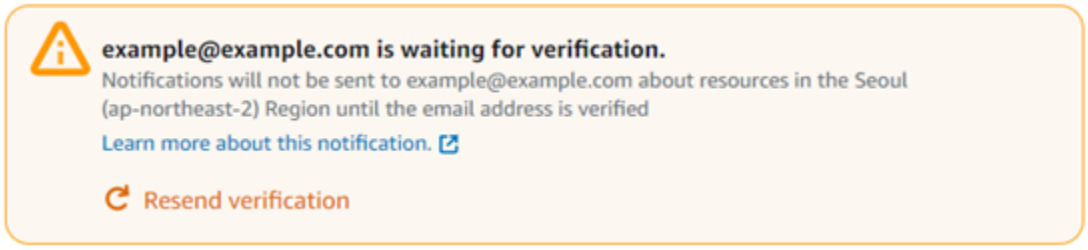
SMS 문자 메시지를 전송할 수 있는 전 세계 국가 및 리전 목록과 SMS 문자 메시징 AWS 리전이 지원되는 최신는 Amazon SNS 개발자 안내서의 지원되는 리전 및 국가를 참조하세요.

이메일 연락처 확인

Lightsail에서 이메일 주소를 알림 연락처로 추가하면 해당 주소로 확인 요청이 전송됩니다. 수신자는 확인 요청 이메일에 포함된 링크를 클릭하여 Lightsail 알림을 받을 것임을 확인해야 합니다. 이메일 주소에 대한 확인 프로세스를 완료해야만 알림이 전송됩니다. 이 확인은 AWS Notifications <no-reply@sns.amazonaws.com>에서 AWS Notification - Subscription Confirmation이라는 제목으로 전송됩니다. SMS 메시징은 확인이 필요하지 않습니다.



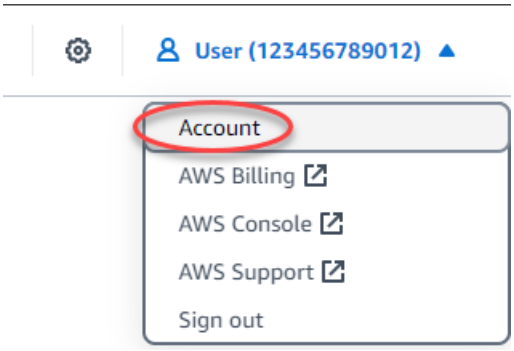
확인 요청이 받은 편지함 폴더에 없는 경우 사서함의 스팸 및 정크 폴더를 확인합니다. 확인 요청이 손실되었거나 삭제된 경우 Lightsail 콘솔과 계정(Account) 페이지에 표시되는 알림 배너에서 확인 재전송(Resend verification)을 선택합니다.



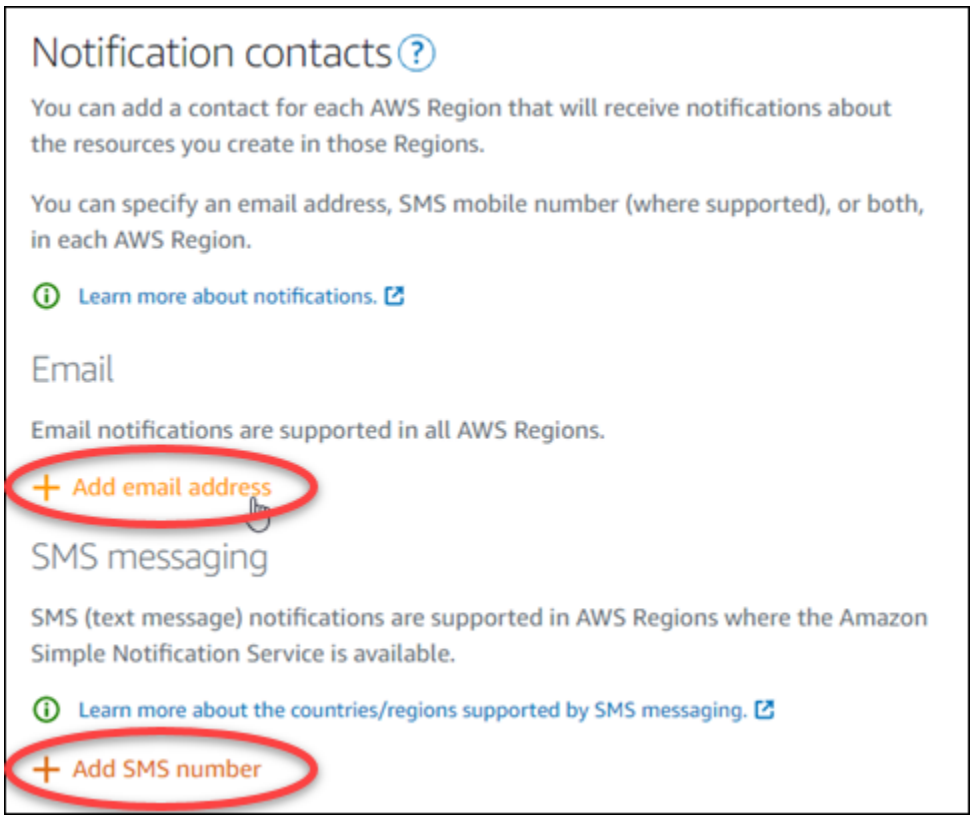
Lightsail 콘솔을 사용하여 알림 연락처 추가

Lightsail 콘솔을 사용하여 알림 연락처를 추가하려면 다음 단계를 수행하세요.

1. Lightsail 콘솔에 로그인합니다.
2. Lightsail 홈 페이지의 상단 탐색 메뉴에서 사용자 또는 역할을 선택합니다.
3. 드롭다운 메뉴에서 계정을 선택합니다.

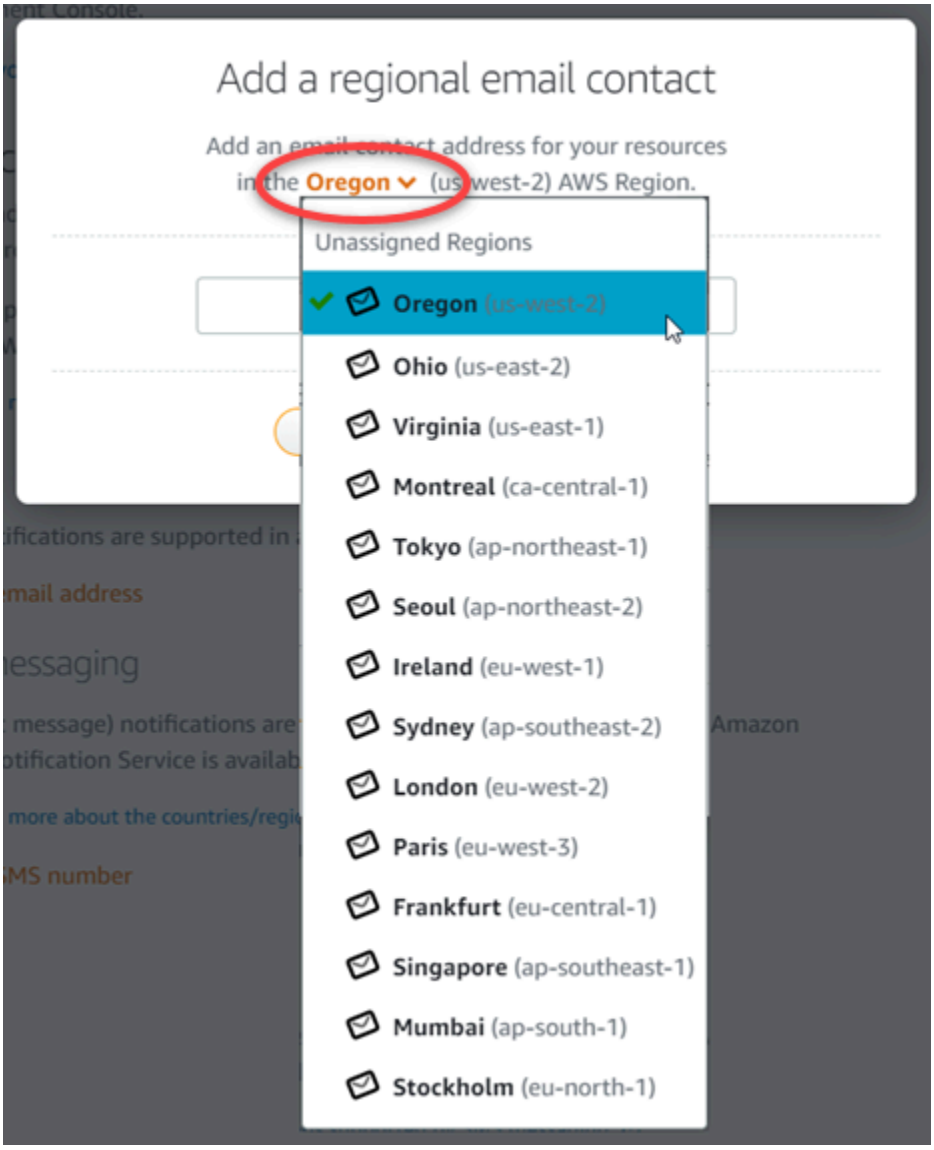


4. 프로필 및 연락처 탭의 알림 연락처 섹션에서 이메일 주소 추가 또는 SMS 번호 추가를 선택합니다.



5. 다음 단계 중 하나를 완료합니다.

   - 이메일 주소를 추가하는 경우 알림 연락처를 추가할 AWS 리전 을 선택합니다. 텍스트 상자에 이메일 주소를 입력합니다.

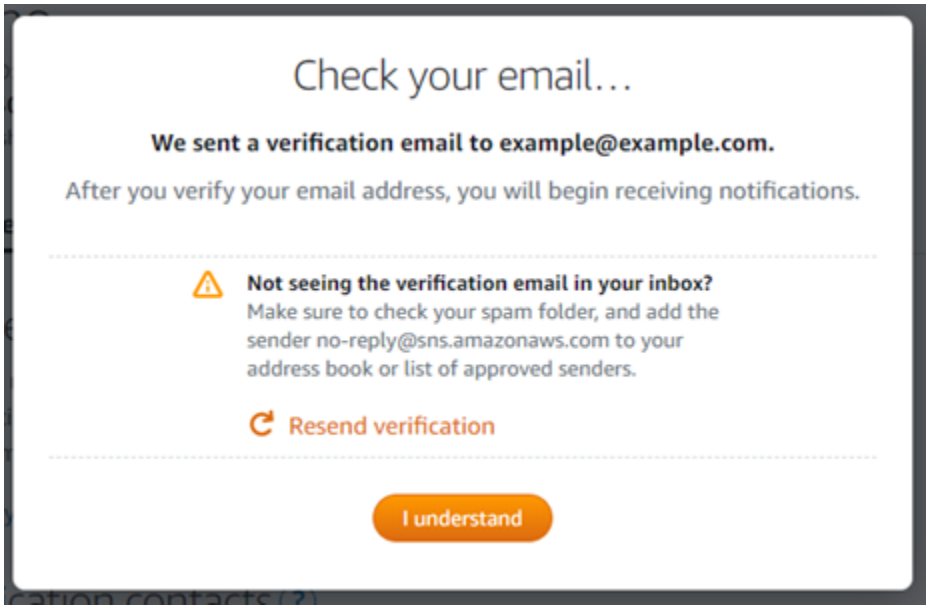


- SMS 번호를 추가하는 경우 알림 연락처를 추가할 AWS 리전 을 선택합니다. 휴대폰 번호의 국가를 선택하고 텍스트 상자에 휴대폰 번호를 입력합니다. 국가 코드는 자동으로 입력됩니다.

> **Important**
>
> SMS 문자 메시지 기능은 일시적으로 비활성화되었으며 Lightsail 리소스를 생성할 수 있는 AWS 리전 에서는 현재 지원되지 않습니다. 자세한 내용은 SMS 문자 메시지 지원을 참조하세요.

6. Add contact(연락처 추가)를 선택합니다.

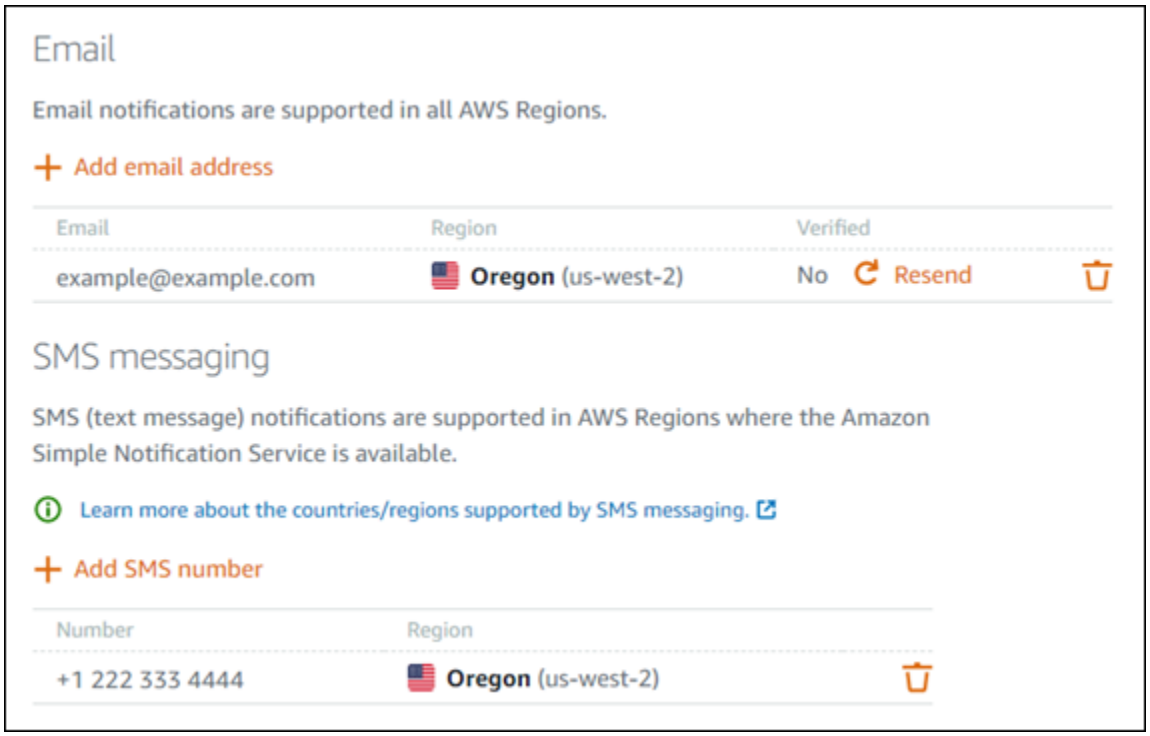
   이메일 주소를 알림 연락처로 추가하면 해당 주소로 확인 요청이 전송됩니다. 수신자는 확인 요청 이메일에 포함된 링크를 클릭하여 Lightsail 알림을 받을 것임을 확인해야 합니다. SMS 메시징은 확인이 필요하지 않습니다.



7. I understand(이해했습니다)를 선택합니다.

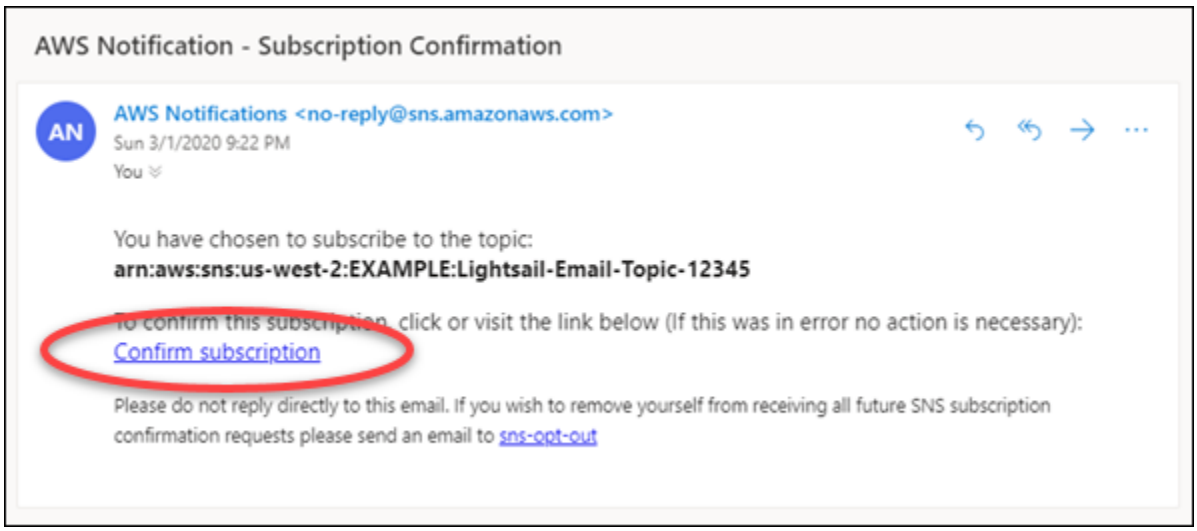
이메일 주소 또는 휴대폰 번호가 Notification contacts(알림 연락처) 섹션에 추가됩니다. 이어지는 단계에서 확인 프로세스를 완료하여 이메일 주소를 확인해야 합니다. 이메일 주소를 확인해야만 알림이 전송됩니다. 확인 요청이 분실되거나 삭제된 경우 리전별 이메일 주소 중 하나의 옆에 있는 Resend(재전송)를 선택하여 다시 확인 요청을 보냅니다.

> **Note**
>
> SMS 메시징은 확인이 필요하지 않습니다. 따라서 SMS 알림 연락처를 추가한 후에는 이 절차의 8~10단계를 수행할 필요가 없습니다.



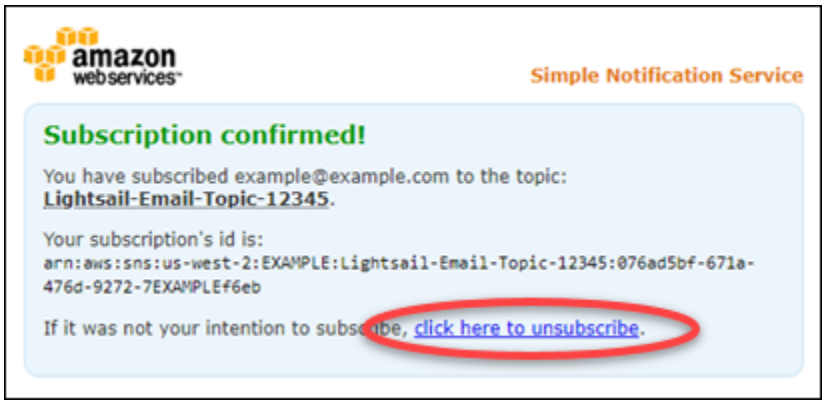
8. Lightsail에서 알림 연락처로 추가한 이메일 주소의 받은 편지함을 엽니다.
9. no-reply@sns.amazonaws.com에서 보낸 AWS 알림 - 구독 확인 이메일을 엽니다.

> **Note**
>
> 확인 요청이 받은 편지함 폴더에 없는 경우 사서함의 스팸 및 정크 폴더를 확인합니다.



10. 이메일에서 Confirm subscription(구독 확인)을 선택하여 Lightsail 알림을 받을 것임을 확인합니다.

    구독을 확인하는 다음 페이지의 브라우저 창이 열립니다. 구독을 취소하려면 페이지에서 여기를 클릭하여 구독을 취소합니다. 또는 페이지를 닫은 경우 알림 연락처를 삭제하는 단계를 완료합니다.



를 사용하여 알림 연락처 추가 AWS CLI

Lightsail(AWS Command Line Interface)를 사용하여AWS CLI에 대한 알림 연락처를 추가하려면 다음 단계를 수행하세요.

1. 터미널 또는 명령 프롬프트 창을 엽니다.

   아직 설치하지 않았다면 를 AWS CLI 설치하고에서 작동하도록 구성합니다Lightsail.

2. 다음 명령을 입력하여 알림 연락처를 추가합니다.

```
aws lightsail create-contact-method --region Region --notificationProtocol Protocol
 --contact-endpoint Destination
```

명령에서 다음과 같이 바꿉니다.

- 알림 연락처를 추가해야 AWS 리전 하는이 있는 *##*입니다.
- *Protocol*을 연락처에 대한 알림 프로토콜(Email 또는 SMS)로 대체
- *Destination*을 해당 이메일 주소 또는 휴대폰 번호로 대체

> **Note**
>
> 휴대폰 번호를 지정할 때는 E.164 형식을 사용합니다. E.164는 국제 통신에 사용되는 전화번호 구조의 표준입니다. 이 형식을 따르는 전화번호는 최대 15자리 숫자를 사용할 수 있으며 더하기 문자(+) 및 국가 코드가 접두사로 추가됩니다. 예를 들어, E.164 형식의 미국 전화번호는 +1XXX5550100으로 지정될 수 있습니다. 자세한 내용은 Wikipedia의 E.164를 참조하세요.

예:

```
aws lightsail create-contact-method --region us-west-2 --notificationProtocol Email
 --contact-endpoint example@example.com
```

```
aws lightsail create-contact-method --region us-east-1 --notificationProtocol SMS
 --contact-endpoint +14445556666
```

Enter 키를 누르면 요청에 대한 세부 정보가 포함된 작업 응답이 표시됩니다.

알림 연락처로 지정한 이메일 주소로 확인 요청이 전송됩니다. 이 절차는 수신자가 Lightsail 알림을 구독하려고 하는지 확인합니다. 이어지는 단계에서 확인 프로세스를 완료하여 이메일 주소를 확인해야 합니다. 이메일 주소에 대한 확인 프로세스를 완료해야만 해당 이메일 주소로 알림이 전송됩니다. 처음 받은 알림이 분실된 경우 리전별 이메일 주소 중 하나의 옆에 있는 Resend(재전송)를 선택하여 다시 확인 요청을 보냅니다.

> **Note**
>
> SMS 메시징은 확인이 필요하지 않습니다. 따라서 SMS 알림 연락처를 추가하는 경우 이 절차의 8~10단계를 수행할 필요가 없습니다.

3. 알림 연락처로 추가한 이메일 주소의 받은 편지함을 엽니다.
4. no-reply@sns.amazonaws.com에서 보낸 AWS 알림 - 구독 확인 이메일을 엽니다.
5. 이메일에서 Confirm subscription(구독 확인)을 선택하여 Lightsail에서 알림을 받을 것임을 확인합니다.

   구독을 확인하는 다음 페이지의 브라우저 창이 열립니다. 구독을 취소하려면 페이지에서 여기를 클릭하여 구독을 취소합니다. 또는 페이지를 닫은 경우 알림 연락처를 삭제하는 단계를 완료합니다.

알림 연락처 추가 후의 다음 단계

알림 연락처에 대해 수행할 수 있는 몇 가지 추가 작업이 있습니다.

- 알림 연락처를 추가한 AWS 리전 에 경보를 추가합니다. 경보가 시작될 때 이메일 및 SMS 문자 메시지로 알림을 받도록 선택할 수 있습니다. 자세한 내용은 경보 섹션을 참조하세요.
- 알림을 받아야 할 때 알림이 수신되지 않는 경우 몇 가지 사항을 점검하여 알림 연락처가 올바르게 구성되었는지 확인해야 합니다. 자세한 내용은 알림 문제 해결을 참조하세요.
- 알림 수신을 중지하려면 Lightsail에서 이메일과 휴대폰 번호를 삭제합니다. 자세한 내용은 지표 경보 삭제 또는 비활성화를 참조하세요. 또한 경보를 비활성화하거나 삭제하여 특정 경보에 대한 알림 수신을 중지할 수도 있습니다. 자세한 내용은 지표 경보 삭제 또는 비활성화를 참조하세요.

Lightsail의 알림 연락처 삭제

Amazon Lightsail에서 이메일 및 휴대폰 번호 알림 연락처를 삭제하여 Lightsail 리소스에 대한 이메일 및 SMS 문자 메시지 알림 수신을 중지합니다. 알림에 대한 자세한 내용은 알림을 참조하세요.

또한 경보를 비활성화하거나 삭제하여 특정 경보에 대한 알림 수신을 중지할 수도 있습니다. 자세한 내용은 지표 경보 삭제 또는 비활성화를 참조하세요.

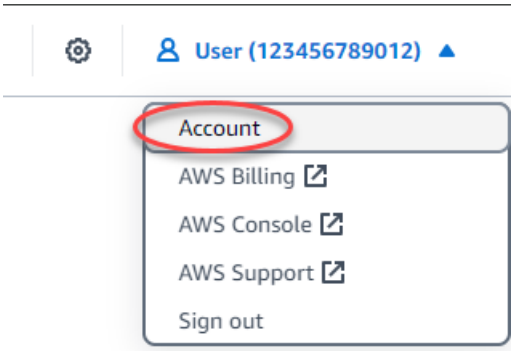
목차

- Lightsail 콘솔을 사용하여 알림 연락처 삭제
- 를 사용하여 알림 연락처 삭제 AWS CLI
- 알림 연락처 삭제 후의 다음 단계

Lightsail 콘솔을 사용하여 알림 연락처 삭제

Lightsail 콘솔을 사용하여 알림 연락처를 삭제하려면 다음 단계를 수행하세요.

1. Lightsail 콘솔에 로그인합니다.
2. Lightsail 홈 페이지의 상단 탐색 메뉴에서 사용자 또는 역할을 선택합니다.
3. 드롭다운 메뉴에서 계정을 선택합니다.



4. 프로필 및 연락처(Profile & contacts) 탭의 알림 연락처(Notification contacts) 섹션에서 삭제할 이메일 주소 또는 휴대폰 번호 옆에 있는 삭제 아이콘을 선택합니다.
5. 예를 선택하여 알림 연락처 삭제를 확인합니다.

를 사용하여 알림 연락처 삭제 AWS CLI

Lightsail(AWS Command Line Interface)를 사용하여AWS CLI에 대한 알림 연락처를 삭제하려면 다음 단계를 수행하세요.

1. 터미널 또는 명령 프롬프트 창을 엽니다.

   아직 설치하지 않은 경우 를 AWS CLI 설치하고에서 작동하도록 구성합니다Lightsail.
2. 다음 명령을 입력하여 알림 연락처를 삭제합니다.

```
aws lightsail delete-contact-method --region Region --notificationProtocol Protocol
```

   명령에서 다음과 같이 바꿉니다.

- 알림 연락처를 삭제해야 AWS 리전 하는 *##*입니다.
- *Protocol*을 삭제할 연락처에 대한 알림 프로토콜(예: Email 또는 SMS)로 대체

예제:

```
aws lightsail delete-contact-method --region us-west-2 --notificationProtocol SMS
```

Enter 키를 누르면 요청에 대한 세부 정보가 포함된 작업 응답이 표시됩니다.

알림 연락처 삭제 후의 다음 단계

알림 연락처를 삭제한 후 수행할 수 있는 몇 가지 추가 작업이 있습니다.

- 알림 연락처를 삭제하면 이메일 및 SMS 문자 메시지 알림이 중지되지만 알림 배너는 Lightsail 콘솔에 계속 표시됩니다. 알림 배너를 중지하고 이메일 및 SMS 문자 메시지 알림도 중지하려면 알림을 유발하는 경보를 비활성화하거나 삭제합니다. 자세한 내용은 지표 경보 삭제 또는 비활성화를 참조하세요.
- 이메일 및 SMS 문자 메시지 알림 수신을 재개하려면 Lightsail에서 이메일 주소 및 휴대폰 번호를 알림 연락처로 추가합니다. 자세한 내용은 알림 연락처 추가를 참조하세요.

확인 보류 중인 Lightsail 경보 알림 및 연락처 검토

Lightsail 콘솔의 경보 알림 페이지에서 모든 Amazon Lightsail 리소스에 대한 활성 경보 및 알림을 검토할 수 있습니다. 이 페이지는 `In alarm` 상태인 경보, 즉 활성화되었으며 정의된 임계값을 현재 위반하는 경보를 통합합니다. 확인 보류 중인 이메일 연락처를 검토할 수도 있습니다. 경보에 대한 자세한 내용은 Lightsail에서 지표 알람 수신 단원을 참조하세요. 경보 알림에 대한 자세한 내용은 Lightsail 리소스에 대한 지표 알림 구성 섹션을 참조하세요.

주제

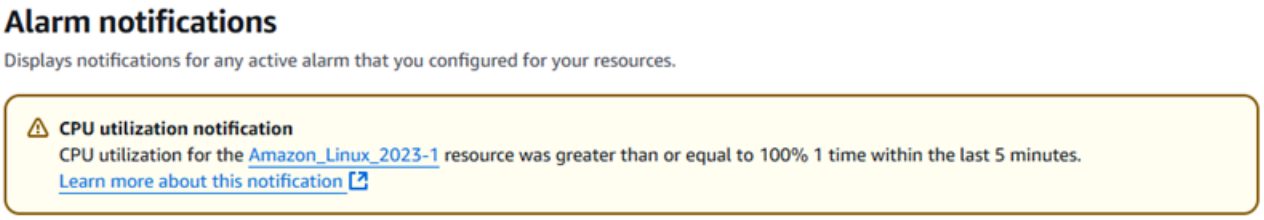
- 활성 경보에 대한 경보 알림 검토
- 확인 보류 중인 이메일 연락처 검토

활성 경보에 대한 경보 알림 검토

Lightsail 콘솔에서 모든 리소스에 대한 Lightsail 경보 알림을 검토할 수 있습니다. 각 항목에는 경보가 활성화되는 이유와 경보와 어느 리소스가 관련되었는지에 대한 추가 세부 정보가 있습니다. 경보를 추가하는 방법에 대한 자세한 내용은 경보 구성 섹션을 참조하세요.

활성 경보에 대한 경보 알림을 검토하려면

1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 경보 알림을 선택합니다.
3. 경보 알림에서 활성 경보를 검토할 수 있습니다.

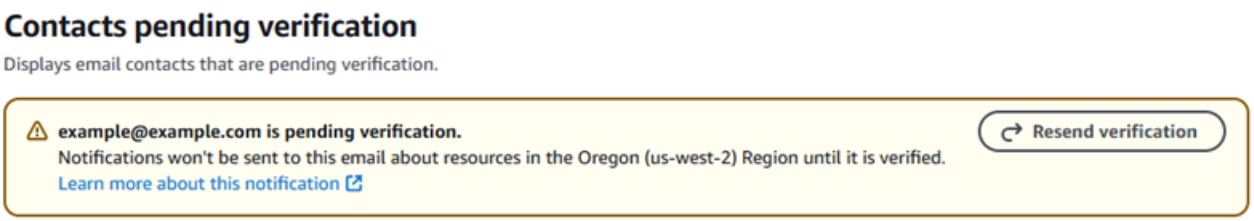


확인 보류 중인 이메일 연락처 검토

Lightsail 콘솔에서 확인 보류 중인 이메일 연락처를 검토할 수 있습니다. 각 항목에는 이메일 주소, AWS 리전 알림, 확인 재전송 기능이 포함됩니다. 이메일 연락처를 추가하는 방법에 대한 자세한 내용은 Lightsail 모니터링을 위한 알림 연락처 설정 섹션을 참조하세요.

확인 보류 중인 이메일 연락처를 검토하려면

1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 경보 알림을 선택합니다.
3. 확인 보류 중인 연락처 아래에서 확인 보류 중인 이메일 연락처를 검토할 수 있습니다.

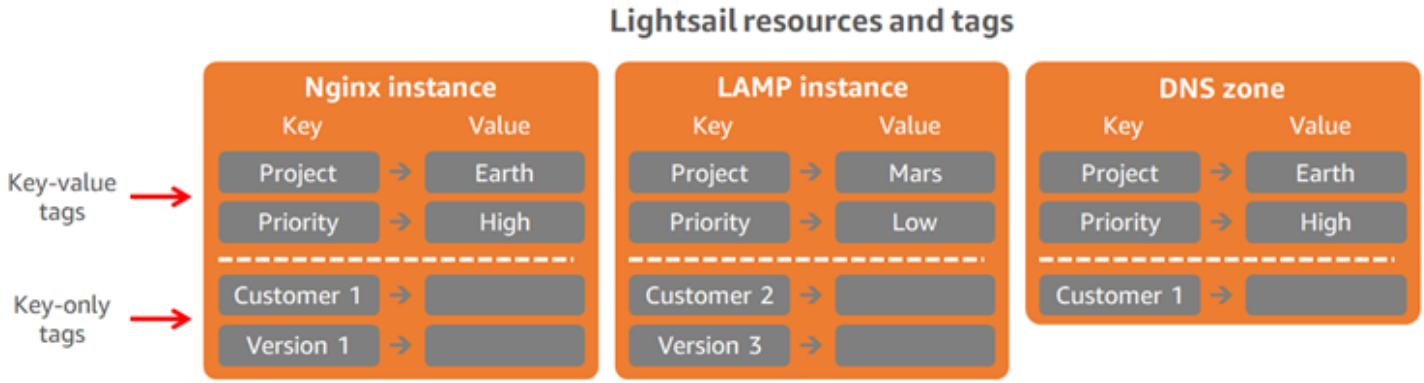


태그를 사용하여 Lightsail 리소스를 필터링하고 구성합니다.

Amazon Lightsail을 사용하면 리소스에 레이블을 태그로 할당할 수 있습니다. 각 태그는 키와 선택적 값으로 구성된 레이블로, 이를 사용하여 리소스를 보다 효율적으로 관리, 검색 및 필터링할 수 있습니다.

Amazon Lightsail을 사용하면 리소스에 레이블을 태그로 할당할 수 있습니다. 각 태그는 키와 선택적 값으로 구성된 레이블로, 이를 사용하여 리소스를 효율적으로 관리, 검색 및 필터링할 수 있습니다. 고유한 태그 유형은 없지만 목적, 소유자, 환경 또는 기타 기준에 따라 Lightsail 리소스를 분류할 수 있습니다. 이는 동일한 유형의 리소스가 많을 때 유용합니다. 지정한 태그를 기반으로 특정 리소스를 신속하게 식별할 수 있습니다. 예를 들어, 각 리소스의 프로젝트 또는 우선 순위를 추적하는 데 도움이 되는 리소스에 대한 태그 세트를 정의하세요.

값이 없는 키는 Lightsail에서 키 전용 태그라고 합니다. 값이 있는 키는 키-값 태그라고 합니다. 다음 다이어그램은 태그 지정 방식을 설명합니다. 이 예에서 각 리소스에는 키-값 및 키 전용 태그 세트가 있습니다. 키-값 태그는 프로젝트와 우선 순위를 식별하며 키 전용 태그는 고객과 애플리케이션 버전을 식별합니다.



태그를 사용하여 결제 구성 및 액세스 제어

또한 태그를 사용하여 결제를 구성하고 Lightsail의 리소스 및 요청에 대한 액세스를 제어하고 태그 키에 대한 액세스를 제어할 수 있습니다. 자세한 내용은 다음 가이드 중 하나를 참조하세요.

- 태그를 사용하여 리소스 비용 구성
- 태그를 사용하여 리소스 액세스 제어

태그 지정을 지원하는 Lightsail 리소스

대부분의 Lightsail 리소스는 생성할 때나 생성한 후에 태그를 지정할 수 있습니다. 리소스 생성 도중 태그를 적용할 수 없는 경우 Lightsail은 리소스 생성 프로세스를 롤백합니다. 이렇게 하면 태그를 사용하여 리소스가 생성되거나 전혀 생성되지 않도록 하고 태그를 지정해야 하는 리소스가 태그 지정되지 않은 상태로 남지 않도록 할 수 있습니다.

다음과 같은 Lightsail 리소스를 Lightsail 콘솔에서 태그 지정할 수 있습니다.

- 인스턴스
- 컨테이너 서비스
- 콘텐츠 전송 네트워크(CDN) 배포
- 버킷
- 데이터베이스 수
- 디스크
- DNS 영역
- 로드 밸런서

> **Important**
>
> Lightsail 콘솔을 사용하여 생성된 스냅샷은 소스 리소스의 태그를 자동으로 상속합니다. 스냅샷에서 생성된 Lightsail 리소스에는 스냅샷을 생성할 때 소스 리소스에 있던 것과 동일한 태그가 있습니다.

Lightsail API, AWS Command Line Interface (AWS CLI) 또는 SDK를 사용하여 다음과 같은 리소스에 태그를 지정할 수 있습니다.

- 데이터베이스 스냅샷
- 데이터베이스 수
- 디스크 스냅샷
- 디스크
- 도메인(DNS 영역)
- 인스턴스 스냅샷
- 인스턴스

- 키 페어
- 로드 밸런서 TLS 인증서(Lightsail을 사용하여 만든 TLS 인증서)
- 로드 밸런서

> **Important**
>
> Lightsail API AWS CLI또는 SDKs를 사용하여 생성된 스냅샷은 소스 리소스에서 태그를 자동으로 상속하지 않습니다. 대신 `tags` 파라미터를 사용하여 소스 리소스의 태그를 수동으로 지정해야 합니다.

태그 제한

태그에 적용되는 기본 제한 사항은 다음과 같습니다.

- 리소스당 최대 태그 수는 50개입니다.
- 각 리소스에 대해 각 태그 키는 고유해야 합니다. 각 태그 키는 하나의 값만 가질 수 있습니다.
- 최대 키 길이는 유니코드 문자(UTF-8) 128자입니다.
- 최대 값 길이는 유니코드 문자(UTF-8) 256자입니다.
- 태그 지정 스키마를 여러 서비스와 리소스에서 사용하는 경우 다른 서비스에서 허용되는 문자에 제한이 있을 수 있음에 유의하세요. 일반적으로 허용되는 문자는 문자, 숫자, 공백 및 특수 문자 + - = . _ : / @입니다.
- 태그 키와 값은 대/소문자를 구분합니다.
- 키 또는 값에 `aws:` 접두사는 사용하지 않습니다. 이 접두사는 AWS용으로 예약되어 있습니다.

태그를 사용하여 Lightsail 리소스 분류

Amazon Lightsail의 태그를 사용하여 용도, 소유자, 환경 또는 기타 기준으로 리소스를 분류합니다. 태그를 생성할 때 또는 생성한 후 리소스에 추가할 수 있습니다. 태그 생성 후 리소스에 추가하려면 다음 단계를 따르세요.

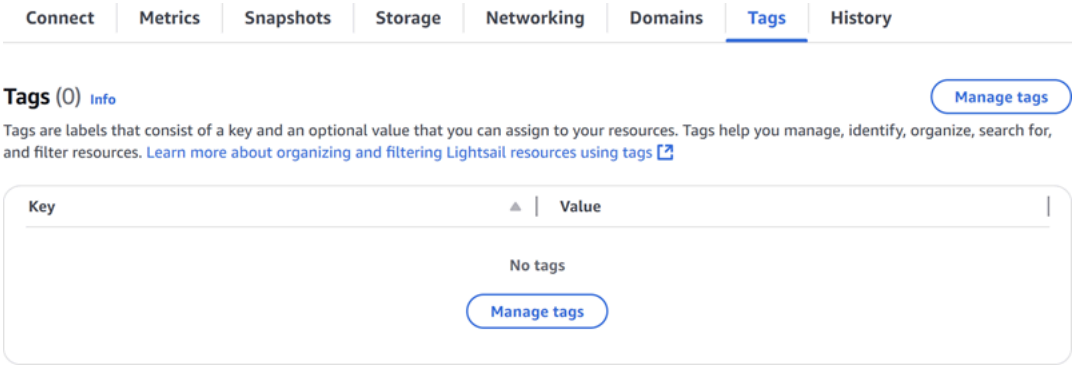
> **Note**
>
> 태그, 태그를 지정할 수 있는 리소스 및 제한 사항에 대한 자세한 내용은 태그를 참조하세요.

태그를 리소스에 추가

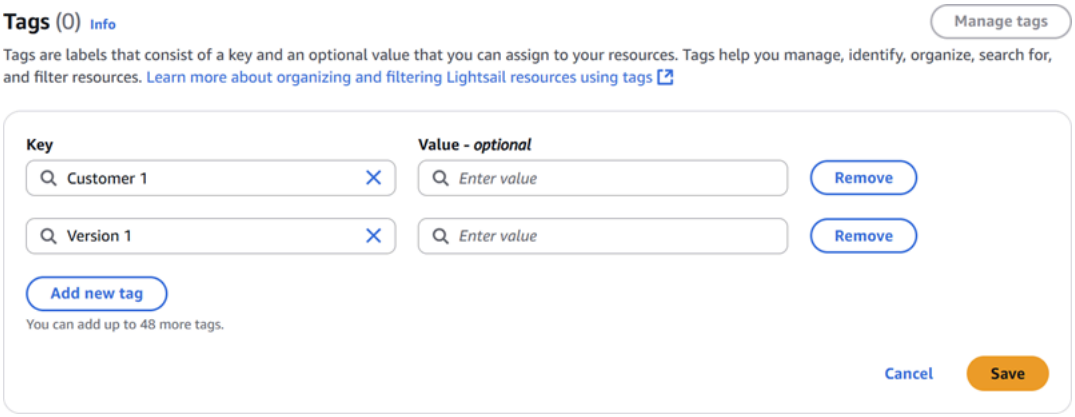
1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 태그를 지정하려는 리소스 유형의 탭을 선택합니다. 예를 들어, DNS 영역에 태그를 추가하려면 네트워킹 탭을 선택합니다. 또는 인스턴스에 태그를 추가하려면 인스턴스 탭을 선택합니다.

> **Note**
>
> Lightsail 콘솔을 사용하여 인스턴스, 컨테이너 서비스, CDN 배포, 버킷, 데이터베이스, 디스크, DNS 영역 및 로드 밸런서에 태그를 지정할 수 있습니다. 그러나 Lightsail API 작업 또는 AWS Command Line Interface(AWS CLI)나 SDK를 사용하여 더 많은 Lightsail 리소스에 태그를 지정할 수 있습니다. 태그 지정을 지원하는 Lightsail 리소스의 전체 목록은 태그를 참조하세요.

3. 태그를 지정할 리소스를 선택합니다.
4. 선택한 리소스에 대한 관리 페이지에서 태그 탭을 선택합니다.



5. 태그 관리를 선택합니다.
6. 생성하고자 하는 태그 유형에 따라 다음 옵션 중 하나를 사용합니다. 이 탭에서 이미 추가된 태그를 편집할 수도 있습니다.
   - 키 전용 태그 생성
     a. 새로운 태그 추가를 선택합니다.
     b. 생성하고자 하는 각 태그의 키 값을 입력합니다.
     c. 태그 입력이 완료되면 저장을 선택하고, 태그를 추가하지 않으려면 취소를 선택합니다.

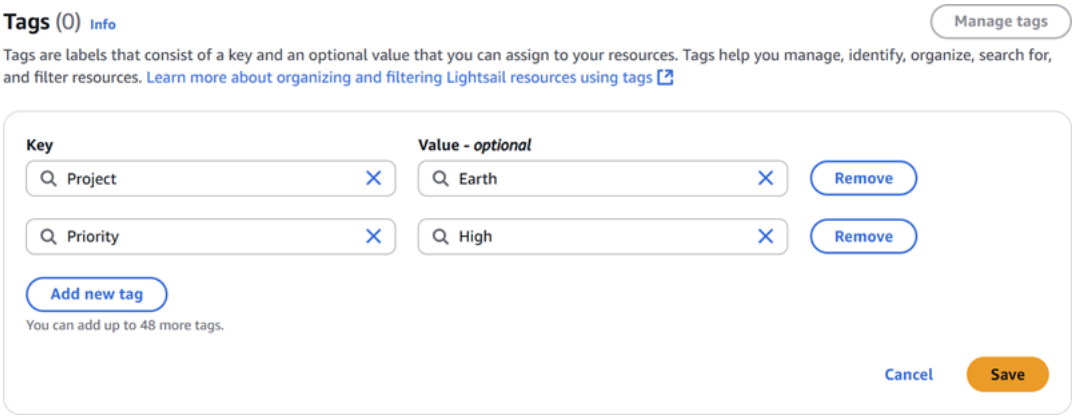


- 키-값 태그 생성

  a. 새로운 태그 추가를 선택합니다.

  b. 생성하고자 하는 각 태그의 키 및 값을 입력합니다.

  c. 태그 입력이 완료되면 저장을 선택하고, 태그를 추가하지 않으려면 취소를 선택합니다.



다음 단계

리소스에 태그를 추가한 다음 수행할 수 있는 작업에 대한 자세한 내용은 다음 안내서를 참조하세요.

- 태그를 사용하여 리소스 구성
- 태그를 사용하여 리소스에 대한 비용 구성
- 태그를 사용하여 리소스에 대한 액세스 제어
- 태그 삭제

Lightsail 리소스에서 태그 제거

Amazon Lightsail 리소스에서 태그를 삭제할 수 있습니다. 한 리소스에서 태그를 삭제해도 다른 모든 리소스에서는 동일한 태그가 삭제되지 않습니다. 모든 리소스에서 태그를 완전히 삭제하려면 각 리소스에서 해당 태그를 제거해야 합니다. 이 안내서에는 리소스에서 태그를 삭제하는 단계가 나와 있습니다.

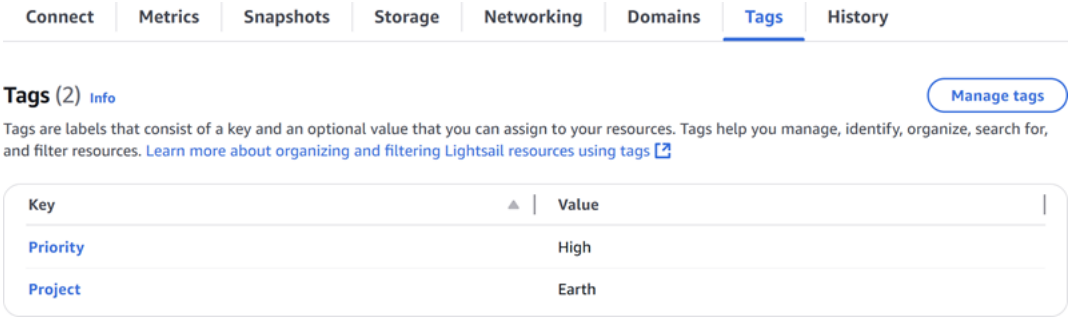
> **Note**
>
> 태그, 태그를 지정할 수 있는 리소스 및 태그 제한 사항에 대한 자세한 내용은 태그를 참조하세요.

리소스에서 태그를 삭제하려면

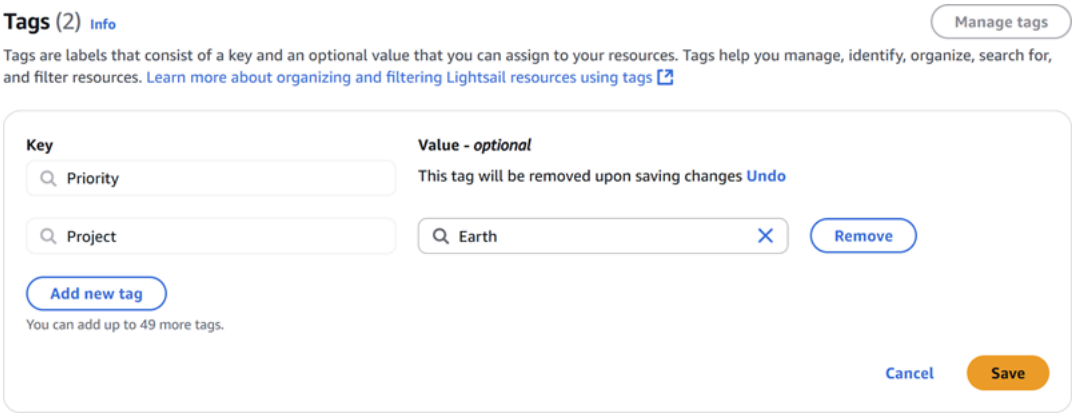
1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 삭제하려는 태그의 리소스 유형을 선택합니다. 예를 들어, DNS 영역에서 태그를 삭제하려면 네트워킹을 선택합니다. 또는 인스턴스에서 태그를 삭제하려면 인스턴스를 선택합니다.

   > **Note**
   >
   > Lightsail 콘솔을 사용하여 인스턴스, 컨테이너 서비스, CDN 배포, 버킷, 데이터베이스, 디스크, DNS 영역 및 로드 밸런서에 태그를 지정할 수 있습니다. 그러나 Lightsail API 작업, AWS Command Line Interface(AWS CLI) 또는 SDK를 사용하여 더 많은 Lightsail 리소스에 태그를 지정할 수 있습니다. 태그 지정을 지원하는 Lightsail 리소스의 전체 목록은 태그를 참조하세요.

3. 태그를 삭제하려는 리소스를 선택합니다.
4. 선택한 리소스에 대한 관리 페이지에서 태그 탭을 선택합니다.



5. 태그 관리를 선택합니다.
6. 리소스에서 삭제하려는 태그에 대해 제거를 선택합니다.
7. 저장을 선택하여 리소스에서 선택된 태그를 삭제하거나 취소를 선택하여 태그를 유지합니다.



Lightsail 리소스 수준 권한 및 태그 기반 권한 부여를 사용하여 리소스에 대한 액세스 제어

Lightsail에서는 일부 API 작업에 대한 리소스 수준 권한 및 태그 기반 권한 부여를 지원합니다. 자세한 내용은 서비스 권한 부여 참조에서 Amazon Lightsail에 사용되는 작업, 리소스 및 조건 키를 참조하세요.

태그를 기반으로 Lightsail 리소스에 대한 액세스 제어

Amazon Lightsail에서 태그를 사용하여 리소스에 대한 액세스, 요청에 대한 액세스, 태그 키에 대한 액세스를 제어할 수 있습니다. 이 설명서에서는 Lightsail 리소스를 생성 또는 삭제하는 데 필요한 키-값 태그를 지정하는 AWS Identity and Access Management (IAM) 정책을 생성하고, 이 정책을 요청해야 하는 사용자 또는 그룹에 연결하는 방법에 대해 알아봅니다.

> **Note**
>
> Lightsail의 태그, 태그를 지정할 수 있는 리소스 및 제한 사항에 대한 자세한 내용은 태그를 참조하세요.

1단계: IAM 정책 생성

먼저 IAM 콘솔에서 다음 IAM 정책을 생성합니다. IAM 정책 생성에 대한 자세한 내용은 IAM 설명서의 IAM 정책 생성를 참조하세요.

다음 정책은 생성 요청을 통해 `allow`의 키 태그와 `true`의 값이 정의되지 않는 한 사용자가 새 Lightsail 리소스를 생성하지 않도록 제한합니다. 또한 이 정책은 `allow/true` 키-값 태그가 없는 한 사용자가 리소스를 삭제하지 않도록 제한합니다.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "lightsail:Create*",
                "lightsail:TagResource",
                "lightsail:UntagResource"
            ],
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "aws:RequestTag/allow": "true"
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "lightsail:Delete*",
                "lightsail:TagResource",
                "lightsail:UntagResource"
            ],
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "aws:ResourceTag/allow": "true"
                }
            }
```

```
        }
    ]
}
```

다음 정책은 사용자가 `allow/false`가 아닌 키-값 태그가 있는 리소스에 대한 태그를 변경하지 않도록 제한합니다.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Deny",
            "Action": [
                "lightsail:TagResource"
            ],
            "Resource": "*",
            "Condition": {
                "StringNotEquals": {
                    "aws:ResourceTag/allow": "false"
                }
            }
        }
    ]
}
```

2단계: 사용자 또는 그룹에 정책 연결

IAM 정책을 생성한 후 해당 정책을 키-값 페어를 사용하여 Lightsail 리소스를 생성해야 하는 사용자 또는 그룹에 연결합니다. 사용자 또는 그룹에 IAM 정책을 연결하는 방법에 대한 자세한 내용은 IAM 설명서의 IAM 정책 추가 및 제거를 참조하십시오.

태그를 사용하여 Lightsail 리소스 비용 구성

에서 태그를 사용하여 자체 비용 구조를 반영하도록 AWS 결제Amazon Lightsail를 구성할 수 있습니다. 이렇게 하려면 Lightsail 리소스에 키-값 태그를 추가합니다. 그런 다음 AWS 결제 및 비용 관리 콘솔에서 해당 태그를 활성화합니다. 마지막으로 등록하여 비용 할당 보고서에 포함된 태그 키 값으로 AWS 계정 청구서를 가져옵니다. 이 안내서에는 이를 설정하기 위한 단계가 나와 있습니다.

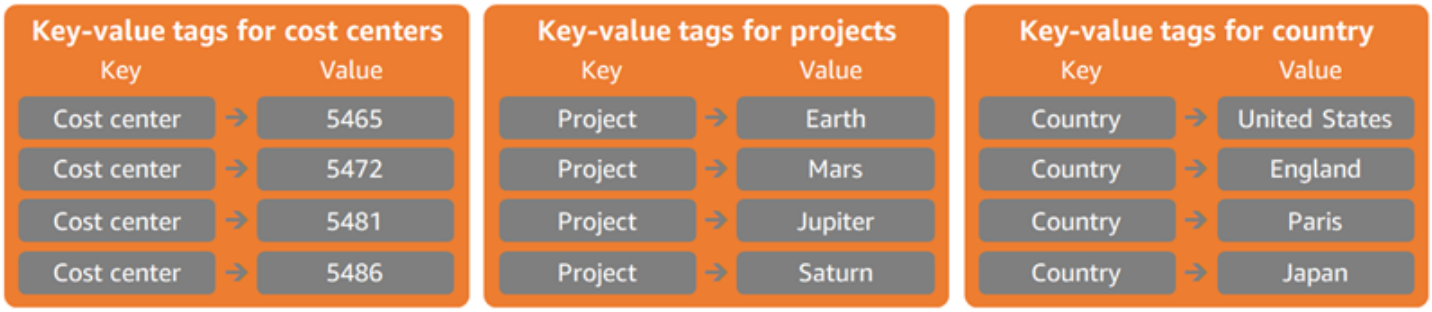
> **Note**
>
> Lightsail의 태그, 태그를 지정할 수 있는 리소스 및 태그 제한 사항에 대한 자세한 내용은 태그를 참조하세요.

> **Important**
>
> Lightsail 데이터베이스 스냅샷은 현재 비용 할당 태그가 추가된 후에도 비용 할당 보고서에서 추적할 수 없습니다.

1단계: 리소스에 키-값 태그 추가

결제 콘솔에 구성하려는 Lightsail 리소스에 키-값 태그를 추가합니다. 키-값 태그에 대한 자세한 내용은 리소스에 태그 추가를 참조하세요.

비용을 정리하는 방법을 설명하는 태그 키 세트를 만드는 것이 좋습니다. 비용 할당 보고서에는 태그 키가 해당하는 값이 각 행에 들어간 추가 열로 표시되어 있습니다. 따라서 태그 키의 일관된 세트를 사용할 경우 비용을 더 효율적으로 추적할 수 있습니다. 예를 들어, 특정 비용 센터가 있는 여러 Lightsail 리소스에 태그를 지정할 수 있습니다. "비용 센터" 키와 숫자 값 페어링으로 이렇게 할 수 있습니다. 그런 다음 여러 리소스에 대한 해당 비용 센터의 결제를 확인할 수 있도록 결제 정보를 구성합니다. 다음은 비용 할당을 구성하는 데 사용할 수 있는 키-값 태그를 보여주는 예입니다.



2단계: 사용자 정의 비용 할당 태그 활성화

필요한 태그를 Lightsail 리소스에 추가한 후 과금 정보 및 비용 관리 콘솔에서 비용 할당에 대해 해당 태그를 활성화합니다. 예를 들어, '비용 센터' 키 태그를 생성한 경우 과금 정보 및 비용 관리 콘솔에서 해당 키 태그를 활성화하여 해당 태그에 대한 비용 할당 보고서를 생성합니다. 자세한 내용은 AWS 결제 및 비용 관리 설명서의 사용자 정의 비용 할당 태그 활성화를 참조하세요.

3단계: 비용 할당 보고서 설정 및 확인

월별 비용 할당 보고서에는 계정의 AWS 사용량이 제품 범주 및 연결된 계정 사용자별로 나열됩니다. 이 보고서에는 세부 결제 보고서와 동일한 행 항목 및 태그 키에 대한 추가 열이 포함되어 있습니다. 월별 비용 할당 보고서를 설정하려면 AWS 결제 및 비용 관리 설명서의 월별 비용 할당 보고서 설정을 참조하세요.

비용 할당 보고서를 설정하면 보고서가 저장되는 Amazon Simple Storage Service(S3) 버킷이 정의됩니다. 정의한 Amazon S3 버킷을 열고 비용 할당 보고서가 사용 가능하게 되면 해당 보고서를 엽니다. 비용 할당 보고서의 내용에 대한 자세한 내용은 AWS 결제 및 비용 관리 설명서의 비용 할당 보고서 보기를 참조하세요.

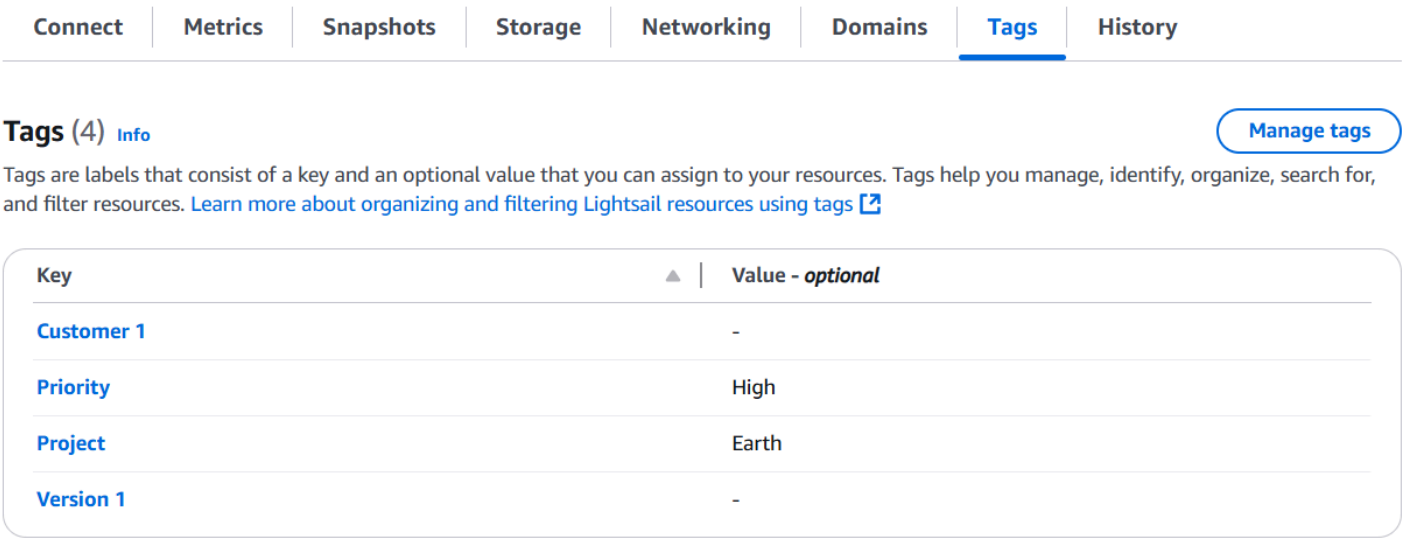
구성 및 필터링을 위한 Lightsail 리소스 태그 지정

Amazon Lightsail 리소스에 태그를 지정한 후 추가한 태그별로 리소스를 필터링할 수 있습니다. Lightsail 콘솔에서 태그를 선택하거나 검색하면 됩니다. 이 안내서에서는 태그별로 Lightsail 리소스를 보고 필터링하는 방법을 보여줍니다.

> **Note**
>
> 태그, 태그를 지정할 수 있는 리소스 및 태그 제한 사항에 대한 자세한 내용은 태그를 참조하세요.

리소스에 대한 태그 보기

인스턴스, 컨테이너 서비스, CDN 배포, 버킷, 데이터베이스, 디스크, DNS 영역 및 로드 밸런서는 Lightsail 콘솔을 사용하여 태그를 지정할 수 있으므로, 태그(Tags) 탭이 포함됩니다. 이 탭은 인스턴스 리소스에 대한 다음 예에 표시된 대로 리소스의 관리 페이지를 통해 액세스할 수 있습니다. Tags(태그) 탭에서 태그를 추가, 편집 또는 삭제할 수 있습니다. 자세한 내용은 리소스에 태그 추가 및 태그 삭제를 참조하세요.



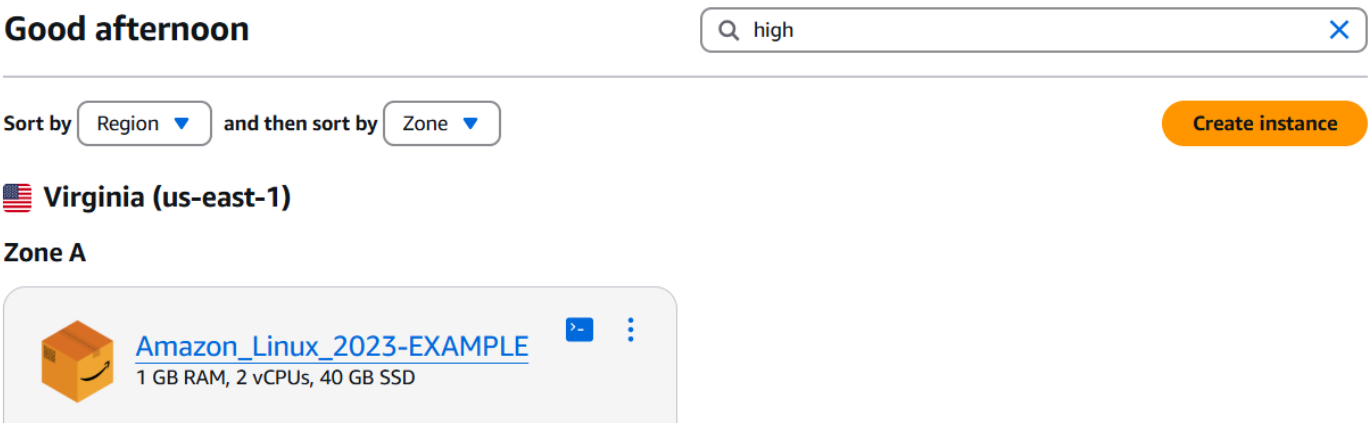
> **Note**
>
> Lightsail 콘솔을 사용하여 인스턴스, 컨테이너 서비스, CDN 배포, 버킷, 데이터베이스, 디스크, DNS 영역 및 로드 밸런서에 태그를 지정할 수 있습니다. 그러나 Lightsail API 작업 또는 AWS Command Line Interface(AWS CLI)나 SDK를 사용하여 더 많은 Lightsail 리소스에 태그를 지정할 수 있습니다. 태그 지정을 지원하는 Lightsail 리소스의 전체 목록은 태그를 참조하세요.

태그를 사용하여 리소스 필터링

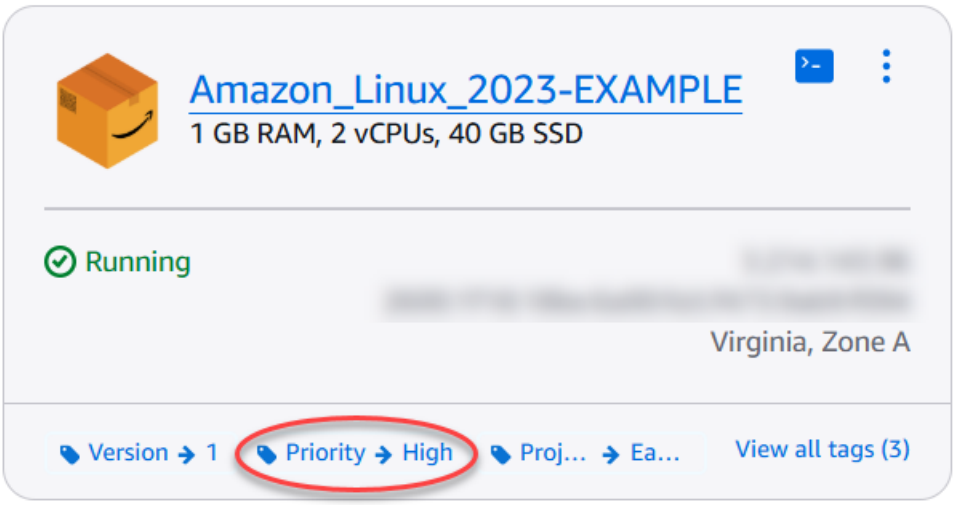
다음은 태그를 사용하여 리소스를 필터링하기 위해 Lightsail 콘솔에서 사용할 수 있는 옵션입니다. 이러한 모든 옵션은 검색하거나 선택한 태그만 표시되도록 Lightsail 홈 페이지를 새로 고칩니다.

> **Note**
>
> 이러한 필터링 옵션은 영구적입니다. 태그별로 필터링하고 Lightsail 홈 페이지의 섹션 간에 이동하면 필터가 계속 적용됩니다.

- Lightsail 홈 페이지의 검색 텍스트 상자에 키 전용 태그 또는 필터링 기준이 되는 값을 입력하고 Enter를 누릅니다.



- Lightsail 홈 페이지의 리소스에 표시되는 태그를 선택합니다.



일반적인 Lightsail 리소스 문제 해결

이 섹션에서는 다음 Amazon Lightsail 리소스에 대한 문제 해결 주제를 다룹니다. 단계별 지침 및 지침에 따라 Lightsail 인스턴스, 데이터베이스, 네트워킹, 로드 밸런서 및 기타 리소스를 사용하는 동안 발생할 수 있는 일반적인 문제를 진단하고 해결합니다.

문제 해결 주제에서는 WordPress 구성 실패, IAM 권한 문제, 디스크 오류, 연결 문제, 서비스 비가용성, IPv6 연결, 인스턴스 용량 제한, 로드 밸런서 오류, 알림 전송 실패 및 SSL/TLS 인증서 문제를 포함하는 다양한 시나리오를 다룹니다. 이 가이드를 따라하면 Lightsail 리소스와 관련된 다양한 문제를 효과적으로 해결하여 애플리케이션 및 워크로드의 원활한 운영과 최적의 성능을 보장할 수 있습니다.

주제

- Lightsail 인스턴스의 WordPress 설정 문제 해결
- Lightsail 콘솔에서 403(권한 없음) 오류 해결
- Lightsail 디스크 연결 및 사용 문제 해결
- Lightsail 브라우저 기반 SSH 및 RDP 클라이언트와의 연결 오류 해결
- Lightsail에서 Ghost 인스턴스의 503 서비스 사용 불가 오류 문제 해결
- Lightsail에서 Identity and Access Management(IAM) 문제 해결
- Lightsail 인스턴스의 IPv6 도달 가능성 확인
- Lightsail에서 인스턴스 용량 부족 오류 해결
- Lightsail 로드 밸런서 문제 해결
- Lightsail에서 알림 전송 문제 해결
- Lightsail의 SSL/TLS 인증서 문제 해결

Lightsail 인스턴스의 WordPress 설정 문제 해결

Amazon Lightsail의 WordPress 설정 워크플로 중에 두 가지 유형의 오류 메시지가 나타날 수 있습니다.

일반적인 오류

> 이러한 유형의 오류는 워크플로의 마지막 단계에서 인증서 생성을 선택한 직후 발생합니다. 이러한 오류는 Lightsail 콘솔 상단의 배너에 표시됩니다. 이는 일반적으로 이전 WordPress 인스턴스에서 설정 워크플로를 실행하거나 잘못된 정보를 제출하면 발생합니다. 예를 들어 인스턴스의 퍼블릭 IP 주소를 가리키지 않는 DNS 레코드를 선택합니다.

설정 실패

이러한 유형의 오류는 워크플로의 마지막 단계를 완료한 후 몇 분 이내에 발생합니다. 이러한 실패 메시지는 인스턴스 연결 탭의 WordPress 웹 사이트 설정 섹션에 표시됩니다. 이러한 오류는 Let's Encrypt HTTPS 인증서를 인스턴스에 구성할 수 없을 때 발생합니다.

다음 주제의 정보를 사용하여 WordPress 설정 안내 워크플로에서 발생할 수 있는 오류를 진단하고 수정할 수 있습니다.

주제

- Lightsail에서 WordPress 설정 오류 해결
- Lightsail에서 WordPress 설정 실패 문제 해결

Amazon Lightsail의 WordPress 설정 안내 워크플로에 대한 자세한 내용은 WordPress 인스턴스 구성을 참조하세요.

Lightsail에서 WordPress 설정 오류 해결

워크플로 중에 제출된 정보에 문제가 있는 경우 Lightsail 콘솔 상단에 오류 메시지가 표시됩니다.

메시지의 첫 번째 줄은 설정에 오류가 발생했음을 알려줍니다.

InstanceRegion 리전의 인스턴스 *InstanceName*에 대한 설정을 완료할 수 없습니다.

두 번째 줄에는 설정에서 발생한 오류가 포함되어 있습니다.

오류가 발생하여 인스턴스에 연결하거나 연결을 유지할 수 없습니다.

문제 해결을 시작하려면 다음 오류 중에서 메시지에 표시된 오류와 일치하는 내용을 확인합니다.

오류

- DNS 레코드를 찾을 수 없습니다. 도메인의 DNS 레코드가 인스턴스의 퍼블릭 IP 주소를 가리키는지 확인하고 DNS 변경 사항이 전파될 시간을 허용합니다.
- DNS 레코드가 일치하지 않습니다. 도메인의 DNS 레코드가 인스턴스의 퍼블릭 IP 주소를 가리키는지 확인하고 DNS 변경 사항이 전파될 시간을 허용합니다.

- 인스턴스에 연결할 수 없습니다. SSH 연결이 준비될 때까지 몇 분 정도 기다립니다. 그런 다음 설정을 다시 시작합니다.
- 지원되지 않는 WordPress 버전입니다. 설정은 WordPress 버전 6 이상만 지원합니다.
- 설정은 2023년 1월 1일 이후에 생성된 WordPress 인스턴스만 지원합니다.
- 인스턴스 방화벽 포트 22, 80 및 443은 설정 워크플로 중에 모든 IP 주소에서 TCP 연결을 허용해야 합니다. 인스턴스 네트워킹 탭에서 해당 설정을 변경할 수 있습니다.

DNS 레코드를 찾을 수 없습니다. 도메인의 DNS 레코드가 인스턴스의 퍼블릭 IP 주소를 가리키는지 확인하고 DNS 변경 사항이 전파될 시간을 허용합니다.

이유

이 오류는 잘못 구성된 DNS 레코드 또는 인터넷의 DNS 전체에 전파할 시간이 충분하지 않은 DNS 레코드로 인해 발생합니다.

수정

A 또는 AAAA DNS 레코드가 DNS 영역에 있고 인스턴스의 퍼블릭 IP 주소를 가리키는지 확인합니다. 자세한 내용은 Lightsail의 DNS를 참조하세요.

Apex 도메인(`example.com`)과 `www` 하위 도메인(`www.example.com`)의 트래픽을 가리키는 DNS 레코드를 추가하거나 업데이트할 때 인터넷의 DNS 전체에 전파해야 합니다. nslookup 또는 MxToolbox의 DNS 조회와 같은 도구를 사용하여 DNS 변경 사항이 적용되었는지 확인할 수 있습니다.

> **Note**
>
> 인터넷의 DNS를 통해 아무 DNS 레코드 변경 사항이 전파될 때까지 기다립니다. 몇 시간 정도 걸릴 수 있습니다.

DNS 레코드가 일치하지 않습니다. 도메인의 DNS 레코드가 인스턴스의 퍼블릭 IP 주소를 가리키는지 확인하고 DNS 변경 사항이 전파될 시간을 허용합니다.

이유

A 또는 AAAA DNS 레코드는 인스턴스의 퍼블릭 IP 주소를 가리키지 않습니다.

수정

A 또는 AAAA DNS 레코드가 DNS 영역에 있고 인스턴스의 퍼블릭 IP 주소를 가리키는지 확인합니다. 자세한 내용은 Lightsail의 DNS를 참조하세요.

> **Note**
>
> 인터넷의 DNS를 통해 아무 DNS 레코드 변경 사항이 전파될 때까지 기다립니다. 몇 시간 정도 걸릴 수 있습니다.

인스턴스에 연결할 수 없습니다. SSH 연결이 준비될 때까지 몇 분 정도 기다립니다. 그런 다음 설정을 다시 시작합니다.

이유

인스턴스가 방금 생성되거나 재부팅되었으며 SSH 연결이 준비되지 않았습니다.

수정

SSH 연결이 준비될 때까지 몇 분 정도 기다립니다. 그런 다음 안내 워크플로를 다시 시도합니다. 자세한 내용은 Lightsail에서의 SSH 문제 해결을 참조하세요.

지원되지 않는 WordPress 버전입니다. 설정은 WordPress 버전 6 이상만 지원합니다.

이유

인스턴스에 설치된 WordPress 버전이 WordPress 버전 6보다 이전 버전입니다. 이전 WordPress 버전에는 HTTPS 인증서 생성을 방지하는 호환되지 않는 소프트웨어 및 종속 항목이 포함되어 있습니다.

수정

Lightsail 콘솔에서 새 WordPress 인스턴스를 생성합니다. 그런 다음 WordPress 웹 사이트를 이전 인스턴스에서 새 인스턴스로 마이그레이션합니다. 자세한 내용은 기존 WordPress 블로그 마이그레이션을 참조하세요.

기존 인스턴스를 대체하기 위해 새 인스턴스를 생성하는 경우 애플리케이션 종속성을 새 인스턴스로 업데이트해야 합니다.

설정은 2023년 1월 1일 이후에 생성된 WordPress 인스턴스만 지원합니다.

이유

설정과 함께 사용 중인 인스턴스에는 오래된 소프트웨어가 포함되어 있을 수 있습니다. 이전 소프트웨어에서는 HTTPS 인증서가 생성되지 않습니다.

수정

Lightsail 콘솔에서 새 WordPress 인스턴스를 생성합니다. 그런 다음 WordPress 웹 사이트를 이전 인스턴스에서 새 인스턴스로 마이그레이션합니다. 자세한 내용은 기존 WordPress 블로그 마이그레이션을 참조하세요.

기존 인스턴스를 대체하기 위해 새 인스턴스를 생성하는 경우 애플리케이션 종속성을 새 인스턴스로 업데이트해야 합니다.

인스턴스 방화벽 포트 22, 80 및 443은 설정 워크플로 중에 모든 IP 주소에서 TCP 연결을 허용해야 합니다. 인스턴스 네트워킹 탭에서 해당 설정을 변경할 수 있습니다.

이유

인스턴스 방화벽 포트 22, 80 및 443은 설정이 실행되는 동안 모든 IP 주소에서 TCP 연결을 허용해야 합니다. 이 오류는 이러한 포트 중 하나 이상이 닫힐 때 생성됩니다. 자세한 내용은 인스턴스 방화벽을 참조하세요.

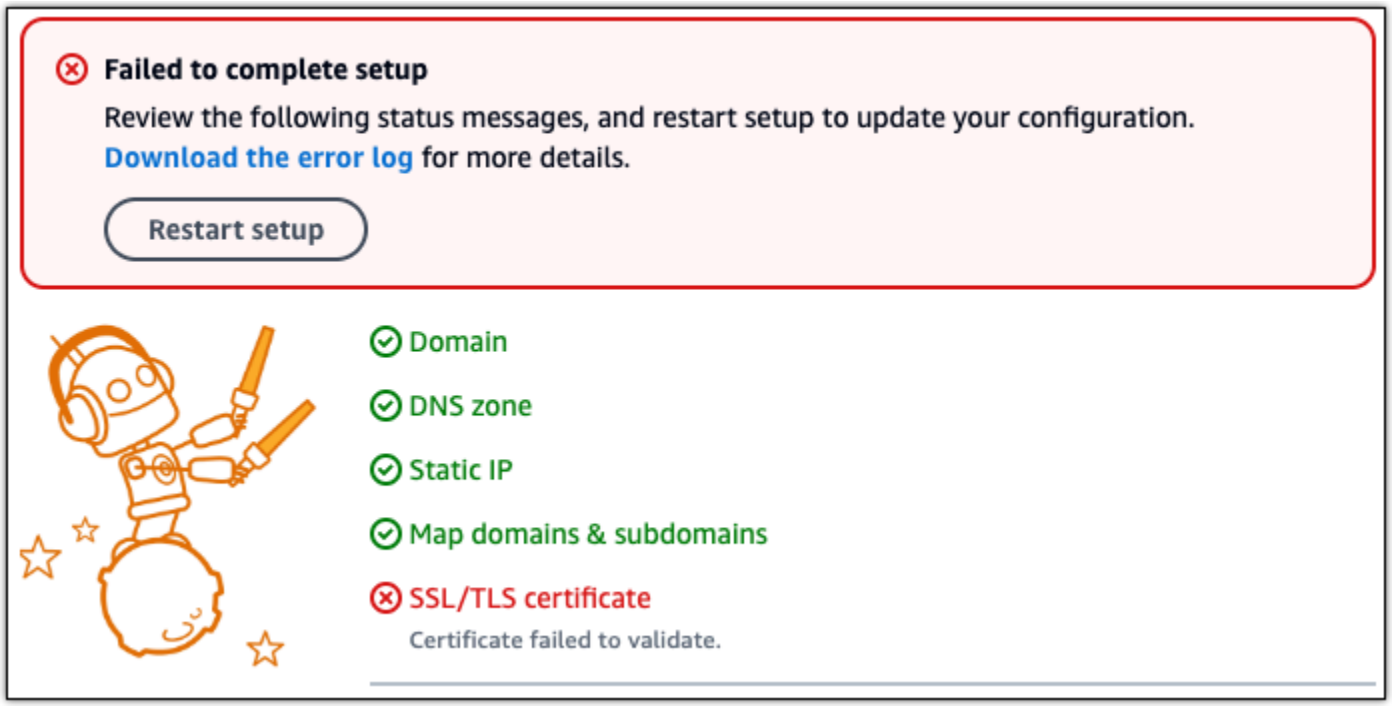
수정

포트 22, 80 및 443을 통한 TCP 연결을 허용하도록 인스턴스의 IPv4 및 IPv6 방화벽 규칙을 추가하거나 편집합니다. 자세한 내용은 방화벽 규칙 추가 및 편집을 참조하세요.

Lightsail에서 WordPress 설정 실패 문제 해결

다음 정보는 인스턴스 연결 탭의 WordPress 웹 사이트 설정 섹션에 나타날 수 있는 장애 메시지 문제를 해결하는 데 도움이 될 수 있습니다. 설치 오류는 워크플로의 마지막 단계를 완료한 후 몇 분 이내에 발생할 수 있습니다. 이는 Let's Encrypt HTTPS 인증서를 인스턴스에 구성할 수 없을 때 발생합니다.

설정 완료 실패 - 다음 상태 메시지를 검토하고 설정을 다시 시작하여 구성을 업데이트합니다. 자세한 내용은 오류 로그를 다운로드하세요.



실패 메시지에서 오류 로그 다운로드 링크를 선택하여 설정이 생성한 오류 로그를 다운로드하고 확인합니다. 문제 해결을 시작하려면 로그의 오류 메시지를 다음 오류 중 하나와 일치시킵니다.

Bitnami

Certbot.errors.AuthorizationError: 일부 문제가 실패했습니다.

이유

이 오류는 잘못 구성된 DNS 레코드 또는 인터넷 전체에 전파할 시간이 충분하지 않은 DNS 레코드로 인해 발생합니다.

수정

A 또는 AAAA DNS 레코드가 DNS 영역에 있고 인스턴스의 퍼블릭 IP 주소를 가리키는지 확인합니다. 자세한 내용은 Lightsail의 DNS를 참조하세요.

Apex 도메인(`example.com`)과 `www` 하위 도메인(`www.example.com`)의 트래픽을 가리키는 DNS 레코드를 추가하거나 업데이트할 때 인터넷 전체에 전파해야 합니다. nslookup 또는 MxToolbox의 DNS 조회와 같은 도구를 사용하여 DNS 변경 사항이 적용되었는지 확인할 수 있습니다.

> **Note**
>
> 인터넷의 DNS를 통해 아무 DNS 레코드 변경 사항이 전파될 때까지 기다립니다. 몇 시간 정도 걸릴 수 있습니다.

Certbot이 일부 도메인을 인증하지 못했습니다.

이유

HTTPS 인증서가 인스턴스에 구성되어 있는 동안 다른 프로세스가 포트 80을 사용하는 경우 이 오류가 발생할 수 있습니다.

수정

WordPress 인스턴스를 다시 시작합니다. 그런 다음 안내 워크플로를 다시 실행합니다. 다시 시작해도 문제가 해결되지 않는 경우 다음 절차에 따라 포트 80에서 실행 중인 인스턴스에서 실행 중인 프로세스를 종료합니다.

절차

1. Lightsail 브라우저 기반 SSH 클라이언트 를 사용하거나 AWS CloudShell을 사용하여 인스턴스에 연결합니다.
2. 인스턴스에서 실행 중인 Bitnami 프로세스를 중지합니다.

```
$ sudo /opt/bitnami/ctlscript.sh stop
```

Bitnami 프로세스가 중지되었는지 확인합니다.

```
$ sudo /opt/bitnami/ctlscript.sh status
```

3. 포트 80을 사용하는 다른 프로세스가 있는지 확인합니다.

```
$ fuser -n tcp 80
```

4. 다른 애플리케이션에서 필요하지 않은 모든 프로세스를 종료합니다.

```
$ fuser -k -n tcp 80
```

5. WordPress 설정을 다시 시작합니다.

리포지토리 http://cdn-aws.deb.debian.org/debian 버스터-백포트에는 더 이상 릴리스 파일이 없습니다.

이유

인스턴스에 업데이트할 수 없는 더 이상 사용되지 않는 Debian 리포지토리가 있습니다.

수정

Debian 리포지토리 파일에 나열된 리포지토리 URL을 편집하려면 다음 절차를 사용합니다.

절차

1. Lightsail 브라우저 기반 SSH 클라이언트 를 사용하거나 AWS CloudShell을 사용하여 인스턴스에 연결합니다.
2. `/etc/apt/sources.list.d/` 디렉터리로 이동합니다.

```
$ cd /etc/apt/sources.list.d/
```

3. 선택한 텍스트 편집기를 사용해 `buster-backports.list` 파일을 엽니다. 이 디렉터리에서 파일을 찾을 수 없는 경우 `/etc/apt/sources.list`에서 확인할 수도 있습니다. 사전 설치된 Vim 텍스트 편집기는 예제 명령에 사용됩니다. 자세한 내용은 Vim 설명서를 참조하세요.

```
$ vim buster-backports.list
```

4. `http://deb.debian.org/debian buster-backports main` 텍스트가 포함된 줄을 찾습니다.

   `deb.debian.org`를 `archive.debian.org`로 바꿉니다. 예를 들어, `http://deb.debian.org/debian buster-backports main contrib non-free`는 `http://archive.debian.org/debian buster-backports main contrib non-free`가 됩니다.
5. 파일을 저장하고 닫습니다.
6. WordPress 설정을 다시 시작합니다.

리포지토리 http://ppa.launchpad.net/certbot/certbot/ubuntu lunar 릴리스에 릴리스 파일이 없습니다.

이유

인스턴스에 업데이트할 수 없는 더 이상 사용되지 않는 Certbot Personal Package Archive(PPA) 리포지토리가 있습니다.

수정

다음 절차에 따라 인스턴스에서 더 이상 사용되지 않는 PPA 리포지토리를 수동으로 제거합니다.

절차

1. Lightsail 브라우저 기반 SSH 클라이언트 를 사용하거나 AWS CloudShell을 사용하여 인스턴스에 연결합니다.
2. `/etc/apt/sources.list.d/` 디렉터리로 이동합니다.

```
$ cd /etc/apt/sources.list.d/
```

3. 선택한 텍스트 편집기를 사용해 `certbot-ubuntu-certbot-`**`version`**`.list` 파일을 엽니다. 사전 설치된 Vim 텍스트 편집기는 예제 명령에 사용됩니다. 자세한 내용은 Vim 설명서를 참조하세요.

   명령에서 리포지토리가 호환되지 않는 Ubuntu 버전으로 **`version`**을 바꿉니다. 이는 오류 메시지에 표시되는 것과 동일한 버전입니다. 예: **`lunar`** 또는 **`mantic`**.

```
$ vim certbot-ubuntu-certbot-version.list
```

4. `http://ppa.launchpad.net/certbot/certbot/ubuntu` 텍스트가 포함된 줄을 제거합니다.
5. 파일을 저장하고 닫습니다.
6. WordPress 설정을 다시 시작합니다.

지난 168시간 동안 이 정확한 도메인 세트에 대해 이미 발급된 인증서(5)가 너무 많음

이유

하나 이상의 도메인 또는 하위 도메인이 지난 주 내에 5개의 인증서를 생성하는 데 이미 사용되었습니다. 자세한 내용은 Let's Encrypt 웹 사이트의 속도 제한을 참조하세요.

수정

1주일(168시간)을 기다린 다음 이 도메인에 대한 안내 워크플로를 다시 시작합니다.

실패한 권한 부여가 너무 많음

이유

요청의 도메인 또는 하위 도메인 중 하나 이상이 시간당 5회의 검증 한도를 초과했습니다. 자세한 내용은 Let's Encrypt 웹 사이트의 속도 제한을 참조하세요.

수정

1시간을 기다린 후 WordPress 설정을 다시 실행합니다. 설정을 다시 시작하기 전에 다른 검증 오류가 수정되었는지 확인합니다.

Lightsail

Certbot.errors.AuthorizationError: 일부 문제가 실패했습니다.

이유

이 오류는 잘못 구성된 DNS 레코드 또는 인터넷 전체에 전파할 시간이 충분하지 않은 DNS 레코드로 인해 발생합니다.

수정

A 또는 AAAA DNS 레코드가 DNS 영역에 있고 인스턴스의 퍼블릭 IP 주소를 가리키는지 확인합니다. 자세한 내용은 Lightsail의 DNS를 참조하세요.

Apex 도메인(`example.com`)과 `www` 하위 도메인(`www.example.com`)의 트래픽을 가리키는 DNS 레코드를 추가하거나 업데이트할 때 인터넷 전체에 전파해야 합니다. nslookup 또는 MxToolbox의 DNS 조회와 같은 도구를 사용하여 DNS 변경 사항이 적용되었는지 확인할 수 있습니다.

> **Note**
>
> 인터넷의 DNS를 통해 아무 DNS 레코드 변경 사항이 전파될 때까지 기다립니다. 몇 시간 정도 걸릴 수 있습니다.

Certbot이 일부 도메인을 인증하지 못했습니다.

이유

HTTPS 인증서가 인스턴스에 구성되어 있는 동안 다른 프로세스가 포트 80을 사용하는 경우 이 오류가 발생할 수 있습니다.

수정

WordPress 인스턴스를 다시 시작합니다. 그런 다음 안내 워크플로를 다시 실행합니다. 다시 시작해도 문제가 해결되지 않는 경우 다음 절차에 따라 포트 80에서 실행 중인 인스턴스에서 실행 중인 프로세스를 종료합니다.

절차

1. Lightsail 브라우저 기반 SSH 클라이언트 를 사용하거나 AWS CloudShell을 사용하여 인스턴스에 연결합니다.
2. 인스턴스에서 실행 중인 Apache 서비스를 중지합니다.

```
$ sudo systemctl stop apache2
```

Apache 서비스가 중지되었는지 확인합니다.

```
$ sudo systemctl status apache2
```

3. 포트 80을 사용하는 다른 프로세스가 있는지 확인합니다.

```
$ fuser -n tcp 80
```

4. 다른 애플리케이션에서 필요하지 않은 모든 프로세스를 종료합니다.

```
$ fuser -k -n tcp 80
```

5. WordPress 설정을 다시 시작합니다.

리포지토리 http://cdn-aws.deb.debian.org/debian 버스터-백포트에는 더 이상 릴리스 파일이 없습니다.

이유

인스턴스에 업데이트할 수 없는 더 이상 사용되지 않는 Debian 리포지토리가 있습니다.

수정

Debian 리포지토리 파일에 나열된 리포지토리 URL을 편집하려면 다음 절차를 사용합니다.

절차

1. Lightsail 브라우저 기반 SSH 클라이언트 를 사용하거나 AWS CloudShell을 사용하여 인스턴스에 연결합니다.
2. `/etc/apt/sources.list.d/` 디렉터리로 이동합니다.

```
$ cd /etc/apt/sources.list.d/
```

3. 선택한 텍스트 편집기를 사용해 `buster-backports.list` 파일을 엽니다. 이 디렉터리에서 파일을 찾을 수 없는 경우 `/etc/apt/sources.list`에서 확인할 수도 있습니다. 사전 설치된 Vim 텍스트 편집기는 예제 명령에 사용됩니다. 자세한 내용은 Vim 설명서를 참조하세요.

```
$ vim buster-backports.list
```

4. `http://deb.debian.org/debian buster-backports main` 텍스트가 포함된 줄을 찾습니다.

   `deb.debian.org`를 `archive.debian.org`로 바꿉니다. 예를 들어, `http://deb.debian.org/debian buster-backports main contrib non-free`는 `http://archive.debian.org/debian buster-backports main contrib non-free`가 됩니다.
5. 파일을 저장하고 닫습니다.
6. WordPress 설정을 다시 시작합니다.

리포지토리 http://ppa.launchpad.net/certbot/certbot/ubuntu lunar 릴리스에 릴리스 파일이 없습니다.

이유

인스턴스에 업데이트할 수 없는 더 이상 사용되지 않는 Certbot Personal Package Archive(PPA) 리포지토리가 있습니다.

수정

다음 절차에 따라 인스턴스에서 더 이상 사용되지 않는 PPA 리포지토리를 수동으로 제거합니다.

절차

1. Lightsail 브라우저 기반 SSH 클라이언트 를 사용하거나 AWS CloudShell을 사용하여 인스턴스에 연결합니다.
2. `/etc/apt/sources.list.d/` 디렉터리로 이동합니다.

```
$ cd /etc/apt/sources.list.d/
```

3. 선택한 텍스트 편집기를 사용해 `certbot-ubuntu-certbot-`**`version`**`.list` 파일을 엽니다. 사전 설치된 Vim 텍스트 편집기는 예제 명령에 사용됩니다. 자세한 내용은 Vim 설명서를 참조하세요.

   명령에서 리포지토리가 호환되지 않는 Ubuntu 버전으로 **`version`**을 바꿉니다. 이는 오류 메시지에 표시되는 것과 동일한 버전입니다. 예: **`lunar`** 또는 **`mantic`**.

```
$ vim certbot-ubuntu-certbot-version.list
```

4. `http://ppa.launchpad.net/certbot/certbot/ubuntu` 텍스트가 포함된 줄을 제거합니다.
5. 파일을 저장하고 닫습니다.
6. WordPress 설정을 다시 시작합니다.

지난 168시간 동안 이 정확한 도메인 세트에 대해 이미 발급된 인증서(5)가 너무 많음

이유

하나 이상의 도메인 또는 하위 도메인이 지난 주 내에 5개의 인증서를 생성하는 데 이미 사용되었습니다. 자세한 내용은 Let's Encrypt 웹 사이트의 속도 제한을 참조하세요.

수정

1주일(168시간)을 기다린 다음 이 도메인에 대한 안내 워크플로를 다시 시작합니다.

실패한 권한 부여가 너무 많음

이유

요청의 도메인 또는 하위 도메인 중 하나 이상이 시간당 5회의 검증 한도를 초과했습니다. 자세한 내용은 Let's Encrypt 웹 사이트의 속도 제한을 참조하세요.

수정

1시간을 기다린 후 WordPress 설정을 다시 실행합니다. 설정을 다시 시작하기 전에 다른 검증 오류가 수정되었는지 확인합니다.

Lightsail 콘솔에서 403(권한 없음) 오류 해결

Lightsail 콘솔에 액세스를 시도할 때 403 오류가 표시되는 경우 당황하지 마십시오. 문제를 해결하려면 다음 단계를 따릅니다.

- AWS 계정 또는 AWS Identity and Access Management (IAM) 사용자가 최근에 생성된 경우 몇 분 정도 기다린 다음 브라우저를 새로 고칩니다.
- 마지막으로 로그인한지 오래된 경우 브라우저를 새로고침합니다. 다시 로그인하라는 메시지가 표시되면 Lightsail 액세스 권한이 있는 IAM 사용자를 사용해야 합니다.
- IAM 사용자가 Lightsail의 액세스 권한이 없는 경우 AWS 계정 루트 사용자 또는 관리자 액세스 권한이 있는 IAM 사용자에게 문의하여 Lightsail 액세스 권한을 요청합니다. 자세히 알아보려면 IAM 사용자의 Amazon Lightsail 액세스 관리를 참조하세요.
- 위의 단계를 시도한 후에도 403 오류가 계속 발생하면 AWS Support에 문의하세요. 드문 경우지만 2011년 이전에 생성된 AWS 계정의 경우 지원에서 계정을에 수동으로 구독해야 합니다Lightsail.

Lightsail 디스크 연결 및 사용 문제 해결

Lightsail에서 블록 스토리지 디스크와 관련된 오류가 발생할 수 있습니다. 이 주제에서는 일반적인 문제와 해당 오류의 해결 방법을 알아봅니다.

일반 디스크 오류

아래에서 귀하의 문제를 가장 잘 설명한 항목을 선택한 다음, 링크를 따라 이동하여 문제를 해결하세요. 목록에 없는 문제가 발생하면 이 페이지 하단의 질문이나 의견이 있으신가요? 링크를 사용하여 피드백을 제출하거나 AWS Support에 문의하세요.

디스크가 인스턴스에 아직 연결되어 있어서 삭제할 수 없습니다.

먼저 인스턴스에서 디스크를 분리한 후 디스크를 삭제해 보십시오. 자세한 내용은 블록 스토리지 디스크 분리 및 삭제를 참조하세요.

실제 오류 메시지: 디스크가 여전히 Lightsail 인스턴스에 연결되어 있기 때문에 이 작업을 수행할 수 없습니다. ***YOUR_INSTANCE***

디스크가 오류 상태입니다.

오류 상태는 Lightsail 디스크와 관련된 기본 하드웨어에 장애가 발생했음을 나타냅니다. 최근 스냅샷에서 디스크를 복원할 수 있습니다. 그러지 않으면 디스크에 기록된 데이터를 복구할 수 없습니다. 자세한 내용은 스냅샷에서 블록 스토리지 디스크 생성을 참조하세요.

오류 상태의 디스크에 대해서는 요금이 청구되지 않습니다.

Lightsail 인스턴스가 아직 실행 중이라서 디스크를 분리할 수 없습니다.

인스턴스를 먼저 중지한 후 디스크를 분리해 보십시오. 자세한 내용은 인스턴스 중지를 참조하세요.

실제 오류 메시지: 지금은 이 디스크를 분리할 수 없습니다. 이 디스크의 상태는 다음과 같습니다. ***DISK_STATE***

16TB(16,384GB) 이상의 사용자 지정 디스크 크기를 지정할 수 없습니다.

더 작은 디스크를 만들어 보십시오. 추가 디스크 크기는 최대 16TB입니다. 디스크가 16TB 미만인데 만들 수 없다면 목록의 다음 오류(디스크가 너무 큼) 때문일 수 있습니다. AWS 계정에서 추가 디스크 스토리지가 20TB를 넘을 수 없습니다. 자세한 내용은 블록 스토리지 디스크를 참조하세요.

실제 오류 메시지: 블록 스토리지 디스크의 크기는 8~16384GB 사이여야 합니다.

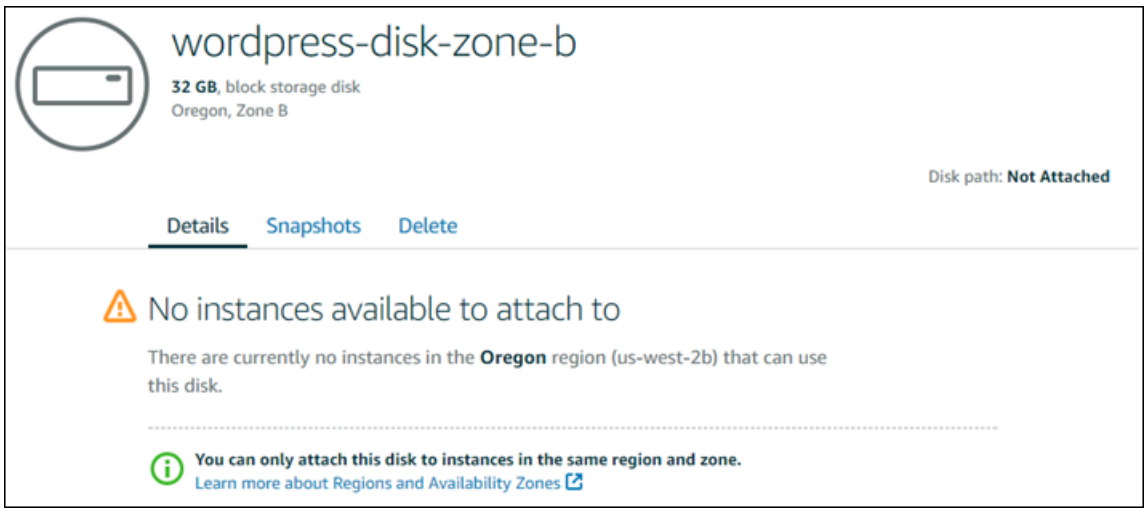
Lightsail에 디스크를 더 만들 수 없습니다.

만들 수 있는 디스크 개수 할당량에 도달했을 수 있습니다. 또는 AWS 계정이 큰 디스크를 너무 많이 만들었을 수 있습니다(디스크 스토리지 총 크기가 20TB를 넘을 수 없음). 자세한 내용은 블록 스토리지 디스크를 참조하세요.

실제 오류 메시지:이 계정의 모든 디스크가 최대 크기 한도에 도달했습니다. 또는 이 계정의 디스크가 한도에 도달했습니다.

디스크를 Lightsail 인스턴스에 연결할 수 없습니다.

다음 오류가 발생하면 디스크를 연결하려는 인스턴스와 동일한 AWS 리전 및 가용 영역에 디스크를 다시 만들어야 합니다.



실제 오류 메시지: 현재 이 디스크를 사용할 수 있는 ***AWS ##***에는 인스턴스가 없습니다.

Lightsail 브라우저 기반 SSH 및 RDP 클라이언트와의 연결 오류 해결

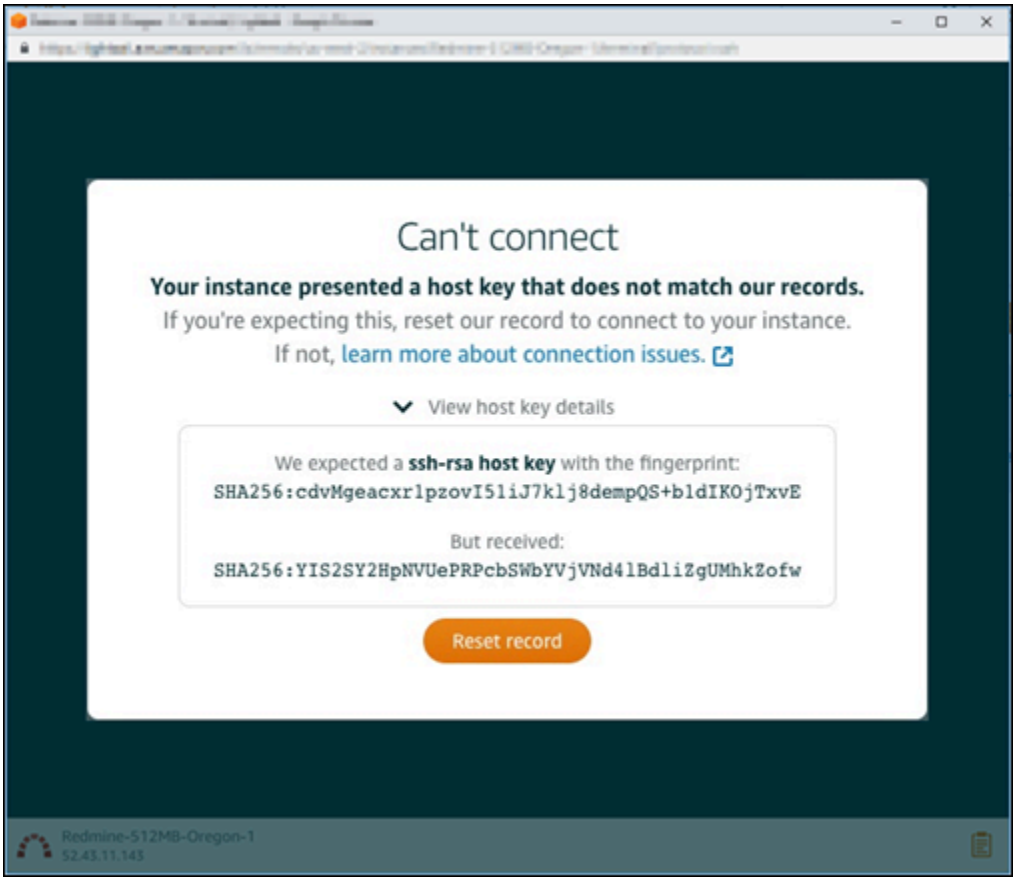
Amazon Lightsail 콘솔에서 사용 가능한 브라우저 기반 SSH 또는 RDP 클라이언트를 사용하여 인스턴스에 연결할 때 오류 메시지가 표시될 수 있습니다. 이 오류의 잠재적 원인은 다음 단원에 나와 있습니다.

오류 메시지: 연결할 수 없음

SSH 및 RDP 브라우저 기반 클라이언트는 연결 시 호스트 키 또는 인증서 확인을 사용하여 인스턴스를 인증합니다. 인스턴스가 Lightsail의 레코드가 있는 호스트 키 또는 인증서와 일치하지 않는 호스트 키 또는 인증서를 제공하면 두 개의 오류 메시지 중 하나가 표시됩니다. 이 단원에는 두 가지 오류 메시지가 모두 표시되어 있습니다.

연결할 수 없음, 레코드 재설정

호스트 키 또는 인증서 불일치가 있는 경우 다음 오류 메시지가 표시되며, Lightsail은 최근 운영 체제 업그레이드 또는 사용자나 다른 사용자에 의한 호스트 키 또는 인증서의 고의적 업데이트로 인해 불일치가 발생할 수 있다고 판단합니다. 이 경우 Lightsail은 호스트 키 또는 인증서 불일치가 브라우저와 인스턴스 간 네트워크에 있는 부적절한 요소에 의해 발생한 것이 아니라고 판단했습니다.



불일치가 예상된 경우 Reset record(레코드 재설정)를 선택합니다. 이 작업은 Lightsail에 인스턴스에 대한 레코드가 있는 호스트 키 또는 인증서를 삭제하며, 브라우저 기반 SSH 또는 RDP 세션이 인스턴스에 연결할 수 있도록 허용합니다.

다음 Lightsail(AWS Command Line Interface) 명령을 사용하여 AWS CLI에 레코드가 있는 호스트 키 또는 인증서를 삭제할 수도 있습니다. *InstanceName*에 잘 알려진 호스트 키 또는 인증서를 삭제하려는 인스턴스의 이름을 입력합니다. *##*에 인스턴스의 AWS 리전을 입력합니다.

```
aws lightsail delete-known-host-keys --region Region --instance-name InstanceName
```

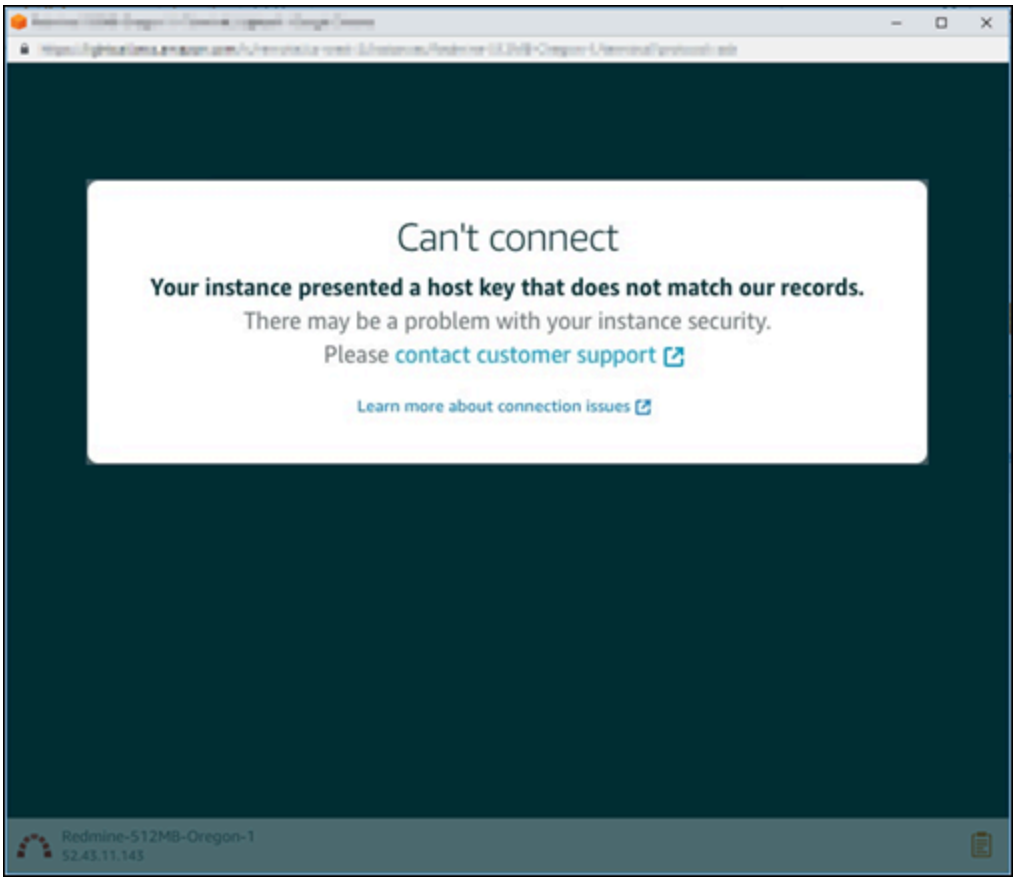
예시

```
aws lightsail delete-known-host-keys --region us-west-2 --instance-
name WordPress-512MB-Oregon-1
```

> **Note**
>
> AWS CLI에 대한 자세한 내용은 AWS CLI가 Lightsail과 함께 작동하도록 구성을 참조하세요.

연결할 수 없음, 고객 지원 센터에 문의

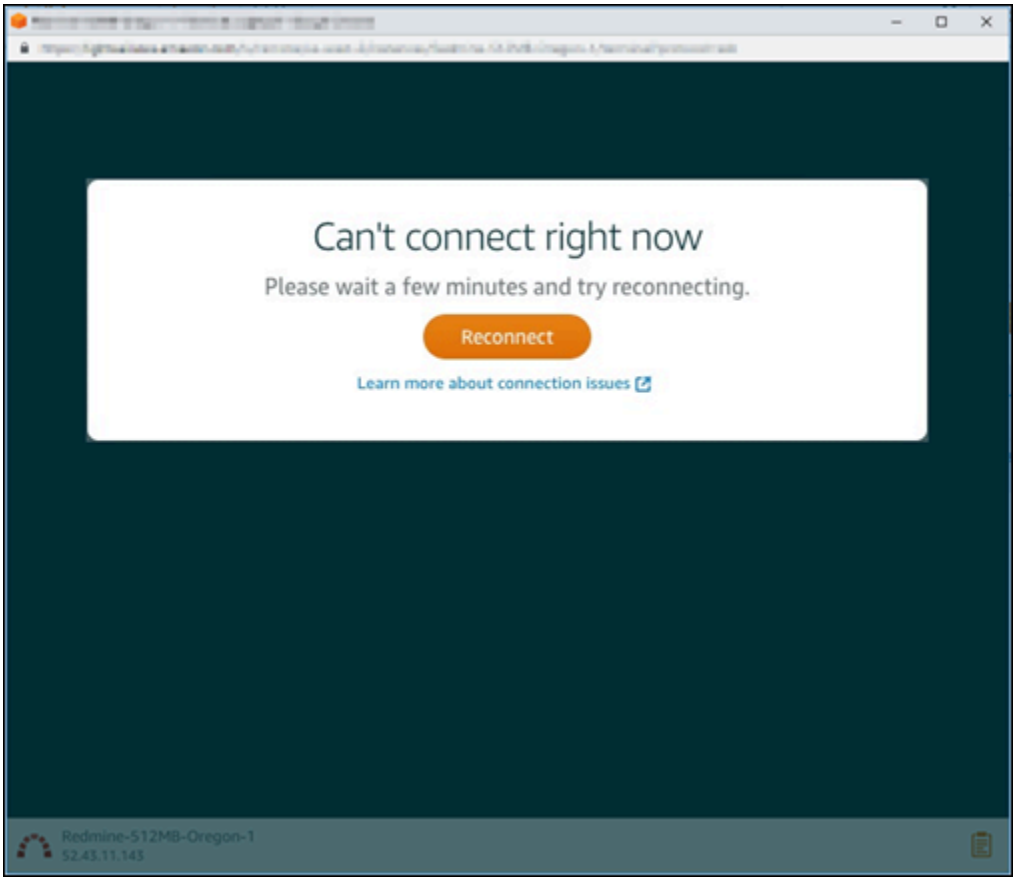
호스트 키 또는 인증서 불일치가 있는 경우 다음 오류 메시지가 표시되며, Lightsail은 중간자 공격과 같은 추가 조사가 필요한 의심스러운 활동이 있다고 판단합니다.



이 오류 메시지는 브라우저 기반 SSH 또는 RDP 클라이언트를 사용하여 인스턴스에 연결할 수 없다는 점을 의미합니다. 지원 센터에 문의하여 지원을 받으십시오.

오류 메시지: 현재 연결할 수 없음

작성, 재부팅 또는 다시 시작한 후 아직 시작되지 않은 인스턴스에 연결하려고 하면 다음 오류 메시지가 표시됩니다. 몇 분간 기다리고 다시 연결을 선택하여 다시 시도합니다.



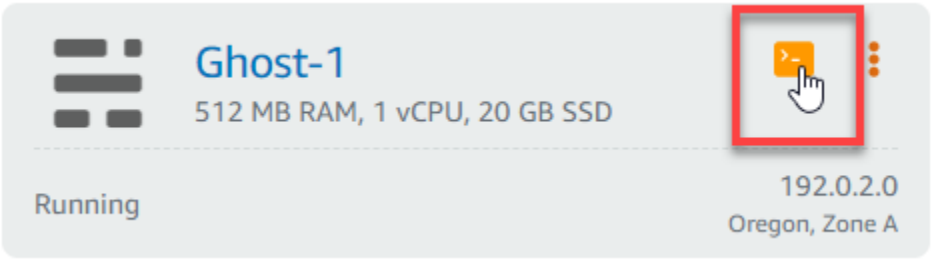
여전히 연결할 수 없는 경우 AWS Support에 문의하세요.

Lightsail에서 Ghost 인스턴스의 503 서비스 사용 불가 오류 문제 해결

Amazon Lightsail에서 새 Ghost 인스턴스를 생성하고 웹 사이트에 액세스하려고 하면 서비스를 사용할 수 없다는 오류(503)가 표시될 수 있습니다. 경우에 따라 인스턴스가 생성될 때 인스턴스의 Ghost 서비스가 자동으로 시작되지 않습니다. 이 문제는 인스턴스에 5 USD/월 번들을 선택할 때 발생할 수 있습니다. Ghost 서비스를 시작하고 발생하는 “service is unavailable(서비스 사용 불가)” 오류를 해결하려면 다음 절차를 따르세요.

Ghost 서비스 시작

1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 인스턴스를 선택합니다.
3. Ghost 인스턴스에 대한 브라우저 기반 SSH 클라이언트 아이콘을 선택합니다.



4. SSH 클라이언트가 연결되면 다음 명령을 입력하여 인스턴스의 모든 서비스를 다시 시작합니다.

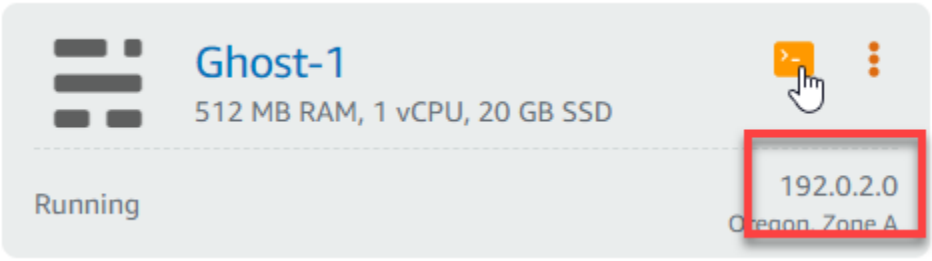
```
sudo /opt/bitnami/ctlscript.sh restart
```

다음 예와 비슷한 결과가 나타나야 합니다.

```
bitnami@ip-172-26-11-214:~$ sudo /opt/bitnami/ctlscript.sh restart
Syntax OK
/opt/bitnami/apache2/scripts/ctl.sh : httpd stopped
/opt/bitnami/apps/ghost/scripts/ctl.sh : ghost not running
/opt/bitnami/mysql/scripts/ctl.sh : mysql stopped
/opt/bitnami/mysql/scripts/ctl.sh : mysql  started at port 3306
✔ Ensuring user is not logged in as ghost user [skipped]
✔ Checking if logged in user is directory owner [skipped]
✔ Checking current folder permissions
✔ Validating config
✔ Checking memory availability
✔ Checking binary dependencies
✔ Starting Ghost: 127-0-0-1

------------------------------------------------------------------------

Your admin interface is located at:

    http://18.237.117.48:80/ghost/

/opt/bitnami/apps/ghost/scripts/ctl.sh : ghost started
Syntax OK
/opt/bitnami/apache2/scripts/ctl.sh : httpd started at port 80
```

5. 인스턴스의 퍼블릭 IP 주소로 이동하여 Ghost 웹 사이트가 실행 중인지 확인합니다.

인스턴스의 퍼블릭 IP 주소는 Lightsail 콘솔의 인스턴스 섹션에서 인스턴스 이름 옆에 나열됩니다.



새 Ghost 인스턴스의 퍼블릭 IP로 이동하면 기본 Ghost 웹 사이트 템플릿이 표시됩니다.

Lightsail에서 Identity and Access Management(IAM) 문제 해결

다음 정보를 사용하여 Lightsail 및 IAM에서 발생할 수 있는 공통적인 문제를 진단하고 수정할 수 있습니다.

Lightsail에서 작업을 수행할 권한이 없음

에서 작업을 수행할 권한이 없다는 AWS Management Console 메시지가 표시되면 관리자에게 문의하여 도움을 받아야 합니다. 관리자는 사용자 이름과 암호를 제공한 사람입니다.

다음 예제 오류는 `mateojackson` IAM 사용자가 Lightsail 콘솔에 액세스하려고 하지만 `lightsail:*`(전체 액세스) 권한이 없는 경우에 발생합니다.

이 경우 Mateo는 `lightsail:*`(full-access) 권한을 사용하여 Lightsail 콘솔에 액세스하도록 허용하는 정책을 업데이트하라고 관리자에게 요청합니다.

iam:PassRole을 수행하도록 인증되지 않음

`iam:PassRole` 작업을 수행할 수 있는 권한이 없다는 오류가 수신되면 Amazon Lightsail에 역할을 전달할 수 있도록 정책을 업데이트해야 합니다.

일부 AWS 서비스 에서는 새 서비스 역할 또는 서비스 연결 역할을 생성하는 대신 기존 역할을 해당 서비스에 전달할 수 있습니다. 이렇게 하려면 역할을 서비스에 전달할 권한이 있어야 합니다.

다음 예 오류는 marymajor라는 IAM 사용자가 콘솔을 사용하여 Amazon Lightsail에서 작업을 수행하려고 하는 경우에 발생합니다. 하지만 작업을 수행하려면 서비스 역할이 부여한 권한이 서비스에 있어야 합니다. Mary는 서비스에 역할을 전달할 권한이 없습니다.

```
User: arn:aws:iam::123456789012:user/marymajor is not authorized to perform:
 iam:PassRole
```

이 경우, Mary가 iam:PassRole작업을 수행할 수 있도록 Mary의 정책을 업데이트해야 합니다.

도움이 필요한 경우 AWS 관리자에게 문의하세요. 관리자는 로그인 자격 증명을 제공한 사람입니다.

액세스 키를 보아야 합니다.

IAM 사용자 액세스 키를 생성한 후에는 언제든지 액세스 키 ID를 볼 수 있습니다. 하지만 보안 액세스 키는 다시 볼 수 없습니다. 보안 액세스 키를 잃어버린 경우 새로운 액세스 키 페어를 생성해야 합니다.

액세스 키는 액세스 키 ID(예: AKIAIOSFODNN7EXAMPLE)와 보안 액세스 키(예: wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY)의 두 가지 부분으로 구성됩니다. 사용자 이름 및 암호와 같이 액세스 키 ID와 보안 액세스 키를 함께 사용하여 요청을 인증해야 합니다. 사용자 이름과 암호를 관리하는 것처럼 안전하게 액세스 키를 관리합니다.

> **Important**
>
> 정식 사용자 ID를 찾는 데 도움이 되더라도 액세스 키를 타사에 제공하지 마시기 바랍니다. 이렇게 하면 누군가에게에 대한 영구 액세스 권한을 부여할 수 있습니다 AWS 계정.

액세스 키 페어를 생성할 때는 액세스 키 ID와 보안 액세스 키를 안전한 위치에 저장하라는 메시지가 나타납니다. 보안 액세스 키는 생성할 때만 사용할 수 있습니다. 하지만 보안 액세스 키를 잃어버린 경우 새로운 액세스 키를 IAM 사용자에게 추가해야 합니다. 최대 두 개의 액세스 키를 가질 수 있습니다. 이미 두 개가 있는 경우 새로 생성하려면 먼저 키 페어 하나를 삭제해야 합니다. 지침을 보려면 IAM 사용 설명서의 액세스 키 관리 단원을 참조하십시오.

관리자인데, 다른 사용자가 Lightsail에 액세스할 수 있게 허용하려고 합니다

다른 사용자가 Amazon Lightsail에 액세스하도록 허용하려면 액세스가 필요한 사람 또는 애플리케이션에 권한을 부여해야 합니다. AWS IAM Identity Center 를 사용하여 사람과 애플리케이션을 관리하는 경우 사용자 또는 그룹에 권한 세트를 할당하여 액세스 수준을 정의합니다. 권한 세트는 IAM 정책을

자동으로 생성하고 사용자 또는 애플리케이션과 연결된 IAM 역할에 할당합니다. 자세한 내용은 AWS IAM Identity Center 사용 설명서에서 권한 세트를 참조하세요.

IAM Identity Center를 사용하지 않는 경우 액세스가 필요한 사람 또는 애플리케이션에 대한 IAM 엔터티(사용자 또는 역할)를 생성해야 합니다. 그런 다음 Amazon Lightsail에 대한 올바른 권한을 부여하는 정책을 엔터티에 연결해야 합니다. 권한이 부여되면 사용자 또는 애플리케이션 개발자에게 자격 증명을 제공합니다. 이들은 이 자격 증명을 사용하여 AWS에 액세스합니다. IAM 사용자, 그룹, 정책 및 권한 생성에 대해 자세히 알아보려면 IAM 사용자 설명서의 IAM 자격 증명과 IAM의 권한 및 정책을 참조하세요.

내 AWS 계정 외부의 사람이 내 Lightsail 리소스에 액세스하도록 허용하고 싶습니다.

다른 계정의 사용자 또는 조직 외부의 사람이 리소스에 액세스할 때 사용할 수 있는 역할을 생성할 수 있습니다. 역할을 수임할 신뢰할 수 있는 사람을 지정할 수 있습니다. 리소스 기반 정책 또는 액세스 제어 목록(ACL)을 지원하는 서비스의 경우, 이러한 정책을 사용하여 다른 사람에게 리소스에 대한 액세스 권한을 부여할 수 있습니다.

자세히 알아보려면 다음을 참조하세요.

- Amazon Lightsail에서 이러한 기능을 지원하는지 여부를 알아보려면 Amazon Lightsail에서 IAM을 사용하는 방식을 참조하세요.
- 소유 AWS 계정 한의 리소스에 대한 액세스 권한을 제공하는 방법을 알아보려면 IAM 사용 설명서의 소유한 다른의 IAM 사용자에게 액세스 권한 제공을 참조 AWS 계정 하세요.
- 리소스에 대한 액세스 권한을 타사에 제공하는 방법을 알아보려면 IAM 사용 설명서의 타사 AWS 계정 소유에 대한 액세스 권한 제공을 AWS 계정참조하세요.
- ID 페더레이션을 통해 액세스 권한을 제공하는 방법을 알아보려면 IAM 사용 설명서의 외부에서 인증된 사용자에게 액세스 권한 제공(ID 페더레이션)을 참조하세요.
- 크로스 계정 액세스에 대한 역할과 리소스 기반 정책 사용의 차이점을 알아보려면 IAM 사용 설명서의 IAM의 크로스 계정 리소스 액세스를 참조하세요.

Lightsail 인스턴스의 IPv6 도달 가능성 확인

Ping 도구를 사용하여 로컬 컴퓨터에서 Amazon Lightsail 인스턴스로의 IPv6 연결을 확인할 수 있습니다. Ping은 두 개 이상의 네트워크로 연결된 디바이스 간의 연결 문제를 해결하는 데 사용되는 네트워크 진단 유틸리티입니다. Ping이 성공하면 IPv6를 통해 인스턴스에 연결할 수 있어야 합니다. 네트워크

설정 또는 디바이스가 IPv6를 허용하도록 구성되지 않은 경우 ping 명령이 실패합니다. 자세한 내용은 IPv6 전용 고려 사항 섹션을 참조하세요.

내용

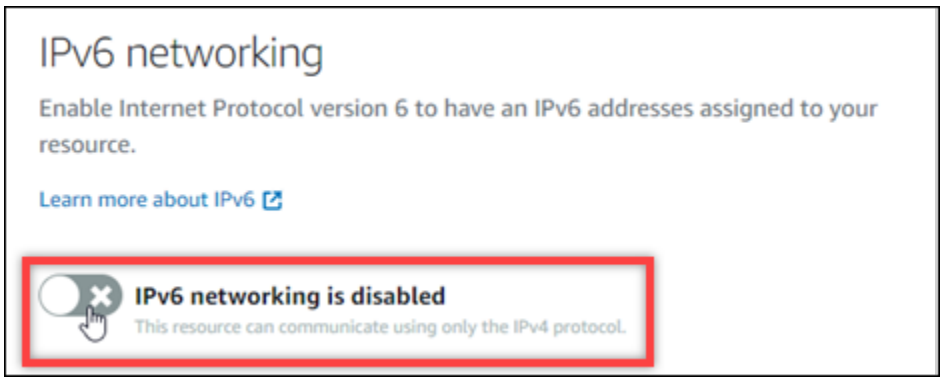
- 이중 스택 인스턴스에 IPv6 활성화
- 인스턴스의 방화벽 구성
- 인스턴스에 대한 연결 가능성 테스트

이중 스택 인스턴스에 IPv6 활성화

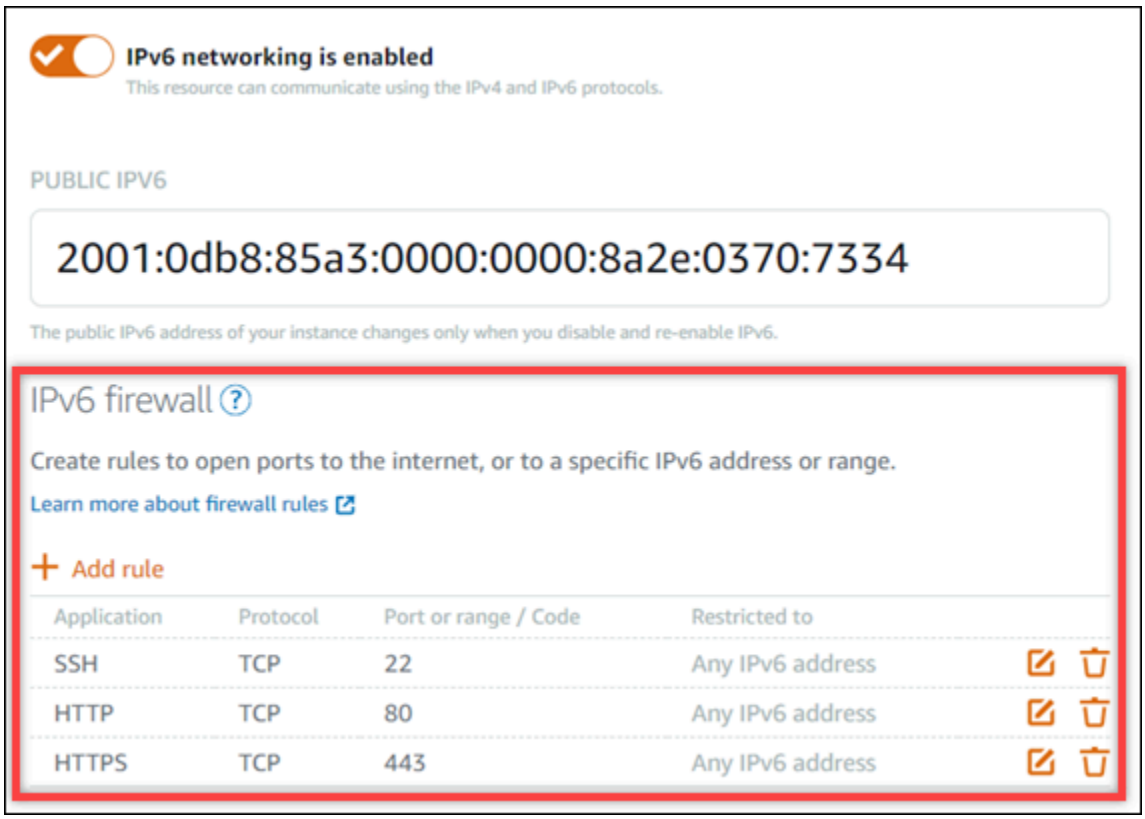
테스트를 시작하기 전에 이중 스택 인스턴스에 대해 IPv6를 활성화합니다. IPv6 전용 인스턴스의 경우 IPv6는 항상 켜져 있습니다.

이중 스택 인스턴스가 활성화되지 않은 경우 다음 절차를 완료하여 이중 스택 인스턴스에서 IPv6를 활성화합니다.

1. Lightsail 콘솔에 로그인합니다.
2. IPv6를 활성화하려는 인스턴스의 이름을 선택합니다. 인스턴스가 실행 중인지 확인합니다.
3. 인스턴스 관리 페이지에서 네트워킹 탭을 선택합니다.
4. 페이지의 IPv6 네트워킹 섹션에서 IPv6를 활성화합니다.



IPv6를 활성화하면 인스턴스에 퍼블릭 IPv6 주소가 할당되고 IPv6 방화벽을 사용할 수 있게 됩니다.



5. 페이지 상단에 인스턴스의 퍼블릭 IPv4 및 퍼블릭 IPv6 주소를 기록해 둡니다. 이 주소는 다음 섹션에서 사용합니다.

인스턴스의 방화벽 구성

Lightsail 콘솔의 방화벽은 가상 방화벽 역할을 합니다. 즉, 퍼블릭 IP 주소를 통해 인스턴스에 연결할 수 있는 트래픽을 제어합니다. Lightsail에서 생성하는 각 이중 스택 인스턴스에는 IPv4 주소용과 IPv6 주소용의 개별 방화벽 2개가 있습니다. 각 방화벽에는 인스턴스로 들어오는 트래픽을 필터링하는 규칙 집합이 포함되어 있습니다. 2개의 방화벽은 서로 독립적이므로, IPv4 및 IPv6 각각에 대해 별도로 방화벽 규칙을 구성해야 합니다. IPv6 전용 인스턴스 플랜이 있는 인스턴스에는 구성할 수 있는 IPv4 방화벽이 없습니다.

다음 절차를 완료하여 인터넷 제어 메시지 프로토콜(ICMP) 트래픽에 대해 인스턴스의 방화벽을 구성합니다. Ping 유틸리티는 ICMP 프로토콜을 사용하여 인스턴스와 통신합니다. 자세한 내용은 Lightsail의 방화벽을 사용하여 인스턴스 트래픽 제어 단원을 참조하십시오.

> **Important**
>
> Windows 및 Linux에는 ping 명령을 차단할 수 있는 운영 체제(OS) 수준 방화벽이 포함되어 있습니다. 계속하기 전에 인스턴스의 OS 방화벽이 IPv4 및 IPv6를 통한 ICMP 트래픽을 수락할 수 있는지 확인합니다. 자세한 내용은 다음 설명서를 참조하세요.
>
> - RDP를 사용하여 Lightsail Windows 인스턴스에 연결
> - Lightsail에 Linux 또는 Unix 인스턴스에 연결

1. Lightsail 콘솔에 로그인합니다.
2. 방화벽 규칙을 설정할 인스턴스의 이름을 선택합니다.
3. 인스턴스 관리 페이지에서 네트워킹 탭을 선택한 다음 사용하려는 방화벽 유형에 해당하는 섹션의 나머지 단계를 완료합니다. IPv4의 경우 IPv4 방화벽 섹션의 단계를 완료합니다. IPv6의 경우 IPv6 방화벽 섹션의 단계를 완료합니다.

   a. 애플리케이션 드롭다운 메뉴에서 Ping(ICMP)을 선택합니다.

   b. IP 주소로 제한 상자를 선택하여 로컬 소스 IP 주소 또는 범위에서 연결을 허용한 다음 소스 IP 주소를 입력합니다. (선택 사항) 모든 IP 주소에서 연결을 허용하도록 상자를 선택하지 않은 상태로 둘 수 있습니다. 테스트 환경에서만 이 옵션을 사용하는 것이 좋습니다.

   c. 생성을 선택하여 새 규칙을 인스턴스에 적용합니다.

인스턴스에 대한 연결 가능성 테스트

로컬 컴퓨터 또는 네트워크에서 Lightsail 인스턴스로 IPv4 또는 IPv6 연결 가능성을 테스트하려면 다음 절차를 완료합니다. Step 5에 기록한 인스턴스의 퍼블릭 IPv4 및 IPv6 주소가 필요합니다.

Linux, Unix 또는 macOS 디바이스에서

1. 로컬 시스템에서 터미널 창을 엽니다.
2. Lightsail 인스턴스를 ping하려면 다음과 같은 명령 중 하나를 사용합니다. 명령에 있는 예제 *IP # #*를 인스턴스의 퍼블릭 IPv4 또는 IPv6 주소로 바꿉니다.

   IPv4를 통해 테스트

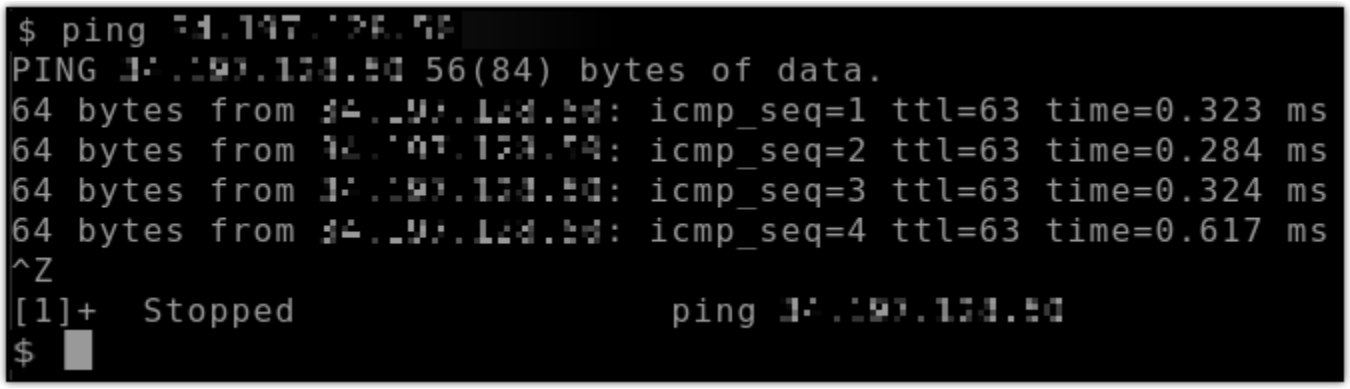
   ```
   ping 192.0.2.0
   ```

IPv6를 통해 테스트

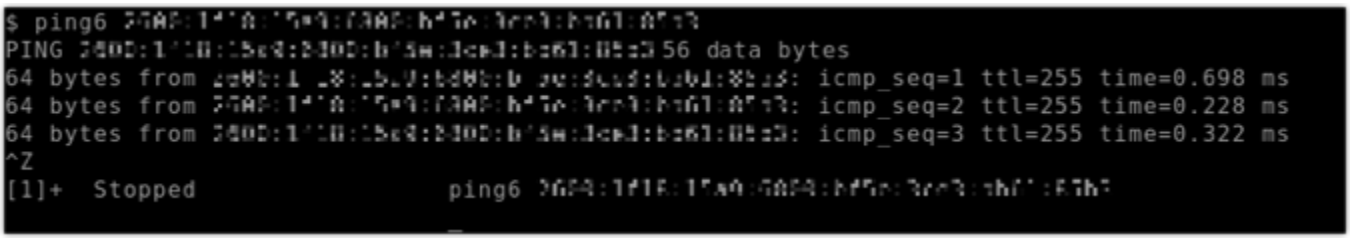
```
ping6 2001:db8::
```

3. 명령이 몇 개의 응답을 반환한 후 디바이스의 키보드에 ctrl+z를 입력하여 명령을 중지합니다.

Ping 명령은 성공하면 인스턴스의 IPv4 주소에서 성공적인 응답을 반환합니다. 결과는 다음 예제와 같아야 합니다.



Ping6 명령은 성공하면 인스턴스의 IPv6 주소에서 성공적인 응답을 반환합니다. 결과는 다음 예제와 같아야 합니다.



인스턴스에 연결할 수 없는 경우 두 명령 모두 요청 제한 시간을 반환합니다.

Windows 디바이스에서

1. 명령 프롬프트를 엽니다.
2. Lightsail 인스턴스를 ping하려면 다음과 같은 명령 중 하나를 사용합니다. 명령에 있는 예제 *IP # #*를 인스턴스의 퍼블릭 IPv4 또는 IPv6 주소로 바꿉니다.

   IPv4를 통해 테스트

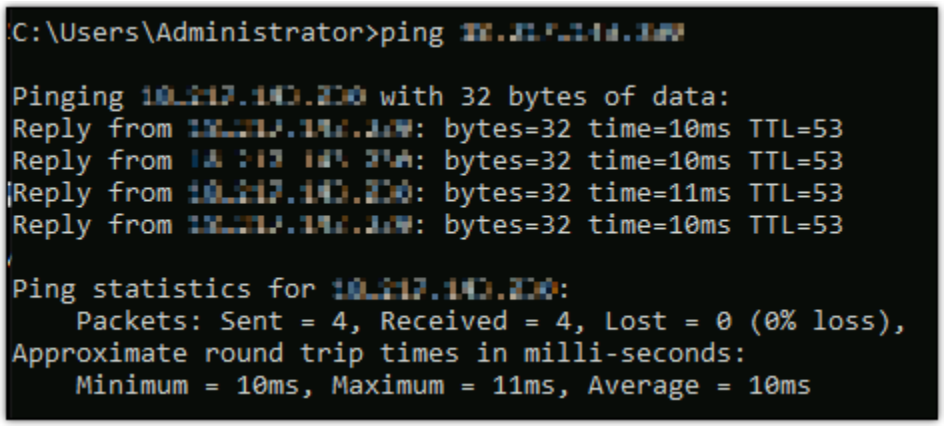
   ```
   ping 192.0.2.0
   ```

IPv6를 통해 테스트

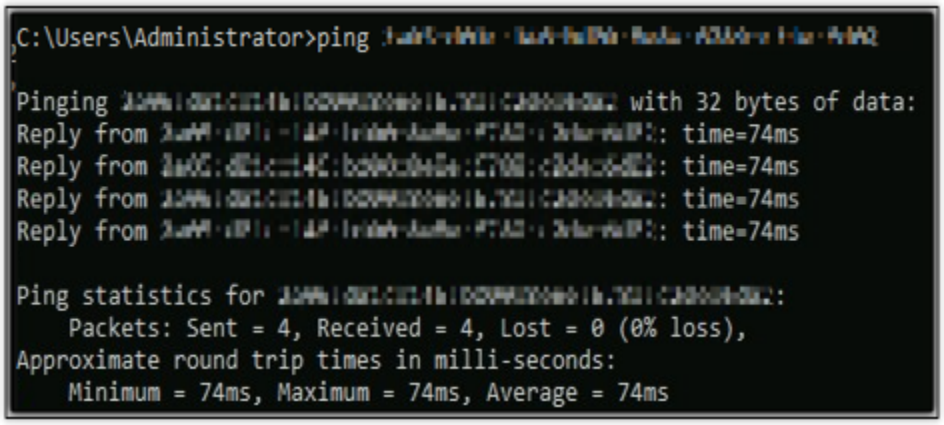
```
ping 2001:db8::
```

3. 명령이 몇 개의 응답을 반환한 후 디바이스의 키보드에 ctrl+z를 입력하여 명령을 중지합니다.

Ping 명령은 성공하면 인스턴스의 IPv4 주소에서 성공적인 응답을 반환합니다. 결과는 다음 예제와 같아야 합니다.



Ping 명령은 성공하면 인스턴스의 IPv6 주소에서 성공적인 응답을 반환합니다. 결과는 다음 예제와 같아야 합니다.



인스턴스에 연결할 수 없는 경우 두 명령 모두 요청 제한 시간을 반환합니다.

Lightsail에서 인스턴스 용량 부족 오류 해결

인스턴스를 시작하려 할 때 또는 중지된 인스턴스를 다시 시작하려 할 때 용량 부족 오류가 발생할 수 있습니다. 즉, 에는 현재 요청을 이행하는 데 사용할 수 있는 인스턴스 용량이 AWS 없습니다. 다음은 인스턴스 용량 부족 오류의 예입니다.

InsufficientInstanceCapacity: There is not enough capacity to fulfill your instance request. Reduce the number of instances in your request, or wait for additional capacity to become available. You can also try launching an instance by selecting a smaller Lightsail plan (which you can resize at a later stage).

이 가이드에서는 인스턴스 용량 부족 오류가 발생할 경우 취할 수 있는 조치에 대해 알아봅니다.

목차

- 새 인스턴스 시작 시 용량 부족
- 중지된 인스턴스 시작 시 용량 부족
- 관련 정보

새 인스턴스 시작 시 용량 부족

새 인스턴스 시작 시 인스턴스 용량 부족 오류가 발생하는 경우 다음 옵션을 사용하세요. 각 옵션을 순서대로 완료하거나 자신에게 맞는 옵션을 선택할 수 있습니다.

1. 몇 분 정도 기다린 후 다시 요청을 제출합니다. 인스턴스 용량은 자주 변할 수 있습니다. 몇 분 정도 기다린 후에도 인스턴스를 생성할 수 없는 경우 옵션 2를 진행합니다.
2. 인스턴스를 생성하는 중인 경우 다른 가용 영역을 선택합니다. 각 AWS 리전 에는 3개 이상의 AZ가 포함되어 있으며, 각 AZ는 서로 다른 인스턴스 용량을 유지합니다. 다른 AZ를 선택하면 해당 AZ의 현재 인스턴스 용량을 활용할 수 있습니다. 다른 AWS 리전 또는 AZ에서 인스턴스를 생성할 수 없는 경우 옵션 3을 계속 진행합니다.
3. 요청의 인스턴스 수를 줄입니다. 동시에 여러 인스턴스를 생성하는 경우 인스턴스 수를 줄이고 요청을 다시 제출합니다. 인스턴스 수를 줄여도 문제가 해결되지 않으면 옵션 4를 진행합니다.
4. 인스턴스를 생성할 때 다른 인스턴스 플랜을 선택합니다. 다른 AZ 또는 리전에서 인스턴스를 생성할 수 없는 경우 다른 인스턴스 플랜을 선택합니다. 이후 단계에서 인스턴스 크기를 조정할 수 있습니다. 인스턴스 크기 조정에 대한 자세한 내용은 스냅샷에서 인스턴스 생성을 참조하세요.

중지된 인스턴스 시작 시 용량 부족

이전에 중지된 기존 인스턴스를 시작할 때 인스턴스 용량 부족 오류가 발생하는 경우 다음 옵션을 사용하세요.

1. 몇 분 정도 기다린 후 다시 요청을 제출합니다. 인스턴스 용량은 자주 변할 수 있습니다. 몇 분 정도 기다린 후에도 인스턴스를 생성할 수 없는 경우 옵션 2를 진행합니다.
2. 스냅샷에서 새 인스턴스를 생성합니다. 중지된 인스턴스의 스냅샷을 만듭니다. 그런 다음 스냅샷을 사용하여 원래 인스턴스와 다른 새 인스턴스를 AZ에 생성합니다. 예를 들어 인스턴스가 현재 us-east-2a(영역 A)에 있는 경우 새 인스턴스를 생성할 때 us-east-2c(영역 C)를 선택합니다. 자세한 내용은 스냅샷에서 인스턴스 생성을 참조하세요.
3. 스냅샷에서 새 인스턴스를 생성할 때 다른 인스턴스 플랜을 선택할 수도 있습니다. 이 작업은 선택 사항입니다.

> **Important**
>
> 새 인스턴스를 실행한 후에는 새 인스턴스에 액세스할 수 있고 모든 것이 제대로 작동하고 있는지 확인하세요. 예를 들어 인스턴스에서 애플리케이션을 실행 중이었다면 애플리케이션이 예상대로 작동하는지 확인합니다. 문제가 없다면 이전 인스턴스를 삭제할 수 있습니다.

관련 정보

자주 묻는 질문

Lightsail의 복원력

Lightsail 로드 밸런서 문제 해결

Lightsail 로드 밸런서와 관련된 오류가 발생할 수 있습니다. 이 주제에서는 일반적인 문제와 해당 오류의 해결 방법을 알아봅니다.

일반적인 로드 밸런서 오류

아래에서 귀하의 문제를 가장 잘 설명한 항목을 선택한 다음, 링크를 따라 이동하여 문제를 해결하세요. 목록에 없는 문제가 발생하면 이 페이지 하단의 질문이나 의견이 있으신가요? 링크를 사용하여 피드백을 제출하거나 AWS 고객 지원 센터에 문의하세요.

인증서를 생성할 수 없습니다.

AWS 계정 하나에서 생성할 수 있는 인증서 수에 대한 할당량이 있습니다. 자세한 내용은 AWS Certificate Manager 사용 설명서의 할당량을 참조하세요. 로드 밸런서용 Lightsail 인증서에도 동일한 할당량이 적용됩니다.

실제 오류 메시지: 죄송합니다만 귀하의 계정에서 너무 많은 인증서를 요청하셨습니다.

내 로드 밸런서에 인스턴스를 더 이상 연결할 수 없습니다.

Lightsail 계정 하나당 Lightsail 인스턴스 수가 총 20개 할당량을 넘지 않는 한 원하는 수만큼 AWS 인스턴스를 로드 밸런서에 추가할 수 있습니다.

실제 오류 메시지: 죄송합니다만, 이 로드 밸런서에 연결할 수 있는 최대 인스턴스 수에 도달했습니다.

내 로드 밸런서에 특정 인스턴스를 연결할 수 없습니다.

먼저 Lightsail 인스턴스가 실행 중인지 확인하세요. 인스턴스가 중지되었으면 인스턴스 관리 페이지에서 시작할 수 있습니다. Lightsail 인스턴스가 실행되고 있어야 로드 밸런서에 제대로 연결됩니다.

동일한 인스턴스를 너무 많은 로드 밸런서에 연결했을 가능성이 있습니다.

실제 오류 메시지: 죄송합니다만, 로드 밸런서에 인스턴스를 등록할 수 있는 최대 횟수에 도달했습니다.

Lightsail내 로드 밸런서에 연결하려는 인스턴스를 이 찾지 못합니다.

더 이상 존재하지 않는 인스턴스 또는 대상 그룹과 다른 VPC에 속하는 인스턴스를 연결하려 했을 가능성이 있습니다.

실제 오류 메시지: 죄송합니다만, 지정하신 인스턴스는 존재하지 않거나 대상 그룹과 동일한 VPC에 없거나 지원되지 않은 인스턴스 유형입니다.

Lightsail에서 알림 전송 문제 해결

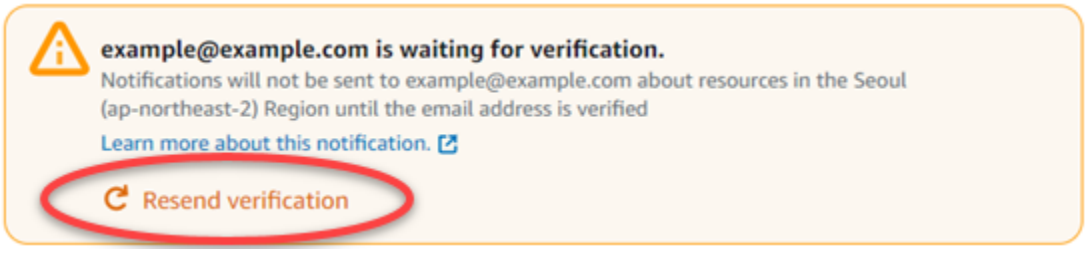
알림을 받아야 할 때 알림이 수신되지 않는 경우 몇 가지 사항을 점검하여 알림 연락처가 올바르게 구성되었는지 확인해야 합니다. 알림에 대한 자세한 내용은 알림을 참조하세요.

다음 목록에서는 발생할 수 있는 일반적인 알림 연락처 문제와 문제의 원인 및 해결 방법을 설명합니다. 목록에 없는 문제가 발생하면 이 페이지 하단의 질문이나 의견이 있으신가요? 링크를 사용하여 피드백을 제출하거나 AWS Support Center에 문의하세요.

이메일 주소를 알림 연락처로 추가했지만 이메일 알림이 수신되지 않음

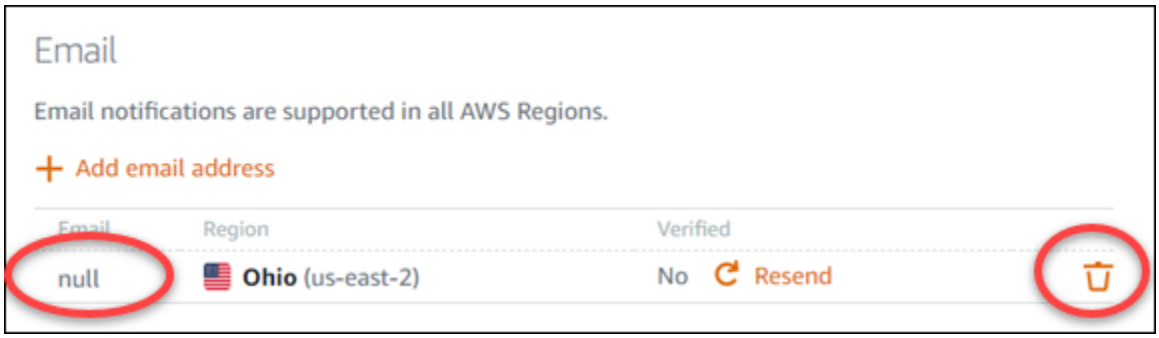
Lightsail에서 이메일 주소를 알림 연락처로 추가하면 해당 주소로 확인 요청이 전송됩니다. 수신자는 확인 요청 이메일에 포함된 링크를 클릭하여 Lightsail 알림을 받을 것임을 확인해야 합니다. 이메일 주소에 대한 확인 프로세스를 완료해야만 알림이 전송됩니다. 이 확인은 AWS Notifications <no-reply@sns.amazonaws.com>에서 AWS Notification - Subscription Confirmation이라는 제목으로 전송됩니다. SMS 메시징은 확인이 필요하지 않습니다.

확인 요청이 받은 편지함 폴더에 없는 경우 사서함의 스팸 및 정크 폴더를 확인합니다. 확인 요청이 유실되었거나 삭제된 경우 Lightsail 콘솔과 계정(Account) 페이지에 표시되는 알림 배너에서 확인 재전송(Resend verification)을 선택합니다.



이메일 알림 연락처로 null이 표시됨

이메일 주소를 추가한 후 24시간 이내에 확인 프로세스를 완료해야 합니다. 24시간 이내에 이메일 주소를 확인하지 않으면 이메일 주소가 자동으로 `invalid` 상태로 지정되고 Lightsail에서 제거됩니다. 따라서 하나 이상의 이메일 알림 연락처에 null 값이 표시될 수 있습니다.



이 문제를 해결하려면 null 이메일 알림 연락처를 제거하고 올바른 이메일 주소를 다시 추가합니다. 이메일 주소를 Lightsail에 추가한 후 즉시 확인해야 합니다. 자세한 내용은 알림을 참조하세요.

SMS 문자 메시지 알림이 수신되지 않거나 최근에 수신이 중지됨

SMS 문자 메시지 알림 수신을 거부했을 수 있습니다. `ARRET`(프랑스어), `CANCEL`, `END`, `OPT-OUT`, `OPTOUT`, `QUIT`, `REMOVE`, `STOP`, `TD` 또는 `UNSUBSCRIBE`를 사용해 SMS 문자 메시지 알림에 응답하여 수신을 거부할 수 있습니다. 알림 수신을 거부한 휴대폰 번호는 30일을 기다린 후에 Lightsail에서 알림 연락처로 다시 추가할 수 있습니다.

Lightsail의 SSL/TLS 인증서 문제 해결

Lightsail 로드 밸런서와 관련된 오류가 발생할 수 있습니다. 이 주제에서는 일반적인 문제와 해당 오류의 해결 방법을 알아봅니다.

아래에서 귀하의 문제를 가장 잘 설명한 항목을 선택한 다음, 링크를 따라 이동하여 문제를 해결하세요. 목록에 없는 문제가 발생하면 이 페이지 하단의 질문이나 의견이 있으신가요? 링크를 사용하여 피드백을 제출하거나 AWS 고객 지원 센터에 문의하세요.

인증서를 생성할 수 없습니다.

AWS 계정에서 생성할 수 있는 인증서 수에는 할당량이 있습니다. 자세한 내용은 AWS Certificate Manager 사용 설명서의 할당량을 참조하세요. 로드 밸런서용 Lightsail 인증서에도 동일한 할당량이 적용됩니다.

실제 오류 메시지: 죄송합니다만 귀하의 계정에서 너무 많은 인증서를 요청하셨습니다.

제 인증서 요청이 실패했습니다.

인증서 요청이 실패한 경우 로드 밸런서 관리 페이지의 인바운드 트래픽(Inbound traffic) 탭에서 재시도할 수 있습니다.

그래도 문제를 밝혀낼 수 없는 경우에는 AWS 고객 지원 센터에 문의하십시오.

제 도메인이 잘못된 것으로 표시됩니다.

도메인 관리자라는 사실을 입증하기 어려울 때는 DNS 관리에 대한 액세스 권한이 있는 것으로 나오는지 확인하십시오. 권한이 있고 이 지침에 따랐는데도 입증하기가 어려우면 AWS 고객 지원 센터에 문의하십시오.

자습서를 사용하여 Lightsail 기능 탐색

이 섹션은 Amazon Lightsail과 관련된 주제를 포함합니다.

주제

- Lightsail 블루프린트를 사용하여 애플리케이션을 빠르게 배포
- Lightsail에서 Bitnami 애플리케이션 및 스택 작업
- Lightsail WordPress 인스턴스 구성 및 관리
- Lightsail에서 다중 사이트를 사용하여 여러 WordPress 사이트 관리
- Let's Encrypt를 사용하여 Lightsail 리소스에 대해 암호화된 통신 활성화
- Lightsail 인스턴스 IPv6 네트워킹 구성
- Lightsail 작업을 AWS CLI 위한 설정 및 구성
- 를 사용하여 Lightsail 리소스 관리 AWS CloudShell
- Lightsail LAMP 인스턴스에 PHP 애플리케이션 배포
- Lightsail에서 Windows Server 2016 인스턴스 시작 및 구성
- 를 사용하여 Lightsail API 활동 모니터링 AWS CloudTrail
- Lightsail 문제를 해결하기 위한 HAR 파일 생성
- Lightsail에서 Prometheus를 사용하여 시스템 리소스 및 앱 모니터링
- Lightsail에서 scp를 사용하여 Linux 인스턴스 간에 파일 전송
- Lightsail 다른 AWS 서비스와 VPC 피어링 통합
- 를 사용하여 Lightsail 리소스 생성 AWS CloudFormation
- 앱 배포를 위한 Lightsail 리소스 살펴보기

각 범주에 제공된 링크를 따라 단계별 가이드, 모범 사례 및 Lightsail 작업의 다양한 측면에 대한 추가 정보에 액세스합니다.

각 주제에서는 애플리케이션 배포, 네트워킹 구성, 모니터링 및 로깅, 다른 AWS 서비스와의 통합 등과 같은 정보를 다룹니다. 이 단원을 살펴보면를 효과적으로 활용하고Lightsail, 다른 AWS 서비스와의 통합을 활용하고, 풍부한 자습서 및 리소스에 액세스하여 클라우드 컴퓨팅 환경을 개선하는 방법을 배울 수 있습니다.

Lightsail 블루프린트를 사용하여 애플리케이션을 빠르게 배포

Lightsail 청사진을 시작하려면 다음 빠른 시작 안내서를 사용하세요. Lightsail 청사진은 운영 체제 및 애플리케이션과 함께 사전 패키징된 가상 이미지입니다. 애플리케이션에는 WordPress, WordPress Multisite, cPanel & WHM, PrestaShop, Drupal, Ghost, Joomla!, Magento, Redmine, Ruby on Rails, LAMP, Nginx(LEMP) 및 Node.js가 포함됩니다.

주제

- Lightsail에서 AlmaLinux 인스턴스 시작 및 설정
- Lightsail에서 cPanel 및 WHM을 사용하여 웹 사이트, 이메일 및 서비스 호스팅
- Lightsail에서 Drupal 웹 사이트 설정 및 사용자 지정
- Lightsail에 Ghost 웹 사이트 배포
- Lightsail에서 GitLab CE 인스턴스 설정 및 구성
- Lightsail에서 Joomla! 시작하기
- 에서 LAMP 스택 배포 및 관리 Lightsail
- Lightsail에 Magento를 설정 및 구성
- Lightsail에서 Nginx 웹 서버 배포 및 관리
- 에서 Node.js 스택 배포 및 관리 Lightsail
- 에서 OpenClaw 시작하기 Lightsail
- Lightsail에 Plesk 호스팅 스택 배포
- Lightsail에서 PrestaShop 웹 사이트 설정
- Lightsail에서 Redmine 인스턴스 구성 및 보호
- 에서 Ruby on Rails 설정 Lightsail
- 에서 WordPress 배포 및 관리 Lightsail
- Lightsail에서 WordPress 멀티사이트 설정

Lightsail에서 AlmaLinux 인스턴스 시작 및 설정

> 알고 계셨나요?
>
> Lightsail는 일별 스냅샷 7개를 저장하고 인스턴스에 대해 자동 스냅샷을 활성화하면 가장 오래된 스냅샷을 최신 스냅샷으로 자동 대체합니다. 자세한 내용은 Lightsail 인스턴스 및 디스크의 자동 스냅샷 구성을 참조하세요.

이 빠른 시작 안내서는 Amazon Lightsail 플랫폼에서 AlmaLinux 인스턴스를 생성하고 구성하기 위한 단계별 지침을 제공합니다. 이 주제에서는 인스턴스 위치 및 플랜 선택, 네트워킹 및 보안 설정, CentOS에서 AlmaLinux로 전환 등 주요 단계를 다룹니다. 다음 단계에 따라 Lightsail에서 AlmaLinux 인스턴스를 빠르게 시작하고 실행할 수 있습니다.

주제

- 사전 조건
- Lightsail에서 AlmaLinux 인스턴스 생성
- (선택 사항) 추가 설정
- Lightsail에서 CentOS에서 AlmaLinux로 데이터 마이그레이션

사전 조건

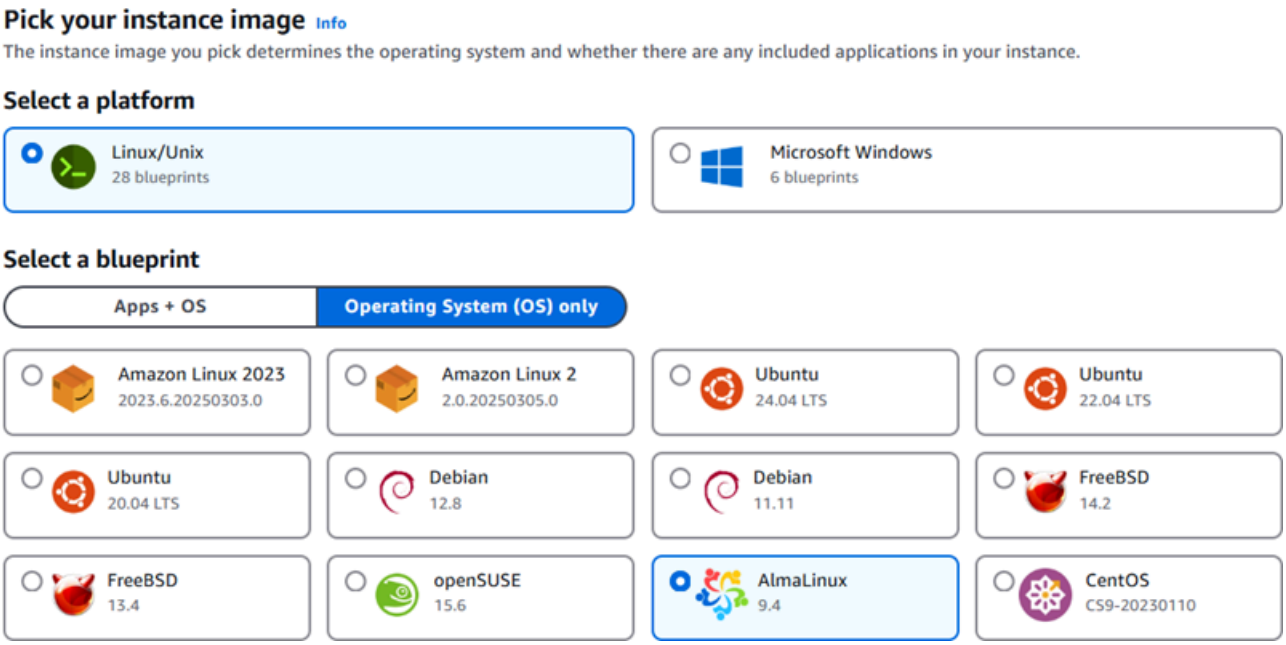
- 신규 AWS 고객인 경우 사용을 시작하기 전에 설정 사전 조건을 완료하세요Amazon Lightsail. 자세한 내용은 에 대한 AWS 계정 및 관리 사용자 설정 Lightsail 단원을 참조하십시오.
- AlmaLinux Wiki 사이트에서 AlmaLinux 설명서를 읽습니다.

Lightsail에서 AlmaLinux 인스턴스 생성

Lightsail 콘솔을 사용하여 AlmaLinux 인스턴스를 생성하려면 다음 절차를 완료합니다.

1. Lightsail 콘솔에 로그인합니다.
2. 홈 페이지에서 인스턴스 생성을 선택합니다.
3. 인스턴스의 위치(AWS 리전 및 가용 영역)를 선택합니다. 지연 시간을 줄 AWS 리전 이려면 물리적 위치와 가장 가까운를 선택합니다.

   다른 위치에 인스턴스를 생성하려면 가용 영역 변경을 선택합니다.
4. Linux 플랫폼을 선택합니다.
5. 운영 체제(OS) 전용을 선택한 다음 AlmaLinux 블루프린트를 선택합니다.



6. 선택적으로 다음을 수행할 수 있습니다.

   a. 인스턴스가 처음 시작될 때 실행될 쉘 스크립트를 추가하려면 시작 스크립트 추가를 선택합니다. 자세한 내용은 Lightsail에서 시작 스크립트를 사용하여 Linux/Unix 인스턴스 구성 단원을 참조하십시오.

   b. 인스턴스의 SSH 키 페어를 변경하려면 SSH 키 아래의 드롭다운 목록에서 키를 선택합니다. 자세한 내용은 Lightsail에 대한 SSH 키 설정 단원을 참조하십시오.

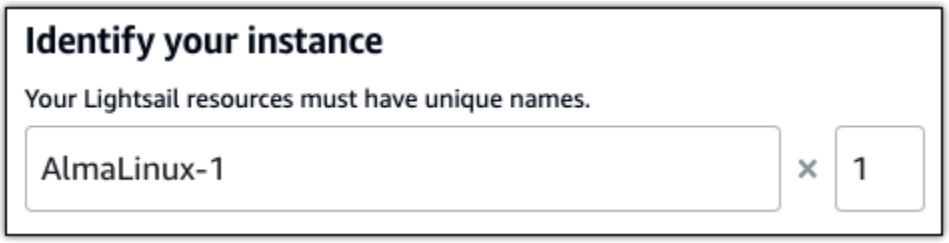
   c. 자동 스냅샷 활성화를 선택하여 인스턴스 및 연결된 디스크에 대해 자동 스냅샷을 활성화합니다. 자세한 내용은 Lightsail 인스턴스 및 디스크의 자동 스냅샷 구성 섹션을 참조하세요.

7. 인스턴스 플랜을 선택합니다. 인스턴스가 이중 스택(IPv4 및 IPv6)을 사용할지 IPv6 전용을 사용할지 선택할 수 있습니다. AlmaLinux 블루프린트는 이중 스택 및 IPv6 전용 번들을 모두 지원합니다. IPv6 전용 네트워킹에 대한 자세한 내용은 Lightsail 인스턴스 IPv6 전용 네트워킹 구성 섹션을 참조하세요.

8. 인스턴스 이름을 입력합니다.

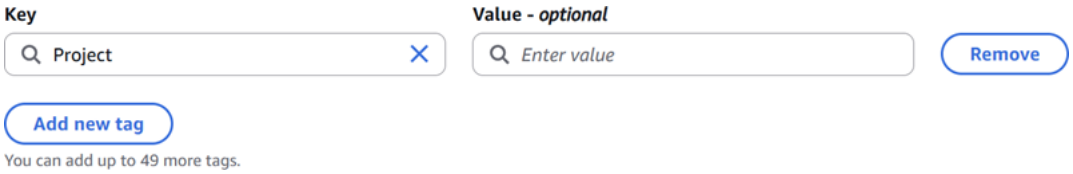
   리소스 이름:

   - Lightsail 계정 AWS 리전 의 각 내에서 고유해야 합니다.
   - 2~255자의 문자로 구성되어야 합니다.
   - 영숫자 문자 또는 숫자로 시작하고 끝나야 합니다.
   - 영숫자 문자, 숫자, 마침표, 대시, 밑줄이 포함될 수 있습니다.

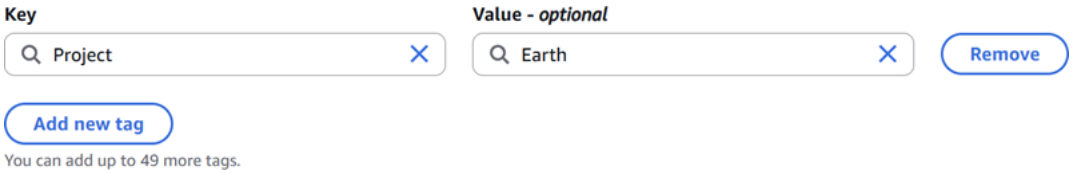


9. (선택 사항) 인스턴스에 태그를 추가하려면 새 태그 추가를 선택합니다. 필요에 따라 이 단계를 반복하여 모든 태그를 추가합니다. 태그 사용에 대한 자세한 내용은 태그를 참조하세요.

   a. 키에는 태그 키를 입력합니다.

   b. (선택 사항) 값에 태그 값을 입력합니다.

10. 인스턴스 생성을 선택합니다.

Lightsail 인스턴스는 몇 분 내에 준비되며 연결할 수 있습니다.

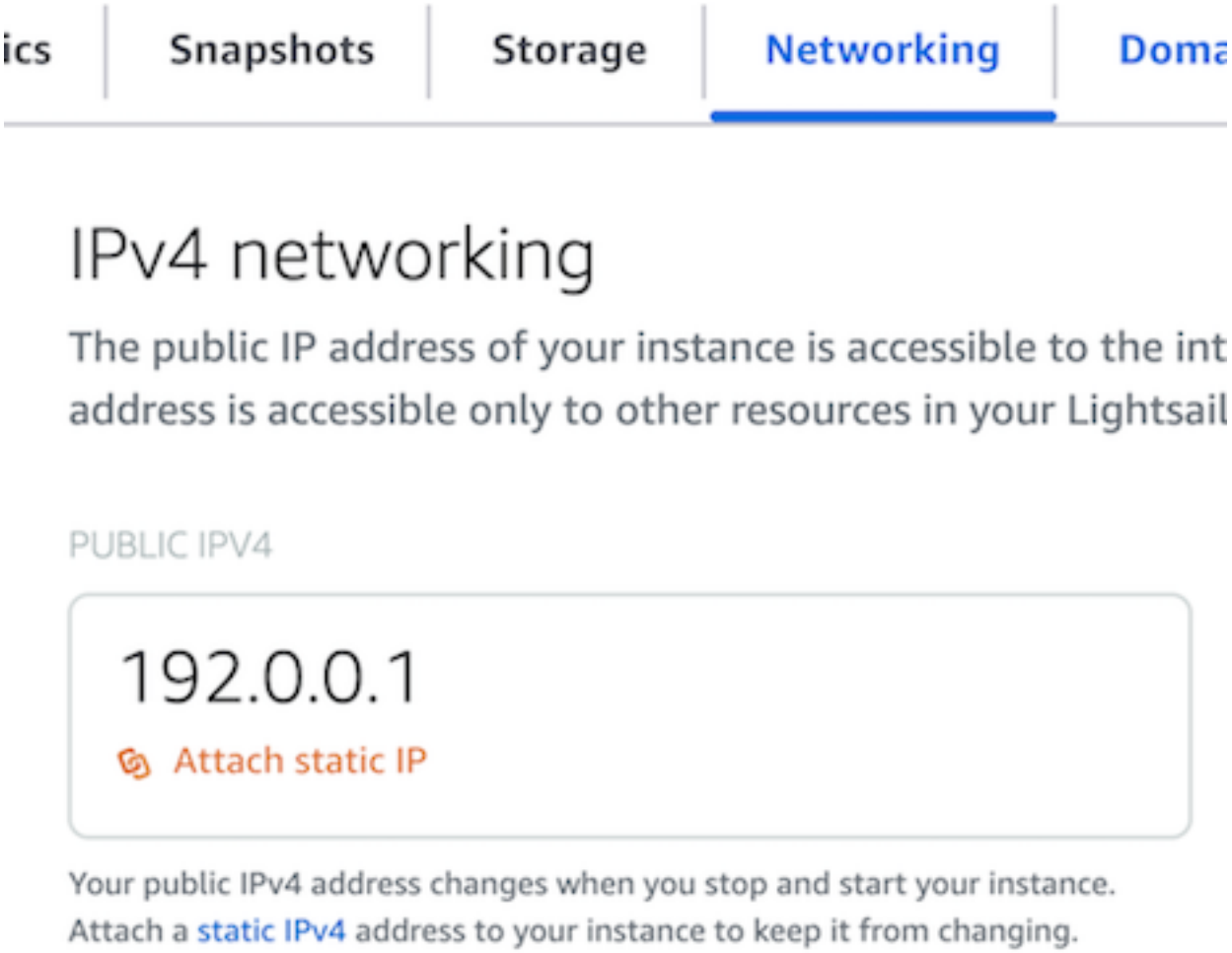
(선택 사항) 추가 설정

다음은 Lightsail에 AlmaLinux 인스턴스를 설치하고 실행한 후 시작해야 할 몇 가지 단계입니다.

- 인스턴스에 고정 IP 주소 연결 - 인스턴스에 연결된 기본 동적 퍼블릭 IP 주소는 인스턴스를 중지하고 시작할 때마다 변경됩니다. 퍼블릭 IP 주소가 변경되지 않도록 고정 IP 주소를 생성하고 인스턴스에 연결하세요. 나중에 인스턴스에 도메인 이름을 사용하면 인스턴스를 중지하고 시작할 때마다 도메인의 DNS 레코드를 업데이트할 필요가 없습니다. 한 인스턴스에 한 개의 고정 IP를 연결할 수 있습니다.

  인스턴스에 연결된 기본 동적 퍼블릭 IP 주소는 인스턴스를 중지하고 시작할 때마다 변경됩니다. 퍼블릭 IP 주소가 변경되지 않도록 고정 IP 주소를 생성하고 인스턴스에 연결할 수 있습니다. 나중에 인스턴스에 도메인 이름을 사용하면 인스턴스를 중지하고 시작할 때마다 도메인의 DNS 레코드를 업데이트할 필요가 없습니다. 각 인스턴스에 1개의 고정 IP 주소만 연결할 수 있습니다.

인스턴스 관리 페이지의 네트워킹 탭에서 고정 IP 생성 또는 고정 IP 연결을 선택합니다. 인스턴스에 연결할 수 있는 고정 IP를 생성해 둔 경우에는 페이지의 지침을 따릅니다. 자세한 내용은 고정 IP를 생성하여 인스턴스에 연결을 참조하세요.



- Lightsail에 도메인 등록 Lightsail에서 도메인 이름을 등록하고 관리합니다. Lightsail은 가용성과 확장성이 우수한 도메인 이름 시스템(DNS) 웹 서비스인 Amazon Route 53을 사용하여 도메인을 등록합니다. 도메인이 등록되면 도메인을 Lightsail 리소스에 할당하거나 DNS 레코드를 관리할 수 있습니다. 자세한 내용은 Lightsail에서 웹 사이트의 도메인 등록 및 관리 단원을 참조하십시오.
- 인스턴스에 도메인 이름 매핑 - `example.com`과 같은 도메인 이름을 인스턴스에 매핑하려면 도메인의 도메인 이름 시스템(DNS)에 레코드를 추가합니다. DNS 레코드는 일반적으로 도메인을 등록한 등록 대행자가 관리 및 호스팅합니다. 그러나 Lightsail 콘솔을 사용하여 관리할 수 있도록 도메인의 DNS 레코드 관리를 Lightsail로 이전하는 것이 좋습니다.

  Lightsail 콘솔 홈 페이지의 도메인 및 DNS(Domains & DNS) 섹션에서 DNS 영역 생성(Create DNS zone)을 선택하고 페이지의 지침에 따릅니다. 자세한 내용은 DNS 영역을 생성하여 Lightsail 인스턴스의 도메인 레코드 관리 단원을 참조하십시오.

- 인스턴스에 스냅샷 생성 - 스냅샷은 시스템 디스크의 복사본이며, 인스턴스의 원본 구성입니다. 스냅샷에는 메모리, CPU, 디스크 크기, 데이터 전송 속도 등의 정보가 포함되어 있습니다. 스냅샷을 새 인스턴스의 기준 또는 데이터 벡업으로 사용할 수 있습니다.

  인스턴스 관리 페이지의 스냅샷 탭에 스냅샷의 이름을 입력한 다음 스냅샷 생성을 선택합니다. 자세한 내용은 스냅샷으로 Linux/Unix Lightsail 인스턴스 백업 섹션을 참조하세요.

CentOS에서 AlmaLinux로 마이그레이션하는 방법을 알아보려면 다음 주제인 Lightsail에서 CentOS에서 AlmaLinux로 데이터 마이그레이션(으)로 진행합니다.

Lightsail에서 CentOS에서 AlmaLinux로 데이터 마이그레이션

CentOS에서 AlmaLinux로 마이그레이션하는 것은 Lightsail의 한 인스턴스에서 다른 인스턴스로 데이터를 이동하는 간단한 프로세스입니다. 이 주제에서는 데이터를 마이그레이션하는 데 사용할 수 있는 두 가지 옵션을 간략하게 설명합니다.

자세한 내용은 AlmaLinux Wiki 사이트에서 AlmaLinux 설명서를 참조하세요.

목차

- 사전 조건
- (선택 사항) 보안 복사(scp)를 사용하여 인스턴스 간에 파일 전송
- (선택 사항) CentOS 인스턴스에서 AlmaLinux 인스턴스로 블록 스토리지 디스크 이동

사전 조건

- 아직 생성하지 않은 경우 AlmaLinux Lightsail 인스턴스를 생성합니다. 자세한 내용은 Lightsail에서 AlmaLinux 인스턴스 시작 및 설정 단원을 참조하십시오.
- AlmaLinux 인스턴스로 이동하려는 디스크의 스냅샷을 생성합니다. 자세한 내용은 백업 또는 기준에 대한 Lightsail 블록 스토리지 디스크 스냅샷 생성 섹션을 참조하세요.

(선택 사항) 보안 복사(scp)를 사용하여 인스턴스 간에 파일 전송

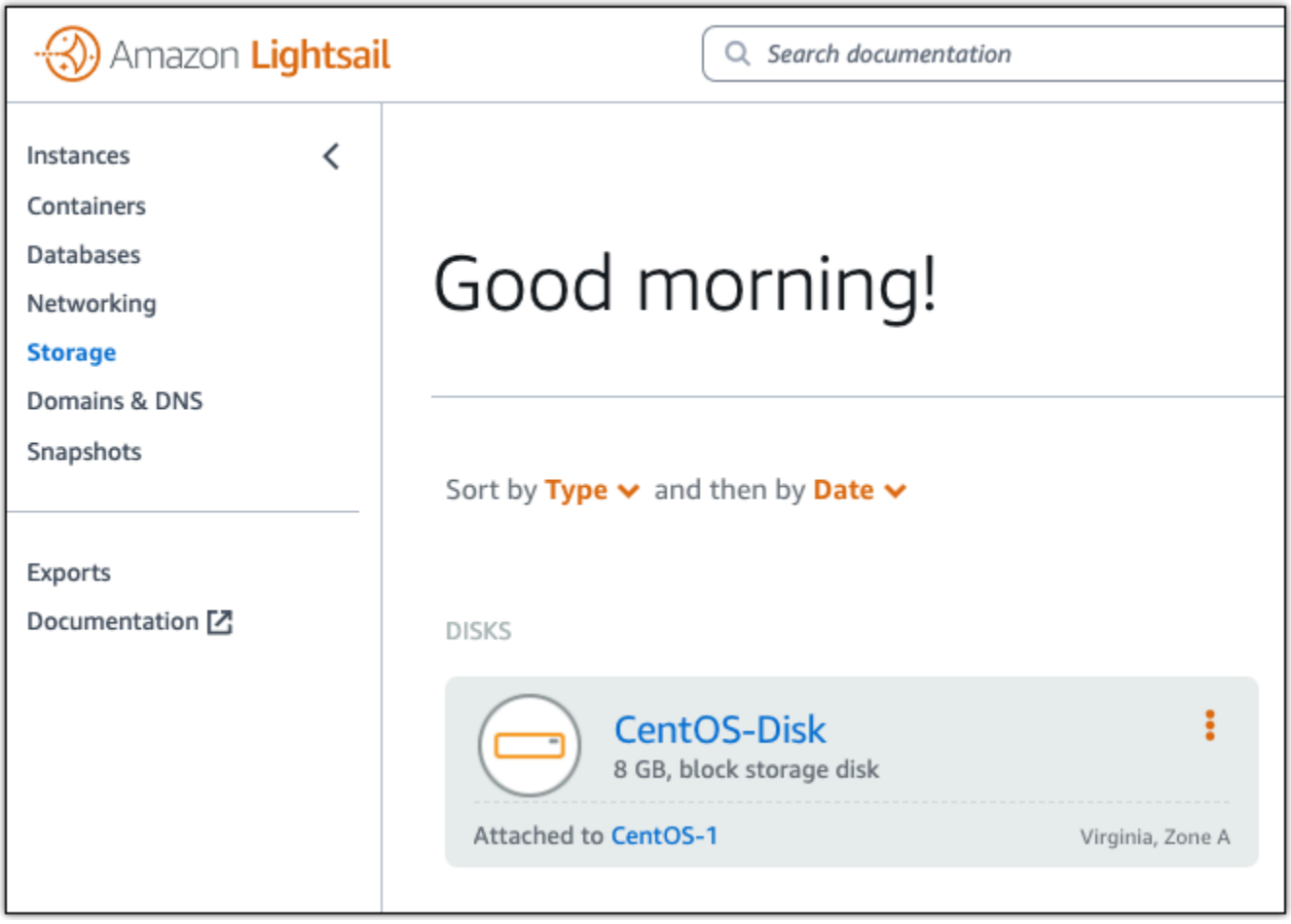
Linux의 보안 복사 명령을 사용하여 CentOS 인스턴스에서 새 AlmaLinux 인스턴스로 파일을 안전하게 전송할 수 있습니다. 자세한 내용은 Lightsail에서 scp를 사용하여 Linux 인스턴스 간에 파일 전송 섹션을 참조하세요.

(선택 사항) CentOS 인스턴스에서 AlmaLinux 인스턴스로 블록 스토리지 디스크 이동

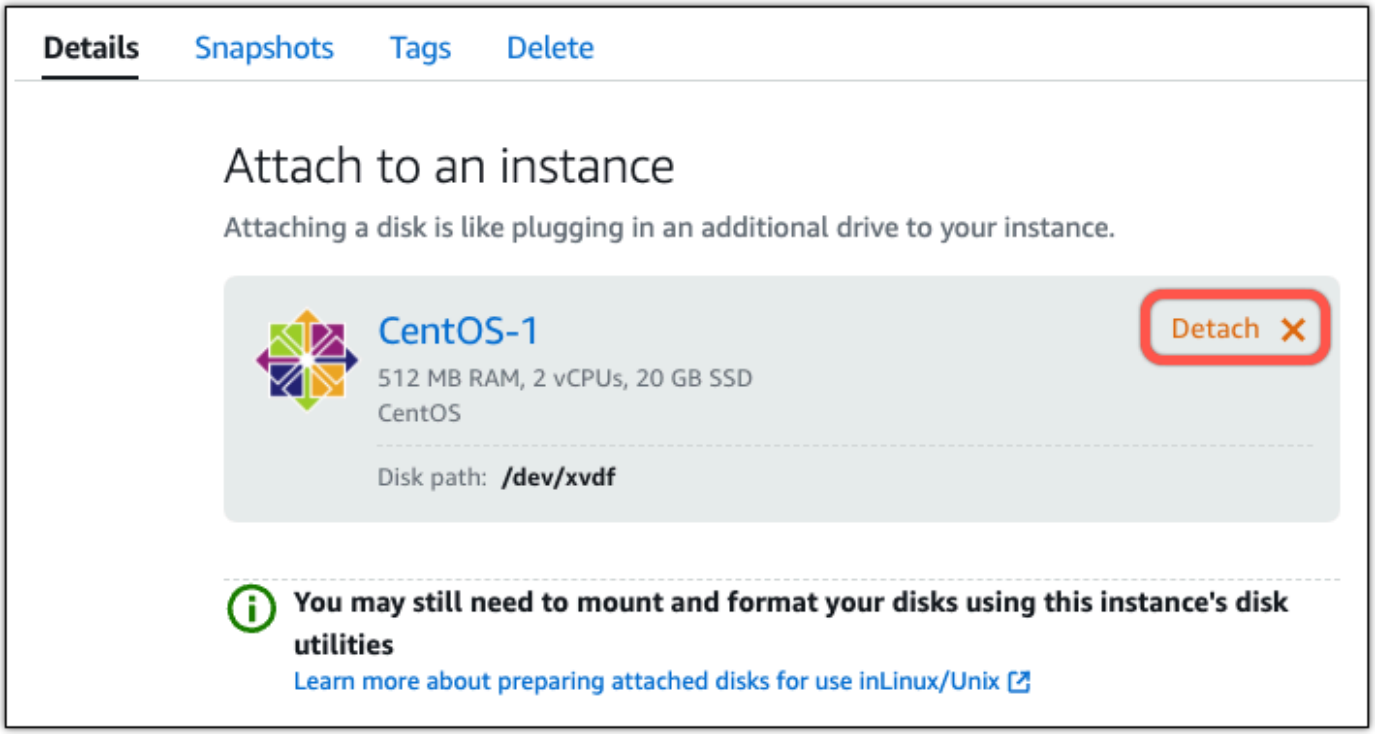
다음 절차에 따라 보조 블록 스토리지 디스크를 CentOS 인스턴스 번들에서 AlmaLinux 번들로 이동합니다. 인스턴스의 부팅 볼륨 디스크, 즉 운영 체제가 포함된 디스크는 분리할 수 없습니다. AlmaLinux 인스턴스에 디스크를 연결한 후에는 해당 인스턴스에 연결하고 디스크를 탑재해야 합니다. 자세한 내용은 Lightsail 블록 스토리지 디스크로 스토리지 및 성능 확장 섹션을 참조하세요.

CentOS 인스턴스가 실행 중인 경우 디스크를 분리하기 전에 중지해야 합니다. 자세한 내용은 실행 중인 인스턴스 중지를 참조하세요.

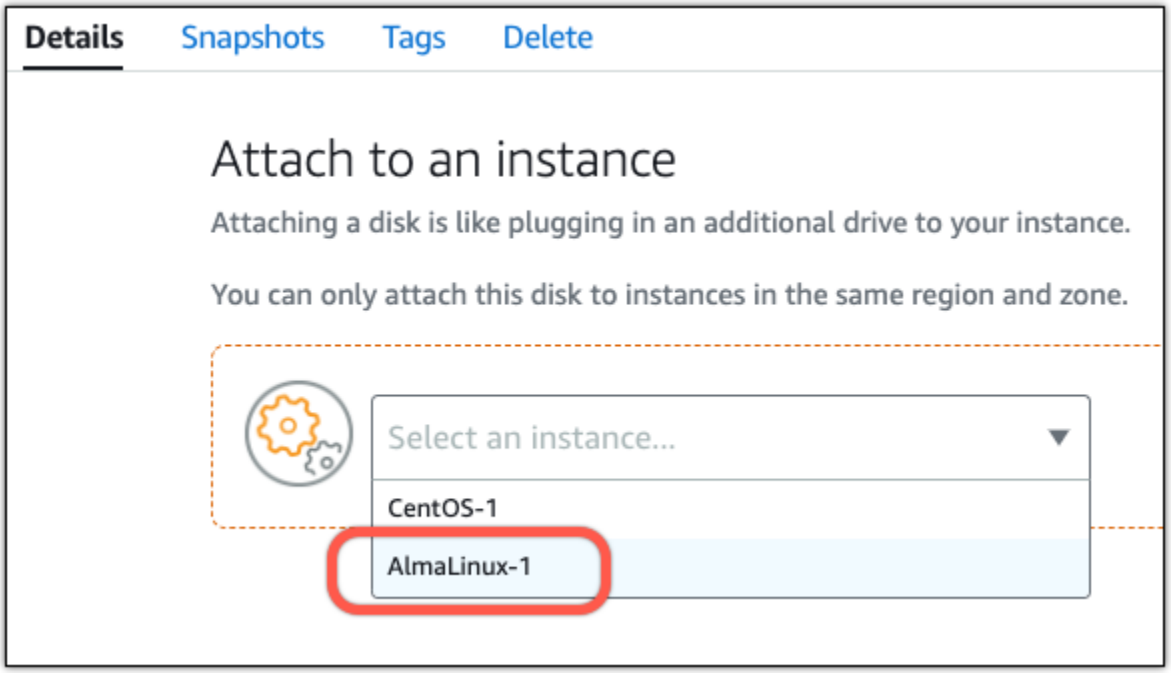
1. Lightsail 콘솔의 스토리지 섹션에서 CentOS 인스턴스에서 분리할 디스크를 선택합니다.



2. 세부 정보 탭에서 분리를 선택합니다.



3. 디스크 세부 정보 페이지에서 인스턴스에 연결 드롭다운 메뉴를 선택합니다. 그런 다음 AlmaLinux 인스턴스의 이름을 선택합니다.



4. 연결을 선택합니다.
5. (선택 사항) 데이터에 액세스하려면 먼저 AlmaLinux 인스턴스에 연결하고 디스크를 탑재해야 할 수 있습니다. 자세한 내용은 인스턴스에 연결하여 디스크를 포맷하고 탑재하기를 참조하세요.

> **Warning**
>
> 위 링크는 연결된 디스크를 탑재하고 포맷하는 방법에 대한 지침을 제공합니다. AlmaLinux 인스턴스에 연결한 디스크를 포맷하지 마세요. 포맷하면 디스크에 저장된 모든 정보가 영구적으로 삭제됩니다.

Lightsail에서 cPanel 및 WHM을 사용하여 웹 사이트, 이메일 및 서비스 호스팅

> **알고 계셨나요?**
>
> Lightsail는 일별 스냅샷 7개를 저장하고 인스턴스에 대해 자동 스냅샷을 활성화하면 가장 오래된 스냅샷을 최신 스냅샷으로 자동 대체합니다. 자세한 내용은 Lightsail 인스턴스 및 디스크의 자동 스냅샷 구성을 참조하세요.

다음은 Amazon Lightsail에 cPanel & WHM 인스턴스를 설치하고 실행한 후 시작하기 위해 수행해야 하는 단계입니다.

> **Important**
>
> - cPanel & WHM 인스턴스에는 15일 평가판 라이선스가 포함되어 있습니다. 15일이 지난 후에도 cPanel & WHM을 계속 사용하려면 cPanel에서 라이선스를 구매해야 합니다. 라이선스를 구매하려는 경우 라이선스를 구매하기 전에 이 가이드의 1~7단계를 완료하세요.
> - 이 블루프린트를 사용하려면 메모리가 2GB 이상인 인스턴스 플랜을 선택해야 합니다.

목차

- 1단계: 루트 사용자의 암호 변경
- 2단계: cPanel & WHM 인스턴스에 고정 IP 주소 연결
- 3단계: 웹 호스트 관리자에 처음으로 로그인
- 4단계: cPanel & WHM 인스턴스의 호스트 이름 및 IP 주소 변경
- 5단계: cPanel & WHM 인스턴스에 도메인 이름 매핑
- 6단계: 인스턴스의 방화벽 편집

- 7단계: Lightsail 인스턴스에서 SMTP 제한 제거
- 8단계: cPanel & WHM 문서 읽기 및 지원받기
- 9단계: cPanel & WHM 라이선스 구매
- 10단계: cPanel & WHM 인스턴스의 스냅샷 생성

1단계: 루트 사용자의 암호 변경

cPanel 인스턴스의 루트 사용자 암호를 변경하려면 다음 절차를 완료하세요. 나중에 웹 호스트 관리자(WHM) 콘솔에 로그인하는 데 루트 사용자와 암호를 사용하게 됩니다.

1. 인스턴스 관리 페이지의 연결(Connect) 탭에서 SSH로 연결(Connect using SSH)을 선택합니다.
2. 연결한 후 다음 명령을 입력하여 루트 사용자의 암호를 변경합니다.

```
sudo passwd
```

3. 강력한 암호를 입력한 후 한 번 더 입력하여 확인합니다.

> **Note**
>
> 암호는 사전 단어를 포함할 수 없으며, 7자 이상이어야 합니다. 이 지침을 따르지 않으면 `BAD PASSWORD` 경고가 표시됩니다.

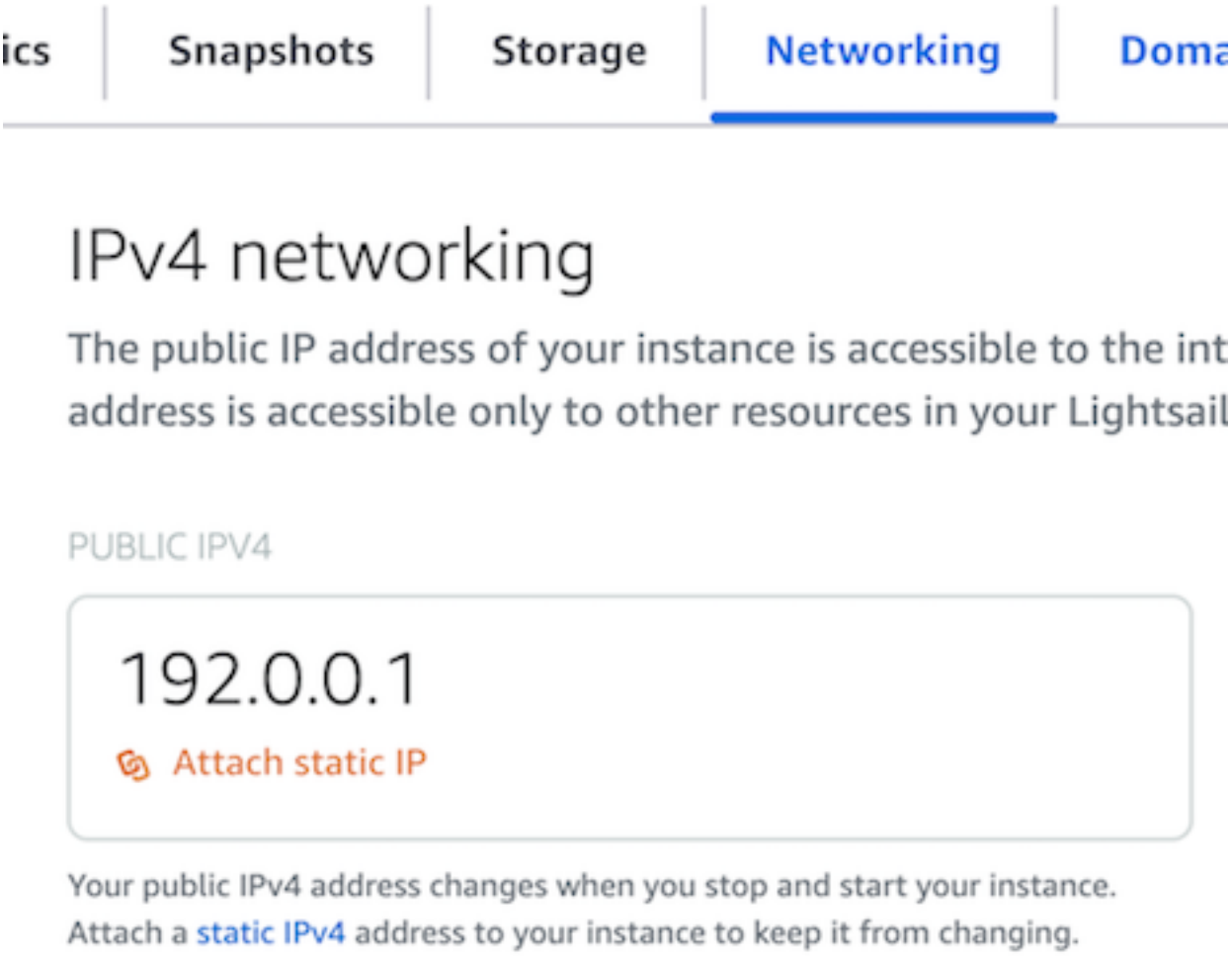
이 가이드의 후반부에서 WHM 콘솔에 로그인하는 데 암호를 사용하므로 기억해 두어야 합니다.

2단계: cPanel & WHM 인스턴스에 고정 IP 주소 연결

> **Important**
>
> cPanel에서 라이선스를 구매할 때는 cPanel & WHM 인스턴스의 퍼블릭 IP 주소를 지정해야 합니다. 구매한 라이선스가 해당 IP 주소에 연결됩니다. 따라서 cPanel에서 라이선스를 구매하려는 경우 cPanel & WHM 인스턴스에 고정 IP를 연결해야 합니다. cPanel에서 라이선스를 구매할 때 고정 IP를 지정하고, cPanel & WHM 라이선스를 Lightsail 인스턴스와 함께 사용하기로 계획한 기간 동안 고정 IP를 유지합니다. 나중에 다른 IP 주소로 라이선스를 전송해야 하는 경우 cPanel로 요청을 제출하면 됩니다. 자세한 내용은 WHM 문서의 라이선스 전송을 참조하세요.

인스턴스에 연결된 기본 동적 퍼블릭 IP 주소는 인스턴스를 중지하고 시작할 때마다 변경됩니다. 퍼블릭 IP 주소가 변경되지 않도록 고정 IP 주소를 생성하고 인스턴스에 연결할 수 있습니다. 나중에 인스턴스에 도메인 이름을 사용하면 인스턴스를 중지하고 시작할 때마다 도메인의 DNS 레코드를 업데이트할 필요가 없습니다. 각 인스턴스에 1개의 고정 IP 주소만 연결할 수 있습니다.

인스턴스 관리 페이지의 네트워킹 탭에서 고정 IP 생성 또는 고정 IP 연결을 선택합니다. 인스턴스에 연결할 수 있는 고정 IP를 생성해 둔 경우에는 페이지의 지침을 따릅니다. 자세한 내용은 고정 IP를 생성하여 인스턴스에 연결을 참조하세요.



3단계: 웹 호스트 관리자에 처음으로 로그인

WHM 콘솔에 처음으로 로그인하는 경우 다음 절차를 완료하세요.

1. 웹 브라우저를 열고 다음 웹 주소로 이동합니다. *<StaticIP>*를 인스턴스의 고정 IP 주소로 바꿉니다. 인스턴스에 연결을 설정할 포트인 `:2087`을 주소 끝에 추가해야 합니다.

   ```
   https://<StaticIP>:2087
   ```

예:

```
https://192.0.2.0:2087
```

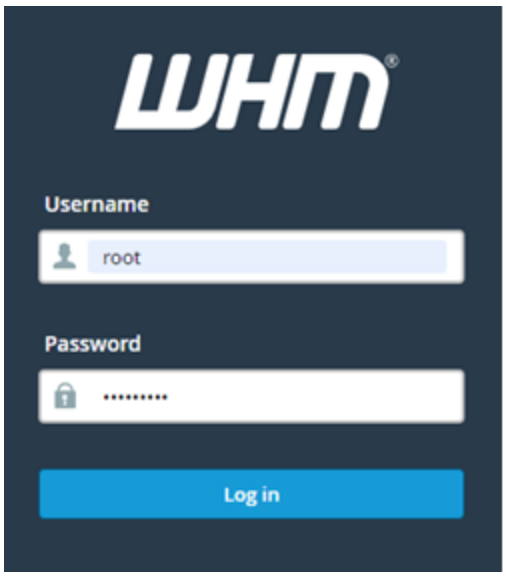
> **Important**
>
> 인스턴스의 IP 주소와 포트로 이동할 때 브라우저의 주소 표시줄에 `https://`를 포함해야 합니다. 그렇지 않으면 사이트에 연결할 수 없다는 오류가 발생합니다.

포트 2087을 통해 인스턴스의 고정 IP 주소를 탐색할 때 연결을 설정할 수 없는 경우, 라우터, VPN 또는 인터넷 서비스 제공업체가 포트 2087을 통한 HTTP/HTTPS 연결을 허용하는지 확인합니다. 허용하지 않는 경우 다른 네트워크를 사용하여 연결을 시도합니다.

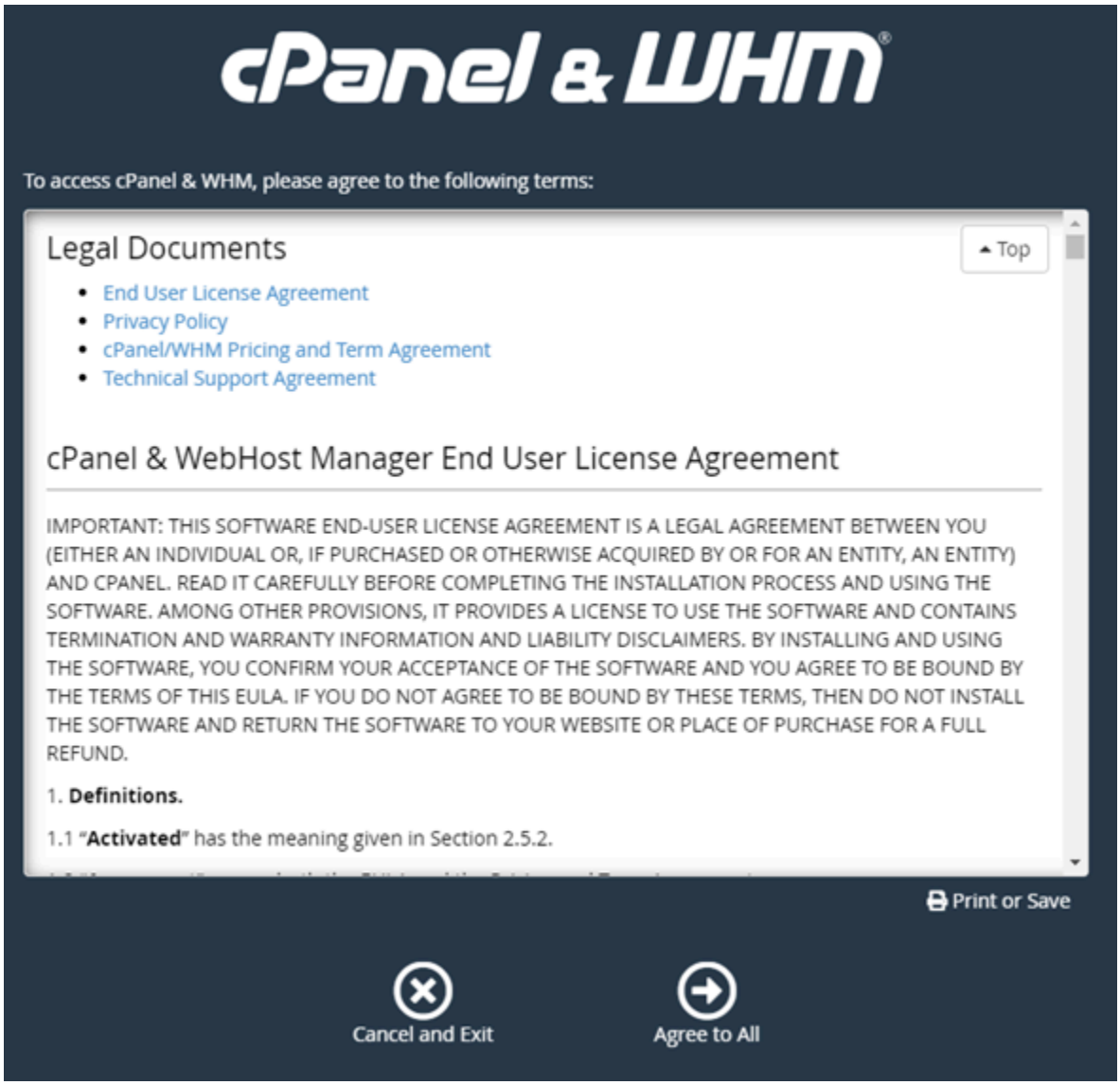
인터넷 연결이 비공개가 아니거나 안전하지 않거나 보안 위험이 있다는 브라우저 경고 메시지가 표시될 수도 있습니다. 이 경고는 cPanel 인스턴스에 아직 SSL/TLS 인증서가 적용되지 않았기 때문에 발생합니다. 브라우저 창에서 Advanced(고급), Details(세부 정보) 또는 More information(추가 정보)을 선택하여 사용할 수 있는 옵션을 표시합니다. 그런 다음 비공개가 아니거나 안전하지 않더라도 웹 사이트로 이동하도록 선택합니다.

2. 사용자 이름(Username) 텍스트 상자에 `root`를 입력합니다.
3. 암호(Password) 텍스트 상자에 루트 사용자 암호를 입력합니다.

   이 암호는 앞서 가이드의 1단계: 루트 사용자의 암호 변경 섹션에서 지정한 암호입니다.
4. 그런 다음 로그인을 선택합니다.

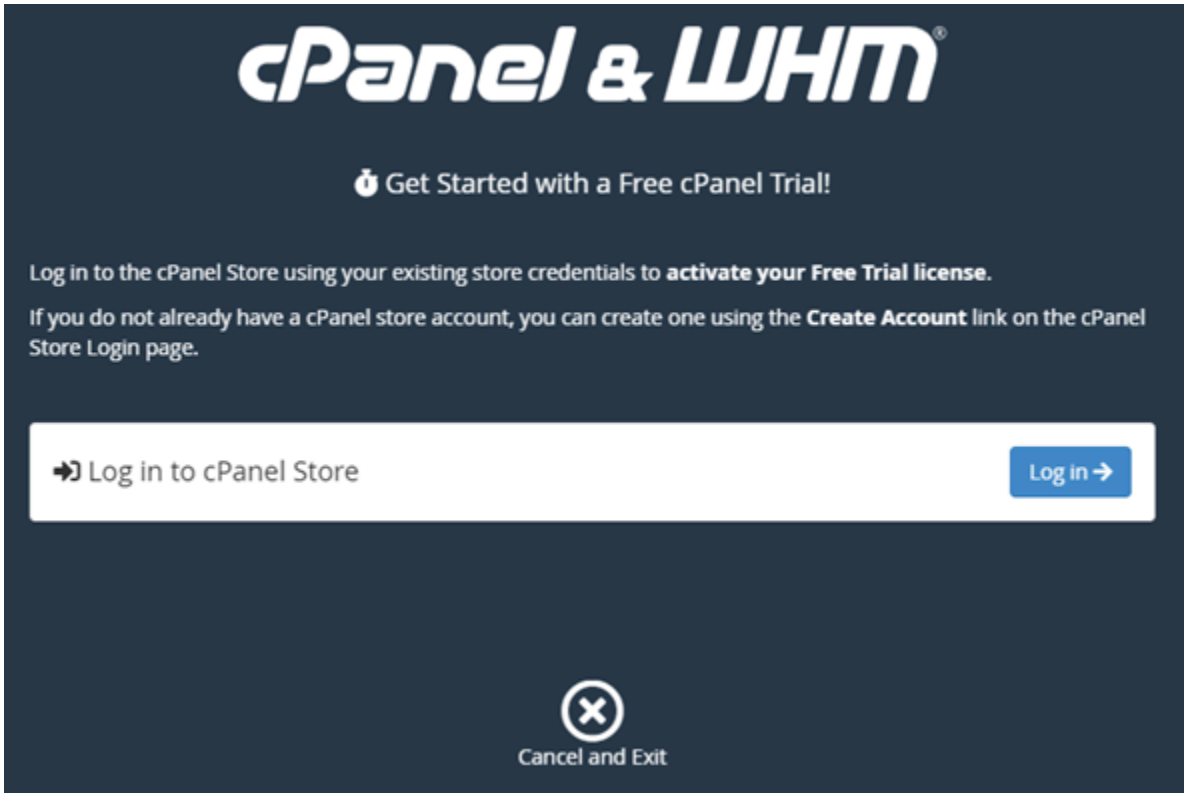


5. cPanel & WHM 약관을 읽은 다음 계속 진행하려면 모두 동의(Agree to all)를 선택합니다.



6. cPanel 무료 평가판 시작하기(Get started with a Free cPanel Trial) 페이지에서 로그인(Log in)을 선택하여 cPanel 스토어에 로그인합니다.

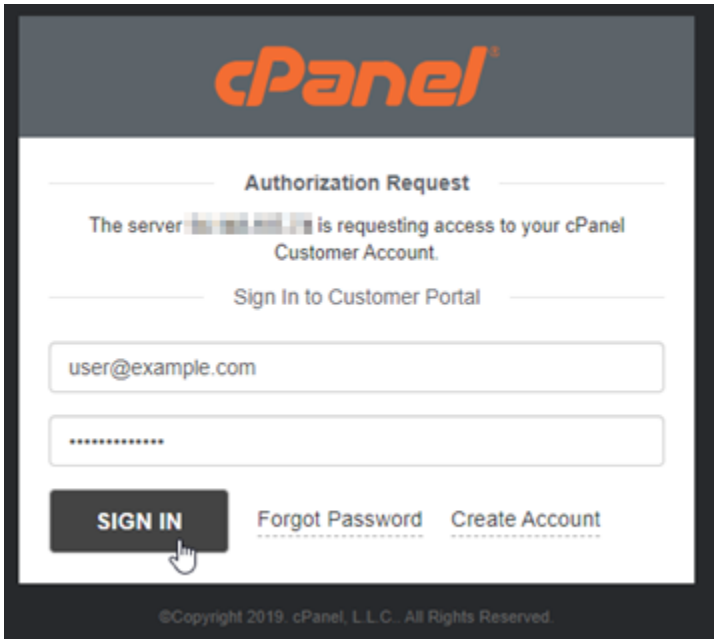
   평가판 라이선스를 계정에 연결하려면 cPanel 스토어에 로그인해야 합니다. cPanel 스토어 계정이 없는 경우에도 마찬가지로 로그인(Log in)을 선택하면 계정을 생성할 수 있는 옵션이 제공됩니다.



7. 권한 부여 요청(Authorization Request) 페이지가 표시되면 이메일 주소 또는 사용자 이름과 cPanel 스토어 계정의 암호를 입력합니다.

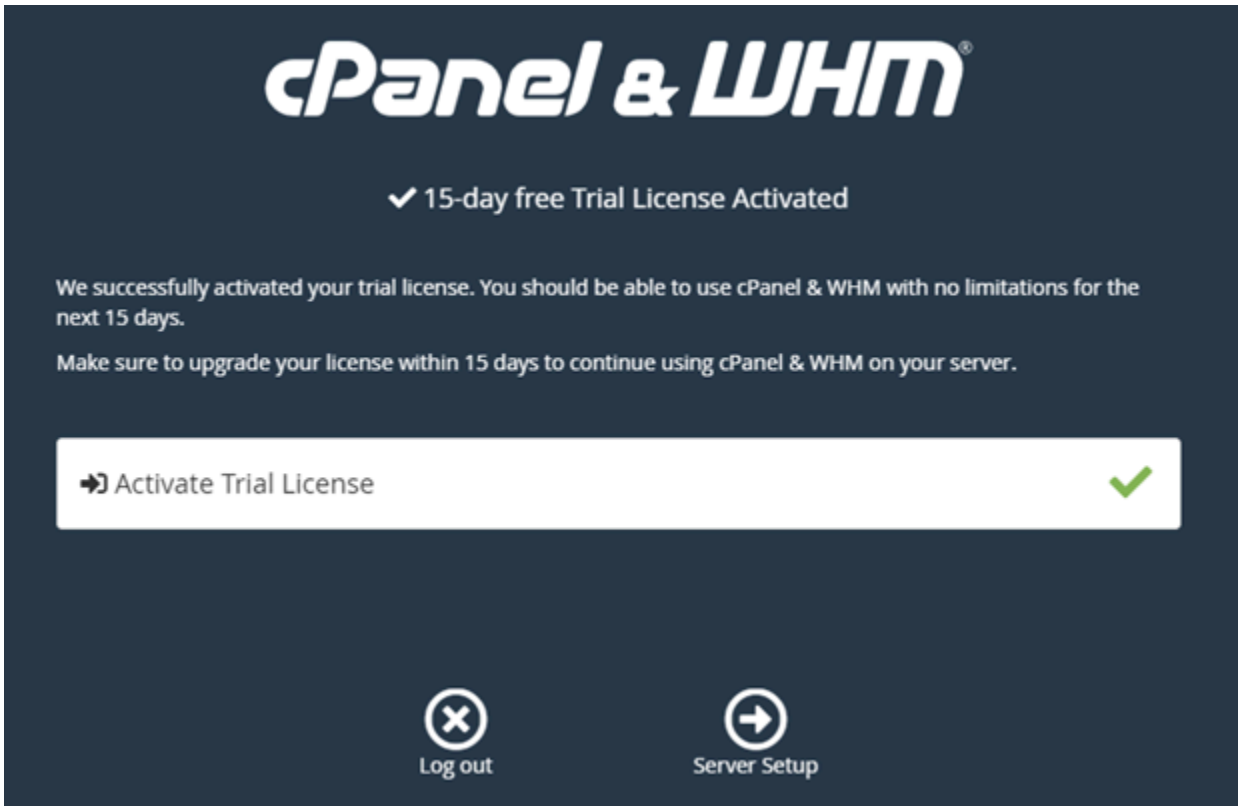
   cPanel 스토어 계정이 없는 경우 계정 생성(Create Account)을 선택하고 메시지에 따라 cPanel 스토어 계정을 새로 생성합니다. 이메일 주소를 입력하라는 메시지가 표시되고 cPanel 스토어 계정 암호를 설정할 수 있는 이메일을 받게 됩니다. 브라우저 탭을 새로 열어서 cPanel 스토어 계정 암호를 설정하는 것이 좋습니다. 암호가 설정되면 해당 탭을 닫고 인스턴스로 돌아가 계정에 권한을 부여한 후 이 절차의 다음 단계를 계속하면 됩니다.

8. 로그인을 선택합니다.

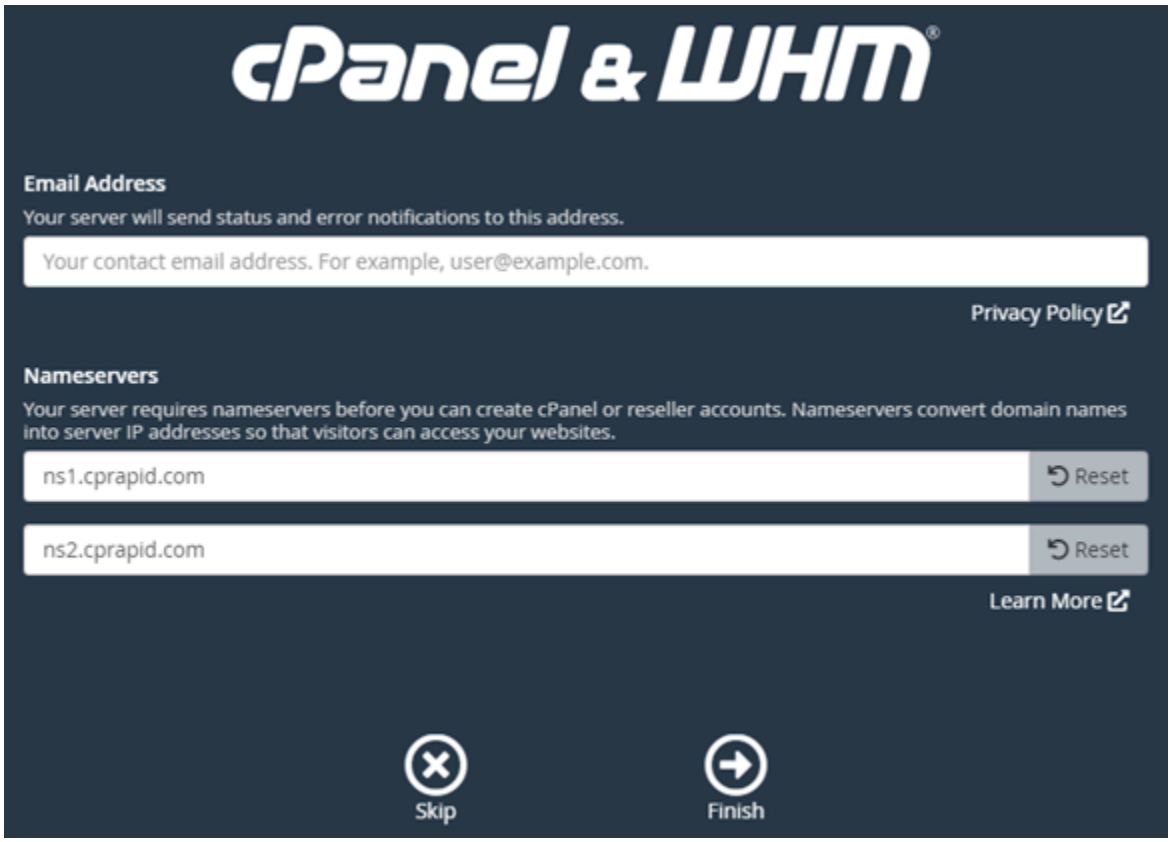


로그인한 후 cPanel 스토어 계정과 연결된 cPanel & WHM 인스턴스에 15일 평가판 라이선스가 제공됩니다. cPanel 스토어의 라이선스 관리(Manage Licenses) 페이지로 이동하여 평가판 라이선스 및 발급된 라이선스를 확인합니다.

9. 서버 설정(Server Setup)을 선택하여 계속 진행합니다.



10. 이메일 주소 및 네임서버 페이지에서 건너뛰기(Skip)를 선택합니다. 이 2가지는 나중에 구성할 수 있습니다.

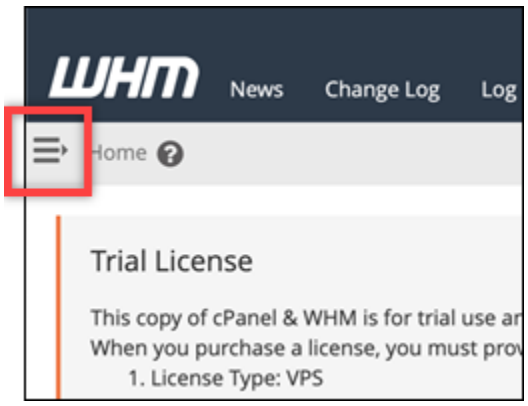


cPanel의 설정 및 기능을 관리할 수 있는 WHM 콘솔이 열립니다.

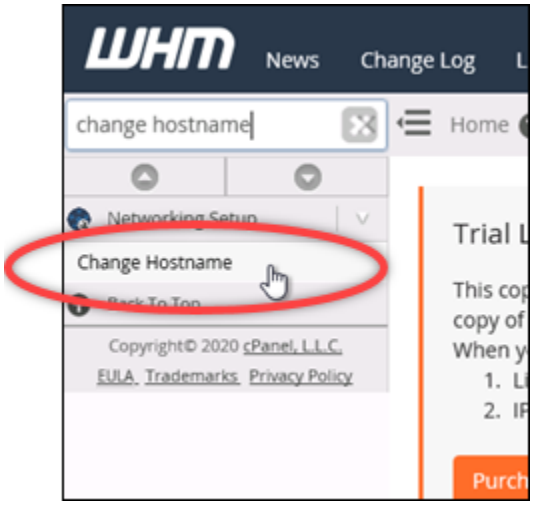
4단계: cPanel & WHM 인스턴스의 호스트 이름 및 IP 주소 변경

퍼블릭 IP 주소를 사용할 필요 없이 WHM 콘솔에 액세스할 수 있도록 인스턴스의 호스트 이름을 변경하려면 다음 단계를 완료하세요. 또한, 인스턴스의 IP 주소를 앞서 가이드의 2단계: cPanel & WHM 인스턴스에 고정 IP 주소 연결 섹션에서 인스턴스에 연결한 새로운 고정 IP 주소로 변경해야 합니다.

1. WHM 콘솔의 왼쪽 상단 섹션에서 탐색 메뉴 아이콘을 선택합니다.

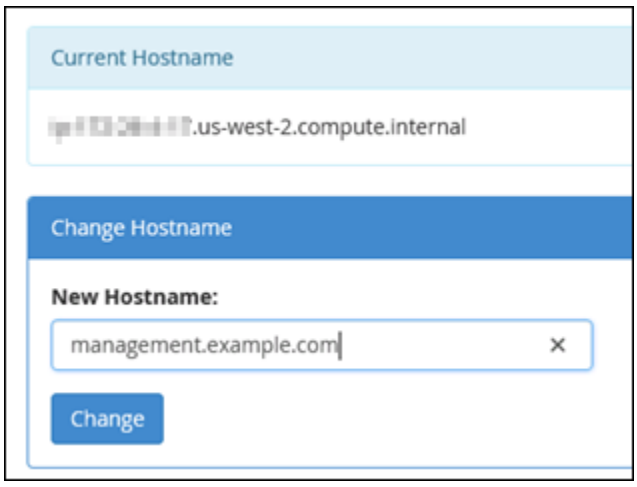


2. WHM 콘솔의 검색 텍스트 상자에 `Change hostname`을 입력한 다음 검색 결과에서 호스트 이름 변경(Change hostname)을 선택합니다.

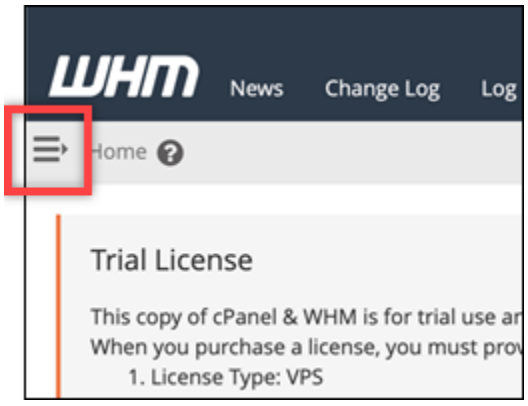


3. 새로운 호스트 이름(New hostname) 텍스트 상자에 WHM 콘솔에 액세스하는 데 사용하려는 호스트 이름을 입력합니다. 예를 들면, `management.example.com` 또는 `administration.example.com` 형식으로 입력합니다.

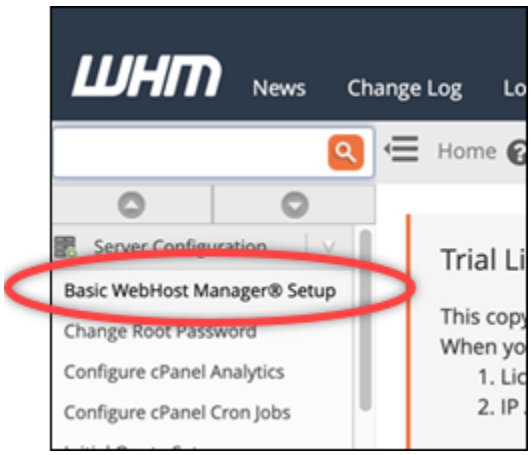
> **Note**
>
> 하위 도메인만 호스트 이름으로 지정할 수 있으며, `whm` 또는 `cpanel`을 하위 도메인으로 지정할 수 없습니다.



4. 변경을 선택합니다.
5. WHM 콘솔의 왼쪽 상단 섹션에서 탐색 메뉴 아이콘을 선택합니다.

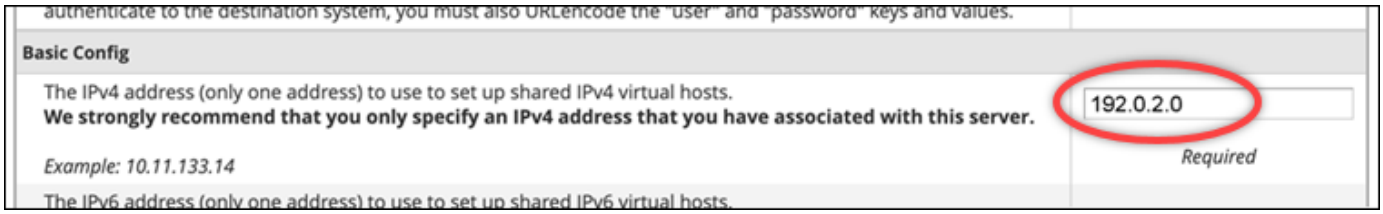


6. 기본 웹 호스트 관리자 설정(Basic WebHost Manager Setup)을 선택합니다.



7. 모두(All) 탭에서 아래로 스크롤하여 페이지의 기본 구성(Basic Config) 섹션을 찾습니다.

8. IPv4 주소 텍스트 상자에 인스턴스의 새로운 고정 IP 주소를 입력합니다. IPv6에 대한 자세한 내용은 cPanel 인스턴스에서 IPv6 구성을 참조하세요.



9. 페이지 맨 아래로 스크롤하고 변경 내용 저장(Save Changes)을 선택합니다.

> **Note**
>
> 잘못된 라이선스 파일(Invalid License file) 오류 메시지가 표시되면 잠시 기다린 후 IP 주소를 변경해 봅니다.

인스턴스의 호스트 이름과 IP 주소가 변경되었지만, 그래도 도메인 이름을 cPanel & WHM 인스턴스에 매핑해야 합니다. 이렇게 하려면 등록된 도메인 이름의 도메인 이름 시스템(DNS)에 주소(A) 레코드를 추가해야 합니다. A 레코드는 인스턴스의 호스트 이름을 인스턴스의 고정 IP 주소로 확인합니다. 이 가이드의 다음 섹션에서 이 작업을 수행하는 방법을 안내합니다.

5단계: cPanel & WHM 인스턴스에 도메인 이름 매핑

> **Note**
>
> 도메인을 cPanel & WHM 인스턴스에 매핑하여 WHM 콘솔에 액세스하는 데 사용할 수 있습니다. 또한, WHM 내에서 여러 도메인을 매핑하여 WHM에서 웹 사이트를 관리하는 데 사용할 수 있습니다. 이 섹션에서는 cPanel & WHM 인스턴스에 도메인을 매핑하는 방법을 설명합니다. 새로운 계정을 생성할 때 WHM 콘솔에서 여러 도메인을 매핑하는 방법에 대한 자세한 내용은 WHM 문서의 신규 계정 생성을 참조하세요.

`management.example.com` 또는 `administration.example.com`과 같은 도메인 이름을 인스턴스에 매핑하려면 도메인의 DNS에 주소(A) 레코드를 추가하면 됩니다. A 레코드는 cPanel & WHM 인스턴스의 호스트 이름을 인스턴스의 고정 IP 주소로 매핑합니다. A 레코드에 지정한 하위 도메인은 앞서 가이드의 4단계: cPanel & WHM 인스턴스의 호스트 이름 및 IP 주소 변경 섹션에서 지정한 호스트 이름과 일치해야 합니다. A 레코드를 추가한 후에는 인스턴스의 고정 IP 주소를 사용하는 대신 다음 주소로 인스턴스의 WHM 콘솔에 액세스할 수 있습니다. *<InstanceHostName>*을 인스턴스의 호스트 이름으로 바꿉니다.

```
https://<InstanceHostName>/whm
```

예:

```
https//management.example.com/whm
```

DNS 레코드는 일반적으로 도메인을 등록한 등록 대행자가 관리 및 호스팅합니다. 그러나 Lightsail 콘솔을 사용하여 관리할 수 있도록 도메인의 DNS 레코드 관리를 Lightsail로 이전하는 것이 좋습니다. 이렇게 하려면 Lightsail 콘솔에 로그인해야 합니다. Lightsail 콘솔 홈 페이지에서 도메인 및 DNS(Domains & DNS) 탭을 선택한 다음 DNS 영역 생성(Create DNS zone)을 선택합니다. 페이지의 지침에 따라 도메인 이름을 Lightsail에 추가합니다. 자세한 내용은 DNS 영역을 생성하여 Lightsail에서 도메인의 DNS 레코드 관리를 참조하세요.

6단계: 인스턴스의 방화벽 편집

다음 방화벽 포트는 cPanel & WHM 인스턴스에서 기본적으로 열립니다.

- SSH - TCP - 22
- DNS(UDP) - UDP - 53
- DNS(TCP) - TCP - 53
- HTTP - TCP - 80
- HTTPS - TCP - 443
- 사용자 지정 - TCP - 2078
- 사용자 지정 - TCP - 2083
- 사용자 지정 - TCP - 2087
- 사용자 지정 - TCP - 2089

인스턴스에서 사용하려는 서비스 및 애플리케이션에 따라 추가 포트를 열어야 할 수도 있습니다. 예를 들어 이메일 서비스의 경우 포트 25, 143, 465, 587, 993, 995, 2096을 열고 달력 서비스의 경우 포트 2080, 2091을 엽니다. 인스턴스 관리 페이지의 네트워킹(Networking) 탭에서 페이지의 방화벽(Firewall) 섹션으로 스크롤하고 규칙 추가(Add rule)를 선택합니다. 열려는 애플리케이션, 프로토콜 및 포트 또는 포트 범위를 선택합니다. 완료했으면 생성(Create)을 선택합니다.

열려는 포트 유형에 대한 자세한 내용은 cPanel 문서의 cPanel 서비스를 위한 방화벽을 구성하는 방법을 참조하세요. Lightsail에서 인스턴스의 방화벽을 편집하는 방법에 대한 자세한 내용은 Amazon Lightsail의 인스턴스 방화벽 규칙 추가 및 편집을 참조하세요.

7단계: Lightsail 인스턴스에서 SMTP 제한 제거

AWS 는 모든 Lightsail 인스턴스의 포트 25에서 아웃바운드 트래픽을 차단합니다. 포트 25에서 아웃바운드 트래픽을 전송하려면 이 제한을 제거하도록 요청하세요. 자세한 내용은 Lightsail 인스턴스에서 포트 25의 제한을 제거하려면 어떻게 하나요?를 참조하세요.

> **Important**
>
> 포트 25, 465 또는 587을 사용하도록 SMTP를 구성할 경우 Lightsail 콘솔의 인스턴스 방화벽에서 해당 포트를 열어야 합니다. 자세한 내용은 Amazon Lightsail의 방화벽 규칙 추가 및 편집을 참조하세요.

8단계: cPanel & WHM 문서 읽기 및 지원받기

cPanel & WHM 문서를 읽고 cPanel & WHM을 사용하여 웹 사이트를 관리하는 방법을 알아봅니다. 자세한 내용은 cPanel & WHM 문서를 참조하세요.

cPanel & WHM 관련 질문이 있거나 지원이 필요한 경우 다음 리소스를 사용하여 cPanel에 문의할 수 있습니다.

- cPanel 설치 문제 해결
- cPanel Discord 채널

9단계: cPanel & WHM 라이선스 구매

cPanel & WHM 인스턴스에는 15일 평가판 라이선스가 포함되어 있습니다. 15일이 지난 후에도 cPanel & WHM을 계속 사용하려면 cPanel에서 라이선스를 구매해야 합니다. 자세한 내용은 cPanel 문서의 cPanel 라이선스를 구매하는 방법을 참조하세요.

> **Important**
>
> cPanel에서 라이선스를 구매할 때는 cPanel & WHM 인스턴스의 퍼블릭 IP 주소를 지정해야 합니다. 구매한 라이선스가 해당 IP 주소에 연결됩니다. 이러한 이유로 가이드의 2단계: cPanel & WHM 인스턴스에 고정 IP 주소 연결 섹션에 나와 있는 대로 cPanel & WHM에 고정 IP를 연결해야 합니다. cPanel에서 라이선스를 구매할 때 고정 IP를 지정하고, cPanel & WHM 라이선스를 Lightsail 인스턴스와 함께 사용하기로 계획한 기간 동안 고정 IP를 유지합니다. 나중에 다

른 IP 주소로 라이선스를 전송해야 하는 경우 cPanel로 요청을 제출하면 됩니다. 자세한 내용은 WHM 문서의 라이선스 전송을 참조하세요.

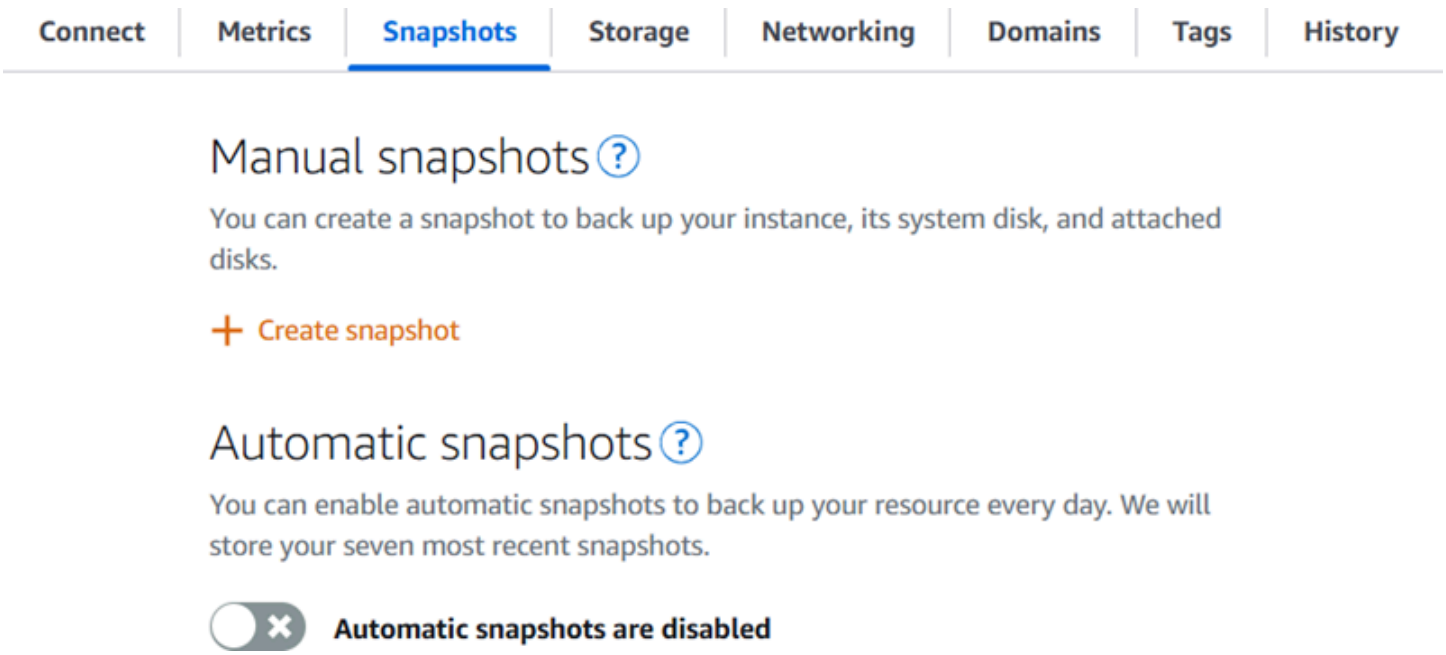
10단계: cPanel & WHM 인스턴스의 스냅샷 생성

> **Note**
>
> - 현재 세대 청사진 AlmaLinux용 cPanel & WHM의 인스턴스 스냅샷을 Amazon EC2로 내보낼 수 있습니다.
> - 이전 세대 청사진 Linux용 cPanel & WHM의 인스턴스 스냅샷을 현재 Amazon EC2로 내보낼 수 없습니다.
> - 스냅샷에서 새 인스턴스를 생성하는 경우 3단계에서 설명한 대로 WHM에 로그인하기 전에 인스턴스가 완전히 시작될 수 있도록 추가 시간을 줍니다.

원하는 방식으로 웹 사이트를 구성한 후 인스턴스의 주기적 스냅샷을 생성하여 백업합니다. 스냅샷은 시스템 디스크의 복사본이며, 인스턴스의 원본 구성입니다. 스냅샷은 스냅샷을 생성한 시점부터 인스턴스를 복원하는 데 필요한 모든 데이터를 포함합니다.

스냅샷을 수동으로 생성하거나 자동 스냅샷을 활성화하여 Lightsail에서 매일 스냅샷을 생성하도록 할 수 있습니다. 인스턴스에 문제가 있는 경우 스냅샷을 사용하여 새 교체 인스턴스를 생성할 수 있습니다.

스냅샷 탭의 인스턴스 관리 페이지에서 스냅샷 작업을 할 수 있습니다. 자세한 내용은 Amazon Lightsail의 스냅샷을 참조하세요.



Lightsail에서 Drupal 웹 사이트 설정 및 사용자 지정

> 알고 계셨나요?
>
> Lightsail는 일별 스냅샷 7개를 저장하고 인스턴스에 대해 자동 스냅샷을 활성화하면 가장 오래된 스냅샷을 최신 스냅샷으로 자동 대체합니다. 자세한 내용은 Lightsail 인스턴스 및 디스크의 자동 스냅샷 구성을 참조하세요.

다음은 Amazon Lightsail에 Drupal 인스턴스를 설치하고 실행한 후 시작해야 할 몇 가지 단계입니다.

목차

- 1단계: Bitnami 설명서 읽기
- 2단계: Drupal 관리 대시보드에 액세스하기 위한 기본 애플리케이션 암호 가져오기
- 3단계: 인스턴스에 고정 IP 주소 연결
- 4단계: Drupal 웹 사이트의 관리 대시보드에 로그인
- 5단계: 등록된 도메인 이름의 트래픽을 Drupal 웹 사이트로 라우팅
- 6단계: Drupal 웹 사이트에 대해 HTTPS 구성
- 7단계: Drupal 설명서 읽기 및 웹 사이트 구성 계속
- 8단계: 인스턴스의 스냅샷 생성

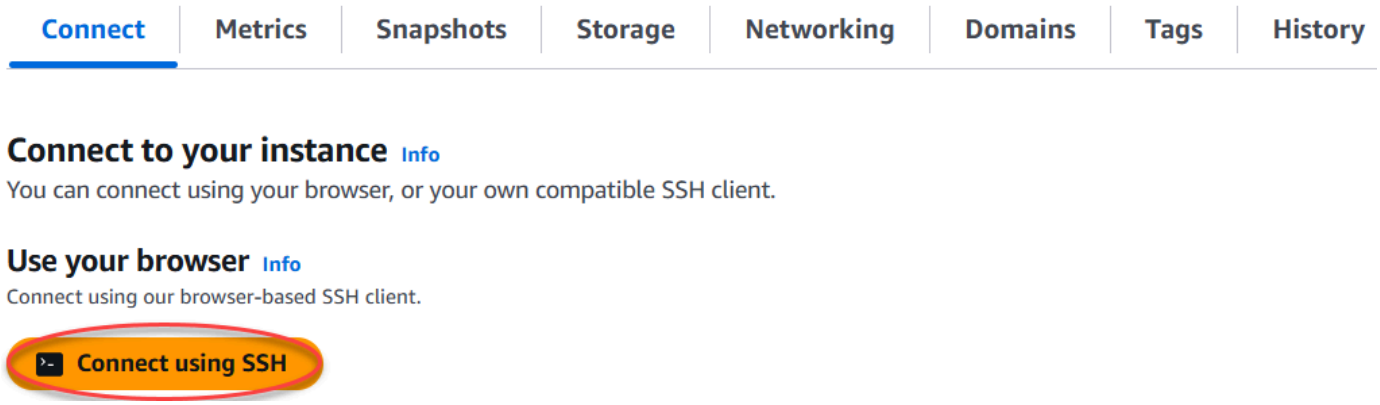
1단계: Bitnami 설명서 읽기

Bitnami 설명서를 읽고 Drupal 애플리케이션을 구성하는 방법을 알아보세요. 자세한 내용은 AWS 클라우드용 Bitnami에서 패키징한 Drupal을 참조하세요.

2단계: Drupal 관리 대시보드에 액세스하기 위한 기본 애플리케이션 암호 가져오기

Drupal 웹 사이트의 관리 대시보드에 액세스하는 데 필요한 기본 애플리케이션 암호를 가져오려면 다음 절차를 완료하세요. 자세한 내용은 Amazon Lightsail의 Bitnami 인스턴스에 대한 애플리케이션 사용자 이름과 암호 가져오기를 참조하세요.

1. 인스턴스 관리 페이지의 연결(Connect) 탭에서 SSH로 연결(Connect using SSH)을 선택합니다.



2. 연결한 후 다음 명령을 입력하여 애플리케이션 암호를 가져옵니다.

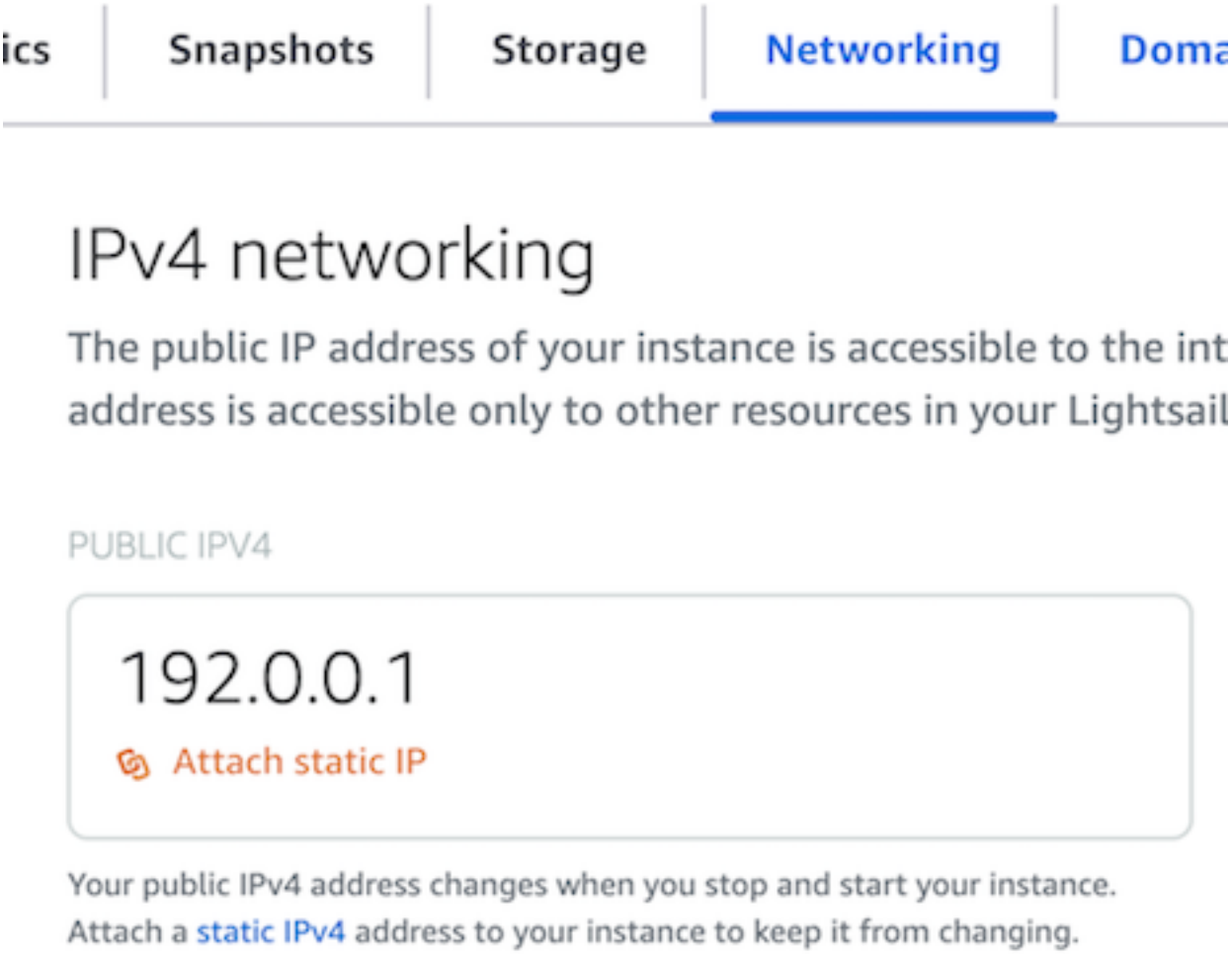
```
cat $HOME/bitnami_application_password
```

기본 애플리케이션 암호가 포함된 다음 예제와 유사한 응답이 표시되어야 합니다.

3단계: 인스턴스에 고정 IP 주소 연결

인스턴스에 연결된 기본 동적 퍼블릭 IP 주소는 인스턴스를 중지하고 시작할 때마다 변경됩니다. 퍼블릭 IP 주소가 변경되지 않도록 고정 IP 주소를 생성하고 인스턴스에 연결할 수 있습니다. 나중에 인스턴스에 도메인 이름을 사용하면 인스턴스를 중지하고 시작할 때마다 도메인의 DNS 레코드를 업데이트할 필요가 없습니다. 각 인스턴스에 1개의 고정 IP 주소만 연결할 수 있습니다.

인스턴스 관리 페이지의 네트워킹 탭에서 고정 IP 생성 또는 고정 IP 연결을 선택합니다. 인스턴스에 연결할 수 있는 고정 IP를 생성해 둔 경우에는 페이지의 지침을 따릅니다. 자세한 내용은 고정 IP를 생성하여 인스턴스에 연결을 참조하세요.



4단계: Drupal 웹 사이트의 관리 대시보드에 로그인

이제 기본 사용자 암호가 있으므로 Drupal 웹 사이트의 홈 페이지로 이동하여 관리 대시보드에 로그인합니다. 로그인한 후 웹 사이트 사용자 지정 및 관리 변경을 시작할 수 있습니다. Drupal에서 수행할 수 있는 작업에 대한 자세한 내용을 알아보려면 이 가이드 뒷부분의 7단계: Drupal 설명서 읽기 및 웹 사이트 구성 계속 섹션을 참조하세요.

1. 인스턴스 관리 페이지의 연결(Connect) 탭에서 인스턴스의 퍼블릭 IP 주소를 기록해 둡니다. 퍼블릭 IP 주소는 인스턴스 관리 페이지의 헤더 섹션에도 표시됩니다.

2. 인스턴스의 퍼블릭 IP 주소로 이동합니다(예: `http://203.0.113.0`으로 이동).

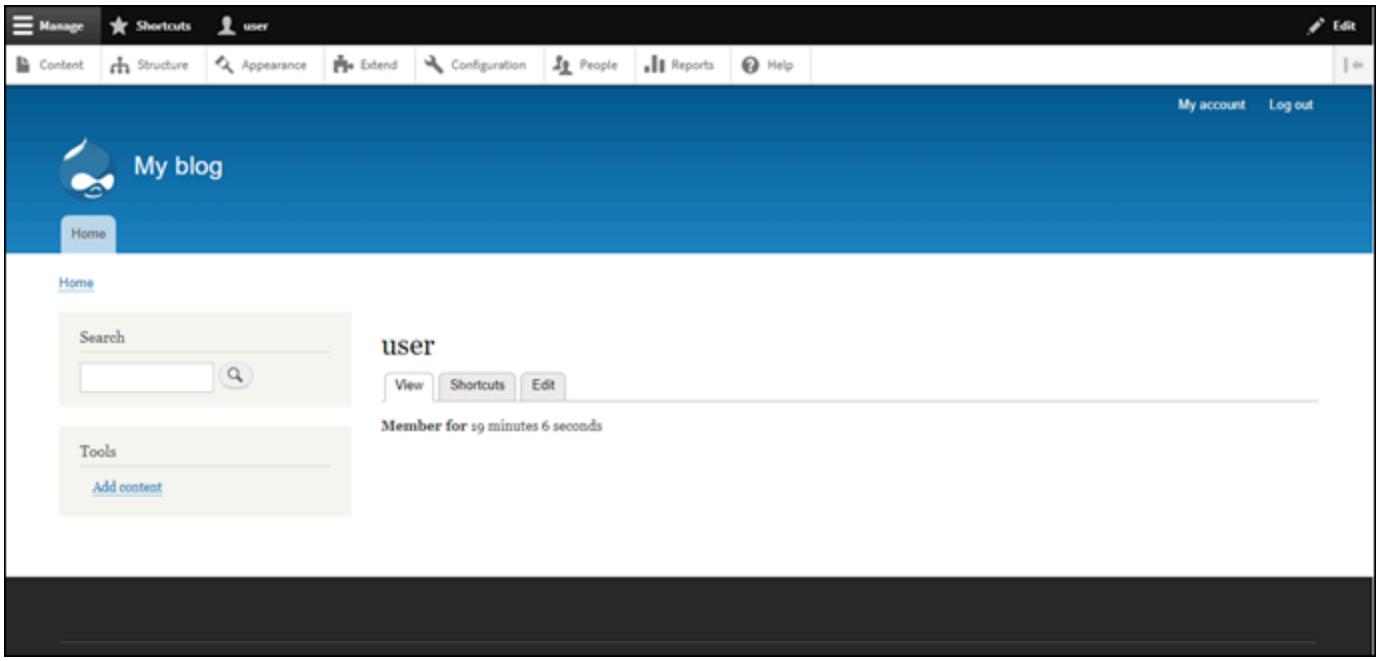
   Drupal 웹 사이트의 홈 페이지가 나타납니다.

3. Drupal 웹 사이트 홈 페이지의 오른쪽 하단 모서리에 있는 관리(Manage)를 선택합니다.

   관리(Manage) 배너가 표시되지 않는 경우 `http://`*`<PublicIP>`*`/user/login`을 통해 로그인 페이지로 이동할 수 있습니다. *`<PublicIP>`*을(를) 인스턴스의 퍼블릭 IP 주소로 바꿉니다.

4. 이 가이드의 초반부에서 검색한 기본 사용자 이름(`user`) 및 기본 암호를 사용하여 로그인합니다.

   Drupal 관리 대시보드가 나타납니다.



5단계: Drupal 웹 사이트로 등록된 도메인 이름의 트래픽 라우팅

등록된 도메인 이름(예: `example.com`)의 트래픽을 Drupal 웹 사이트로 라우팅하려면 도메인의 도메인 이름 시스템(DNS)에 레코드를 추가하면 됩니다. DNS 레코드는 일반적으로 도메인을 등록한 등록 대행자가 관리 및 호스팅합니다. 그러나 Lightsail 콘솔을 사용하여 관리할 수 있도록 도메인의 DNS 레코드 관리를 Lightsail로 이전하는 것이 좋습니다.

Lightsail 콘솔 홈 페이지의 도메인 및 DNS(Domains & DNS) 탭에서 DNS 영역 생성(Create DNS zone)을 선택하고 페이지의 지침에 따릅니다. 자세한 내용은 DNS 영역을 생성하여 Lightsail에서 도메인의 DNS 레코드 관리를 참조하세요.

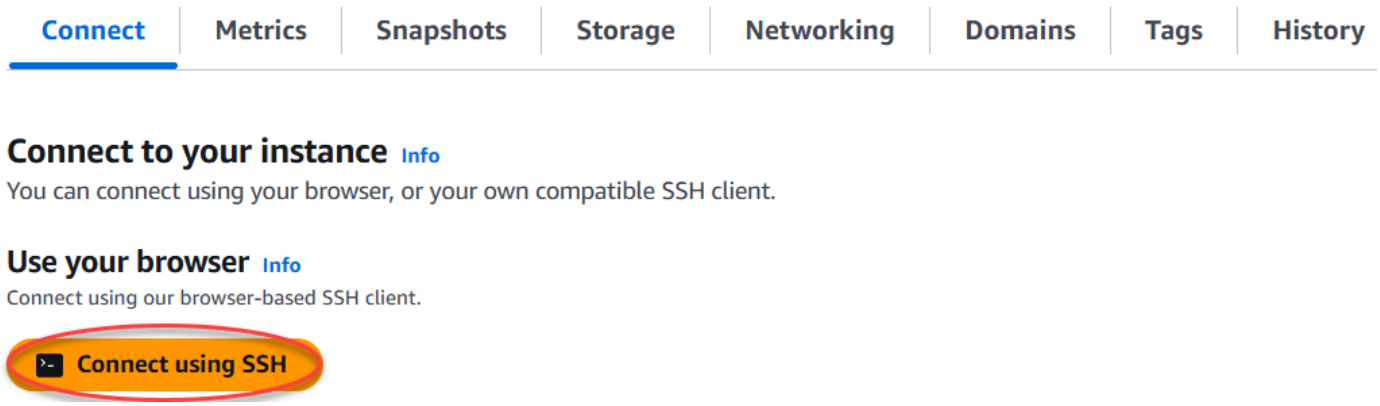
인스턴스에 대해 구성한 도메인 이름으로 이동하면 Drupal 웹 사이트의 홈 페이지로 리디렉션됩니다. 다음으로 Drupal 웹 사이트에 대한 HTTPS 연결을 활성화하기 위해 SSL/TLS 인증서를 생성하고 구성해야 합니다. 자세한 내용을 보려면 이 가이드의 다음 6단계: Drupal 웹 사이트에 대해 HTTPS 구성 섹션으로 계속하세요.

6단계: Drupal 웹 사이트에 대해 HTTPS 구성

Drupal 웹 사이트에서 HTTPS를 구성하려면 다음 절차를 완료하세요. 이 단계에서는 Let's Encrypt SSL/TLS 인증서를 요청하기 위한 명령줄 도구인 Bitnami HTTPS Configuration Tool(`bncert-tool`)을 사용하는 방법을 보여줍니다. 자세한 내용을 알아보려면 Bitnami 설명서의 Learn About The Bitnami HTTPS Configuration Tool(Bitnami Configuration Tool에 대해 알아보기)을 참조하세요.

> **Important**
>
> 이 절차를 시작하기 전에 Drupal 인스턴스로 트래픽을 라우팅하도록 도메인을 구성했는지 확인합니다. 그렇지 않으면 SSL/TLS 인증서 검증 프로세스가 실패합니다.

1. 인스턴스 관리 페이지의 연결(Connect) 탭에서 SSH로 연결(Connect using SSH)을 선택합니다.



2. 연결한 후 다음 명령을 입력하여 bncert 도구가 인스턴스에 설치되었는지 확인합니다.

```
sudo /opt/bitnami/bncert-tool
```

다음과 같은 응답 중 하나가 표시됩니다.

- 응답에 명령을 찾을 수 없음(command not found)이 표시되면 bncert 도구가 인스턴스에 설치되지 않은 것입니다. 이 절차의 다음 단계를 계속 진행하여 인스턴스에 bncert 도구를 설치합니다.
- 응답에 Bitnami HTTPS 구성 도구 시작(Welcome to the Bitnami HTTPS configuration tool)이 표시되면 bncert 도구가 인스턴스에 설치된 것입니다. 이 절차의 8단계로 계속합니다.
- bncert 도구가 일시적으로 인스턴스에 설치된 경우 업데이트된 버전의 도구를 사용할 수 있다는 메시지가 표시될 수 있습니다. 다운로드하도록 선택하고 `sudo /opt/bitnami/bncert-tool` 명령을 입력하여 bncert 도구를 다시 실행합니다. 이 절차의 8단계로 계속합니다.

3. 다음 명령을 입력하여 bncert 실행 파일을 인스턴스로 다운로드합니다.

```
wget -O bncert-linux-x64.run https://downloads.bitnami.com/files/bncert/latest/
bncert-linux-x64.run
```

4. 다음 명령을 입력하여 인스턴스에서 bncert 도구 실행 파일의 디렉터리를 생성합니다.

```
sudo mkdir /opt/bitnami/bncert
```

5. 다음 명령을 입력하여 bncert에서 프로그램으로 실행할 수 있는 파일을 실행하도록 합니다.

```
sudo chmod +x /opt/bitnami/bncert/bncert-linux-x64.run
```

6. 다음 명령을 입력하여 sudo /opt/bitnami/bncert-tool 명령을 입력할 때 bncert 도구를 실행하는 심볼 링크를 생성합니다.

```
sudo ln -s /opt/bitnami/bncert/bncert-linux-x64.run /opt/bitnami/bncert-tool
```

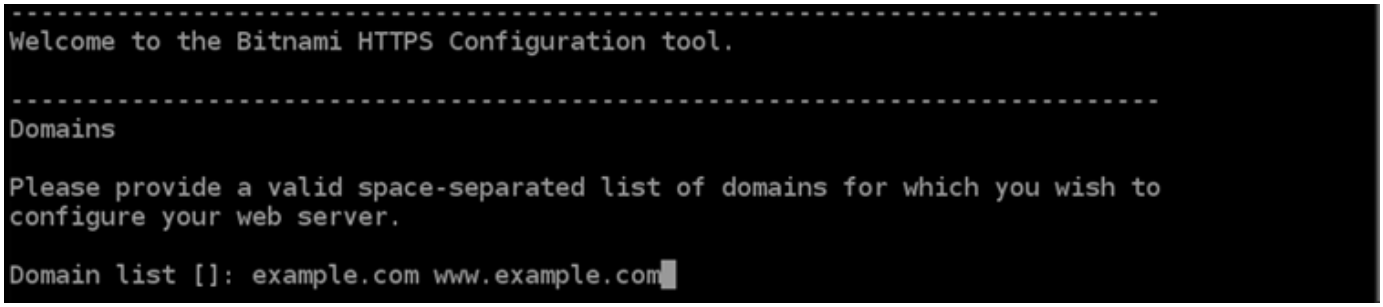
이제 인스턴스에 bncert 도구를 설치했습니다.

7. 다음 명령을 입력하여 bncert 도구를 실행합니다.

```
sudo /opt/bitnami/bncert-tool
```

8. 다음 예와 같이 기본 도메인 이름과 대체 도메인 이름을 공백으로 구분하여 입력합니다.

도메인이 트래픽을 인스턴스의 퍼블릭 IP 주소로 라우팅하도록 구성되지 않은 경우, `bncert` 도구에서 계속하기 전에 해당 구성을 설정하라는 메시지를 표시합니다. 도메인은 `bncert` 도구를 사용하여 인스턴스에서 HTTPS를 활성화한 인스턴스의 퍼블릭 IP 주소로 트래픽을 라우팅해야 합니다. 이렇게 해야 도메인을 소유하고 있음을 확인하고 인증서를 검증하는 역할을 할 수 있습니다.



9. bncert 도구는 웹 사이트의 리디렉션을 어떻게 구성할지 묻는 메시지를 표시합니다. 사용할 수 있는 옵션은 다음과 같습니다.

   - HTTP에서 HTTPS로 리디렉션 활성화(Enable HTTP to HTTPS redirection) - HTTP 버전 웹 사이트(예: http:/example.com)를 방문하는 사용자를 HTTPS 버전(예: https://example.com)으로 자동 리디렉션할지 지정합니다. 모든 방문자가 암호화된 연결을 사용하도록 강제하기 때문에 이 옵션을 활성화하는 것이 좋습니다. Y를 입력하고 Enter 키를 눌러 활성화합니다.
   - 비 www에서 www로 리디렉션 활성화(Enable non-www to www redirection) - 도메인의 정점(https://example.com)을 방문하는 사용자를 도메인의 www 하위 도메인(https://www.example.com)으로 자동 리디렉션할지 지정합니다. 이 옵션은 활성화하는 것이 좋습니다. 그러나 Google의 웹 마스터 도구와 같은 검색 엔진 도구에서 도메인의 정점을 기본 웹 사이트 주소로 지정했거나 정점이 IP를 직접 가리키고 www 하위 도메인이 CNAME 레코드를 통해 정점을 참조하는 경우, 이 옵션을 비활성화하고 대체 옵션(www에서 비 www로 리디렉션 활성화)을 활성화할 수 있습니다. Y를 입력하고 Enter 키를 눌러 활성화합니다.
   - www에서 비 www로 리디렉션 활성화(Enable www to non-www redirection) - 도메인의 www 하위 도메인(https://www.example.com)을 방문하는 사용자를 도메인의 정점(https://example.com)으로 자동 리디렉션할지 지정합니다. 비 www에서 www로 리디렉션을 활성화한 경우 이 옵션을 비활성화하는 것이 좋습니다. N를 입력하고 Enter 키를 눌러 비활성화합니다.

   선택한 내용은 다음 예와 같아야 합니다.

```
Enable/disable redirections

Please select the redirections you wish to enable or disable on your Bitnami
installation.



Enable HTTP to HTTPS redirection [Y/n]: Y



Enable non-www to www redirection [Y/n]: Y



Enable www to non-www redirection [y/N]: N
```

10. 적용될 변경 사항의 목록이 나열됩니다. Y를 입력하고 Enter 키를 눌러 확인하고 계속합니다.

```
Changes to perform

The following changes will be performed to your Bitnami installation:

1. Stop web server
2. Configure web server to use a free Let's Encrypt certificate for the domains:
example.com www.example.com
3. Configure a cron job to automatically renew the certificate each month
4. Configure web server name to: example.com
5. Enable HTTP to HTTPS redirection (example: redirect http://example.com to
https://example.com)
6. Enable non-www to www redirection (example: redirect example.com to
 www.example.com)
7. Start web server once all changes have been performed



Do you agree to these changes? [Y/n]: Y
```

11. Let's Encrypt 인증서에 연결할 이메일 주소를 입력하고 Enter 키를 누릅니다.

```
Create a free HTTPS certificate with Let's Encrypt

Please provide a valid e-mail address for which to associate your Let's Encrypt
certificate.

Domain list: example.com www.example.com

Server name: example.com

E-mail address []:
```

12. Let's Encrypt 구독자 계약을 검토합니다. Y를 입력하고 Enter 키를 눌러 계약을 수락하고 계속합니다.

```
The Let's Encrypt Subscriber Agreement can be found at:

https://letsencrypt.org/documents/LE-SA-v1.2-November-15-2017.pdf

Do you agree to the Let's Encrypt Subscriber Agreement? [Y/n]:
```

인증서 요청 및 지정한 리디렉션 구성을 비롯하여 인스턴스에서 HTTPS를 활성화하는 작업이 수행됩니다.

```
Performing changes to your installation

The Bitnami HTTPS Configuration Tool will perform any necessary actions to your
Bitnami installation. This may take some time, please be patient.
```

인증서가 성공적으로 발급되고 검증됩니다. 다음 예와 유사한 메시지가 표시되면 인스턴스에서 리디렉션이 성공적으로 구성된 것입니다.

```
Success

The Bitnami HTTPS Configuration Tool succeeded in modifying your installation.

The configuration report is shown below.

Backup files:
* /opt/bitnami/apache2/conf/httpd.conf.back.202005290035
* /opt/bitnami/apache2/conf/bitnami/bitnami-apps-prefix.conf.back.202005290035
* /opt/bitnami/apache2/conf/bitnami/bitnami.conf.back.202005290035

Find more details in the log file:

/tmp/bncert-202005290035.log

If you find any issues, please check Bitnami Support forums at:

https://community.bitnami.com

Press [Enter] to continue:
```

`bncert` 도구는 인증서가 만료되기 80일 전마다 인증서를 자동으로 갱신합니다. 인스턴스와 함께 추가 도메인 및 하위 도메인을 사용하고 이러한 도메인에서 HTTPS를 활성화하려면 위의 단계를 반복합니다.

이제 Drupal 인스턴스에서 HTTPS가 활성화되었습니다. 다음에 구성한 도메인을 사용하여 Drupal 웹 사이트로 이동하면 HTTPS 연결로 리디렉션됩니다.

7단계: Drupal 설명서 읽기 및 웹 사이트 구성 계속

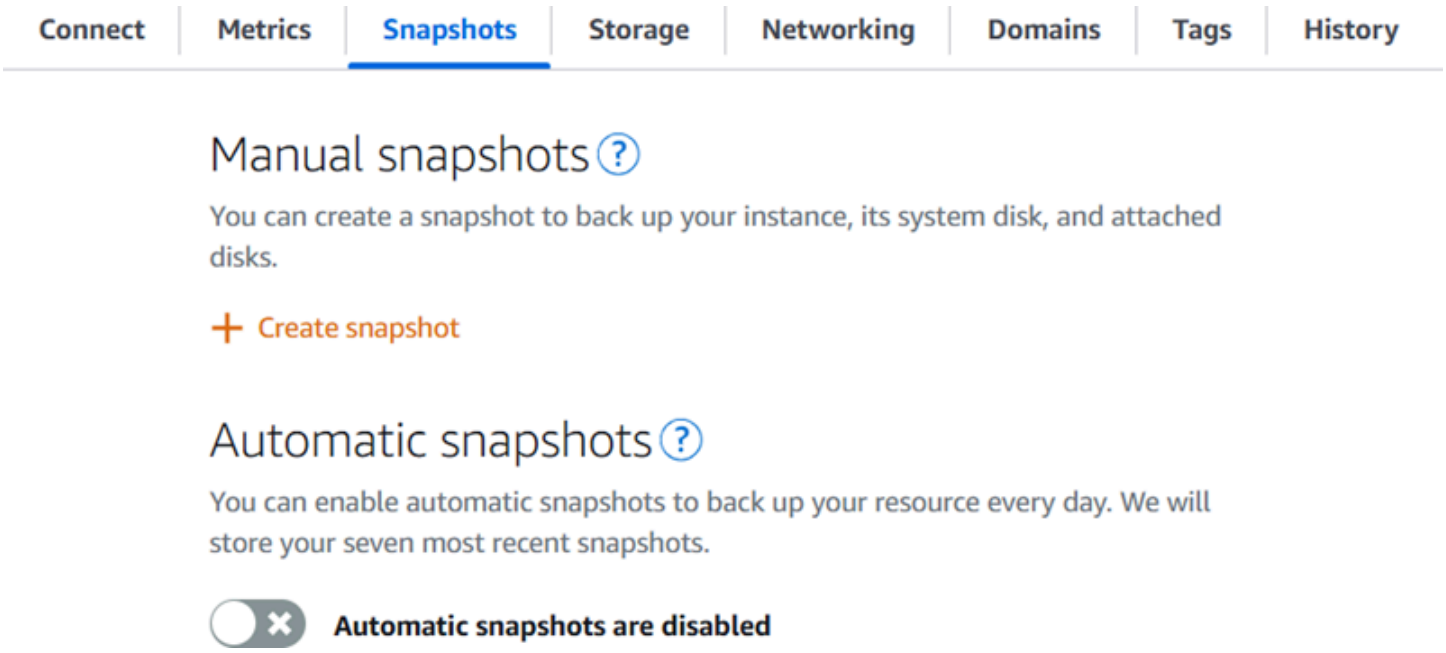
Drupal 설명서를 읽고 웹 사이트를 관리하고 사용자 지정하는 방법을 알아보세요. 자세한 정보는 Drupal 설명서를 참조하세요.

8단계: 인스턴스의 스냅샷 생성

원하는 방식으로 웹 사이트를 구성한 후 인스턴스의 주기적 스냅샷을 생성하여 백업합니다. 스냅샷은 시스템 디스크의 복사본이며, 인스턴스의 원본 구성입니다. 스냅샷은 스냅샷을 생성한 시점부터 인스턴스를 복원하는 데 필요한 모든 데이터를 포함합니다.

스냅샷을 수동으로 생성하거나 자동 스냅샷을 활성화하여 Lightsail에서 매일 스냅샷을 생성하도록 할 수 있습니다. 인스턴스에 문제가 있는 경우 스냅샷을 사용하여 새 교체 인스턴스를 생성할 수 있습니다.

스냅샷 탭의 인스턴스 관리 페이지에서 스냅샷 작업을 할 수 있습니다. 자세한 내용은 Amazon Lightsail의 스냅샷을 참조하세요.



Lightsail에 Ghost 웹 사이트 배포

> 알고 계셨나요?
>
> Lightsail는 일별 스냅샷 7개를 저장하고 인스턴스에 대해 자동 스냅샷을 활성화하면 가장 오래된 스냅샷을 최신 스냅샷으로 자동 대체합니다. 자세한 내용은 Lightsail 인스턴스 및 디스크의 자동 스냅샷 구성을 참조하세요.

다음은 Amazon Lightsail에 Ghost 인스턴스를 설치하고 실행한 후 시작해야 할 몇 가지 단계입니다.

목차

- 1단계: Bitnami 설명서 읽기
- 2단계: Ghost 관리 대시보드에 액세스하기 위한 기본 애플리케이션 암호 가져오기
- 3단계: 인스턴스에 고정 IP 주소 연결
- 4단계: Ghost 웹 사이트의 관리 대시보드에 로그인
- 5단계: Ghost 웹 사이트로 등록된 도메인 이름의 트래픽 라우팅
- 6단계: Ghost 웹 사이트에 대해 HTTPS 구성
- 7단계: Ghost 설명서 읽기 및 웹 사이트 구성 계속
- 8단계: 인스턴스의 스냅샷 생성

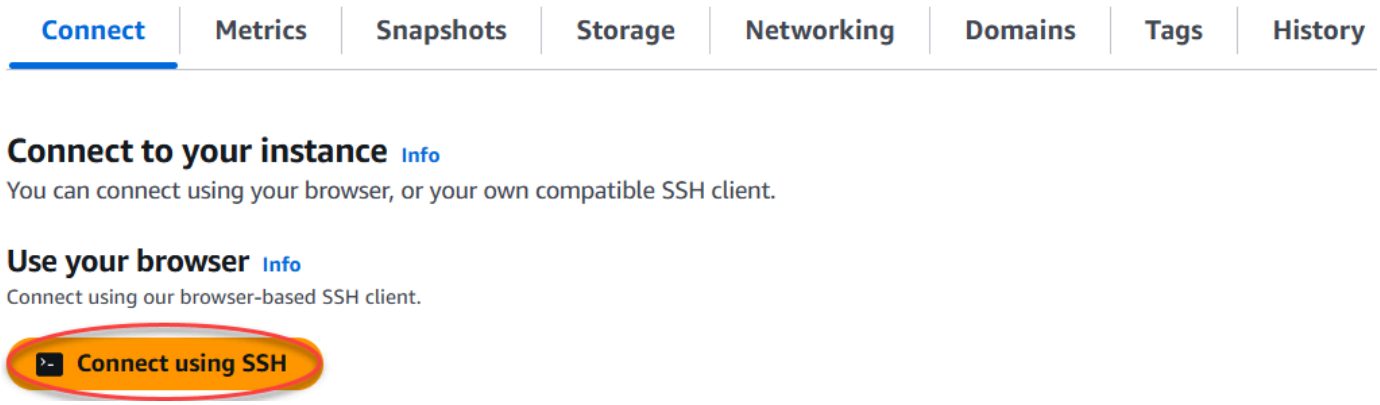
1단계: Bitnami 설명서 읽기

Bitnami 설명서를 읽고 Ghost 애플리케이션을 구성하는 방법을 알아보세요. 자세한 내용은 AWS 클라우드용 Bitnami에서 패키징한 Ghost를 참조하세요.

2단계: Ghost 관리 대시보드에 액세스하기 위한 기본 애플리케이션 암호 가져오기

Ghost 웹 사이트의 관리 대시보드에 액세스하는 데 필요한 기본 애플리케이션 암호를 가져오려면 다음 절차를 완료하세요. 자세한 내용은 Amazon Lightsail의 Bitnami 인스턴스에 대한 애플리케이션 사용자 이름과 암호 가져오기를 참조하세요.

1. 인스턴스 관리 페이지의 연결(Connect) 탭에서 SSH로 연결(Connect using SSH)을 선택합니다.



2. 연결한 후 다음 명령을 입력하여 애플리케이션 암호를 가져옵니다.

```
$ cat $HOME/bitnami_application_password
```

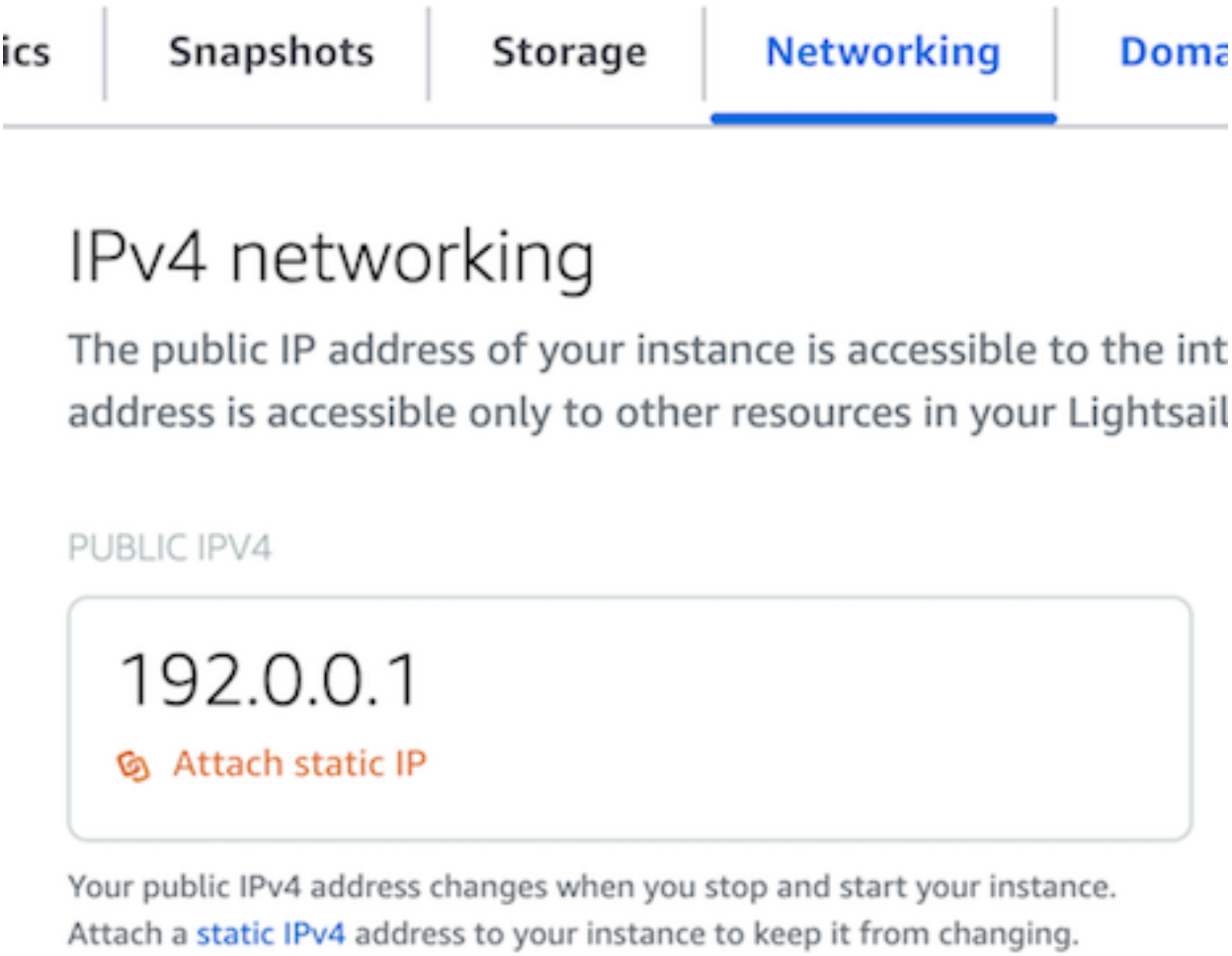
기본 애플리케이션 암호가 포함된 다음과 유사한 응답이 표시되어야 합니다.

```
bitnami@ip-192-0-2-0:~$ cat $HOME/bitnami_application_password
wB2Ex@mplEK6
```

3단계: 인스턴스에 고정 IP 주소 연결

인스턴스에 연결된 기본 동적 퍼블릭 IP 주소는 인스턴스를 중지하고 시작할 때마다 변경됩니다. 퍼블릭 IP 주소가 변경되지 않도록 고정 IP 주소를 생성하고 인스턴스에 연결할 수 있습니다. 나중에 인스턴스에 도메인 이름을 사용하면 인스턴스를 중지하고 시작할 때마다 도메인의 DNS 레코드를 업데이트할 필요가 없습니다. 각 인스턴스에 1개의 고정 IP 주소만 연결할 수 있습니다.

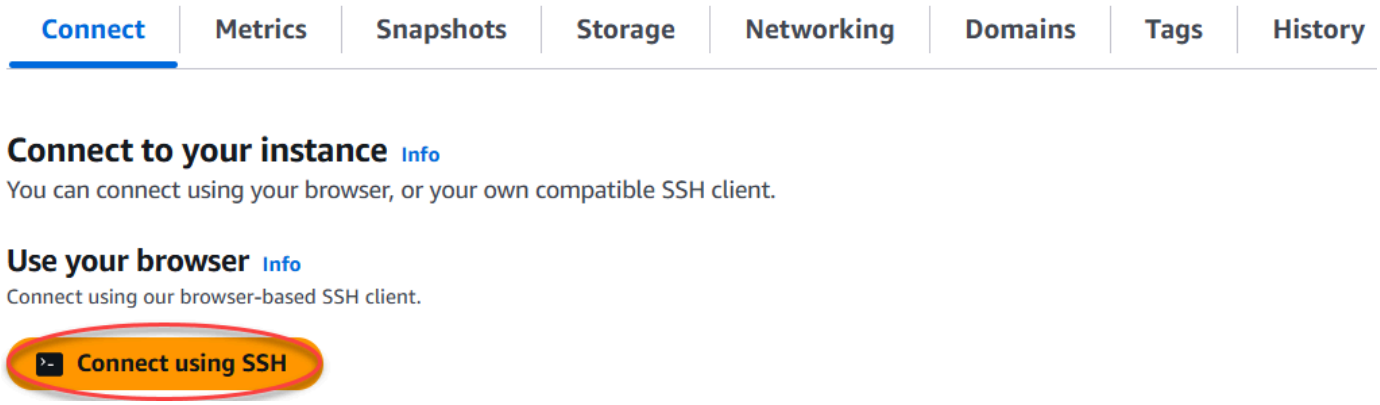
인스턴스 관리 페이지의 네트워킹 탭에서 고정 IP 생성 또는 고정 IP 연결을 선택합니다. 인스턴스에 연결할 수 있는 고정 IP를 생성해 둔 경우에는 페이지의 지침을 따릅니다. 자세한 내용은 고정 IP를 생성하여 인스턴스에 연결을 참조하세요.



새로운 고정 IP 주소가 인스턴스에 연결되면 애플리케이션이 새로운 고정 IP 주소를 인식하도록 다음 단계를 완료해야 합니다.

1. 인스턴스의 고정 IP 주소를 기록해 둡니다. 고정 IP 주소는 인스턴스 관리 페이지의 머리말 섹션에 나와 있습니다.

2. 인스턴스 관리 페이지의 연결(Connect) 탭에서 SSH를 사용하여 연결(Connect using SSH)을 선택합니다.



3. 연결한 후 다음 명령을 입력합니다. *<StaticIP>*를 인스턴스의 새 고정 IP 주소로 바꿉니다.

```
sudo /opt/bitnami/configure_app_domain --domain <StaticIP>
```

예:

```
sudo /opt/bitnami/configure_app_domain --domain 203.0.113.0
```

다음과 유사한 응답이 나타납니다. 인스턴스의 애플리케이션에서 이제 새 고정 IP 주소를 인식해야 합니다.

```
bitnami@ip-203.0.113.0:~$ sudo /opt/bitnami/configure_app_domain --domain
 203.0.113.0
Configuring domain to 203.0.113.0
2024-06-06T21:43:42.393Z - info: Saving configuration info to disk
ghost 21:43:42.78 INFO  ==> Configuring Ghost URL to http://203.0.113.0
Disabling automatic domain update for IP address changes
```

4단계: Ghost 웹 사이트의 관리 대시보드에 로그인

이제 기본 애플리케이션 암호가 있으므로 다음 절차를 완료하여 Ghost 웹 사이트의 홈 페이지로 이동하고 관리 대시보드에 로그인합니다. 로그인한 후 웹 사이트 사용자 지정 및 관리 변경을 시작할 수 있습니다. Ghost에서 수행할 수 있는 작업에 대한 자세한 내용을 알아보려면 이 가이드 뒷부분의 6단계: Ghost 설명서 읽기 및 웹 사이트 구성 계속 섹션을 참조하세요.

1. 인스턴스 관리 페이지의 연결(Connect) 탭에서 인스턴스의 퍼블릭 IP 주소를 기록해 둡니다. 이전에 인스턴스에 고정 IP를 연결한 경우 고정 IP 주소가 됩니다. 퍼블릭 IP 주소는 인스턴스 관리 페이지의 헤더 섹션에도 표시됩니다.

2. 인스턴스의 퍼블릭 IP 주소로 이동합니다(예: `http://203.0.113.0`으로 이동).

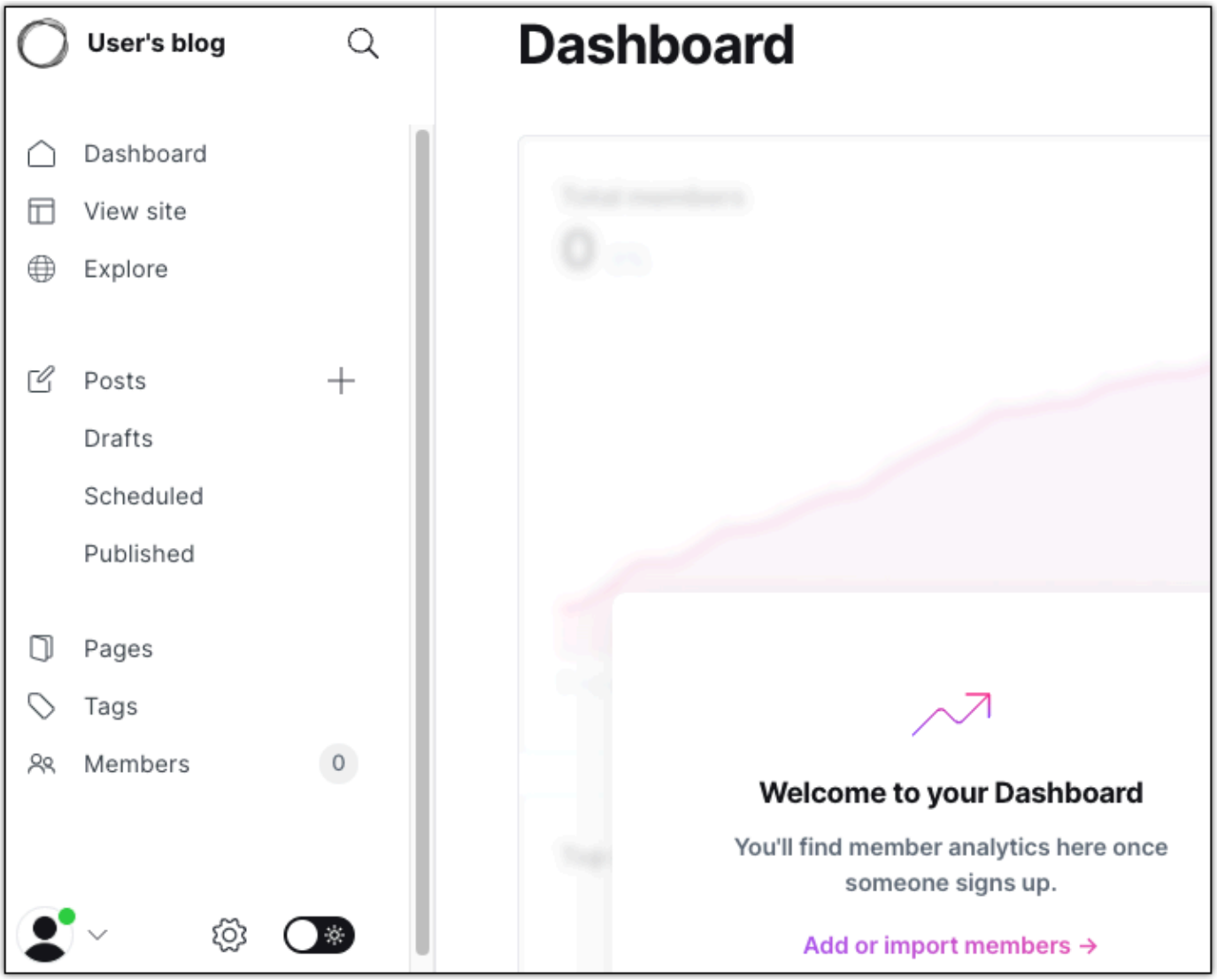
   Ghost 웹 사이트의 홈 페이지가 나타납니다.

3. Ghost 웹 사이트 홈 페이지의 오른쪽 하단 모서리에 있는 관리(Manage)를 선택합니다.

   관리(Manage) 배너가 표시되지 않는 경우 `http://`*`<PublicIP>`*`/ghost`을 통해 로그인 페이지로 이동할 수 있습니다. *`<PublicIP>`*을(를) 인스턴스의 퍼블릭 IP 주소로 바꿉니다.

4. 이 가이드의 초반부에서 검색한 기본 사용자 이름(`user@example.com`) 및 기본 암호를 사용하여 로그인합니다.

   Ghost 관리 대시보드가 나타납니다.



5단계: Ghost 웹 사이트로 등록된 도메인 이름의 트래픽 라우팅

등록된 도메인 이름(예: `example.com`)의 트래픽을 Ghost 웹 사이트로 라우팅하려면 도메인의 DNS에 레코드를 추가하면 됩니다. DNS 레코드는 일반적으로 도메인을 등록한 등록 대행자가 관리 및 호스팅합니다. 그러나 Lightsail 콘솔을 사용하여 관리할 수 있도록 도메인의 DNS 레코드 관리를 Lightsail로 이전하는 것이 좋습니다.

Lightsail 콘솔 홈 페이지의 도메인 및 DNS(Domains & DNS) 섹션에서 DNS 영역 생성을 선택하고 페이지의 지침에 따릅니다. 자세한 내용은 DNS 영역을 생성하여 Lightsail에서 도메인의 DNS 레코드 관리를 참조하세요.

도메인 이름이 트래픽을 인스턴스로 라우팅한 후 Ghost 애플리케이션이 새 도메인을 인식하도록 하려면 다음 단계를 완료해야 합니다.

1. 인스턴스 관리 페이지의 연결(Connect) 탭에서 SSH를 사용하여 연결(Connect using SSH)을 선택합니다.
2. 연결한 후 다음 명령을 입력합니다. *<DomainName>*을 Ghost 인스턴스로 트래픽을 보낼 도메인 이름으로 바꿉니다.

```
$ sudo /opt/bitnami/configure_app_domain --domain <DomainName>
```

예:

```
$ sudo /opt/bitnami/configure_app_domain --domain example.com
```

다음 예와 비슷한 응답이 나타나는 것을 볼 수 있습니다. 이제 Ghost 애플리케이션에서 도메인을 인식합니다.

```
bitnami@ip-203.0.113.0:~$ sudo /opt/bitnami/configure_app_domain --domain
 example.com
Configuring domain to example.com
2024-06-06T21:50:00.393Z - info: Saving configuration info to disk
ghost 21:50:25.78 INFO  ==> Configuring Ghost URL to http://example.com
Disabling automatic domain update for IP address changes
```

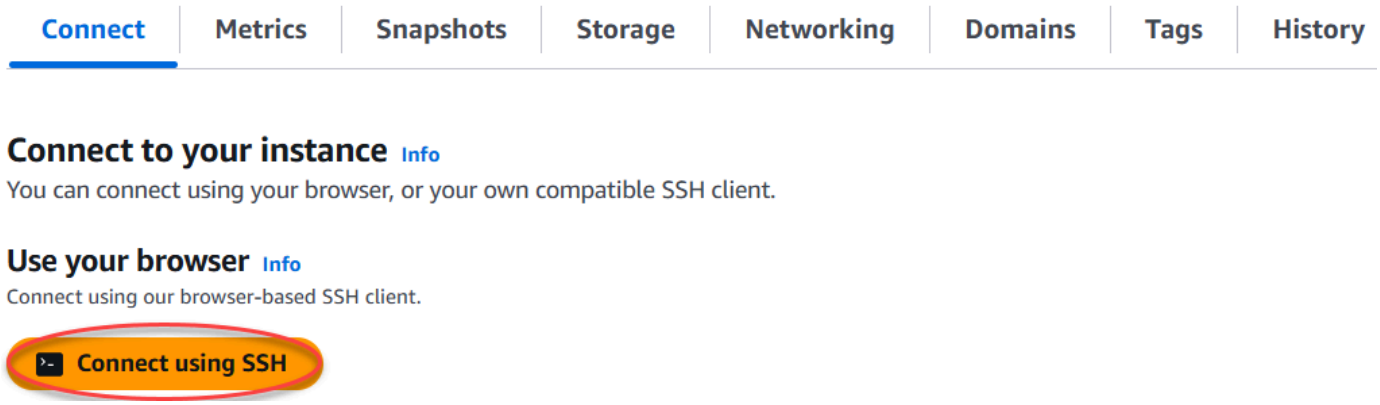
인스턴스에 대해 구성한 도메인 이름으로 이동하면 Ghost 웹 사이트의 홈 페이지로 리디렉션됩니다. 다음으로 Ghost 웹 사이트에 대한 HTTPS 연결을 활성화하기 위해 SSL/TLS 인증서를 생성하고 구성해야 합니다. 자세한 내용을 보려면 이 가이드의 다음 6단계: Ghost 웹 사이트에 대해 HTTPS 구성 섹션으로 계속하세요.

6단계: Ghost 웹 사이트에 대해 HTTPS 구성

Ghost 웹 사이트에서 HTTPS를 구성하려면 다음 절차를 완료하세요. 이 단계에서는 Let's Encrypt SSL/TLS 인증서를 요청하기 위한 명령줄 도구인 Bitnami HTTPS Configuration Tool(`bncert-tool`)을 사용하는 방법을 보여줍니다. 자세한 내용을 알아보려면 Bitnami 설명서의 Learn About The Bitnami HTTPS Configuration Tool(Bitnami Configuration Tool에 대해 알아보기)을 참조하세요.

> **Important**
>
> 이 절차를 시작하기 전에 Ghost 인스턴스로 트래픽을 라우팅하도록 도메인을 구성했는지 확인합니다. 그렇지 않으면 SSL/TLS 인증서 검증 프로세스가 실패합니다.

1. 인스턴스 관리 페이지의 연결(Connect) 탭에서 SSH로 연결(Connect using SSH)을 선택합니다.



2. 연결한 후 다음 명령을 입력하여 bncert 도구가 인스턴스에 설치되었는지 확인합니다.

```
sudo /opt/bitnami/bncert-tool
```

다음과 같은 응답 중 하나가 표시됩니다.

- 응답에 명령을 찾을 수 없음(command not found)이 표시되면 bncert 도구가 인스턴스에 설치되지 않은 것입니다. 이 절차의 다음 단계를 계속 진행하여 인스턴스에 bncert 도구를 설치합니다.
- 응답에 Bitnami HTTPS 구성 도구 시작(Welcome to the Bitnami HTTPS configuration tool)이 표시되면 bncert 도구가 인스턴스에 설치된 것입니다. 이 절차의 8단계로 계속합니다.
- bncert 도구가 일시적으로 인스턴스에 설치된 경우 업데이트된 버전의 도구를 사용할 수 있다는 메시지가 표시될 수 있습니다. 다운로드하도록 선택하고 `sudo /opt/bitnami/bncert-tool` 명령을 입력하여 bncert 도구를 다시 실행합니다. 이 절차의 8단계로 계속합니다.

3. 다음 명령을 입력하여 bncert 실행 파일을 인스턴스로 다운로드합니다.

```
wget -O bncert-linux-x64.run https://downloads.bitnami.com/files/bncert/latest/
bncert-linux-x64.run
```

4. 다음 명령을 입력하여 인스턴스에서 bncert 도구 실행 파일의 디렉터리를 생성합니다.

```
sudo mkdir /opt/bitnami/bncert
```

5. 다음 명령을 입력하여 bncert에서 프로그램으로 실행할 수 있는 파일을 실행하도록 합니다.

```
sudo chmod +x /opt/bitnami/bncert/bncert-linux-x64.run
```

6. 다음 명령을 입력하여 sudo /opt/bitnami/bncert-tool 명령을 입력할 때 bncert 도구를 실행하는 심볼 링크를 생성합니다.

```
sudo ln -s /opt/bitnami/bncert/bncert-linux-x64.run /opt/bitnami/bncert-tool
```

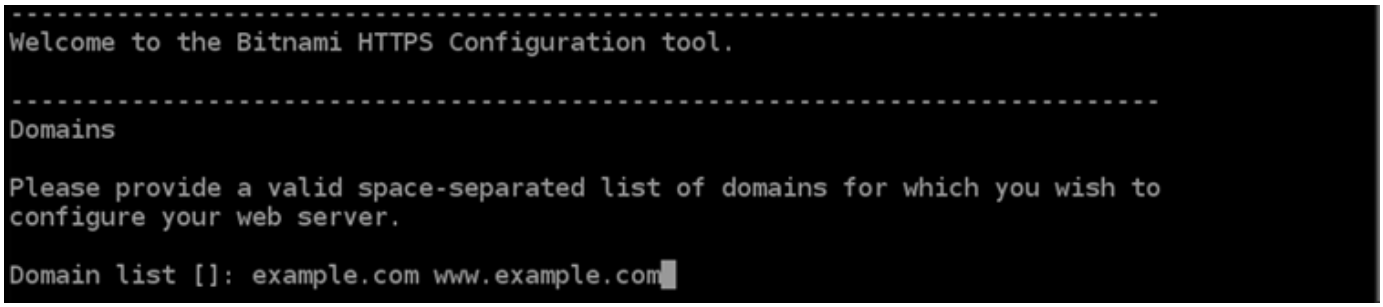
이제 인스턴스에 bncert 도구를 설치했습니다.

7. 다음 명령을 입력하여 bncert 도구를 실행합니다.

```
sudo /opt/bitnami/bncert-tool
```

8. 다음 예와 같이 기본 도메인 이름과 대체 도메인 이름을 공백으로 구분하여 입력합니다.

도메인이 트래픽을 인스턴스의 퍼블릭 IP 주소로 라우팅하도록 구성되지 않은 경우, bncert 도구에서 계속하기 전에 해당 구성을 설정하라는 메시지를 표시합니다. 도메인은 bncert 도구를 사용하여 인스턴스에서 HTTPS를 활성화한 인스턴스의 퍼블릭 IP 주소로 트래픽을 라우팅해야 합니다. 이렇게 해야 도메인을 소유하고 있음을 확인하고 인증서를 검증하는 역할을 할 수 있습니다.



9. bncert 도구는 웹 사이트의 리디렉션을 어떻게 구성할지 묻는 메시지를 표시합니다. 사용할 수 있는 옵션은 다음과 같습니다.

- HTTP에서 HTTPS로 리디렉션 활성화(Enable HTTP to HTTPS redirection) - HTTP 버전 웹 사이트(예: http:/example.com)를 방문하는 사용자를 HTTPS 버전(예: https://example.com)으로 자동 리디렉션할지 지정합니다. 모든 방문자가 암호화된 연결을 사용하도록 강제하기 때문에 이 옵션을 활성화하는 것이 좋습니다. Y를 입력하고 Enter 키를 눌러 활성화합니다.
- 비 www에서 www로 리디렉션 활성화(Enable non-www to www redirection) - 도메인의 정점(https://example.com)을 방문하는 사용자를 도메인의 www 하위 도메인(https://www.example.com)으로 자동 리디렉션할지 지정합니다. 이 옵션은 활성화하는 것이 좋습니다. 그러나 Google의 웹 마스터 도구와 같은 검색 엔진 도구에서 도메인의 정점을 기본 웹 사이트 주소로 지정했거나 정점이 IP를 직접 가리키고 www 하위 도메인이 CNAME 레코드를 통해 정점을 참조하는 경우, 이 옵션을 비활성화하고 대체 옵션(www에서 비 www로 리디렉션 활성화)을 활성화할 수 있습니다. Y를 입력하고 Enter 키를 눌러 활성화합니다.

- www에서 비 www로 리디렉션 활성화(Enable www to non-www redirection) - 도메인의 www 하위 도메인(`https://www.example.com`)을 방문하는 사용자를 도메인의 정점(`https://example.com`)으로 자동 리디렉션할지 지정합니다. 비 www에서 www로 리디렉션을 활성화한 경우 이 옵션을 비활성화하는 것이 좋습니다. N를 입력하고 Enter 키를 눌러 비활성화합니다.

선택한 내용은 다음 예와 같아야 합니다.

```
Enable/disable redirections

Please select the redirections you wish to enable or disable on your Bitnami
installation.



Enable HTTP to HTTPS redirection [Y/n]: Y



Enable non-www to www redirection [Y/n]: Y



Enable www to non-www redirection [y/N]: N
```

10. 적용될 변경 사항의 목록이 나열됩니다. Y를 입력하고 Enter 키를 눌러 확인하고 계속합니다.

```
Changes to perform

The following changes will be performed to your Bitnami installation:

1. Stop web server
2. Configure web server to use a free Let's Encrypt certificate for the domains:
example.com www.example.com
3. Configure a cron job to automatically renew the certificate each month
4. Configure web server name to: example.com
5. Enable HTTP to HTTPS redirection (example: redirect http://example.com to
https://example.com)
6. Enable non-www to www redirection (example: redirect example.com to
 www.example.com)
7. Start web server once all changes have been performed



Do you agree to these changes? [Y/n]: Y
```

11. Let's Encrypt 인증서에 연결할 이메일 주소를 입력하고 Enter 키를 누릅니다.

```
Create a free HTTPS certificate with Let's Encrypt

Please provide a valid e-mail address for which to associate your Let's Encrypt
certificate.

Domain list: example.com www.example.com

Server name: example.com

E-mail address []:
```

12. Let's Encrypt 구독자 계약을 검토합니다. Y를 입력하고 Enter 키를 눌러 계약을 수락하고 계속합니다.

```
The Let's Encrypt Subscriber Agreement can be found at:

https://letsencrypt.org/documents/LE-SA-v1.2-November-15-2017.pdf

Do you agree to the Let's Encrypt Subscriber Agreement? [Y/n]:
```

인증서 요청 및 지정한 리디렉션 구성을 비롯하여 인스턴스에서 HTTPS를 활성화하는 작업이 수행됩니다.

```
Performing changes to your installation

The Bitnami HTTPS Configuration Tool will perform any necessary actions to your
Bitnami installation. This may take some time, please be patient.

|
```

인증서가 성공적으로 발급되고 검증됩니다. 다음 예와 유사한 메시지가 표시되면 인스턴스에서 리디렉션이 성공적으로 구성된 것입니다.

```
Success

The Bitnami HTTPS Configuration Tool succeeded in modifying your installation.

The configuration report is shown below.

Backup files:
* /opt/bitnami/apache2/conf/httpd.conf.back.202005290035
* /opt/bitnami/apache2/conf/bitnami/bitnami-apps-prefix.conf.back.202005290035
* /opt/bitnami/apache2/conf/bitnami/bitnami.conf.back.202005290035

Find more details in the log file:

/tmp/bncert-202005290035.log

If you find any issues, please check Bitnami Support forums at:

https://community.bitnami.com

Press [Enter] to continue:
```

`bncert` 도구는 인증서가 만료되기 80일 전마다 인증서를 자동으로 갱신합니다. 인스턴스와 함께 추가 도메인 및 하위 도메인을 사용하고 이러한 도메인에서 HTTPS를 활성화하려면 위의 단계를 반복합니다.

> **Tip**
>
> 다음 명령을 입력하여 인스턴스의 서비스를 다시 시작합니다.
>
> ```
> sudo /opt/bitnami/ctlscript.sh restart
> ```

이제 Ghost 인스턴스에서 HTTPS가 활성화되었습니다. 다음에 구성한 도메인을 사용하여 Ghost 웹 사이트로 이동하면 HTTPS 연결로 리디렉션됩니다.

7단계: Ghost 설명서 읽기 및 웹 사이트 구성 계속

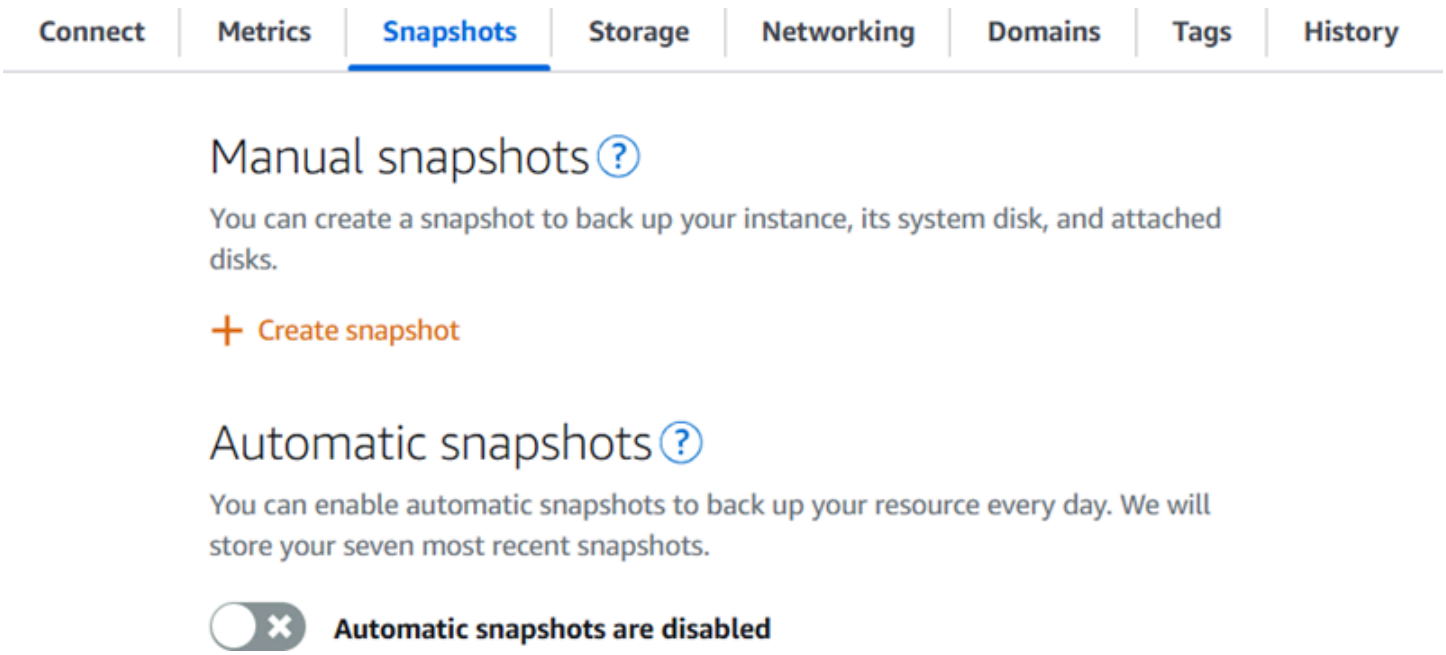
Ghost 설명서를 읽고 웹 사이트를 관리하고 사용자 지정하는 방법을 알아보세요. 자세한 내용을 알아보려면 Ghost 설명서를 참조하세요.

8단계: 인스턴스의 스냅샷 생성

원하는 방식으로 웹 사이트를 구성한 후 인스턴스의 주기적 스냅샷을 생성하여 백업합니다. 스냅샷은 시스템 디스크의 복사본이며, 인스턴스의 원본 구성입니다. 스냅샷은 스냅샷을 생성한 시점부터 인스턴스를 복원하는 데 필요한 모든 데이터를 포함합니다.

스냅샷을 수동으로 생성하거나 자동 스냅샷을 활성화하여 Lightsail에서 매일 스냅샷을 생성하도록 할 수 있습니다. 인스턴스에 문제가 있는 경우 스냅샷을 사용하여 새 교체 인스턴스를 생성할 수 있습니다.

스냅샷 탭의 인스턴스 관리 페이지에서 스냅샷 작업을 할 수 있습니다. 자세한 내용은 Amazon Lightsail의 스냅샷을 참조하세요.



Lightsail에서 GitLab CE 인스턴스 설정 및 구성

> 알고 계셨나요?
>
> Lightsail는 일별 스냅샷 7개를 저장하고 인스턴스에 대해 자동 스냅샷을 활성화하면 가장 오래된 스냅샷을 최신 스냅샷으로 자동 대체합니다. 자세한 내용은 Lightsail 인스턴스 및 디스크의 자동 스냅샷 구성을 참조하세요.

다음은 Amazon Lightsail에 GitLab CE 인스턴스를 설치하고 실행한 후 시작해야 할 몇 가지 단계입니다.

목차

- 1단계: Bitnami 설명서 읽기
- 2단계: GitLab CE 관리 영역에 액세스하기 위한 기본 애플리케이션 암호 가져오기
- 3단계: 인스턴스에 고정 IP 주소 연결
- 4단계: Gitlab CE 웹 사이트의 관리 영역에 로그인
- 5단계: GitLab CE 웹 사이트로 등록된 도메인 이름의 트래픽 라우팅
- 6단계: GitLab CE 웹 사이트에 대해 HTTPS 구성
- 7단계: GitLab CE 설명서 읽기 및 웹 사이트 구성 계속

- 8단계: 인스턴스의 스냅샷 생성

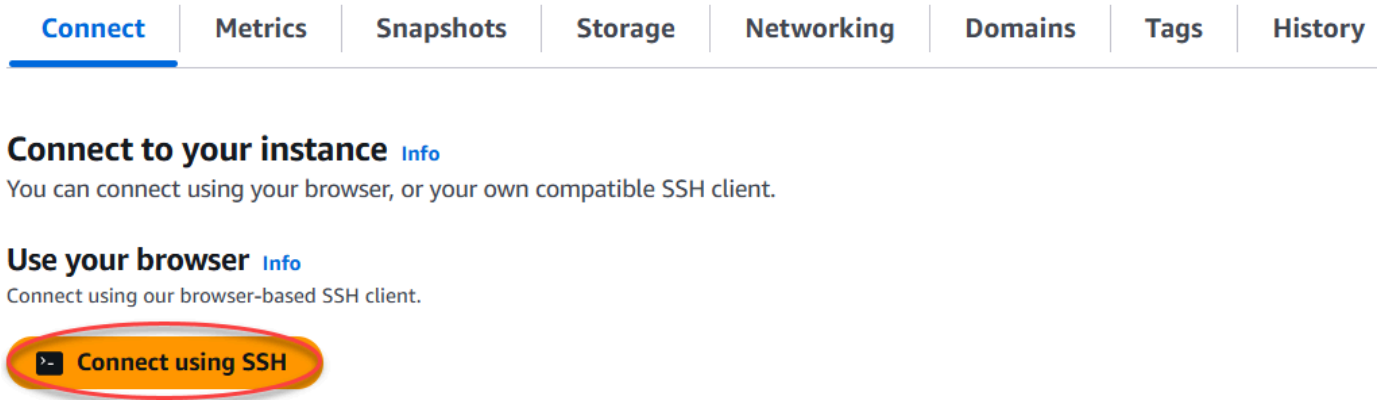
1단계: Bitnami 설명서 읽기

Bitnami 설명서를 읽고 GitLab CE 애플리케이션을 구성하는 방법을 알아보세요. 자세한 내용은 AWS 클라우드용 Bitnami에서 패키징한 GitLab CE를 참조하세요.

2단계: GitLab CE 관리 영역에 액세스하기 위한 기본 애플리케이션 암호 가져오기

Gitlab CE 웹 사이트의 관리 영역에 액세스하는 데 필요한 기본 애플리케이션 암호를 가져오려면 다음 절차를 완료하세요. 자세한 내용은 Amazon Lightsail의 Bitnami 인스턴스에 대한 애플리케이션 사용자 이름과 암호 가져오기를 참조하세요.

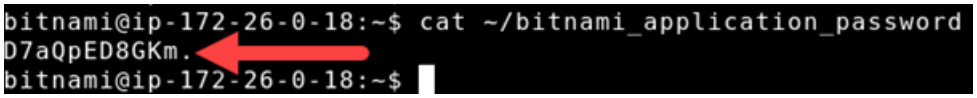
1. 인스턴스 관리 페이지의 연결(Connect) 탭에서 SSH로 연결(Connect using SSH)을 선택합니다.



2. 연결한 후 다음 명령을 입력하여 애플리케이션 암호를 가져옵니다.

```
cat $HOME/bitnami_application_password
```

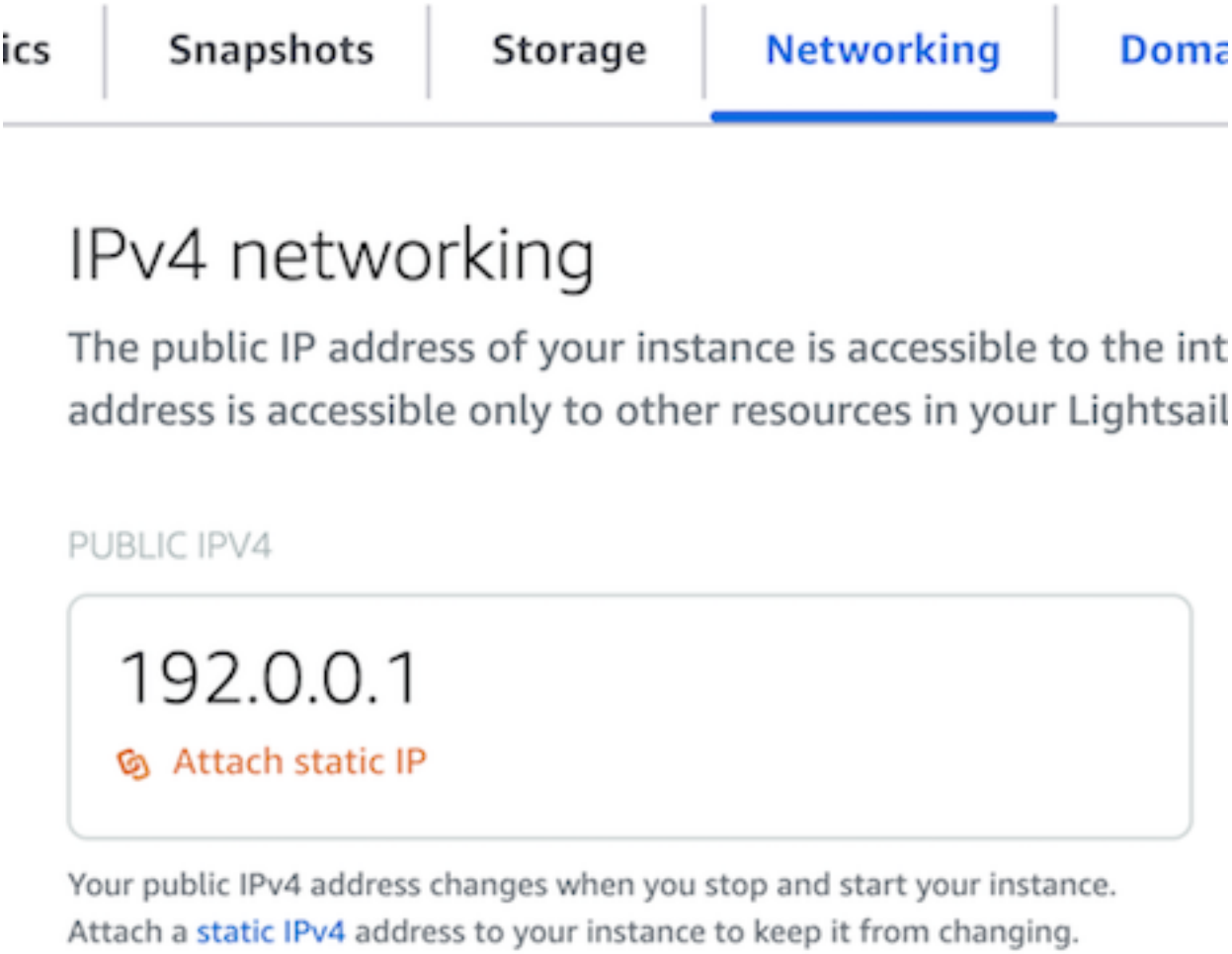
기본 애플리케이션 암호가 포함된 다음 예제와 유사한 응답이 표시되어야 합니다.



3단계: 인스턴스에 고정 IP 주소 연결

인스턴스에 연결된 기본 동적 퍼블릭 IP 주소는 인스턴스를 중지하고 시작할 때마다 변경됩니다. 퍼블릭 IP 주소가 변경되지 않도록 고정 IP 주소를 생성하고 인스턴스에 연결할 수 있습니다. 나중에 인스턴스에 도메인 이름을 사용하면 인스턴스를 중지하고 시작할 때마다 도메인의 DNS 레코드를 업데이트할 필요가 없습니다. 각 인스턴스에 1개의 고정 IP 주소만 연결할 수 있습니다.

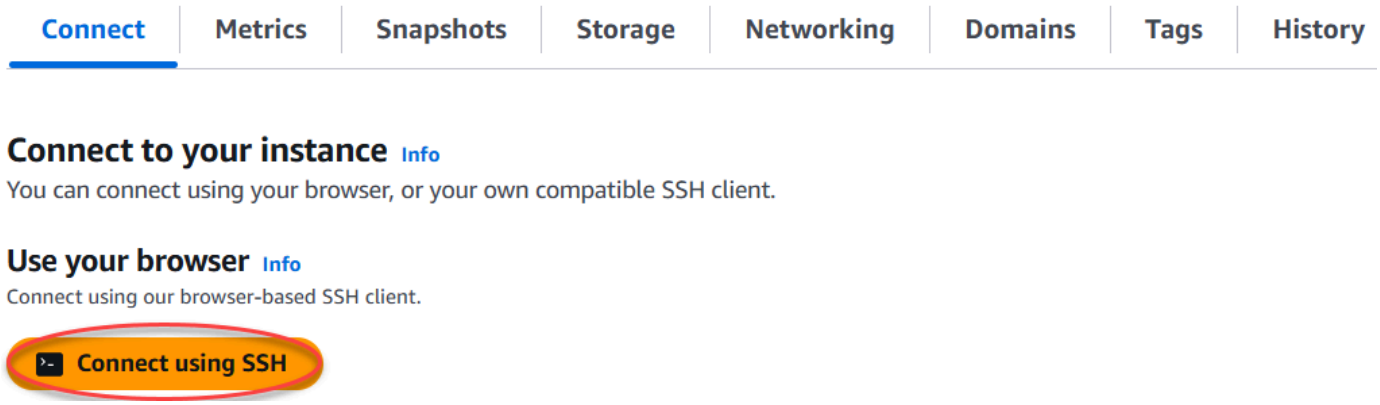
인스턴스 관리 페이지의 네트워킹 탭에서 고정 IP 생성 또는 고정 IP 연결을 선택합니다. 인스턴스에 연결할 수 있는 고정 IP를 생성해 둔 경우에는 페이지의 지침을 따릅니다. 자세한 내용은 고정 IP를 생성하여 인스턴스에 연결을 참조하세요.



새로운 고정 IP 주소가 인스턴스에 연결되면 애플리케이션이 새로운 고정 IP 주소를 인식하도록 다음 단계를 완료해야 합니다.

1. 인스턴스의 고정 IP 주소를 기록해 둡니다. 고정 IP 주소는 인스턴스 관리 페이지의 머리말 섹션에 나와 있습니다.

2. 인스턴스 관리 페이지의 연결(Connect) 탭에서 SSH를 사용하여 연결(Connect using SSH)을 선택합니다.



3. 연결한 후 다음 명령을 입력합니다. *<StaticIP>*를 인스턴스의 새 고정 IP 주소로 바꿉니다.

```
sudo /opt/bitnami/configure_app_domain --domain <StaticIP>
```

예:

```
sudo /opt/bitnami/configure_app_domain --domain 203.0.113.0
```

다음 예와 비슷한 응답이 나타나는 것을 볼 수 있습니다. 인스턴스의 애플리케이션에서 이제 새 고정 IP 주소를 인식해야 합니다.

```
bitnami@ip-[redacted]:~$ sudo /opt/bitnami/configure_app_domain --domain 203.0.113.0
Configuring domain to 203.0.113.0
2022-06-09T16:47:06.737Z - info: Saving configuration info to disk
gitlab 16:47:06.86 INFO  ==> Updating external URL in GitLab configuration
gitlab 16:47:06.88 INFO  ==> Reconfiguring GitLab
gitlab 16:47:45.29 INFO  ==> Starting GitLab services
Disabling automatic domain update for IP address changes
```

4단계: Gitlab CE 웹 사이트의 관리 영역에 로그인

이제 기본 사용자 암호가 있으므로 GitLab CE 웹 사이트의 홈 페이지로 이동하여 관리 영역에 로그인합니다. 로그인한 후 웹 사이트 사용자 지정 및 관리 변경을 시작할 수 있습니다. GitLab CE에서 수행할 수 있는 작업에 대한 자세한 내용을 알아보려면 이 가이드 뒷부분의 7단계: GitLab CE 설명서 읽기 및 웹 사이트 구성 계속 섹션을 참조하세요.

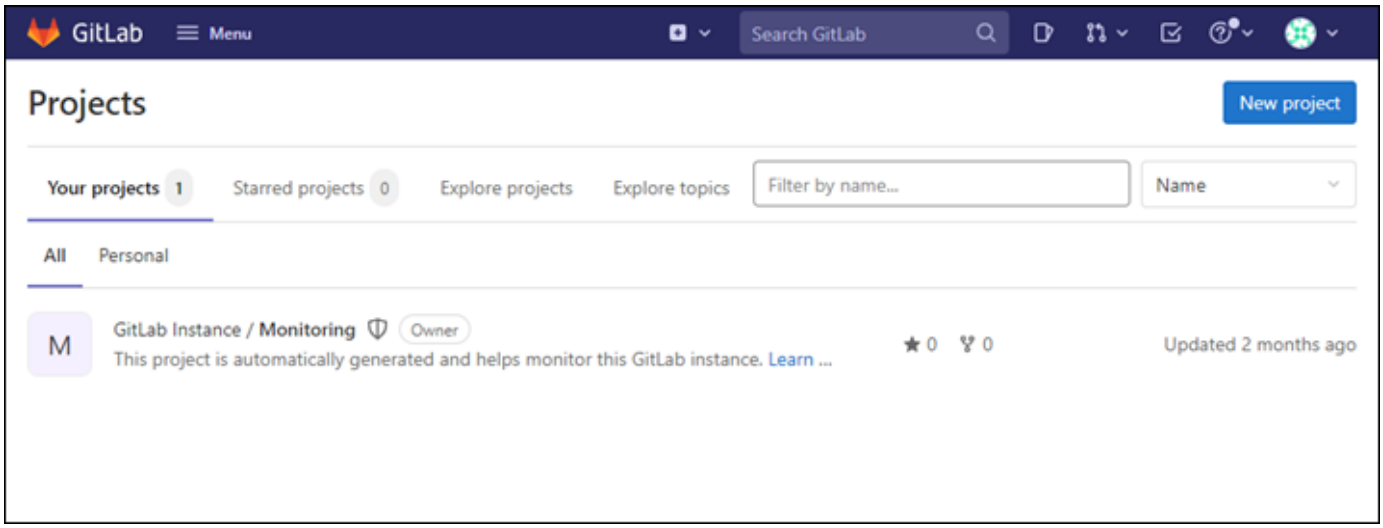
1. 인스턴스 관리 페이지의 연결(Connect) 탭에서 인스턴스의 퍼블릭 IP 주소를 기록해 둡니다. 퍼블릭 IP 주소는 인스턴스 관리 페이지의 헤더 섹션에도 표시됩니다.

2. 인스턴스의 퍼블릭 IP 주소로 이동합니다(예: `http://203.0.113.0`으로 이동).

   GitLab CE 웹 사이트의 홈 페이지가 나타납니다. 인터넷 연결이 비공개가 아니거나 안전하지 않거나 보안 위험이 있다는 브라우저 경고 메시지가 표시될 수도 있습니다. 이 경고는 GitLab CE 인스턴스에 아직 SSL/TLS 인증서가 적용되지 않았기 때문에 발생합니다. 브라우저 창에서 Advanced(고급), Details(세부 정보) 또는 More information(추가 정보)을 선택하여 사용할 수 있는 옵션을 표시합니다. 그런 다음 비공개가 아니거나 안전하지 않더라도 웹 사이트로 이동하도록 선택합니다.

3. 이 가이드의 초반부에서 검색한 기본 사용자 이름(`root`) 및 기본 암호를 사용하여 로그인합니다.

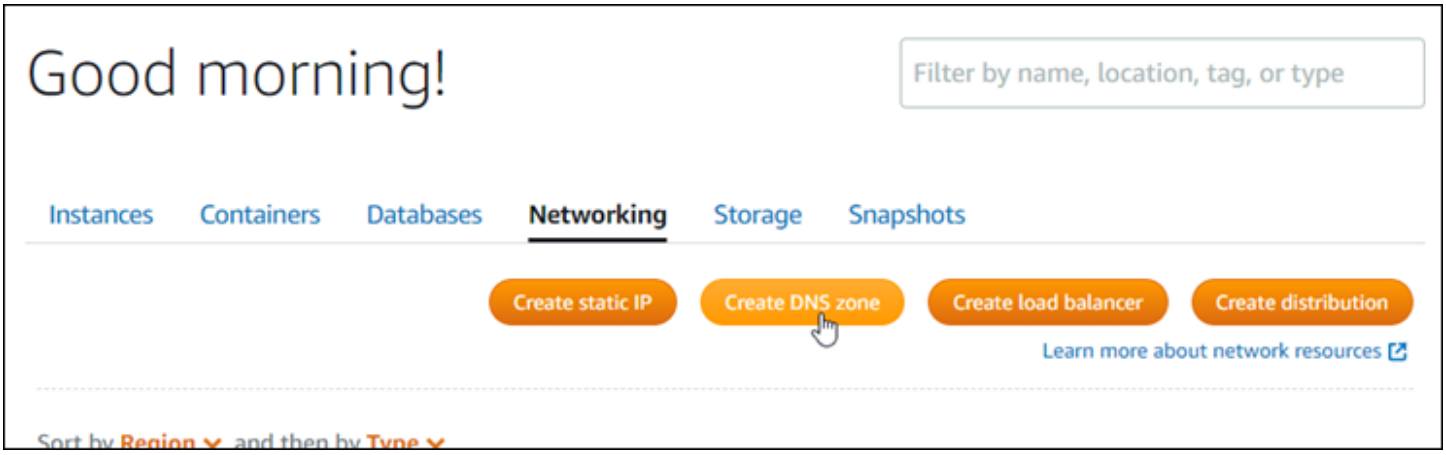
   Gitlab CE 관리 대시보드가 나타납니다.



5단계: GitLab CE 웹 사이트로 등록된 도메인 이름의 트래픽 라우팅

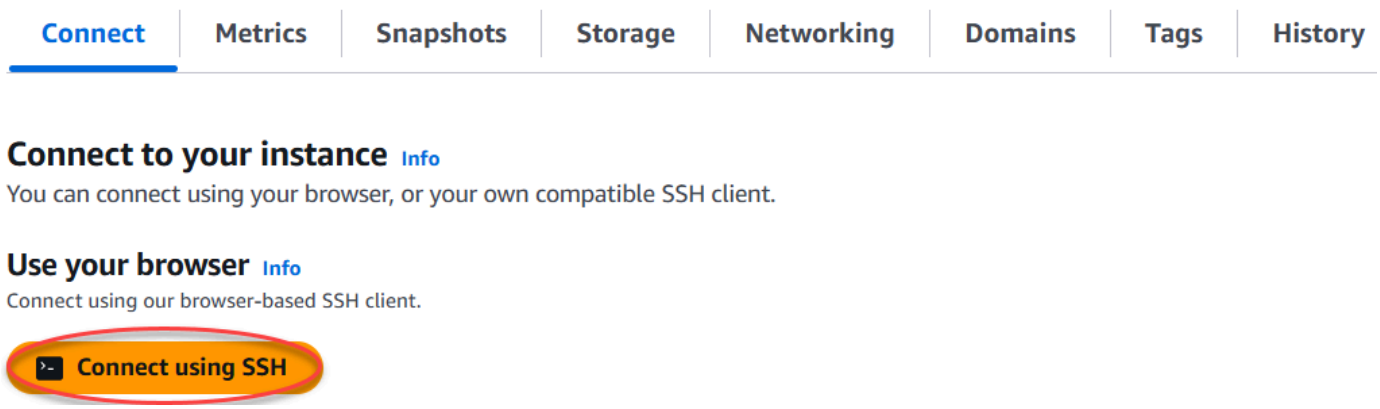
등록된 도메인 이름(예: `example.com`)의 트래픽을 GitLab CE 웹 사이트로 라우팅하려면 도메인의 도메인 이름 시스템(DNS)에 레코드를 추가하면 됩니다. DNS 레코드는 일반적으로 도메인을 등록한 등록 대행자가 관리 및 호스팅합니다. 그러나 Lightsail 콘솔을 사용하여 관리할 수 있도록 도메인의 DNS 레코드 관리를 Lightsail로 이전하는 것이 좋습니다.

Lightsail 콘솔 홈 페이지의 네트워킹(Networking) 탭에서 DNS 영역 생성(Create DNS zone)을 선택하고 페이지의 지침에 따릅니다. 자세한 내용은 DNS 영역을 생성하여 도메인의 DNS 레코드 관리를 참조하세요.



도메인 이름이 트래픽을 인스턴스로 라우팅한 후 GitLab CE가 도메인 이름을 인식하도록 하려면 다음 절차를 완료해야 합니다.

1. 인스턴스 관리 페이지의 연결(Connect) 탭에서 SSH를 사용하여 연결(Connect using SSH)을 선택합니다.



2. 연결한 후 다음 명령을 입력합니다. *<DomainName>*을 인스턴스로 트래픽을 라우팅할 도메인 이름으로 바꿉니다.

```
sudo /opt/bitnami/configure_app_domain --domain <DomainName>
```

예:

```
sudo /opt/bitnami/configure_app_domain --domain example.com
```

다음 예와 비슷한 응답이 나타나는 것을 볼 수 있습니다. 이제 GitLab CE 인스턴스에서 도메인 이름을 인식합니다.

```
bitnami@ip-[redacted]:~$ sudo /opt/bitnami/configure_app_domain --domain example.com
Configuring domain to example.com
2022-06-09T18:44:00.235Z - info: Saving configuration info to disk
gitlab 18:44:00.36 INFO  ==> Updating external URL in GitLab configuration
gitlab 18:44:00.37 INFO  ==> Reconfiguring GitLab
gitlab 18:44:38.79 INFO  ==> Starting GitLab services
Disabling automatic domain update for IP address changes
```

Note

명령이 실패하면 이전 GitLab CE 인스턴스 버전을 사용하고 있는 것일 수 있습니다. 이 경우 대신 다음 명령을 실행합니다. *<DomainName>*을 인스턴스로 트래픽을 라우팅할 도메인 이름으로 바꿉니다.

```
cd /opt/bitnami/apps/gitlab && \
sudo ./bnconfig --machine_hostname <DomainName>
```

위 명령을 실행한 후 다음 명령을 입력하여 서버가 다시 시작될 때마다 bnconfig 도구가 자동으로 실행되지 않도록 합니다.

```
sudo mv bnconfig bnconfig.disabled
```

다음으로 GitLab CE 웹 사이트에 대한 HTTPS 연결을 활성화하기 위해 SSL/TLS 인증서를 생성하고 구성해야 합니다. 자세한 내용을 보려면 이 가이드의 다음 6단계: GitLab CE 웹 사이트에 대해 HTTPS 구성 섹션으로 계속하세요.

6단계: GitLab CE 웹 사이트에 대해 HTTPS 구성

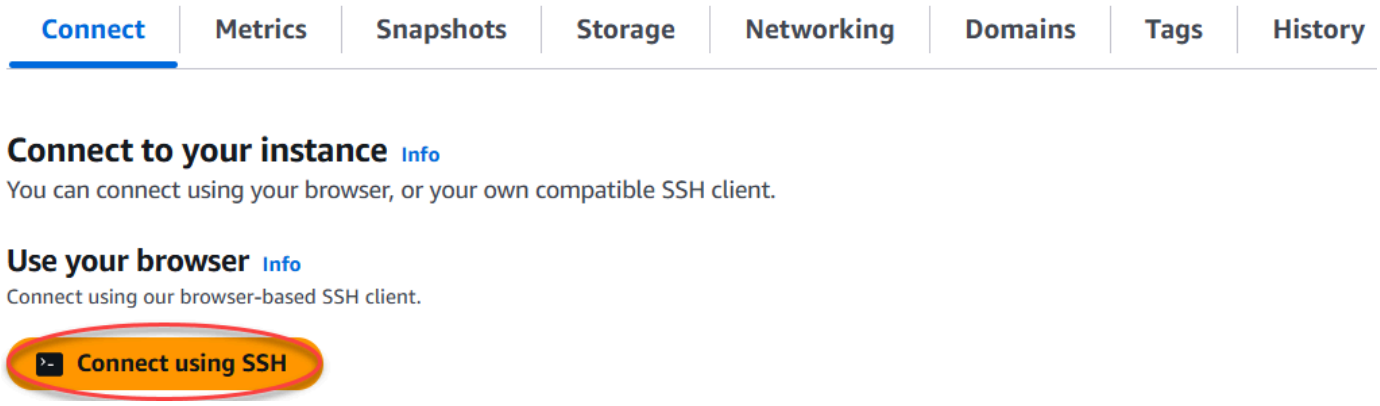
GitLab CE 웹 사이트에서 HTTPS를 구성하려면 다음 절차를 완료하세요. 이 단계에서는 Let's Encrypt SSL/TLS 인증서를 요청하기 위한 명령줄 도구인 Lego 클라이언트를 사용하는 방법을 보여줍니다.

Important

이 절차를 시작하기 전에 GitLab CE 인스턴스로 트래픽을 라우팅하도록 도메인을 구성했는지 확인합니다. 그렇지 않으면 SSL/TLS 인증서 검증 프로세스가 실패합니다. 등록된 도메인 이름의 트래픽을 라우팅하려면 도메인의 DNS에 레코드를 추가하면 됩니다. DNS 레코드는 일반적

으로 도메인을 등록한 등록 대행자가 관리 및 호스팅합니다. 그러나 Lightsail 콘솔을 사용하여 관리할 수 있도록 도메인의 DNS 레코드 관리를 Lightsail로 이전하는 것이 좋습니다. Lightsail 콘솔 홈 페이지의 도메인 및 DNS(Domains & DNS) 탭에서 DNS 영역 생성(Create DNS zone)을 선택하고 페이지의 지침에 따릅니다. 자세한 내용은 DNS 영역을 생성하여 Lightsail에서 도메인의 DNS 레코드 관리를 참조하세요.

1. 인스턴스 관리 페이지의 연결(Connect) 탭에서 SSH로 연결(Connect using SSH)을 선택합니다.



2. 연결한 후 다음 명령을 입력하여 디렉터리를 임시(/tmp) 디렉터리로 변경합니다.

```
cd /tmp
```

3. 다음 명령을 입력하여 최신 버전의 Lego 클라이언트를 다운로드합니다. 이 명령은 테이프 아카이브(tar) 파일을 다운로드합니다.

```
curl -Ls https://api.github.com/repos/xenolf/lego/releases/latest | grep
 browser_download_url | grep linux_amd64 | cut -d '"' -f 4 | wget -i -
```

4. 다음 명령을 입력하여 tar 파일에서 파일을 추출합니다. *X.Y.Z*를 다운로드한 Lego 클라이언트의 버전으로 바꿉니다.

```
tar xf lego_vX.Y.Z_linux_amd64.tar.gz
```

예:

```
tar xf lego_v4.7.0_linux_amd64.tar.gz
```

5. 다음 명령을 입력하여 Lego 클라이언트 파일을 이동할 `/opt/bitnami/letsencrypt` 디렉터리를 생성합니다.

```
sudo mkdir -p /opt/bitnami/letsencrypt
```

6. 다음 명령을 입력하여 Lego 클라이언트 파일을 생성한 디렉터리로 이동합니다.

```
sudo mv lego /opt/bitnami/letsencrypt/lego
```

7. 다음 명령을 입력하여 인스턴스에서 실행 중인 애플리케이션 서비스를 중지합니다.

```
sudo service bitnami stop && \
sudo service gitlab-runsvdir stop
```

8. 다음 명령을 입력하여 Lego 클라이언트를 사용하여 Let's Encrypt SSL/TLS 인증서를 요청합니다.

```
sudo /opt/bitnami/letsencrypt/lego --tls --email="EmailAddress" --
domains="RootDomain" --domains="WwwSubDomain" --path="/opt/bitnami/letsencrypt" run
```

명령에서 다음 예제 값을 사용자의 값으로 바꿉니다.

- *EmailAddress* - 등록 알림을 위한 이메일 주소입니다.
- *RootDomain* - GitLab CE 웹 사이트(예: `example.com`)로 트래픽을 라우팅하는 기본 루트 도메인입니다.
- *WwwSubDomain* - GitLab CE 웹 사이트(예: `www.example.com`)로 트래픽을 라우팅하는 기본 루트 도메인의 `www` 하위 도메인입니다.

  명령에 추가 `--domains` 파라미터를 지정하여 인증서에 대해 여러 도메인을 지정할 수 있습니다. 여러 도메인을 지정하면 Lego에서 지정된 모든 도메인에 대해 하나의 인증서만 유효한 제목 대체 이름(SAN) 인증서를 생성합니다. 목록의 첫 번째 도메인은 인증서의 "CommonName"으로 추가되고 나머지는 인증서 내의 SAN 확장에 "DNSNames"로 추가됩니다.

예:

```
sudo /opt/bitnami/letsencrypt/lego --tls --email="user@example.com" --
domains="example.com" --domains="www.example.com" --path="/opt/bitnami/letsencrypt"
 run
```

9. 메시지가 나타날 때 서비스 약관에 동의하려면 Y 키를 누르고 Enter 키를 누릅니다.

다음 예와 비슷한 응답이 나타나는 것을 볼 수 있습니다.

```
2022/06/09 19:23:27 [INFO] [ example.com ] Server responded with a certificate.
```

성공하면 인증서 집합이 `/opt/bitnami/letsencrypt/certificates` 디렉터리에 저장됩니다. 이 집합에는 서버 인증서 파일(예: `example.com.crt`)과 서버 인증서 키 파일(예: `example.com.key`)이 포함됩니다.

10. 다음 명령을 입력하여 인스턴스의 기존 인증서 이름을 변경합니다. 나중에 이러한 기존 인증서를 새 Let's Encrypt 인증서로 바꿉니다.

```
sudo mv /etc/gitlab/ssl/tls.crt /etc/gitlab/ssl/tls.crt.old && \
sudo mv /etc/gitlab/ssl/tls.key /etc/gitlab/ssl/tls.key.old
```

11. 다음 명령을 입력하여 GitLab CE 인스턴스의 기본 인증서 디렉터리인 `/etc/gitlab/ssl` 디렉터리에 새 Let's Encript 인증서에 대한 심볼 링크를 생성합니다.

```
sudo ln -sf /opt/bitnami/letsencrypt/certificates/Domain.key /etc/gitlab/ssl/
tls.key && \
sudo ln -sf /opt/bitnami/letsencrypt/certificates/Domain.crt /etc/gitlab/ssl/
tls.crt
```

명령에서 *Domain*을 Let's Encrypt 인증서를 요청할 때 지정한 기본 루트 도메인으로 바꿉니다.

예:

```
sudo ln -sf /opt/bitnami/letsencrypt/certificates/example.com.key /etc/gitlab/ssl/
tls.key && \
sudo ln -sf /opt/bitnami/letsencrypt/certificates/example.com.crt /etc/gitlab/ssl/
tls.crt
```

12. 다음 명령을 입력하여 새 Let's Encrypt 인증서의 권한을 이동한 디렉터리에서 변경합니다.

```
sudo chown root:root /etc/gitlab/ssl/tls* && \
sudo chmod 600 /etc/gitlab/ssl/tls*
```

13. 다음 명령을 입력하여 GitLab CE 인스턴스에서 애플리케이션 서비스를 다시 시작합니다.

```
sudo service bitnami start && \
sudo service gitlab-runsvdir start
```

다음에 구성한 도메인을 사용하여 GitLab CE 웹 사이트로 이동하면 HTTPS 연결로 리디렉션됩니다. GitLab CE 인스턴스가 새 인증서를 인식하는 데 1시간까지 걸릴 수 있습니다. GitLab CE 웹 사이트에서 연결을 거부하면 인스턴스를 중지했다가 시작하고 다시 시도합니다.

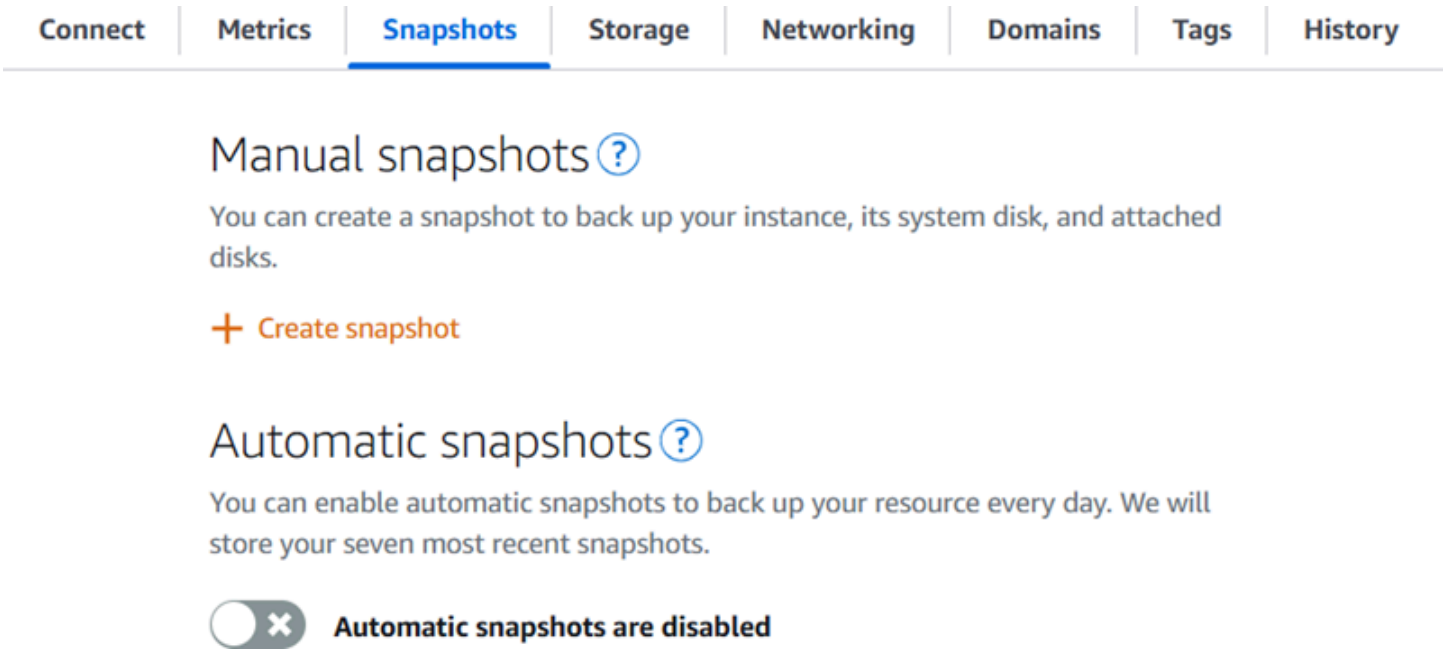
7단계: GitLab CE 설명서 읽기 및 웹 사이트 구성 계속

GitLab CE 설명서를 읽고 웹 사이트를 관리하고 사용자 지정하는 방법을 알아보세요. 자세한 내용을 알아보려면 GitLab 설명서를 참조하세요.

8단계: 인스턴스의 스냅샷 생성

원하는 방식으로 GitLab CE 웹 사이트를 구성한 후 인스턴스의 주기적 스냅샷을 생성하여 백업합니다. 스냅샷을 수동으로 생성하거나 자동 스냅샷을 활성화하여 Lightsail에서 매일 스냅샷을 생성하도록 할 수 있습니다. 인스턴스에 문제가 있는 경우 스냅샷을 사용하여 새 교체 인스턴스를 생성할 수 있습니다. 자세한 내용은 스냅샷을 참조하세요.

인스턴스 관리 페이지의 스냅샷(Snapshot) 탭에서 스냅샷 생성(Create a snapshot)을 선택하여 자동 스냅샷을 활성화합니다.



자세한 내용은 Amazon Lightsail에서 Linux 또는 Unix 인스턴스의 스냅샷 생성 또는 Amazon Lightsail에서 인스턴스 또는 디스크의 자동 스냅샷 활성화 또는 비활성화를 참조하세요.

Lightsail에서 Joomla! 시작하기

> 알고 계셨나요?
>
> Lightsail는 일별 스냅샷 7개를 저장하고 인스턴스에 대해 자동 스냅샷을 활성화하면 가장 오래된 스냅샷을 최신 스냅샷으로 자동 대체합니다. 자세한 내용은 Lightsail 인스턴스 및 디스크의 자동 스냅샷 구성을 참조하세요.

다음은 Amazon Lightsail에 Joomla! 인스턴스를 설치하고 실행한 후 시작해야 할 몇 가지 단계입니다.

목차

- 1단계: Bitnami 설명서 읽기
- 2단계: Joomla! 제어판에 액세스하기 위한 기본 애플리케이션 암호 가져오기
- 3단계: 인스턴스에 고정 IP 주소 연결
- 4단계: Joomla! 웹 사이트의 제어판에 로그인
- 5단계: Joomla! 웹 사이트로 등록된 도메인 이름의 트래픽 라우팅
- 6단계: Joomla! 웹 사이트에 대해 HTTPS 구성
- 7단계: Joomla! 설명서 읽기 및 웹 사이트 구성 계속
- 8단계: 인스턴스의 스냅샷 생성

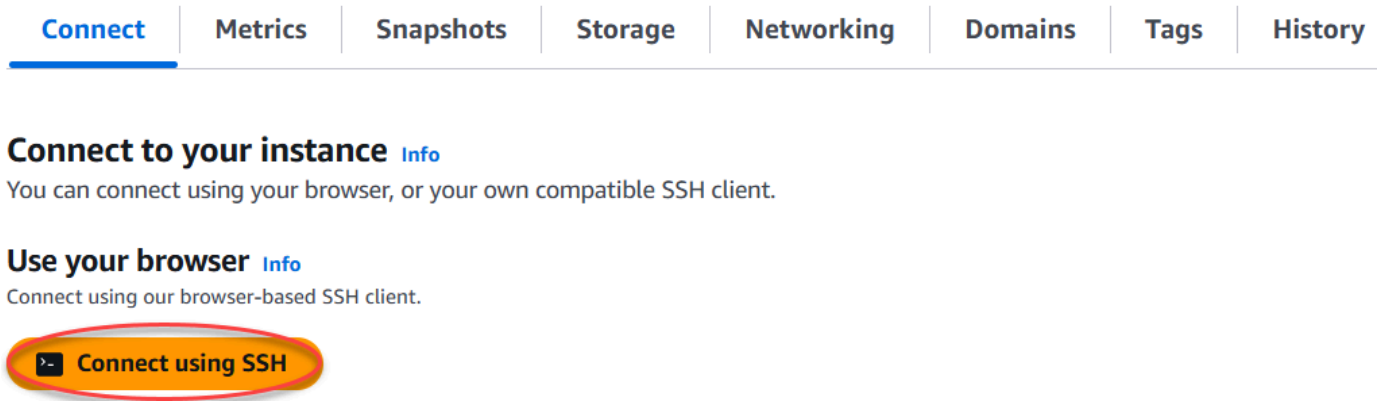
1단계: Bitnami 설명서 읽기

Bitnami 설명서를 읽고 Joomla! 애플리케이션을 구성하는 방법을 알아보세요. 자세한 내용은 Bitnami에서 패키징했습니다 AWS 클라우드.

2단계: Joomla! 제어판에 액세스하기 위한 기본 애플리케이션 암호 가져오기

Joomla! 웹 사이트의 제어판에 액세스하는 데 필요한 기본 애플리케이션 암호를 가져오려면 다음 절차를 완료하세요. 자세한 내용은 Amazon Lightsail의 Bitnami 인스턴스에 대한 애플리케이션 사용자 이름과 암호 가져오기를 참조하세요.

1. 인스턴스 관리 페이지의 연결(Connect) 탭에서 SSH로 연결(Connect using SSH)을 선택합니다.



2. 연결한 후 다음 명령을 입력하여 애플리케이션 암호를 가져옵니다.

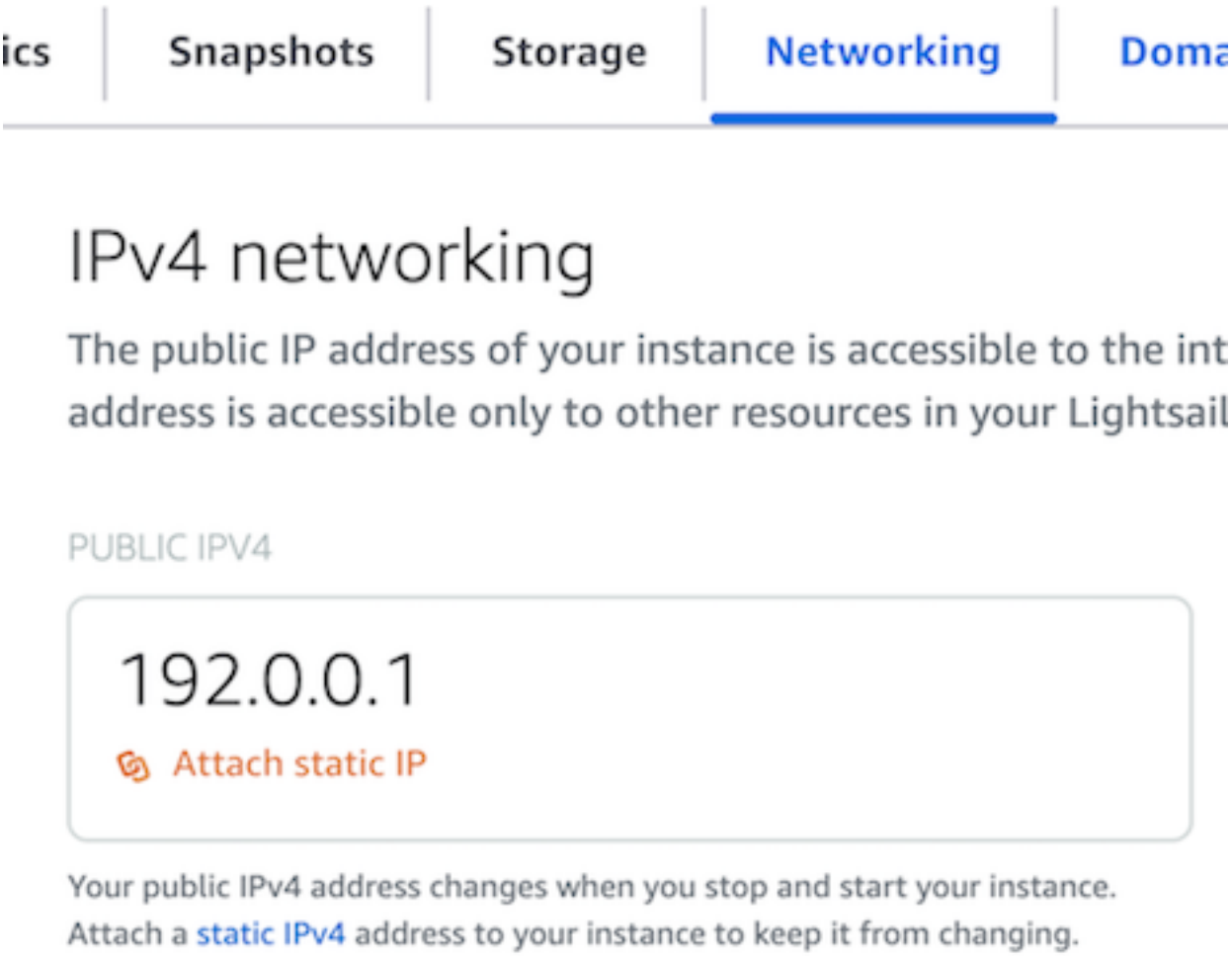
```
cat $HOME/bitnami_application_password
```

기본 애플리케이션 암호가 포함된 다음 예제와 유사한 응답이 표시되어야 합니다.

3단계: 인스턴스에 고정 IP 주소 연결

인스턴스에 연결된 기본 동적 퍼블릭 IP 주소는 인스턴스를 중지하고 시작할 때마다 변경됩니다. 퍼블릭 IP 주소가 변경되지 않도록 고정 IP 주소를 생성하고 인스턴스에 연결할 수 있습니다. 나중에 인스턴스에 도메인 이름을 사용하면 인스턴스를 중지하고 시작할 때마다 도메인의 DNS 레코드를 업데이트할 필요가 없습니다. 각 인스턴스에 1개의 고정 IP 주소만 연결할 수 있습니다.

인스턴스 관리 페이지의 네트워킹 탭에서 고정 IP 생성 또는 고정 IP 연결을 선택합니다. 인스턴스에 연결할 수 있는 고정 IP를 생성해 둔 경우에는 페이지의 지침을 따릅니다. 자세한 내용은 고정 IP를 생성하여 인스턴스에 연결을 참조하세요.



4단계: Joomla! 웹 사이트의 제어판에 로그인

이제 기본 애플리케이션 암호가 있으므로 다음 절차를 완료하여 Joomla! 웹 사이트의 홈 페이지로 이동하고 제어판에 로그인합니다. 로그인한 후 웹 사이트 사용자 지정 및 관리 변경을 시작할 수 있습니다. Joomla!에서 수행할 수 있는 작업에 대한 자세한 내용을 알아보려면 이 가이드 뒷부분의 7단계: Joomla! 설명서 읽기 및 웹 사이트 구성 계속 섹션을 참조하세요.

1. 인스턴스 관리 페이지의 연결(Connect) 탭에서 인스턴스의 퍼블릭 IP 주소를 기록해 둡니다. 퍼블릭 IP 주소는 인스턴스 관리 페이지의 헤더 섹션에도 표시됩니다.

2. 인스턴스의 퍼블릭 IP 주소로 이동합니다(예: `http://203.0.113.0`으로 이동).

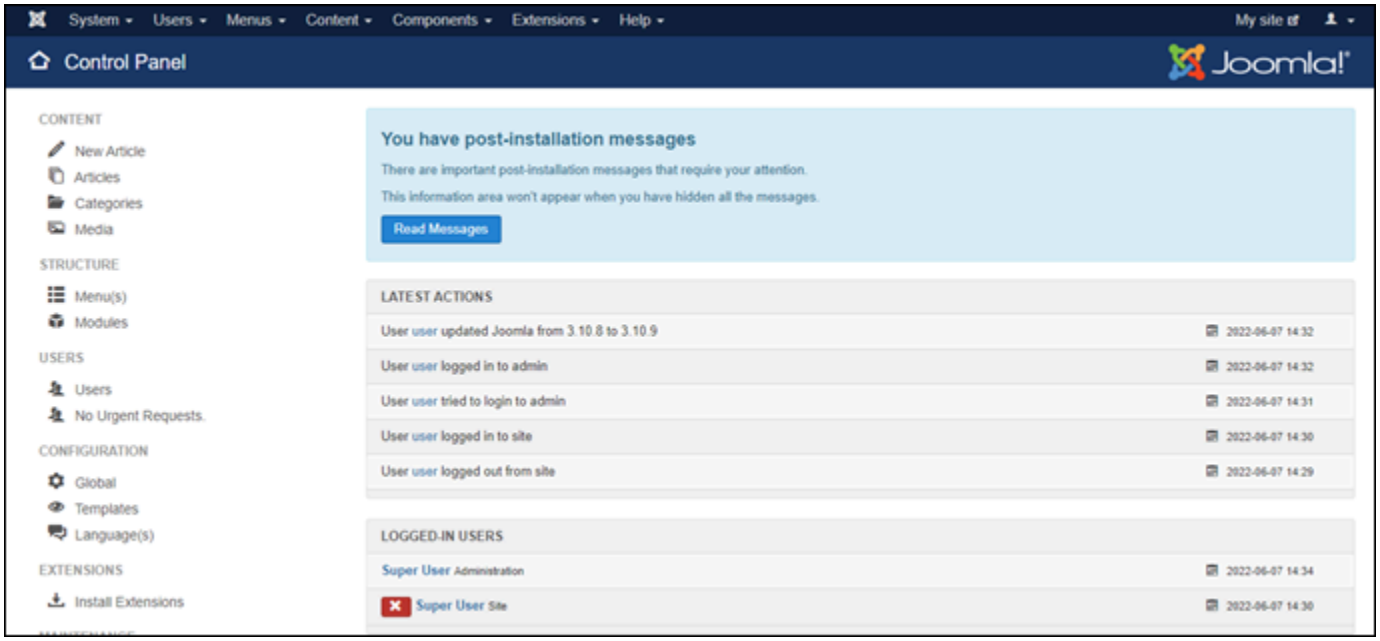
Joomla! 웹 사이트의 홈 페이지가 나타납니다.

3. Joomla! 웹 사이트 홈 페이지의 오른쪽 하단 모서리에 있는 관리(Manage)를 선택합니다.

   관리(Manage) 배너가 표시되지 않는 경우 `http://<PublicIP>/administrator/`을 통해 로그인 페이지로 이동할 수 있습니다. *<PublicIP>*을(를) 인스턴스의 퍼블릭 IP 주소로 바꿉니다.

4. 이 가이드의 초반부에서 검색한 기본 사용자 이름(`user`) 및 기본 암호를 사용하여 로그인합니다.

   Joomla! 관리 제어판이 나타납니다.



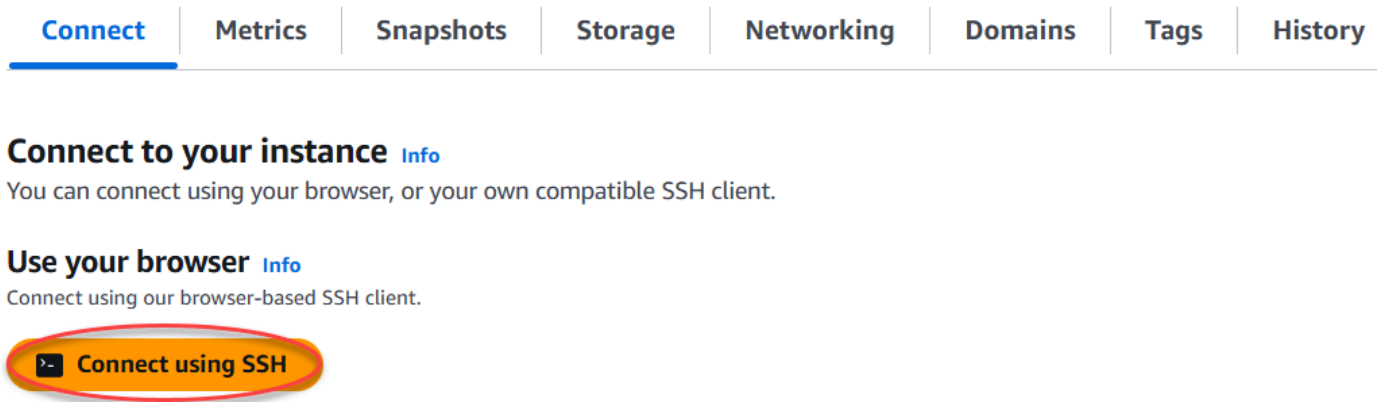
5단계: Joomla! 웹 사이트로 등록된 도메인 이름의 트래픽 라우팅

등록된 도메인 이름(예: `example.com`)의 트래픽을 Joomla! 웹 사이트로 라우팅하려면 도메인의 도메인 이름 시스템(DNS)에 레코드를 추가하면 됩니다. DNS 레코드는 일반적으로 도메인을 등록한 등록 대행자가 관리 및 호스팅합니다. 그러나 Lightsail 콘솔을 사용하여 관리할 수 있도록 도메인의 DNS 레코드 관리를 Lightsail로 이전하는 것이 좋습니다.

Lightsail 콘솔 홈 페이지의 도메인 및 DNS(Domains & DNS) 탭에서 DNS 영역 생성(Create DNS zone)을 선택하고 페이지의 지침에 따릅니다. 자세한 내용은 DNS 영역을 생성하여 Lightsail에서 도메인의 DNS 레코드 관리를 참조하세요.

도메인 이름이 트래픽을 인스턴스로 라우팅한 후 Joomla! 소프트웨어가 도메인 이름을 인식하도록 하려면 다음 단계를 완료해야 합니다.

1. 인스턴스 관리 페이지의 연결(Connect) 탭에서 SSH를 사용하여 연결(Connect using SSH)을 선택합니다.



2. Bitnami는 많은 청사진에 대한 파일 구조를 수정하는 중입니다. 이 절차의 파일 경로는 Bitnami 청사진이 네이티브 Linux 시스템 패키지(접근법 A)를 사용하는지 아니면 자체 포함 설치(접근법 B)인지에 따라 달라질 수 있습니다. Bitnami 설치 유형과 따라야 할 접근 방식을 식별하려면 연결된 후 다음 명령을 실행하세요.

```
test ! -f "/opt/bitnami/common/bin/openssl" && echo "Approach A: Using system
 packages." || echo "Approach B: Self-contained installation."
```

3. 이전 명령의 결과에 접근 방식 A를 사용해야 한다고 표시되면 다음 단계를 완료하고, 그렇지 않고 이전 명령의 결과에 접근 방식 B를 사용해야 한다고 표시되면 4단계로 계속하세요.

   1. 다음 명령을 입력하여 Vim을 사용하여 Apache 가상 호스트 구성 파일을 열고 도메인 이름에 대한 가상 호스트를 생성합니다.

      ```
      sudo vim /opt/bitnami/apache2/conf/vhosts/joomla-vhost.conf
      ```

   2. I를 눌러 Vim을 삽입 모드를 설정합니다.
   3. 다음 예제와 같이 파일에 도메인 이름을 추가합니다. 이 예제에서는 `example.com` 및 `www.example.com` 도메인을 사용합니다.

```
<VirtualHost 127.0.0.1:80 _default_:80>
  ServerName www.example.com
  ServerAlias example.com
  DocumentRoot /opt/bitnami/joomla
  <Directory "/opt/bitnami/joomla">
    Options -Indexes +FollowSymLinks -MultiViews
    AllowOverride None
    Require all granted
  </Directory>
  Include "/opt/bitnami/apache/conf/vhosts/htaccess/joomla-htaccess.conf"
</VirtualHost>
```

4. Esc 키를 누른 다음 `:wq!`를 입력하여 편집 내용을 저장(작성)하고 Vim을 종료합니다.
5. 다음 명령을 입력하여 Apache 서버를 다시 시작합니다.

```
sudo /opt/bitnami/ctlscript.sh restart apache
```

4. 이전 명령의 결과에 접근 방식 B를 사용해야 한다고 표시되면 다음 단계를 완료하세요.

1. 다음 명령을 입력하여 Vim을 사용하여 Apache 가상 호스트 구성 파일을 열고 도메인 이름에 대한 가상 호스트를 생성합니다.

```
sudo vim /opt/bitnami/apps/joomla/conf/httpd-vhosts.conf
```

2. I를 눌러 Vim을 삽입 모드를 설정합니다.
3. 다음 예제와 같이 파일에 도메인 이름을 추가합니다. 이 예제에서는 `example.com` 및 `www.example.com` 도메인을 사용합니다.

```
<VirtualHost *:80>
  ServerName example.com
  ServerAlias www.example.com
  ...
```

4. Esc 키를 누른 다음 `:wq!`를 입력하여 편집 내용을 저장(작성)하고 Vim을 종료합니다.
5. 다음 명령을 입력하여 `bitnami-apps-vhosts.conf` 파일에 Joomla!용 `httpd-vhosts.conf` 파일이 포함되어 있는지 확인합니다.

```
sudo vim /opt/bitnami/apache2/conf/bitnami/bitnami-apps-vhosts.conf
```

파일에서 다음 줄을 찾습니다. 누락된 경우 추가합니다.

```
Include "/opt/bitnami/apps/joomla/conf/httpd-vhosts.conf"
```

6. 다음 명령을 입력하여 Apache 서버를 다시 시작합니다.

```
sudo /opt/bitnami/ctlscript.sh restart apache
```

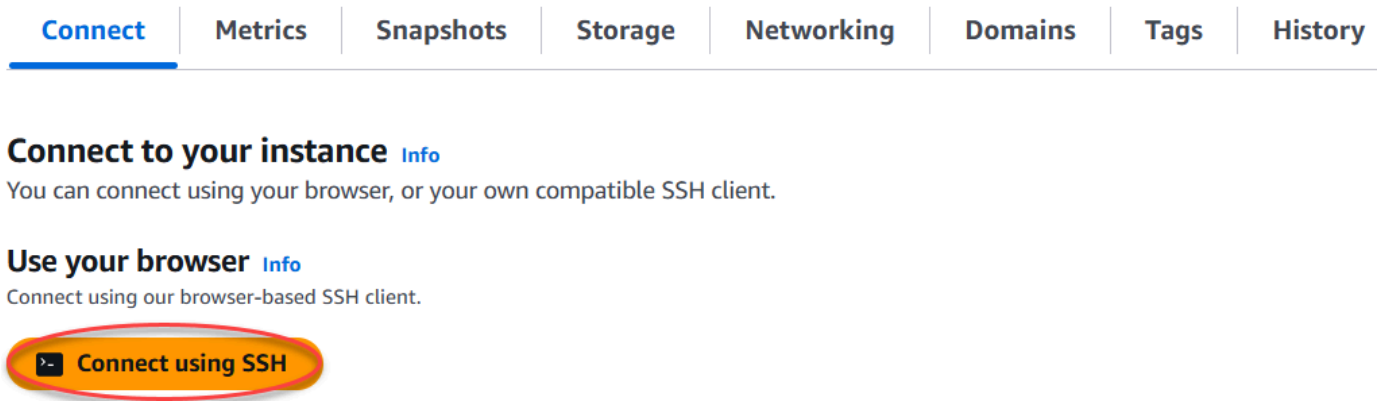
인스턴스에 대해 구성한 도메인 이름으로 이동하면 Joomla! 웹 사이트의 홈 페이지로 리디렉션됩니다. 다음으로 Joomla! 웹 사이트에 대한 HTTPS 연결을 활성화하기 위해 SSL/TLS 인증서를 생성하고 구성해야 합니다. 자세한 내용을 보려면 이 가이드의 다음 6단계: Joomla! 웹 사이트에 대해 HTTPS 구성 섹션으로 계속하세요.

6단계: Joomla! 웹 사이트에 대해 HTTPS 구성

Joomla! 웹 사이트에서 HTTPS를 구성하려면 다음 절차를 완료하세요. 이 단계에서는 Let's Encrypt SSL/TLS 인증서를 요청하기 위한 명령줄 도구인 Bitnami HTTPS Configuration Tool(`bncert-tool`)을 사용하는 방법을 보여줍니다. 자세한 내용을 알아보려면 Bitnami 설명서의 Learn About The Bitnami HTTPS Configuration Tool(Bitnami Configuration Tool에 대해 알아보기)을 참조하세요.

> **Important**
>
> 이 절차를 시작하기 전에 Joomla! 인스턴스로 트래픽을 라우팅하도록 도메인을 구성했는지 확인합니다. 그렇지 않으면 SSL/TLS 인증서 검증 프로세스가 실패합니다.

1. 인스턴스 관리 페이지의 연결(Connect) 탭에서 SSH로 연결(Connect using SSH)을 선택합니다.



2. 연결한 후 다음 명령을 입력하여 bncert 도구가 인스턴스에 설치되었는지 확인합니다.

```
sudo /opt/bitnami/bncert-tool
```

다음과 같은 응답 중 하나가 표시됩니다.

- 응답에 명령을 찾을 수 없음(command not found)이 표시되면 bncert 도구가 인스턴스에 설치되지 않은 것입니다. 이 절차의 다음 단계를 계속 진행하여 인스턴스에 bncert 도구를 설치합니다.
- 응답에 Bitnami HTTPS 구성 도구 시작(Welcome to the Bitnami HTTPS configuration tool)이 표시되면 bncert 도구가 인스턴스에 설치된 것입니다. 이 절차의 8단계로 계속합니다.
- bncert 도구가 일시적으로 인스턴스에 설치된 경우 업데이트된 버전의 도구를 사용할 수 있다는 메시지가 표시될 수 있습니다. 다운로드하도록 선택하고 `sudo /opt/bitnami/bncert-tool` 명령을 입력하여 bncert 도구를 다시 실행합니다. 이 절차의 8단계로 계속합니다.

3. 다음 명령을 입력하여 bncert 실행 파일을 인스턴스로 다운로드합니다.

```
wget -O bncert-linux-x64.run https://downloads.bitnami.com/files/bncert/latest/
bncert-linux-x64.run
```

4. 다음 명령을 입력하여 인스턴스에서 bncert 도구 실행 파일의 디렉터리를 생성합니다.

```
sudo mkdir /opt/bitnami/bncert
```

5. 다음 명령을 입력하여 bncert에서 프로그램으로 실행할 수 있는 파일을 실행하도록 합니다.

```
sudo chmod +x /opt/bitnami/bncert/bncert-linux-x64.run
```

6. 다음 명령을 입력하여 sudo /opt/bitnami/bncert-tool 명령을 입력할 때 bncert 도구를 실행하는 심볼 링크를 생성합니다.

```
sudo ln -s /opt/bitnami/bncert/bncert-linux-x64.run /opt/bitnami/bncert-tool
```

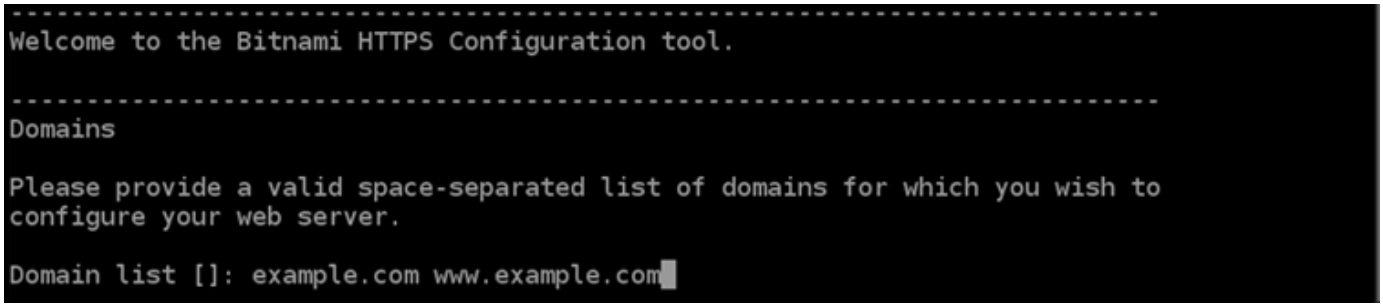
이제 인스턴스에 bncert 도구를 설치했습니다.

7. 다음 명령을 입력하여 bncert 도구를 실행합니다.

```
sudo /opt/bitnami/bncert-tool
```

8. 다음 예와 같이 기본 도메인 이름과 대체 도메인 이름을 공백으로 구분하여 입력합니다.

도메인이 트래픽을 인스턴스의 퍼블릭 IP 주소로 라우팅하도록 구성되지 않은 경우, `bncert` 도구에서 계속하기 전에 해당 구성을 설정하라는 메시지를 표시합니다. 도메인은 `bncert` 도구를 사용하여 인스턴스에서 HTTPS를 활성화한 인스턴스의 퍼블릭 IP 주소로 트래픽을 라우팅해야 합니다. 이렇게 해야 도메인을 소유하고 있음을 확인하고 인증서를 검증하는 역할을 할 수 있습니다.



9. bncert 도구는 웹 사이트의 리디렉션을 어떻게 구성할지 묻는 메시지를 표시합니다. 사용할 수 있는 옵션은 다음과 같습니다.

   - HTTP에서 HTTPS로 리디렉션 활성화(Enable HTTP to HTTPS redirection) - HTTP 버전 웹 사이트(예: http:/example.com)를 방문하는 사용자를 HTTPS 버전(예: https://example.com)으로 자동 리디렉션할지 지정합니다. 모든 방문자가 암호화된 연결을 사용하도록 강제하기 때문에 이 옵션을 활성화하는 것이 좋습니다. Y를 입력하고 Enter 키를 눌러 활성화합니다.
   - 비 www에서 www로 리디렉션 활성화(Enable non-www to www redirection) - 도메인의 정점(https://example.com)을 방문하는 사용자를 도메인의 www 하위 도메인(https://www.example.com)으로 자동 리디렉션할지 지정합니다. 이 옵션은 활성화하는 것이 좋습니다. 그러나 Google의 웹 마스터 도구와 같은 검색 엔진 도구에서 도메인의 정점을 기본 웹 사이트 주소로 지정했거나 정점이 IP를 직접 가리키고 www 하위 도메인이 CNAME 레코드를 통해 정점을 참조하는 경우, 이 옵션을 비활성화하고 대체 옵션(www에서 비 www로 리디렉션 활성화)을 활성화할 수 있습니다. Y를 입력하고 Enter 키를 눌러 활성화합니다.
   - www에서 비 www로 리디렉션 활성화(Enable www to non-www redirection) - 도메인의 www 하위 도메인(https://www.example.com)을 방문하는 사용자를 도메인의 정점(https://example.com)으로 자동 리디렉션할지 지정합니다. 비 www에서 www로 리디렉션을 활성화한 경우 이 옵션을 비활성화하는 것이 좋습니다. N를 입력하고 Enter 키를 눌러 비활성화합니다.

   선택한 내용은 다음 예와 같아야 합니다.

```
Enable/disable redirections

Please select the redirections you wish to enable or disable on your Bitnami
installation.



Enable HTTP to HTTPS redirection [Y/n]: Y



Enable non-www to www redirection [Y/n]: Y



Enable www to non-www redirection [y/N]: N
```

10. 적용될 변경 사항의 목록이 나열됩니다. Y를 입력하고 Enter 키를 눌러 확인하고 계속합니다.

```
Changes to perform

The following changes will be performed to your Bitnami installation:

1. Stop web server
2. Configure web server to use a free Let's Encrypt certificate for the domains:
example.com www.example.com
3. Configure a cron job to automatically renew the certificate each month
4. Configure web server name to: example.com
5. Enable HTTP to HTTPS redirection (example: redirect http://example.com to
https://example.com)
6. Enable non-www to www redirection (example: redirect example.com to
 www.example.com)
7. Start web server once all changes have been performed



Do you agree to these changes? [Y/n]: Y
```

11. Let's Encrypt 인증서에 연결할 이메일 주소를 입력하고 Enter 키를 누릅니다.

```
Create a free HTTPS certificate with Let's Encrypt

Please provide a valid e-mail address for which to associate your Let's Encrypt
certificate.

Domain list: example.com www.example.com

Server name: example.com

E-mail address []:
```

12. Let's Encrypt 구독자 계약을 검토합니다. Y를 입력하고 Enter 키를 눌러 계약을 수락하고 계속합니다.

```
The Let's Encrypt Subscriber Agreement can be found at:

https://letsencrypt.org/documents/LE-SA-v1.2-November-15-2017.pdf

Do you agree to the Let's Encrypt Subscriber Agreement? [Y/n]:
```

인증서 요청 및 지정한 리디렉션 구성을 비롯하여 인스턴스에서 HTTPS를 활성화하는 작업이 수행됩니다.

```
Performing changes to your installation

The Bitnami HTTPS Configuration Tool will perform any necessary actions to your
Bitnami installation. This may take some time, please be patient.
```

인증서가 성공적으로 발급되고 검증됩니다. 다음 예와 유사한 메시지가 표시되면 인스턴스에서 리디렉션이 성공적으로 구성된 것입니다.

```
Success

The Bitnami HTTPS Configuration Tool succeeded in modifying your installation.

The configuration report is shown below.

Backup files:
* /opt/bitnami/apache2/conf/httpd.conf.back.202005290035
* /opt/bitnami/apache2/conf/bitnami/bitnami-apps-prefix.conf.back.202005290035
* /opt/bitnami/apache2/conf/bitnami/bitnami.conf.back.202005290035

Find more details in the log file:

/tmp/bncert-202005290035.log

If you find any issues, please check Bitnami Support forums at:

https://community.bitnami.com

Press [Enter] to continue:
```

`bncert` 도구는 인증서가 만료되기 80일 전마다 인증서를 자동으로 갱신합니다. 인스턴스와 함께 추가 도메인 및 하위 도메인을 사용하고 이러한 도메인에서 HTTPS를 활성화하려면 위의 단계를 반복합니다.

이제 Joomla! 인스턴스에서 HTTPS가 활성화되었습니다. 다음에 구성한 도메인을 사용하여 Joomla! 웹 사이트로 이동하면 HTTPS 연결로 리디렉션됩니다.

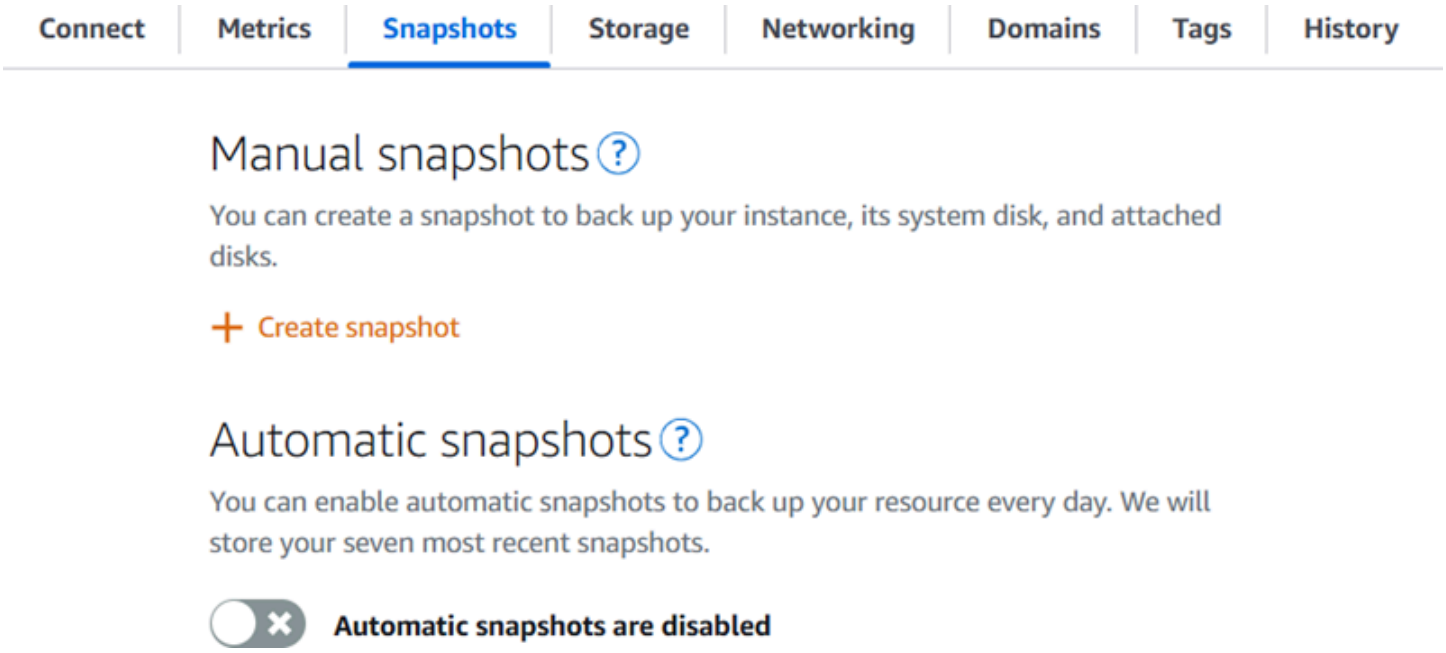
7단계: Joomla! 설명서 읽기 및 웹 사이트 구성 계속

Joomla! 설명서를 읽고 웹 사이트를 관리하고 사용자 지정하는 방법을 알아보세요. 자세한 내용을 알아보려면 Joomla! 설명서를 참조하세요.

8단계: 인스턴스의 스냅샷 생성

원하는 방식으로 Joomla! 웹 사이트를 구성한 후 인스턴스의 주기적 스냅샷을 생성하여 백업합니다. 스냅샷을 수동으로 생성하거나 자동 스냅샷을 활성화하여 Lightsail에서 매일 스냅샷을 생성하도록 할 수 있습니다. 인스턴스에 문제가 있는 경우 스냅샷을 사용하여 새 교체 인스턴스를 생성할 수 있습니다. 자세한 내용은 스냅샷을 참조하세요.

인스턴스 관리 페이지의 스냅샷(Snapshot) 탭에서 스냅샷 생성(Create a snapshot)을 선택하여 자동 스냅샷을 활성화합니다.

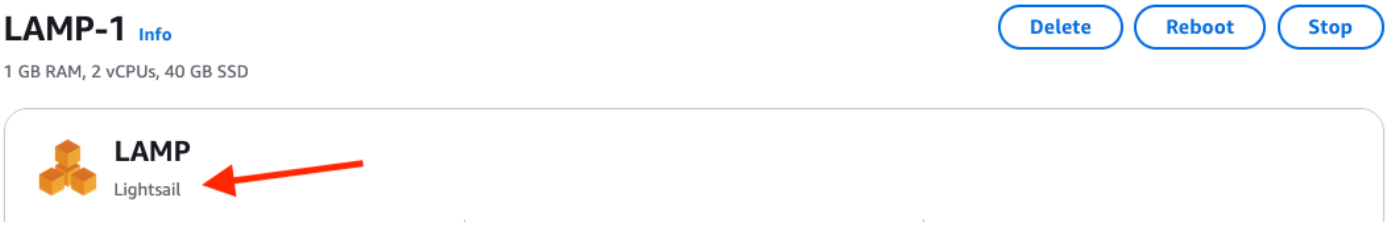


자세한 내용은 Amazon Lightsail에서 Linux 또는 Unix 인스턴스의 스냅샷 생성 또는 Amazon Lightsail에서 인스턴스 또는 디스크의 자동 스냅샷 활성화 또는 비활성화를 참조하세요.

에서 LAMP 스택 배포 및 관리 Lightsail

> 알고 계셨나요?
>
> Lightsail는 일별 스냅샷 7개를 저장하고 인스턴스에 대해 자동 스냅샷을 활성화하면 가장 오래된 스냅샷을 최신 스냅샷으로 자동 대체합니다. 자세한 내용은 Lightsail 인스턴스 및 디스크의 자동 스냅샷 구성을 참조하세요.

다음은에서 LAMP 인스턴스를 설치하고 실행한 후 시작하기 위해 수행해야 하는 몇 가지 단계입니다 Amazon Lightsail. 시작하기 전에 인스턴스 관리 페이지에서 블루프린트 공급업체를 식별합니다.



LAMP 인스턴스에 적합한 가이드를 선택합니다.

Bitnami

1단계: LAMP 인스턴스에 대한 기본 애플리케이션 암호 가져오기

인스턴스의 사전 설치된 애플리케이션 또는 서비스에 액세스하려면 기본 애플리케이션 암호가 필요합니다.

1. 인스턴스 관리 페이지의 연결(Connect) 탭에서 SSH로 연결(Connect using SSH)을 선택합니다.
2. 연결한 후 다음 명령을 입력하여 애플리케이션 암호를 가져옵니다.

```
cat bitnami_application_password
```

> **Note**
>
> 사용자 홈 디렉터리가 아닌 다른 디렉터리에 있는 경우 `cat $HOME/bitnami_application_password`를 입력합니다.

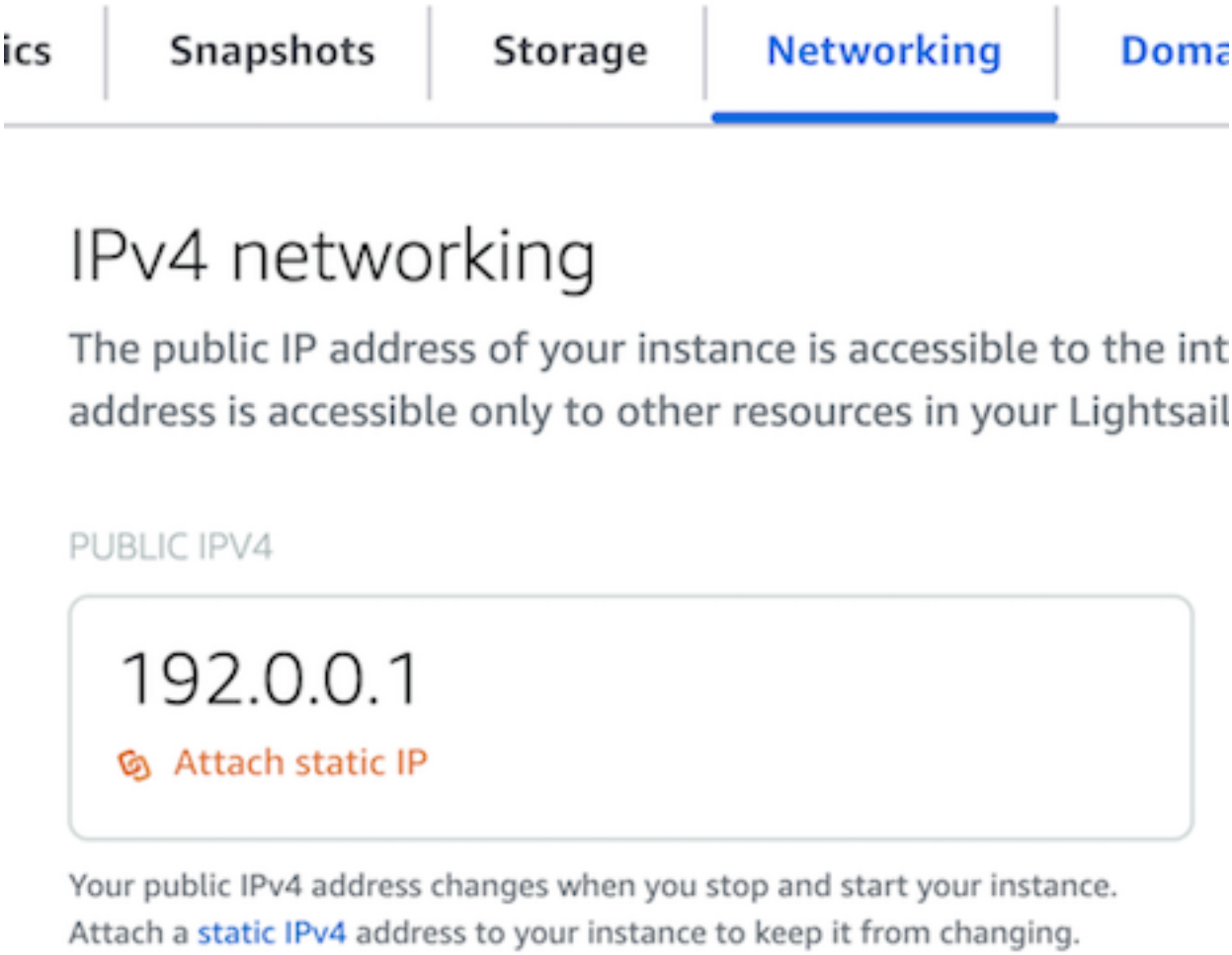
기본 애플리케이션 암호가 포함되어 있는 다음과 유사한 응답이 표시되어야 합니다.

자세한 내용은 Amazon Lightsail의 Bitnami 인스턴스에 대한 애플리케이션 사용자 이름과 암호 가져오기를 참조하세요.

2단계: 고정 IP 주소를 LAMP 인스턴스에 연결

인스턴스에 연결된 기본 동적 퍼블릭 IP 주소는 인스턴스를 중지하고 시작할 때마다 변경됩니다. 퍼블릭 IP 주소가 변경되지 않도록 고정 IP 주소를 생성하고 인스턴스에 연결할 수 있습니다. 나중에 인스턴스에 도메인 이름을 사용하면 인스턴스를 중지하고 시작할 때마다 도메인의 DNS 레코드를 업데이트할 필요가 없습니다. 각 인스턴스에 1개의 고정 IP 주소만 연결할 수 있습니다.

인스턴스 관리 페이지의 네트워킹 탭에서 고정 IP 생성 또는 고정 IP 연결을 선택합니다. 인스턴스에 연결할 수 있는 고정 IP를 생성해 둔 경우에는 페이지의 지침을 따릅니다. 자세한 내용은 고정 IP를 생성하여 인스턴스에 연결을 참조하세요.



3단계: LAMP 인스턴스 시작 페이지 방문

인스턴스의 퍼블릭 IP 주소로 이동하여 해당 주소에 설치된 애플리케이션, phpMyAdmin 또는 Bitnami 설명서에 액세스합니다.

1. 인스턴스 관리 페이지의 연결 탭에서 퍼블릭 IP를 기록해 둡니다.

2. 퍼블릭 IP 주소로 이동합니다(예: `http://192.0.2.3`으로 이동).

자세한 내용은 Amazon Lightsail의 Bitnami 인스턴스에 대한 애플리케이션 사용자 이름과 암호 가져오기를 참조하세요.

4단계: 도메인 이름을 LAMP 인스턴스에 매핑

`example.com`과 같은 도메인 이름을 인스턴스에 매핑하려면 도메인의 도메인 이름 시스템(DNS)에 레코드를 추가합니다. DNS 레코드는 일반적으로 도메인을 등록한 등록 대행자가 관리 및 호스팅합니다. 그러나 Lightsail 콘솔을 사용하여 관리할 수 있도록 도메인의 DNS 레코드 관리를 Lightsail로 이전하는 것이 좋습니다.

Lightsail 콘솔 홈 페이지의 네트워킹 탭에서 DNS zonelamp 생성을 선택한 다음 페이지의 지침을 따릅니다.

자세한 내용은 도메인의 DNS 레코드를 관리하기 위한 DNS 영역 생성을 참조하세요.

5단계: Bitnami 설명서 읽기

Bitnami 설명서를 읽고 애플리케이션을 배포하고, SSL 인증서로 HTTPS 지원을 활성화하고, SFTP로 서버에 파일을 업로드하는 방법 등을 알아봅니다.

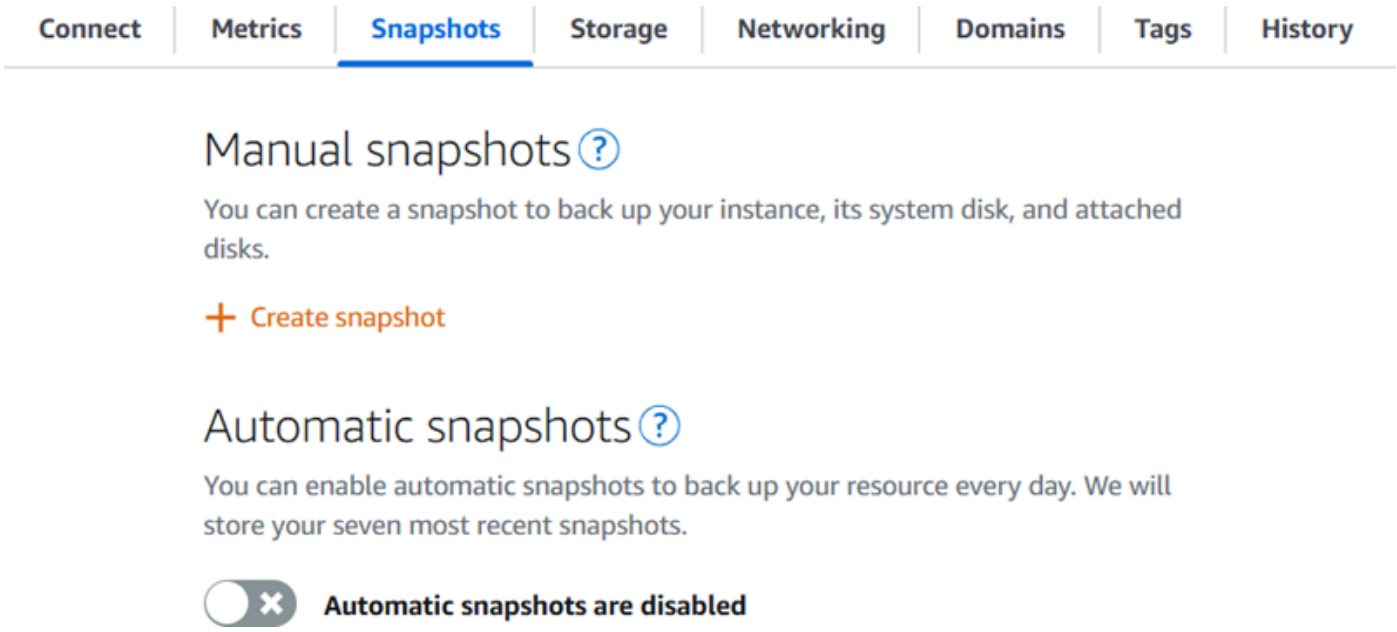
자세한 내용은 AWS 클라우드에 대한 Bitnami LAMP를 참조하세요.

6단계: LAMP 인스턴스의 스냅샷 생성

원하는 방식으로 웹 사이트를 구성한 후 인스턴스의 주기적 스냅샷을 생성하여 백업합니다. 스냅샷은 시스템 디스크의 복사본이며, 인스턴스의 원본 구성입니다. 스냅샷은 스냅샷을 생성한 시점부터 인스턴스를 복원하는 데 필요한 모든 데이터를 포함합니다.

스냅샷을 수동으로 생성하거나 자동 스냅샷을 활성화하여 Lightsail에서 매일 스냅샷을 생성하도록 할 수 있습니다. 인스턴스에 문제가 있는 경우 스냅샷을 사용하여 새 교체 인스턴스를 생성할 수 있습니다.

스냅샷 탭의 인스턴스 관리 페이지에서 스냅샷 작업을 할 수 있습니다. 자세한 내용은 Amazon Lightsail의 스냅샷을 참조하세요.



Lightsail

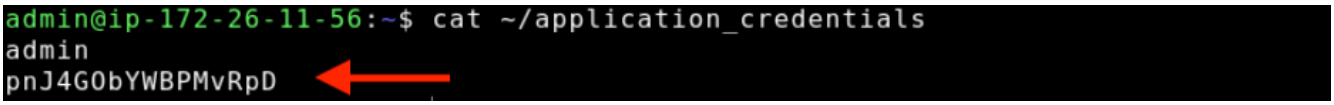
1단계: LAMP 인스턴스에 대한 기본 애플리케이션 암호 가져오기

인스턴스의 사전 설치된 애플리케이션 또는 서비스에 액세스하려면 기본 애플리케이션 암호가 필요합니다.

1. 인스턴스 관리 페이지의 연결(Connect) 탭에서 SSH로 연결(Connect using SSH)을 선택합니다.
2. 연결한 후 다음 명령을 입력하여 기본 애플리케이션 암호를 가져옵니다.

```
cat ~/application_credentials
```

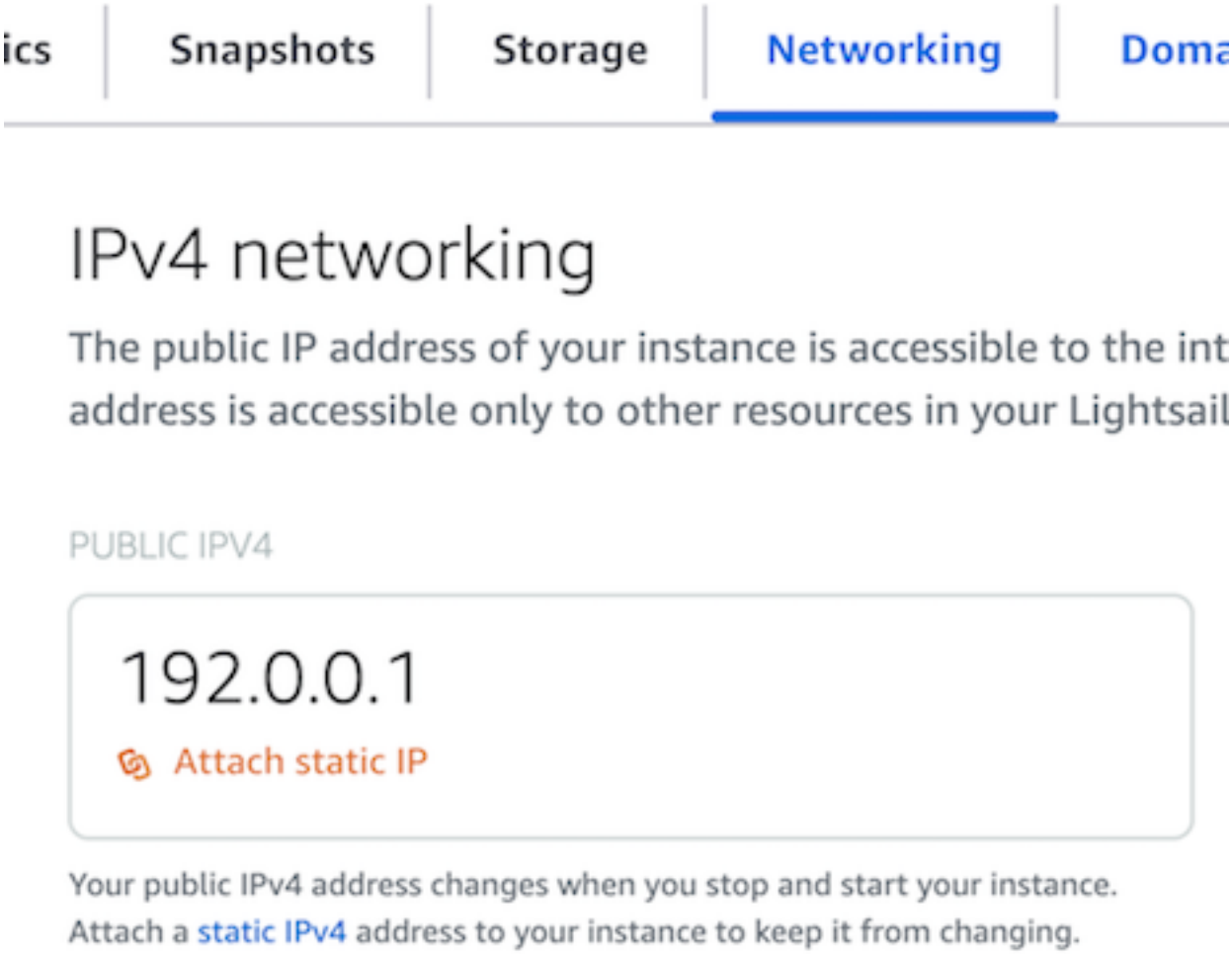
기본 애플리케이션 암호가 포함되어 있는 다음과 유사한 응답이 표시되어야 합니다.



2단계: 고정 IP 주소를 LAMP 인스턴스에 연결

인스턴스에 연결된 기본 동적 퍼블릭 IP 주소는 인스턴스를 중지하고 시작할 때마다 변경됩니다. 퍼블릭 IP 주소가 변경되지 않도록 고정 IP 주소를 생성하고 인스턴스에 연결할 수 있습니다. 나중에 인스턴스에 도메인 이름을 사용하면 인스턴스를 중지하고 시작할 때마다 도메인의 DNS 레코드를 업데이트할 필요가 없습니다. 각 인스턴스에 1개의 고정 IP 주소만 연결할 수 있습니다.

인스턴스 관리 페이지의 네트워킹 탭에서 고정 IP 생성 또는 고정 IP 연결을 선택합니다. 인스턴스에 연결할 수 있는 고정 IP를 생성해 둔 경우에는 페이지의 지침을 따릅니다. 자세한 내용은 고정 IP를 생성하여 인스턴스에 연결을 참조하세요.



3단계: LAMP 인스턴스 시작 페이지 방문

인스턴스의 고정 IP 주소로 이동하여 인스턴스에 설치된 애플리케이션에 액세스합니다.

1. 인스턴스 관리 페이지에서 고정 IP 주소를 복사합니다.

2. 예를 들어 정적 IP 주소를 브라우저 주소에 붙여 넣습니다`http://192.0.0.1`.

4단계: 도메인 이름을 LAMP 인스턴스에 매핑

`example.com`과 같은 도메인 이름을 인스턴스에 매핑하려면 도메인의 도메인 이름 시스템(DNS)에 레코드를 추가합니다. DNS 레코드는 일반적으로 도메인을 등록한 등록 대행자가 관리 및 호스팅합니다. 그러나 Lightsail 콘솔을 사용하여 관리할 수 있도록 도메인의 DNS 레코드 관리를 Lightsail로 이전하는 것이 좋습니다.

Lightsail 콘솔 홈 페이지의 네트워킹 탭에서 DNS 영역 생성을 선택하고 페이지의 지침에 따릅니다.

자세한 내용은 도메인의 DNS 레코드를 관리하기 위한 DNS 영역 생성을 참조하세요.

HTTPS를 활성화하려면 Let's Encrypt SSL/TLS를 사용하여 Lightsail LAMP 웹 사이트 보안을 참조하세요.

5단계: 애플리케이션 배포

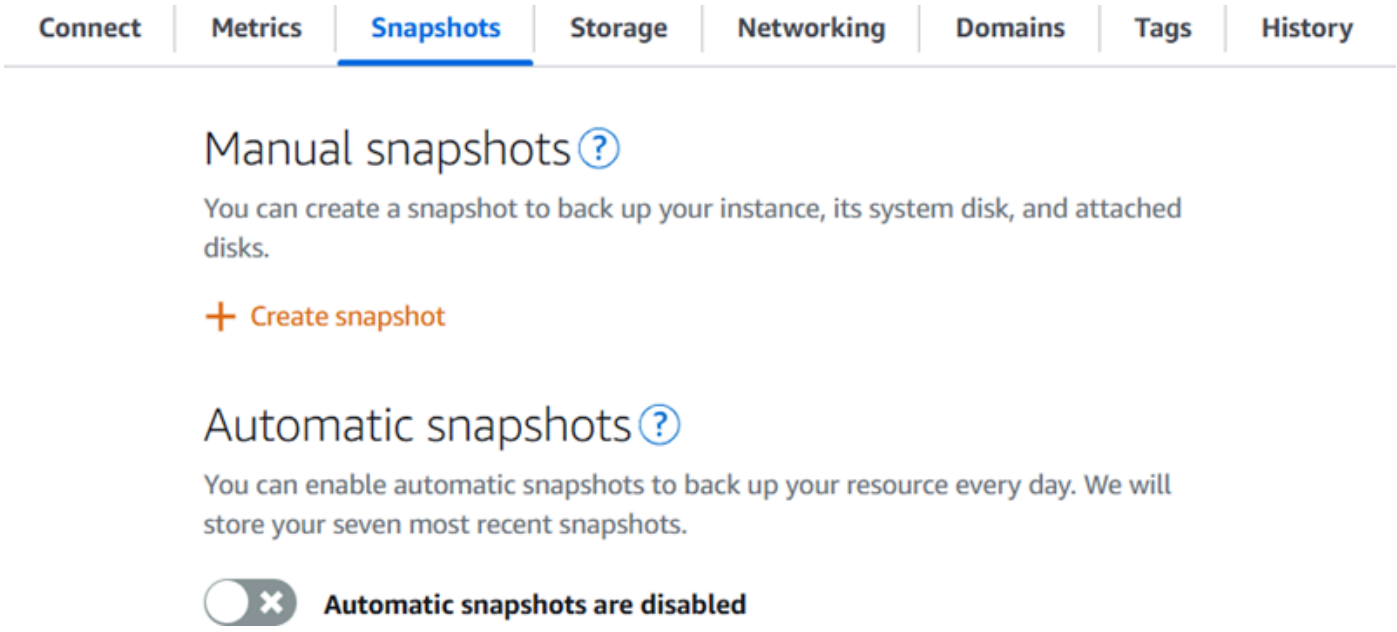
1. scp를 Lightsail 사용하여에 애플리케이션을 복사할 때 Linux 인스턴스 간에 파일 전송의 지침을 따릅니다. `/var/www/html`
2. 인스턴스 관리 페이지의 연결(Connect) 탭에서 SSH로 연결(Connect using SSH)을 선택합니다.
3. `sudo systemctl restart apache2` 실행
4. 인스턴스의 고정 IP 주소로 이동합니다.

6단계: LAMP 인스턴스의 스냅샷 생성

원하는 방식으로 웹 사이트를 구성한 후 인스턴스의 주기적 스냅샷을 생성하여 백업합니다. 스냅샷은 시스템 디스크의 복사본이며, 인스턴스의 원본 구성입니다. 스냅샷은 스냅샷을 생성한 시점부터 인스턴스를 복원하는 데 필요한 모든 데이터를 포함합니다.

스냅샷을 수동으로 생성하거나 자동 스냅샷을 활성화하여 Lightsail에서 매일 스냅샷을 생성하도록 할 수 있습니다. 인스턴스에 문제가 있는 경우 스냅샷을 사용하여 새 교체 인스턴스를 생성할 수 있습니다.

스냅샷 탭의 인스턴스 관리 페이지에서 스냅샷 작업을 할 수 있습니다. 자세한 내용은 Amazon Lightsail의 스냅샷을 참조하세요.



Lightsail에 Magento를 설정 및 구성

> 알고 계셨나요?
>
> Lightsail는 일별 스냅샷 7개를 저장하고 인스턴스에 대해 자동 스냅샷을 활성화하면 가장 오래된 스냅샷을 최신 스냅샷으로 자동 대체합니다. 자세한 내용은 Lightsail 인스턴스 및 디스크의 자동 스냅샷 구성을 참조하세요.

다음은 Amazon Lightsail에 Magento 인스턴스를 설치하고 실행한 후 시작하기 위해 완료해야 하는 몇 가지 단계입니다.

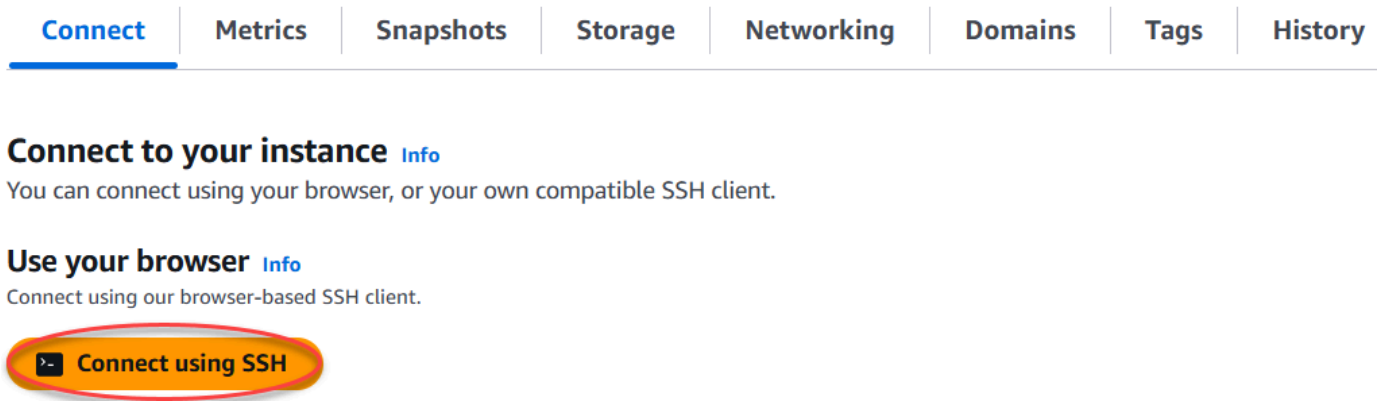
목차

- 1단계: Magento 웹 사이트의 기본 애플리케이션 암호 가져오기
- 2단계: Magento 인스턴스에 고정 IP 주소 연결
- 3단계: Magento 웹 사이트의 관리 대시보드에 로그인
- 4단계: Magento 웹 사이트로 등록된 도메인 이름의 트래픽 라우팅
- 5단계: Magento 웹 사이트에 대해 HTTPS 구성
- 6단계: 이메일 알림을 위한 SMTP 구성
- 7단계: Bitnami 및 Magento 설명서 읽기
- 8단계: Magento 인스턴스의 스냅샷 생성

1단계: Magento 웹 사이트의 기본 애플리케이션 암호 가져오기

Magento 웹 사이트의 기본 애플리케이션 암호를 가져오려면 다음 단계를 완료하세요. 자세한 내용은 Amazon Lightsail의 Bitnami 인스턴스에 대한 애플리케이션 사용자 이름과 암호 가져오기를 참조하세요.

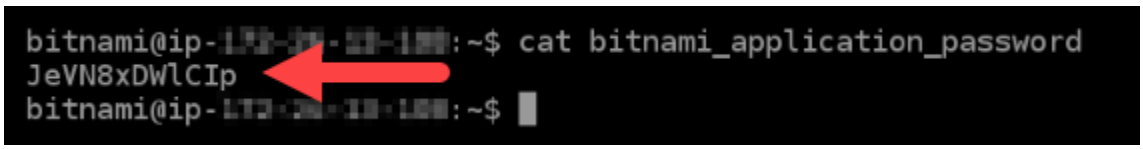
1. 인스턴스 관리 페이지의 연결(Connect) 탭에서 SSH를 사용하여 연결(Connect using SSH)을 선택합니다.



2. 연결한 후 다음 명령을 입력하여 기본 애플리케이션 암호를 가져옵니다.

```
cat $HOME/bitnami_application_password
```

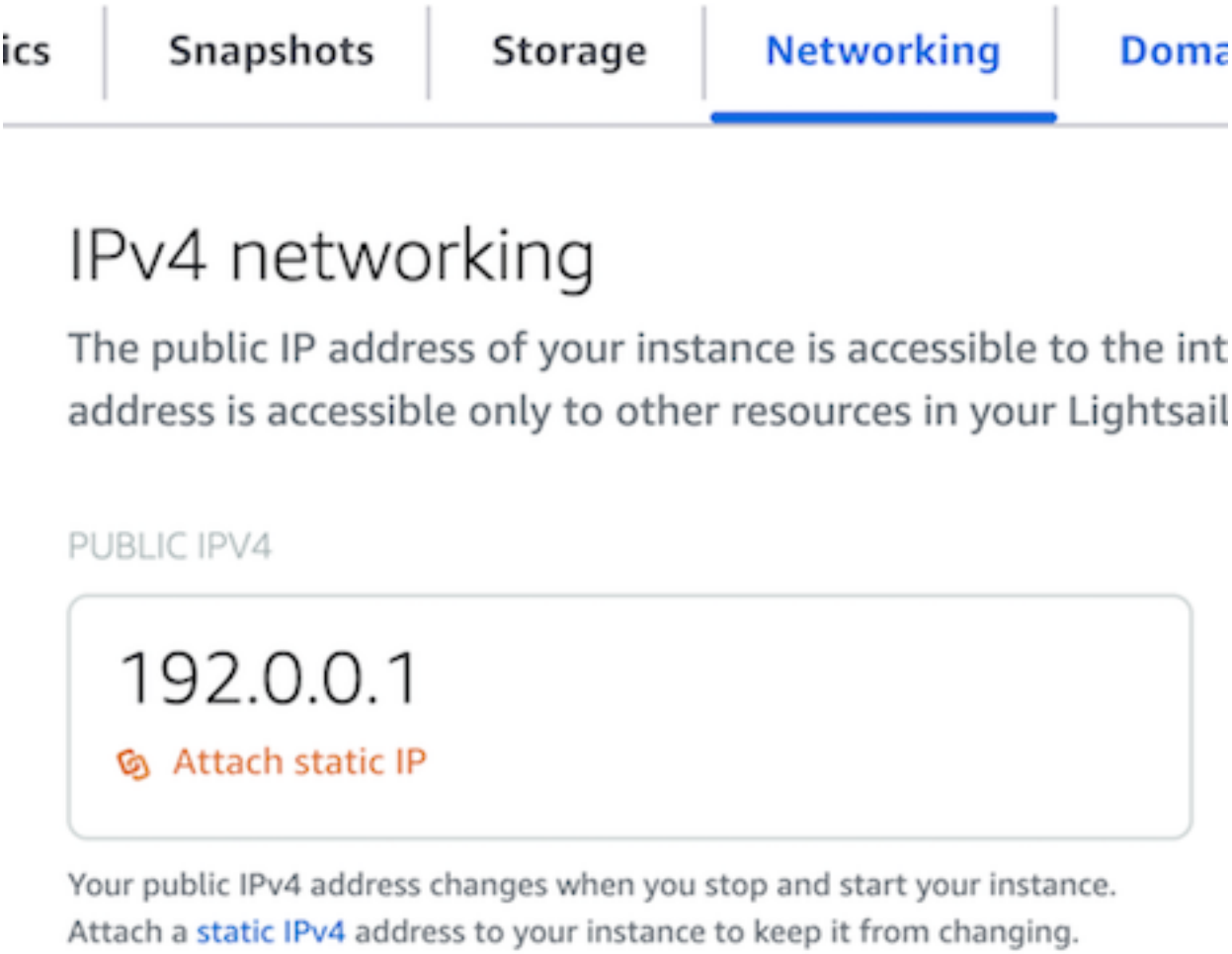
다음 예와 유사한 응답이 표시되며, 여기에 기본 애플리케이션 암호가 포함되어 있습니다. 암호를 안전한 위치에 저장합니다. 이 자습서의 다음 섹션에서 Magento 웹 사이트의 관리 대시보드에 로그인하는 데 이 암호를 사용하게 됩니다.



2단계: Magento 인스턴스에 고정 IP 주소 연결

인스턴스에 연결된 기본 동적 퍼블릭 IP 주소는 인스턴스를 중지하고 시작할 때마다 변경됩니다. 퍼블릭 IP 주소가 변경되지 않도록 고정 IP 주소를 생성하고 인스턴스에 연결할 수 있습니다. 나중에 인스턴스에 도메인 이름을 사용하면 인스턴스를 중지하고 시작할 때마다 도메인의 DNS 레코드를 업데이트할 필요가 없습니다. 각 인스턴스에 1개의 고정 IP 주소만 연결할 수 있습니다.

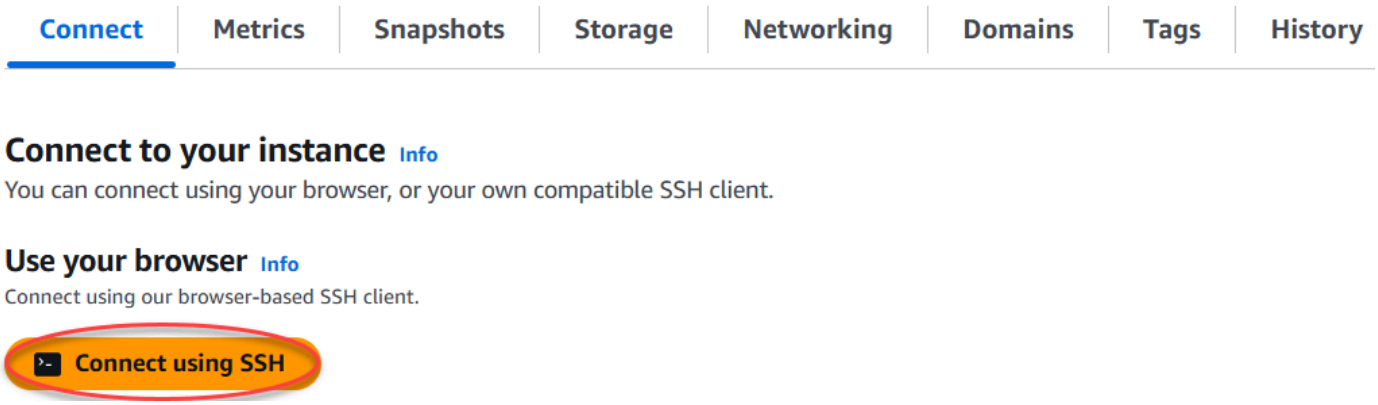
인스턴스 관리 페이지의 네트워킹 탭에서 고정 IP 생성 또는 고정 IP 연결을 선택합니다. 인스턴스에 연결할 수 있는 고정 IP를 생성해 둔 경우에는 페이지의 지침을 따릅니다. 자세한 내용은 고정 IP를 생성하여 인스턴스에 연결을 참조하세요.



새로운 고정 IP 주소가 인스턴스에 연결되면 Magento 소프트웨어가 새로운 고정 IP 주소를 인식하도록 다음 단계를 완료해야 합니다.

1. 인스턴스의 고정 IP 주소를 기록해 둡니다. 고정 IP 주소는 인스턴스 관리 페이지의 머리말 섹션에 나와 있습니다.

2. 인스턴스 관리 페이지의 연결(Connect) 탭에서 SSH를 사용하여 연결(Connect using SSH)을 선택합니다.



3. 연결한 후 다음 명령을 입력합니다. *<StaticIP>*를 인스턴스의 새로운 고정 IP 주소로 대체해야 합니다.

```
sudo /opt/bitnami/configure_app_domain --domain <StaticIP>
```

예:

```
sudo /opt/bitnami/configure_app_domain --domain 203.0.113.0
```

다음 예와 비슷한 응답이 나타나는 것을 볼 수 있습니다. Magento 소프트웨어에서 이제 새로운 고정 IP 주소를 인식합니다.

```
bitnami@ip-[redacted]:~$ sudo /opt/bitnami/configure_app_domain --domain 203.0.113.0
Configuring domain to 203.0.113.0
2021-03-12T15:49:22.000Z - info: Saving configuration info to disk
prestashop 15:49:22.41 INFO  ==> Trying to connect to the database server
prestashop 15:49:22.44 INFO  ==> Updating hostname in database
prestashop 15:49:22.46 INFO  ==> Purging cache
Disabling automatic domain update for IP address changes
```

> **Note**
>
> Magento는 현재 IPv6 주소를 지원하지 않습니다. 인스턴스에 IPv6를 활성화할 수 있지만, Magento 소프트웨어는 IPv6 네트워크를 통한 요청에 응답하지 않습니다.

3단계: Magento 웹 사이트의 관리 대시보드에 로그인

Magento 웹 사이트에 액세스하고 관리 대시보드에 로그인하려면 다음 단계를 완료하세요. 기본 사용자 이름(`user`)과 이 가이드의 앞부분에서 확인한 기본 애플리케이션 암호를 사용하여 로그인할 수 있습니다.

1. Lightsail 콘솔의 인스턴스 관리 페이지 머리말 영역에 나와 있는 퍼블릭 또는 고정 IP 주소를 기록해 둡니다.

2. 다음 주소를 검색하여 Magento 웹 사이트의 관리 대시보드 로그인 페이지에 액세스합니다. 여기서 *<InstanceIpAddress>*를 인스턴스의 퍼블릭 IP 주소 또는 고정 IP 주소로 대체해야 합니다.

```
http://<InstanceIpAddress>/admin
```

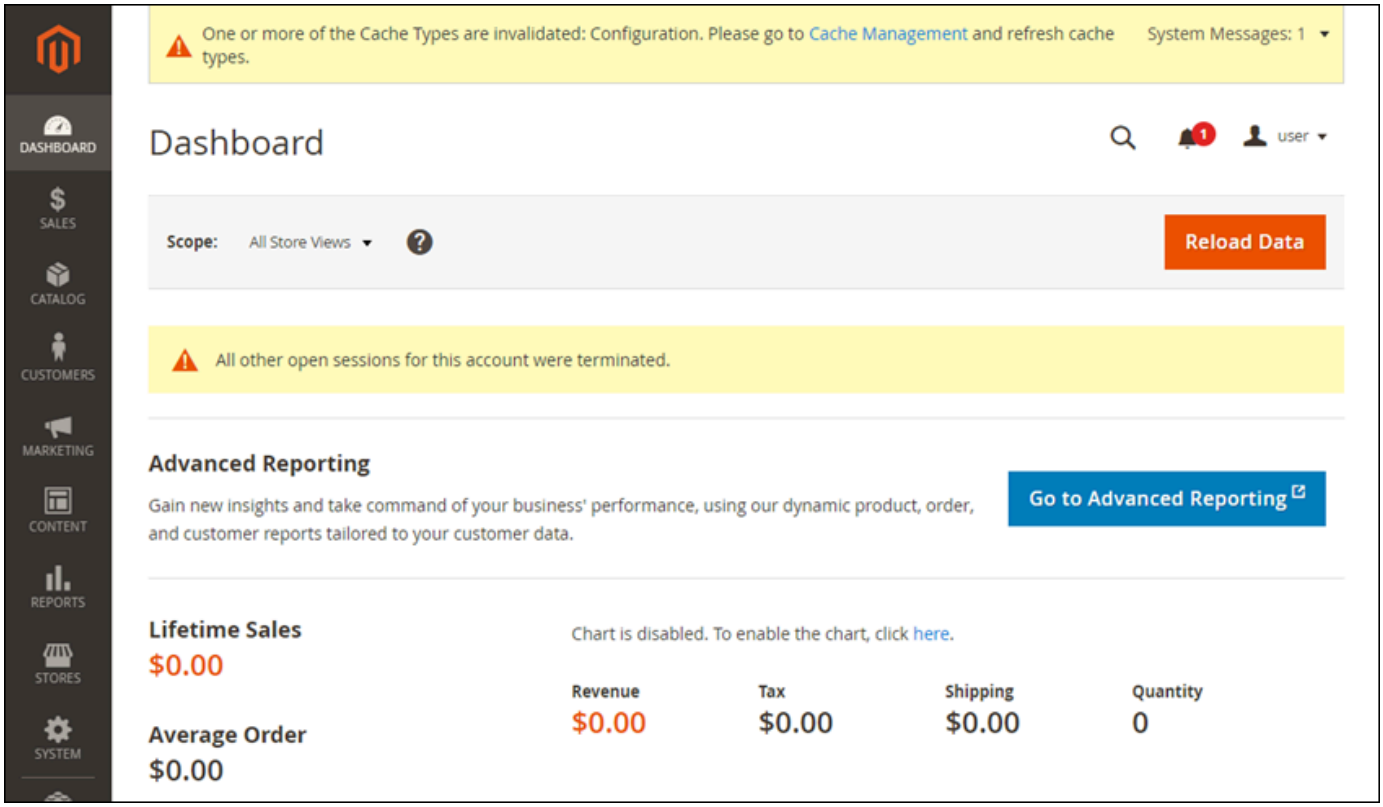
예:

```
http://203.0.113.0/admin
```

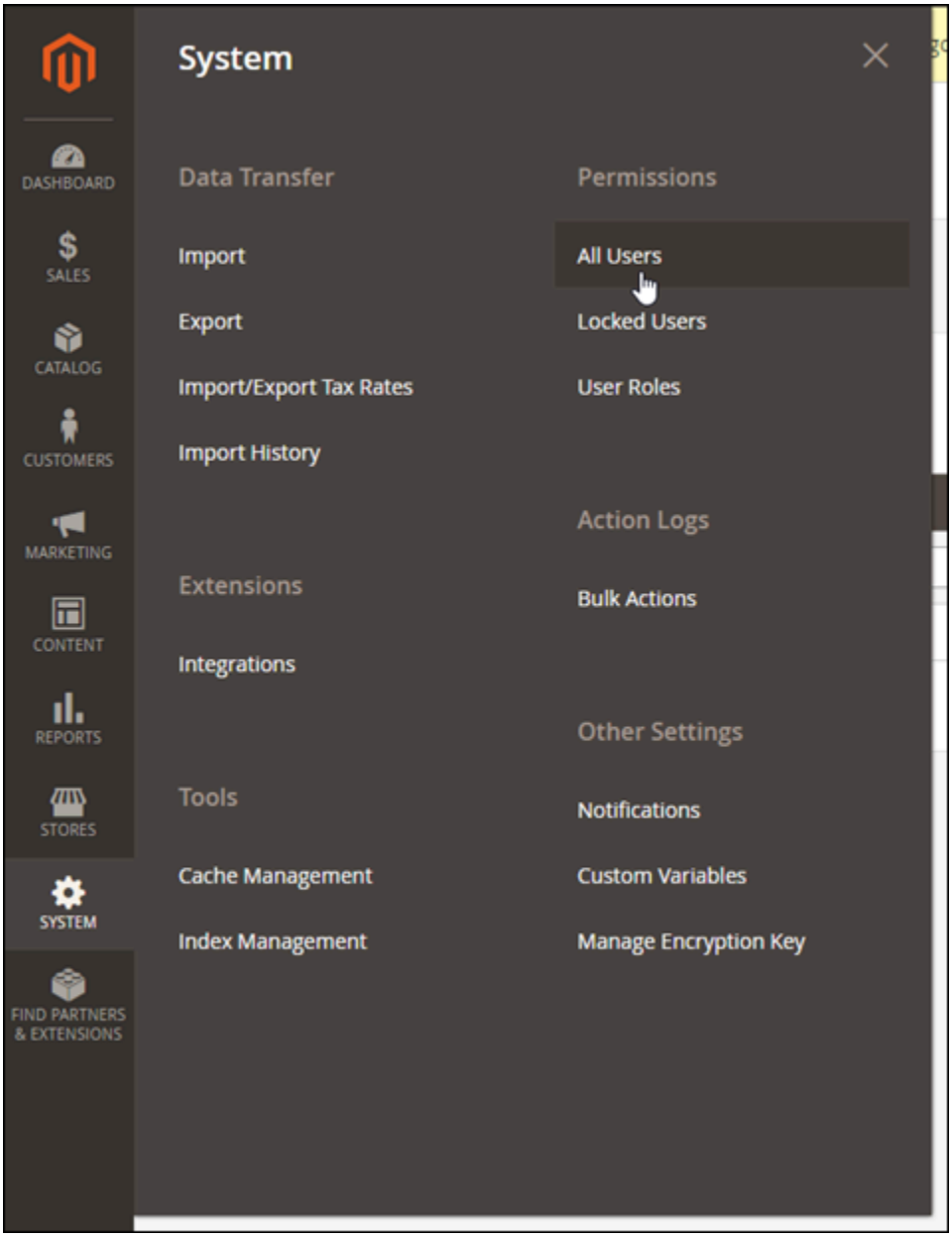
> **Note**
>
> Magento 관리 대시보드 로그인 페이지에 액세스할 수 없을 경우 인스턴스를 재부팅해야 할 수 있습니다.

3. 이 가이드의 앞부분에서 확인한 기본 애플리케이션 암호와 기본 사용자 이름(`user`)을 입력하고 로그인(Sign in)을 선택합니다.

Magento 관리 대시보드가 나타납니다.



Magento 웹 사이트의 관리 대시보드에 로그인하는 데 사용하는 기본 사용자 이름 또는 암호를 변경하려면 탐색 창에서 시스템(System)을 선택하고 모든 사용자(All Users)를 선택합니다. 자세한 내용을 알아보려면 Magento 설명서의 Adding users(사용자 추가)를 참조하세요.



관리 대시보드에 대한 자세한 내용을 알아보려면 Magento 2.4 User Guide(Magento 2.4 사용자 설명서)를 참조하세요.

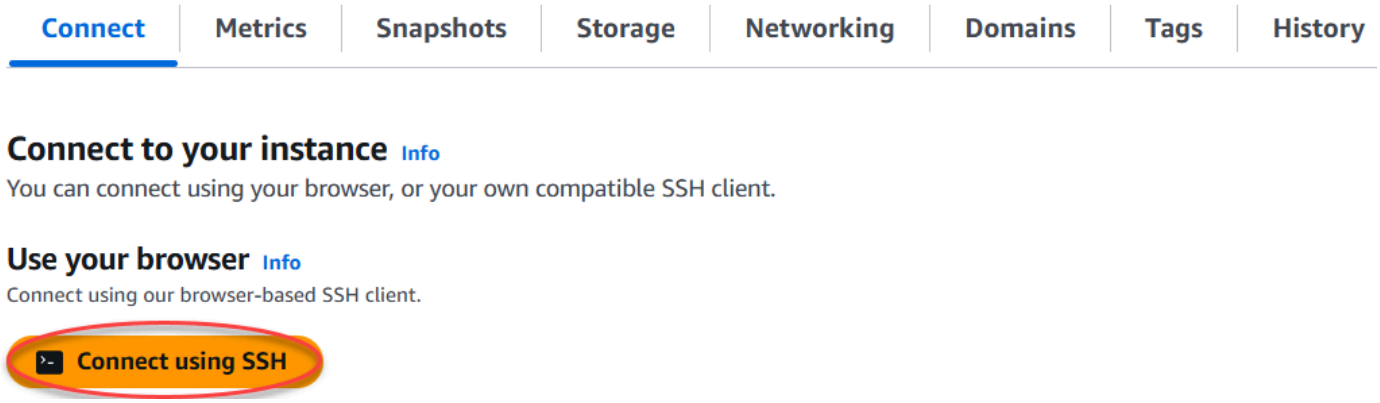
4단계: Magento 웹 사이트로 등록된 도메인 이름의 트래픽 라우팅

등록된 도메인 이름(예: `example.com`)의 트래픽을 Magento 웹 사이트로 라우팅하려면 도메인의 도메인 이름 시스템(DNS)에 레코드를 추가하면 됩니다. DNS 레코드는 일반적으로 도메인을 등록한 등록 대행자가 관리 및 호스팅합니다. 그러나 Lightsail 콘솔을 사용하여 관리할 수 있도록 도메인의 DNS 레코드 관리를 Lightsail로 이전하는 것이 좋습니다.

Lightsail 콘솔 홈 페이지의 도메인 및 DNS(Domains & DNS) 탭에서 DNS 영역 생성(Create DNS zone)을 선택하고 페이지의 지침에 따릅니다. 자세한 내용은 DNS 영역을 생성하여 Lightsail에서 도메인의 DNS 레코드 관리를 참조하세요.

도메인 이름이 트래픽을 인스턴스로 라우팅한 후 Magento 소프트웨어가 도메인 이름을 인식하도록 하려면 다음 단계를 완료해야 합니다.

1. 인스턴스 관리 페이지의 연결(Connect) 탭에서 SSH를 사용하여 연결(Connect using SSH)을 선택합니다.



2. 연결한 후 다음 명령을 입력합니다. *<DomainName>*을 트래픽을 인스턴스로 라우팅할 도메인 이름으로 대체해야 합니다.

```
sudo /opt/bitnami/configure_app_domain --domain <DomainName>
```

예:

```
sudo /opt/bitnami/configure_app_domain --domain www.example.com
```

다음 예와 비슷한 응답이 나타나는 것을 볼 수 있습니다. 이제 Magento 소프트웨어가 도메인 이름을 인식합니다.

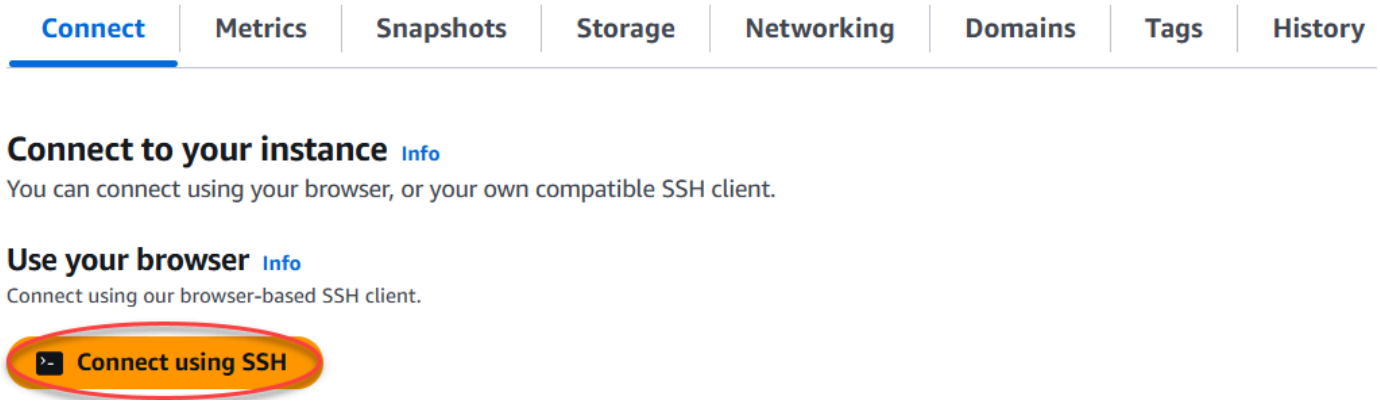
```
bitnami@ip-[redacted]:~$ sudo /opt/bitnami/configure_app_domain --domain www.example.com
Configuring domain to www.example.com
2021-03-12T15:49:22.000Z - info: Saving configuration info to disk
prestashop 15:49:22.41 INFO  ==> Trying to connect to the database server
prestashop 15:49:22.44 INFO  ==> Updating hostname in database
prestashop 15:49:22.46 INFO  ==> Purging cache
Disabling automatic domain update for IP address changes
```

5단계: Magento 웹 사이트에 대해 HTTPS 구성

Magento 웹 사이트에서 HTTPS를 구성하려면 다음 단계를 완료하세요. 다음 단계에서는 SSL/TLS 인증서 요청, 리디렉션(예: HTTP에서 HTTPS로) 설정 및 인증서 갱신에 사용하는 명령줄 도구인 Bitnami HTTPS 구성 도구(bncert)를 사용하는 방법을 안내합니다.

> **Important**
>
> bncert 도구는 Magento 인스턴스의 퍼블릭 IP 주소로 트래픽을 라우팅하고 있는 도메인에 대해서만 인증서를 발급합니다. 이러한 단계를 시작하기 전에 Magento 웹 사이트에서 사용할 모든 도메인의 DNS에 DNS 레코드를 추가해야 합니다.

1. 인스턴스 관리 페이지의 연결(Connect) 탭에서 SSH를 사용하여 연결(Connect using SSH)을 선택합니다.



2. 연결한 후 다음 명령을 입력하여 bncert-tool을 시작합니다.

```
sudo /opt/bitnami/bncert-tool
```

다음 예와 비슷한 응답이 나타나는 것을 볼 수 있습니다.

```
bitnami@ip-[redacted]:~$ sudo /opt/bitnami/bncert-tool
Warning: Custom redirections are not supported in the Bitnami Magento Stack.
This tool will not be able to enable/disable redirections.
Press [Enter] to continue:
```

3. 다음 예와 같이 기본 도메인 이름과 대체 도메인 이름을 공백으로 구분하여 입력합니다.

```
------------------------------------------------------------------------------
Welcome to the Bitnami HTTPS Configuration tool.

------------------------------------------------------------------------------
Domains

Please provide a valid space-separated list of domains for which you wish to
configure your web server.

Domain list []: example.com www.example.com
```

4. 적용될 변경 사항의 목록이 나열됩니다. Y를 입력하고 Enter 키를 눌러 확인하고 계속합니다.

```
------------------------------------------------------------------------------
Changes to perform

The following changes will be performed to your Bitnami installation:

1. Stop web server
2. Configure web server to use a free Let's Encrypt certificate for the domains:
 example.com www.example.com
3. Configure a cron job to automatically renew the certificate each month
4. Configure web server name to: example.com
5. Start web server once all changes have been performed



Do you agree to these changes? [Y/n]: Y
```

5. Let's Encrypt 인증서에 연결할 이메일 주소를 입력하고 Enter 키를 누릅니다.

```
Create a free HTTPS certificate with Let's Encrypt

Please provide a valid e-mail address for which to associate your Let's Encrypt
certificate.

Domain list: example.com www.example.com

Server name: example.com

E-mail address []:
```

6. Let's Encrypt 구독자 계약을 검토합니다. Y를 입력하고 Enter 키를 눌러 계약을 수락하고 계속합니다.

```
The Let's Encrypt Subscriber Agreement can be found at:

https://letsencrypt.org/documents/LE-SA-v1.2-November-15-2017.pdf

Do you agree to the Let's Encrypt Subscriber Agreement? [Y/n]:
```

인증서 요청 및 지정한 리디렉션 구성을 비롯하여 인스턴스에서 HTTPS를 활성화하는 작업이 수행됩니다.

```
Performing changes to your installation

The Bitnami HTTPS Configuration Tool will perform any necessary actions to your
Bitnami installation. This may take some time, please be patient.
```

인증서가 성공적으로 발급되고 검증됩니다. 다음 예와 유사한 메시지가 표시되면 인스턴스에서 리디렉션이 성공적으로 구성된 것입니다.

```
Success

The Bitnami HTTPS Configuration Tool succeeded in modifying your installation.

The configuration report is shown below.

Backup files:
* /opt/bitnami/apache/conf/httpd.conf.back.202104052147
* /opt/bitnami/apache/conf/bitnami/bitnami.conf.back.202104052147
* /opt/bitnami/apache/conf/bitnami/bitnami-ssl.conf.back.202104052147
* /opt/bitnami/apache/conf/vhosts/magento-https-vhost.conf.back.202104052147
* /opt/bitnami/apache/conf/vhosts/magento-vhost.conf.back.202104052147

Find more details in the log file:

/tmp/bncert-202104052147.log

If you find any issues, please check Bitnami Support forums at:

https://community.bitnami.com

Press [Enter] to continue:

bitnami@ip-172-26-3-145:~$
```

bncert 도구는 인증서가 만료되기 80일 전마다 인증서를 자동으로 갱신합니다. 다음 단계를 계속 진행하여 Magento 웹 사이트에서 HTTPS 활성화를 마칩니다.

7. 다음 주소를 검색하여 Magento 웹 사이트의 관리 대시보드 로그인 페이지에 액세스합니다. *<DomainName>*을 트래픽을 인스턴스로 라우팅하는 등록된 도메인 이름으로 대체해야 합니다.

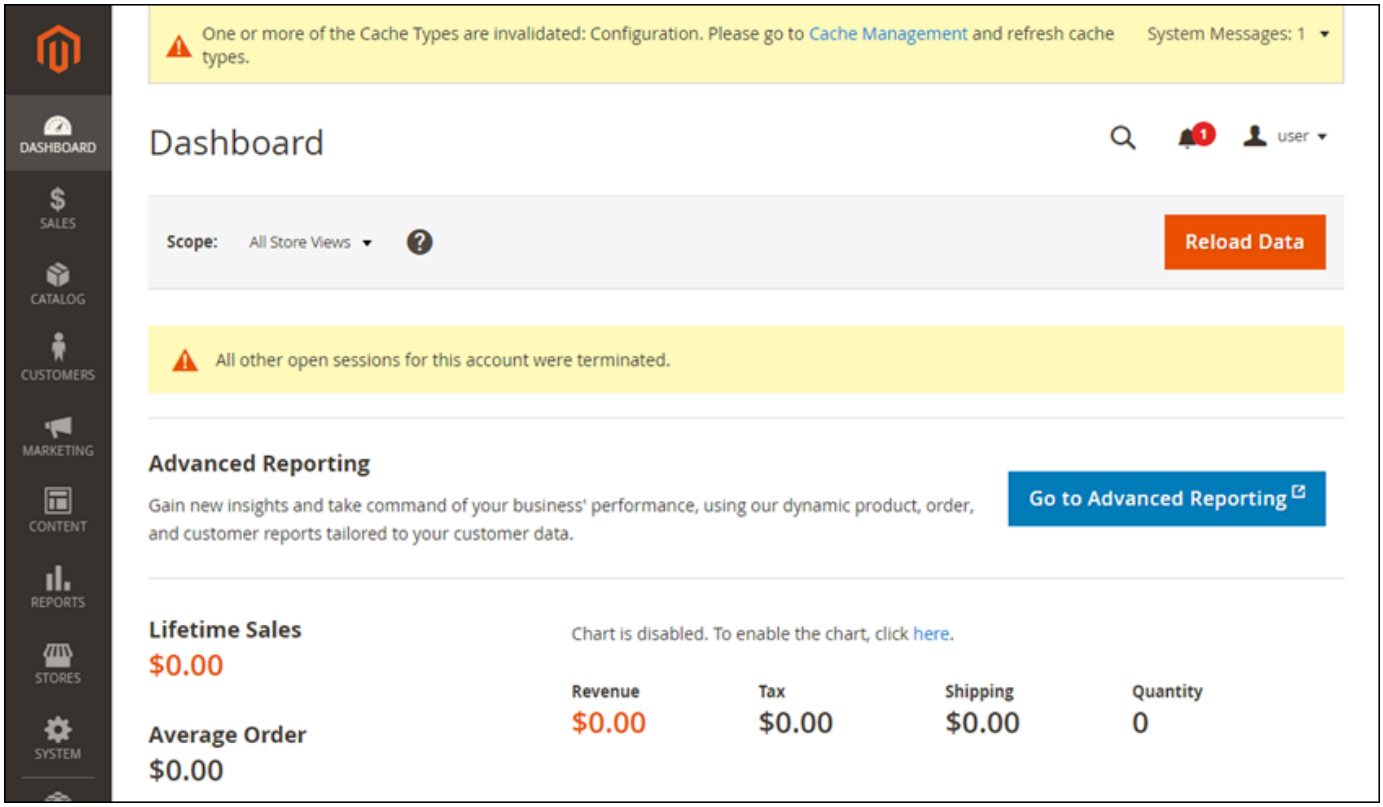
```
http://<DomainName>/admin
```

예:

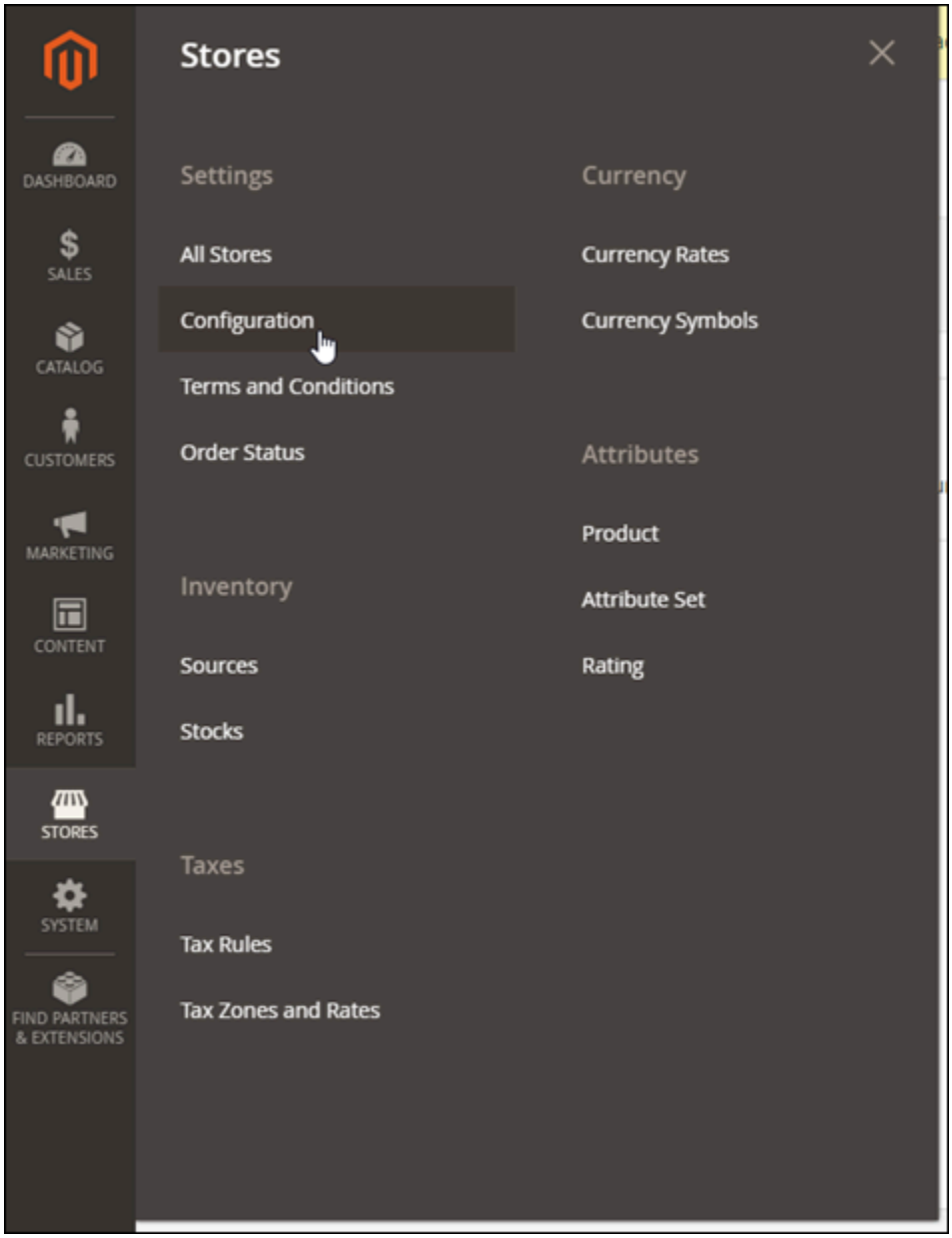
```
http://www.example.com/admin
```

8. 이 가이드의 앞부분에서 확인한 기본 애플리케이션 암호와 기본 사용자 이름(`user`)을 입력하고 로그인(Sign in)을 선택합니다.

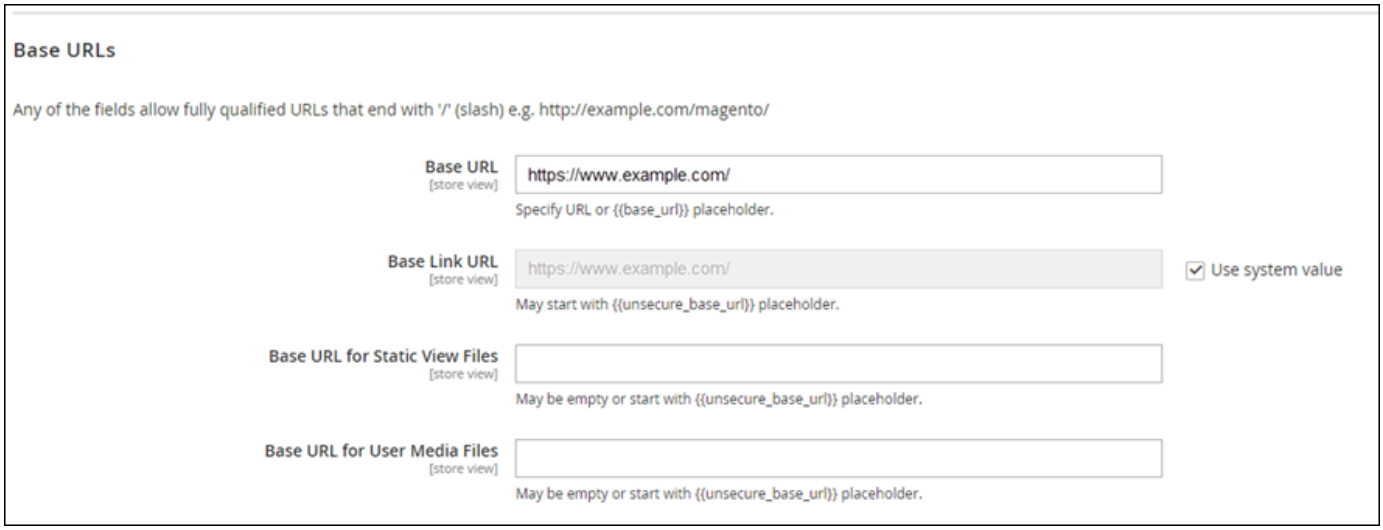
Magento 관리 대시보드가 나타납니다.



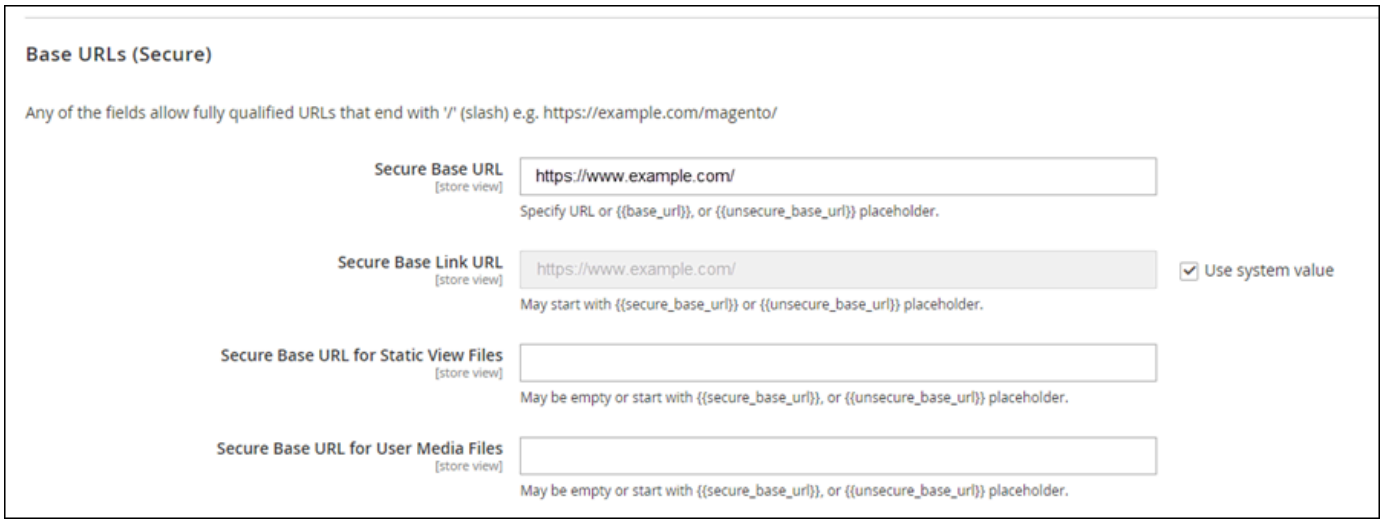
9. 탐색 창에서 스토어(Stores)를 선택하고 구성(Configuration)을 선택합니다.



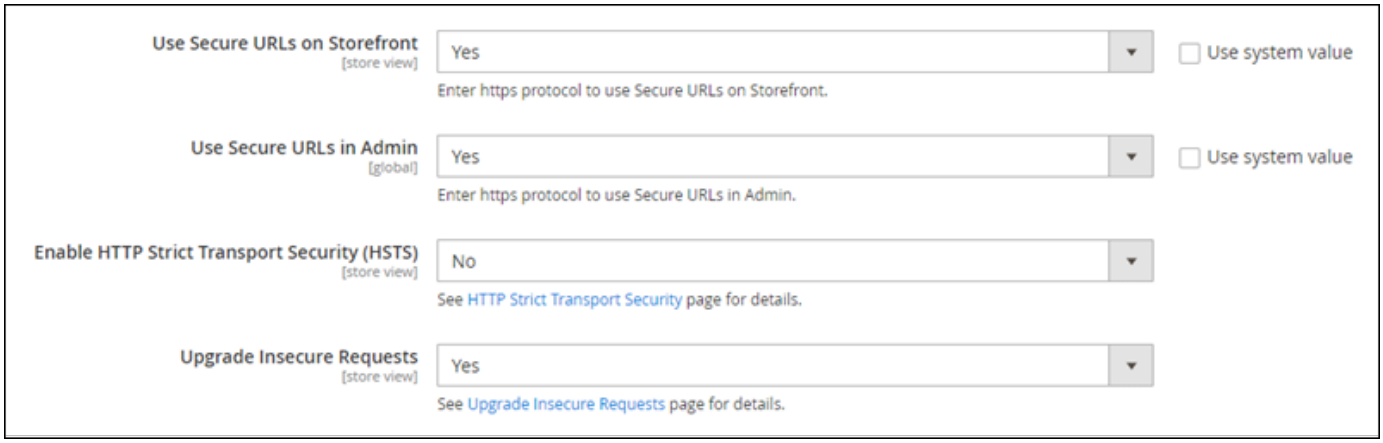
10. 웹(Web)을 선택하고 기본 URL(Base URLs) 노드를 확장합니다.
11. 기본 URL(Base URL) 텍스트 상자에 웹 사이트의 전체 URL을 입력합니다(예: `https://www.example.com/`).



12. 기본 URL(보안)(Base URLs (Secure)) 노드를 확장합니다.
13. 보안 기본 URL(Secure Base URL) 텍스트 상자에 웹 사이트의 전체 URL을 입력합니다(예: `https://www.example.com/`).



14. 스토어프론트에서 보안 URL 사용(Use Secure URLs on Storefront), 관리자에서 보안 URL 사용(Use Secure URLs in Admin) 및 보안되지 않은 요청 업그레이드(Upgrade Insecure Requests) 옵션에 대해 예(Yes)를 선택합니다.



15. 페이지 상단에서 구성 저장(Save Config)을 선택합니다.

    이제 Magento 웹 사이트에 대해 HTTPS가 구성되었습니다. 고객이 HTTP 버전(예: `http://www.example.com`)의 Magento 웹 사이트로 이동하면 HTTPS 버전(예: `https://www.example.com`)으로 자동 리디렉션됩니다.

6단계: 이메일 알림을 위한 SMTP 구성

Magento 웹 사이트의 SMTP 설정을 구성하여 이메일 알림을 활성화할 수 있습니다. 자세한 내용을 알아보려면 Bitnami 설명서의 Install the Magento Magepal SMTP extension(Magento Magegal SMTP 확장 설치)을 참조하세요.

> **Important**
>
> 포트 25, 465 또는 587을 사용하도록 SMTP를 구성할 경우 Lightsail 콘솔의 인스턴스 방화벽에서 해당 포트를 열어야 합니다. 자세한 내용은 Amazon Lightsail의 방화벽 규칙 추가 및 편집을 참조하세요.
>
> Magento 웹 사이트에서 이메일을 보내도록 Gmail 계정을 구성한 경우 Gmail에 로그인할 때 사용하는 표준 암호 대신 앱 암호를 사용해야 합니다. 자세한 내용은 앱 암호로 로그인을 참조하세요.

7단계: Bitnami 및 Magento 설명서 읽기

Bitnami 설명서를 읽고 Magento 인스턴스와 웹 사이트에서 플러그 인 설치 및 테마 사용자 지정과 같은 관리 태스크를 수행하는 방법을 알아봅니다. 자세한 내용을 알아보려면 Bitnami 설명서의 AWS 클라우드용 Bitnami Magento 스택을 참조하세요.

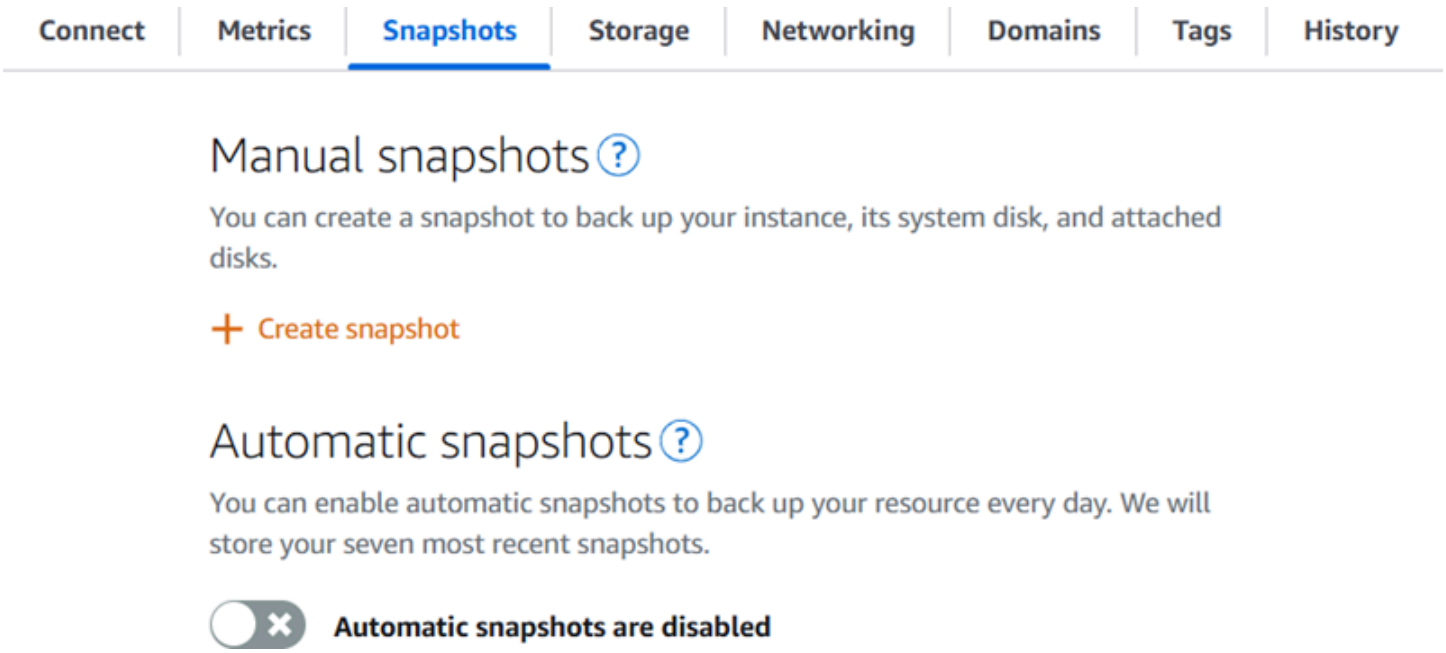
또한 Magento 설명서를 읽고 Magento 웹 사이트를 관리하는 방법을 알아보세요. 자세한 내용을 알아보려면 Magento 2.4 User Guide(Magento 2.4 사용 설명서)를 참조하세요.

8단계: Magento 인스턴스의 스냅샷 생성

원하는 방식으로 웹 사이트를 구성한 후 인스턴스의 주기적 스냅샷을 생성하여 백업합니다. 스냅샷은 시스템 디스크의 복사본이며, 인스턴스의 원본 구성입니다. 스냅샷은 스냅샷을 생성한 시점부터 인스턴스를 복원하는 데 필요한 모든 데이터를 포함합니다.

스냅샷을 수동으로 생성하거나 자동 스냅샷을 활성화하여 Lightsail에서 매일 스냅샷을 생성하도록 할 수 있습니다. 인스턴스에 문제가 있는 경우 스냅샷을 사용하여 새 교체 인스턴스를 생성할 수 있습니다.

스냅샷 탭의 인스턴스 관리 페이지에서 스냅샷 작업을 할 수 있습니다. 자세한 내용은 Amazon Lightsail의 스냅샷을 참조하세요.

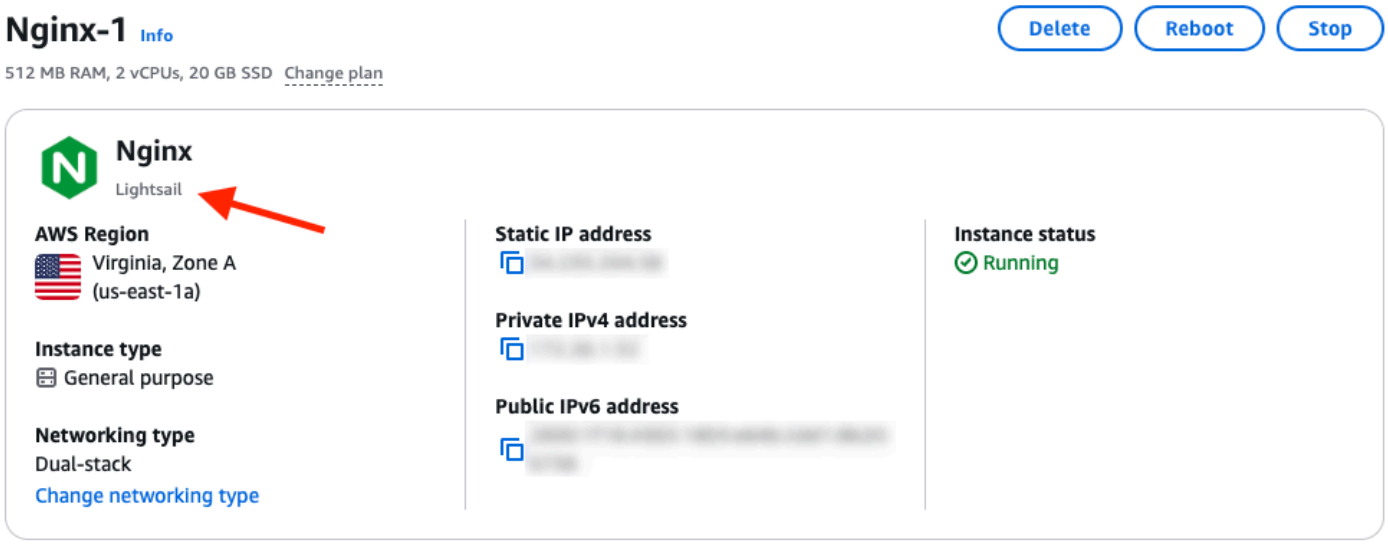


Lightsail에서 Nginx 웹 서버 배포 및 관리

> 알고 계셨나요?
>
> Lightsail는 일별 스냅샷 7개를 저장하고 인스턴스에 대해 자동 스냅샷을 활성화하면 가장 오래된 스냅샷을 최신 스냅샷으로 자동 대체합니다. 자세한 내용은 Lightsail 인스턴스 및 디스크의 자동 스냅샷 구성을 참조하세요.

Nginx 블루프린트 공급업체 식별

다음은에서 Nginx 인스턴스를 설치하고 실행한 후 시작하기 위해 수행해야 하는 몇 가지 단계입니다 Amazon Lightsail. 시작하기 전에 인스턴스 관리 페이지에서 블루프린트 공급업체를 식별합니다.



Nginx 인스턴스에 적합한 가이드를 선택합니다.

Bitnami

1단계: Nginx 인스턴스에 대한 기본 애플리케이션 암호 가져오기

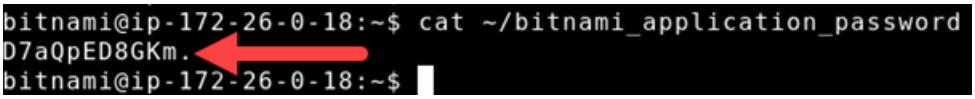
인스턴스의 사전 설치된 애플리케이션 또는 서비스에 액세스하려면 기본 애플리케이션 암호가 필요합니다.

1. 인스턴스 관리 페이지의 연결(Connect) 탭에서 SSH로 연결(Connect using SSH)을 선택합니다.
2. 연결한 후 다음 명령을 입력하여 기본 애플리케이션 암호를 가져옵니다.

```
cat bitnami_application_password
```

> **Note**
>
> 사용자 홈 디렉터리가 아닌 다른 디렉터리에 있는 경우 `cat $HOME/bitnami_application_password`를 입력합니다.

기본 애플리케이션 암호가 포함되어 있는 다음과 유사한 응답이 표시되어야 합니다.

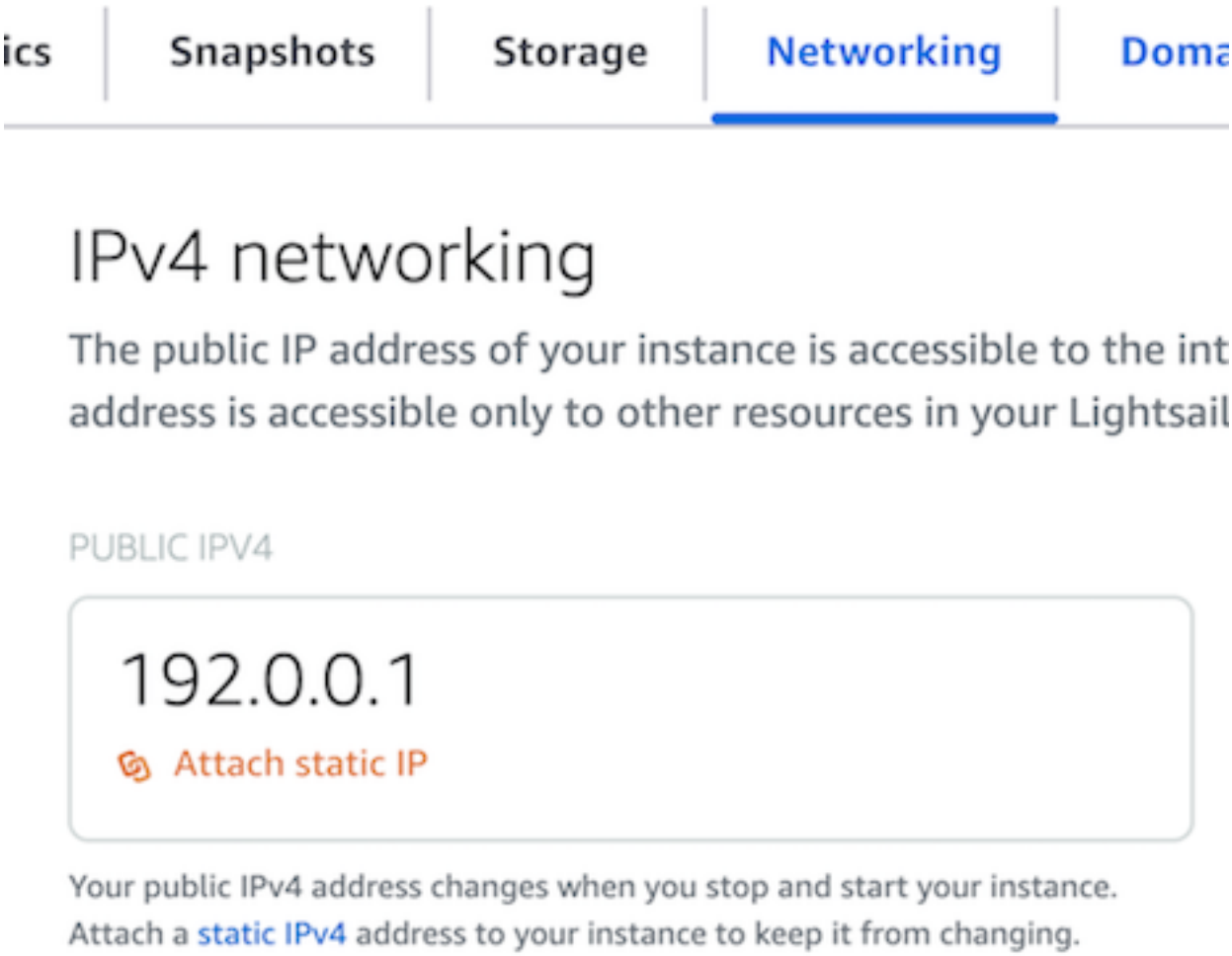


자세한 내용은 Amazon Lightsail의 Bitnami 인스턴스에 대한 애플리케이션 사용자 이름과 암호 가져오기를 참조하세요.

2단계: 고정 IP 주소를 Nginx 인스턴스에 연결

인스턴스에 연결된 기본 동적 퍼블릭 IP 주소는 인스턴스를 중지하고 시작할 때마다 변경됩니다. 퍼블릭 IP 주소가 변경되지 않도록 고정 IP 주소를 생성하고 인스턴스에 연결할 수 있습니다. 나중에 인스턴스에 도메인 이름을 사용하면 인스턴스를 중지하고 시작할 때마다 도메인의 DNS 레코드를 업데이트할 필요가 없습니다. 각 인스턴스에 1개의 고정 IP 주소만 연결할 수 있습니다.

인스턴스 관리 페이지의 네트워킹 탭에서 고정 IP 생성 또는 고정 IP 연결을 선택합니다. 인스턴스에 연결할 수 있는 고정 IP를 생성해 둔 경우에는 페이지의 지침을 따릅니다. 자세한 내용은 고정 IP를 생성하여 인스턴스에 연결을 참조하세요.



3단계: Nginx 인스턴스 시작 페이지 방문

인스턴스의 퍼블릭 IP 주소로 이동하여 해당 주소에 설치된 애플리케이션, phpMyAdmin 또는 Bitnami 설명서에 액세스합니다.

1. 인스턴스 관리 페이지의 연결 탭에서 퍼블릭 IP를 기록해 둡니다.
2. 퍼블릭 IP 주소로 이동합니다(예: `http://192.0.2.3`으로 이동).

자세한 내용은 Amazon Lightsail의 Bitnami 인스턴스에 대한 애플리케이션 사용자 이름과 암호 가져오기를 참조하세요.

4단계: 도메인 이름을 Nginx 인스턴스에 매핑

`example.com`과 같은 도메인 이름을 인스턴스에 매핑하려면 도메인의 도메인 이름 시스템(DNS)에 레코드를 추가합니다. DNS 레코드는 일반적으로 도메인을 등록한 등록 대행자가 관리 및 호

스팅합니다. 그러나 Lightsail 콘솔을 사용하여 관리할 수 있도록 도메인의 DNS 레코드 관리를 Lightsail로 이전하는 것이 좋습니다.

Lightsail 콘솔 홈 페이지의 네트워킹 탭에서 DNS 영역 생성을 선택하고 페이지의 지침에 따릅니다.

자세한 내용은 DNS 영역을 생성하여 도메인의 DNS 레코드 관리를 참조하세요.

5단계: Bitnami 설명서 읽기

Bitnami 설명서를 읽고 Nginx 애플리케이션 배포 방법, SSL 인증서로 HTTPS 지원을 활성화하는 방법, SFTP를 사용하여 서버에 파일을 업로드하는 방법 등을 알아보세요.

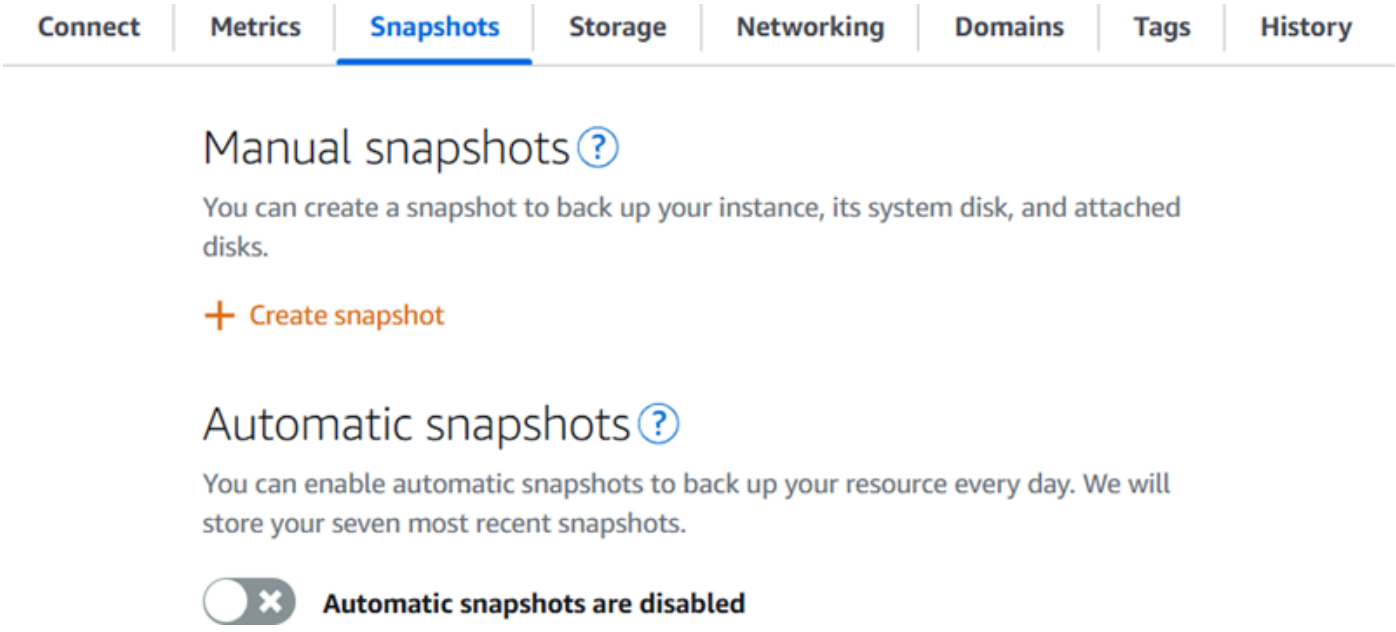
자세한 내용은 AWS 클라우드에 대한 Bitnami Nginx를 참조하세요.

6단계: Nginx 인스턴스의 스냅샷 생성

원하는 방식으로 웹 사이트를 구성한 후 인스턴스의 주기적 스냅샷을 생성하여 백업합니다. 스냅샷은 시스템 디스크의 복사본이며, 인스턴스의 원본 구성입니다. 스냅샷은 스냅샷을 생성한 시점부터 인스턴스를 복원하는 데 필요한 모든 데이터를 포함합니다.

스냅샷을 수동으로 생성하거나 자동 스냅샷을 활성화하여 Lightsail에서 매일 스냅샷을 생성하도록 할 수 있습니다. 인스턴스에 문제가 있는 경우 스냅샷을 사용하여 새 교체 인스턴스를 생성할 수 있습니다.

스냅샷 탭의 인스턴스 관리 페이지에서 스냅샷 작업을 할 수 있습니다. 자세한 내용은 Amazon Lightsail의 스냅샷을 참조하세요.



Lightsail

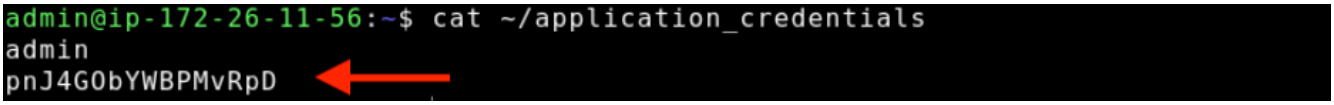
1단계: Nginx 인스턴스에 대한 기본 애플리케이션 암호 가져오기

인스턴스의 사전 설치된 애플리케이션 또는 서비스에 액세스하려면 기본 애플리케이션 암호가 필요합니다.

1. 인스턴스 관리 페이지의 연결(Connect) 탭에서 SSH로 연결(Connect using SSH)을 선택합니다.
2. 연결한 후 다음 명령을 입력하여 기본 애플리케이션 암호를 가져옵니다.

```
cat ~/application_credentials
```

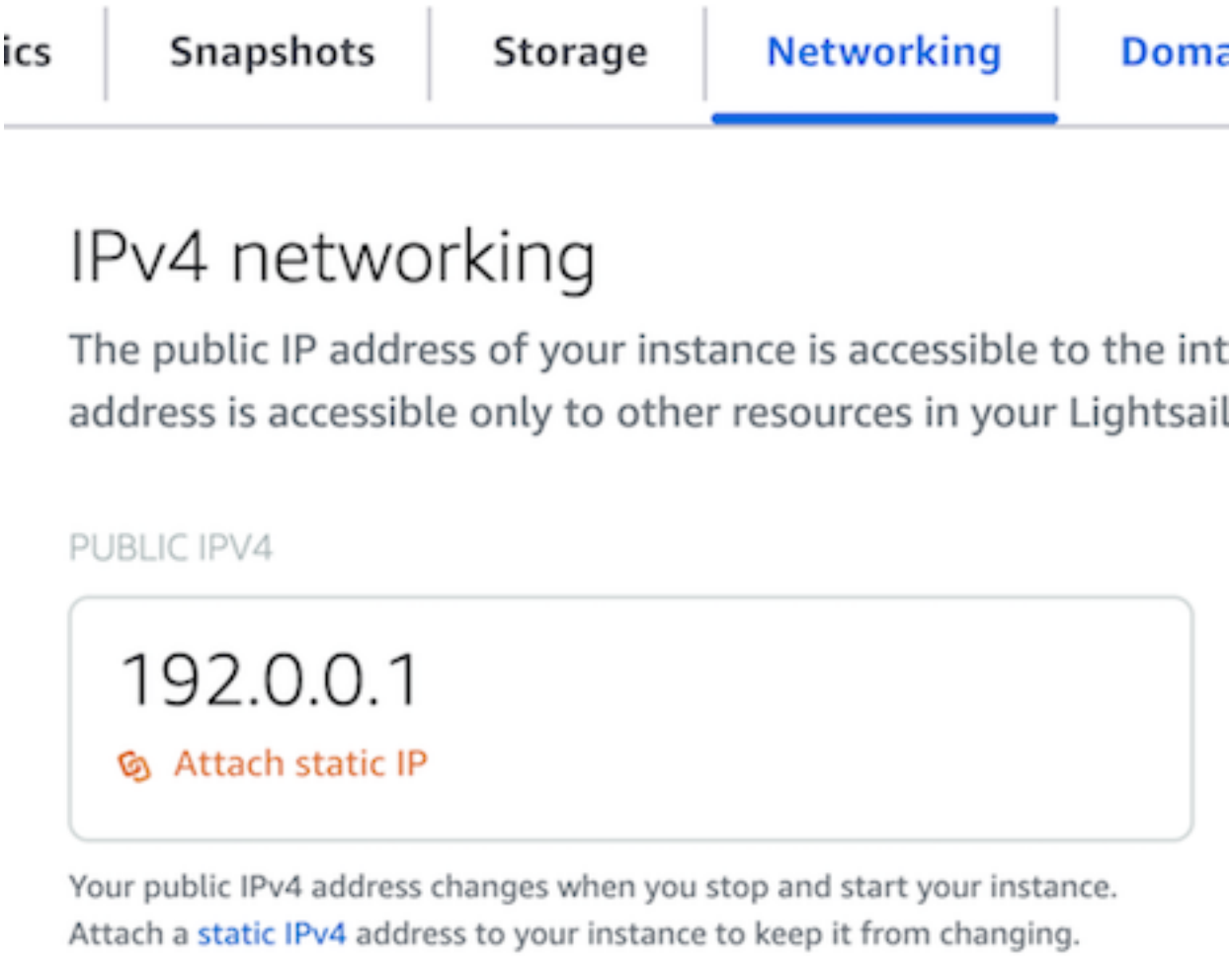
기본 애플리케이션 암호가 포함되어 있는 다음과 유사한 응답이 표시되어야 합니다.



2단계: 고정 IP 주소를 Nginx 인스턴스에 연결

인스턴스에 연결된 기본 동적 퍼블릭 IP 주소는 인스턴스를 중지하고 시작할 때마다 변경됩니다. 퍼블릭 IP 주소가 변경되지 않도록 고정 IP 주소를 생성하고 인스턴스에 연결할 수 있습니다. 나중에 인스턴스에 도메인 이름을 사용하면 인스턴스를 중지하고 시작할 때마다 도메인의 DNS 레코드를 업데이트할 필요가 없습니다. 각 인스턴스에 1개의 고정 IP 주소만 연결할 수 있습니다.

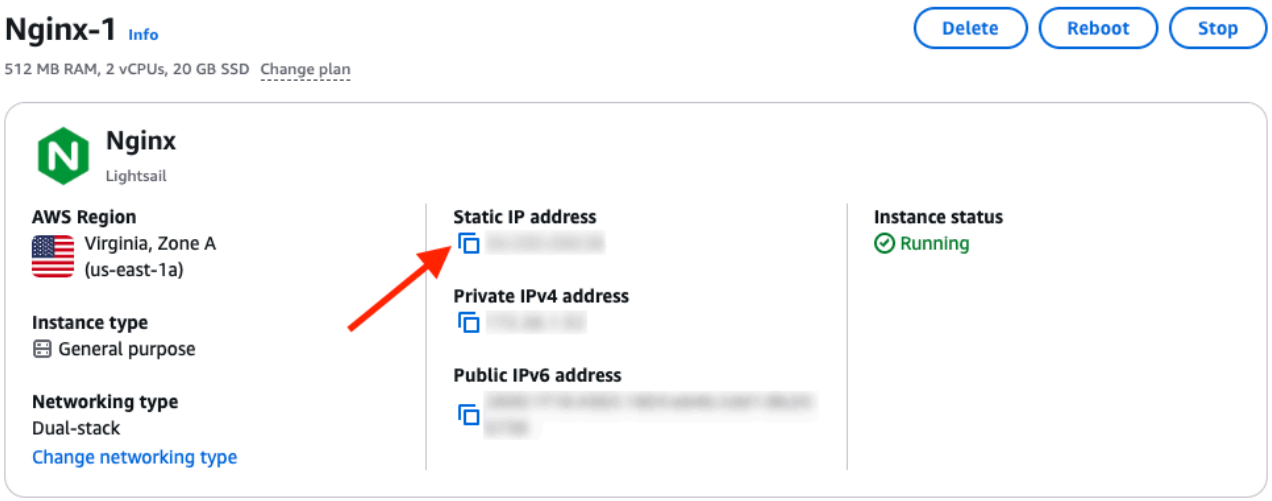
인스턴스 관리 페이지의 네트워킹 탭에서 고정 IP 생성 또는 고정 IP 연결을 선택합니다. 인스턴스에 연결할 수 있는 고정 IP를 생성해 둔 경우에는 페이지의 지침을 따릅니다. 자세한 내용은 고정 IP를 생성하여 인스턴스에 연결을 참조하세요.



3단계: Nginx 인스턴스 시작 페이지 방문

인스턴스의 고정 IP 주소로 이동하여 인스턴스에 설치된 애플리케이션에 액세스합니다.

1. 인스턴스 관리 페이지에서 고정 IP 주소를 복사합니다.



2. 예를 들어 정적 IP 주소를 브라우저 주소에 붙여 넣습니다`http://192.0.0.1`.

자세한 내용은 Nginx 관리자 안내서를 참조하세요.

4단계: 도메인 이름을 Nginx 인스턴스에 매핑

`example.com`과 같은 도메인 이름을 인스턴스에 매핑하려면 도메인의 도메인 이름 시스템(DNS)에 레코드를 추가합니다. DNS 레코드는 일반적으로 도메인을 등록한 등록 대행자가 관리 및 호스팅합니다. 그러나 Lightsail 콘솔을 사용하여 관리할 수 있도록 도메인의 DNS 레코드 관리를 Lightsail로 이전하는 것이 좋습니다.

Lightsail 콘솔 홈 페이지의 네트워킹 탭에서 DNS 영역 생성을 선택하고 페이지의 지침에 따릅니다.

자세한 내용은 DNS 영역을 생성하여 도메인의 DNS 레코드 관리를 참조하세요.

HTTPS를 활성화하려면 Let's Encrypt SSL/TLS를 사용하여 Lightsail Nginx 웹 사이트 보안을 참조하세요.

5단계: 애플리케이션 배포

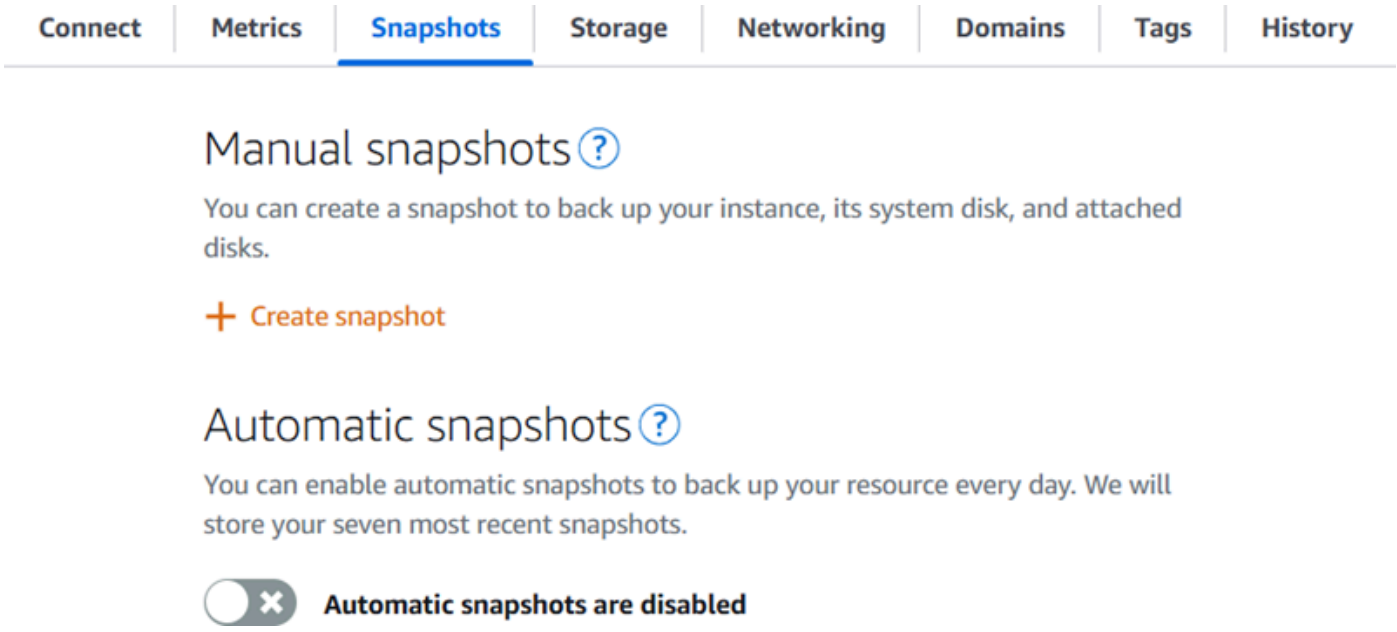
1. scp를 Lightsail 사용하여에서 Linux 인스턴스 간에 파일 전송의 지침에 따라 애플리케이션을 에 복사합니다. `/usr/share/nginx/html`
2. 인스턴스 관리 페이지의 연결(Connect) 탭에서 SSH로 연결(Connect using SSH)을 선택합니다.
3. `sudo systemctl restart nginx` 실행
4. 인스턴스의 고정 IP 주소로 이동

6단계: Nginx 인스턴스의 스냅샷 생성

원하는 방식으로 웹 사이트를 구성한 후 인스턴스의 주기적 스냅샷을 생성하여 백업합니다. 스냅샷은 시스템 디스크의 복사본이며, 인스턴스의 원본 구성입니다. 스냅샷은 스냅샷을 생성한 시점부터 인스턴스를 복원하는 데 필요한 모든 데이터를 포함합니다.

스냅샷을 수동으로 생성하거나 자동 스냅샷을 활성화하여 Lightsail에서 매일 스냅샷을 생성하도록 할 수 있습니다. 인스턴스에 문제가 있는 경우 스냅샷을 사용하여 새 교체 인스턴스를 생성할 수 있습니다.

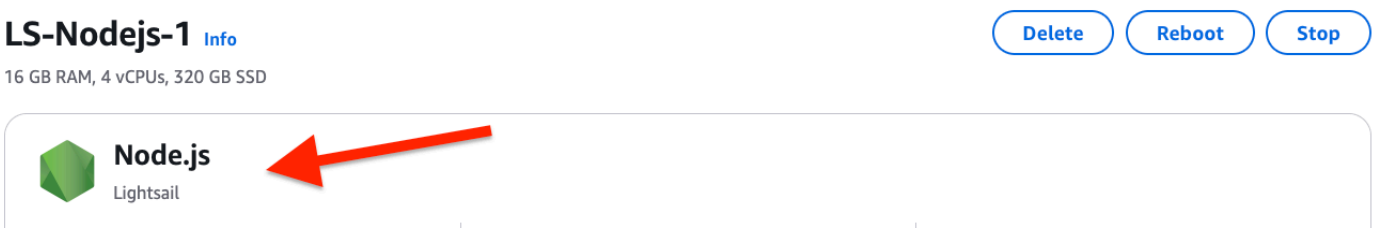
스냅샷 탭의 인스턴스 관리 페이지에서 스냅샷 작업을 할 수 있습니다. 자세한 내용은 Amazon Lightsail의 스냅샷을 참조하세요.



에서 Node.js 스택 배포 및 관리 Lightsail

> 알고 계셨나요?
>
> Lightsail는 일별 스냅샷 7개를 저장하고 인스턴스에 대해 자동 스냅샷을 활성화하면 가장 오래된 스냅샷을 최신 스냅샷으로 자동 대체합니다. 자세한 내용은 Lightsail 인스턴스 및 디스크의 자동 스냅샷 구성을 참조하세요.

다음은에서 Node.js 인스턴스를 설치하고 실행한 후 시작하기 위해 수행해야 하는 몇 가지 단계입니다 Amazon Lightsail. 시작하기 전에 인스턴스 관리 페이지에서 블루프린트 공급업체를 식별합니다.



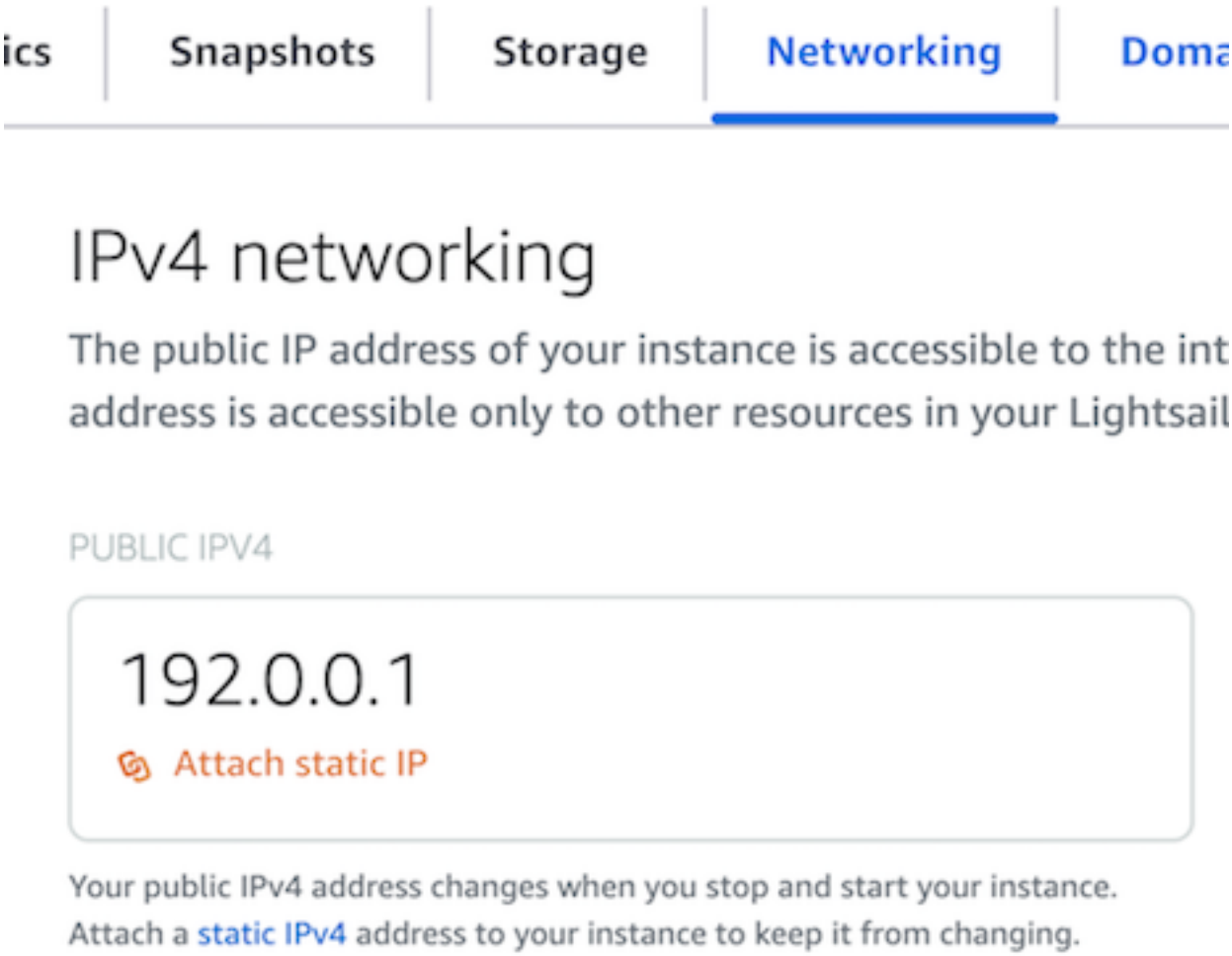
Node.js 인스턴스에 적합한 가이드를 선택합니다.

Bitnami

1단계: Node.js 인스턴스에 고정 IP 주소 연결

인스턴스에 연결된 기본 동적 퍼블릭 IP 주소는 인스턴스를 중지하고 시작할 때마다 변경됩니다. 퍼블릭 IP 주소가 변경되지 않도록 고정 IP 주소를 생성하고 인스턴스에 연결할 수 있습니다. 나중에 인스턴스에 도메인 이름을 사용하면 인스턴스를 중지하고 시작할 때마다 도메인의 DNS 레코드를 업데이트할 필요가 없습니다. 각 인스턴스에 1개의 고정 IP 주소만 연결할 수 있습니다.

인스턴스 관리 페이지의 네트워킹 탭에서 고정 IP 생성 또는 고정 IP 연결을 선택합니다. 인스턴스에 연결할 수 있는 고정 IP를 생성해 둔 경우에는 페이지의 지침을 따릅니다. 자세한 내용은 고정 IP를 생성하여 인스턴스에 연결을 참조하세요.



2단계: Node.js 인스턴스 시작 페이지 방문

인스턴스의 퍼블릭 IP 주소로 이동하여 해당 주소에 설치된 애플리케이션, phpMyAdmin 또는 Bitnami 설명서에 액세스합니다.

1. 인스턴스 관리 페이지의 연결 탭에서 퍼블릭 IP를 기록해 둡니다.
2. 퍼블릭 IP 주소로 이동합니다(예: `http://192.0.2.3`으로 이동).

자세한 내용은 Amazon Lightsail의 Bitnami 인스턴스에 대한 애플리케이션 사용자 이름과 암호 가져오기를 참조하세요.

3단계: 도메인 이름을 Node.js 인스턴스에 매핑

`example.com`과 같은 도메인 이름을 인스턴스에 매핑하려면 도메인의 도메인 이름 시스템(DNS)에 레코드를 추가합니다. DNS 레코드는 일반적으로 도메인을 등록한 등록 대행자가 관리 및 호

스팅합니다. 그러나 Lightsail 콘솔을 사용하여 관리할 수 있도록 도메인의 DNS 레코드 관리를 Lightsail로 이전하는 것이 좋습니다.

Lightsail 콘솔 홈 페이지의 네트워킹 탭에서 DNS 영역 생성을 선택하고 페이지의 지침에 따릅니다.

자세한 내용은 DNS 영역을 생성하여 도메인의 DNS 레코드 관리를 참조하세요.

4단계: Bitnami 설명서 읽기

Bitnami 설명서를 읽고 Node.js 애플리케이션 배포 방법, SSL 인증서로 HTTPS 지원을 활성화하는 방법, SFTP를 사용하여 서버에 파일을 업로드하는 방법 등을 알아보세요.

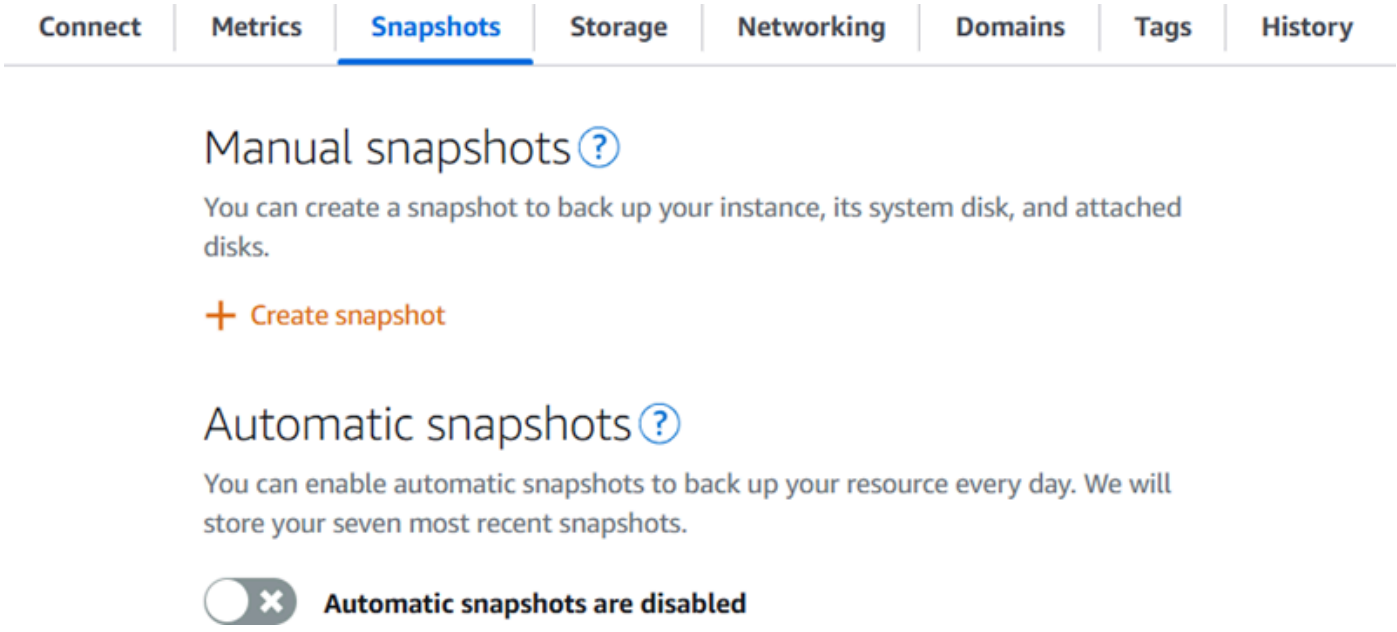
자세한 내용은 AWS 클라우드에 대한 Bitnami Node.js를 참조하세요.

5단계: Node.js 인스턴스의 스냅샷 생성

원하는 방식으로 웹 사이트를 구성한 후 인스턴스의 주기적 스냅샷을 생성하여 백업합니다. 스냅샷은 시스템 디스크의 복사본이며, 인스턴스의 원본 구성입니다. 스냅샷은 스냅샷을 생성한 시점부터 인스턴스를 복원하는 데 필요한 모든 데이터를 포함합니다.

스냅샷을 수동으로 생성하거나 자동 스냅샷을 활성화하여 Lightsail에서 매일 스냅샷을 생성하도록 할 수 있습니다. 인스턴스에 문제가 있는 경우 스냅샷을 사용하여 새 교체 인스턴스를 생성할 수 있습니다.

스냅샷 탭의 인스턴스 관리 페이지에서 스냅샷 작업을 할 수 있습니다. 자세한 내용은 Amazon Lightsail의 스냅샷을 참조하세요.



Lightsail

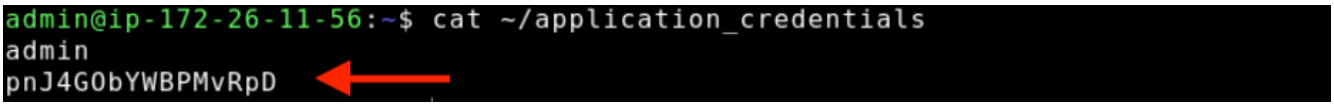
1단계: Node.js 인스턴스에 대한 기본 애플리케이션 암호 가져오기

인스턴스의 사전 설치된 애플리케이션 또는 서비스에 액세스하려면 기본 애플리케이션 암호가 필요합니다.

1. 인스턴스 관리 페이지의 연결(Connect) 탭에서 SSH로 연결(Connect using SSH)을 선택합니다.
2. 연결한 후 다음 명령을 입력하여 기본 애플리케이션 암호를 가져옵니다.

```
cat ~/application_credentials
```

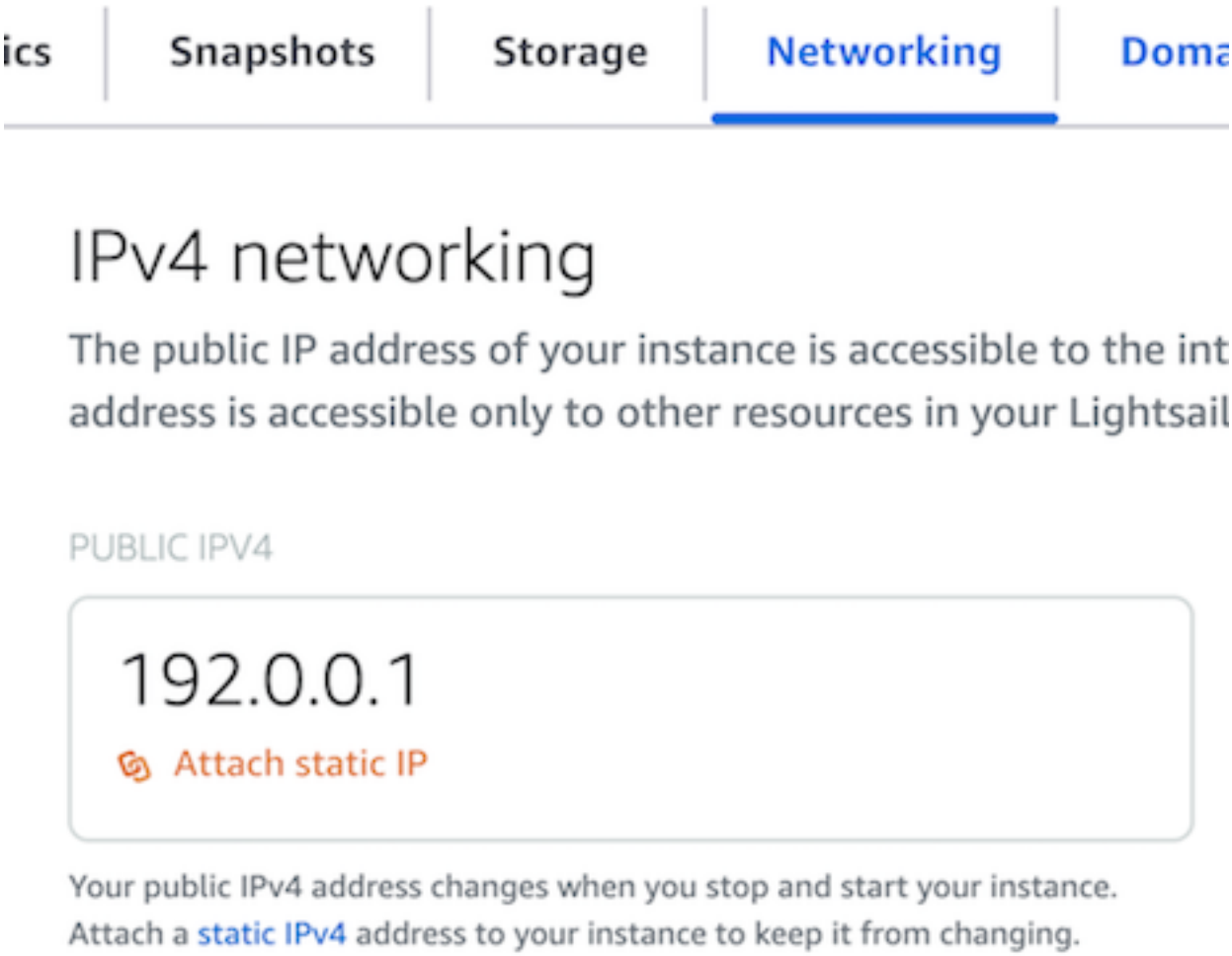
기본 애플리케이션 암호가 포함되어 있는 다음과 유사한 응답이 표시되어야 합니다.



2단계: 고정 IP 주소를 Node.js 인스턴스에 연결

인스턴스에 연결된 기본 동적 퍼블릭 IP 주소는 인스턴스를 중지하고 시작할 때마다 변경됩니다. 퍼블릭 IP 주소가 변경되지 않도록 고정 IP 주소를 생성하고 인스턴스에 연결할 수 있습니다. 나중에 인스턴스에 도메인 이름을 사용하면 인스턴스를 중지하고 시작할 때마다 도메인의 DNS 레코드를 업데이트할 필요가 없습니다. 각 인스턴스에 1개의 고정 IP 주소만 연결할 수 있습니다.

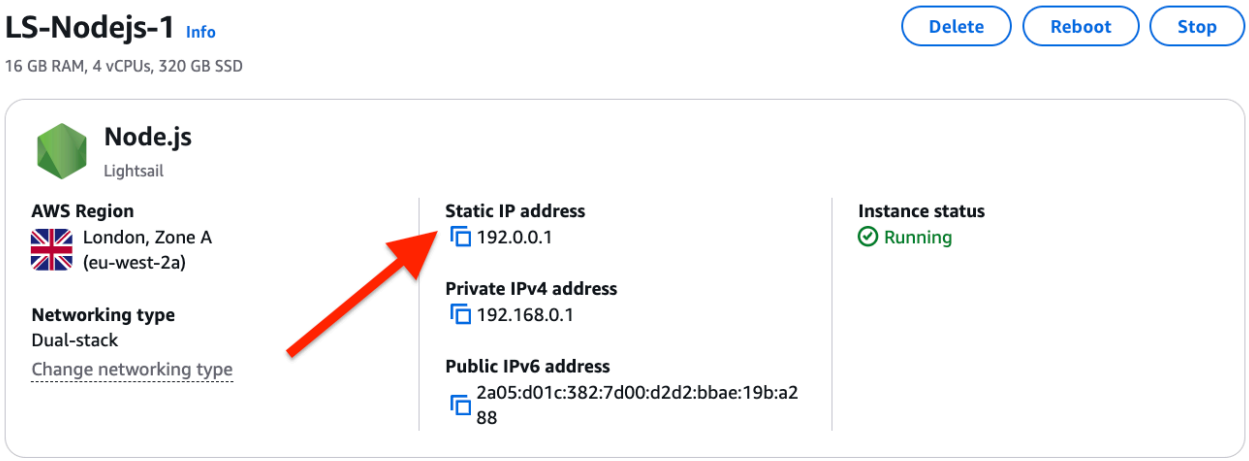
인스턴스 관리 페이지의 네트워킹 탭에서 고정 IP 생성 또는 고정 IP 연결을 선택합니다. 인스턴스에 연결할 수 있는 고정 IP를 생성해 둔 경우에는 페이지의 지침을 따릅니다. 자세한 내용은 고정 IP를 생성하여 인스턴스에 연결을 참조하세요.



3단계: Node.js 인스턴스 시작 페이지 방문

인스턴스의 고정 IP 주소로 이동하여 인스턴스에 설치된 애플리케이션에 액세스하거나 phpMyAdmin에 액세스합니다.

1. 인스턴스 관리 페이지에서 고정 IP 주소를 복사합니다.



2. 예를 들어 정적 IP 주소를 브라우저 주소에 붙여 넣습니다`http://192.0.0.1`.

자세한 내용은 Node.js 관리자 안내서를 참조하세요.

4단계: 도메인 이름을 Node.js 인스턴스에 매핑

`example.com`과 같은 도메인 이름을 인스턴스에 매핑하려면 도메인의 도메인 이름 시스템(DNS)에 레코드를 추가합니다. DNS 레코드는 일반적으로 도메인을 등록한 등록 대행자가 관리 및 호스팅합니다. 그러나 Lightsail 콘솔을 사용하여 관리할 수 있도록 도메인의 DNS 레코드 관리를 Lightsail로 이전하는 것이 좋습니다.

Lightsail 콘솔 홈 페이지의 네트워킹 탭에서 DNS 영역 생성을 선택하고 페이지의 지침에 따릅니다.

자세한 내용은 DNS 영역을 생성하여 도메인의 DNS 레코드 관리를 참조하세요.

5단계: 애플리케이션 배포

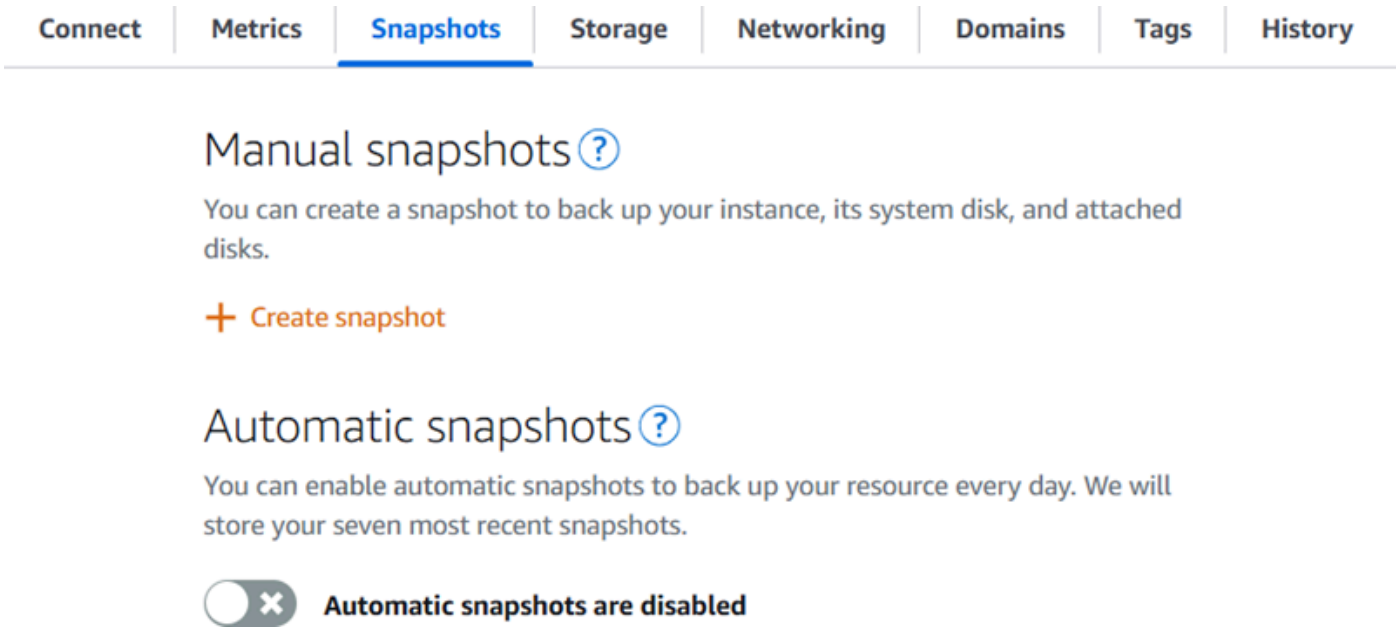
1. scp를 Lightsail 사용하여에서 Linux 인스턴스 간에 파일 전송의 지침에 따라 애플리케이션을 에 복사합니다. `/var/www/html/`
2. 인스턴스 관리 페이지의 연결(Connect) 탭에서 SSH로 연결(Connect using SSH)을 선택합니다.
3. `sudo pm2 restart nodeapp` 실행
4. 인스턴스의 고정 IP 주소로 이동

6단계: Node.js 인스턴스의 스냅샷 생성

원하는 방식으로 웹 사이트를 구성한 후 인스턴스의 주기적 스냅샷을 생성하여 백업합니다. 스냅샷은 시스템 디스크의 복사본이며, 인스턴스의 원본 구성입니다. 스냅샷은 스냅샷을 생성한 시점부터 인스턴스를 복원하는 데 필요한 모든 데이터를 포함합니다.

스냅샷을 수동으로 생성하거나 자동 스냅샷을 활성화하여 Lightsail에서 매일 스냅샷을 생성하도록 할 수 있습니다. 인스턴스에 문제가 있는 경우 스냅샷을 사용하여 새 교체 인스턴스를 생성할 수 있습니다.

스냅샷 탭의 인스턴스 관리 페이지에서 스냅샷 작업을 할 수 있습니다. 자세한 내용은 Amazon Lightsail의 스냅샷을 참조하세요.



에서 OpenClaw 시작하기 Lightsail

OpenClaw는에서 실행되는 AI 기반 채팅 게이트웨이로Amazon Lightsail, 브라우저, 텔레그램, WhatsApp 등에서 액세스할 수 있는 프라이빗 자체 호스팅 AI 어시스턴트를 제공합니다. 이 자습서에서는 Amazon Lightsail OpenClaw 인스턴스를 시작하고, 브라우저를 페어링하고, AI 기능을 활성화하고, 선택적으로 메시징 채널을 연결하는 방법을 안내합니다.

> 알고 계셨나요?
>
> Lightsail OpenClaw 인스턴스는 Amazon Bedrock이 기본 AI 모델 공급자로 사전 구성되어 제공됩니다. 설정을 완료하면 추가 구성 없이 AI 어시스턴트와 즉시 채팅을 시작할 수 있습니다.

1단계: OpenClaw 인스턴스 생성

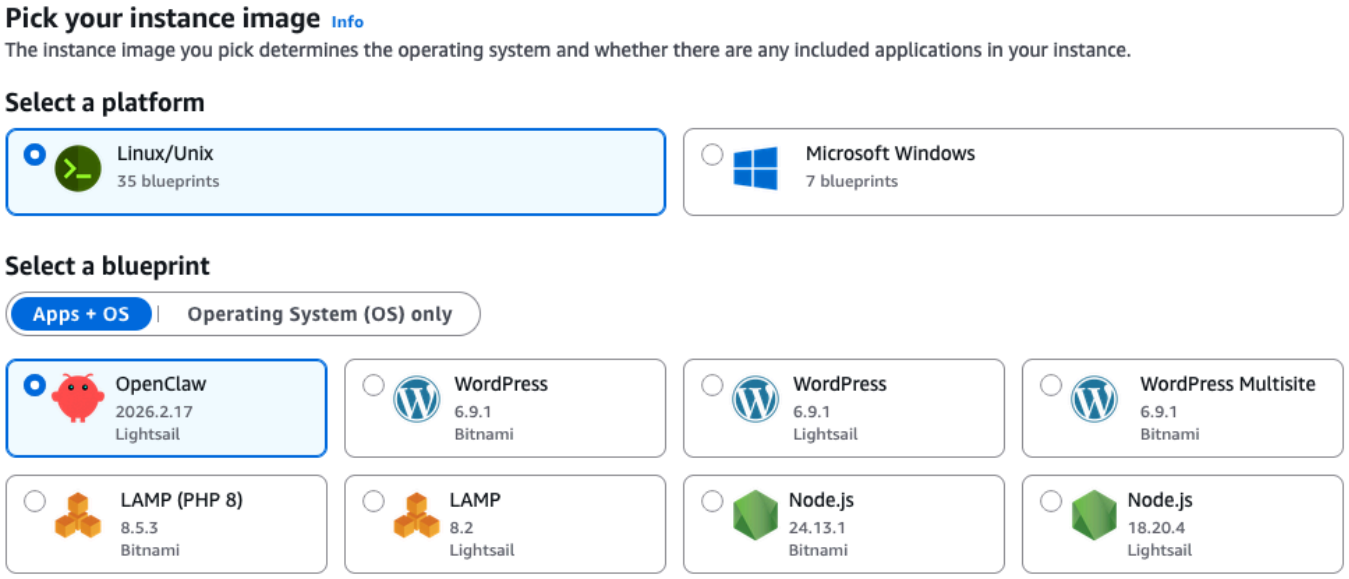
이 단계에서는 OpenClaw를 실행하는 Lightsail 인스턴스를 생성합니다.

OpenClaw를 사용하여 Lightsail 인스턴스를 생성하려면

1. Lightsail 콘솔에 로그인합니다.
2. Lightsail 홈 페이지의 인스턴스(Instances) 섹션에서 인스턴스 생성(Create instance)을 선택합니다.
3. 인스턴스의 AWS 리전 및 가용 영역을 선택합니다.

4. 다음과 같이 인스턴스의 이미지를 선택합니다.

   - 플랫폼 선택에서 Linux/Unix를 선택합니다.
   - 블루프린트 선택에서 OpenClaw를 선택합니다.



5. 인스턴스 플랜 선택에서 인스턴스 플랜을 선택합니다(최적의 성능을 위해 4GB 메모리 플랜 권장).
6. 인스턴스의 이름을 입력하거나 제공된 기본 이름을 사용합니다.

> **Note**
>
> 인스턴스 이름은 Lightsail 계정 내에서 고유해야 하며, 2~255자여야 하고, 영숫자 문자로 시작하고 끝나야 하며, 영숫자 문자, 마침표, 대시 또는 밑줄만 포함해야 합니다.

7. 인스턴스 생성을 선택합니다.

인스턴스가 시작되는 동안 몇 분 동안 인스턴스가 시작 중 상태가 됩니다. 다음 단계를 진행하기 전에 상태가 실행 중으로 표시될 때까지 기다립니다.

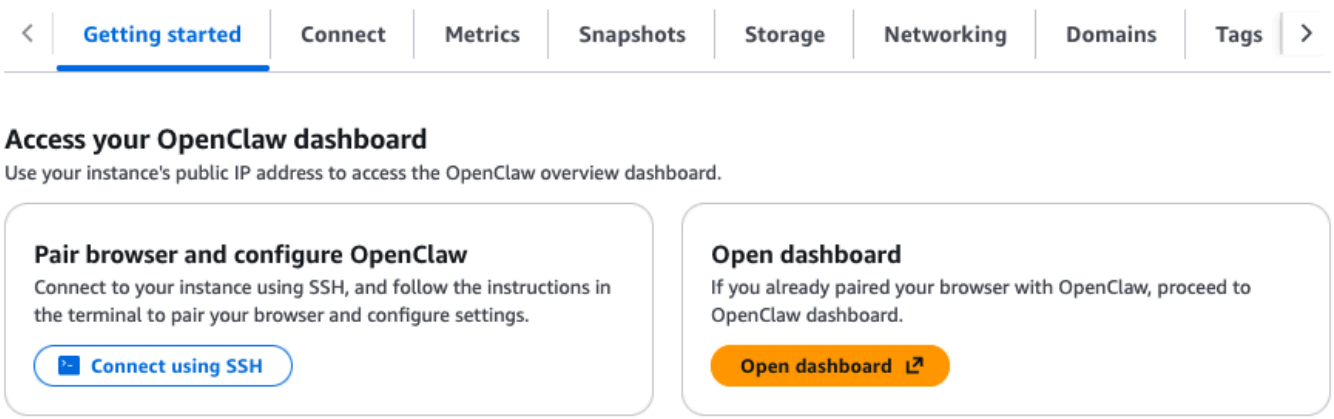
2단계: 브라우저를 OpenClaw와 페어링

OpenClaw 대시보드를 사용하려면 먼저 브라우저를 OpenClaw와 페어링해야 합니다. 이렇게 하면 브라우저 세션과 OpenClaw 간에 보안 연결이 생성됩니다.

> **Tip**
>
> OpenClaw 대시보드에 액세스하는 데 사용할 디바이스와 동일한 디바이스에서 브라우저를 준비합니다. 이 단계에서는에서 토큰을 복사Lightsail하여 OpenClaw 대시보드에 붙여 넣습니다. 인스턴스를 중지하고 시작하면 OpenClaw 인스턴스의 기본 퍼블릭 IP 주소가 변경됩니다. 인스턴스에 고정 IP 주소를 연결할 때 인스턴스를 중지했다가 시작한 경우에도 동일하게 유지됩니다. 자세한 내용은 Lightsail 리소스의 IP 주소 보기 및 관리를 참조하세요.

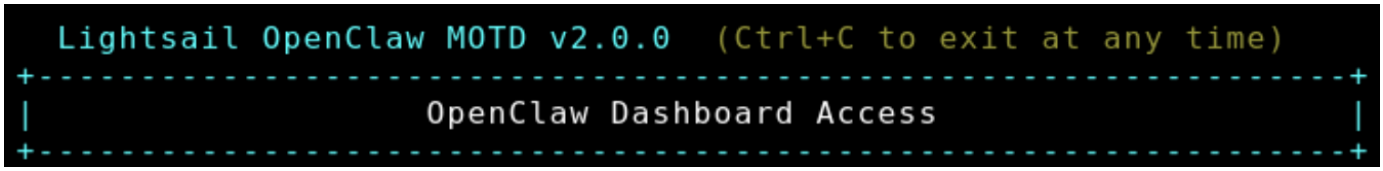
브라우저를 OpenClaw와 페어링하려면

1. Lightsail 콘솔의 인스턴스 섹션에서 OpenClaw 인스턴스의 이름을 선택하여 인스턴스 관리 페이지를 엽니다.
2. 시작하기 탭의 브라우저를 OpenClaw에 페어링에서 SSH를 사용하여 연결을 선택합니다. 브라우저 기반 SSH 터미널이 열립니다.

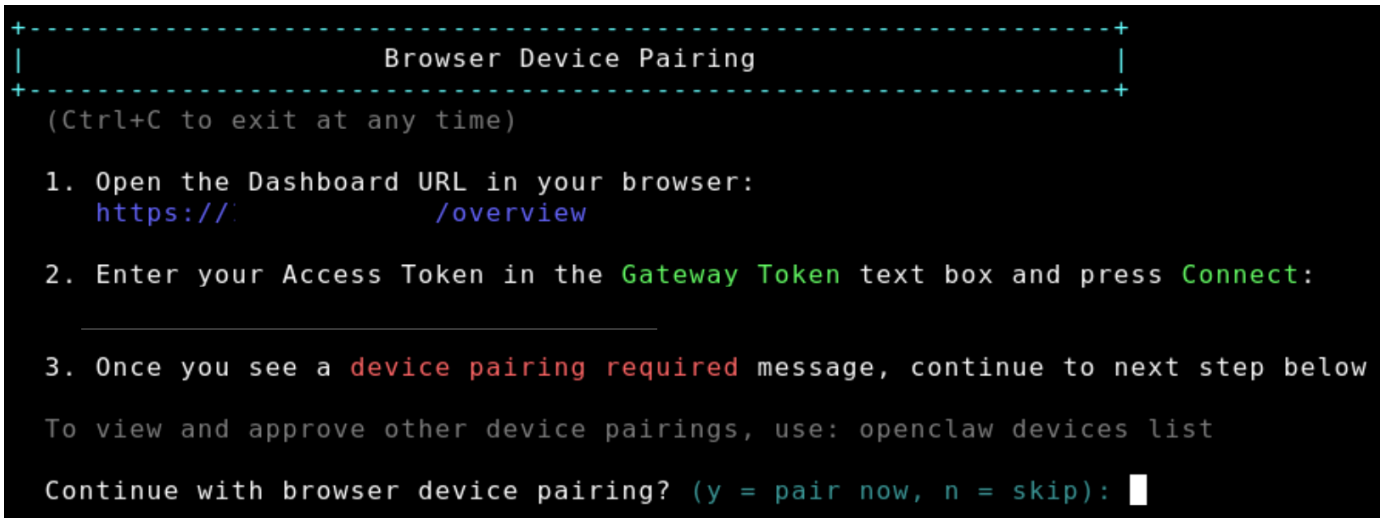


> **알고 계셨나요?**
>
> OpenClaw 인스턴스에서 실행되는 오늘의 메시지(MOTD) 서비스는 오리진 감지, 인증서 관리 및 토큰 교체를 비롯한 여러 자동 구성 작업을 관리합니다. SSH를 통해 인스턴스에 연결하여 MOTD 버전을 확인할 수 있습니다.

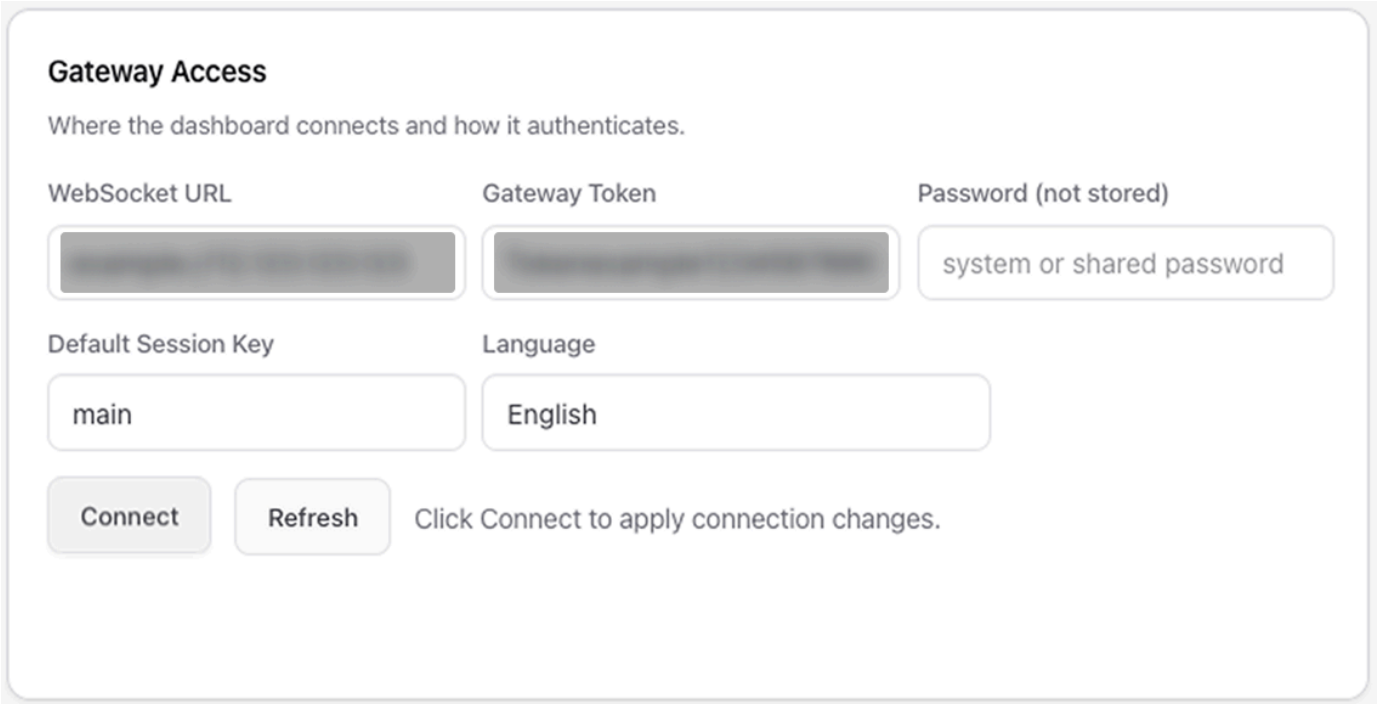
OpenClaw 인스턴스는 인스턴스의 IP 주소에서 연결을 수락하도록 게이트웨이를 자동으로 구성합니다. MOTD 버전 2.0.0에는 인스턴스 시작 중에 실행되고 허용된 오리진을 인스턴스의 현재 IP 주소로 구성하는 자동 오리진 감지 기능이 포함되어 있습니다. 인스턴스에 고정 IP 주소를 연결하면 시스템에서 대신 고정 IP 주소를 사용하도록 허용된 오리진을 자동으로 업데이트합니다.



3. SSH 터미널에서 시작 메시지에 표시된 대시보드 URL을 찾습니다. 이 URL을 복사하여 새 브라우저 탭에서 엽니다.

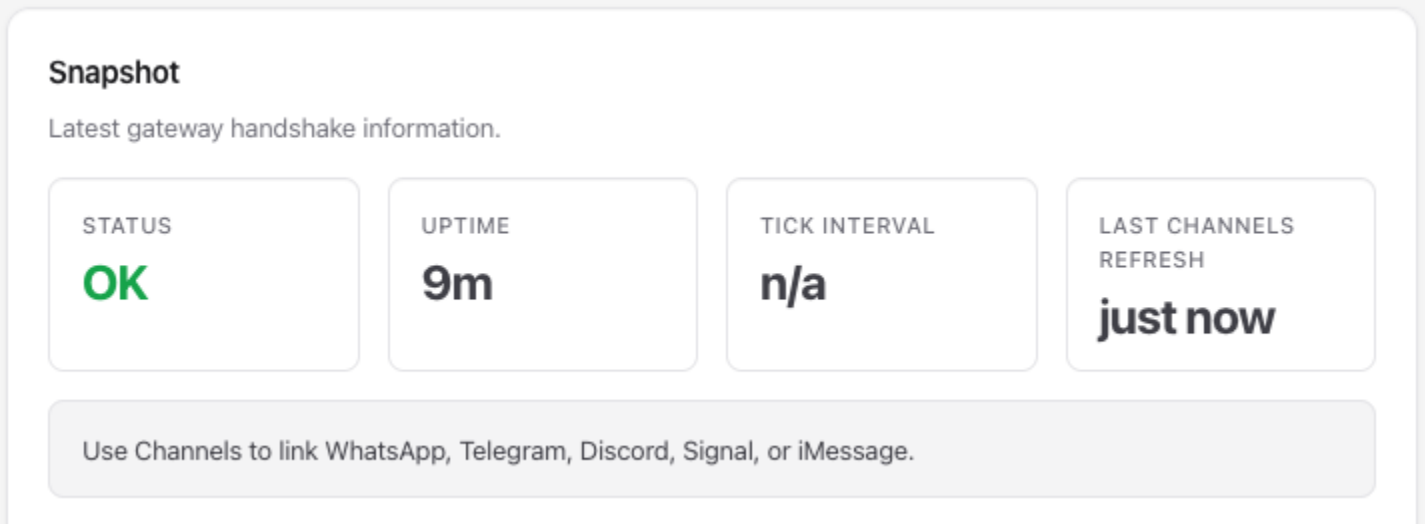


4. 열리는 OpenClaw 대시보드에서 게이트웨이 토큰 필드를 찾습니다.
5. SSH 터미널로 돌아가서 표시된 액세스 토큰을 복사합니다.
6. 복사한 액세스 토큰을 OpenClaw 대시보드의 게이트웨이 토큰 필드에 붙여넣은 다음 연결을 클릭합니다.



7. SSH 터미널로 돌아갑니다. 메시지가 표시되면 **y**를 눌러 디바이스 페어링을 계속합니다.
8. **a**를 눌러 디바이스 페어링 요청을 승인합니다.

페어링이 완료되면 OpenClaw 대시보드의 상태가 OK로 표시됩니다. 이제 브라우저가 OpenClaw 인스턴스에 연결되었습니다.



> **Tip**
>
> 나중에 추가 브라우저를 페어링해야 하는 경우 인스턴스에 SSH를 다시 연결하고 위의 페어링 단계를 반복합니다.

3단계: Amazon Bedrock으로 AI 기능 활성화

Lightsail OpenClaw 인스턴스는 Amazon Bedrock을 사용하여 AI 어시스턴트에 전원을 공급하도록 구성되어 있습니다. 이 단계에서는 Bedrock API를 호출하는 데 필요한 권한을 인스턴스에 부여합니다.

Bedrock API 액세스를 활성화하려면

1. OpenClaw 인스턴스 관리 페이지에서 시작하기 탭을 선택합니다.
2. Amazon Bedrock을 모델 공급자로 활성화에서 스크립트 복사 버튼을 클릭합니다. 그런 다음 CloudShell 시작 버튼을 클릭하여 CloudShell을 엽니다.

   > **설정 스크립트는 어떤 역할을 합니까?**
   >
   > 설정 스크립트는 다음 작업을 수행합니다.는 OpenClaw 인스턴스에 대해 특별히 IAM 역할을 생성하고, Amazon Bedrock APIs에 대한 액세스 권한을 부여하는 정책을 연결하고, AWS Marketplace 권한(타사 모델에 필요)을 부여하는 정책을 연결하고,이 역할을 사용하도록 인스턴스 프로파일을 구성합니다. 스크립트를 실행한 후 IAM 콘솔에서 IAM 정책 세부 정보를 검토할 수 있습니다. IAM 역할의 이름은 입니다`LightsailRoleFor-[your-instance-id]`.

3. 복사한 명령을 CloudShell 터미널에 붙여넣고를 누릅니다Enter.
4. 작업이 완료될 때까지 기다립니다. 출력에 완료가 표시되면 권한이 성공적으로 적용된 것입니다.

이 단계가 완료되면 OpenClaw 대시보드에서 채팅으로 이동하여 AI 어시스턴트 사용을 시작합니다.

참고: Lightsail OpenClaw 인스턴스는 기본적으로 Anthropic Claude Sonnet 4.6을 사용합니다. Amazon Bedrock에서 Anthropic 모델을 처음 사용하는 경우 액세스를 얻으려면 최초 사용(FTU) 양식을 작성해야 합니다. Anthropic 모델에 액세스하는 방법에 대해 자세히 알아봅니다.

4단계: 메시징 채널 연결(선택 사항)

텔레그램 및 WhatsApp과 같은 메시징 앱으로 작업하도록 OpenClaw를 확장할 수 있으므로 전화 또는 메시징 클라이언트에서 직접 AI 어시스턴트와 상호 작용할 수 있습니다. OpenClaw를 메시징 채널에 연결하려면 먼저 브라우저를 OpenClaw와 페어링해야 합니다(2단계 참조).

텔레그램 연결

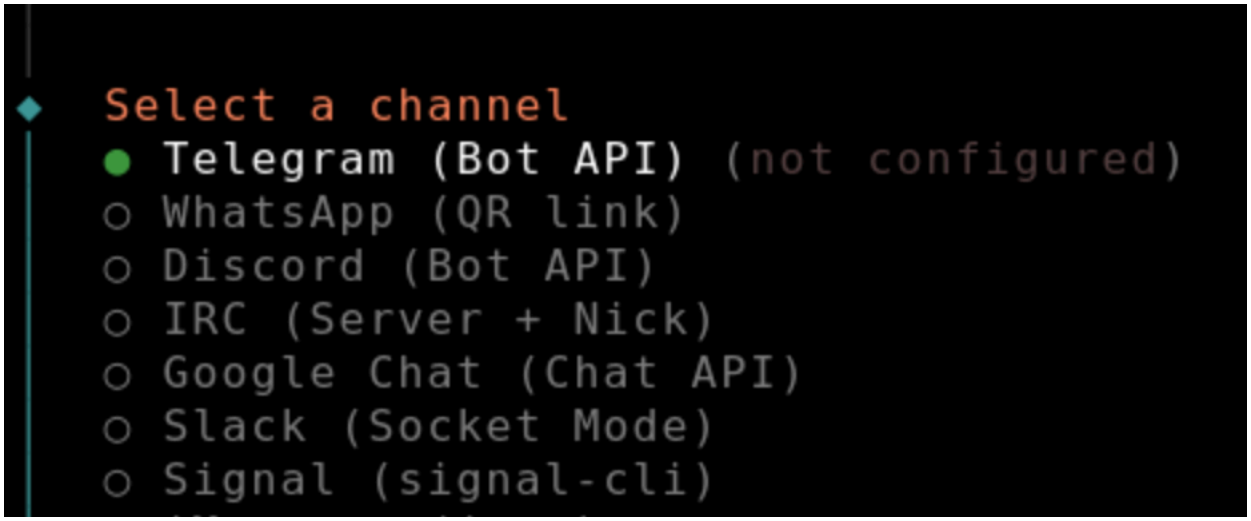
텔레그램 채널을 추가하려면

1. 텔레그램을 열고를 검색합니다`@BotFather`.
2. 명령을 보내고 프롬프트에 `/newbot` 따라 새 봇을 생성합니다. BotFather는 봇 토큰과 봇에 대한 딥 링크를 제공합니다.

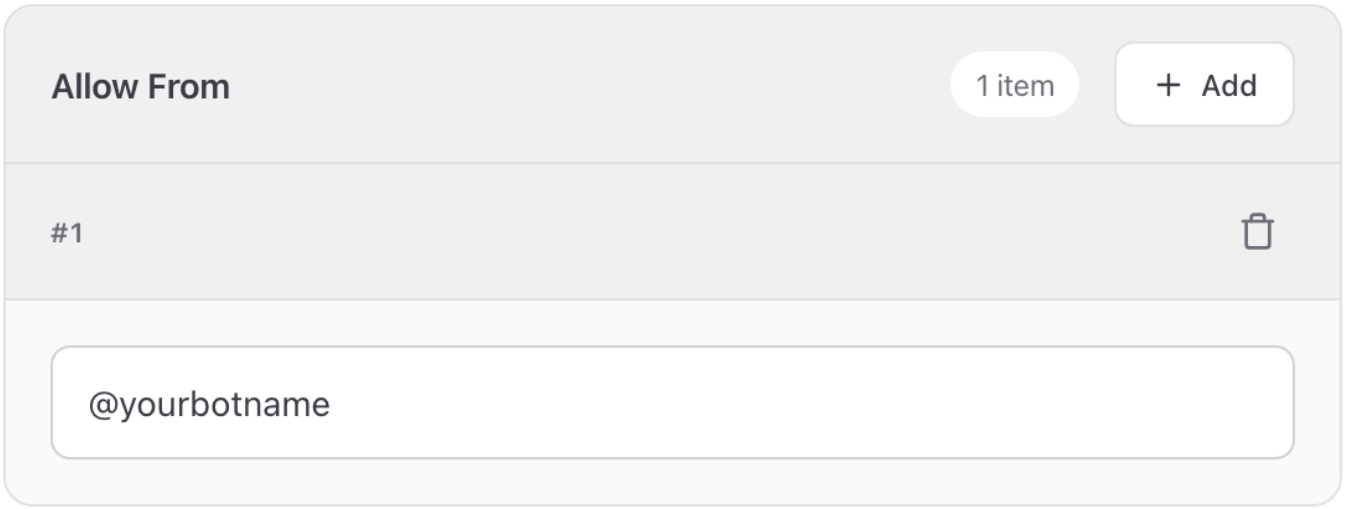
3. SSH를 사용하여 OpenClaw 인스턴스에 연결합니다. 브라우저 기반 SSH 터미널이 열립니다.
4. OpenClaw 인스턴스에 연결된 SSH 터미널에서 다음을 실행합니다.

```
openclaw channels add
```

5. 사용 가능한 채널 목록에서 텔레그램을 선택합니다.

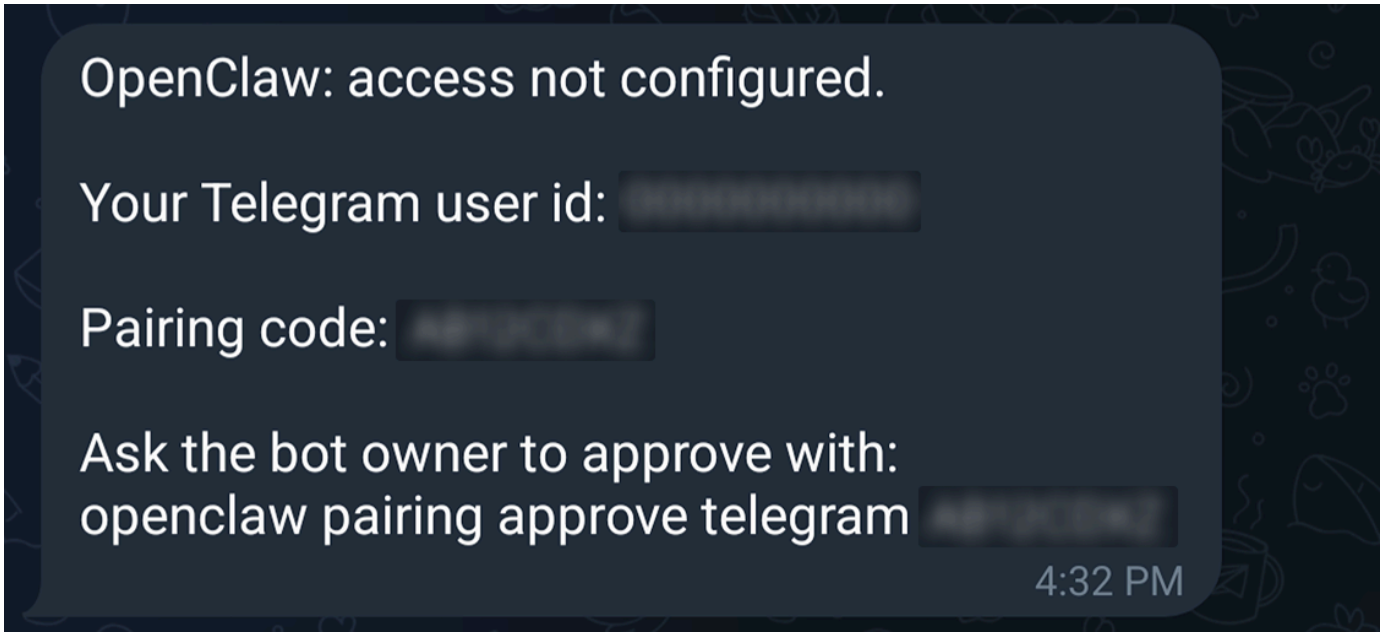


6. 메시지가 표시되면 2단계에서 BotFather에서 받은 봇 토큰을 입력합니다.
7. OpenClaw 대시보드에서 채널 섹션으로 이동하여 텔레그램 사용자 ID를 허용 목록에 추가합니다.

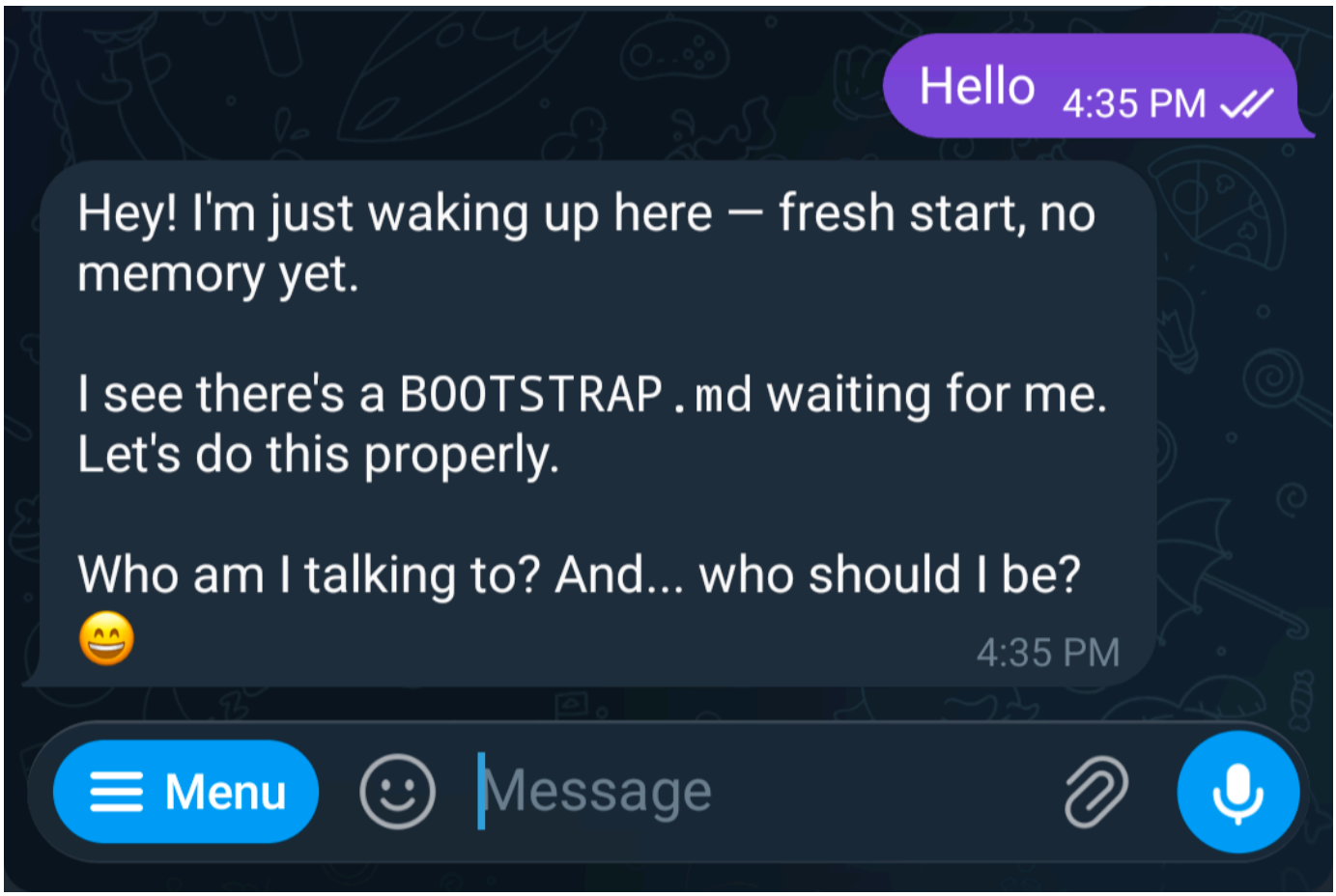


8. 2단계의 텔레그램에서 봇에 메시지를 전송하여 통합을 테스트합니다.
9. 텔레그램에 OpenClaw 페어링을 승인하라는 메시지가 표시됩니다. SSH 터미널에서 다음을 실행합니다.

```
openclaw pairing approve telegram [pairing code]
```



10. 2단계의 텔레그램에서 생성한 봇에 메시지를 전송하여 통합을 다시 테스트합니다.



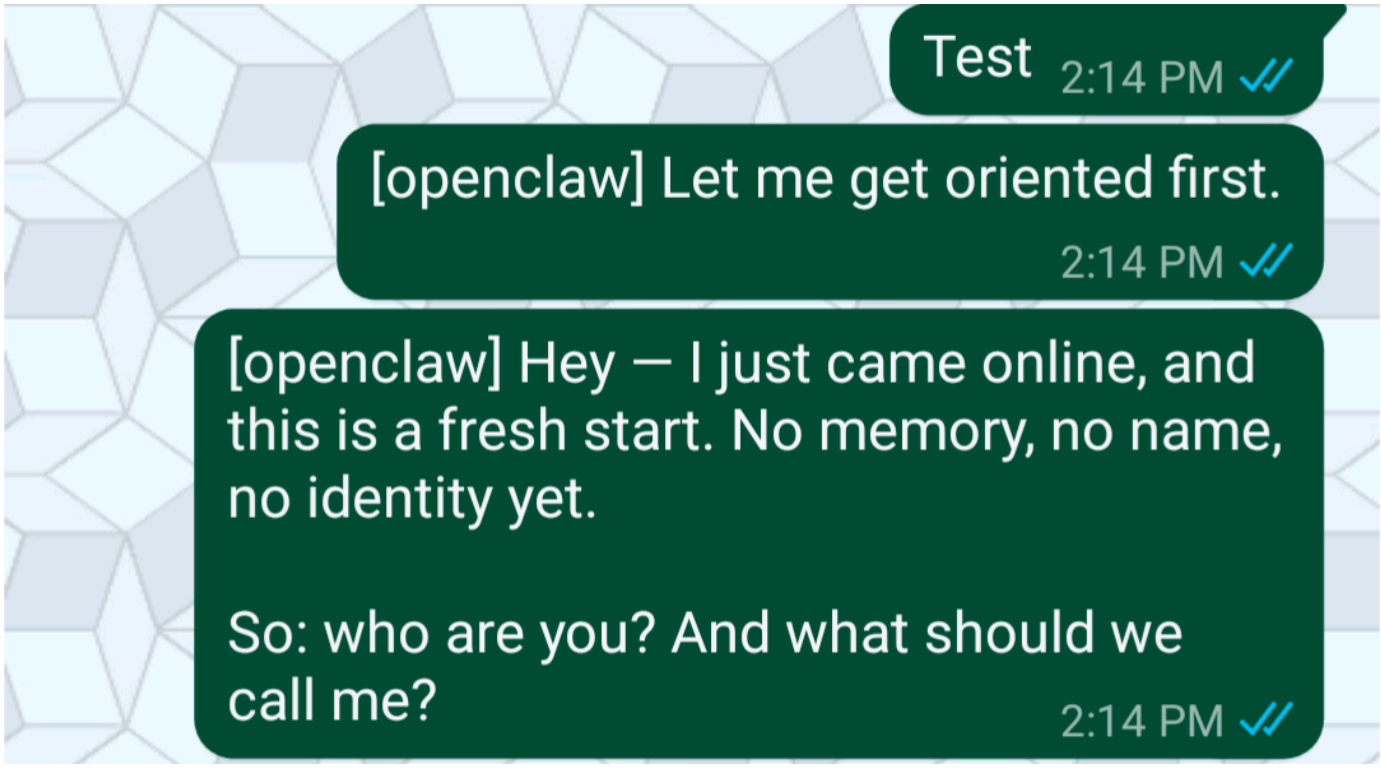
WhatsApp 연결

WhatsApp 채널을 추가하려면

1. SSH를 사용하여 OpenClaw 인스턴스에 연결합니다. 브라우저 기반 SSH 터미널이 열립니다.
2. OpenClaw 인스턴스에 연결된 SSH 터미널에서 다음을 실행합니다.

```
openclaw channels add
```

3. 사용 가능한 채널 목록에서 WhatsApp을 선택합니다.
4. 화면에 표시되는 지시 사항을 따릅니다. QR 코드가 터미널에 표시됩니다.
5. 휴대폰에서 WhatsApp을 열고 연결된 디바이스 기능을 사용하여 QR 코드를 스캔합니다.
6. 휴대폰에서 페어링을 완료합니다.
7. 위의 단계에서 페어링한 연락처 번호로 메시지를 보내 WhatsApp을 통해 직접 OpenClaw 어시스턴트에 메시지를 보내 통합을 테스트합니다.



5단계: 인스턴스의 스냅샷 생성(선택 사항)

설정을 완료한 후에는 OpenClaw 인스턴스의 스냅샷을 생성하는 것이 좋습니다. 스냅샷은 인스턴스를 잘 알려진 상태에서 복원하여 신뢰할 수 있는 복구 메커니즘을 제공하는 point-in-time 백업입니다. 스냅샷에서 원하는 계획의 새 인스턴스를 생성할 수도 있습니다.

수동 스냅샷을 생성하는 방법

1. Lightsail 콘솔의 인스턴스 섹션에서 OpenClaw 인스턴스의 이름을 선택합니다.
2. 스냅샷 탭을 선택합니다.
3. 수동 스냅샷에서 스냅샷 생성을 선택합니다.
4. 스냅샷 이름을 입력하고 생성을 선택합니다.

> 알고 계셨나요?
>
> Lightsail는 일별 스냅샷 7개를 저장하고 인스턴스에 대해 자동 스냅샷을 활성화하면 가장 오래된 스냅샷을 최신 스냅샷으로 자동 대체합니다. 자세한 내용은 Lightsail 인스턴스 및 디스크의 자동 스냅샷 구성을 참조하세요.

자주 묻는 질문(FAQ)

추가 브라우저를 OpenClaw 인스턴스와 페어링하려면 어떻게 해야 하나요?

필요한 만큼 브라우저를 페어링할 수 있습니다. 새 브라우저를 페어링하려면 SSH를 OpenClaw 인스턴스에 다시 연결합니다(Lightsail콘솔의 인스턴스 관리 페이지에서 SSH를 사용하여 연결을 선택합니다). 2단계와 동일한 페어링 단계를 따릅니다. 대시보드 URL로 이동하여 터미널에서 액세스 토큰을 복사하고 새 브라우저의 OpenClaw 대시보드에 있는 게이트웨이 토큰 필드에 붙여 넣은 다음 페어링 요청을 승인합니다.

OpenClaw 인스턴스에 부여된 IAM 권한을 사용자 지정할 수 있나요?

예. 3단계의 설정 스크립트는 Amazon Bedrock에 대한 액세스 권한을 부여하는 정책을 사용하여 IAM 역할을 생성합니다. 언제든지이 정책을 보거나 편집하거나 제한할 수 있습니다.

- IAM 콘솔을 열고 역할로 이동합니다.
- 와 같이 OpenClaw 인스턴스에 대해 생성된 역할을 찾습니다`LightsailRoleFor-i-0d15d5483571b95bb`.

- 역할을 선택하여 연결된 정책을 봅니다.
- 권한을 편집할 정책 이름을 선택합니다.

권한을 수정할 때 주의하십시오. 필요한 Bedrock 권한을 제거하면 OpenClaw가 AI 응답을 생성할 수 없습니다. 자세한 내용은 AWS 설명서의 IAM 정책을 참조하세요.

OpenClaw에 질문할 수 있습니까?

예, OpenClaw의 기본 제공 채팅 도우미는 OpenClaw 자체에 대한 질문에 답변할 수 있습니다. OpenClaw에서 수행할 수 있는 작업이 확실하지 않은 경우 채팅 인터페이스에서 직접 요청하면 됩니다. 예를 들어 다음을 입력할 수 있습니다.

- “어떻게 도와드릴까요?”
- "OpenClaw에 연결할 수 있는 채널은 무엇입니까?"
- “새 메시징 채널을 추가하려면 어떻게 해야 하나요?”

OpenClaw는 기능에 따라 지침에 따라 응답합니다. 이는 OpenClaw 대시보드를 벗어나지 않고 기능을 탐색하는 좋은 방법입니다.

참고: Amazon Bedrock을 사용하여 AI 기능 활성화 단계(시작 안내서의 3단계)를 완료하거나 채팅이 작동하도록 자체 모델 공급자를 구성해야 합니다. Bedrock 설정에는 인스턴스의 시작하기 탭에서 원클릭 스크립트를 실행하여 필요한 권한을 부여하고, Anthropic 모델을 처음 사용하는 경우 Amazon Bedrock 콘솔에 간단한 최초 사용 양식을 제출하는 작업이 포함됩니다. 이 단계가 없으면 채팅 인터페이스에 연결할 AI 모델이 없습니다.

OpenClaw를 실행하는 데 드는 Lightsail 비용은 얼마인가요?

다음은 비용 분석입니다.

- Lightsail 인스턴스 - 선택한 인스턴스 요금제(예: 4GB 요금제)에 대해 요금을 지불합니다. Lightsail 요금제에는 온디맨드 시간당 요금이 청구되므로 사용한 만큼만 요금을 지불합니다. 고객이 이용하는 모든 Lightsail 플랜에 대해 월간 최대 요금까지 고정된 시간당 가격으로 계산해 요금이 청구됩니다.
- AI 모델 사용량(토큰) - OpenClaw 어시스턴트와 주고받는 모든 메시지는 토큰 기반 요금 모델을 사용하여 Amazon Bedrock을 통해 처리됩니다. 비용은 모델에 따라 다릅니다. 일부 모델은 다른 모델보다 토큰당 비용이 더 많이 듭니다.

- 타사 모델 구독 - AWS Marketplace를 통해 배포된 타사 모델(예: Anthropic Claude 또는 Cohere)을 선택하는 경우 토큰당 비용 외에 추가 소프트웨어 요금이 발생할 수 있습니다. 이는 청구서의 AWS Marketplace 아래에 별도의 품목으로 표시됩니다.
- 데이터 전송 초과 - 각 Lightsail 플랜에는 월별 데이터 전송 허용량이 포함됩니다. OpenClaw 인스턴스가 플랜에 포함된 것보다 많은 데이터를 보내거나 받는 경우 데이터 전송에 초과 요금이 적용됩니다.
- 스냅샷 - Lightsail 인스턴스의 수동 및 자동 스냅샷은 사용된 스토리지 양에 따라 요금이 청구됩니다.

Anthropic 모델을 사용하고 싶습니다. 추가로 해야 할 일이 있나요?

Anthropic에는 표준 권한 외에도 한 가지 추가 요구 사항이 있습니다. Anthropic 모델을 처음 호출하기 전에 최초 사용(FTU) 양식을 작성해야 합니다. 이는 Anthropic 요구 사항이며 AWS 계정당 한 번 또는 AWS Organization의 관리 계정 수준에서 한 번 적용되며, 조직의 모든 멤버 계정에서 상속됩니다. OpenClaw 인스턴스는 기본적으로 Anthropic Claude-Sonnet 4.6을 사용합니다.

Lightsail는 3단계 설정의 일부로 기본 IAM 및 Marketplace 권한을 처리합니다. CloudShell 스크립트는 세 가지 필수 AWS Marketplace 권한(`aws-marketplace:Subscribe`, `aws-marketplace:Unsubscribe`및)을 포함하는 IAM 역할을 생성합니다`aws-marketplace:ViewSubscriptions`. 이는 Amazon Bedrock이 타사 모델을 처음 호출할 때 자동으로 활성화하는 데 필요합니다. 계정에서 모델이 활성화되면 계정의 모든 사용자가 Marketplace 권한 없이 호출할 수 있습니다. 구독은 한 번만 수행하면 됩니다.

Anthropic별 FTU 요구 사항을 완료하려면:

- Amazon Bedrock 콘솔을 엽니다.
- 모델 카탈로그로 이동하여 Anthropic 모델(예: Claude)을 선택합니다.
- 사용 사례 세부 정보를 제출하라는 메시지가 표시됩니다. 양식을 작성하고 제출합니다.

Anthropic 모델에 대한 액세스 권한은 양식이 성공적으로 제출된 직후 부여됩니다. 완료되면 OpenClaw 대시보드에서 Anthropic 모델을 선택하고 즉시 사용할 수 있습니다.

> **Note**
>
> Amazon, Meta, Mistral AI, DeepSeek 및 Qwen의 모델은 AWS Marketplace를 통해 판매되지 않으며이 단계가 필요하지 않습니다.

자세한 내용은 Amazon Bedrock 사용 설명서의 Amazon Bedrock 파운데이션 모델 액세스를 참조하세요.

HTTPS는 OpenClaw 인스턴스에서 어떻게 작동하나요?

OpenClaw 인스턴스에는 Let's Encrypt 인증서로 보호되는 HTTPS 엔드포인트가 내장되어 있습니다. 인스턴스가 생성되면 인스턴스의 IPv4 주소에 대해 Let's Encrypt 인증서가 자동으로 발급되므로 수동 설정이 필요하지 않습니다.

인스턴스의 IP 주소가 변경되면 SSL 인증서는 어떻게 되나요?

OpenClaw 인스턴스에는 인스턴스의 IP 주소를 모니터링하는 내장 인증서 관리 데몬(`lightsail-manage-certd`)이 포함되어 있습니다. 예를 들어 고정 IP를 연결하거나 분리할 때 IP 주소가 변경되면 데몬은 자동으로 변경 사항을 감지하고 새 IP 주소에 대해 새 Let's Encrypt 인증서를 발급합니다. SSL 인증서에는 수동 작업이 필요하지 않습니다.

참고: 게이트웨이 액세스 토큰은 동일하게 유지되지만 2단계: 브라우저를 OpenClaw와 페어링의 단계에 따라 브라우저를 다시 페어링해야 합니다.

SSL 인증서는 얼마나 자주 갱신되나요?

OpenClaw 인스턴스에 대해 발급된 Let's Encrypt 인증서는 7일 동안 유효합니다. 인증서 관리 데몬은 인증서가 만료되기 2일 전에 인증서를 자동으로 갱신하므로 인스턴스는 중단이나 수동 개입 없이 보안을 유지합니다.

OpenClaw에 플러그인을 설치할 수 있습니까?

예. OpenClaw는 플러그인 설치를 지원하며, 일부 플러그인 또는 구성 변경 사항을 적용하려면 OpenClaw 게이트웨이 서비스를 수동으로 다시 시작해야 할 수 있습니다.

플러그인을 설치하거나 구성을 업데이트한 후 게이트웨이를 관리하려면 OpenClaw 인스턴스에 SSH를 설치하고 다음 명령을 사용합니다.

- OpenClaw 게이트웨이 서비스를 중지합니다. `openclaw gateway stop`
- OpenClaw 게이트웨이 서비스를 시작합니다. `openclaw gateway start`
- 서비스의 현재 상태를 확인합니다. `openclaw gateway status`

참고:MOTD 1.0.0(OpenClaw 2026.2.17)을 사용하는 경우 대신 다음 명령을 사용합니다.

- OpenClaw 게이트웨이 서비스를 중지합니다. `sudo systemctl stop openclaw-gateway`
- OpenClaw 게이트웨이 서비스를 시작합니다. `sudo systemctl start openclaw-gateway`

- 서비스의 현재 상태를 확인합니다. `sudo systemctl status openclaw-gateway`

게이트웨이 토큰이 손상되면 어떻게 되나요?

토큰이 노출되는 경우, 예를 들어 로그에 누출되거나, 실수로 공유되거나, 프롬프트 주입 공격을 통해 노출된 경우 토큰을 보유한 사람은 누구나 수동으로 재생성할 때까지 OpenClaw 게이트웨이에 액세스할 수 있습니다.

게이트웨이 토큰이 자동으로 교체되나요?

예. 매일 3:00 UTC에 자동으로 교체됩니다. 이렇게 교체하려면 브라우저를 OpenClaw 인스턴스와 다시 페어링해야 합니다.

게이트웨이 토큰을 수동으로 교체하려면 어떻게 해야 합니까?

게이트웨이 토큰을 교체하려면:

- Lightsail 콘솔에서 OpenClaw 인스턴스로 SSH.
- 다음 명령을 실행하여 토큰을 다시 생성합니다.

```
openclaw token rotate
```

- 이전 토큰은 즉시 무효화됩니다. 현재 이전 토큰과 페어링된 모든 브라우저 또는 클라이언트의 연결이 해제됩니다.
- 2단계: 브라우저를 OpenClaw와 페어링의 단계에 따라 새 토큰을 사용하여 브라우저를 다시 페어링합니다.

> **Tip**
>
> 토큰을 교체한 후 사용을 재개하기 전에 신뢰할 수 있는 모든 디바이스가 다시 페어링되었는지 확인합니다.

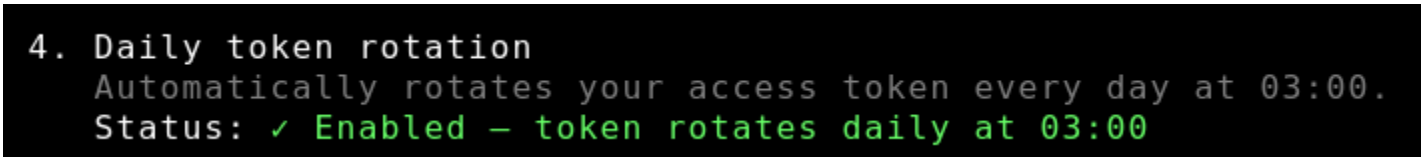
자동 토큰 교체는 어떻게 작동하나요?

MOTD 2.0.0에는 게이트웨이 액세스 토큰을 매일 교체하여 보안을 강화하는 자동 토큰 교체가 포함되어 있습니다.

중요한 영향:

- 토큰이 자동으로 교체되면 페어링된 모든 브라우저와 디바이스의 연결이 해제됩니다.
- 2단계: 브라우저를 OpenClaw와 페어링의 단계에 따라 브라우저를 다시 페어링해야 합니다.

토큰을 교체하지 않으려면 보안 설정을 변경하여 MOTD에서 토큰을 비활성화할 수 있습니다.



메시징 채널 자격 증명(Telegram, WhatsApp, Slack)을 교체하려면 어떻게 해야 하나요?

텔레그램 봇 토큰, WhatsApp 세션 자격 증명 또는 Slack 토큰과 같이 OpenClaw 인스턴스에 저장된 메시징 플랫폼 토큰 또는 자격 증명이 손상된 경우 메시징 채널에 대한 무단 액세스를 방지하기 위해 즉시 교체해야 합니다.

연결된 채널의 자격 증명은 인스턴스의 `~/.openclaw/credentials/`에 저장됩니다. 자격 증명을 교체하려면:

- 소스에서 손상된 토큰을 취소합니다.
  - 텔레그램: 텔레그램, 메시지 `@BotFather`를 열고 `/revoke`를 사용하여 기존 봇 토큰을 무효화하고 새 봇 토큰을 생성합니다.
  - WhatsApp: 휴대폰의 WhatsApp에서 연결된 디바이스 세션을 로그아웃합니다(`Settings` → `Linked Devices` → OpenClaw 세션 선택 → `Log out`). 그런 다음 QR 코드 페어링 흐름을 사용하여 다시 연결합니다.
- OpenClaw 인스턴스에서 자격 증명을 업데이트합니다. SSH를 인스턴스로 업데이트하고 다음을 실행합니다.

  ```
  openclaw channels update
  ```

  업데이트할 채널을 선택하고 메시지가 표시되면 새 토큰 또는 자격 증명을 입력합니다.
- 업데이트된 채널을 통해 테스트 메시지를 전송하여 채널이 작동하는지 확인합니다.

> **Note**
>
> 메시징 자격 증명을 교체해도 게이트웨이 토큰 또는 기타 연결된 채널에는 영향을 주지 않습니다. 각 자격 증명은 독립적으로 관리됩니다.

`setup-lightsail-openclaw-bedrock-role.sh` 스크립트는 무엇을 하나요?

OpenClaw 인스턴스만 Amazon Bedrock 및 AWS Marketplace를 통해 사용할 수 있는 기본 모델을 사용하도록 허용하는 IAM 역할을 생성합니다.

스냅샷에서 OpenClaw 인스턴스를 복원하려면 어떻게 해야 합니까?

- 기존 OpenClaw 스냅샷에서 새 인스턴스를 생성합니다. 자세한 내용은 스냅샷에서 인스턴스 생성을 참조하세요.
- Lightsail 콘솔에서 새 OpenClaw 인스턴스로 SSH
- 다음 명령을 실행하여 인스턴스의 Lightsail 인스턴스 ID를 가져옵니다. 예: *i-1234567890abcdef1*:

```
TOKEN=`curl -s -X PUT "http://169.254.169.254/latest/api/token" -H "X-aws-ec2-
metadata-token-ttl-seconds: 21600"` && curl -w "\n" -s -H "X-aws-ec2-metadata-token:
 $TOKEN" http://169.254.169.254/latest/meta-data/instance-id
```

- 다음 명령을 실행하여 인스턴스와 연결된 IAM 역할을 가져옵니다.

```
grep 'role_arn' /home/ubuntu/.aws/config | head -1 | awk '{print $3}'
```

- IAM 콘솔의 이전 단계에서 검색한 역할을 찾습니다. 예: `LightsailRoleFor-i-0d15d5483571b95bb`
- `Trust relationships` 선택
- `Edit trust policy` 선택
- *"arn:aws:sts::0123456789012:assumed-role/Amazon LightsailInstance/i-1234567890abcdef1"*과 같이 앞서 검색한 인스턴스 ID의 ARN으로 신뢰 정책을 업데이트합니다.
- `Update policy` 선택

OpenClaw 인스턴스에 대해 AllowedOrigin을 구성하려면 어떻게 해야 합니까?

AllowedOrigin은 OpenClaw 게이트웨이에 연결할 수 있는 웹 주소(오리진)를 제어하는 보안 설정입니다. 이렇게 하면 권한이 없는 웹 사이트가 인스턴스에 액세스하는 것을 방지하고 교차 오리진 보안 문제로부터 보호할 수 있습니다.

MOTD 2.0.0(OpenClaw 2026.3.2 이상): AllowedOrigin은 MOTD 서비스에 의해 자동으로 관리됩니다. 인스턴스가 시작되거나 IP 주소가 변경되면 서비스가 올바른 오리진을 자동으로 감지하고 구성을 업데이트합니다. 수동 작업은 필요하지 않습니다.

MOTD 1.0.0(OpenClaw 2026.2.17): 특정 도메인에서 OpenClaw에 액세스하는 경우 AllowedOrigin을 수동으로 구성해야 합니다. 인스턴스에 SSH를 추가하고 아래 지침에 따라 OpenClaw 구성 파일을 편집하여 허용되는 오리진을 추가합니다.

- Lightsail 콘솔에서 OpenClaw 인스턴스로 SSH
- 구성 파일을 엽니다. `~/.openclaw/openclaw.json`
- AllowedOrigin 설정을 추가하거나 수정합니다.

```
{
    "gateway": {
        "controlUi": {
            "allowedOrigins": [
                "https://<your-domain.com>"
            ]
        }
    }
}
```

- OpenClaw 게이트웨이 서비스를 다시 시작합니다. `sudo systemctl restart openclaw-gateway`

OpenClaw를 최신 버전으로 업데이트하려면 어떻게 해야 하나요?

OpenClaw 게이트웨이를 최신 버전으로 업데이트하려면:

- OpenClaw 인스턴스로의 SSH
- update 명령을 실행합니다. `sudo openclaw update`

중요 참고 사항:

- OpenClaw 블루프린트는 인스턴스에 게이트웨이를 전역적으로 설치하므로 sudo 권한이 필요합니다.
- 업데이트 명령이 완료되면 루트 사용자로 게이트웨이를 시작하려고 시도하지만 게이트웨이가 ubuntu 사용자로 실행되어야 하므로 실패합니다.
- 업데이트 후 게이트웨이를 수동으로 다시 시작해야 합니다. `openclaw gateway restart`

- OpenClaw 제어 UI 대시보드의 "업데이트" 버튼에는 sudo 권한이 없으므로 작동하지 않습니다.

고정 IP 주소를 연결하면 디바이스 페어링은 어떻게 되나요?

OpenClaw 인스턴스에 고정 IP 주소를 연결하면 인스턴스의 IP 주소가 변경됩니다. 이는 디바이스 페어링에 중요한 영향을 미칩니다.

- 이전에 페어링된 모든 브라우저와 디바이스의 연결이 끊어집니다.
- 게이트웨이 토큰은 유효한 상태로 유지되지만 연결 엔드포인트가 변경되었습니다.
- 고정 IP를 연결한 후 모든 브라우저와 디바이스를 다시 명시적으로 페어링해야 합니다.

디바이스를 다시 페어링하려면:

- 인스턴스에 대한 SSH(SSH 연결은 새 정적 IP와 함께 작동함)
- 2단계의 페어링 단계에 따라 각 브라우저를 다시 연결합니다.
- 메시징 채널(Telegram, WhatsApp)의 경우 페어링을 다시 승인해야 할 수도 있습니다.

도구를 활성화하기 위한 샌드박스 권한을 부여하려면 어떻게 해야 합니까?

기본적으로 OpenClaw는 격리된 Docker 컨테이너 환경(샌드박스)에서 도구와 플러그인을 실행하여 잠재적으로 유해한 작업으로부터 인스턴스를 보호합니다. 이 격리는 시스템 명령, 파일 시스템 액세스, 네트워크 연결, 호스트 시스템 리소스를 포함하여 액세스할 수 있는 도구를 제한합니다.

이렇게 하면 강력한 보안이 제공되지만 일부 도구가 제대로 작동하려면 덜 제한적인 설정이 필요할 수 있습니다. 예를 들어 웹 스크레이핑 도구에는 네트워크 액세스가 필요하고 파일 관리 도구에는 더 광범위한 파일 시스템 액세스가 필요합니다. 이러한 권한이 없으면 샌드박스는 주로 기능이 제한된 기본 챗봇 역할을 합니다.

샌드박스의 제한을 줄이려면:

- Lightsail 콘솔에서 OpenClaw 인스턴스로 SSH
- 다음 명령을 실행하여 도구 실행 설정을 구성합니다.

```
openclaw config set tools.exec.host gateway
openclaw config set tools.exec.ask off
openclaw config set tools.exec.security full
```

- 변경 사항을 적용하려면 OpenClaw 게이트웨이 서비스를 다시 시작합니다.

```
openclaw gateway restart
```

이러한 설정이 수행하는 작업:

- `tools.exec.host gateway` - 도구가 격리된 Docker 컨테이너 대신 게이트웨이 호스트에서 직접 실행되도록 허용하여 시스템 명령 및 리소스에 대한 액세스 권한을 부여합니다.
- `tools.exec.ask off` - 도구 실행 전에 권한 프롬프트를 비활성화하여 수동 승인 없이 도구가 자동으로 실행되도록 합니다.
- `tools.exec.security full` - 도구 실행의 보안 수준을 설정합니다.

보안 고려 사항: 이러한 설정은 도구와 시스템 간의 격리를 크게 줄입니다. 사용 중인 도구를 신뢰하는 경우에만 이러한 설정을 구성합니다. 덜 제한적인 샌드박스 설정으로 도구를 실행하면 도구가 손상되거나 악의적인 경우 인스턴스가 보안 위험에 노출될 수 있습니다.

MOTD 버전 간의 차이점은 무엇입니까?

OpenClaw 인스턴스는 언제 생성되었는지에 따라 다른 MOTD(Message of the Day) 버전을 사용합니다. 알아야 할 사항은 다음과 같습니다.

MOTD 1.0.0(OpenClaw 2026.2.17):

- 게이트웨이 관리: 사용 `sudo systemctl start/stop/status openclaw-gateway`
- 토큰 교체: 수동 전용(사용`openclaw token rotate`)

MOTD 2.0.0(OpenClaw 2026.3.2 이상):

- 게이트웨이 관리: 간소화된 명령 사용 `openclaw gateway start/stop/status`
- 토큰 교체: 자동 일일 교체

MOTD 버전을 확인하는 방법: 인스턴스에 SSH를 넣고 표시된 환영 메시지를 확인합니다. MOTD 버전이 상단에 표시됩니다.

Lightsail에 Plesk 호스팅 스택 배포

> **알고 계셨나요?**
>
> Lightsail는 일별 스냅샷 7개를 저장하고 인스턴스에 대해 자동 스냅샷을 활성화하면 가장 오래된 스냅샷을 최신 스냅샷으로 자동 대체합니다. 자세한 내용은 Lightsail 인스턴스 및 디스크의 자동 스냅샷 구성을 참조하세요.

Amazon Lightsail에서 Plesk 인스턴스를 생성하는 방법과 사용자 이름과 암호를 만들어 Plesk 사용자 인터페이스에 처음으로 로그인하는 방법을 보여줍니다. 또한 Plesk 인스턴스가 실행된 후 이에 연결하고 구성하는 방법을 알아봅니다.

> **Important**
>
> Plesk Hosting Stack on Ubuntu(BYOL) 블루프린트로 시작된 인스턴스에는 30일 평가판 라이선스가 있습니다. 30일이 지나면 Plesk에서 라이선스를 구매하고 설치해야 Plesk 애플리케이션을 계속 사용할 수 있습니다. 자세한 내용은 8단계: Plesk 라이선스 구매 단원을 참조하십시오.

Lightsail에서 Plesk 호스팅 스택을 사용하면 다음과 같은 작업을 수행할 수 있습니다.

- WP Toolkit의 그래픽 인터페이스를 사용하여 WordPress 사이트 관리 자동화
- 무료 SSL 인증서로 사이트를 보호하고 Let's Encrypt를 사용하여 HTTPS 트래픽 구성
- FTP를 사용하여 인스턴스와 파일 주고받기
- Plesk Firewall, Logs 및 ModSecurity를 포함한 웹 기반 도구를 사용하여 서버 모니터링 및 보호
- Docker 프록시 규칙을 사용하여 인바운드 트래픽을 특정 도메인에서 지정된 컨테이너로 라우팅

Plesk 호스팅 스택 배포 시 고려 사항

Plesk 인스턴스 배포를 시작하기 전에 웹 사이트 도메인을 등록해야 하는지 여부와 관리 방법을 결정합니다. IP 주소가 아닌 논리명(예: `http://example.com`)으로 웹 사이트에 액세스하려면 등록된 도메인이 필요합니다. 또한 인증서(HTTPS 트래픽에 필요)를 사용하여 웹 사이트에 대한 연결을 보호하려면 도메인 내에 DNS 이름을 생성해야 합니다.

도메인 구성 및 관리에 대한 다음 옵션을 검토합니다.

- Lightsail에 사용하려는 등록 도메인이 이미 있는 경우, 이 자습서의 단계를 시작할 수 있습니다.
- DNS 레코드 관리를 대해 선호하는 다른 등록 대행자를 통해 등록한 도메인이 있는 경우, 이 자습서의 단계를 시작할 수 있습니다. 또는 도메인의 DNS 레코드 관리를 이전할 수 있습니다. 이렇게 하면 Lightsail 내에서 모든 컴퓨팅 및 DNS 리소스를 보다 효율적으로 관리할 수 있습니다.
- 사용하려는 도메인이 없는 경우 웹 사이트를 위한 도메인을 찾아 등록할 수 있습니다. Lightsail로 도메인을 등록하는 방법에 대한 자세한 내용은 Lightsail에서 웹 사이트의 도메인 등록 및 관리를 참조하세요.

1단계: Plesk 인스턴스 생성

Lightsail에서 Plesk 인스턴스를 생성하려면 다음 단계를 완료하세요.

1. https://lightsail.aws.amazon.com/ Lightsail 콘솔에 로그인합니다.
2. 홈 페이지의 인스턴스에서 인스턴스 생성을 선택합니다.
3. 인스턴스의 위치(AWS 리전 및 가용 영역)를 선택합니다.

   변경 AWS 리전 및 가용 영역을 선택하여 다른 위치에 인스턴스를 생성합니다.
4. 선택적으로, 가용 영역을 변경할 수 있습니다.

   가용 영역 변경을 선택합니다.
5. 앱 + OS에서 Ubuntu의 Plesk 호스팅 스택을 선택합니다.
6. 사용 가능한 인스턴스 플랜을 선택합니다.

   > **Tip**
   >
   > 최적의 웹 사이트 성능을 위해, 설치하려는 플러그인 및 확장의 리소스 수요를 수용할 수 있는 인스턴스 계획을 선택합니다. 나중에 더 많은 컴퓨팅 성능이 필요한 경우 스냅샷에서 인스턴스의 크기를 늘릴 수 있습니다. 플러그인 및 확장의 요구 사항에 대한 자세한 내용은 해당 공급업체의 설명서를 참조하세요.
7. 인스턴스 이름을 입력합니다.

   리소스 이름:

   - Lightsail 계정 AWS 리전 의 각 내에서 고유해야 합니다.
   - 2~255자의 문자로 구성되어야 합니다.
   - 영숫자 문자 또는 숫자로 시작하고 끝나야 합니다.

- 영숫자 문자, 숫자, 마침표, 대시, 밑줄이 포함될 수 있습니다.

8. (선택 사항) 인스턴스에 태그를 추가하려면 새 태그 추가를 선택합니다. 필요에 따라 이 단계를 반복하여 모든 태그를 추가합니다. 청구 및 리소스 조직과 같은 태그 사용에 대한 자세한 내용은 태그를 참조하세요.

   - 키에는 태그 키를 입력합니다.
   - (선택 사항) 값에 태그 값을 입력합니다.

9. 인스턴스 생성을 선택합니다.

인스턴스를 생성한 후 프로비저닝하고 사용할 수 있게 되기까지 몇 분 정도 걸립니다.

> **Tip**
>
> Plesk의 초기 설정 프로세스를 간소화하려면 지금은 인스턴스에 고정 IP를 할당하지 말고 이후 단계에서 이 작업을 수행하도록 합니다. 지금 Plesk 인스턴스에 고정 IP를 연결하는 경우, 이전의 퍼블릭 IP 주소를 사용하는 일회성 로그인 URL이 표시될 수 있습니다. 이는 인스턴스가 고정 IP 주소가 아닌 처음에 할당된 퍼블릭 IP 주소를 사용할 수 있기 때문에 발생할 수 있습니다.

2단계: Plesk 사용자 인터페이스에 처음 로그인

> **Tip**
>
> 이 자습서를 완료할 때까지 Lightsail 콘솔 및 Plesk 사용자 인터페이스 브라우저 탭을 열어 둡니다. 각각에서 여러 단계를 거치게 됩니다.

다음 절차에 따라 관리자로서 Plesk 사용자 인터페이스에 액세스할 수 있는 일회성 로그인 URL을 가져옵니다. 이 인터페이스에서 웹 사이트를 생성하고 관리할 수 있습니다.

일회성 로그인 URL을 얻으려면

1. 왼쪽 탐색 창에서 인스턴스를 선택합니다.
2. 생성한 Plesk 인스턴스의 이름을 선택합니다.
3. 인스턴스 관리 페이지의 연결(Connect) 탭에서 SSH로 연결(Connect using SSH)을 선택합니다.

> **Note**
>
> 인스턴스를 시작한 후 너무 빨리 연결을 시도하면 연결 또는 SSH 키 오류가 발생할 수 있습니다. 연결하는 동안 이러한 문제가 발생하는 경우, 몇 분 정도 기다렸다가 다시 시도하세요.

4. 연결한 후 다음 명령을 입력하여 일회용 로그인 URL을 가져옵니다.

```
sudo plesk login | grep plesk.page
```

일회용 로그인 URL이 포함되어 있는 다음 예와 유사한 응답이 표시되어야 합니다.

```
https://heuristic-bassi.192-0-2-0.plesk.page/login?secret=ce-
EXAMPLE298fc1c149afbf4c8996fb92427
```

> **Note**
>
> 퍼블릭 IP 주소가 변경된 경우(예: 시작 이후 고정 IP를 할당한 경우), 로그인 페이지에 액세스하려면 반환된 URL을 새 주소와 일치하도록 수정해야 합니다.

5. 브라우저 기반 연결 창에서 일회성 로그인 URL을 선택하여 복사합니다.
6. 웹 브라우저에 URL을 붙여 넣어 Plesk 로그인 페이지에 액세스합니다.
7. 페이지의 지침에 따라 Plesk에 대한 로그인 자격 증명을 생성합니다. 처음으로 로그인할 때 Plesk에 도메인을 추가하는 옵션이 나타납니다.

3단계: 고정 IP 주소를 Plesk 인스턴스에 연결

인스턴스에 연결된 기본 동적 퍼블릭 IP 주소는 인스턴스를 중지하고 시작할 때마다 변경됩니다. 퍼블릭 IP 주소가 변경되지 않도록 하려면 고정 IP 주소를 생성하여 인스턴스에 연결합니다. 이후 단계에서 이 고정 IP 주소를 도메인 이름에 매핑합니다. 이 매핑을 구성하면 인스턴스를 중지하고 시작할 때마다 도메인의 DNS 레코드를 업데이트할 필요가 없습니다.

> **Note**
>
> 한 Lightsail 인스턴스에 한 개의 고정 IP만 연결할 수 있습니다. 고정 IP 주소는 인스턴스에 연결되어 있을 때에만 무료입니다.

Plesk에서 고정 IP를 연결하고 퍼블릭 IP 주소를 업데이트하려면

1. https://lightsail.aws.amazon.com/ Lightsail 콘솔에 로그인합니다.
2. Lightsail 콘솔의 인스턴스 홈 페이지에서 Plesk 인스턴스의 이름을 선택합니다.
3. 네트워킹 탭에서 고정 IP 연결을 선택합니다.
4. 고정 IP를 생성하여 연결합니다.
   - 리전에 고정 IP가 없는 경우 고정 IP를 생성하고 연결하려면:
     a. 고정 IP를 식별하는 이름을 입력합니다.
     b. 생성 및 연결을 선택합니다.
     c. 계속을 선택합니다.
   - 리전에 고정 IP가 이미 있는 경우 새 고정 IP를 생성하고 연결하려면:
     a. 고정 IP 선택 메뉴를 선택하여 사용 가능한 옵션을 표시합니다.
     b. 드롭다운 메뉴에서 새 고정 IP 생성을 선택합니다.
     c. 고정 IP를 식별하는 이름을 입력합니다.
     d. 생성 및 연결을 선택합니다.
     e. 계속을 선택합니다.
   - 리전에서 기존 고정 IP를 사용하려면:
     a. 고정 IP 선택 메뉴를 선택하여 사용 가능한 옵션을 표시합니다.
     b. 드롭다운 메뉴에서 이미 사용 가능한 고정 IP를 선택합니다.
     c. 연결을 선택합니다.

퍼블릭 IP 주소가 변경되면 `https://StaticIPAddress:8443`과 같은 URL을 사용하여 Plesk 사용자 인터페이스에 액세스합니다. `StaticIPAddress`는 인스턴스의 고정 IP 주소로 변경합니다. 예를 들어 `https://192.0.2.0:8443`입니다. 로그인 페이지에서 이전에 생성한 사용자 이름과 암호를 입력하여 Plesk 사용자 인터페이스에 로그인합니다.

> **Note**
>
> URL의 IP 주소로 연결할 때 인터넷 연결이 비공개가 아니거나 안전하지 않거나 보안 위험이 있다는 브라우저 경고가 표시될 수 있습니다. 이는 Plesk 인스턴스에 적용된 SSL/TLS 인증서가 사용 중인 새 퍼블릭 IP와 일치하지 않기 때문에 발생합니다. 브라우저 창에서 Advanced(고급), Details(세부 정보) 또는 More information(추가 정보)을 선택하여 사용할 수 있는 옵션을

표시합니다. 그런 다음 비공개가 아니거나 안전하지 않더라도 웹 사이트로 이동하도록 선택합니다.

4단계: Plesk에서 퍼블릭 IP 주소 업데이트

이제 인스턴스에 고정 IP 주소가 할당되었으므로 Plesk에서 사용하는 퍼블릭 IP 주소를 업데이트합니다. 이렇게 하면 Plesk가 인스턴스와 더 이상 연결되지 않는 동적 주소가 아닌 업데이트된 퍼블릭 IP 주소를 사용할 수 있습니다. 이 단계를 수행하지 않으면 Plesk 사용자 인터페이스에서 불일치에 대한 경고가 표시될 수 있습니다.

Plesk에서 퍼블릭 IP 주소를 업데이트하려면

1. 할당된 새 퍼블릭 IP 주소를 사용하여 Plesk 사용자 인터페이스에 액세스합니다. 예를 들어 `https://192.0.2.0:8443`입니다.
2. 이전에 2단계: Plesk 사용자 인터페이스에 처음 로그인에서 생성한 로그인 자격 증명을 사용하여 인증합니다.
3. Plesk 사용자 인터페이스의 왼쪽 하단에서 보기 변경을 선택한 다음 고급 사용자 보기로 전환을 선택합니다.
4. Plesk 사용자 인터페이스의 왼쪽 탐색 창에서 도구 및 설정을 선택합니다.
5. 도구 및 리소스 아래에서 IP 주소를 선택합니다.
6. 퍼블릭 IP 업데이트를 선택합니다.

이제 퍼블릭 IP 주소 열에 인스턴스에 연결된 고정 IP와 일치하는 값이 표시됩니다.

5단계: Plesk 인스턴스에 웹 사이트 추가

Plesk 사용자 인터페이스 내에서 하나 이상의 도메인 이름(웹 사이트)을 매핑할 수 있습니다. Plesk 인스턴스에 도메인을 추가하면 콘텐츠를 업로드하고, Presence Builder를 사용하고, 콘텐츠 관리 시스템(CMS)을 설치할 수 있습니다.

Plesk에서 새 도메인을 추가하려면

1. Plesk 사용자 인터페이스의 왼쪽 탐색 창에서 웹 사이트 및 도메인을 선택합니다.
2. 도메인 추가를 선택합니다.
3. 웹 사이트를 생성하는 방법에 대한 옵션을 선택합니다.

4. 도메인 이름을 입력하고 추가 필수 정보 필드를 입력합니다.
5. 도메인 추가를 선택합니다.

자세한 내용은 Plesk 관리자 설명서의 도메인 추가 및 제거를 참조하세요.

6단계: 도메인 이름을 Plesk 인스턴스에 매핑

도메인을 Plesk 인스턴스에 매핑하면 생성한 도메인 이름을 사용하여 Plesk 인스턴스에 액세스할 수 있습니다. 또한, Plesk 사용자 인터페이스 내에서 여러 도메인을 매핑하여 웹 사이트를 관리하는 데 사용할 수 있습니다. Plesk 사용자 인터페이스 내에서 여러 도메인을 매핑하는 방법에 대한 자세한 내용은 Plesk 문서 및 도움말 포털의 Plesk에서 도메인 추가를 참조하세요.

`example.com`과 같은 도메인 이름을 인스턴스에 매핑하려면 도메인의 도메인 이름 시스템(DNS)에 레코드를 추가합니다. DNS 레코드는 일반적으로 도메인을 등록한 등록 대행자가 관리 및 호스팅합니다. 그러나 Lightsail 콘솔을 사용하여 관리할 수 있도록 도메인의 DNS 레코드 관리를 Lightsail로 이전하는 것이 좋습니다. 자세한 내용은 Lightsail 도메인에 대한 전송 DNS 관리 단원을 참조하십시오.

도메인 이름을 Lightsail의 Plesk 인스턴스에 매핑하려면

1. https://console.aws.amazon.com/lightsail/에서 Amazon Lightsail 콘솔을 엽니다.
2. Plesk 사용자 인터페이스의 왼쪽 탐색 창에서 도메인 및 DNS를 선택합니다.
3. DNS 영역의 이름을 선택합니다.
4. DNS records(DNS 레코드) 탭을 선택합니다.
5. 레코드 추가를 선택합니다.
6. 레코드 이름에 `www`와 같은 값을 입력합니다.
7. 해결 대상에 Plesk 인스턴스에 연결된 고정 IP 주소를 입력합니다.
8. 저장을 선택합니다.

이제 구성한 도메인 이름을 사용하여 Plesk 웹 사이트에 액세스할 수 있습니다.

7단계: Plesk 설명서 읽기

Plesk 설명서를 읽고 웹 사이트를 관리하는 방법, Plesk 사용자 인터페이스를 사용자 지정하는 방법 등을 알아보세요.

자세한 내용은 Plesk 문서 및 도움말 포털의 Plesk에서 웹 사이트 관리 시작하기를 참조하세요.

8단계: Plesk 라이선스 구매

Plesk 인스턴스에는 30일 평가판 라이선스가 포함되어 있습니다. 30일이 지나면 Plesk에서 라이선스를 구매해야 계속 사용할 수 있습니다. 자세한 내용은 Plesk 웹사이트의 요금을 참조하세요.

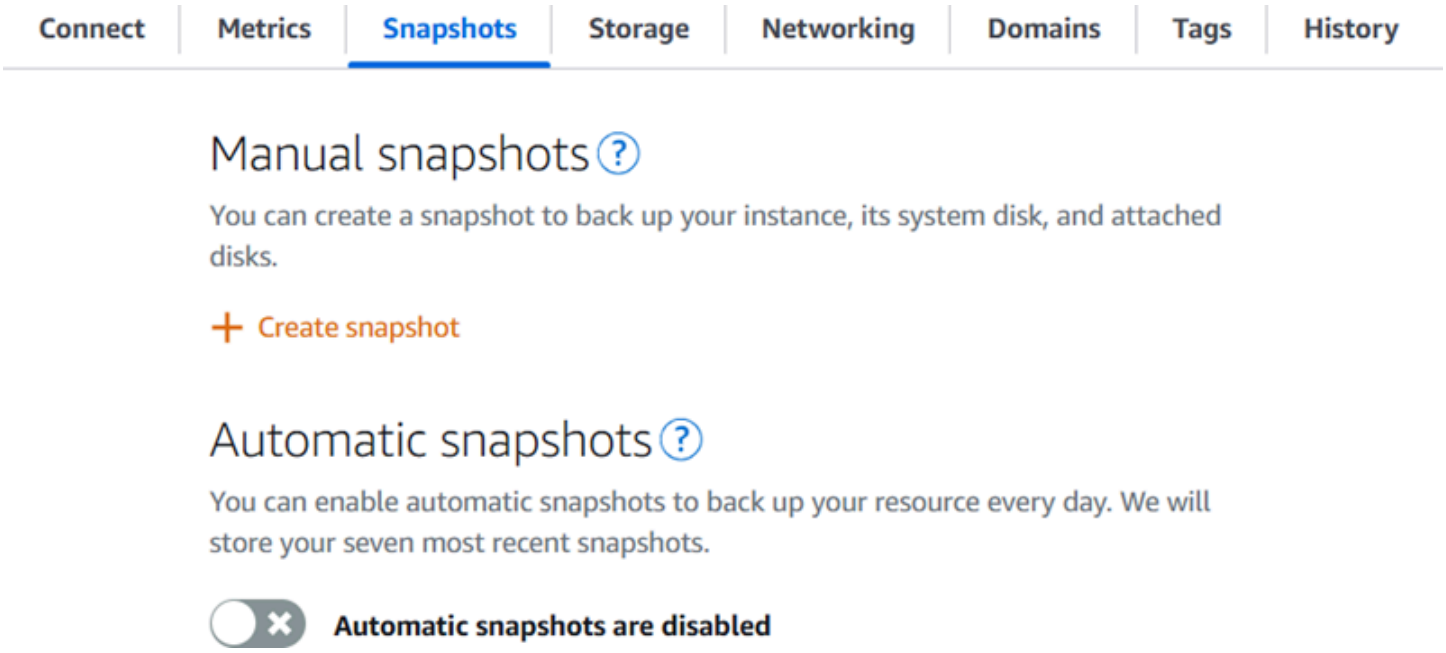
Plesk에서 라이선스를 구매한 후에는 라이선스를 설치해야 합니다. Plesk 라이선스를 설치하려면 Plesk 지원 웹 사이트에 Plesk 라이선스를 설치하는 방법을 참조하세요.

9단계: Plesk 인스턴스의 스냅샷 생성

원하는 방식으로 웹 사이트를 구성한 후 인스턴스의 주기적 스냅샷을 생성하여 백업합니다. 스냅샷은 시스템 디스크의 복사본이며, 인스턴스의 원본 구성입니다. 스냅샷은 스냅샷을 생성한 시점부터 인스턴스를 복원하는 데 필요한 모든 데이터를 포함합니다.

스냅샷을 수동으로 생성하거나 자동 스냅샷을 활성화하여 Lightsail에서 매일 스냅샷을 생성하도록 할 수 있습니다. 인스턴스에 문제가 있는 경우 스냅샷을 사용하여 새 교체 인스턴스를 생성할 수 있습니다.

스냅샷 탭의 인스턴스 관리 페이지에서 스냅샷 작업을 할 수 있습니다. 자세한 내용은 Amazon Lightsail의 스냅샷을 참조하세요.



Lightsail에서 PrestaShop 웹 사이트 설정

> 알고 계셨나요?
>
> Lightsail는 일별 스냅샷 7개를 저장하고 인스턴스에 대해 자동 스냅샷을 활성화하면 가장 오래된 스냅샷을 최신 스냅샷으로 자동 대체합니다. 자세한 내용은 Lightsail 인스턴스 및 디스크의 자동 스냅샷 구성을 참조하세요.

다음은 Amazon Lightsail에 PrestaShop 인스턴스를 설치하고 실행한 후 시작하기 위해 완료해야 하는 몇 가지 단계입니다.

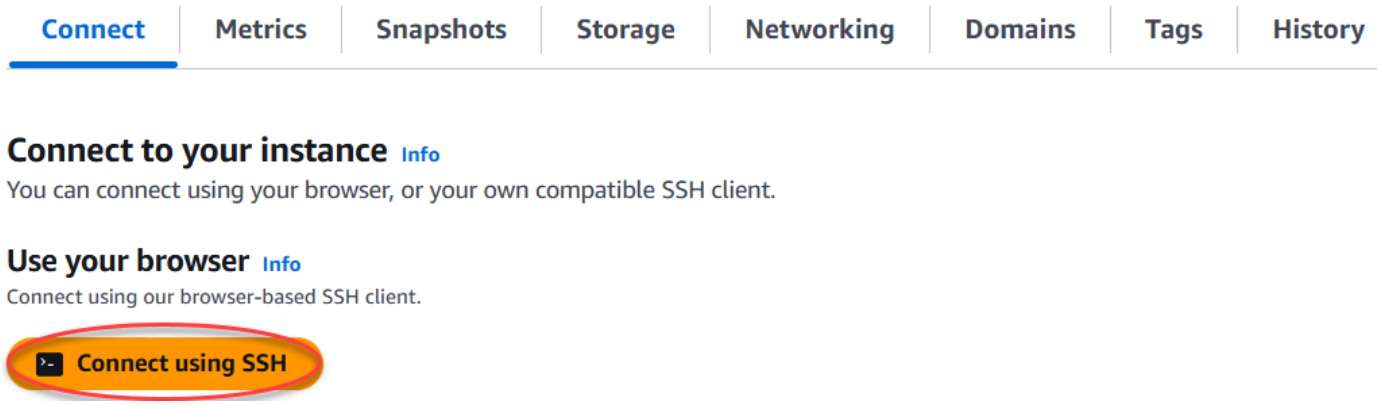
목차

- 1단계: PrestaShop 웹 사이트의 기본 애플리케이션 암호 가져오기
- 2단계: PrestaShop 인스턴스에 고정 IP 주소 연결
- 3단계: PrestaShop 웹 사이트의 관리 대시보드에 로그인
- 4단계: 등록된 도메인 이름의 트래픽을 PrestaShop 웹 사이트로 라우팅
- 5단계: PrestaShop 웹 사이트에 대해 HTTPS 구성
- 6단계: 이메일 알림을 위한 SMTP 구성
- 7단계: Bitnami 및 PrestaShop 문서 읽기
- 8단계: PrestaShop 인스턴스의 스냅샷 생성

1단계: PrestaShop 웹 사이트의 기본 애플리케이션 암호 가져오기

PrestaShop 웹 사이트의 기본 애플리케이션 암호를 가져오려면 다음 단계를 완료하세요.

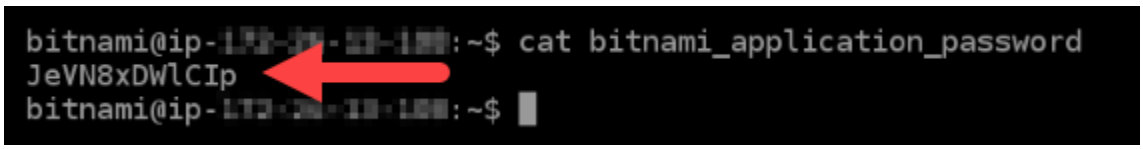
1. 인스턴스 관리 페이지의 연결(Connect) 탭에서 SSH를 사용하여 연결(Connect using SSH)을 선택합니다.



2. 연결한 후 다음 명령을 입력하여 기본 애플리케이션 암호를 가져옵니다.

```
cat $HOME/bitnami_application_password
```

다음 예와 유사한 응답이 표시되며, 여기에 기본 애플리케이션 암호가 포함되어 있습니다. 암호를 안전한 위치에 저장합니다. 이 자습서의 다음 섹션에서 PrestaShop 웹 사이트의 관리 대시보드에 로그인하는 데 이 암호를 사용하게 됩니다.

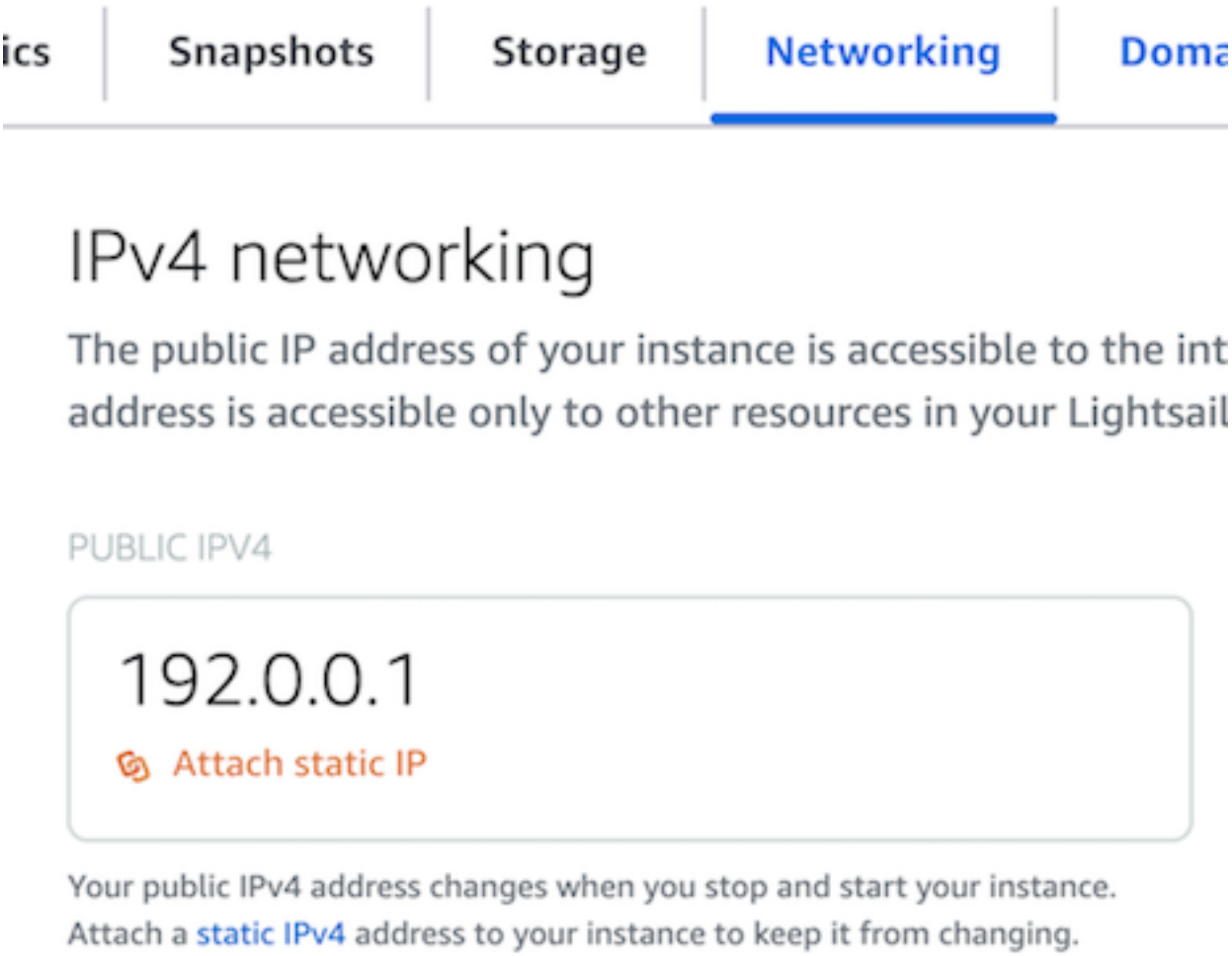


자세한 내용은 Amazon Lightsail의 Bitnami 인스턴스에 대한 애플리케이션 사용자 이름과 암호 가져오기를 참조하세요.

2단계: PrestaShop 인스턴스에 고정 IP 주소 연결

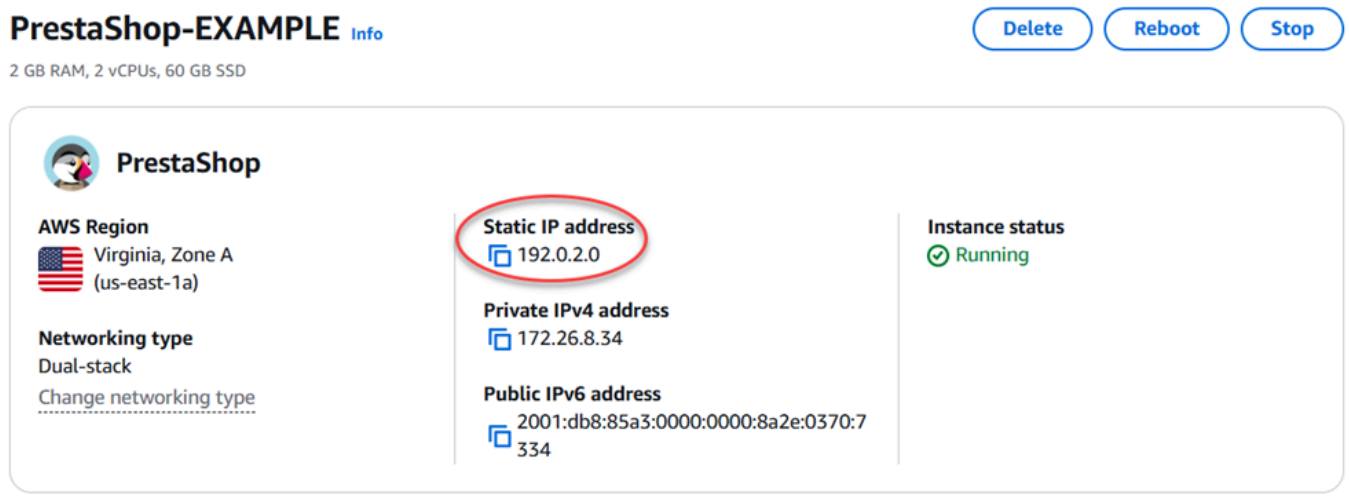
인스턴스에 연결된 기본 동적 퍼블릭 IP 주소는 인스턴스를 중지하고 시작할 때마다 변경됩니다. 퍼블릭 IP 주소가 변경되지 않도록 고정 IP 주소를 생성하고 인스턴스에 연결할 수 있습니다. 나중에 인스턴스에 도메인 이름을 사용하면 인스턴스를 중지하고 시작할 때마다 도메인의 DNS 레코드를 업데이트할 필요가 없습니다. 각 인스턴스에 1개의 고정 IP 주소만 연결할 수 있습니다.

인스턴스 관리 페이지의 네트워킹 탭에서 고정 IP 생성 또는 고정 IP 연결을 선택합니다. 인스턴스에 연결할 수 있는 고정 IP를 생성해 둔 경우에는 페이지의 지침을 따릅니다. 자세한 내용은 고정 IP를 생성하여 인스턴스에 연결을 참조하세요.

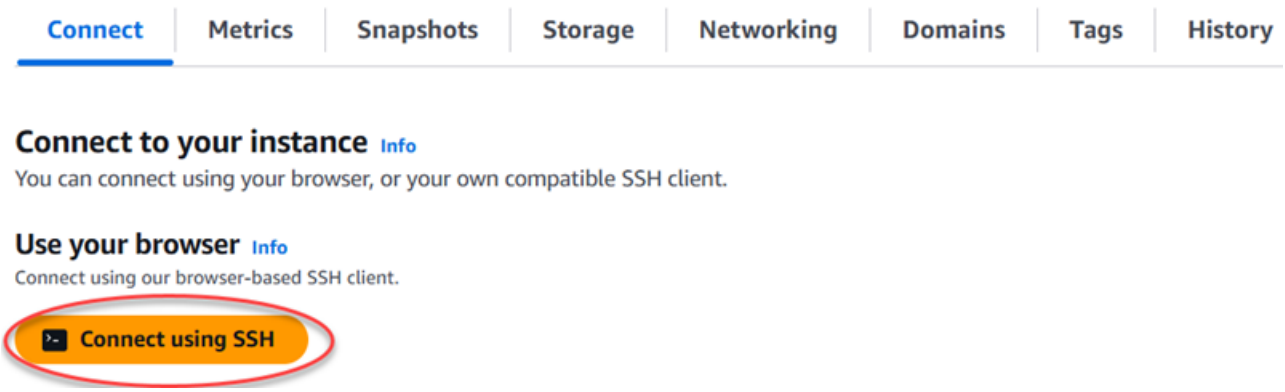


새로운 고정 IP 주소가 인스턴스에 연결되면 PrestaShop 소프트웨어가 새로운 고정 IP 주소를 인식하도록 다음 단계를 완료해야 합니다.

1. 인스턴스의 고정 IP 주소를 기록해 둡니다. 고정 IP 주소는 인스턴스 관리 페이지의 머리말 섹션에 나와 있습니다.



2. 인스턴스 관리 페이지의 연결(Connect) 탭에서 SSH를 사용하여 연결(Connect using SSH)을 선택합니다.



3. 연결한 후 다음 명령을 입력합니다. *<StaticIP>*를 인스턴스의 새로운 고정 IP 주소로 대체해야 합니다.

```
sudo /opt/bitnami/configure_app_domain --domain <StaticIP>
```

예:

```
sudo /opt/bitnami/configure_app_domain --domain 203.0.113.0
```

다음 예와 비슷한 응답이 나타나는 것을 볼 수 있습니다. PrestaShop 소프트웨어에서 이제 새로운 고정 IP 주소를 인식합니다.

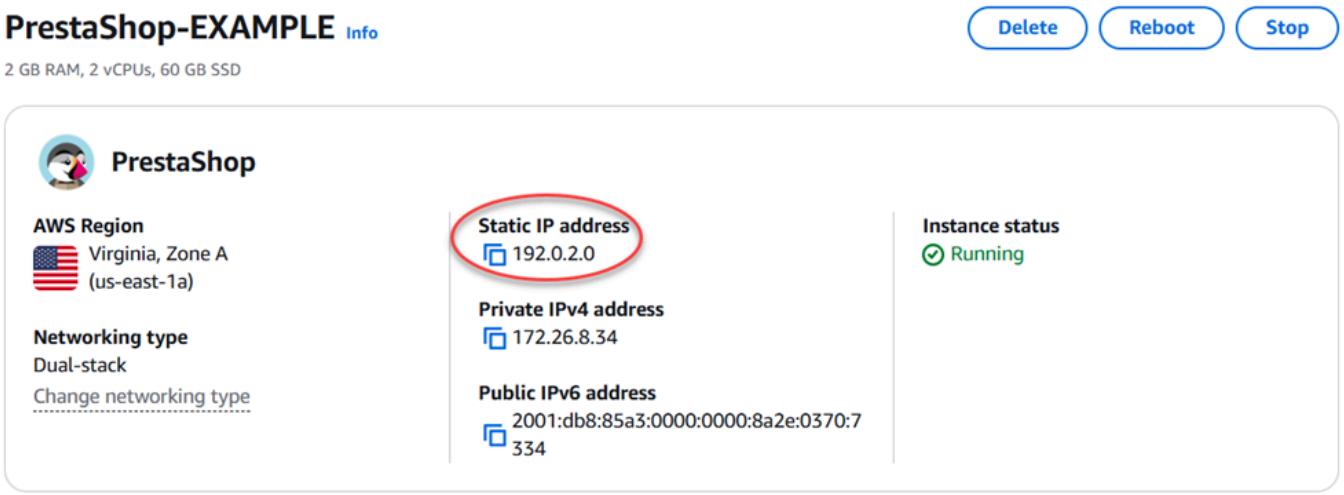
```
bitnami@ip-[redacted]:~$ sudo /opt/bitnami/configure_app_domain --domain 203.0.113.0
Configuring domain to 203.0.113.0
2021-03-12T15:49:22.000Z - info: Saving configuration info to disk
prestashop 15:49:22.41 INFO  ==> Trying to connect to the database server
prestashop 15:49:22.44 INFO  ==> Updating hostname in database
prestashop 15:49:22.46 INFO  ==> Purging cache
Disabling automatic domain update for IP address changes
```

> **Note**
>
> PrestaShop은 현재 IPv6 주소를 지원하지 않습니다. 인스턴스에 IPv6를 활성화할 수 있지만, PrestaShop 소프트웨어는 IPv6 네트워크를 통한 요청에 응답하지 않습니다.

3단계: PrestaShop 웹 사이트의 관리 대시보드에 로그인

PrestaShop 웹 사이트에 액세스하고 관리 대시보드에 로그인하려면 다음 단계를 완료하세요. 기본 사용자 이름(`user@example.com`)과 이 가이드의 앞부분에서 확인한 기본 애플리케이션 암호를 사용하여 로그인할 수 있습니다.

1. Lightsail 콘솔의 인스턴스 관리 페이지 머리말 영역에 나와 있는 퍼블릭 또는 고정 IP 주소를 기록해 둡니다.



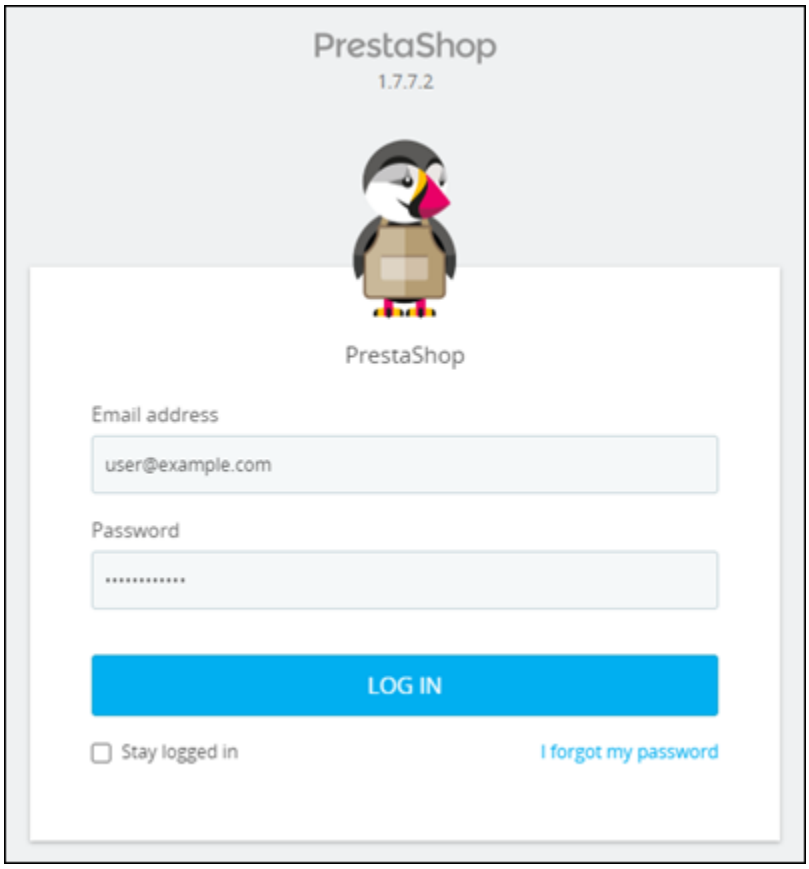
2. 다음 주소를 검색하여 PrestaShop 웹 사이트의 관리 대시보드 로그인 페이지에 액세스합니다. 여기서 *<InstanceIpAddress>*를 인스턴스의 퍼블릭 IP 주소 또는 고정 IP 주소로 대체해야 합니다.

```
http://<InstanceIpAddress>/administration
```

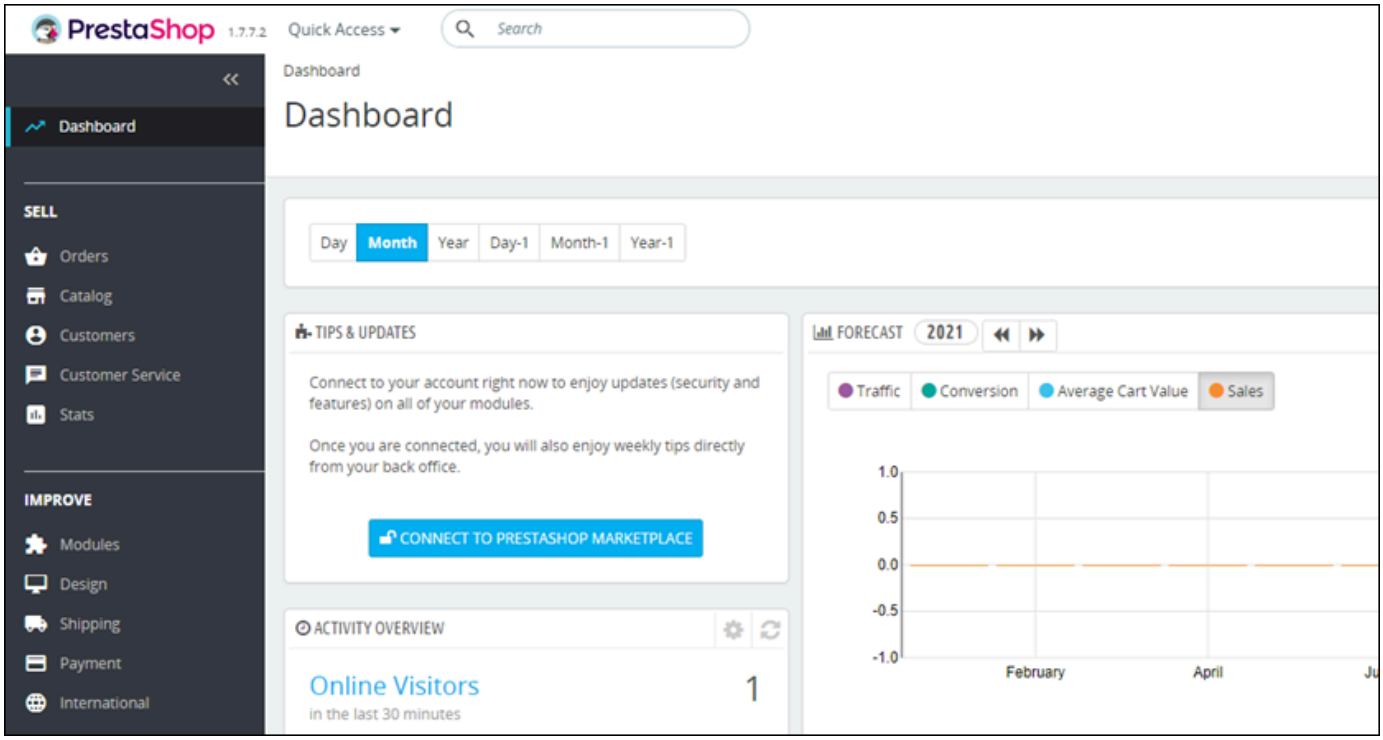
예:

```
http://203.0.113.0/administration
```

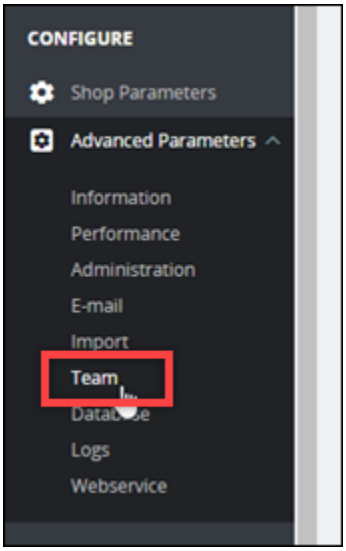
3. 이 가이드의 앞부분에서 확인한 기본 애플리케이션 암호와 기본 사용자 이름(`user@example.com`)을 입력하고 로그인(Log in)을 선택합니다.



PrestaShop 관리 대시보드가 표시됩니다.



PrestaShop 웹 사이트의 관리 대시보드에 로그인하는 데 사용하는 기본 사용자 이름 또는 암호를 변경하려면 탐색 창에서 고급 파라미터(Advanced Parameters)를 선택한 다음, 팀(Team)을 선택합니다. 자세한 내용은 PrestaShop 문서의 PrestaShop 사용 설명서를 참조하세요.



관리 대시보드에 대한 자세한 내용은 PrestaShop 설명서의 사용 설명서의 PrestaShop을 참조하세요.

4단계: 등록된 도메인 이름의 트래픽을 PrestaShop 웹 사이트로 라우팅

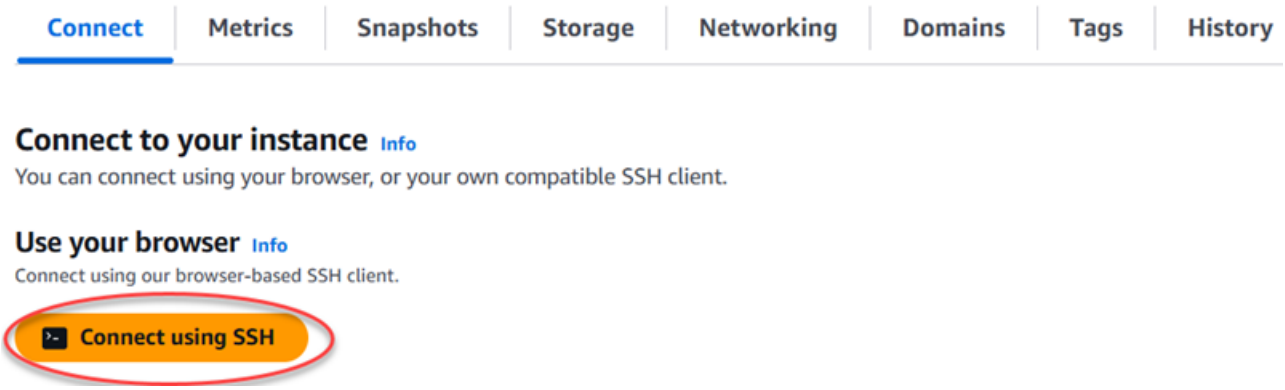
등록된 도메인 이름(예: `example.com`)의 트래픽을 PrestaShop 웹 사이트로 라우팅하려면 도메인의 도메인 이름 시스템(DNS)에 레코드를 추가하면 됩니다. DNS 레코드는 일반적으로 도메인을 등록한 등록 대행자가 관리 및 호스팅합니다. 그러나 Lightsail 콘솔을 사용하여 관리할 수 있도록 도메인의 DNS 레코드 관리를 Lightsail로 이전하는 것이 좋습니다.

Lightsail 콘솔 홈 페이지의 도메인 및 DNS 탭에서 DNS 영역 생성을 선택하고 페이지의 지침에 따릅니다.

자세한 내용은 DNS 영역을 생성하여 Lightsail에서 도메인의 DNS 레코드 관리를 참조하세요.

도메인 이름이 트래픽을 인스턴스로 라우팅한 후 PrestaShop 소프트웨어가 도메인 이름을 인식하도록 하려면 다음 단계를 완료해야 합니다.

1. 인스턴스 관리 페이지의 연결(Connect) 탭에서 SSH를 사용하여 연결(Connect using SSH)을 선택합니다.



2. 연결한 후 다음 명령을 입력합니다. *<DomainName>*을 트래픽을 인스턴스로 라우팅할 도메인 이름으로 대체해야 합니다.

```
sudo /opt/bitnami/configure_app_domain --domain <DomainName>
```

예:

```
sudo /opt/bitnami/configure_app_domain --domain www.example.com
```

다음 예와 비슷한 응답이 나타나는 것을 볼 수 있습니다. 이제 PrestaShop 소프트웨어가 도메인 이름을 인식합니다.

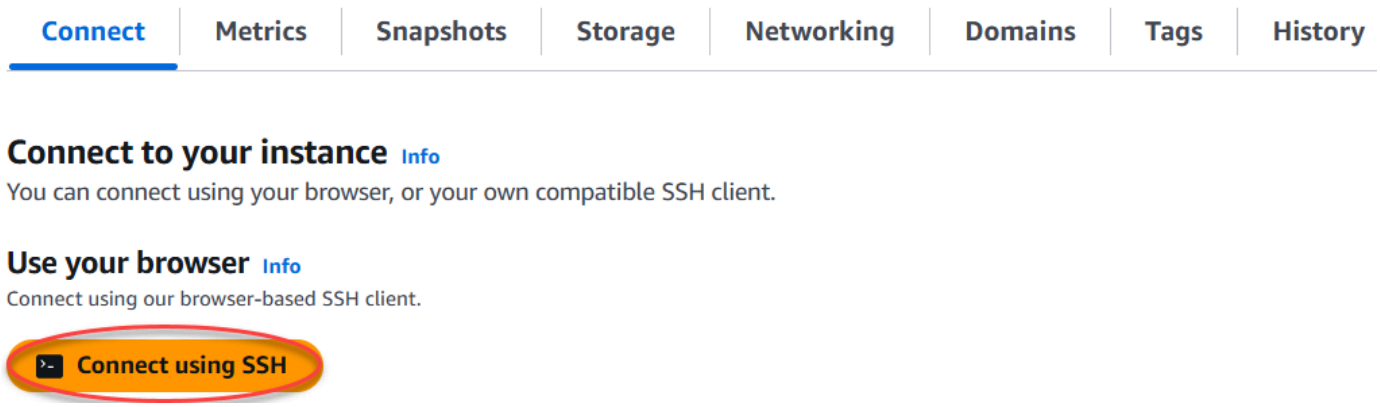
```
bitnami@ip-[redacted]:~$ sudo /opt/bitnami/configure_app_domain --domain www.example.com
Configuring domain to www.example.com
2021-03-12T15:49:22.000Z - info: Saving configuration info to disk
prestashop 15:49:22.41 INFO  ==> Trying to connect to the database server
prestashop 15:49:22.44 INFO  ==> Updating hostname in database
prestashop 15:49:22.46 INFO  ==> Purging cache
Disabling automatic domain update for IP address changes
```

5단계: PrestaShop 웹 사이트에 대해 HTTPS 구성

PrestaShop 웹 사이트에서 HTTPS를 구성하려면 다음 단계를 완료하세요. 다음 단계에서는 SSL/TLS 인증서 요청, 리디렉션(예: HTTP에서 HTTPS로) 설정 및 인증서 갱신에 사용하는 명령줄 도구인 Bitnami HTTPS 구성 도구(bncert)를 사용하는 방법을 안내합니다.

> **Important**
>
> bncert 도구는 PrestaShop 인스턴스의 퍼블릭 IP 주소로 트래픽을 라우팅하고 있는 도메인에 대해서만 인증서를 발급합니다. 이러한 단계를 시작하기 전에 PrestaShop 웹 사이트에서 사용할 모든 도메인의 DNS에 DNS 레코드를 추가해야 합니다.

1. 인스턴스 관리 페이지의 연결(Connect) 탭에서 SSH를 사용하여 연결(Connect using SSH)을 선택합니다.



2. 연결한 후 다음 명령을 입력하여 bncert-tool을 시작합니다.

```
sudo /opt/bitnami/bncert-tool
```

다음 예와 비슷한 응답이 나타나는 것을 볼 수 있습니다.

```
bitnami@ip-[redacted]:~$ sudo /opt/bitnami/bncert-tool
--------------------------------------------------------------------------------
Welcome to the Bitnami HTTPS Configuration tool.

--------------------------------------------------------------------------------
Domains

Please provide a valid space-separated list of domains for which you wish to
configure your web server.

Domain list []:
```

3. 다음 예와 같이 기본 도메인 이름과 대체 도메인 이름을 공백으로 구분하여 입력합니다.

```
--------------------------------------------------------------------------------
Welcome to the Bitnami HTTPS Configuration tool.

--------------------------------------------------------------------------------
Domains

Please provide a valid space-separated list of domains for which you wish to
configure your web server.

Domain list []: example.com www.example.com
```

4. bncert 도구가 웹 사이트의 리디렉션을 어떻게 구성할지 묻는 메시지를 표시합니다. 사용할 수 있는 옵션은 다음과 같습니다.

   - HTTP에서 HTTPS로 리디렉션 활성화(Enable HTTP to HTTPS redirection) - HTTP 버전 웹 사이트(예: `http:/example.com`)를 방문하는 사용자를 HTTPS 버전(예: `https://example.com`)으로 자동 리디렉션할지 지정합니다. 모든 방문자가 암호화된 연결을 사용하도록 강제하기 때문에 이 옵션을 활성화하는 것이 좋습니다. Y를 입력하고 Enter 키를 눌러 활성화합니다.
   - 비 www에서 www로 리디렉션 활성화(Enable non-www to www redirection) - 도메인의 정점(`https://example.com`)을 방문하는 사용자를 도메인의 www 하위 도메인(`https://www.example.com`)으로 자동 리디렉션할지 지정합니다. 이 옵션은 활성화하는 것이 좋습니다. 그러나 Google의 웹 마스터 도구와 같은 검색 엔진 도구에서 도메인의 정점을 기본 웹 사이트 주소로 지정했거나 정점이 IP를 직접 가리키고 www 하위 도메인이 CNAME 레코드를 통해 정점을 참조하는 경우, 이 옵션을 비활성화하고 대체 옵션(www에서 비 www로 리디렉션 활성화)을 활성화할 수 있습니다. Y를 입력하고 Enter 키를 눌러 활성화합니다.
   - www에서 비 www로 리디렉션 활성화(Enable www to non-www redirection) - 도메인의 www 하위 도메인(`https://www.example.com`)을 방문하는 사용자를 도메인의 정점(`https://example.com`)으로 자동 리디렉션할지 지정합니다. 비 www에서 www로 리디렉션을 활성화한 경우 이 옵션을 비활성화하는 것이 좋습니다. N를 입력하고 Enter 키를 눌러 비활성화합니다.

선택한 내용은 다음 예와 같아야 합니다.

```
Enable/disable redirections

Please select the redirections you wish to enable or disable on your Bitnami
installation.



Enable HTTP to HTTPS redirection [Y/n]: Y




Enable non-www to www redirection [Y/n]: Y




Enable www to non-www redirection [y/N]: N
```

5. 적용될 변경 사항의 목록이 나열됩니다. Y를 입력하고 Enter 키를 눌러 확인하고 계속합니다.

```
Changes to perform

The following changes will be performed to your Bitnami installation:

1. Stop web server
2. Configure web server to use a free Let's Encrypt certificate for the domains:
example.com www.example.com
3. Configure a cron job to automatically renew the certificate each month
4. Configure web server name to: example.com
5. Enable HTTP to HTTPS redirection (example: redirect http://example.com to
https://example.com)
6. Enable non-www to www redirection (example: redirect example.com to
 www.example.com)
7. Start web server once all changes have been performed



Do you agree to these changes? [Y/n]: Y
```

6. Let's Encrypt 인증서에 연결할 이메일 주소를 입력하고 Enter 키를 누릅니다.

```
Create a free HTTPS certificate with Let's Encrypt

Please provide a valid e-mail address for which to associate your Let's Encrypt
certificate.

Domain list: example.com www.example.com

Server name: example.com

E-mail address []:
```

7. Let's Encrypt 구독자 계약을 검토합니다. Y를 입력하고 Enter 키를 눌러 계약을 수락하고 계속합니다.

```
The Let's Encrypt Subscriber Agreement can be found at:

https://letsencrypt.org/documents/LE-SA-v1.2-November-15-2017.pdf

Do you agree to the Let's Encrypt Subscriber Agreement? [Y/n]:
```

인증서 요청 및 지정한 리디렉션 구성을 비롯하여 인스턴스에서 HTTPS를 활성화하는 작업이 수행됩니다.

```
Performing changes to your installation

The Bitnami HTTPS Configuration Tool will perform any necessary actions to your
Bitnami installation. This may take some time, please be patient.
```

인증서가 성공적으로 발급되고 검증됩니다. 다음 예와 유사한 메시지가 표시되면 인스턴스에서 리디렉션이 성공적으로 구성된 것입니다.

```
Success

The Bitnami HTTPS Configuration Tool succeeded in modifying your installation.

The configuration report is shown below.

Backup files:
* /opt/bitnami/apache2/conf/httpd.conf.back.202005290035
* /opt/bitnami/apache2/conf/bitnami/bitnami-apps-prefix.conf.back.202005290035
* /opt/bitnami/apache2/conf/bitnami/bitnami.conf.back.202005290035

Find more details in the log file:

/tmp/bncert-202005290035.log

If you find any issues, please check Bitnami Support forums at:

https://community.bitnami.com

Press [Enter] to continue:
```

bncert 도구는 인증서가 만료되기 80일 전마다 인증서를 자동으로 갱신합니다. 다음 단계를 계속 진행하여 PrestaShop 웹 사이트에서 HTTPS를 마저 활성화합니다.

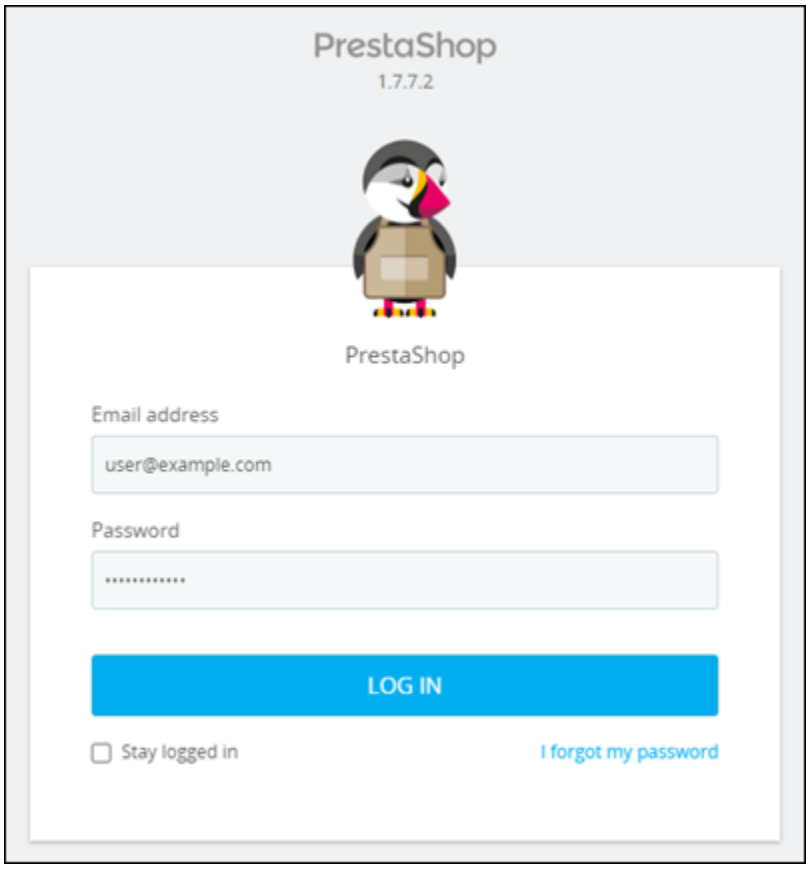
8. 다음 주소를 검색하여 PrestaShop 웹 사이트의 관리 대시보드 로그인 페이지에 액세스합니다. *<DomainName>*을 트래픽을 인스턴스로 라우팅하는 등록된 도메인 이름으로 대체해야 합니다.

```
http://<DomainName>/administration
```

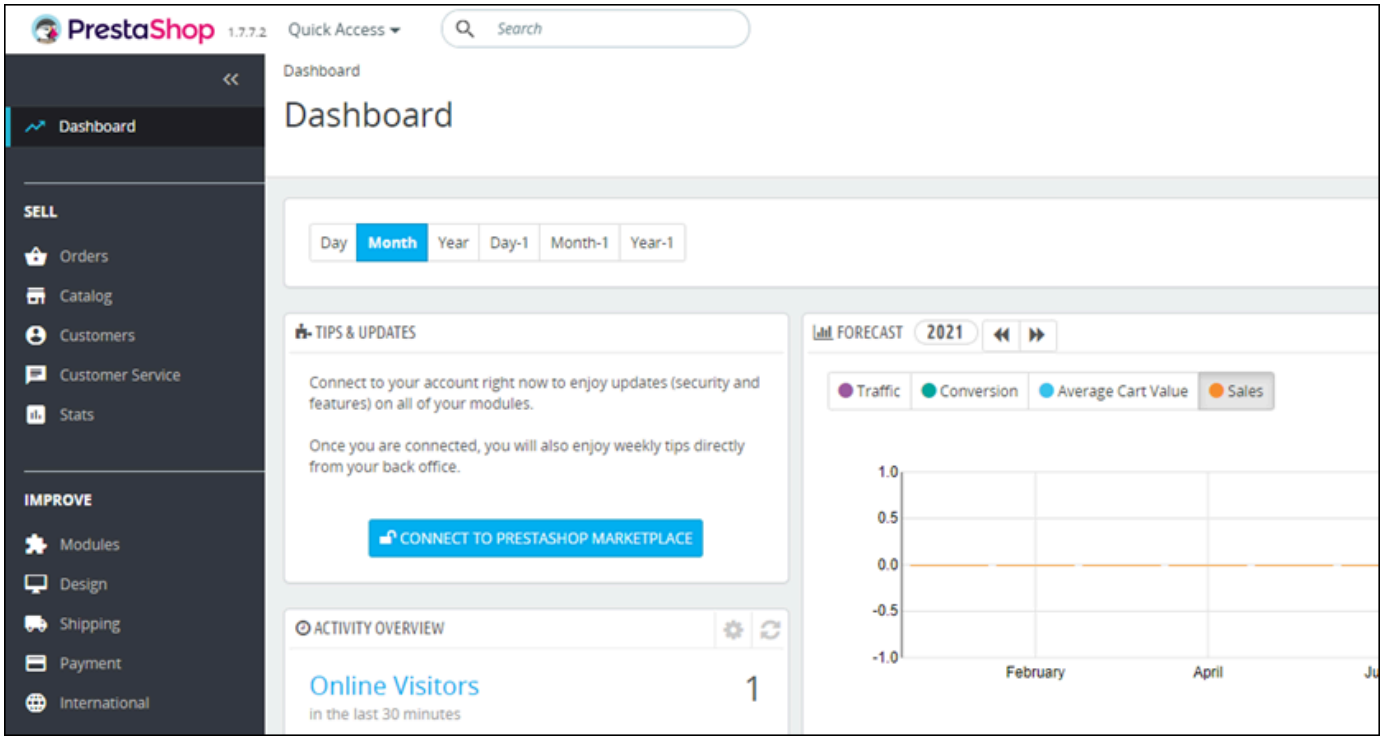
예:

```
http://www.example.com/administration
```

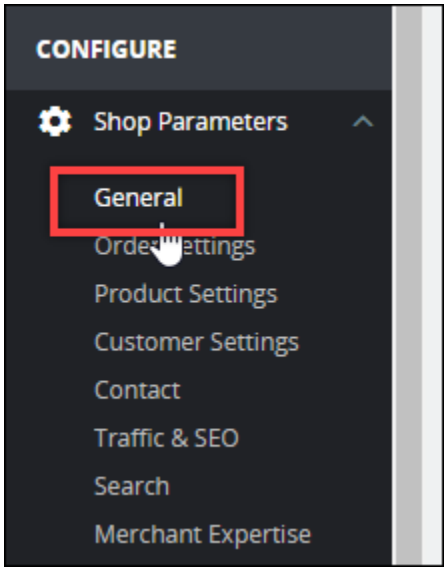
9. 이 가이드의 앞부분에서 확인한 기본 애플리케이션 암호와 기본 사용자 이름(`user@example.com`)을 입력하고 로그인(Log in)을 선택합니다.



PrestaShop 관리 대시보드가 표시됩니다.



10. 탐색 창에서 상점 파라미터(Shop Parameters)를 선택한 다음 일반(General)을 선택합니다.



11. SSL 활성화(Enable SSL) 옆에 있는 예(Yes)를 선택합니다.

12. 페이지의 하단으로 스크롤하고 저장(Save)을 선택합니다.

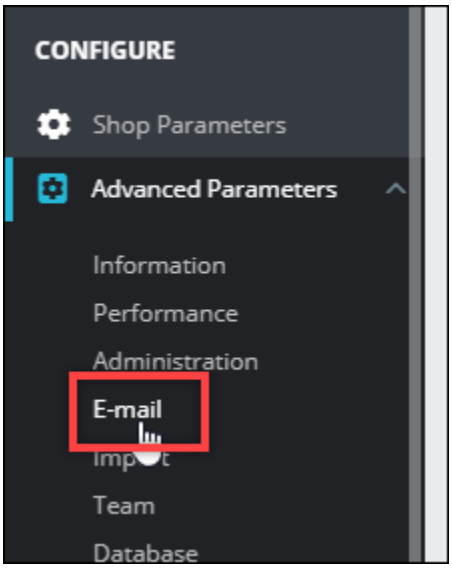
13. 일반(General) 페이지가 다시 로드되면 모든 페이지에서 SSL 활성화(Enable SSL on all pages) 옆에 있는 예(Yes)를 선택합니다.

14. 페이지의 하단으로 스크롤하고 저장(Save)을 선택합니다.

    이제 PrestaShop 웹 사이트에 대해 HTTPS를 구성했습니다. 고객이 PrestaShop 웹 사이트의 HTTP 버전(예: `http://www.example.com`)에 방문하면 HTTPS 버전(예: `https://www.example.com`)으로 자동 리디렉션됩니다.

6단계: 이메일 알림을 위한 SMTP 구성

PrestaShop 웹 사이트의 SMTP 설정을 구성하여 이메일 알림을 활성화할 수 있습니다. 이렇게 하려면 PrestaShop 웹 사이트의 관리 대시보드에 로그인해야 합니다. 탐색 창에서 고급 파라미터(Advanced Parameters)를 선택한 다음 이메일(E-mail)을 선택합니다. 또한 그에 따라 이메일 연락처를 조정해야 합니다. 이를 위해 탐색 창에서 상점 파라미터를 선택한 다음 연락처를 선택합니다.



자세한 내용은 PrestaShop 문서의 PrestaShop 사용 설명서 및 Bitnami 문서의 아웃바운드 이메일을 위한 SMTP 구성을 참조하세요.

> **Important**
>
> 포트 25, 465 또는 587을 사용하도록 SMTP를 구성할 경우 Lightsail 콘솔의 인스턴스 방화벽에서 해당 포트를 열어야 합니다. 자세한 내용은 Amazon Lightsail의 방화벽 규칙 추가 및 편집을 참조하세요.
>
> PrestaShop 웹 사이트에서 이메일을 보내도록 Gmail 계정을 구성한 경우 Gmail에 로그인할 때 사용하는 표준 암호 대신 앱 암호를 사용해야 합니다. 자세한 내용은 앱 암호로 로그인을 참조하세요.

7단계: Bitnami 및 PrestaShop 문서 읽기

Bitnami 문서를 읽고 PrestaShop 인스턴스와 웹 사이트에서 플러그 인 설치 및 테마 사용자 지정과 같은 관리 태스크를 수행하는 방법을 알아봅니다. 자세한 내용은 Bitnami 문서의 AWS 클라우드용 Bitnami PrestaShop 스택을 참조하세요.

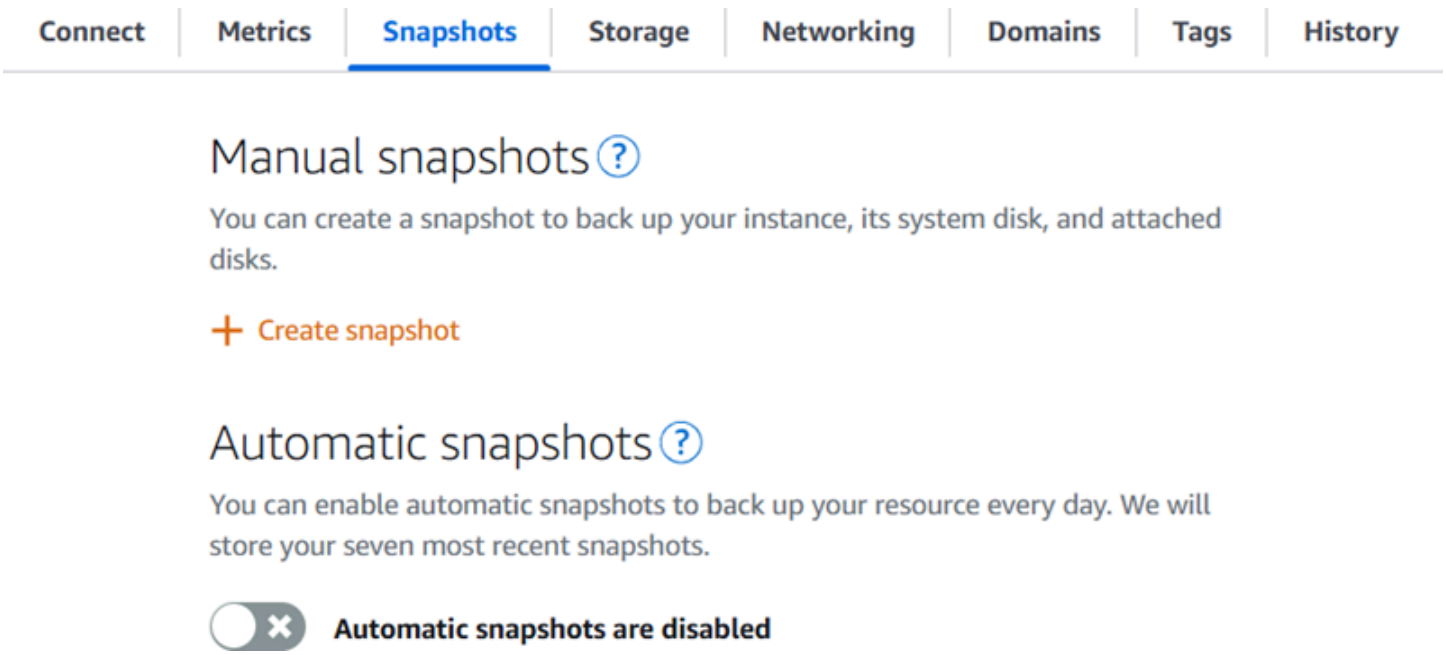
PrestaShop 문서를 살펴보고 PrestaShop 웹 사이트를 관리하는 방법을 익혀야 합니다. 자세한 내용은 PrestaShop 문서의 PrestaShop 사용 설명서를 참조하세요.

8단계: PrestaShop 인스턴스의 스냅샷 생성

원하는 방식으로 웹 사이트를 구성한 후 인스턴스의 주기적 스냅샷을 생성하여 백업합니다. 스냅샷은 시스템 디스크의 복사본이며, 인스턴스의 원본 구성입니다. 스냅샷은 스냅샷을 생성한 시점부터 인스턴스를 복원하는 데 필요한 모든 데이터를 포함합니다.

스냅샷을 수동으로 생성하거나 자동 스냅샷을 활성화하여 Lightsail에서 매일 스냅샷을 생성하도록 할 수 있습니다. 인스턴스에 문제가 있는 경우 스냅샷을 사용하여 새 교체 인스턴스를 생성할 수 있습니다.

스냅샷 탭의 인스턴스 관리 페이지에서 스냅샷 작업을 할 수 있습니다. 자세한 내용은 Amazon Lightsail의 스냅샷을 참조하세요.



Lightsail에서 Redmine 인스턴스 구성 및 보호

> 알고 계셨나요?
>
> Lightsail는 일별 스냅샷 7개를 저장하고 인스턴스에 대해 자동 스냅샷을 활성화하면 가장 오래된 스냅샷을 최신 스냅샷으로 자동 대체합니다. 자세한 내용은 Lightsail 인스턴스 및 디스크의 자동 스냅샷 구성을 참조하세요.

다음은 Amazon Lightsail에 Redmine 인스턴스를 설치하고 실행한 후 시작해야 할 몇 가지 단계입니다.

목차

- 1단계: Bitnami 설명서 읽기
- 2단계: Redmine 관리 대시보드에 액세스하기 위한 기본 애플리케이션 암호 가져오기
- 3단계: 인스턴스에 고정 IP 주소 연결
- 4단계: Redmine 웹 사이트의 관리 대시보드에 로그인
- 5단계: Redmine 웹 사이트로 등록된 도메인 이름의 트래픽 라우팅
- 6단계: Redmine 웹 사이트에 대해 HTTPS 구성
- 7단계: Redmine 설명서 읽기 및 웹 사이트 구성 계속
- 8단계: 인스턴스의 스냅샷 생성

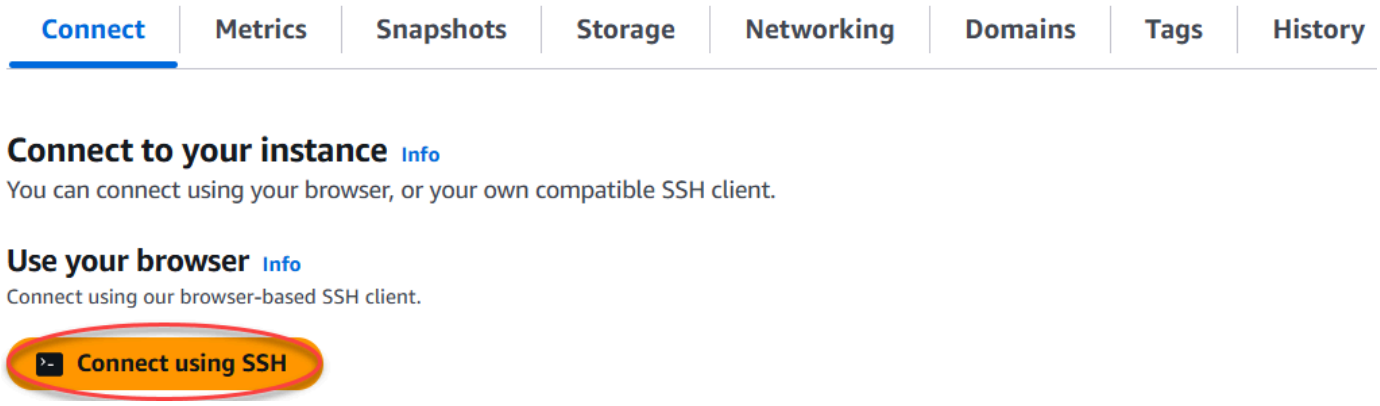
1단계: Bitnami 설명서 읽기

Redmine 설명서를 읽고 Ghost 애플리케이션을 구성하는 방법을 알아보세요. 자세한 내용은 AWS 클라우드용 Bitnami에서 패키징한 Redmine를 참조하세요.

2단계: Redmine 관리 대시보드에 액세스하기 위한 기본 애플리케이션 암호 가져오기

Redmine 웹 사이트의 관리 대시보드에 액세스하는 데 필요한 기본 애플리케이션 암호를 가져오려면 다음 절차를 완료하세요. 자세한 내용은 Amazon Lightsail의 Bitnami 인스턴스에 대한 애플리케이션 사용자 이름과 암호 가져오기를 참조하세요.

1. 인스턴스 관리 페이지의 연결(Connect) 탭에서 SSH로 연결(Connect using SSH)을 선택합니다.



2. 연결한 후 다음 명령을 입력하여 애플리케이션 암호를 가져옵니다.

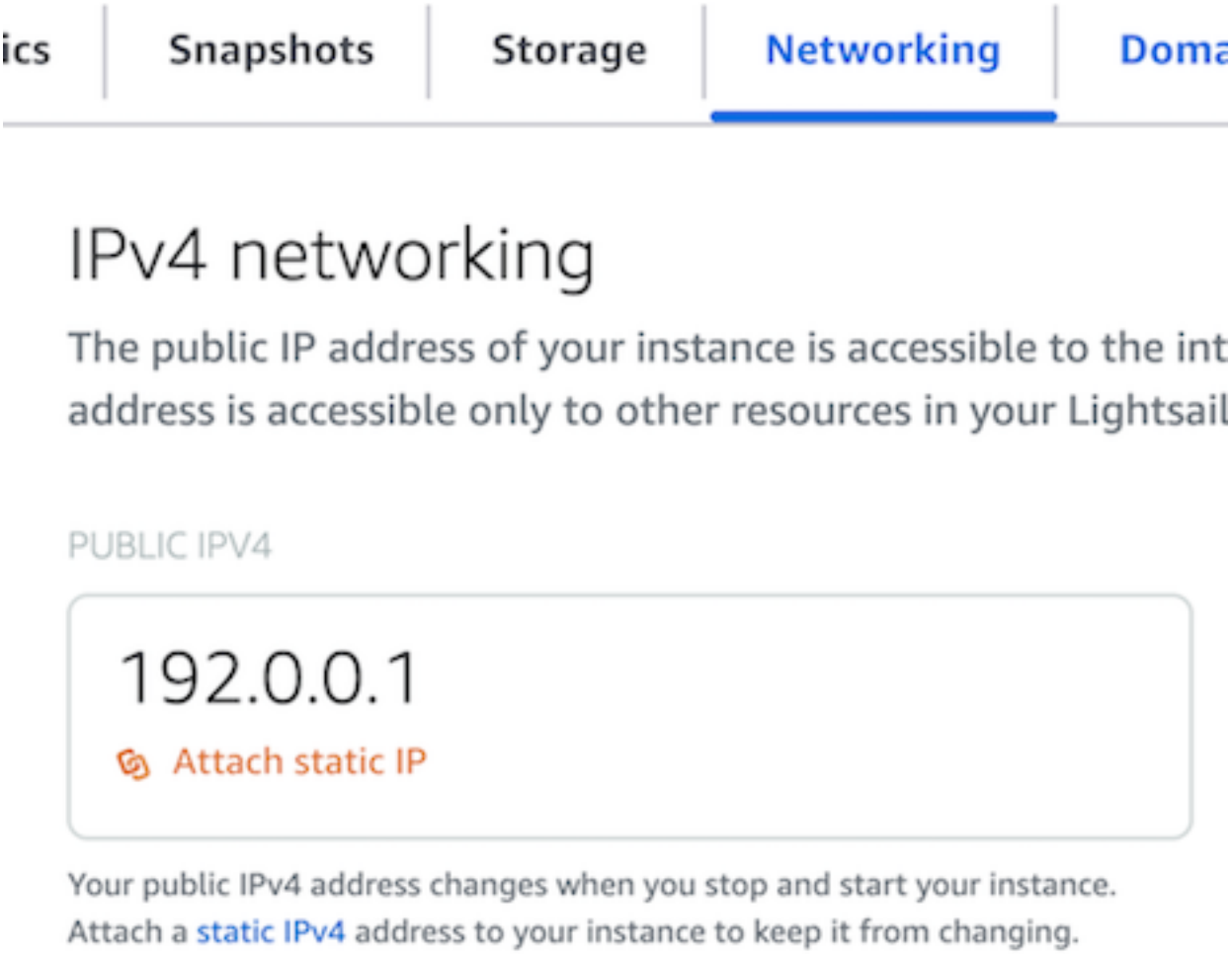
```
cat $HOME/bitnami_application_password
```

기본 애플리케이션 암호가 포함된 다음 예제와 유사한 응답이 표시되어야 합니다.

3단계: 인스턴스에 고정 IP 주소 연결

인스턴스에 연결된 기본 동적 퍼블릭 IP 주소는 인스턴스를 중지하고 시작할 때마다 변경됩니다. 퍼블릭 IP 주소가 변경되지 않도록 고정 IP 주소를 생성하고 인스턴스에 연결할 수 있습니다. 나중에 인스턴스에 도메인 이름을 사용하면 인스턴스를 중지하고 시작할 때마다 도메인의 DNS 레코드를 업데이트할 필요가 없습니다. 각 인스턴스에 1개의 고정 IP 주소만 연결할 수 있습니다.

인스턴스 관리 페이지의 네트워킹 탭에서 고정 IP 생성 또는 고정 IP 연결을 선택합니다. 인스턴스에 연결할 수 있는 고정 IP를 생성해 둔 경우에는 페이지의 지침을 따릅니다. 자세한 내용은 고정 IP를 생성하여 인스턴스에 연결을 참조하세요.



4단계: Redmine 웹 사이트의 관리 대시보드에 로그인

이제 기본 애플리케이션 암호가 있으므로 다음 절차를 완료하여 Redmine 웹 사이트의 홈 페이지로 이동하고 관리 대시보드에 로그인합니다. 로그인한 후 웹 사이트 사용자 지정 및 관리 변경을 시작할 수 있습니다. Joomla!에서 수행할 수 있는 작업에 대한 자세한 내용을 알아보려면 이 가이드 뒷부분의 7단계: Redmine 설명서 읽기 및 웹 사이트 구성 계속 섹션을 참조하세요.

1. 인스턴스 관리 페이지의 연결(Connect) 탭에서 인스턴스의 퍼블릭 IP 주소를 기록해 둡니다. 퍼블릭 IP 주소는 인스턴스 관리 페이지의 헤더 섹션에도 표시됩니다.

2. 인스턴스의 퍼블릭 IP 주소로 이동합니다(예: `http://203.0.113.0`으로 이동).

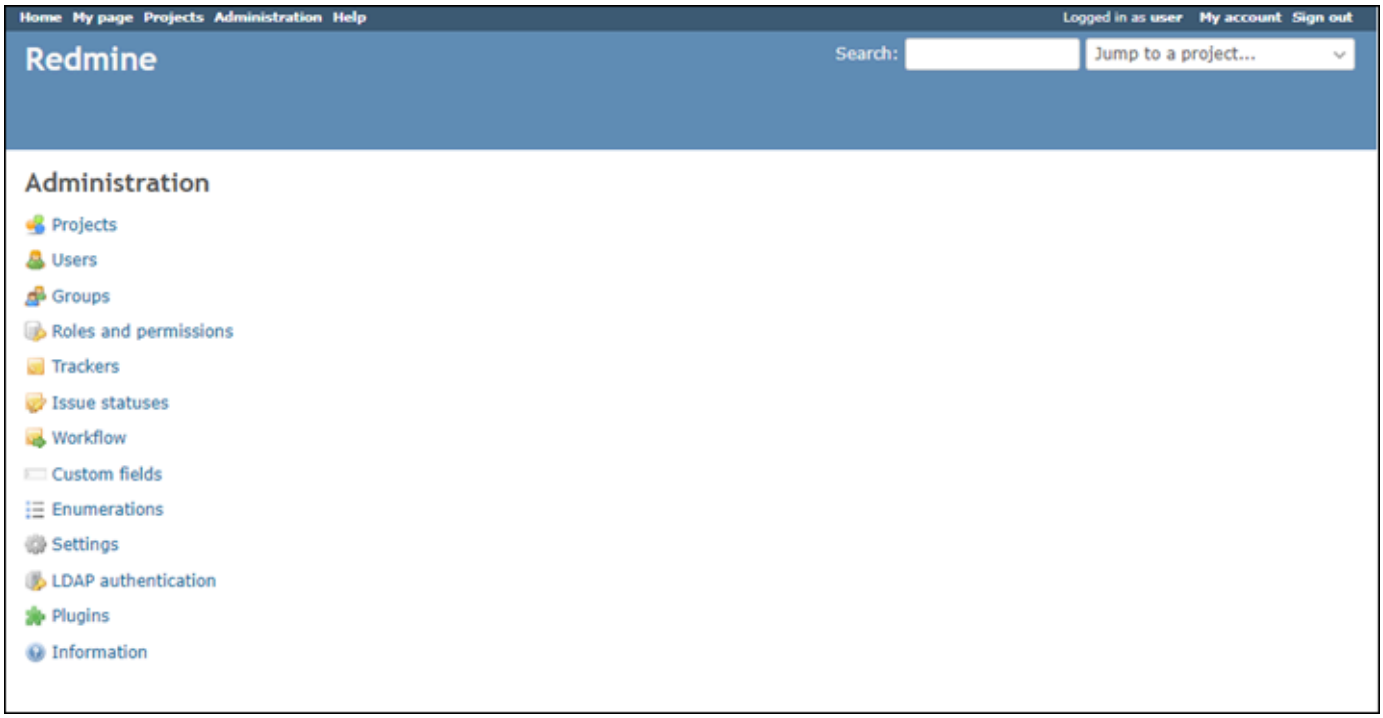
   Redmine 웹 사이트의 홈 페이지가 나타납니다.

3. Redmine 웹 사이트 홈 페이지의 오른쪽 하단 모서리에 있는 관리(Manage)를 선택합니다.

   관리(Manage) 배너가 표시되지 않는 경우 `http://`*`<PublicIP>`*`/admin`을 통해 로그인 페이지로 이동할 수 있습니다. *`<PublicIP>`*을(를) 인스턴스의 퍼블릭 IP 주소로 바꿉니다.

4. 이 가이드의 초반부에서 검색한 기본 사용자 이름(`user`) 및 기본 암호를 사용하여 로그인합니다.

   Redmine 관리 대시보드가 나타납니다.



5단계: Redmine 웹 사이트로 등록된 도메인 이름의 트래픽 라우팅

등록된 도메인 이름(예: `example.com`)의 트래픽을 Redmine 웹 사이트로 라우팅하려면 도메인의 DNS에 레코드를 추가하면 됩니다. DNS 레코드는 일반적으로 도메인을 등록한 등록 대행자가 관리 및 호스팅합니다. 그러나 Lightsail 콘솔을 사용하여 관리할 수 있도록 도메인의 DNS 레코드 관리를 Lightsail로 이전하는 것이 좋습니다.

Lightsail 콘솔 홈 페이지의 도메인 및 DNS(Domains & DNS) 탭에서 DNS 영역 생성(Create DNS zone)을 선택하고 페이지의 지침에 따릅니다. 자세한 내용은 DNS 영역을 생성하여 Lightsail에서 도메인의 DNS 레코드 관리를 참조하세요.

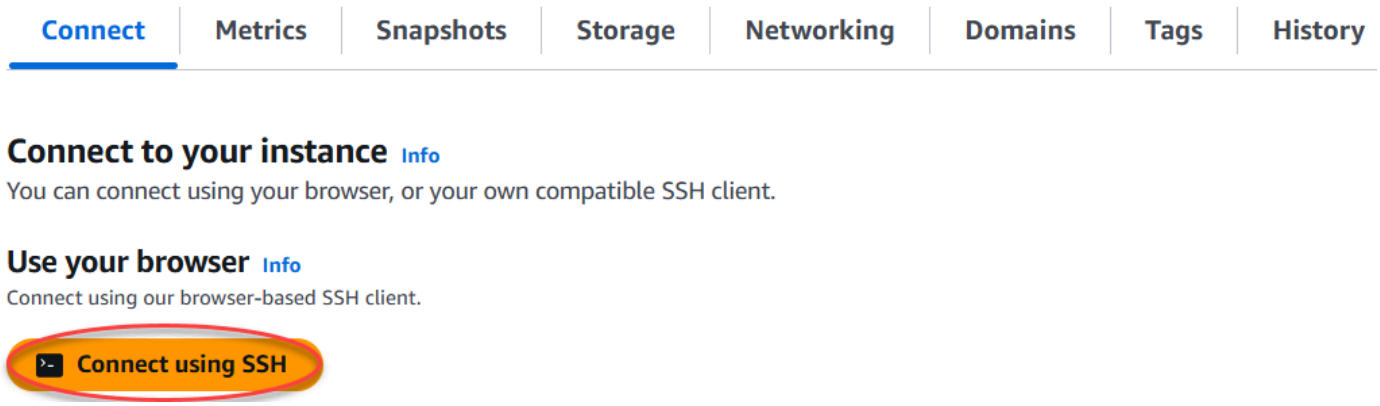
인스턴스에 대해 구성한 도메인 이름으로 이동하면 Redmine 웹 사이트의 홈 페이지로 리디렉션됩니다. 다음으로 Redmine 웹 사이트에 대한 HTTPS 연결을 활성화하기 위해 SSL/TLS 인증서를 생성하고 구성해야 합니다. 자세한 내용을 보려면 이 가이드의 다음 6단계: Redmine 웹 사이트에 대해 HTTPS 구성 섹션으로 계속하세요.

6단계: Redmine 웹 사이트에 대해 HTTPS 구성

Redmine 웹 사이트에서 HTTPS를 구성하려면 다음 절차를 완료하세요. 이 단계에서는 Let's Encrypt SSL/TLS 인증서를 요청하기 위한 명령줄 도구인 Bitnami HTTPS Configuration Tool(`bncert-tool`)을 사용하는 방법을 보여줍니다. 자세한 내용을 알아보려면 Bitnami 설명서의 Learn About The Bitnami HTTPS Configuration Tool(Bitnami Configuration Tool에 대해 알아보기)을 참조하세요.

> **Important**
>
> 이 절차를 시작하기 전에 Redmine 인스턴스로 트래픽을 라우팅하도록 도메인을 구성했는지 확인합니다. 그렇지 않으면 SSL/TLS 인증서 검증 프로세스가 실패합니다.

1. 인스턴스 관리 페이지의 연결(Connect) 탭에서 SSH로 연결(Connect using SSH)을 선택합니다.



2. 연결한 후 다음 명령을 입력하여 bncert 도구가 인스턴스에 설치되었는지 확인합니다.

```
sudo /opt/bitnami/bncert-tool
```

다음과 같은 응답 중 하나가 표시됩니다.

- 응답에 명령을 찾을 수 없음(command not found)이 표시되면 bncert 도구가 인스턴스에 설치되지 않은 것입니다. 이 절차의 다음 단계를 계속 진행하여 인스턴스에 bncert 도구를 설치합니다.
- 응답에 Bitnami HTTPS 구성 도구 시작(Welcome to the Bitnami HTTPS configuration tool)이 표시되면 bncert 도구가 인스턴스에 설치된 것입니다. 이 절차의 8단계로 계속합니다.
- bncert 도구가 일시적으로 인스턴스에 설치된 경우 업데이트된 버전의 도구를 사용할 수 있다는 메시지가 표시될 수 있습니다. 다운로드하도록 선택하고 `sudo /opt/bitnami/bncert-tool` 명령을 입력하여 bncert 도구를 다시 실행합니다. 이 절차의 8단계로 계속합니다.

3. 다음 명령을 입력하여 bncert 실행 파일을 인스턴스로 다운로드합니다.

```
wget -O bncert-linux-x64.run https://downloads.bitnami.com/files/bncert/latest/
bncert-linux-x64.run
```

4. 다음 명령을 입력하여 인스턴스에서 bncert 도구 실행 파일의 디렉터리를 생성합니다.

```
sudo mkdir /opt/bitnami/bncert
```

5. 다음 명령을 입력하여 bncert에서 프로그램으로 실행할 수 있는 파일을 실행하도록 합니다.

```
sudo chmod +x /opt/bitnami/bncert/bncert-linux-x64.run
```

6. 다음 명령을 입력하여 sudo /opt/bitnami/bncert-tool 명령을 입력할 때 bncert 도구를 실행하는 심볼 링크를 생성합니다.

```
sudo ln -s /opt/bitnami/bncert/bncert-linux-x64.run /opt/bitnami/bncert-tool
```

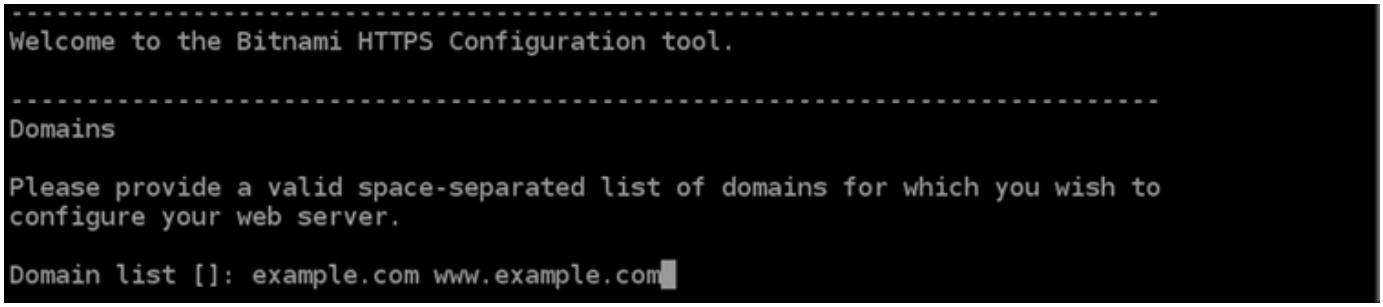
이제 인스턴스에 bncert 도구를 설치했습니다.

7. 다음 명령을 입력하여 bncert 도구를 실행합니다.

```
sudo /opt/bitnami/bncert-tool
```

8. 다음 예와 같이 기본 도메인 이름과 대체 도메인 이름을 공백으로 구분하여 입력합니다.

도메인이 트래픽을 인스턴스의 퍼블릭 IP 주소로 라우팅하도록 구성되지 않은 경우, `bncert` 도구에서 계속하기 전에 해당 구성을 설정하라는 메시지를 표시합니다. 도메인은 `bncert` 도구를 사용하여 인스턴스에서 HTTPS를 활성화한 인스턴스의 퍼블릭 IP 주소로 트래픽을 라우팅해야 합니다. 이렇게 해야 도메인을 소유하고 있음을 확인하고 인증서를 검증하는 역할을 할 수 있습니다.



9. bncert 도구는 웹 사이트의 리디렉션을 어떻게 구성할지 묻는 메시지를 표시합니다. 사용할 수 있는 옵션은 다음과 같습니다.

   - HTTP에서 HTTPS로 리디렉션 활성화(Enable HTTP to HTTPS redirection) - HTTP 버전 웹 사이트(예: http:/example.com)를 방문하는 사용자를 HTTPS 버전(예: https://example.com)으로 자동 리디렉션할지 지정합니다. 모든 방문자가 암호화된 연결을 사용하도록 강제하기 때문에 이 옵션을 활성화하는 것이 좋습니다. Y를 입력하고 Enter 키를 눌러 활성화합니다.
   - 비 www에서 www로 리디렉션 활성화(Enable non-www to www redirection) - 도메인의 정점(https://example.com)을 방문하는 사용자를 도메인의 www 하위 도메인(https://www.example.com)으로 자동 리디렉션할지 지정합니다. 이 옵션은 활성화하는 것이 좋습니다. 그러나 Google의 웹 마스터 도구와 같은 검색 엔진 도구에서 도메인의 정점을 기본 웹 사이트 주소로 지정했거나 정점이 IP를 직접 가리키고 www 하위 도메인이 CNAME 레코드를 통해 정점을 참조하는 경우, 이 옵션을 비활성화하고 대체 옵션(www에서 비 www로 리디렉션 활성화)을 활성화할 수 있습니다. Y를 입력하고 Enter 키를 눌러 활성화합니다.
   - www에서 비 www로 리디렉션 활성화(Enable www to non-www redirection) - 도메인의 www 하위 도메인(https://www.example.com)을 방문하는 사용자를 도메인의 정점(https://example.com)으로 자동 리디렉션할지 지정합니다. 비 www에서 www로 리디렉션을 활성화한 경우 이 옵션을 비활성화하는 것이 좋습니다. N를 입력하고 Enter 키를 눌러 비활성화합니다.

   선택한 내용은 다음 예와 같아야 합니다.

```
Enable/disable redirections

Please select the redirections you wish to enable or disable on your Bitnami
installation.



Enable HTTP to HTTPS redirection [Y/n]: Y



Enable non-www to www redirection [Y/n]: Y



Enable www to non-www redirection [y/N]: N
```

10. 적용될 변경 사항의 목록이 나열됩니다. Y를 입력하고 Enter 키를 눌러 확인하고 계속합니다.

```
Changes to perform

The following changes will be performed to your Bitnami installation:

1. Stop web server
2. Configure web server to use a free Let's Encrypt certificate for the domains:
example.com www.example.com
3. Configure a cron job to automatically renew the certificate each month
4. Configure web server name to: example.com
5. Enable HTTP to HTTPS redirection (example: redirect http://example.com to
https://example.com)
6. Enable non-www to www redirection (example: redirect example.com to
 www.example.com)
7. Start web server once all changes have been performed



Do you agree to these changes? [Y/n]: Y
```

11. Let's Encrypt 인증서에 연결할 이메일 주소를 입력하고 Enter 키를 누릅니다.

```
Create a free HTTPS certificate with Let's Encrypt

Please provide a valid e-mail address for which to associate your Let's Encrypt
certificate.

Domain list: example.com www.example.com

Server name: example.com

E-mail address []:
```

12. Let's Encrypt 구독자 계약을 검토합니다. Y를 입력하고 Enter 키를 눌러 계약을 수락하고 계속합니다.

```
The Let's Encrypt Subscriber Agreement can be found at:

https://letsencrypt.org/documents/LE-SA-v1.2-November-15-2017.pdf

Do you agree to the Let's Encrypt Subscriber Agreement? [Y/n]:
```

인증서 요청 및 지정한 리디렉션 구성을 비롯하여 인스턴스에서 HTTPS를 활성화하는 작업이 수행됩니다.

```
Performing changes to your installation

The Bitnami HTTPS Configuration Tool will perform any necessary actions to your
Bitnami installation. This may take some time, please be patient.
```

인증서가 성공적으로 발급되고 검증됩니다. 다음 예와 유사한 메시지가 표시되면 인스턴스에서 리디렉션이 성공적으로 구성된 것입니다.

```
Success

The Bitnami HTTPS Configuration Tool succeeded in modifying your installation.

The configuration report is shown below.

Backup files:
* /opt/bitnami/apache2/conf/httpd.conf.back.202005290035
* /opt/bitnami/apache2/conf/bitnami/bitnami-apps-prefix.conf.back.202005290035
* /opt/bitnami/apache2/conf/bitnami/bitnami.conf.back.202005290035

Find more details in the log file:

/tmp/bncert-202005290035.log

If you find any issues, please check Bitnami Support forums at:

https://community.bitnami.com

Press [Enter] to continue:
```

`bncert` 도구는 인증서가 만료되기 80일 전마다 인증서를 자동으로 갱신합니다. 인스턴스와 함께 추가 도메인 및 하위 도메인을 사용하고 이러한 도메인에서 HTTPS를 활성화하려면 위의 단계를 반복합니다.

이제 Redmine 인스턴스에서 HTTPS가 활성화되었습니다. 다음에 구성한 도메인을 사용하여 Redmine 웹 사이트로 이동하면 HTTPS 연결로 리디렉션됩니다.

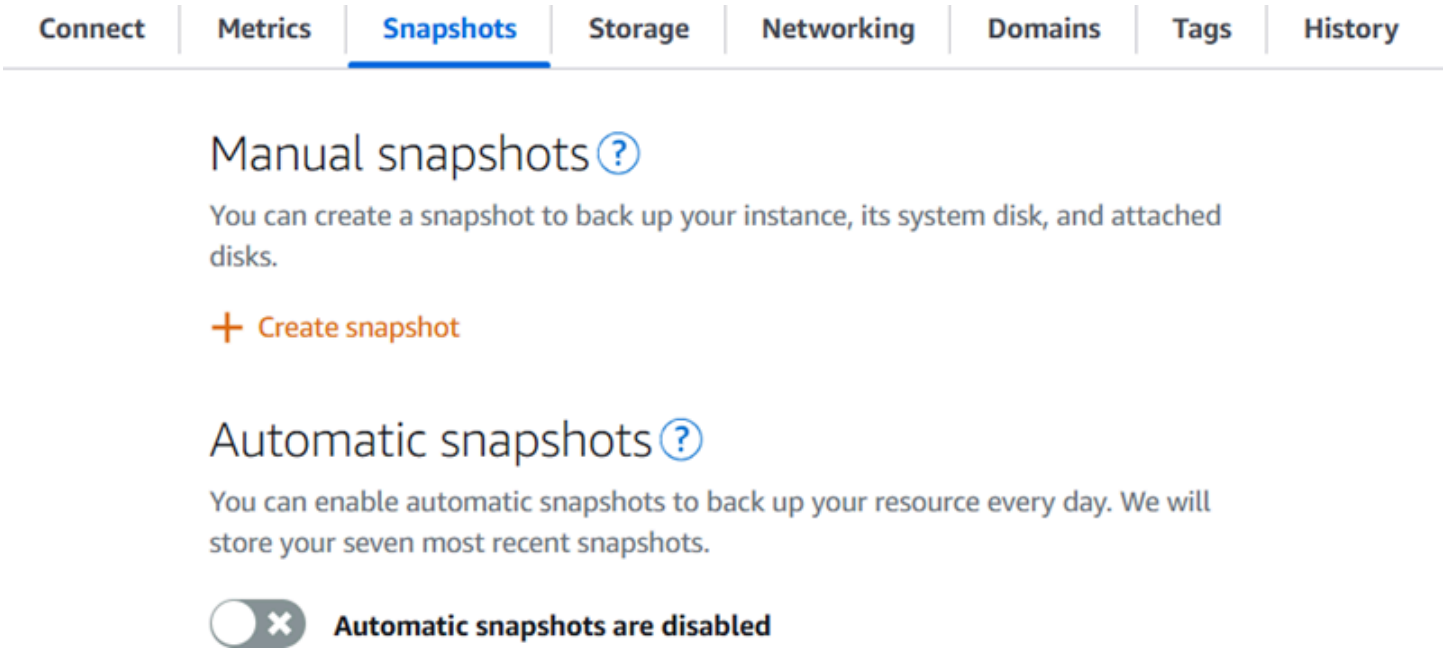
7단계: Redmine 설명서 읽기 및 웹 사이트 구성 계속

Redmine 설명서를 읽고 웹 사이트를 관리하고 사용자 지정하는 방법을 알아보세요. 자세한 내용을 알아보려면 Redmine 가이드를 참조하세요.

8단계: 인스턴스의 스냅샷 생성

원하는 방식으로 Redmine 웹 사이트를 구성한 후 인스턴스의 주기적 스냅샷을 생성하여 백업합니다. 스냅샷을 수동으로 생성하거나 자동 스냅샷을 활성화하여 Lightsail에서 매일 스냅샷을 생성하도록 할 수 있습니다. 인스턴스에 문제가 있는 경우 스냅샷을 사용하여 새 교체 인스턴스를 생성할 수 있습니다. 자세한 내용은 스냅샷을 참조하세요.

인스턴스 관리 페이지의 스냅샷(Snapshot) 탭에서 스냅샷 생성(Create a snapshot)을 선택하여 자동 스냅샷을 활성화합니다.



자세한 내용은 Amazon Lightsail에서 Linux 또는 Unix 인스턴스의 스냅샷 생성 또는 Amazon Lightsail에서 인스턴스 또는 디스크의 자동 스냅샷 활성화 또는 비활성화를 참조하세요.

에서 Ruby on Rails 설정 Lightsail

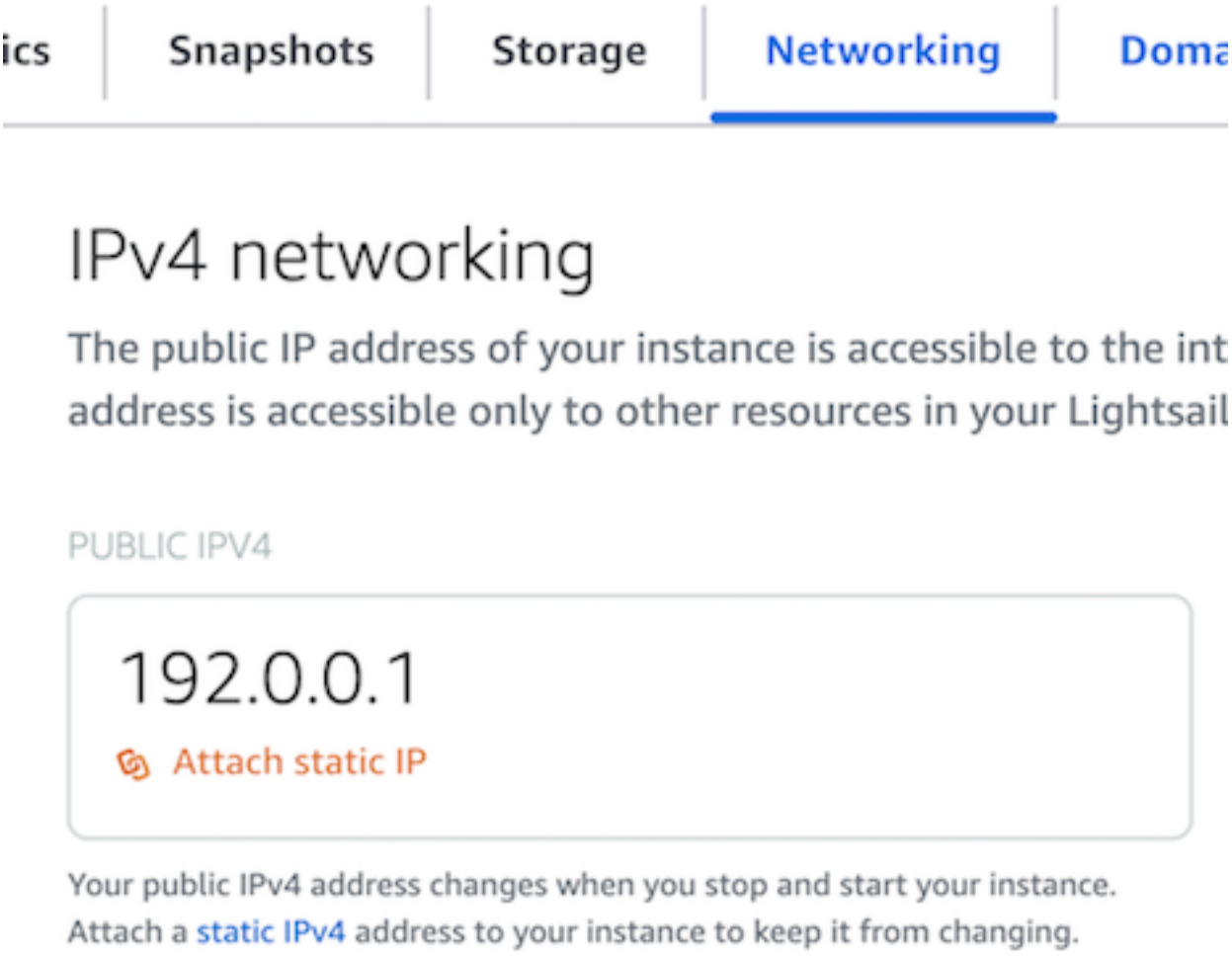
> 알고 계셨나요?
>
> Lightsail는 일별 스냅샷 7개를 저장하고 인스턴스에 대해 자동 스냅샷을 활성화하면 가장 오래된 스냅샷을 최신 스냅샷으로 자동 대체합니다. 자세한 내용은 Lightsail 인스턴스 및 디스크의 자동 스냅샷 구성을 참조하세요.

다음은 Ruby on Rails 인스턴스가에서 실행된 후 시작하기 위해 수행해야 하는 몇 가지 단계입니다. Amazon Lightsail

1단계: Ruby on Rails 인스턴스에 고정 IP 주소 연결

인스턴스에 연결된 기본 동적 퍼블릭 IP 주소는 인스턴스를 중지하고 시작할 때마다 변경됩니다. 퍼블릭 IP 주소가 변경되지 않도록 고정 IP 주소를 생성하고 인스턴스에 연결할 수 있습니다. 나중에 인스턴스에 도메인 이름을 사용하면 인스턴스를 중지하고 시작할 때마다 도메인의 DNS 레코드를 업데이트할 필요가 없습니다. 각 인스턴스에 1개의 고정 IP 주소만 연결할 수 있습니다.

인스턴스 관리 페이지의 네트워킹 탭에서 고정 IP 생성 또는 고정 IP 연결을 선택합니다. 인스턴스에 연결할 수 있는 고정 IP를 생성해 둔 경우에는 페이지의 지침을 따릅니다. 자세한 내용은 고정 IP를 생성하여 인스턴스에 연결을 참조하세요.



2단계: Ruby on Rails 인스턴스 시작 페이지 방문

인스턴스의 고정 IP 주소로 이동하여 인스턴스에 설치된 애플리케이션에 액세스합니다.

1. 인스턴스 관리 페이지의 연결 탭에서 고정 IP를 기록해 둡니다.
2. 와 같은 고정 IP 주소를 찾습니다`http://192.0.0.1:3000`.

자세한 내용은 Ruby on Rails 가이드를 참조하세요.

3단계: 애플리케이션 배포

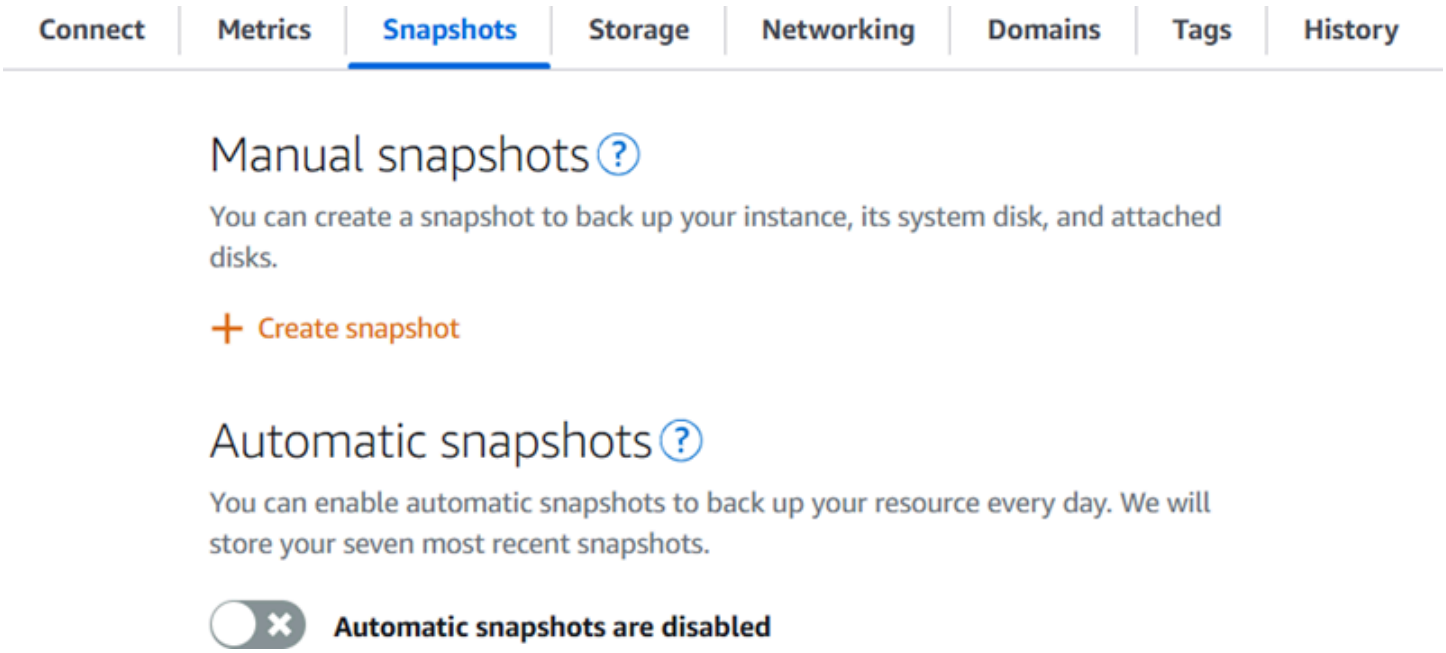
1. SFTP를 사용하여 Lightsail Linux 인스턴스로 안전하게 파일 전송의 지침에 따라 애플리케이션을 에 복사합니다. `/home/ec2-user/my_app`
2. 인스턴스 관리 페이지의 연결(Connect) 탭에서 SSH로 연결(Connect using SSH)을 선택합니다.
3. `sudo systemctl restart rails-server` 실행
4. 인스턴스의 고정 IP 주소로 이동합니다.

4단계: Ruby on Rails 인스턴스의 스냅샷 생성

원하는 방식으로 웹 사이트를 구성한 후 인스턴스의 주기적 스냅샷을 생성하여 백업합니다. 스냅샷은 시스템 디스크의 복사본이며, 인스턴스의 원본 구성입니다. 스냅샷은 스냅샷을 생성한 시점부터 인스턴스를 복원하는 데 필요한 모든 데이터를 포함합니다.

스냅샷을 수동으로 생성하거나 자동 스냅샷을 활성화하여 Lightsail에서 매일 스냅샷을 생성하도록 할 수 있습니다. 인스턴스에 문제가 있는 경우 스냅샷을 사용하여 새 교체 인스턴스를 생성할 수 있습니다.

스냅샷 탭의 인스턴스 관리 페이지에서 스냅샷 작업을 할 수 있습니다. 자세한 내용은 Amazon Lightsail의 스냅샷을 참조하세요.

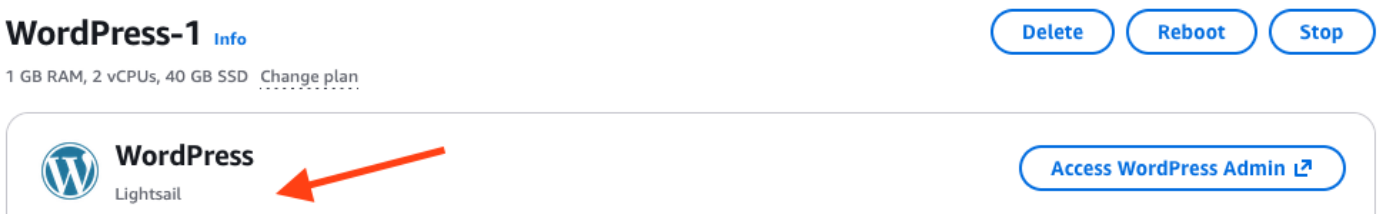


에서 WordPress 배포 및 관리 Lightsail

> 알고 계셨나요?
>
> Lightsail는 일별 스냅샷 7개를 저장하고 인스턴스에 대해 자동 스냅샷을 활성화하면 가장 오래된 스냅샷을 최신 스냅샷으로 자동 대체합니다. 자세한 내용은 Lightsail 인스턴스 및 디스크의 자동 스냅샷 구성을 참조하세요.

빠른 시작 안내서를 통해 Amazon Lightsail에서 WordPress 인스턴스를 시작하고 구성하는 방법을 알아봅니다.

다음은에서 WordPress 인스턴스를 설치하고 실행한 후 시작해야 하는 몇 가지 단계입니다Amazon Lightsail. 시작하기 전에 인스턴스 관리 페이지에서 블루프린트 공급업체를 식별합니다.



WordPress 인스턴스에 적합한 가이드를 선택합니다.

Bitnami

1단계: WordPress 인스턴스 구성

다음을 구성하는 단계별 안내 워크플로를 사용하여 WordPress 인스턴스를 구성할 수 있습니다.

- 등록된 도메인 이름 - WordPress 사이트에는 기억하기 쉬운 도메인 이름이 필요합니다. 사용자는 이 도메인 이름을 지정하여 WordPress 사이트에 액세스합니다. 자세한 내용은 도메인 및 DNS 섹션을 참조하세요.
- DNS 관리 - 도메인의 DNS 레코드를 관리하는 방법을 결정해야 합니다. DNS 레코드는 도메인이나 하위 도메인이 연결된 IP 주소나 호스트 이름을 DNS 서버에 알립니다. DNS 영역에는 도메인의 DNS 레코드가 포함됩니다. 자세한 내용은 the section called “Lightsail의 DNS” 섹션을 참조하세요.
- 고정 IP 주소 - 인스턴스를 중지했다가 시작하면 WordPress 인스턴스의 기본 퍼블릭 IP 주소가 변경됩니다. 인스턴스에 고정 IP 주소를 연결할 때 인스턴스를 중지했다가 시작한 경우에도 동일하게 유지됩니다. 자세한 내용은 the section called “IP 주소” 섹션을 참조하세요.
- SSL/TLS 인증서 - 검증된 인증서를 생성하고 인스턴스에 설치한 후 등록된 도메인을 통해 인스턴스로 라우팅되는 트래픽이 HTTPS를 사용하여 암호화되도록 WordPress 웹 사이트에 HTTPS를 활성화할 수 있습니다. 자세한 내용은 the section called “HTTPS 활성화” 섹션을 참조하세요.

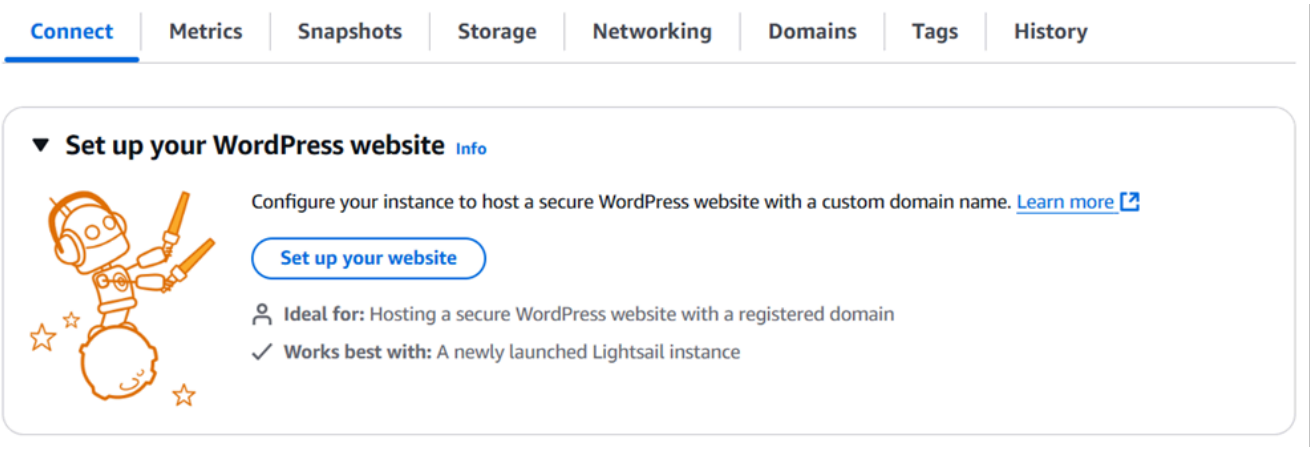
> **Tip**
>
> 시작하기 전에 다음 팁을 확인하세요. 문제 해결 정보는 WordPress 설정 문제 해결을 참조하세요.
>
> - 설치는 2023년 1월 1일 이후에 생성된 WordPress 버전 6 이상의 Lightsail 인스턴스를 지원합니다.
> - 설정 중에 실행되는 Certbot 종속성 파일, HTTPS 재작성 스크립트 및 인증서 갱신 스크립트는 인스턴스의 `/opt/bitnami/lightsail/scripts/` 디렉터리에 저장됩니다.
> - 인스턴스가 Running 상태여야 합니다. 인스턴스가 방금 시작된 경우 SSH 연결이 준비될 때까지 몇 분 정도 기다립니다.
> - 인스턴스 방화벽의 포트 22, 80 및 443은 설정이 실행되는 동안 모든 IP 주소에서 TCP 연결을 허용해야 합니다. 자세한 내용은 인스턴스 방화벽을 참조하세요.
> - Apex 도메인(`example.com`)과 `www` 하위 도메인(`www.example.com`)의 트래픽을 가리키는 DNS 레코드를 추가하거나 업데이트할 때 인터넷 전체에 전파해야 합니다.

nslookup 또는 MxToolbox의 DNS 조회와 같은 도구를 사용하여 DNS 변경 사항이 적용되었는지 확인할 수 있습니다.

- 2023년 1월 1일 이전에 생성된 Wordpress 인스턴스에는 더 이상 사용되지 않는 Certbot Personal Package Archive(PPA) 리포지토리가 포함되어 있어 웹 사이트 설정이 실패할 수 있습니다. 설정 중에 이 리포지토리가 있는 경우 기존 경로에서 제거되고 인스턴스의 `~/opt/bitnami/lightsail/repo.backup` 위치에 백업됩니다. 더 이상 사용되지 않는 PPA에 대한 자세한 내용은 Canonical 웹 사이트의 Certbot PPA를 참조하세요.
- 인증서를 암호화하면 60~90일마다 자동으로 갱신됩니다.
- 설정이 진행 중인 동안에는 인스턴스를 중지하거나 변경하지 마세요. 인스턴스를 구성하는 데 최대 15분이 걸릴 수 있습니다. 인스턴스 연결 탭에서 각 단계의 진행 상황을 볼 수 있습니다.

웹 사이트 설정 마법사를 사용하여 인스턴스를 구성

1. 인스턴스 관리 페이지의 연결 탭에서 웹 사이트 설정을 선택합니다.



2. 도메인 이름 지정에서 기존 Lightsail 관리형 도메인을 사용하거나, Lightsail에 새 도메인을 등록하거나, 다른 도메인 등록 기관을 사용하여 등록한 도메인을 사용합니다. 이 도메인 사용을 선택하여 다음 단계로 이동합니다.
3. DNS 구성에 대해 다음 작업 중 한 가지를 수행하세요.
   - Lightsail DNS 영역을 사용하려면 Lightsail 관리형 도메인을 선택합니다. 다음 단계로 이동하려면 이 DNS 영역 사용을 선택합니다.
   - 도메인의 DNS 레코드를 관리하는 호스팅 서비스를 사용하려면 타사 도메인을 선택합니다. 나중에 사용하기로 결정할 경우를 대비하여 Lightsail 계정에 일치하는 DNS 영역을 생성합니다. 타사 DNS 사용을 선택하여 다음 단계로 이동합니다.
4. 고정 IP 주소 생성에서 고정 IP 주소의 이름을 입력한 다음 고정 IP 생성을 선택합니다.

5. 도메인 할당 관리에서 할당 추가를 선택하고 도메인 유형을 선택한 다음 추가를 선택합니다. 계속을 선택하여 다음 단계로 이동합니다.
6. SSL/TLS 인증서 생성에서 도메인 및 하위 도메인을 선택하고 이메일 주소를 입력한 다음 Lightsail에 권한을 부여하여 인스턴스에서 Let's Encrypt 인증서를 구성하기를 선택하고 인증서 생성을 선택합니다. Lightsail 리소스 구성을 시작합니다.

   설정이 진행 중인 동안에는 인스턴스를 중지하거나 변경하지 마세요. 인스턴스를 구성하는 데 최대 15분이 걸릴 수 있습니다. 인스턴스 연결 탭에서 각 단계의 진행 상황을 볼 수 있습니다.
7. 웹 사이트 설정이 완료되면 도메인 할당 단계에서 지정한 URL이 WordPress 사이트를 여는지 확인합니다.

2단계: WordPress 웹 사이트의 기본 애플리케이션 암호 가져오기

WordPress 웹 사이트의 관리 대시보드에 로그인하려면 기본 애플리케이션 암호가 필요합니다.

WordPress 관리자 기본 암호 가져오기

1. WordPress 인스턴스의 인스턴스 관리 페이지를 엽니다.
2. WordPress 패널에서 기본 암호 검색을 선택합니다. 이렇게 하면 페이지 하단의 액세스 기본 암호가 확장됩니다.

3. CloudShell 시작을 선택합니다. 이는 페이지 하단에 분할 창을 엽니다.
4. 복사를 선택한 다음 콘텐츠를 CloudShell 창에 붙여넣습니다. 커서를 CloudShell 프롬프트에 놓고 Ctrl+V를 누르거나 마우스 오른쪽 버튼을 클릭하여 메뉴를 열고 붙여넣기를 선택할 수 있습니다.

5. CloudShell 창에 표시된 암호를 기록해 둡니다. 나중에 WordPress 웹사이트의 관리 대시보드에 로그인하는 데 사용합니다.

```
[cloudshell-user@ip-[redacted] ~]$ AWS_REGION=us-east-1 ~/lightsail_connect WordPress-1 cat bitnami_application_password
JKzhKzhKzhR!
```

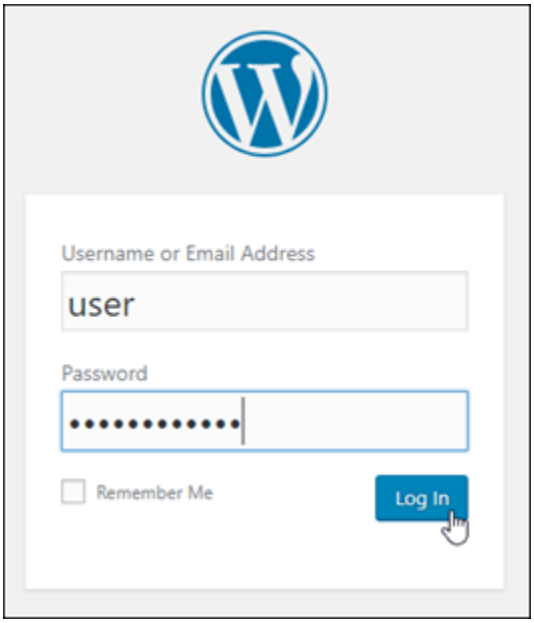
3단계: WordPress 웹 사이트에 로그인

이제 기본 사용자 암호가 있으므로 WordPress 웹 사이트의 홈 페이지로 이동하여 관리 대시보드에 로그인합니다. 로그인한 후 기본 암호를 변경할 수 있습니다.

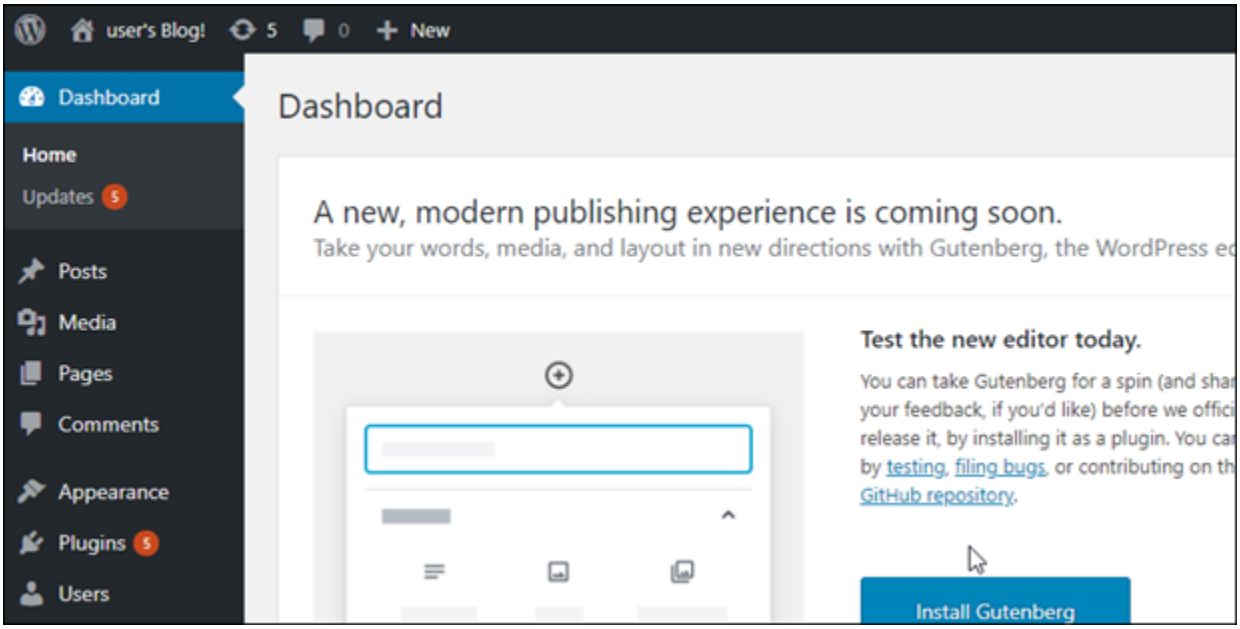
관리자 대시보드에 로그인

1. WordPress 인스턴스의 인스턴스 관리 페이지를 엽니다.
2. WordPress 패널에서 WordPress 관리자 액세스를 선택합니다.
3. WordPress 관리 대시보드 액세스 패널의 퍼블릭 IP 주소 사용에서 다음 형식의 링크를 선택합니다.

   http://*public-ipv4-address*./wp-admin
4. 사용자 이름 또는 이메일 주소 상자에 **user**를 입력합니다.
5. 암호에 이전 단계에서 얻은 암호를 입력합니다.
6. 그런 다음 로그인을 선택합니다.



이제 관리 작업을 수행할 수 있는 WordPress 웹사이트의 관리 대시보드에 로그인되었습니다. WordPress 웹사이트 관리에 대한 자세한 내용은 WordPress 설명서의 WordPress Codex를 참조하세요.



4단계: Bitnami 설명서 읽기

Bitnami 설명서를 읽고 플러그인 설치, 테마 사용자 지정, WordPress 버전 업그레이드 등 WordPress 웹사이트에 대한 관리 작업을 수행하는 방법에 대해 알아보세요.

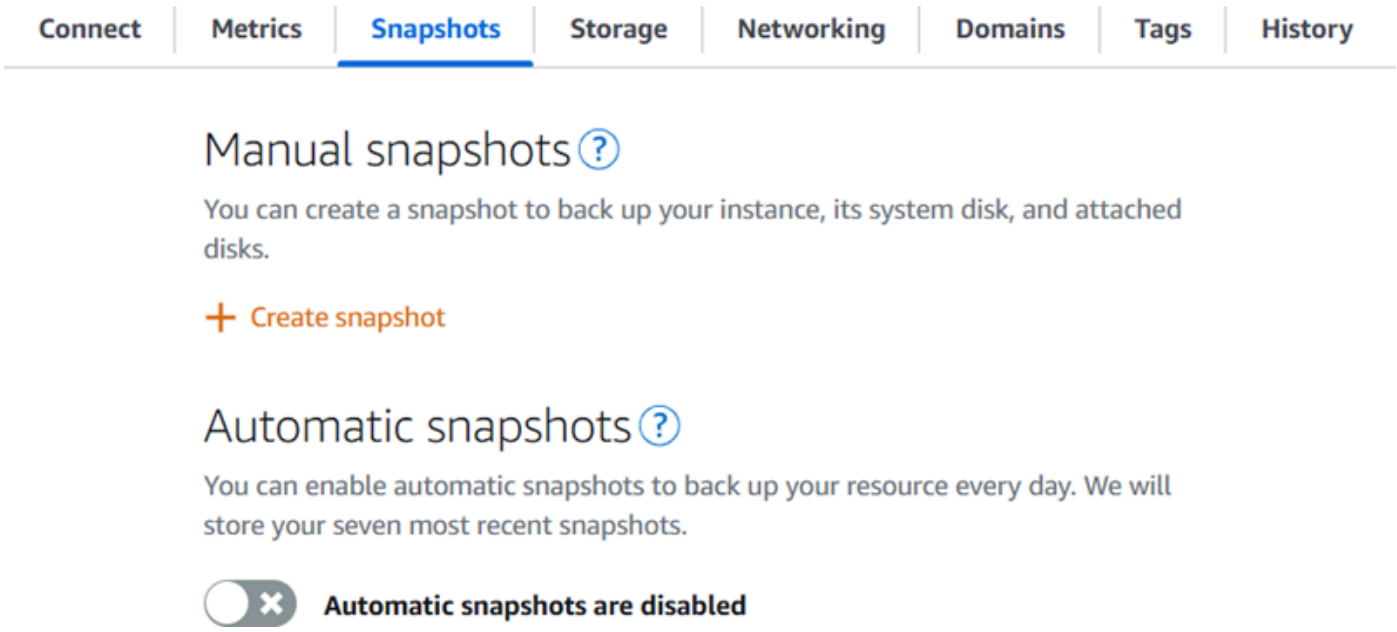
자세한 내용은 AWS 클라우드에 대한 Bitnami WordPress를 참조하세요.

5단계: 인스턴스의 스냅샷 생성

원하는 방식으로 웹 사이트를 구성한 후 인스턴스의 주기적 스냅샷을 생성하여 백업합니다. 스냅샷은 시스템 디스크의 복사본이며, 인스턴스의 원본 구성입니다. 스냅샷은 스냅샷을 생성한 시점부터 인스턴스를 복원하는 데 필요한 모든 데이터를 포함합니다.

스냅샷을 수동으로 생성하거나 자동 스냅샷을 활성화하여 Lightsail에서 매일 스냅샷을 생성하도록 할 수 있습니다. 인스턴스에 문제가 있는 경우 스냅샷을 사용하여 새 교체 인스턴스를 생성할 수 있습니다.

스냅샷 탭의 인스턴스 관리 페이지에서 스냅샷 작업을 할 수 있습니다. 자세한 내용은 Amazon Lightsail의 스냅샷을 참조하세요.



Lightsail

1단계: WordPress 인스턴스 구성

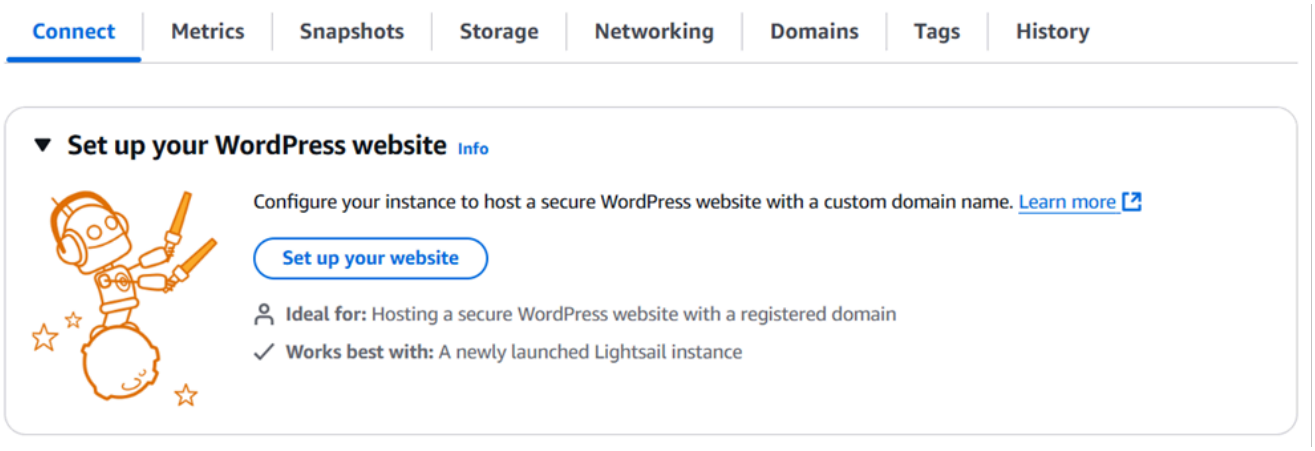
다음을 구성하는 단계별 안내 워크플로를 사용하여 WordPress 인스턴스를 구성할 수 있습니다.

- 등록된 도메인 이름 - WordPress 사이트에는 기억하기 쉬운 도메인 이름이 필요합니다. 사용자는 이 도메인 이름을 지정하여 WordPress 사이트에 액세스합니다. 자세한 내용은 도메인 및 DNS 섹션을 참조하세요.
- DNS 관리 - 도메인의 DNS 레코드를 관리하는 방법을 결정해야 합니다. DNS 레코드는 도메인이나 하위 도메인이 연결된 IP 주소나 호스트 이름을 DNS 서버에 알립니다. DNS 영역에는 도메인의 DNS 레코드가 포함됩니다. 자세한 내용은 the section called “Lightsail의 DNS” 섹션을 참조하세요.
- 고정 IP 주소 - 인스턴스를 중지했다가 시작하면 WordPress 인스턴스의 기본 퍼블릭 IP 주소가 변경됩니다. 인스턴스에 고정 IP 주소를 연결할 때 인스턴스를 중지했다가 시작한 경우에도 동일하게 유지됩니다. 자세한 내용은 the section called “IP 주소” 섹션을 참조하세요.
- SSL/TLS 인증서 - 검증된 인증서를 생성하고 인스턴스에 설치한 후 등록된 도메인을 통해 인스턴스로 라우팅되는 트래픽이 HTTPS를 사용하여 암호화되도록 WordPress 웹 사이트에 HTTPS를 활성화할 수 있습니다. 자세한 내용은 the section called “HTTPS 활성화” 섹션을 참조하세요.

> **Tip**
>
> 시작하기 전에 다음 팁을 확인하세요. 문제 해결 정보는 WordPress 설정 문제 해결을 참조하세요.
>
> - 인스턴스가 Running 상태여야 합니다. 인스턴스가 방금 시작된 경우 SSH 연결이 준비될 때까지 몇 분 정도 기다립니다.
> - 인스턴스 방화벽의 포트 22, 80 및 443은 설정이 실행되는 동안 모든 IP 주소에서 TCP 연결을 허용해야 합니다. 자세한 내용은 인스턴스 방화벽을 참조하세요.
> - Apex 도메인(`example.com`)과 `www` 하위 도메인(`www.example.com`)의 트래픽을 가리키는 DNS 레코드를 추가하거나 업데이트할 때 인터넷 전체에 전파해야 합니다. nslookup 또는 MxToolbox의 DNS 조회와 같은 도구를 사용하여 DNS 변경 사항이 적용되었는지 확인할 수 있습니다.
> - 인증서를 암호화하면 60~90일마다 자동으로 갱신됩니다.
> - 설정이 진행 중인 동안에는 인스턴스를 중지하거나 변경하지 마세요. 인스턴스를 구성하는 데 최대 15분이 걸릴 수 있습니다. 인스턴스 연결 탭에서 각 단계의 진행 상황을 볼 수 있습니다.

웹 사이트 설정 마법사를 사용하여 인스턴스를 구성

1. 인스턴스 관리 페이지의 연결 탭에서 웹 사이트 설정을 선택합니다.



2. 도메인 이름 지정에서 기존 Lightsail 관리형 도메인을 사용하거나, Lightsail에 새 도메인을 등록하거나, 다른 도메인 등록 기관을 사용하여 등록한 도메인을 사용합니다. 이 도메인 사용을 선택하여 다음 단계로 이동합니다.
3. DNS 구성에 대해 다음 작업 중 한 가지를 수행하세요.

- Lightsail DNS 영역을 사용하려면 Lightsail 관리형 도메인을 선택합니다. 다음 단계로 이동하려면 이 DNS 영역 사용을 선택합니다.
- 도메인의 DNS 레코드를 관리하는 호스팅 서비스를 사용하려면 타사 도메인을 선택합니다. 나중에 사용하기로 결정할 경우를 대비하여 Lightsail 계정에 일치하는 DNS 영역을 생성합니다. 타사 DNS 사용을 선택하여 다음 단계로 이동합니다.

4. 고정 IP 주소 생성에서 고정 IP 주소의 이름을 입력한 다음 고정 IP 생성을 선택합니다.
5. 도메인 할당 관리에서 할당 추가를 선택하고 도메인 유형을 선택한 다음 추가를 선택합니다. 계속을 선택하여 다음 단계로 이동합니다.
6. SSL/TLS 인증서 생성에서 도메인 및 하위 도메인을 선택하고 이메일 주소를 입력한 다음 Lightsail에 권한을 부여하여 인스턴스에서 Let's Encrypt 인증서를 구성하기를 선택하고 인증서 생성을 선택합니다. Lightsail 리소스 구성을 시작합니다.

   설정이 진행 중인 동안에는 인스턴스를 중지하거나 변경하지 마세요. 인스턴스를 구성하는 데 최대 15분이 걸릴 수 있습니다. 인스턴스 연결 탭에서 각 단계의 진행 상황을 볼 수 있습니다.
7. 웹 사이트 설정이 완료되면 도메인 할당 단계에서 지정한 URL이 WordPress 사이트를 여는지 확인합니다.

2단계: WordPress 웹 사이트의 기본 애플리케이션 암호 가져오기

WordPress 웹 사이트의 관리 대시보드에 로그인하려면 기본 애플리케이션 암호가 필요합니다.

WordPress 관리자 기본 암호 가져오기

1. WordPress 인스턴스의 인스턴스 관리 페이지를 엽니다.
2. WordPress 패널에서 기본 암호 검색을 선택합니다. 이렇게 하면 페이지 하단의 액세스 기본 암호가 확장됩니다.

3. CloudShell 시작을 선택합니다. 이는 페이지 하단에 분할 창을 엽니다.
4. 복사를 선택한 다음 콘텐츠를 CloudShell 창에 붙여넣습니다. 커서를 CloudShell 프롬프트에 놓고 Ctrl+V를 누르거나 마우스 오른쪽 버튼을 클릭하여 메뉴를 열고 붙여넣기를 선택할 수 있습니다.
5. CloudShell 창에 표시된 암호를 기록해 둡니다. 나중에 WordPress 웹사이트의 관리 대시보드에 로그인하는 데 사용합니다.

```
~ $ AWS_REGION=us-east-1 ~/lightsail_connect WordPress-1 tail -n 1 application_credentials
LnPLnPLnPLnPLnP9
```

3단계: WordPress 웹 사이트에 로그인

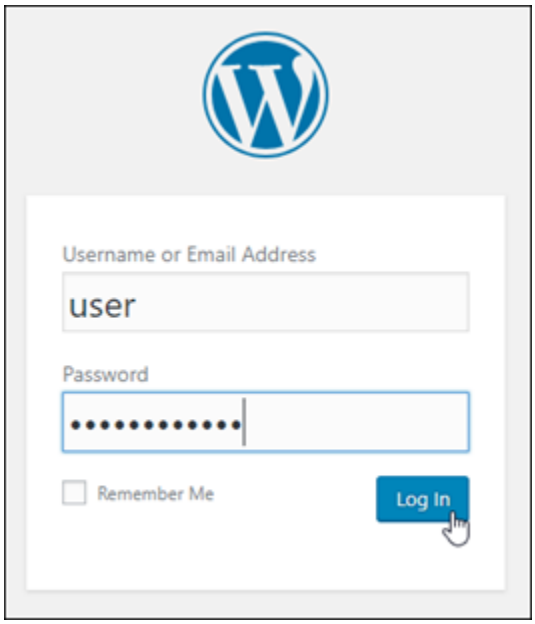
이제 기본 사용자 암호가 있으므로 WordPress 웹 사이트의 홈 페이지로 이동하여 관리 대시보드에 로그인합니다. 로그인한 후 기본 암호를 변경할 수 있습니다.

관리자 대시보드에 로그인

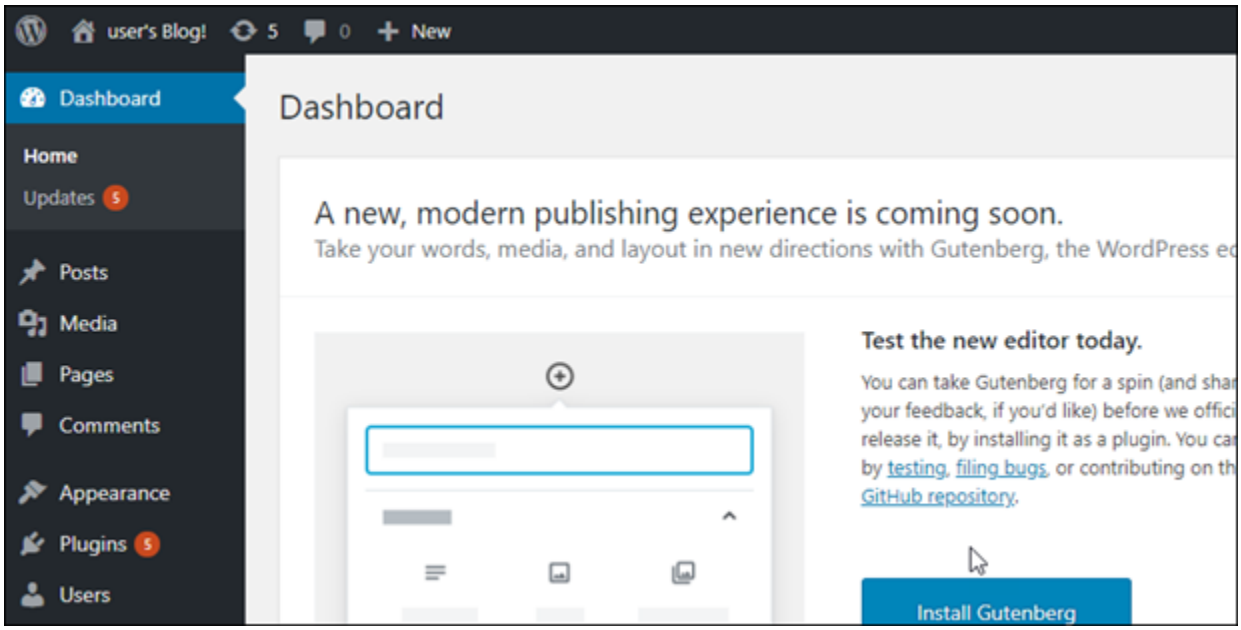
1. WordPress 인스턴스의 인스턴스 관리 페이지를 엽니다.
2. WordPress 패널에서 WordPress 관리자 액세스를 선택합니다.
3. WordPress 관리 대시보드 액세스 패널의 퍼블릭 IP 주소 사용에서 다음 형식의 링크를 선택합니다.

   http://*public-ipv4-address*./wp-admin
4. 사용자 이름 또는 이메일 주소 상자에 **user**를 입력합니다.
5. 암호에 이전 단계에서 얻은 암호를 입력합니다.

6. 그런 다음 로그인을 선택합니다.



이제 관리 작업을 수행할 수 있는 WordPress 웹사이트의 관리 대시보드에 로그인되었습니다. WordPress 웹사이트 관리에 대한 자세한 내용은 WordPress 설명서의 WordPress Codex를 참조하세요.

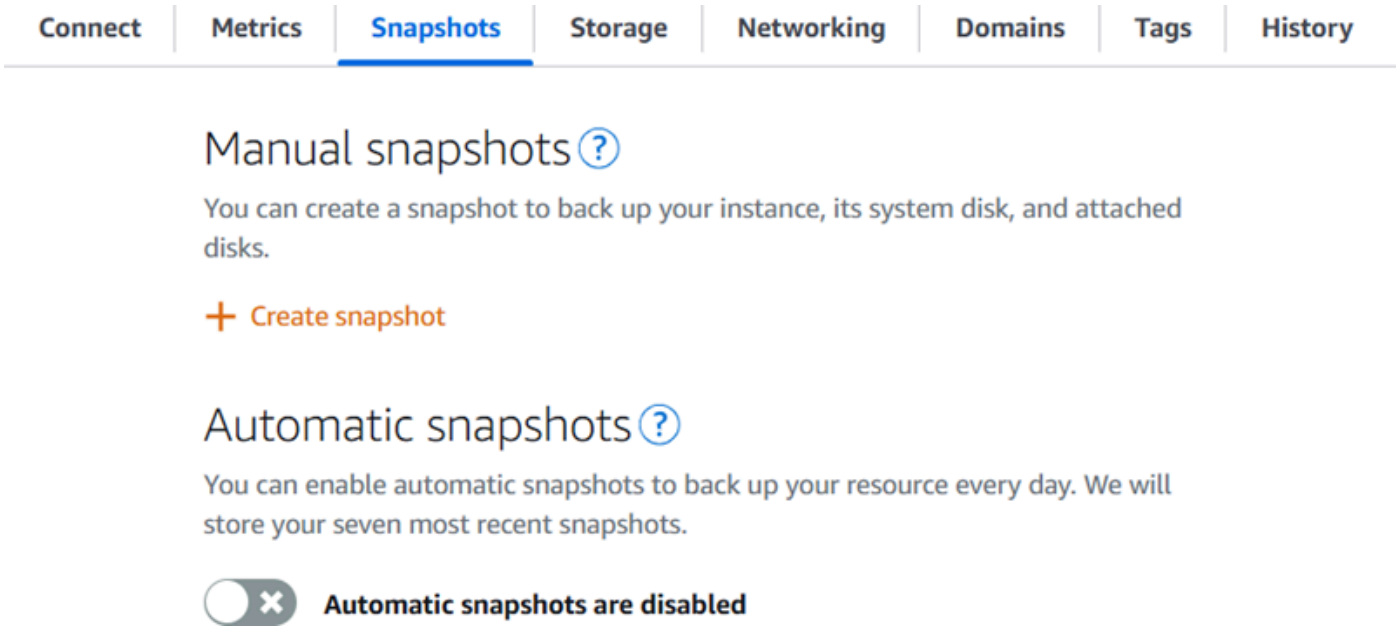


4단계: 인스턴스의 스냅샷 생성

원하는 방식으로 웹 사이트를 구성한 후 인스턴스의 주기적 스냅샷을 생성하여 백업합니다. 스냅샷은 시스템 디스크의 복사본이며, 인스턴스의 원본 구성입니다. 스냅샷은 스냅샷을 생성한 시점부터 인스턴스를 복원하는 데 필요한 모든 데이터를 포함합니다.

스냅샷을 수동으로 생성하거나 자동 스냅샷을 활성화하여 Lightsail에서 매일 스냅샷을 생성하도록 할 수 있습니다. 인스턴스에 문제가 있는 경우 스냅샷을 사용하여 새 교체 인스턴스를 생성할 수 있습니다.

스냅샷 탭의 인스턴스 관리 페이지에서 스냅샷 작업을 할 수 있습니다. 자세한 내용은 Amazon Lightsail의 스냅샷을 참조하세요.



Lightsail에서 WordPress 멀티사이트 설정

> 알고 계셨나요?
>
> Lightsail는 일별 스냅샷 7개를 저장하고 인스턴스에 대해 자동 스냅샷을 활성화하면 가장 오래된 스냅샷을 최신 스냅샷으로 자동 대체합니다. 자세한 내용은 Lightsail 인스턴스 및 디스크의 자동 스냅샷 구성을 참조하세요.

다음은 Amazon Lightsail에 WordPress Multisite 인스턴스를 설치하고 실행한 후 시작해야 할 몇 가지 단계입니다.

목차

- 1단계: Bitnami 설명서 읽기
- 2단계: WordPress 관리 대시보드에 액세스하기 위한 기본 애플리케이션 암호 가져오기
- 3단계: 인스턴스에 고정 IP 주소 연결
- 4단계: WordPress Multisite 웹 사이트의 관리 대시보드에 로그인
- 5단계: WordPress Multisite 웹 사이트로 등록된 도메인 이름의 트래픽 라우팅
- 6단계: WordPress Multisite 웹 사이트에 도메인 또는 하위 도메인으로 블로그 추가
- 7단계: WordPress Multisite 설명서 읽기 및 웹 사이트 구성 계속
- 8단계: 인스턴스의 스냅샷 생성

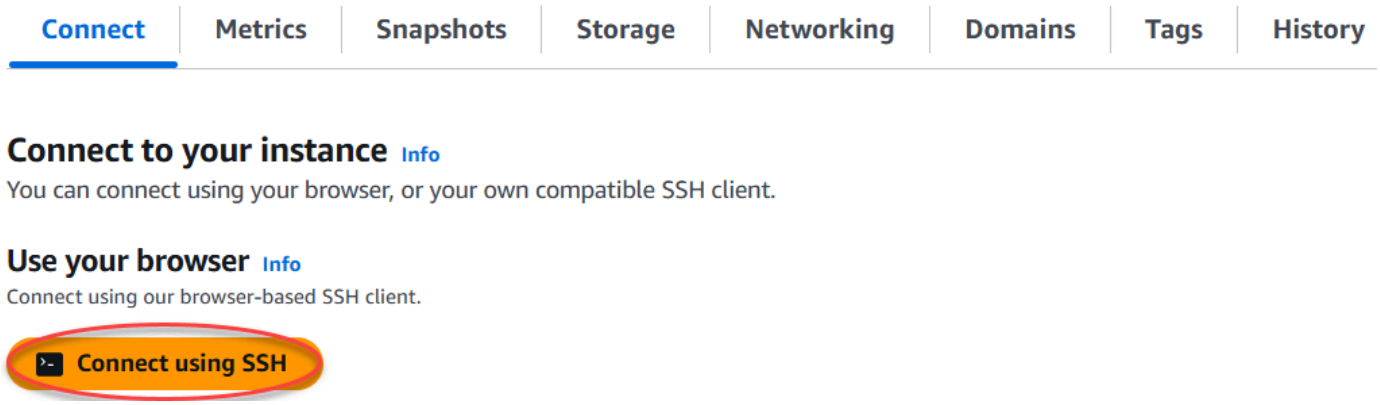
1단계: Bitnami 설명서 읽기

Bitnami 설명서를 읽고 WordPress Multisite 인스턴스를 구성하는 방법을 알아보세요. 자세한 내용은 AWS 클라우드용 Bitnami에서 패키징한 WordPress Multisite를 참조하세요.

2단계: WordPress 관리 대시보드에 액세스하기 위한 기본 애플리케이션 암호 가져오기

WordPress Multisite 웹 사이트의 관리 대시보드에 액세스하는 데 필요한 기본 애플리케이션 암호를 가져오려면 다음 절차를 완료하세요. 자세한 내용은 Amazon Lightsail의 Bitnami 인스턴스에 대한 애플리케이션 사용자 이름과 암호 가져오기를 참조하세요.

1. 인스턴스 관리 페이지의 연결(Connect) 탭에서 SSH로 연결(Connect using SSH)을 선택합니다.



2. 연결한 후 다음 명령을 입력하여 기본 애플리케이션 암호를 가져옵니다.

```
cat $HOME/bitnami_application_password
```

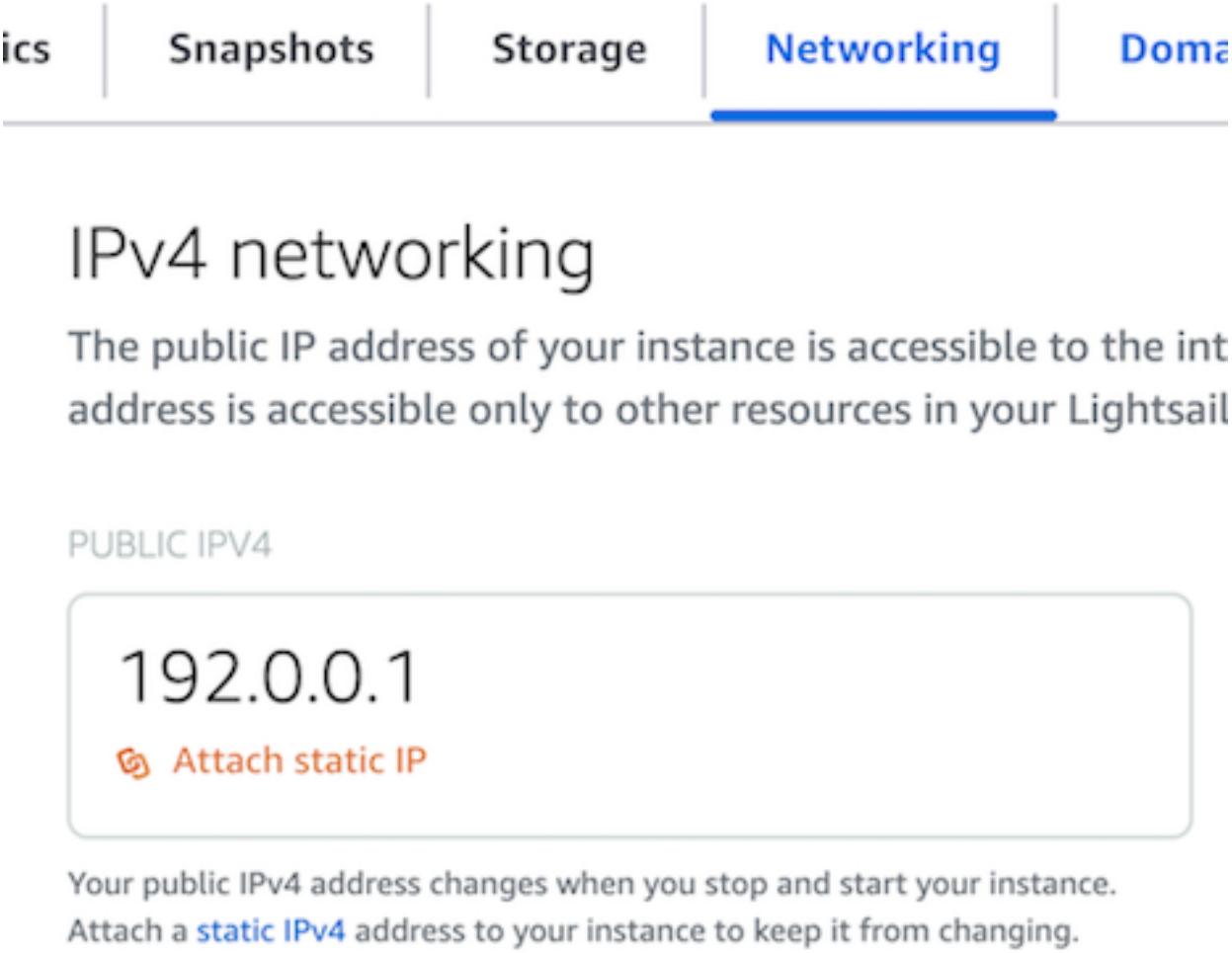
다음 예와 유사한 응답이 표시되며, 여기에 기본 애플리케이션 암호가 포함되어 있습니다. 이 암호를 사용하여 WordPress Multisite 웹 사이트의 관리 대시보드에 로그인합니다.

```
bitnami@ip-172-26-0-18:~$ cat ~/bitnami_application_password
D7aQpED8GKm.
bitnami@ip-172-26-0-18:~$
```

3단계: 인스턴스에 고정 IP 주소 연결

인스턴스에 연결된 기본 동적 퍼블릭 IP 주소는 인스턴스를 중지하고 시작할 때마다 변경됩니다. 퍼블릭 IP 주소가 변경되지 않도록 고정 IP 주소를 생성하고 인스턴스에 연결할 수 있습니다. 나중에 인스턴스에 도메인 이름을 사용하면 인스턴스를 중지하고 시작할 때마다 도메인의 DNS 레코드를 업데이트할 필요가 없습니다. 각 인스턴스에 1개의 고정 IP 주소만 연결할 수 있습니다.

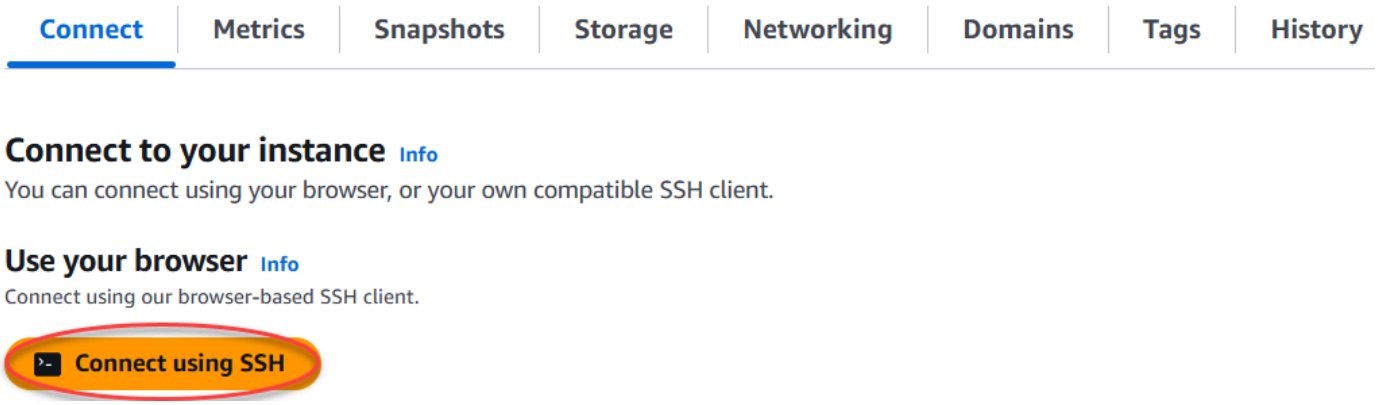
인스턴스 관리 페이지의 네트워킹 탭에서 고정 IP 생성 또는 고정 IP 연결을 선택합니다. 인스턴스에 연결할 수 있는 고정 IP를 생성해 둔 경우에는 페이지의 지침을 따릅니다. 자세한 내용은 고정 IP를 생성하여 인스턴스에 연결을 참조하세요.



새로운 고정 IP 주소가 인스턴스에 연결되면 WordPress가 새로운 고정 IP 주소를 인식하도록 다음 절차를 완료해야 합니다.

1. 인스턴스의 새 고정 IP 주소를 기록해 둡니다. 고정 IP 주소는 인스턴스 관리 페이지의 머리말 섹션에 나와 있습니다.

2. 인스턴스 관리 페이지의 연결(Connect) 탭에서 SSH를 사용하여 연결(Connect using SSH)을 선택합니다.



3. 연결한 후 다음 명령을 입력합니다. *<StaticIP>*를 인스턴스의 새 고정 IP 주소로 바꿉니다.

```
sudo /opt/bitnami/configure_app_domain --domain <StaticIP>
```

예:

```
sudo /opt/bitnami/configure_app_domain --domain 203.0.113.0
```

다음 예와 비슷한 응답이 나타나는 것을 볼 수 있습니다. 인스턴스의 WordPress 웹 사이트에서 이제 새 고정 IP 주소를 인식합니다.

```
bitnami@ip-[redacted]:~$ sudo /opt/bitnami/configure_app_domain --domain 203.0.113.0
Configuring domain to 203.0.113.0
2021-03-12T15:49:22.000Z - info: Saving configuration info to disk
prestashop 15:49:22.41 INFO  ==> Trying to connect to the database server
prestashop 15:49:22.44 INFO  ==> Updating hostname in database
prestashop 15:49:22.46 INFO  ==> Purging cache
Disabling automatic domain update for IP address changes
```

> **Note**
>
> 해당 명령이 실패하면 이전 WordPress Multisite 인스턴스 버전을 사용하고 있는 것일 수 있습니다. 대신 다음 명령을 실행해 봅니다. *<StaticIP>*를 인스턴스의 새 고정 IP 주소로 바꿉니다.
>
> ```
> cd /opt/bitnami/apps/wordpress
> sudo ./bnconfig --machine_hostname <StaticIP>
> ```
>
> 해당 명령을 실행하고 나서 다음 명령을 입력하여 서버가 재시작될 때마다 bnconfig 도구가 자동으로 실행되지 않도록 합니다.
>
> ```
> sudo mv bnconfig bnconfig.disabled
> ```

4단계: WordPress Multisite 웹 사이트의 관리 대시보드에 로그인

이제 기본 애플리케이션 암호가 있으므로 다음 절차를 완료하여 WordPress Multisite 웹 사이트의 홈 페이지로 이동하고 관리 대시보드에 로그인합니다. 로그인한 후 웹 사이트 사용자 지정 및 관리 변경을 시작할 수 있습니다. WordPress에서 수행할 수 있는 작업에 대한 자세한 내용을 알아보려면 이 가이드 뒷부분의 7단계: WordPress Multisite 설명서 읽기 및 웹 사이트 구성 계속 섹션을 참조하세요.

1. 인스턴스 관리 페이지의 연결(Connect) 탭에서 인스턴스의 퍼블릭 IP 주소를 기록해 둡니다. 퍼블릭 IP 주소는 인스턴스 관리 페이지의 헤더 섹션에도 표시됩니다.

2. 인스턴스의 퍼블릭 IP 주소로 이동합니다(예: `http://203.0.113.0`으로 이동).

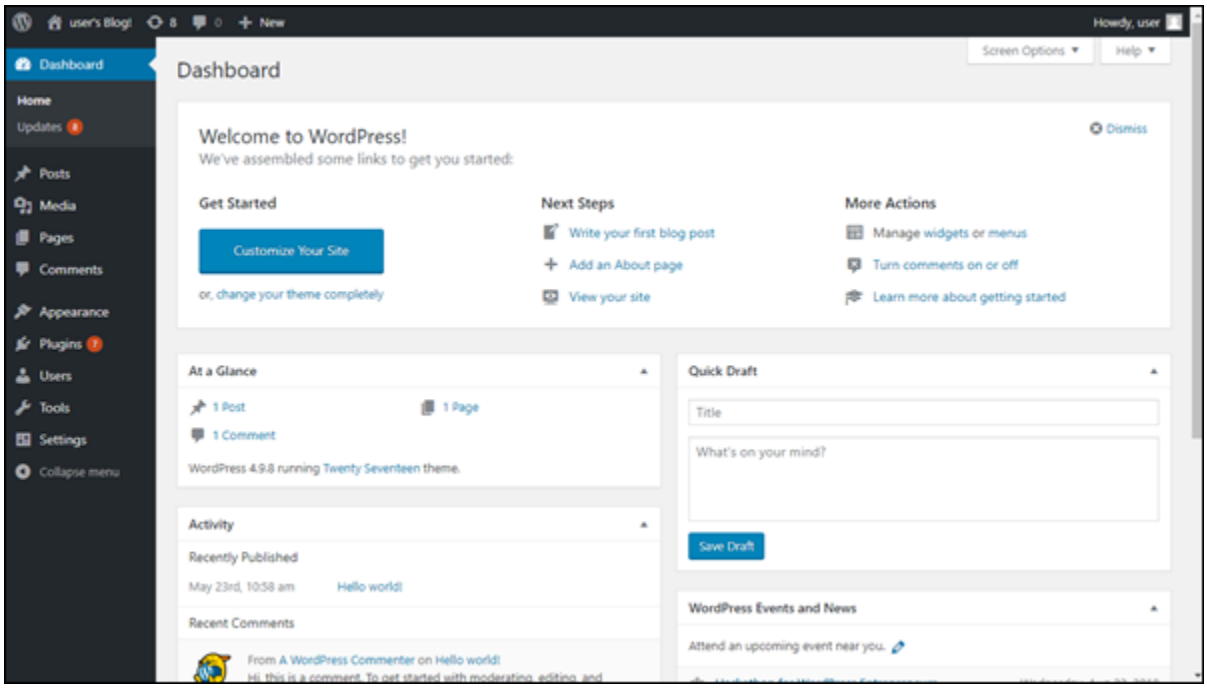
   WordPress 웹 사이트의 홈 페이지가 나타납니다.

3. WordPress 웹 사이트 홈 페이지의 오른쪽 하단 모서리에 있는 관리를 선택합니다.

   관리(Manage) 배너가 표시되지 않는 경우 `http://`*<PublicIP>*`/wp-login.php`을 통해 로그인 페이지로 이동할 수 있습니다. *<PublicIP>*을(를) 인스턴스의 퍼블릭 IP 주소로 바꿉니다.

4. 이 가이드의 초반부에서 검색한 기본 사용자 이름(`user`) 및 기본 암호를 사용하여 로그인합니다.

WordPress 관리 대시보드가 나타납니다.



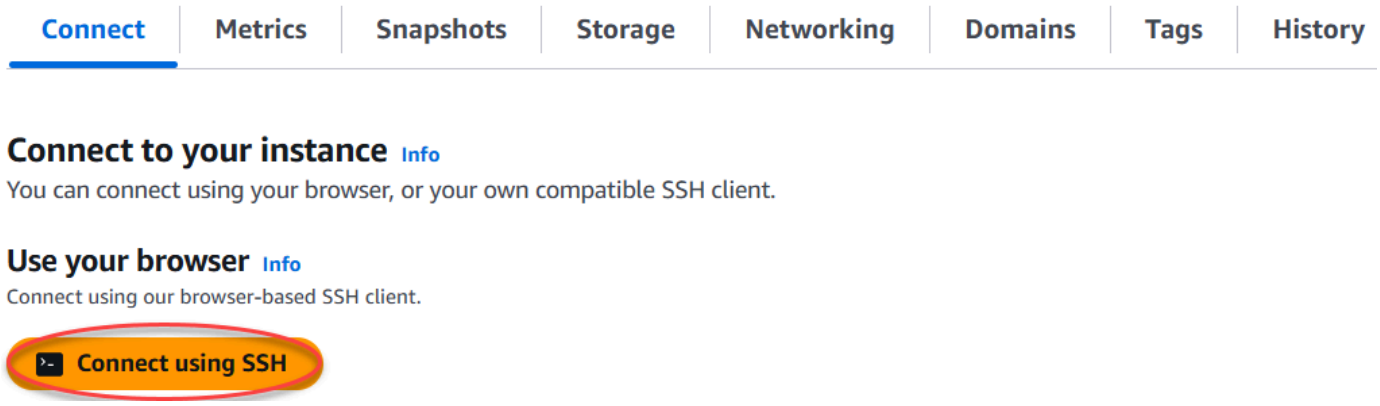
5단계: WordPress Multisite 웹 사이트로 등록된 도메인 이름의 트래픽 라우팅

등록된 도메인 이름(예: `example.com`)의 트래픽을 WordPress Multisite 웹 사이트로 라우팅하려면 도메인의 DNS에 레코드를 추가하면 됩니다. DNS 레코드는 일반적으로 도메인을 등록한 등록 대행자가 관리 및 호스팅합니다. 그러나 Lightsail 콘솔을 사용하여 관리할 수 있도록 도메인의 DNS 레코드 관리를 Lightsail로 이전하는 것이 좋습니다.

Lightsail 콘솔 홈 페이지의 도메인 및 DNS(Domains & DNS) 탭에서 DNS 영역 생성(Create DNS zone)을 선택하고 페이지의 지침에 따릅니다. 자세한 내용은 DNS 영역을 생성하여 Lightsail에서 도메인의 DNS 레코드 관리를 참조하세요.

도메인 이름이 트래픽을 인스턴스로 라우팅한 후 WordPress가 도메인 이름을 인식하도록 하려면 다음 절차를 완료해야 합니다.

1. 인스턴스 관리 페이지의 연결(Connect) 탭에서 SSH를 사용하여 연결(Connect using SSH)을 선택합니다.



2. 연결한 후 다음 명령을 입력합니다. *<DomainName>*을 인스턴스로 트래픽을 라우팅할 도메인 이름으로 바꿉니다.

```
sudo /opt/bitnami/configure_app_domain --domain <DomainName>
```

예:

```
sudo /opt/bitnami/configure_app_domain --domain www.example.com
```

다음 예와 비슷한 응답이 나타나는 것을 볼 수 있습니다. 이제 WordPress Multisite 소프트웨어가 도메인 이름을 인식합니다.

```
bitnami@ip-[redacted]:~$ sudo /opt/bitnami/configure_app_domain --domain www.example.com
Configuring domain to www.example.com
2021-03-12T15:49:22.000Z - info: Saving configuration info to disk
prestashop 15:49:22.41 INFO  ==> Trying to connect to the database server
prestashop 15:49:22.44 INFO  ==> Updating hostname in database
prestashop 15:49:22.46 INFO  ==> Purging cache
Disabling automatic domain update for IP address changes
```

> **Note**
>
> 해당 명령이 실패하면 이전 WordPress Multisite 인스턴스 버전을 사용하고 있는 것일 수 있습니다. 대신 다음 명령을 실행해 봅니다. *<DomainName>*을 인스턴스로 트래픽을 라우팅할 도메인 이름으로 바꿉니다.
>
> ```
> cd /opt/bitnami/apps/wordpress
> sudo ./bnconfig --machine_hostname <DomainName>
> ```

해당 명령을 실행하고 나서 다음 명령을 입력하여 서버가 재시작될 때마다 bnconfig 도구가 자동으로 실행되지 않도록 합니다.

```
sudo mv bnconfig bnconfig.disabled
```

인스턴스에 대해 구성한 도메인 이름으로 이동하면 WordPress Multisite 웹 사이트의 기본 블로그로 리디렉션됩니다. 다음으로 WordPress Multisite 웹 사이트에 블로그를 도메인으로 추가할지 하위 도메인으로 추가할지 결정해야 합니다. 자세한 내용을 알아보려면 이 가이드의 6단계: WordPress Multisite 웹 사이트에 도메인 또는 하위 도메인으로 블로그 추가 섹션을 참조하세요.

6단계: WordPress Multisite 웹 사이트에 도메인 또는 하위 도메인으로 블로그 추가

WordPress Multisite는 WordPress의 한 인스턴스에서 여러 블로그 웹 사이트를 호스팅하도록 설계되었습니다. WordPress Multisite에 새 블로그 웹 사이트를 추가할 때 자체 도메인 또는 WordPress Multisite 기본 도메인의 하위 도메인을 사용하도록 구성할 수 있습니다. 이러한 옵션 중 하나만 사용하도록 WordPress Multisite를 구성할 수 있습니다. 예를 들어 블로그 사이트를 도메인으로 추가하기로 선택한 경우 블로그 사이트를 하위 도메인으로 추가할 수 없으며 그 반대의 경우도 마찬가지입니다. 이러한 옵션 중 하나를 구성하려면 다음 가이드 중 하나를 참조하세요.

- `example1.com` 및 `example2.com`과 같은 블로그 사이트를 도메인으로 추가하려면 Lightsail의 WordPress Multisite 인스턴스에 도메인으로 블로그 추가(Add blogs as domains to your WordPress Multisite instance)를 참조하세요.
- `one.example.com` 및 `two.example.com`과 같은 WordPress Multisite 기본 도메인의 하위 도메인으로 블로그 사이트를 추가하려면 Lightsail의 WordPress Multisite 인스턴스에 하위 도메인으로 블로그 추가(Add blogs as subdomains to your WordPress Multisite instance)를 참조하세요.

7단계: WordPress Multisite 설명서 읽기 및 웹 사이트 구성 계속

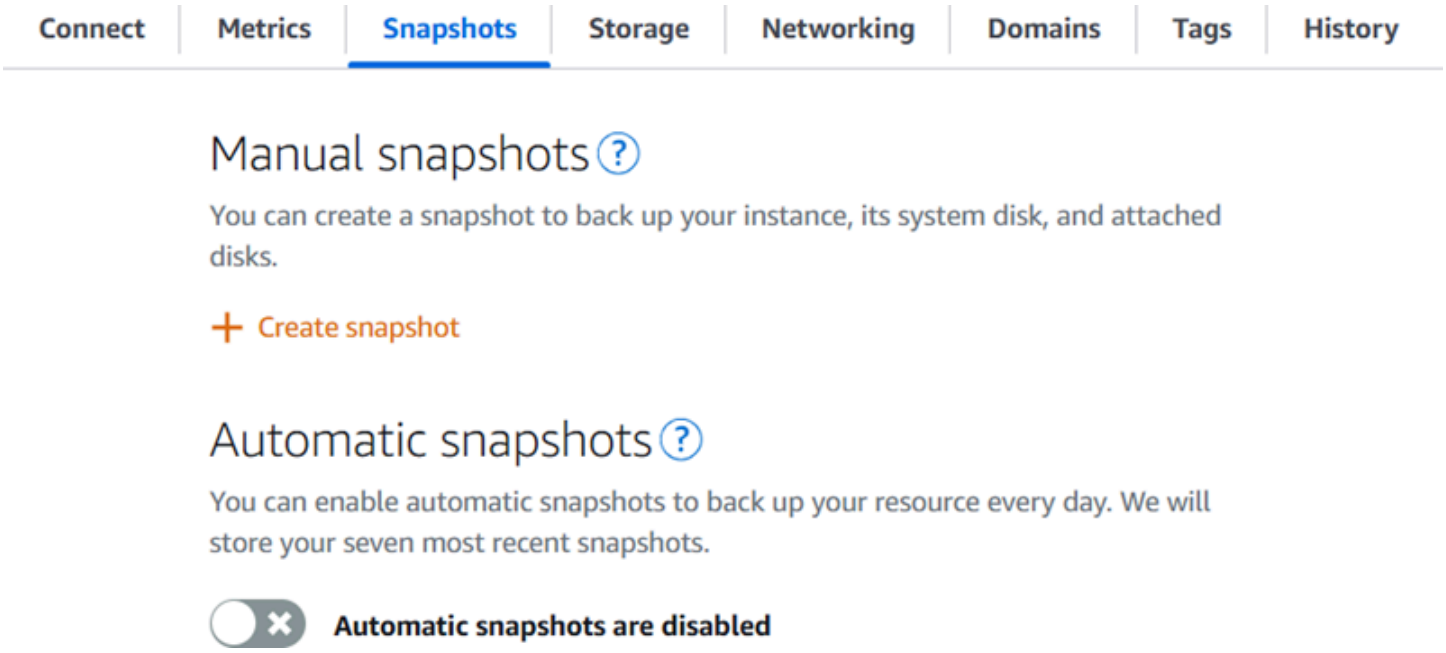
WordPress Multisite 설명서를 읽고 웹 사이트를 관리하고 사용자 지정하는 방법을 알아보세요. 자세한 내용을 알아보려면 WordPress Multisite Network Administration Documentation(WordPress Multisite 네트워크 관리 설명서)를 참조하세요.

8단계: 인스턴스의 스냅샷 생성

원하는 방식으로 웹 사이트를 구성한 후 인스턴스의 주기적 스냅샷을 생성하여 백업합니다. 스냅샷은 시스템 디스크의 복사본이며, 인스턴스의 원본 구성입니다. 스냅샷은 스냅샷을 생성한 시점부터 인스턴스를 복원하는 데 필요한 모든 데이터를 포함합니다.

스냅샷을 수동으로 생성하거나 자동 스냅샷을 활성화하여 Lightsail에서 매일 스냅샷을 생성하도록 할 수 있습니다. 인스턴스에 문제가 있는 경우 스냅샷을 사용하여 새 교체 인스턴스를 생성할 수 있습니다.

스냅샷 탭의 인스턴스 관리 페이지에서 스냅샷 작업을 할 수 있습니다. 자세한 내용은 Amazon Lightsail의 스냅샷을 참조하세요.



Lightsail에서 Bitnami 애플리케이션 및 스택 작업

이 섹션에서는 Amazon Lightsail 인스턴스의 Bitnami 애플리케이션과 관련된 다음 주제를 다룹니다.

주제

- Lightsail Bitnami 인스턴스에 대한 기본 애플리케이션 사용자 이름 및 암호 가져오기
- Lightsail 인스턴스에서 Bitnami 배너 제거

Lightsail Bitnami 인스턴스에 대한 기본 애플리케이션 사용자 이름 및 암호 가져오기

Bitnami는 가상 프라이빗 서버인 Amazon Lightsail 인스턴스로 만들 수 있는 여러 애플리케이션 인스턴스 이미지 또는 블루프린트를 제공합니다. 이러한 블루프린트는 Lightsail 콘솔의 인스턴스 생성 페이지에서 “Bitnami에서 패키징”으로 표시됩니다.

Bitnami 블루프린트를 사용하여 인스턴스를 생성했으면 인스턴스에 로그인하여 관리합니다. 그러려면 인스턴스에서 실행되는 애플리케이션 및/또는 데이터베이스의 기본 사용자 이름과 암호를 가져와야 합니다. 이 문서에서는 다음 블루프린트로 생성된 Lightsail 인스턴스에 로그인 및 관리하기 위해 필요한 정보를 가져오는 방법을 보여줍니다.

- WordPress 블로깅 및 콘텐츠 관리 애플리케이션
- WordPress Multisite 블로깅 및 콘텐츠 관리 애플리케이션, 같은 인스턴스에서 여러 웹 사이트 지원
- Django 개발 스택
- Ghost 블로깅 및 콘텐츠 관리 애플리케이션
- LAMP 개발 스택(PHP 7)
- Node.js 개발 스택
- Joomla 콘텐츠 관리 애플리케이션
- Magento 전자 상거래 애플리케이션
- MEAN 개발 스택
- Drupal 콘텐츠 관리 애플리케이션
- GitLab CE 리포지토리 애플리케이션
- Redmine 프로젝트 관리 애플리케이션
- Nginx(LEMP) 개발 스택

기본 Bitnami 애플리케이션 및 데이터베이스 사용자 이름 가져오기

다음은 Bitnami 블루프린트를 사용하여 생성된 Lightsail 인스턴스의 기본 애플리케이션 및 데이터베이스 사용자 이름입니다.

> **Note**
>
> 모든 Bitnami 블루프린트에 애플리케이션이나 데이터베이스가 포함되지는 않습니다. 블루프린트에 애플리케이션이나 데이터베이스가 포함되지 않으면 사용자 이름이 해당 사항 없음(N/A)으로 표시됩니다.

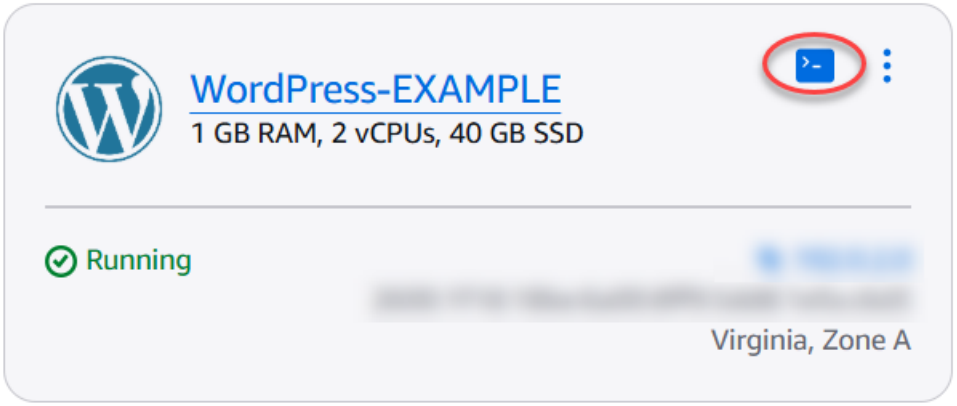
애플리케이션 이름	애플리케이션 사용자 이름	데이터베이스 사용자 이름
WordPress, WordPress Multisite 포함	user	Root
PrestaShop	user@example.com을 선택합니다.	Root
Django	해당 사항 없음	Root
Ghost	user@example.com을 선택합니다.	Root
LAMP 스택(PHP 5 및 PHP 7)	해당 사항 없음	Root
Node.js	해당 사항 없음	해당 사항 없음
Joomla	user	Root
Magento	user	Root
MEAN	해당 사항 없음	Root
Drupal	user	Root
GitLab CE	user	postgres
Redmine	user	Root
Nginx	해당 사항 없음	Root

기본 Bitnami 애플리케이션 및 데이터베이스 암호 가져오기

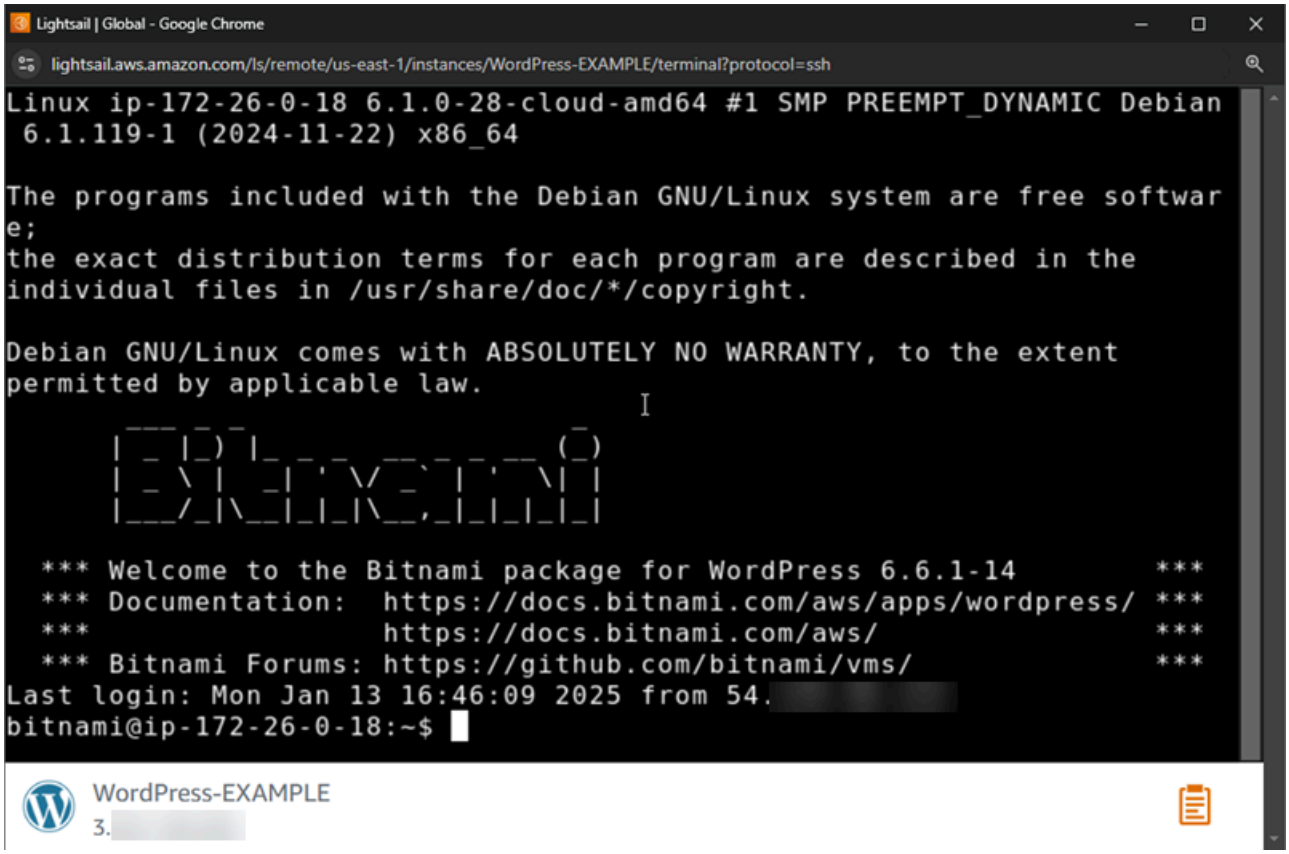
기본 애플리케이션 및 데이터베이스 암호는 인스턴스에 저장됩니다. Lightsail 콘솔에서 브라우저 기반 SSH 터미널을 사용하고 특수한 명령을 실행하여 인스턴스에 연결해 암호를 검색합니다.

기본 Bitnami 애플리케이션 및 데이터베이스 암호를 가져오려면

1. Lightsail 콘솔에 로그인합니다.
2. 아직 만들지 않았다면 Bitnami 블루프린트를 사용하여 인스턴스를 만듭니다. 자세한 내용은 Amazon Lightsail VPS 생성을 참조하세요.
3. Lightsail 홈 페이지에서 연결할 인스턴스의 빠른 연결 아이콘을 선택합니다.



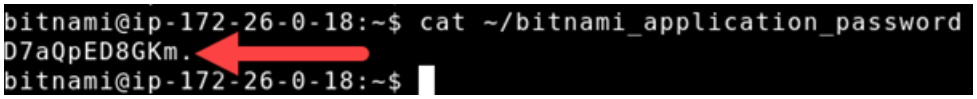
다음 예와 같이 브라우저 기반 SSH 클라이언트 창이 열립니다.



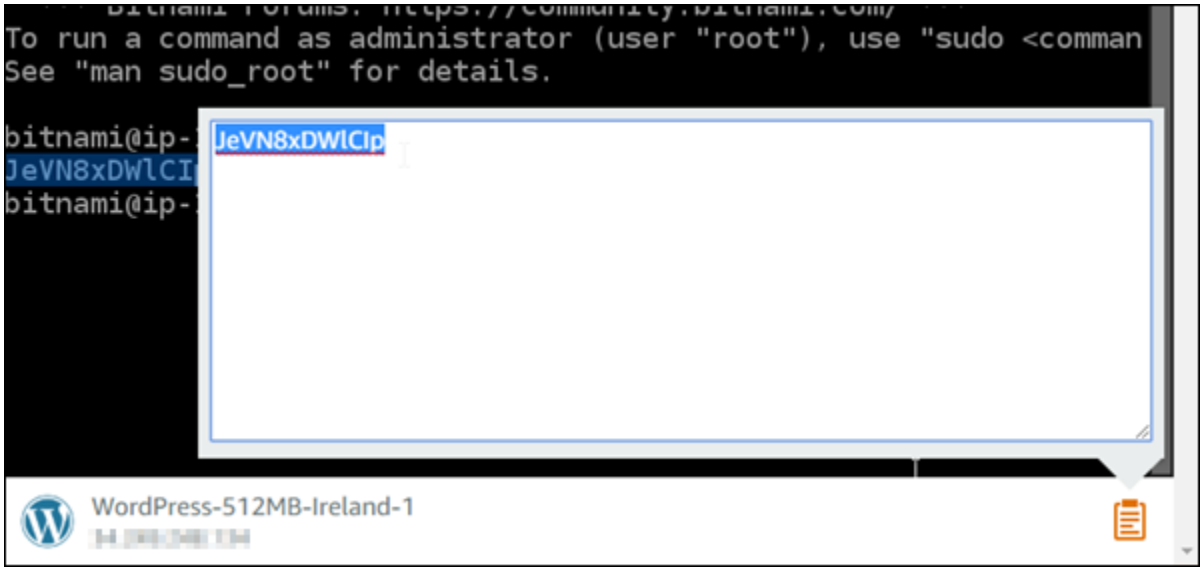
4. 다음 명령을 입력하여 기본 애플리케이션 암호를 검색합니다.

```
cat ~/bitnami_application_password
```

애플리케이션 암호가 포함된 응답이 다음과 유사하게 표시되어야 합니다.



5. 터미널 화면에서 암호를 강조 표시하고 브라우저 기반 SSH 클라이언트 창 오른쪽 아래 모서리에서 클립보드 아이콘을 선택합니다.
6. 클립보드 텍스트 상자에서 복사할 텍스트를 강조 표시한 후 Ctrl+C 또는 Cmd+C를 눌러 로컬 클립보드에 텍스트를 복사합니다.



> **Important**
>
> 이때 암호를 어딘가에 저장해야 합니다. 나중에 인스턴스에서 Bitnami 애플리케이션에 로그인한 후 변경할 수 있습니다.

인스턴스의 Bitnami 애플리케이션에 로그인

WordPress, Joomla, Magento, Drupal, GitLab CE 및 Redmine 블루프린트로 만든 인스턴스에서는 인스턴스의 퍼블릭 IP 주소를 탐색하여 애플리케이션에 로그인합니다.

Bitnami 애플리케이션에 로그인하려면

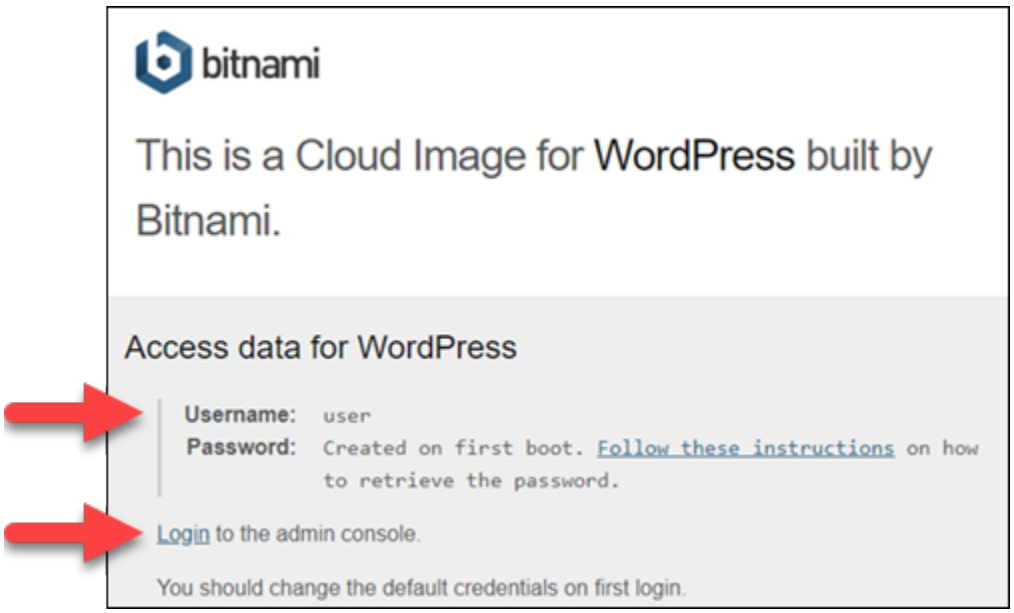
1. 브라우저 창에서 인스턴스의 퍼블릭 IP 주소로 이동합니다.

   Bitnami 애플리케이션 홈 페이지가 열립니다. 인스턴스에 대해 선택한 Bitnami 블루프린트에 따라 홈 페이지가 표시됩니다. 예를 들어 다음은 WordPress 애플리케이션 홈 페이지입니다.

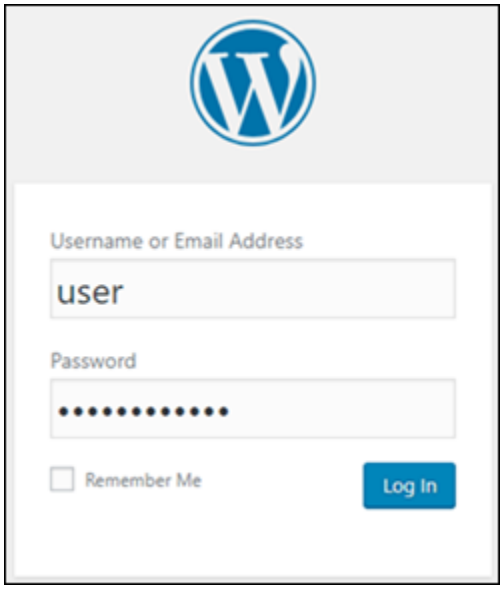
2. 애플리케이션 홈 페이지 오른쪽 아래 모서리의 Bitnami 로고를 선택하여 애플리케이션 정보 페이지로 이동합니다.

> **Note**
>
> GitLab CE 애플리케이션에는 Bitnami 로고가 표시되지 않습니다. 대신 GitLab CE 홈 페이지에서 사용자 이름 및 암호 텍스트 필드를 사용하여 로그인하세요.

애플리케이션 정보 페이지에는 기본 사용자 이름과 인스턴스의 애플리케이션에 맞는 로그인 페이지 링크가 있습니다.



3. 페이지의 로그인 링크를 선택하여 인스턴스의 애플리케이션에 맞는 로그인 페이지로 이동합니다.
4. 좀 전에 얻은 사용자 이름과 암호를 입력한 후 로그인을 선택합니다.



다음 단계

다음 링크를 사용하여 Bitnami 블루프린트에 대해 자세히 알아보고 자습서를 살펴보세요. 예를 들어 WordPress 인스턴스에 대한 플러그인 설치 또는 SSL 인증서를 이용한 HTTPS 지원 활성화를 수행할 수 있습니다.

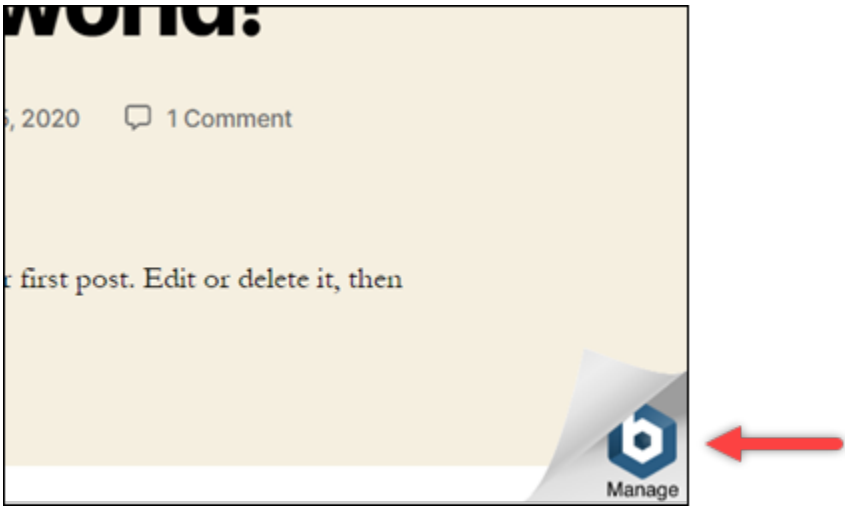
- Amazon Web Services용 Bitnami WordPress
- Amazon Web Services용 Bitnami LAMP 스택

- Amazon Web Services용 Bitnami Node.js
- Amazon Web Services용 Bitnami Joomla
- Amazon Web Services용 Bitnami Magento
- Amazon Web Services용 Bitnami MEAN 스택
- Amazon Web Services용 Bitnami Drupal
- Amazon Web Services용 Bitnami GitLab
- Amazon Web Services용 Bitnami Redmine
- Amazon Web Services용 Bitnami Nginx(LEMP 스택)

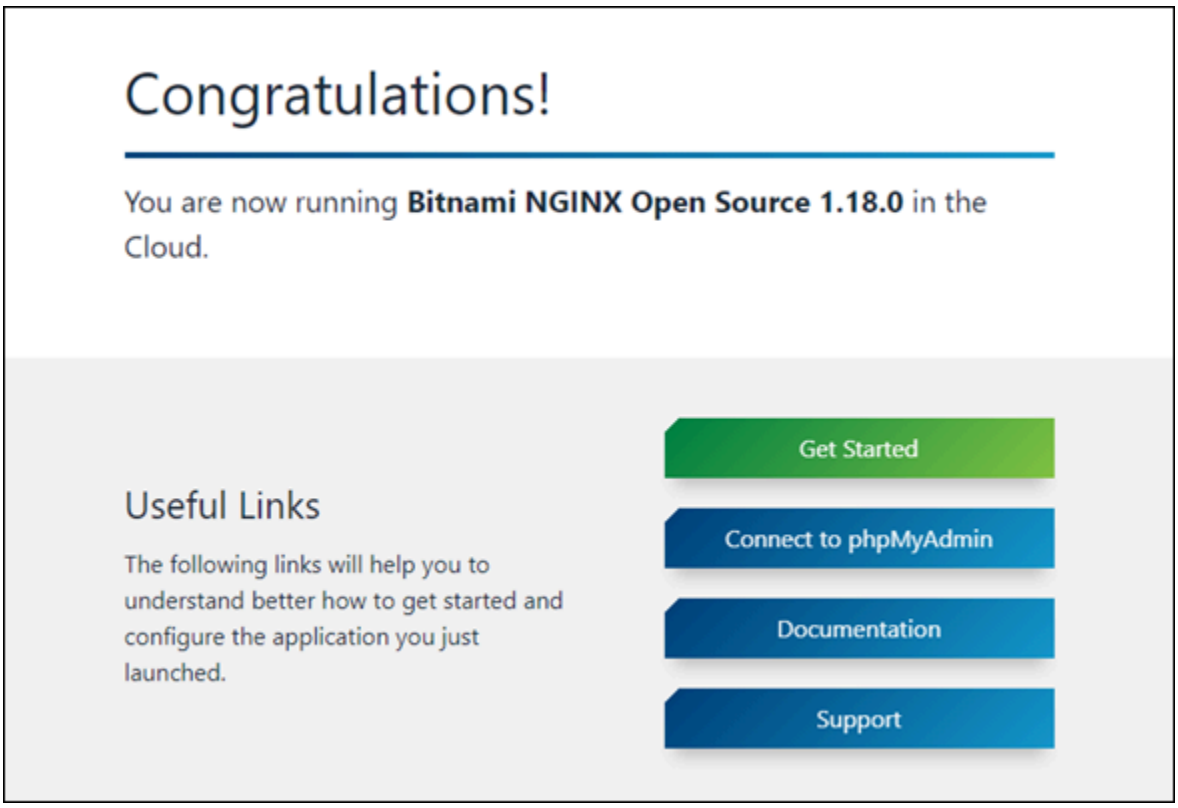
자세한 내용은 Get Started with Bitnami Applications using Amazon Lightsail 또는 Amazon Lightsail 사용 FAQ를 참조하세요.

Lightsail 인스턴스에서 Bitnami 배너 제거

Amazon Lightsail 인스턴스에 대해 선택할 수 있는 Bitnami 블루프린트 중 일부는 애플리케이션의 홈 페이지에 Bitnami 배너를 표시합니다. 다음 예에서 'Bitnami 인증' WordPress 인스턴스의 Bitnami 배너는 홈 페이지의 오른쪽 하단 모서리에 표시됩니다. 이 가이드에서는 인스턴스의 애플리케이션 홈 페이지에서 Bitnami 아이콘을 영구적으로 제거하는 방법을 안내합니다.



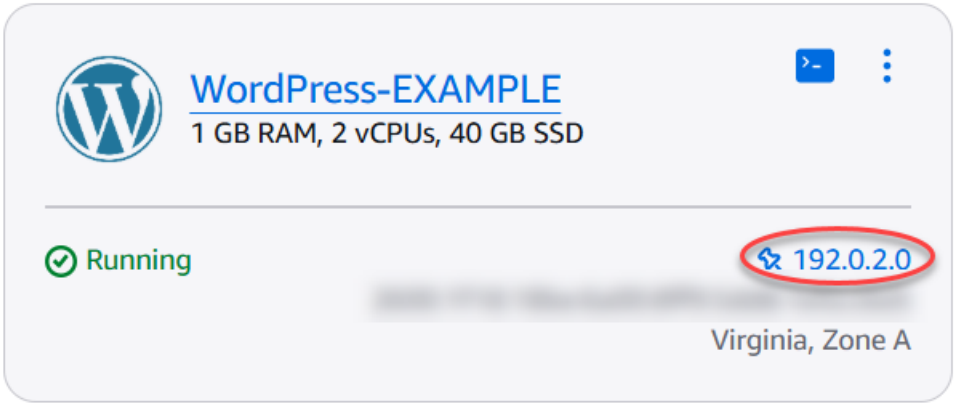
일부 Bitnami 블루프린트 애플리케이션은 애플리케이션 홈 페이지에 Bitnami 배너를 표시합니다. Bitnami 배너가 표시되는지 확인하려면 Lightsail 인스턴스의 홈 페이지를 방문하세요. 다음 예에서 "Bitnami에서 패키징"한 Nginx 인스턴스의 경우 Bitnami 아이콘이 표시되지 않습니다. 대신 자리 표시자 정보 페이지가 표시되며, 이 페이지가 인스턴스에 배포하도록 선택한 애플리케이션으로 대체됩니다. 인스턴스에 Bitnami 배너가 나타나지 않으면 이 가이드의 절차를 따르지 않아도 됩니다.



인스턴스에서 Bitnami 배너 제거

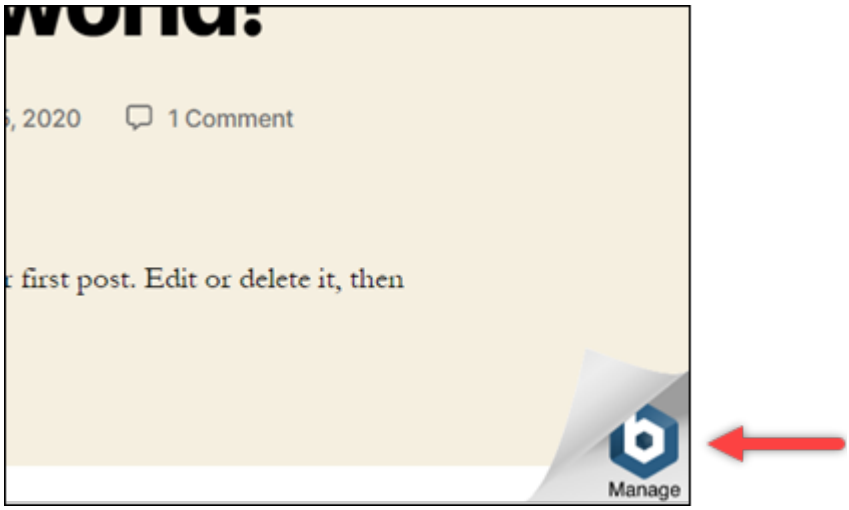
인스턴스의 애플리케이션 홈 페이지에 Bitnami 아이콘이 표시되는지 확인하고 제거하려면 다음 절차를 완료하세요.

1. Lightsail 콘솔에 로그인합니다.
2. Lightsail 홈 페이지의 인스턴스 섹션에서 확인할 인스턴스의 퍼블릭 IP 주소를 복사합니다.



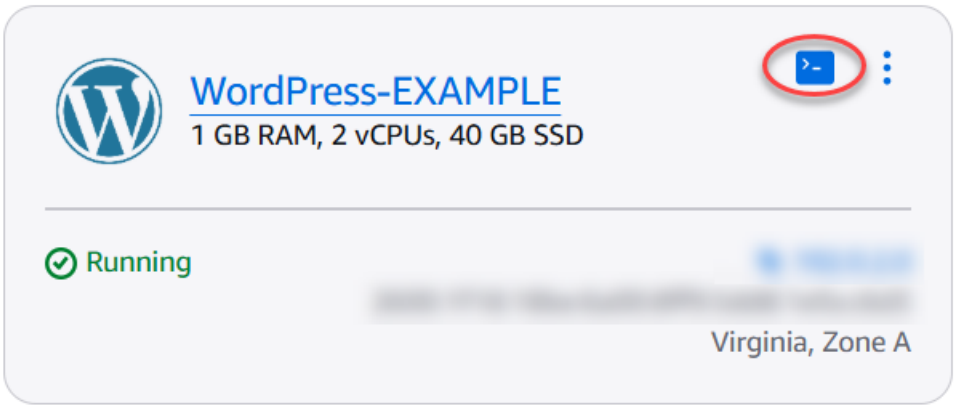
3. 새 브라우저 탭을 열고 주소 표시줄에 인스턴스의 퍼블릭 IP 주소를 입력한 다음 Enter 키를 누릅니다.
4. 다음 옵션 중 하나를 확인합니다.

1. 페이지에 Bitnami 아이콘이 표시되지 않으면 다음 절차를 수행하지 마세요. 애플리케이션 홈 페이지에서 Bitnami 아이콘을 제거할 필요가 없습니다.
2. 다음 예와 같이 페이지의 오른쪽 하단 모서리에 Bitnami 아이콘이 표시되면 다음 단계를 계속 진행하여 Bitnami를 제거합니다.



다음 단계에서는 Lightsail 브라우저 기반 SSH 클라이언트를 사용하여 인스턴스에 연결합니다. 연결한 후에는 Bitnami Configuration Tool(bnconfig) 도구를 실행하여 애플리케이션 홈 페이지에서 Bitnami 아이콘을 제거합니다. bnconfig 도구는 Bitnami 블루프린트 인스턴스의 애플리케이션을 구성할 수 있도록 지원하는 명령줄 도구입니다. 자세한 내용은 Bitnami 문서의 Bitnami Configuration Tool 정보 알아보기를 참조하세요.

5. Lightsail 홈 페이지에 있는 브라우저 탭으로 돌아갑니다.
6. 연결할 인스턴스 이름 옆에 있는 브라우저 기반 SSH 클라이언트 아이콘을 선택합니다.



7. SSH 클라이언트를 인스턴스에 연결한 후 다음 명령 중 하나를 입력합니다.

1. 인스턴스에서 Apache를 사용하는 경우 다음 명령 중 하나를 입력합니다. 명령 중 하나가 실패하는 경우 다른 명령을 시도해 보세요. 이 명령의 첫 번째 부분은 Bitnami 배너를 비활성화하고 두 번째 부분은 Apache 서비스를 다시 시작합니다.

```
sudo /opt/bitnami/apps/wordpress/bnconfig --disable_banner 1 && sudo /opt/
bitnami/ctlscript.sh restart apache
```

```
sudo /opt/bitnami/wordpress/bnconfig --disable_banner 1 && sudo /opt/bitnami/
ctlscript.sh restart apache
```

   인스턴스의 퍼블릭 IP 주소에 방문했을 때 Bitnami 아이콘이 사라졌으면 프로세스가 성공적으로 수행된 것입니다.

단계별 지침에 따라 Bitnami 애플리케이션 및 데이터베이스의 기본 자격 증명을 검색하고, 애플리케이션의 관리자 패널에 로그인하고, 선택적으로 애플리케이션의 홈 페이지에서 Bitnami 브랜딩 배너를 제거하는 방법을 알아봅니다.

이 안내서에서는 WordPress, Joomla, Drupal, Ghost, LAMP, LEMP, MEAN, Node.js 등을 포함하여 Lightsail에서 사용할 수 있는 다양한 Bitnami 블루프린트를 다룹니다. 애플리케이션과 데이터베이스의 기본 사용자 이름과 기본 암호를 안전하게 가져오는 명령을 제공합니다. 이 가이드를 따르면 Lightsail 인스턴스에서 실행되는 Bitnami 애플리케이션에 쉽게 액세스하고 관리하여 요구 사항에 따라 사용자 지정하고 원치 않는 브랜딩 요소를 제거할 수 있습니다.

Lightsail WordPress 인스턴스 구성 및 관리

이 안내서에서는 Lightsail의 WordPress 인스턴스와 관련된 다음 주제를 다룹니다.

주제

- Lightsail에서 WordPress 인스턴스 시작 및 구성
- WP Offload Media를 사용하여 Lightsail의 WordPress 웹 사이트를 Amazon S3에 연결
- Lightsail WordPress 인스턴스를 Amazon Aurora 데이터베이스에 연결
- Lightsail에서 MySQL 관리형 데이터에 WordPress 데이터베이스 전송
- 정적 콘텐츠를 위해 WordPress 인스턴스를 Lightsail 버킷에 연결
- Lightsail 콘텐츠 전송 네트워크를 사용하여 WordPress 구성

- Lightsail의 WordPress 인스턴스에서 이메일 활성화
- Lightsail에서 HTTPS로 WordPress 사이트 보호
- WordPress 블로그를 Lightsail로 마이그레이션

Lightsail에서 WordPress 인스턴스 시작 및 구성

Amazon Lightsail은 Amazon Web Services(AWS)를 가장 쉽게 시작할 수 있는 방법입니다. Lightsail에는 인스턴스(가상 프라이빗 서버), 관리형 데이터베이스, SSD 기반 스토리지, 백업(스냅샷), 데이터 전송, 도메인 DNS 관리, 정적 IP 및 로드 밸런서 등 프로젝트를 신속하게 시작하는 데 필요한 모든 것을 예측 가능한 저렴한 가격에 포함합니다.

이 자습서에서는 Lightsail에서 WordPress 인스턴스를 시작하고 구성하는 방법을 알아봅니다. 여기에는 사용자 지정 도메인 이름을 구성, HTTPS로 인터넷 트래픽 보호, SSH를 사용하여 인스턴스에 연결, WordPress 웹 사이트에 로그인하는 단계가 포함됩니다. 이 자습서를 마치면 Lightsail에서 인스턴스를 실행하기 위한 기본 지식을 얻게 됩니다.

> **Note**
>
> AWS 프리 티어의 일부로 일부 인스턴스 번들에서 Amazon Lightsail를 무료로 시작할 수 있습니다. 자세한 내용은 Amazon Lightsail 요금 페이지의 AWS 프리 티어를 참조하세요.

사전 조건

1단계:에 가입 AWS

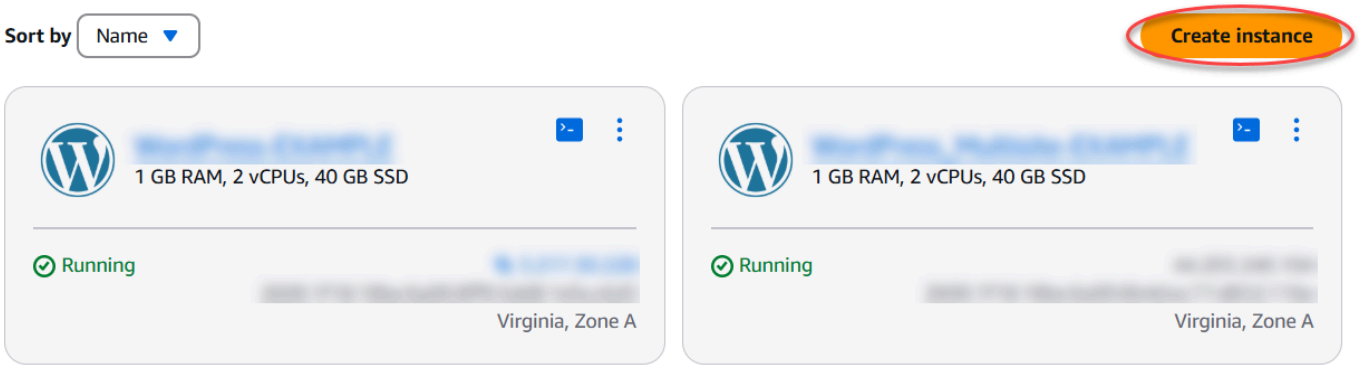
Amazon Lightsail 에는가 필요합니다 AWS 계정. 계정이 이미 있는 경우 에 가입 AWS하거나 에 로그인 AWS합니다.

2단계: WordPress 인스턴스 생성

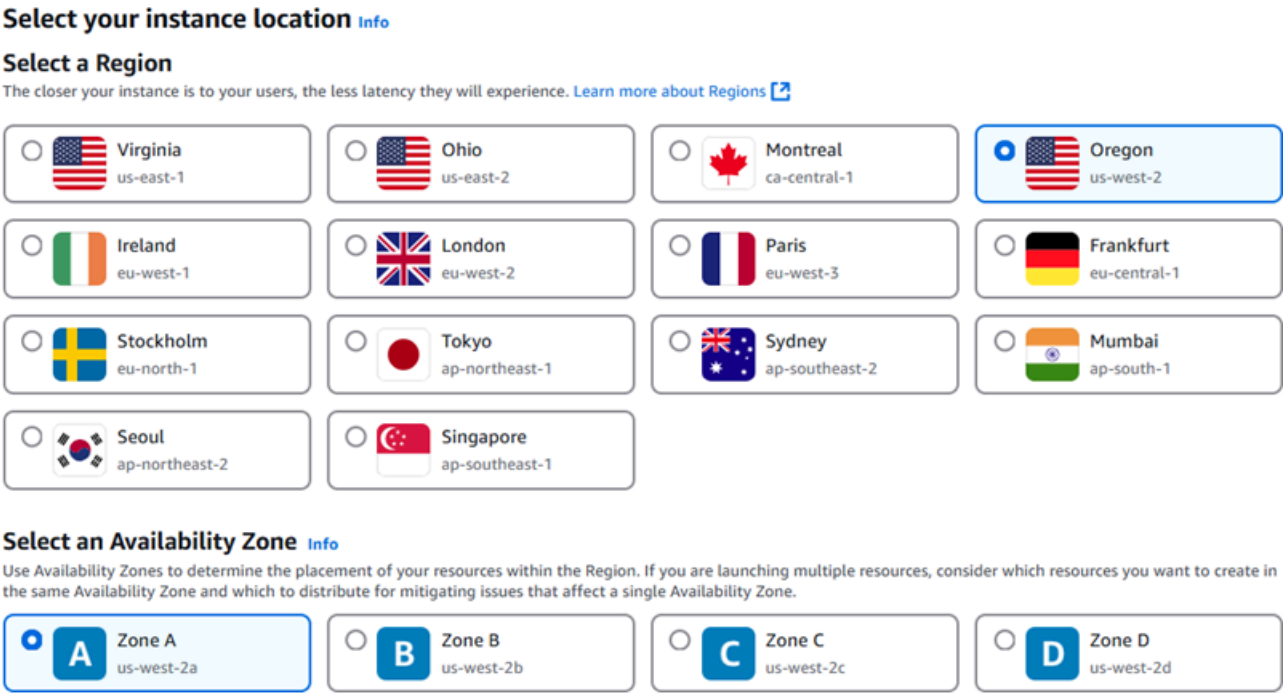
WordPress 인스턴스를 바로 시작하려면 다음 단계를 완료합니다. 자세한 내용은 the section called “인스턴스 생성” 단원을 참조하십시오.

WordPress용 Lightsail 인스턴스 생성

1. Lightsail 콘솔에 로그인합니다.
2. Lightsail 홈 페이지의 인스턴스(Instances) 섹션에서 인스턴스 생성(Create instance)을 선택합니다.



3. 인스턴스의 AWS 리전 및 가용 영역을 선택합니다.



4. 다음과 같이 인스턴스의 이미지를 선택합니다.

   a. 플랫폼 선택에서 Linux/Unix를 선택합니다.

   b. WordPress 블루프린트를 선택합니다.

      Amazon Lightsail 또는 Bitnami 공급자를 선택할 수 있습니다.

5. 인스턴스 플랜을 선택합니다.

   플랜에는 예측 가능하고 저렴한 비용의 머신 구성(RAM, SSD, vCPU)과 데이터 전송 허용량이 포함됩니다.

6. 인스턴스 이름을 입력합니다. 리소스 이름:

   - Lightsail 계정 AWS 리전 의 각 내에서 고유해야 합니다.

- 2~255자의 문자로 구성되어야 합니다.
- 영숫자 문자 또는 숫자로 시작하고 끝나야 합니다.
- 영숫자 문자, 숫자, 마침표, 대시, 밑줄이 포함될 수 있습니다.

7. 인스턴스 생성을 선택합니다.
8. 테스트 블로그 게시물을 보려면 인스턴스 관리 페이지로 이동하여 페이지 오른쪽 상단에 표시된 퍼블릭 IPv4 주소를 복사합니다. 주소를 인터넷에 연결된 웹 브라우저의 주소 필드에 붙여넣습니다. 브라우저에 테스트 블로그 게시물이 표시됩니다.

인스턴스가 실행된 후 인스턴스 관리 페이지에서 WordPress 블루프린트 공급업체를 식별한 다음 아래에서 적절한 가이드를 선택합니다.

Bitnami

3단계: WordPress 인스턴스 구성

단계별 안내 워크플로를 사용하여 WordPress 인스턴스를 구성하거나 개별 작업을 완료할 수 있습니다. 두 옵션 중 하나를 사용하여 다음을 구성합니다.

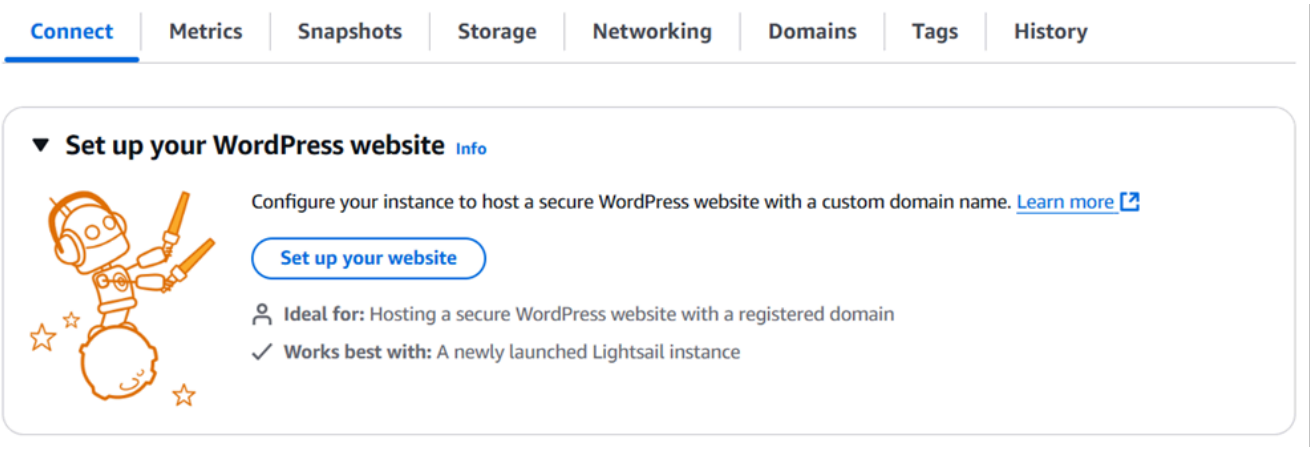
- 등록된 도메인 이름 - WordPress 사이트에는 기억하기 쉬운 도메인 이름이 필요합니다. 사용자는 이 도메인 이름을 지정하여 WordPress 사이트에 액세스합니다. 자세한 내용은 도메인 및 DNS 섹션을 참조하세요.
- DNS 관리 - 도메인의 DNS 레코드를 관리하는 방법을 결정해야 합니다. DNS 레코드는 도메인이나 하위 도메인이 연결된 IP 주소나 호스트 이름을 DNS 서버에 알립니다. DNS 영역에는 도메인의 DNS 레코드가 포함됩니다. 자세한 내용은 the section called “Lightsail의 DNS” 섹션을 참조하세요.
- 고정 IP 주소 - 인스턴스를 중지했다가 시작하면 WordPress 인스턴스의 기본 퍼블릭 IP 주소가 변경됩니다. 인스턴스에 고정 IP 주소를 연결할 때 인스턴스를 중지했다가 시작한 경우에도 동일하게 유지됩니다. 자세한 내용은 the section called “IP 주소” 섹션을 참조하세요.
- SSL/TLS 인증서 - 검증된 인증서를 생성하고 인스턴스에 설치한 후 등록된 도메인을 통해 인스턴스로 라우팅되는 트래픽이 HTTPS를 사용하여 암호화되도록 WordPress 웹 사이트에 HTTPS를 활성화할 수 있습니다. 자세한 내용은 the section called “HTTPS 활성화” 섹션을 참조하세요.

옵션: 안내형 워크플로

> **Tip**
>
> 시작하기 전에 다음 팁을 확인하세요. 문제 해결 정보는 WordPress 설정 문제 해결을 참조하세요.
>
> - 설치는 2023년 1월 1일 이후에 생성된 WordPress 버전 6 이상의 Lightsail 인스턴스를 지원합니다.
> - 설정 중에 실행되는 Certbot 종속성 파일, HTTPS 재작성 스크립트 및 인증서 갱신 스크립트는 인스턴스의 `/opt/bitnami/lightsail/scripts/` 디렉터리에 저장됩니다.
> - 인스턴스가 Running 상태여야 합니다. 인스턴스가 방금 시작된 경우 SSH 연결이 준비될 때까지 몇 분 정도 기다립니다.
> - 인스턴스 방화벽의 포트 22, 80 및 443은 설정이 실행되는 동안 모든 IP 주소에서 TCP 연결을 허용해야 합니다. 자세한 내용은 인스턴스 방화벽을 참조하세요.
> - Apex 도메인(`example.com`)과 `www` 하위 도메인(`www.example.com`)의 트래픽을 가리키는 DNS 레코드를 추가하거나 업데이트할 때 인터넷 전체에 전파해야 합니다. nslookup 또는 MxToolbox의 DNS 조회와 같은 도구를 사용하여 DNS 변경 사항이 적용되었는지 확인할 수 있습니다.
> - 2023년 1월 1일 이전에 생성된 Wordpress 인스턴스에는 더 이상 사용되지 않는 Certbot Personal Package Archive(PPA) 리포지토리가 포함되어 있어 웹 사이트 설정이 실패할 수 있습니다. 설정 중에 이 리포지토리가 있는 경우 기존 경로에서 제거되고 인스턴스의 `~/opt/bitnami/lightsail/repo.backup` 위치에 백업됩니다. 더 이상 사용되지 않는 PPA에 대한 자세한 내용은 Canonical 웹 사이트의 Certbot PPA를 참조하세요.
> - 인증서를 암호화하면 60~90일마다 자동으로 갱신됩니다.
> - 설정이 진행 중인 동안에는 인스턴스를 중지하거나 변경하지 마세요. 인스턴스를 구성하는 데 최대 15분이 걸릴 수 있습니다. 인스턴스 연결 탭에서 각 단계의 진행 상황을 볼 수 있습니다.

웹 사이트 설정 마법사를 사용하여 인스턴스를 구성

1. 인스턴스 관리 페이지의 연결 탭에서 웹 사이트 설정을 선택합니다.



2. 도메인 이름 지정에서 기존 Lightsail 관리형 도메인을 사용하거나, Lightsail에 새 도메인을 등록하거나, 다른 도메인 등록 기관을 사용하여 등록한 도메인을 사용합니다. 이 도메인 사용을 선택하여 다음 단계로 이동합니다.
3. DNS 구성에 대해 다음 작업 중 한 가지를 수행하세요.
   - Lightsail DNS 영역을 사용하려면 Lightsail 관리형 도메인을 선택합니다. 다음 단계로 이동하려면 이 DNS 영역 사용을 선택합니다.
   - 도메인의 DNS 레코드를 관리하는 호스팅 서비스를 사용하려면 타사 도메인을 선택합니다. 나중에 사용하기로 결정할 경우를 대비하여 Lightsail 계정에 일치하는 DNS 영역을 생성합니다. 타사 DNS 사용을 선택하여 다음 단계로 이동합니다.
4. 고정 IP 주소 생성에서 고정 IP 주소의 이름을 입력한 다음 고정 IP 생성을 선택합니다.
5. 도메인 할당 관리에서 할당 추가를 선택하고 도메인 유형을 선택한 다음 추가를 선택합니다. 계속을 선택하여 다음 단계로 이동합니다.
6. SSL/TLS 인증서 생성에서 도메인 및 하위 도메인을 선택하고 이메일 주소를 입력한 다음 Lightsail에 권한을 부여하여 인스턴스에서 Let's Encrypt 인증서를 구성하기를 선택하고 인증서 생성을 선택합니다. Lightsail 리소스 구성을 시작합니다.

   설정이 진행 중인 동안에는 인스턴스를 중지하거나 변경하지 마세요. 인스턴스를 구성하는 데 최대 15분이 걸릴 수 있습니다. 인스턴스 연결 탭에서 각 단계의 진행 상황을 볼 수 있습니다.
7. 웹 사이트 설정이 완료되면 도메인 할당 단계에서 지정한 URL이 WordPress 사이트를 여는지 확인합니다.

옵션: 개별 작업

개별 작업을 완료하여 인스턴스를 구성

1. 고정 IP 주소 생성

   인스턴스 관리 페이지의 네트워킹 탭에서 고정 IP 생성을 선택합니다. 고정 IP 위치 및 인스턴스가 선택됩니다. 고정 IP 주소의 이름을 지정한 다음 생성 및 연결을 선택합니다.

2. DNS 영역 생성

   탐색 창에서 도메인 및 DNS를 선택합니다. DNS 영역 생성을 선택하고 도메인을 입력한 다음 DNS 영역 생성을 선택합니다. 웹 트래픽이 현재 도메인으로 라우팅되는 경우 도메인의 현재 DNS 호스팅 공급자에서 이름 서버를 변경하기 전에 모든 기존 DNS 레코드가 Lightsail DNS 영역에 있는지 확인합니다. 이런 식으로 Lightsail DNS 영역으로 이전된 후 트래픽이 중단되지 않고 계속 흐릅니다.

3. 도메인 할당 관리

   DNS 영역 페이지에서 할당 탭에서 할당 추가를 선택합니다. 도메인 또는 하위 도메인을 선택하고 인스턴스를 선택한 다음 정적 IP 주소를 연결한 다음 할당을 선택합니다.

   > **Tip**
   >
   > 도메인이 WordPress 인스턴스로 트래픽을 라우팅하기 전에 인터넷으로 이 변경 사항이 전파될 때까지 기다립니다.

4. SSL/TLS 인증서 생성 및 설치

   단계별 지침은 the section called “HTTPS 활성화” 섹션을 참조하세요.

5. 도메인 할당 단계에서 지정한 URL이 WordPress 사이트를 여는지 확인합니다.

4단계: WordPress 웹 사이트에 대한 관리자 암호 가져오기

WordPress 웹사이트의 관리 대시보드에 로그인하기 위한 기본 암호가 인스턴스에 저장됩니다. 암호를 가져오려면 다음 단계를 완료합니다.

WordPress 관리자의 기본 암호 가져오기

1. WordPress 인스턴스의 인스턴스 관리 페이지를 엽니다.

2. WordPress 패널에서 기본 암호 검색을 선택합니다. 이렇게 하면 페이지 하단의 액세스 기본 암호가 확장됩니다.

3. CloudShell 시작을 선택합니다. 이는 페이지 하단에 분할 창을 엽니다.
4. 복사를 선택한 다음 콘텐츠를 CloudShell 창에 붙여넣습니다. 커서를 CloudShell 프롬프트에 놓고 Ctrl+V를 누르거나 마우스 오른쪽 버튼을 클릭하여 메뉴를 열고 붙여넣기를 선택할 수 있습니다.
5. CloudShell 창에 표시된 암호를 기록해 둡니다. 나중에 WordPress 웹사이트의 관리 대시보드에 로그인하는 데 사용합니다.

```
[cloudshell-user@ip-[redacted] ~]$ AWS_REGION=us-east-1 ~/lightsail_connect WordPress-1 cat bitnami_applic
ation_password
JKzhKzhKzhR!
```

5단계: WordPress 웹사이트의 관리 대시보드 로그인

이제 WordPress 웹사이트의 관리 대시보드에 대한 암호가 있으므로 로그인할 수 있습니다. 관리 대시보드에서 사용자 암호를 변경하고, 플러그인을 설치하고, 웹사이트의 테마를 변경하는 등의 작업을 할 수 있습니다.

WordPress 웹사이트의 관리 대시보드에 로그인하려면 다음 단계를 완료합니다.

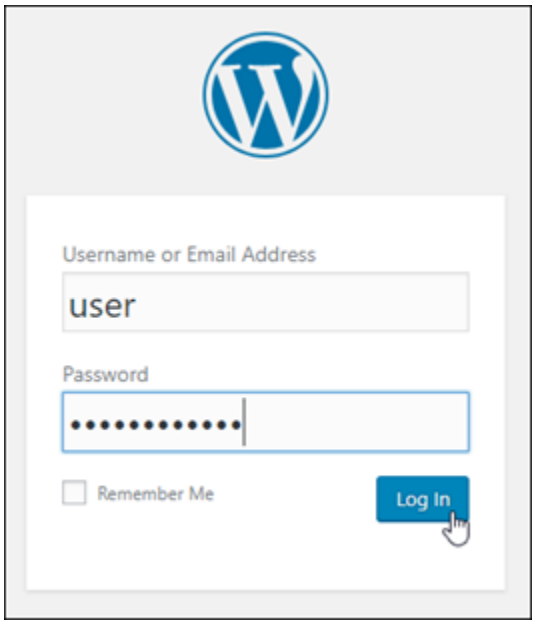
관리자 대시보드에 로그인

1. WordPress 인스턴스의 인스턴스 관리 페이지를 엽니다.
2. WordPress 패널에서 WordPress 관리자 액세스를 선택합니다.

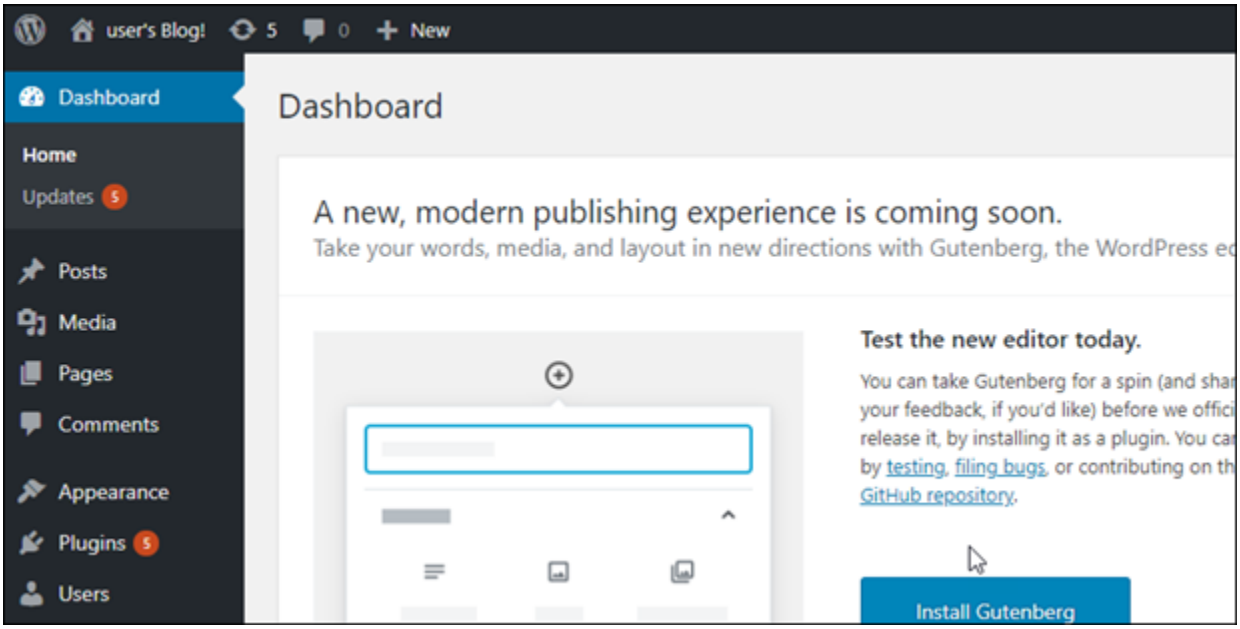
3. WordPress 관리 대시보드 액세스 패널의 퍼블릭 IP 주소 사용에서 다음 형식의 링크를 선택합니다.

   http://*public-ipv4-address*./wp-admin

4. 사용자 이름 또는 이메일 주소 상자에 **user**를 입력합니다.
5. 암호에 이전 단계에서 얻은 암호를 입력합니다.
6. 그런 다음 로그인을 선택합니다.



이제 관리 작업을 수행할 수 있는 WordPress 웹사이트의 관리 대시보드에 로그인되었습니다. WordPress 웹사이트 관리에 대한 자세한 내용은 WordPress 설명서의 WordPress Codex를 참조하세요.



Lightsail

3단계: WordPress 인스턴스 구성

단계별 안내 워크플로를 사용하여 WordPress 인스턴스를 구성하거나 개별 작업을 완료할 수 있습니다. 두 옵션 중 하나를 사용하여 다음을 구성합니다.

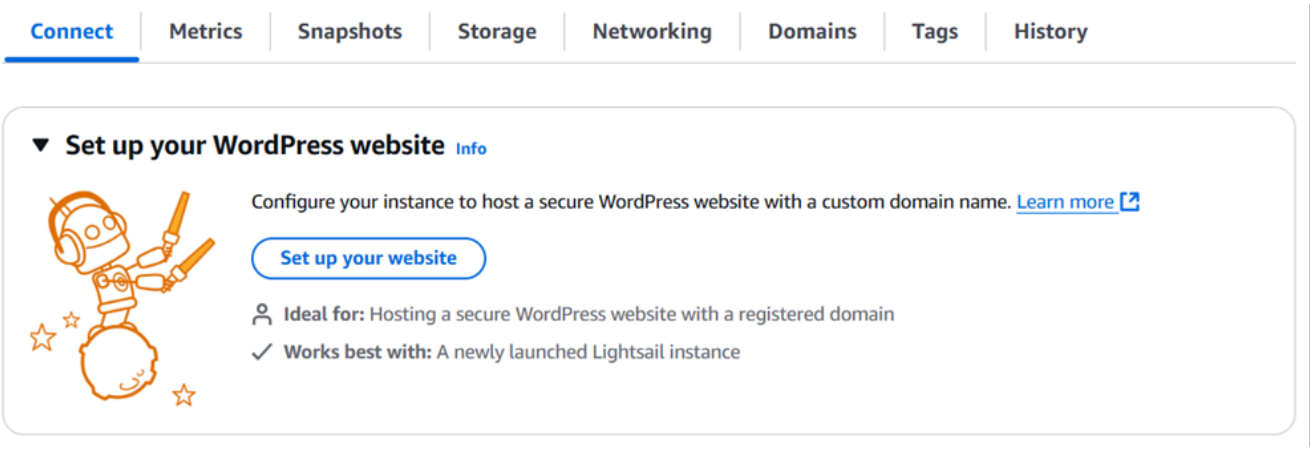
- 등록된 도메인 이름 - WordPress 사이트에는 기억하기 쉬운 도메인 이름이 필요합니다. 사용자는 이 도메인 이름을 지정하여 WordPress 사이트에 액세스합니다. 자세한 내용은 도메인 및 DNS 섹션을 참조하세요.
- DNS 관리 - 도메인의 DNS 레코드를 관리하는 방법을 결정해야 합니다. DNS 레코드는 도메인이나 하위 도메인이 연결된 IP 주소나 호스트 이름을 DNS 서버에 알립니다. DNS 영역에는 도메인의 DNS 레코드가 포함됩니다. 자세한 내용은 the section called “Lightsail의 DNS” 섹션을 참조하세요.
- 고정 IP 주소 - 인스턴스를 중지했다가 시작하면 WordPress 인스턴스의 기본 퍼블릭 IP 주소가 변경됩니다. 인스턴스에 고정 IP 주소를 연결할 때 인스턴스를 중지했다가 시작한 경우에도 동일하게 유지됩니다. 자세한 내용은 the section called “IP 주소” 섹션을 참조하세요.
- SSL/TLS 인증서 - 검증된 인증서를 생성하고 인스턴스에 설치한 후 등록된 도메인을 통해 인스턴스로 라우팅되는 트래픽이 HTTPS를 사용하여 암호화되도록 WordPress 웹 사이트에 HTTPS를 활성화할 수 있습니다. 자세한 내용은 the section called “HTTPS 활성화” 섹션을 참조하세요.

옵션: 안내형 워크플로

> **Tip**
>
> 시작하기 전에 다음 팁을 확인하세요. 문제 해결 정보는 WordPress 설정 문제 해결을 참조하세요.
>
> - 인스턴스가 Running 상태여야 합니다. 인스턴스가 방금 시작된 경우 SSH 연결이 준비될 때까지 몇 분 정도 기다립니다.
> - 인스턴스 방화벽의 포트 22, 80 및 443은 설정이 실행되는 동안 모든 IP 주소에서 TCP 연결을 허용해야 합니다. 자세한 내용은 인스턴스 방화벽을 참조하세요.
> - Apex 도메인(`example.com`)과 `www` 하위 도메인(`www.example.com`)의 트래픽을 가리키는 DNS 레코드를 추가하거나 업데이트할 때 인터넷 전체에 전파해야 합니다. nslookup 또는 MxToolbox의 DNS 조회와 같은 도구를 사용하여 DNS 변경 사항이 적용되었는지 확인할 수 있습니다.
> - 인증서를 암호화하면 60~90일마다 자동으로 갱신됩니다.
> - 설정이 진행 중인 동안에는 인스턴스를 중지하거나 변경하지 마세요. 인스턴스를 구성하는 데 최대 15분이 걸릴 수 있습니다. 인스턴스 연결 탭에서 각 단계의 진행 상황을 볼 수 있습니다.

웹 사이트 설정 마법사를 사용하여 인스턴스를 구성

1. 인스턴스 관리 페이지의 연결 탭에서 웹 사이트 설정을 선택합니다.



2. 도메인 이름 지정에서 기존 Lightsail 관리형 도메인을 사용하거나, Lightsail에 새 도메인을 등록하거나, 다른 도메인 등록 기관을 사용하여 등록한 도메인을 사용합니다. 이 도메인 사용을 선택하여 다음 단계로 이동합니다.

3. DNS 구성에 대해 다음 작업 중 한 가지를 수행하세요.
   - Lightsail DNS 영역을 사용하려면 Lightsail 관리형 도메인을 선택합니다. 다음 단계로 이동하려면 이 DNS 영역 사용을 선택합니다.
   - 도메인의 DNS 레코드를 관리하는 호스팅 서비스를 사용하려면 타사 도메인을 선택합니다. 나중에 사용하기로 결정할 경우를 대비하여 Lightsail 계정에 일치하는 DNS 영역을 생성합니다. 타사 DNS 사용을 선택하여 다음 단계로 이동합니다.
4. 고정 IP 주소 생성에서 고정 IP 주소의 이름을 입력한 다음 고정 IP 생성을 선택합니다.
5. 도메인 할당 관리에서 할당 추가를 선택하고 도메인 유형을 선택한 다음 추가를 선택합니다. 계속을 선택하여 다음 단계로 이동합니다.
6. SSL/TLS 인증서 생성에서 도메인 및 하위 도메인을 선택하고 이메일 주소를 입력한 다음 Lightsail에 권한을 부여하여 인스턴스에서 Let's Encrypt 인증서를 구성하기를 선택하고 인증서 생성을 선택합니다. Lightsail 리소스 구성을 시작합니다.

   설정이 진행 중인 동안에는 인스턴스를 중지하거나 변경하지 마세요. 인스턴스를 구성하는 데 최대 15분이 걸릴 수 있습니다. 인스턴스 연결 탭에서 각 단계의 진행 상황을 볼 수 있습니다.
7. 웹 사이트 설정이 완료되면 도메인 할당 단계에서 지정한 URL이 WordPress 사이트를 여는지 확인합니다.

옵션: 개별 작업

개별 작업을 완료하여 인스턴스를 구성

1. 고정 IP 주소 생성

   인스턴스 관리 페이지의 네트워킹 탭에서 고정 IP 생성을 선택합니다. 고정 IP 위치 및 인스턴스가 선택됩니다. 고정 IP 주소의 이름을 지정한 다음 생성 및 연결을 선택합니다.
2. DNS 영역 생성

   탐색 창에서 도메인 및 DNS를 선택합니다. DNS 영역 생성을 선택하고 도메인을 입력한 다음 DNS 영역 생성을 선택합니다. 웹 트래픽이 현재 도메인으로 라우팅되는 경우 도메인의 현재 DNS 호스팅 공급자에서 이름 서버를 변경하기 전에 모든 기존 DNS 레코드가 Lightsail DNS 영역에 있는지 확인합니다. 이런 식으로 Lightsail DNS 영역으로 이전된 후 트래픽이 중단되지 않고 계속 흐릅니다.
3. 도메인 할당 관리

   DNS 영역 페이지에서 할당 탭에서 할당 추가를 선택합니다. 도메인 또는 하위 도메인을 선택하고 인스턴스를 선택한 다음 정적 IP 주소를 연결한 다음 할당을 선택합니다.

> **Tip**
>
> 도메인이 WordPress 인스턴스로 트래픽을 라우팅하기 전에 인터넷으로 이 변경 사항이 전파될 때까지 기다립니다.

4. SSL/TLS 인증서 생성 및 설치

   단계별 지침은 the section called “HTTPS 활성화” 섹션을 참조하세요.

5. 도메인 할당 단계에서 지정한 URL이 WordPress 사이트를 여는지 확인합니다.

4단계: WordPress 웹 사이트에 대한 관리자 암호 가져오기

WordPress 웹사이트의 관리 대시보드에 로그인하기 위한 기본 암호가 인스턴스에 저장됩니다. 암호를 가져오려면 다음 단계를 완료합니다.

WordPress 관리자의 기본 암호 가져오기

1. WordPress 인스턴스의 인스턴스 관리 페이지를 엽니다.
2. WordPress 패널에서 기본 암호 검색을 선택합니다. 이렇게 하면 페이지 하단의 액세스 기본 암호가 확장됩니다.

3. CloudShell 시작을 선택합니다. 이는 페이지 하단에 분할 창을 엽니다.
4. 복사를 선택한 다음 콘텐츠를 CloudShell 창에 붙여넣습니다. 커서를 CloudShell 프롬프트에 놓고 Ctrl+V를 누르거나 마우스 오른쪽 버튼을 클릭하여 메뉴를 열고 붙여넣기를 선택할 수 있습니다.

5. CloudShell 창에 표시된 암호를 기록해 둡니다. 나중에 WordPress 웹사이트의 관리 대시보드에 로그인하는 데 사용합니다.

```
~ $ AWS_REGION=us-east-1 ~/lightsail_connect WordPress-1 tail -n 1 application_credentials
LnPLnPLnPLnPLnP9
```

5단계: WordPress 웹사이트의 관리 대시보드 로그인

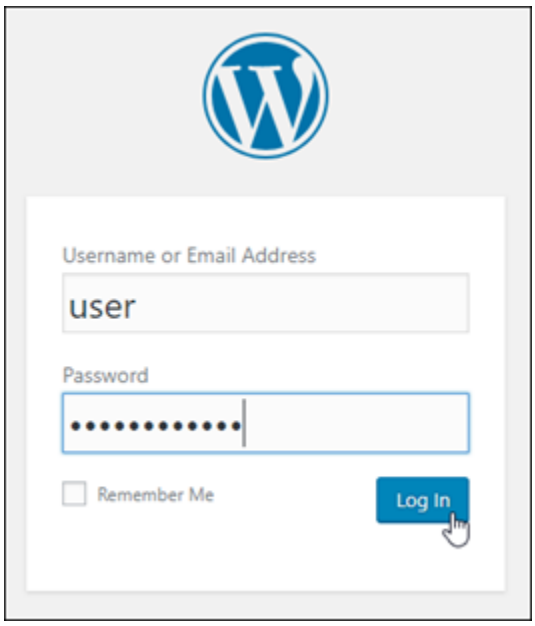
이제 WordPress 웹사이트의 관리 대시보드에 대한 암호가 있으므로 로그인할 수 있습니다. 관리 대시보드에서 사용자 암호를 변경하고, 플러그인을 설치하고, 웹사이트의 테마를 변경하는 등의 작업을 할 수 있습니다.

WordPress 웹사이트의 관리 대시보드에 로그인하려면 다음 단계를 완료합니다.

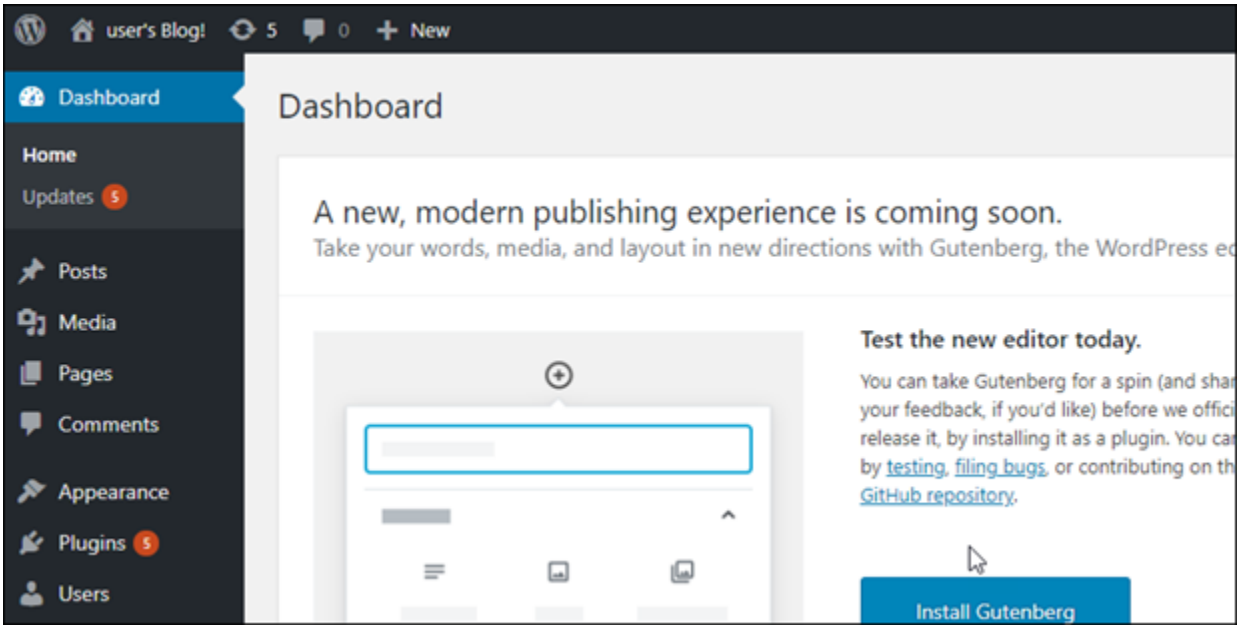
관리자 대시보드에 로그인

1. WordPress 인스턴스의 인스턴스 관리 페이지를 엽니다.
2. WordPress 패널에서 WordPress 관리자 액세스를 선택합니다.
3. WordPress 관리 대시보드 액세스 패널의 퍼블릭 IP 주소 사용에서 다음 형식의 링크를 선택합니다.

   http://*public-ipv4-address*./wp-admin
4. 사용자 이름 또는 이메일 주소 상자에 **user**를 입력합니다.
5. 암호에 이전 단계에서 얻은 암호를 입력합니다.
6. 그런 다음 로그인을 선택합니다.



이제 관리 작업을 수행할 수 있는 WordPress 웹사이트의 관리 대시보드에 로그인되었습니다. WordPress 웹사이트 관리에 대한 자세한 내용은 WordPress 설명서의 WordPress Codex를 참조하세요.



추가 정보

다음은 Amazon Lightsail에서 WordPress 인스턴스를 실행한 후 수행할 수 있는 몇 가지 추가 단계입니다.

- the section called “CDN 구성”
- Linux 또는 Unix 인스턴스의 스냅샷 생성
- 인스턴스 또는 디스크의 자동 스냅샷 활성화 또는 비활성화
- 블록 스토리지 디스크를 추가로 생성하고 Linux 기반의 인스턴스에 연결

WP Offload Media를 사용하여 Lightsail의 WordPress 웹 사이트를 Amazon S3에 연결

이 자습서에서는 Amazon Lightsail 인스턴스에서 실행되는 WordPress 웹 사이트를 Amazon Simple Storage Service(Amazon S3) 버킷에 연결하여 웹 사이트 이미지와 첨부 파일을 저장하는 데 필요한 단계를 설명합니다. 이 작업을 위해 Amazon Web Services(AWS) 계정 보안 인증 세트로 WordPress 플러그인을 구성합니다. 그러면 플러그인이 Amazon S3 버킷을 만들고, 웹 사이트 이미지와 첨부 파일을 위해 인스턴스의 디스크 대신 버킷을 사용하도록 웹 사이트를 구성합니다.

주제

- 1단계: 필수 구성 요소 완성
- 2단계: WordPress 웹 사이트에서 WP Offload Media 플러그인 설치
- 3단계: IAM 정책 생성
- 4단계: IAM 사용자 생성
- 5단계: IAM 사용자에 대한 액세스 키를 생성합니다.
- 6단계: WordPress 구성 파일 편집
- 7단계: WP Offload Media 플러그인을 사용하여 Amazon S3 버킷 생성
- 8단계: 다음 단계

1단계: 필수 구성 요소 완성

시작하기 전에 Lightsail에서 WordPress 인스턴스를 생성하고 실행 중 상태가 되게 합니다. 자세한 내용은 자습서: WordPress 인스턴스 시작 및 구성을 참조하세요.

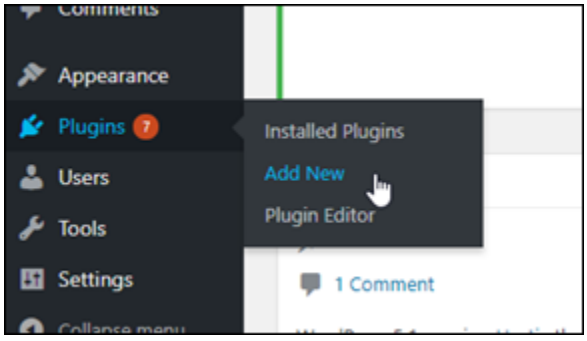
2단계: WordPress 웹 사이트에서 WP Offload Media 플러그인 설치

플러그인을 사용하여 Amazon S3 버킷을 사용하도록 웹 사이트를 구성해야 합니다. 이 구성에 사용할 수 있는 여러 플러그인 중에 WP Offload Media Lite가 있습니다.

WordPress 웹 사이트에 WP Offload Media 플러그인을 설치하려면

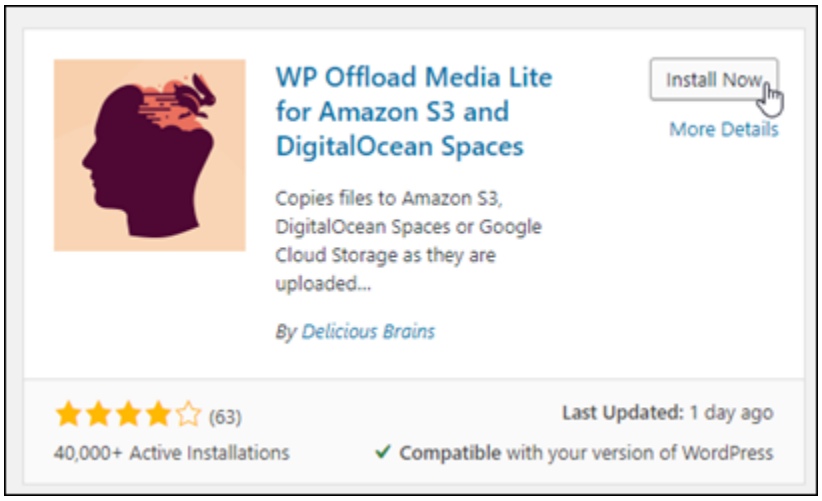
1. WordPress 대시보드에 관리자로 로그인합니다.

   자세한 내용은 Amazon Lightsail의 Bitnami 인스턴스에 대한 애플리케이션 사용자 이름과 암호 가져오기를 참조하세요.
2. 왼쪽 탐색 메뉴에서 Plugins(플러그인) 위로 마우스를 이동하고 새로 추가를 선택합니다.



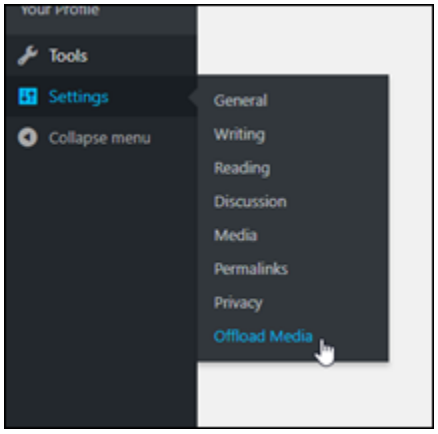
3. WP Offload Media Lite를 검색합니다.

4. 검색 결과에서 WP Offload Media 플러그 인 옆에 있는 지금 설치(Install Now)를 선택합니다.



5. 플러그 인 설치가 끝나면 활성화(Activate)를 선택합니다.

6. 왼쪽 탐색 메뉴에서 설정을 선택한 후 Offload Media를 선택합니다.



7. Offload Media 페이지에서 Amazon S3를 스토리지 공급자로 선택한 후 wp-config.php에서 액세스 키 정의를 선택합니다.

   이 옵션을 사용하면 인스턴스의 `wp-config.php`에 AWS 계정 자격 증명을 추가해야 합니다. 이러한 단계는 본 자습서 뒷부분에서 다룹니다.

Offload Media 페이지를 열어 놓으세요. 본 자습서 뒷부분에서 이 페이지로 돌아갑니다. 이 자습서의 3단계: IAM 정책 생성 섹션으로 계속 진행합니다.

3단계: IAM 정책 생성

> **Warning**
>
> 이 시나리오에서는 프로그래밍 방식 액세스 권한과 장기 보안 인증이 있는 IAM 사용자가 필요하며 이는 보안 위험을 내포합니다. 이 위험을 줄이려면 이러한 사용자에게 작업을 수행하는 데 필요한 권한만 제공하고 더 이상 필요하지 않을 경우 이러한 사용자를 제거하는 것이 좋습니다. 필요한 경우 액세스 키를 업데이트할 수 있습니다. 자세한 내용은 IAM 사용 설명서의 액세스 키 업데이트를 참조하세요.

WP Offload Media 플러그인을 사용하려면 AWS 계정에 액세스하여 Amazon S3 버킷을 생성하고 웹 사이트 이미지 및 첨부 파일을 업로드해야 합니다.

WP Offload Media 플러그인에 대한 새 AWS Identity and Access Management (IAM) 정책을 생성하려면

1. 새 브라우저 탭을 열고 IAM 콘솔에 로그인합니다.
2. 왼쪽 탐색 메뉴의 액세스 관리에서 정책을 선택합니다.
3. 정책 생성을 선택합니다.

4. 정책 생성 페이지에서 JSON 탭을 선택한 정책 편집기에서 모든 콘텐츠를 제거합니다.
5. 정책 편집기에서 다음 콘텐츠를 지정하여 예제 버킷 이름인 *amzn-s3-demo-bucket*을 자체 이름으로 바꿉니다.

   JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "s3:*",
            "Resource": [
                "arn:aws:s3:::amzn-s3-demo-bucket/*",
                "arn:aws:s3:::amzn-s3-demo-bucket"
            ]
        }
    ]
}
```

6. 다음을 선택합니다.
7. [정책 이름(Policy name)]에 정책의 이름을 입력합니다.

> **Tip**
> 나중에 유지 관리를 수행할 때 쉽게 식별할 수 있도록 **wp_s3_user_policy** 또는 **wp_offload_media_plugin_user_policy**와 같이 설명적인 이름을 지정합니다.

8. 정책 생성을 선택합니다.

   다음 단계를 위해 IAM 콘솔을 열어 둡니다.

4단계: IAM 사용자 생성

새 IAM 사용자를 생성하고 이전에 생성된 정책을 연결하여 WP Offload Media 플러그인을 사용하는 데 필요한 권한을 부여합니다.

WP Offload Media 플러그인에 대한 새 AWS Identity and Access Management (IAM) 사용자를 생성하려면

1. 필요한 경우 IAM 콘솔을 엽니다.
2. 왼쪽 탐색 메뉴의 액세스 관리에서 사용자를 선택합니다.
3. 사용자 생성을 선택합니다.
4. 사용자 이름에 새 사용자의 이름을 입력한 후 다음을 선택합니다.

   > **Tip**
   >
   > 나중에 유지 관리를 수행할 때 쉽게 식별할 수 있도록 **wp_s3_user** 또는 **wp_offload_media_plugin_user**와 같이 설명적인 이름을 지정합니다.

5. 정책 직접 연결을 선택합니다.
6. 권한 정책에 이전에 검색 모음에서 생성한 정책의 이름을 입력합니다.
7. 정책을 선택하고 다음을 선택합니다.
8. 사용자 생성을 선택합니다.

   다음 단계를 위해 IAM 콘솔을 열어 둡니다.

5단계: IAM 사용자에 대한 액세스 키를 생성합니다.

WP Offload Media 플러그인이 사용할 IAM 사용자의 액세스 키를 생성합니다.

WP Offload Media 플러그인에 대한 새 AWS Identity and Access Management (IAM) 사용자를 생성하려면

1. 필요한 경우 IAM 콘솔을 엽니다.
2. 탐색 창의 액세스 관리에서 사용자를 선택합니다.
3. 사용자 이름을 선택하여 사용자 세부 정보 페이지로 이동합니다.
4. 보안 자격 증명 탭의 액세스 키 섹션에서 액세스 키 생성을 선택합니다.
5. 기타를 선택한 후 다음을 선택합니다.
6. 액세스 키 생성을 선택합니다.
7. IAM 사용자에 대한 액세스 키 ID 및 시크릿 액세스 키를 기록해 둡니다. .csv 다운로드를 선택하여 이러한 값의 복사본을 로컬 드라이브에 저장할 수도 있습니다. WordPress 인스턴스에서 `wp-config.php` 파일을 편집할 때 이후의 몇 단계에서 이 값이 필요합니다.

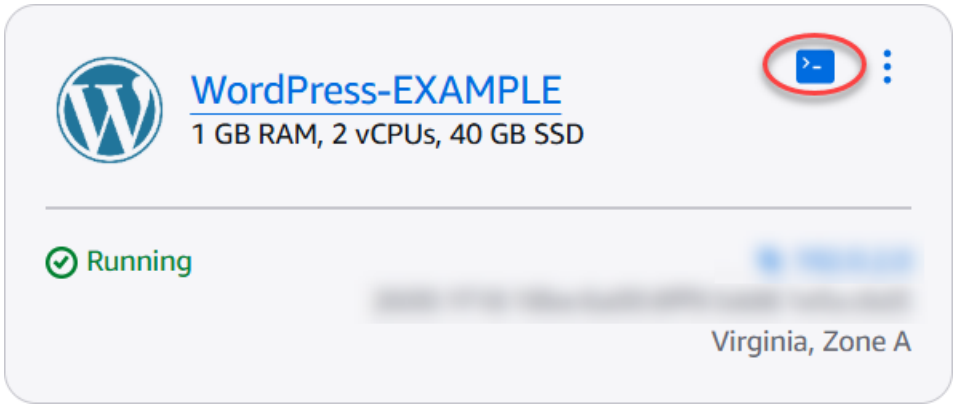
이제 IAM 콘솔을 닫고 Lightsail 콘솔에서 다음 단계를 계속할 수 있습니다.

6단계: WordPress 구성 파일 편집

`wp-config.php` 파일에는 데이터베이스 연결 정보 등 웹 사이트의 기본 구성 세부 정보가 포함됩니다.

WordPress 인스턴스에서 **`wp-config.php`** 파일을 편집하려면

1. Lightsail 콘솔에 로그인합니다.
2. WordPress 인스턴스에 대한 브라우저 기반 SSH 클라이언트 아이콘을 선택합니다.



> **Note**
>
> 자체 SSH 클라이언트를 사용하여 인스턴스에 연결할 수도 있습니다. 자세한 내용은 Lightsail에서 SSH를 사용하여 연결할 PuTTY 다운로드 및 설정을 참조하세요.

3. SSH 클라이언트 창이 나타나면 다음 명령을 입력하여 문제가 생길 경우에 대비해 `wp-config.php` 파일의 백업을 생성합니다.

```
sudo cp /opt/bitnami/wordpress/wp-config.php /opt/bitnami/wordpress/wp-config.php.backup
```

4. 다음 명령을 입력하여 텍스트 편집기 nano를 사용해 `wp-config.php` 파일을 엽니다.

```
nano /opt/bitnami/wordpress/wp-config.php
```

5. `/* That's all, stop editing! Happy blogging. */` 텍스트 위에 다음 텍스트를 입력합니다.

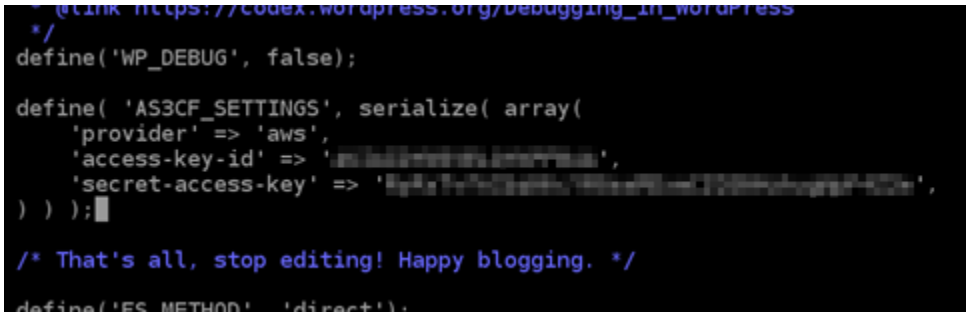
*AccessKeyID*를 액세스 키 ID로 바꾸고 *SecretAccessKey*를 이 단계 전에 만든 IAM 사용자의 시크릿 액세스 키로 바꿉니다.

```
define( 'AS3CF_SETTINGS', serialize( array(
    'provider' => 'aws',
    'access-key-id' => 'AccessKeyID',
    'secret-access-key' => 'SecretAccessKey',
) ) );
```

예제:

```
define( 'AS3CF_SETTINGS', serialize( array(
    'provider' => 'aws',
    'access-key-id' => 'AKIAIOSFODNN7EXAMPLE',
    'secret-access-key' => 'wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY',
) ) );
```

결과는 다음 예제와 같아야 합니다.



6. **Ctrl+X**를 눌러 Nano를 끝낸 후 **Y**를 누르고 **Enter** 키를 눌러 편집 내용을 `wp-config.php` 파일에 저장합니다.
7. 다음 명령을 입력하여 인스턴스에서 서비스를 다시 시작합니다.

```
sudo /opt/bitnami/ctlscript.sh restart
```

서비스가 다시 시작되면 다음과 비슷한 결과가 표시됩니다.

```
bitnami@ip-172-26-13-236:~$ sudo /opt/bitnami/ctlscript.sh restart
Syntax OK
/opt/bitnami/apache2/scripts/ctl.sh : httpd stopped
/opt/bitnami/php/scripts/ctl.sh : php-fpm stopped
/opt/bitnami/mysql/scripts/ctl.sh : mysql stopped
/opt/bitnami/mysql/scripts/ctl.sh : mysql  started at port 3306
/opt/bitnami/php/scripts/ctl.sh : php-fpm started
Syntax OK
/opt/bitnami/apache2/scripts/ctl.sh : httpd started at port 80
bitnami@ip-172-26-13-236:~$
```

SSH 창을 닫고, 이 자습서 앞부분에서 열어 놓은 Offload Media 페이지로 전환합니다. 이제 WP Offload Media 플러그인을 사용하여 Amazon S3 버킷을 생성할 준비가 되었습니다.

7단계: WP Offload Media 플러그인을 사용하여 Amazon S3 버킷 생성

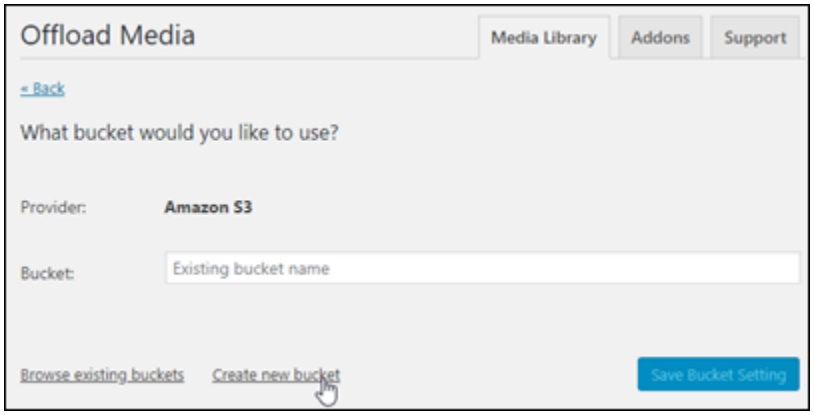
이제 `wp-config.php` 파일이 AWS 자격 증명으로 구성되었으므로 Offload Media 페이지로 돌아가 절차를 완료할 수 있습니다.

WP Offload Media 플러그인을 사용하여 Amazon S3 버킷을 생성하려면

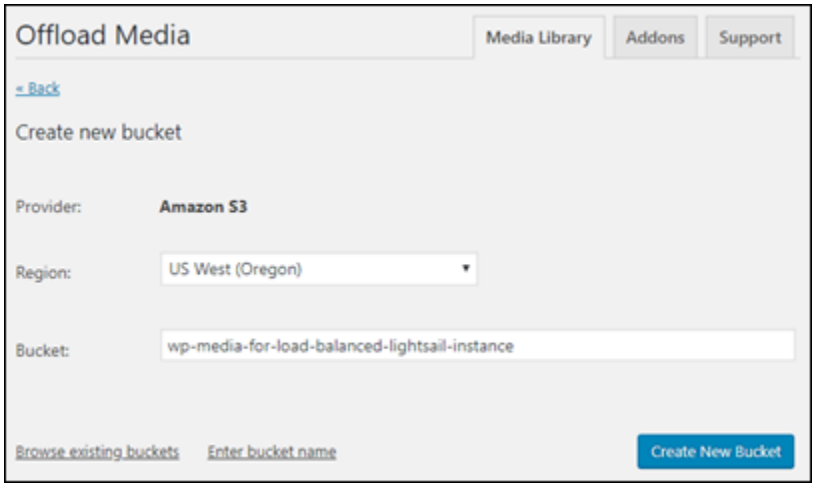
1. Offload Media 페이지를 새로 고치거나 다음을 선택합니다.

   Amazon S3 공급자가 구성되어 있어야 합니다.

2. 버킷 새로 만들기를 선택합니다.

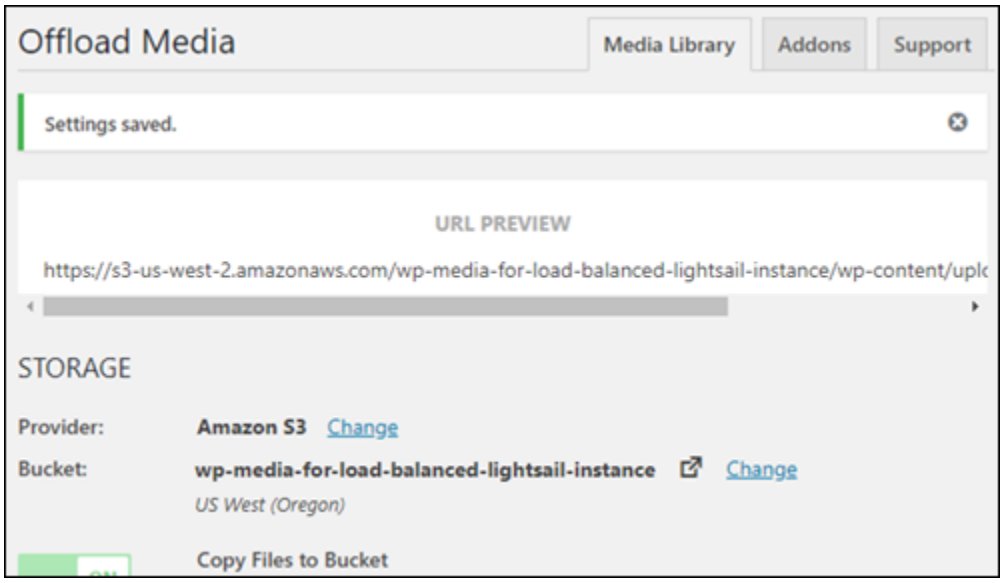


3. 리전 드롭다운 메뉴에서 원하는 AWS 리전을 선택합니다. WordPress 인스턴스가 있는 리전과 동일한 리전을 선택하는 것이 좋습니다.
4. 버킷 텍스트 상자에 새 S3 버킷 이름을 입력합니다.



5. 버킷 새로 만들기를 선택합니다.

   새 버킷이 생성되었음을 확인하도록 페이지가 새로 고쳐집니다. 표시되는 설정을 검토하고, 원하는 WordPress 웹 사이트 작동 방식에 맞게 설정을 조정합니다.



   이제부터 블로그 게시물에 추가하는 이미지와 첨부 파일이 생성된 Amazon S3 버킷에 자동으로 업로드됩니다.

8단계: 다음 단계

WordPress 웹 사이트를 Amazon S3 버킷에 연결한 후에는 WordPress 인스턴스 스냅샷을 만들어 변경 내용을 백업해야 합니다. 자세한 내용은 Linux 또는 Unix 인스턴스의 스냅샷 생성을 참조하세요.

Lightsail WordPress 인스턴스를 Amazon Aurora 데이터베이스에 연결

게시물, 페이지 및 사용자에 대한 웹 사이트 데이터는 Amazon Lightsail의 WordPress 인스턴스에서 실행 중인 데이터베이스에 저장됩니다. 인스턴스에 장애가 발생하면 데이터가 복구 불가능한 상태가 될 수 있습니다. 이러한 상황을 방지하려면 Amazon Relational Database Service(RDS)의 Amazon Aurora 데이터베이스로 웹 사이트 데이터를 전송해야 합니다.

Amazon Aurora는 클라우드용으로 구축된 MySQL 및 PostgreSQL 호환 관계형 데이터베이스입니다. 이는 기존 엔터프라이즈 데이터베이스의 성능 및 가용성과 오픈 소스 데이터베이스의 단순성 및 비용 효율성을 결합합니다. Aurora는 Amazon RDS의 일부로 제공됩니다. Amazon RDS는 클라우드에서 관계형 데이터베이스를 더 쉽게 설치, 운영 및 크기 조정할 수 있는 관리형 데이터베이스 서비스입니다. 자세한 내용은 https://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/Amazon Relational Database Service 사용 설명서와 https://docs.aws.amazon.com/AmazonRDS/latest/AuroraUserGuide/Amazon Aurora - Aurora 사용 설명서를 참조하세요.

이 자습서에서는 Lightsail의 WordPress 인스턴스에 있는 웹사이트 데이터베이스를 Amazon RDS의 Aurora 관리형 데이터베이스에 연결하는 방법을 보여줍니다.

목차

- 1단계: 필수 구성 요소 완성
- 2단계: Aurora 데이터베이스를 위한 보안 그룹 구성
- 3단계: Lightsail 인스턴스에서 Aurora 데이터베이스로 연결
- 4단계: WordPress 인스턴스에서 Aurora 데이터베이스로 MySQL 데이터베이스 전송
- 5단계: Aurora 관리형 데이터베이스에 연결하도록 WordPress 구성

1단계: 필수 구성 요소 완성

시작하기 전에 다음 사전 조건을 완료합니다.

1. Lightsail에 WordPress 인스턴스를 생성하고 여기에 애플리케이션을 구성합니다. 계속하기 전에 인스턴스가 실행 중인 상태여야 합니다. 자세한 내용은 자습서: Amazon Lightsail에서 WordPress 인스턴스 시작 및 구성을 참조하세요.
2. Lightsail 계정에서 VPC 피어링을 켭니다. 자세한 내용은 외부 AWS 리소스로 작업하도록 피어링 설정을 참조하세요Lightsail.

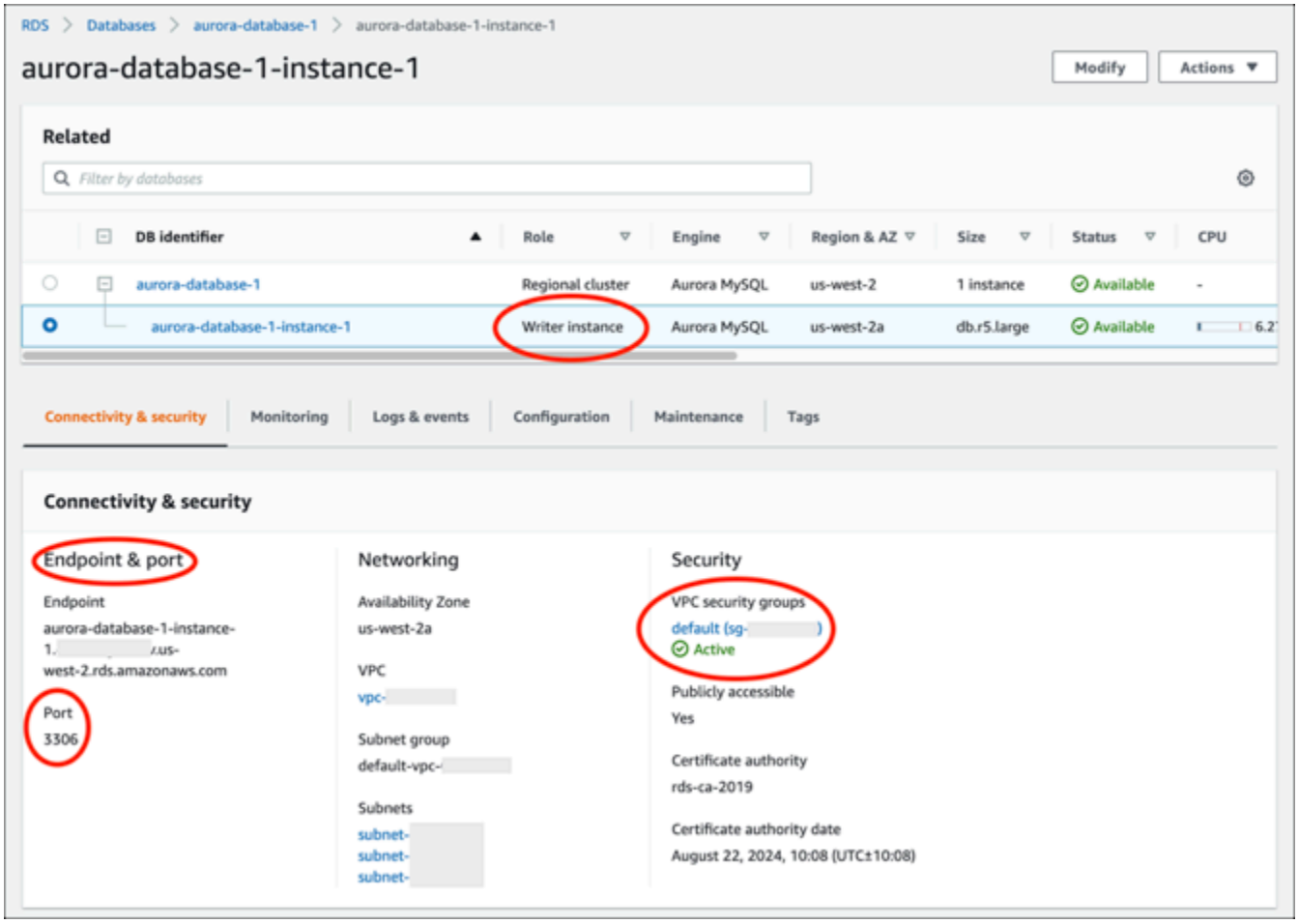
3. Amazon RDS에 Aurora 관리형 데이터베이스를 생성합니다. 데이터베이스는 WordPress 인스턴스와 동일한 AWS 리전 에 있어야 합니다. 계속하기 전에 데이터베이스도 실행 중인 상태여야 합니다. 자세한 내용은 Amazon Aurora 사용 설명서의 Amazon Aurora 시작하기를 참조하세요.

2단계: Aurora 데이터베이스를 위한 보안 그룹 구성

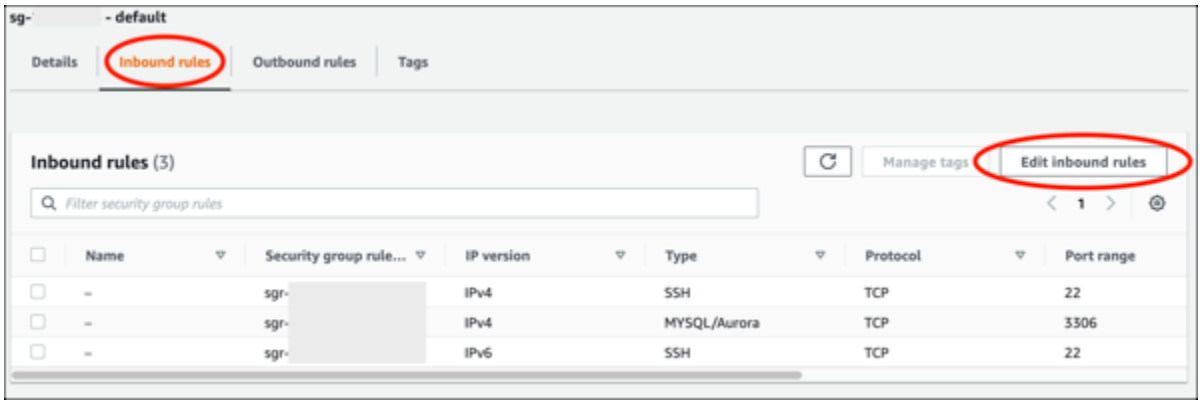
AWS 보안 그룹은 AWS 리소스의 가상 방화벽 역할을 합니다. 이 보안 그룹은 Amazon RDS의 Aurora 데이터베이스에 연결할 수 있는 수신 및 발신 트래픽을 제어합니다. 보안 그룹에 대한 자세한 내용은 Amazon Virtual Private Cloud 사용 설명서의 보안 그룹을 사용하여 리소스에 대한 트래픽 제어를 참조하세요.

다음 절차를 완료하여 WordPress 인스턴스가 Aurora 데이터베이스에 연결을 설정할 수 있도록 보안 그룹을 구성합니다.

1. Amazon RDS 콘솔에 로그인합니다.
2. 탐색 창에서 [Databases(데이터베이스)]를 선택합니다.
3. WordPress 인스턴스가 연결할 Aurora 데이터베이스의 라이터 인스턴스를 선택합니다.
4. 연결 및 보안(Connectivity & security) 탭을 선택합니다.
5. 엔드포인트 및 포트(Endpoint & port) 섹션에서라이터 인스턴스(Writer instance)의 엔드포인트 이름(Endpoint name)과 포트(Port)를 기록해 둡니다. 이 정보는 나중에 데이터베이스에 연결하기 위해 Lightsail 인스턴스를 구성할 때 필요합니다.
6. 보안(Security) 섹션에서 활성 VPC 보안 그룹 링크를 선택합니다. 데이터베이스의 보안 그룹으로 리디렉션됩니다.



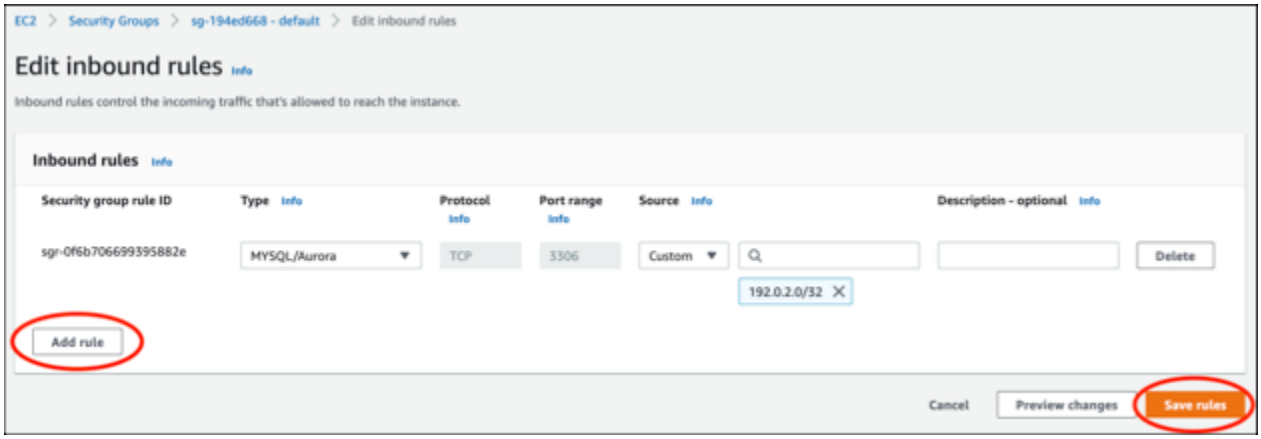
7. Aurora 데이터베이스에 대한 보안 그룹이 선택되어 있는지 확인합니다.
8. 인바운드 규칙 탭을 선택합니다.
9. 인바운드 규칙 편집을 선택합니다.



10. 인바운드 규칙 편집(Edit inbound rules) 페이지에서 규칙 추가(Add rule)를 선택합니다.
11. 다음 단계 중 하나를 완료합니다.

- 기본 MySQL 포트 3306을 사용하는 경우, 유형(Type) 드롭다운 메뉴에서 MySQL/Aurora를 선택합니다.
- 데이터베이스에 사용자 지정 포트를 사용하는 경우, 유형(Type) 드롭다운 메뉴에서 사용자 지정 TCP(Custom TCP)를 선택하고 포트 범위(Port Range)텍스트 상자에 포트 번호를 입력합니다.

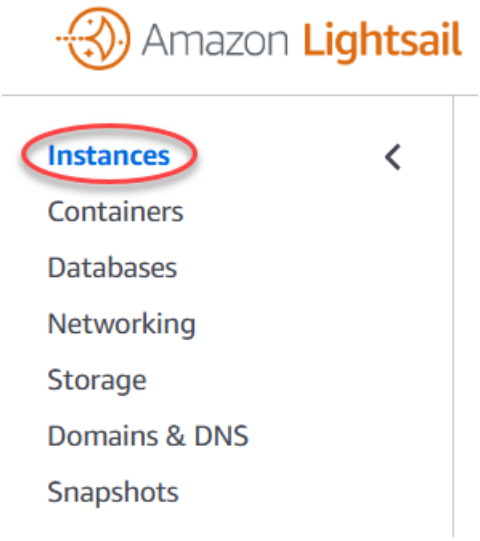
12. 소스(Source) 텍스트 상자에 WordPress 인스턴스의 프라이빗 IP 주소를 추가합니다. IP 주소는 CIDR 표기법으로 입력해야 합니다. 즉, `/32`를 추가해야 합니다. 예를 들어, `192.0.2.0`을 허용하려면 `192.0.2.0/32`를 입력합니다.
13. 규칙 저장을 선택합니다.



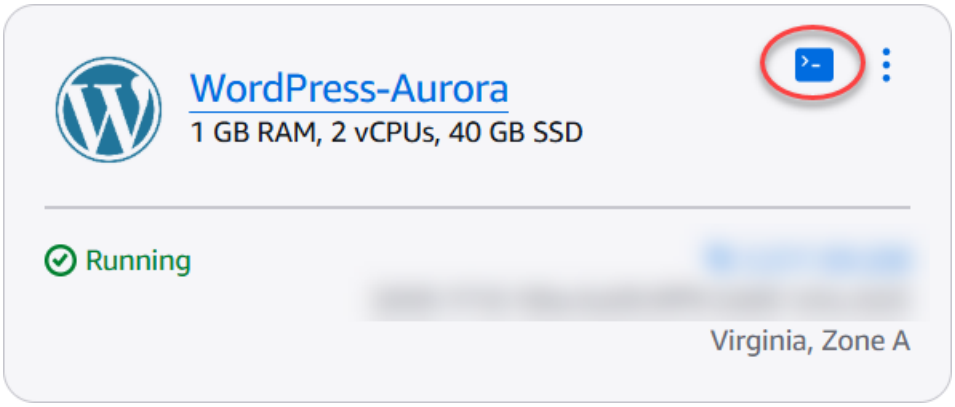
3단계: Lightsail 인스턴스에서 Aurora 데이터베이스로 연결

다음 절차를 완료하여 Lightsail 인스턴스에서 Aurora 데이터베이스에 연결할 수 있는지 확인합니다.

1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 인스턴스를 선택합니다.



3. WordPress 인스턴스가 SSH를 사용하여 연결할 브라우저 기반 SSH 클라이언트 아이콘을 선택합니다.



4. 인스턴스에 연결한 후 다음 명령을 입력하여 Aurora 데이터베이스에 연결합니다. 명령에서 *DatabaseEndpoint*는 Aurora 데이터베이스의 엔드포인트 주소로, *Port*는 데이터베이스의 포트로 바꿉니다. *MyUserName*은 데이터베이스를 생성할 때 입력했던 사용자 이름으로 바꿉니다.

```
mysql -h DatabaseEndpoint -P Port -u MyUserName -p
```

인스턴스가 Aurora 데이터베이스에 액세스 및 연결할 수 있음을 확인해 주는 다음 예와 유사한 응답이 표시되어야 합니다.

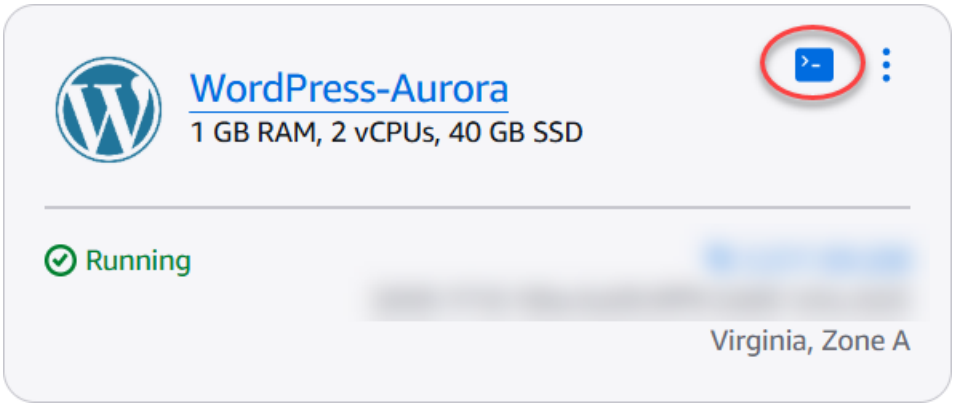
```
bitnami@ip-            $ mysql -h database.cluster-            .us-west-2.rds.amazonaws.
com -P 3306 -u admin -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MySQL connection id is 215
Server version: 5.6.10 MySQL Community Server (GPL)

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MySQL [(none)]>
```

이 응답이 표시되지 않거나 오류 메시지가 나타나면 Lightsail 인스턴스의 프라이빗 IP 주소가 연결할 수 있도록 Aurora 데이터베이스의 보안 그룹을 구성해야 할 수 있습니다. 자세한 내용은 이 설명서의 Aurora 데이터베이스에 대한 보안 그룹 구성을 참조하세요.

4단계: WordPress 인스턴스에서 Aurora 데이터베이스로 데이터베이스 전송

인스턴스에서 데이터베이스에 연결할 수 있다는 것을 확인했으므로 이제 WordPress 웹 사이트 데이터를 Aurora 데이터베이스로 전송해야 합니다.

1. Lightsail 콘솔에 로그인합니다.
2. 인스턴스(Instances) 탭에서 WordPress 인스턴스에 대한 브라우저 기반 SSH 클라이언트를 선택합니다.



3. 브라우저 기반SSH 클라이언트를 WordPress 인스턴스에 연결한 후 다음 명령을 입력합니다. 이 명령은 인스턴스에 있는 `bitnami_wordpress` 데이터베이스에서 Aurora 데이터베이스로 데이터를 전송합니다. 명령에서 *`DatabaseUserName`*을 Aurora 데이터베이스 생성 시 입력한 기본 사용자의 이름을 바꿉니다. *`DatabaseEndpoint`*를 Aurora 데이터베이스의 엔드포인트 주소로 바꿉니다.

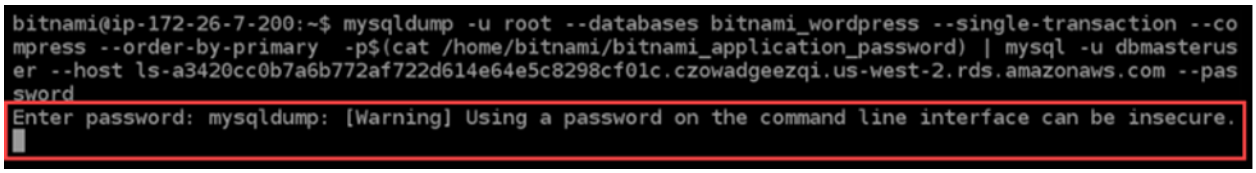
```
sudo mysqldump -u root --databases bitnami_wordpress --single-transaction --
compress --order-by-primary -p$(cat /home/bitnami/bitnami_application_password) |
 sudo mysql -u DatabaseUserName --host  DatabaseEndpoint --password
```

예제

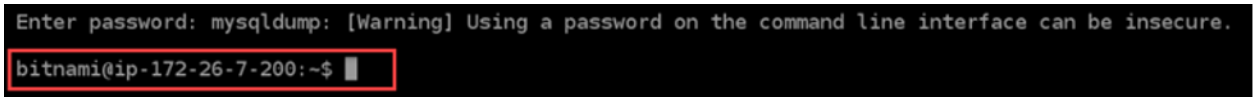
```
sudo mysqldump -u root --databases bitnami_wordpress --single-transaction --
compress --order-by-primary -p$(cat /home/bitnami/bitnami_application_password)
 | sudo mysql -u DBuser --host  abc123exampleE67890.czowadgeezqi.us-
west-2.rds.amazonaws.com --password
```

4. `Enter password` 프롬프트에서 Aurora 데이터베이스의 암호를 입력하고 Enter 키를 누릅니다.

   입력하는 동안에는 암호를 볼 수 없습니다.



   데이터가 성공적으로 전송된 경우 다음 예와 유사한 응답이 표시됩니다.



   오류가 발생하면 사용 중인 데이터베이스 사용자 이름, 암호 및 엔드포인트가 올바른지 확인하고 다시 시도합니다.

5단계: Aurora 데이터베이스에 연결하도록 WordPress 구성

애플리케이션 데이터를 Aurora 데이터베이스로 전송한 후 이에 연결하도록 WordPress를 구성해야 합니다. 다음 절차를 완료하여 웹 사이트가 Aurora 데이터베이스에 연결되도록 WordPress 구성 파일(`wp-config.php`)을 편집합니다.

1. WordPress 인스턴스에 연결된 브라우저 기반 SSH 클라이언트에서 다음 명령을 입력하여 `wp-config.php` 파일의 백업을 생성합니다.

```
cp /opt/bitnami/wordpress/wp-config.php /opt/bitnami/wordpress/wp-config.php-backup
```

2. 다음 명령을 입력하여 `wp-config.php` 파일을 쓰기 가능 파일로 만듭니다.

```
sudo chmod 664 /opt/bitnami/wordpress/wp-config.php
```

3. config 파일의 데이터베이스 사용자 이름을 Aurora 데이터베이스 생성 시 입력한 기본 사용자의 이름으로 바꿉니다.

```
sudo wp config set DB_USER DatabaseUserName
```

4. config 파일의 데이터베이스 호스트를 Aurora 데이터베이스의 엔드포인트 주소 및 포터 번호로 편집합니다. 예를 들어 abc123exampleE67890.czowadgeezqi.us-west-2.rds.amazonaws.com:3306입니다.

```
sudo wp config set DB_HOST DatabaseEndpoint:Port
```

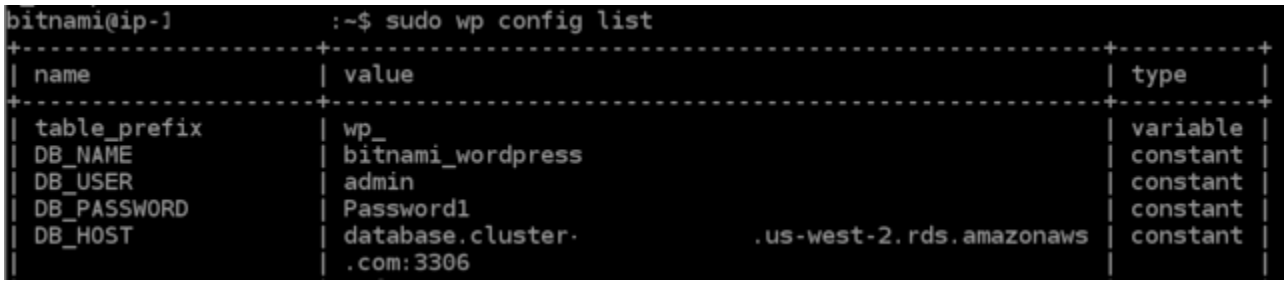
5. config 파일의 데이터베이스 암호를 Aurora 데이터베이스에 대한 암호로 편집합니다.

```
sudo wp config set DB_PASSWORD DatabasePassword
```

6. wp config list 명령을 입력하여 wp-config.php 파일에 입력한 정보가 올바른지 확인합니다.

```
sudo wp config list
```

다음 예와 같이 구성 세부 정보가 표시된 결과가 나타납니다.



7. 다음 명령을 입력하여 인스턴스의 웹 서비스를 다시 시작합니다.

```
sudo /opt/bitnami/ctlscript.sh restart
```

서비스가 다시 시작되면 다음 예와 유사한 결과가 표시됩니다.

축하합니다! 이제 WordPress 사이트가 Aurora 데이터베이스를 사용하도록 구성되었습니다.

> **Note**
>
> 원본 `wp-config.php` 파일을 복원해야 하는 경우에는 다음 명령을 입력하여 이 자습서 앞부분에서 생성한 백업을 통해 복원하면 됩니다.
>
> ```
> cp /opt/bitnami/wordpress/wp-config.php-backup /opt/bitnami/wordpress/wp-config.php
> ```

Lightsail에서 MySQL 관리형 데이터에 WordPress 데이터베이스 전송

게시물, 페이지 및 사용자에 대한 중요한 WordPress 웹 사이트 데이터는 Amazon Lightsail의 인스턴스에서 실행 중인 MySQL 데이터베이스에 저장됩니다. 인스턴스에 장애가 발생하면 데이터가 복구 불가능한 상태가 될 수 있습니다. 이러한 상황을 방지하려면 MySQL 관리형 데이터베이스로 웹 사이트 데이터를 전송해야 합니다.

이 자습서에서는 WordPress 웹 사이트 데이터를 Lightsail의 MySQL 관리형 데이터베이스로 전송하는 방법을 안내합니다. 또한, 웹 사이트가 관리형 데이터베이스에 연결되고 인스턴스에서 실행 중인 데이터베이스에 대한 연결을 중지하도록 인스턴스의 WordPress 구성(`wp-config.php`) 파일을 편집하는 방법도 보여줍니다.

목차

- 1단계: 사전 조건 완료
- 2단계: MySQL 관리형 데이터베이스에 WordPress 데이터베이스 전송
- 3단계: MySQL 관리형 데이터베이스에 연결되도록 WordPress 구성
- 4단계: 다음 단계 완료

1단계: 필수 구성 요소 완성

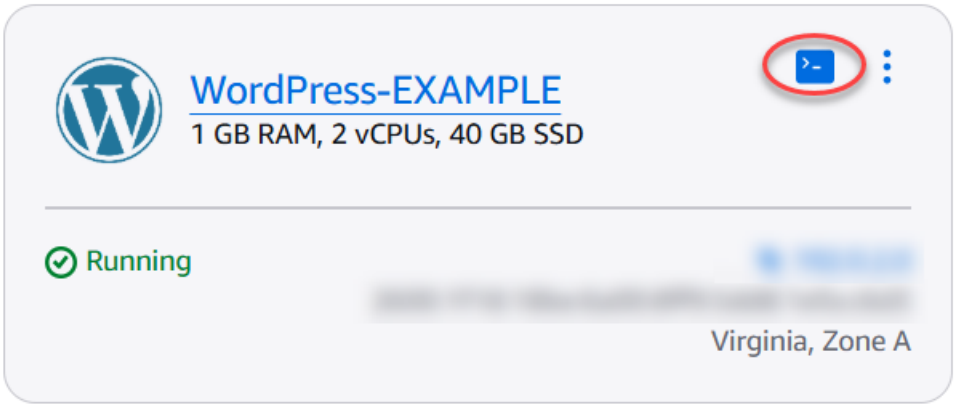
시작하기 전에 다음 사전 조건을 완료하세요.

- Lightsail에서 WordPress 인스턴스를 생성하고 이것이 실행 상태인지 확인합니다. 자세한 내용은 자습서: Amazon Lightsail에서 WordPress 인스턴스 시작 및 구성을 참조하세요.
- WordPress 인스턴스와 동일한 AWS 리전의 Lightsail에서 MySQL 관리형 데이터베이스를 생성하고 이것이 실행 상태인지 확인합니다. WordPress는 Lightsail에서 사용할 수 있는 모든 MySQL 데이터베이스 옵션을 지원합니다. 자세한 내용은 Amazon Lightsail에서 데이터베이스 생성을 참조하세요.
- MySQL 관리형 데이터베이스의 퍼블릭 모드 및 데이터 가져오기 모드를 사용하도록 설정합니다. 이 자습서에 나와 있는 단계들을 완료한 후에는 이들 모드를 비활성화할 수 있습니다. 자세한 내용은 데이터베이스의 퍼블릭 모드 구성 및 데이터베이스의 데이터 가져오기 모드 구성을 참조하세요.

2단계: MySQL 관리형 데이터베이스에 WordPress 데이터베이스 전송

다음 절차를 완료하여 WordPress 웹 사이트 데이터를 Lightsail의 MySQL 관리형 데이터베이스로 전송합니다.

1. Lightsail 콘솔에 로그인합니다.
2. 인스턴스 탭에서 WordPress 인스턴스에 대한 브라우저 기반 SSH 클라이언트 아이콘을 선택합니다.



3. 브라우저 기반 SSH 클라이언트가 WordPress 인스턴스에 연결된 후 다음 명령을 입력하여 인스턴스에 있는 `bitnami_wordpress` 데이터베이스의 데이터를 MySQL 관리형 데이터베이스로 전송합니다. *`DbUserName`*을 관리형 데이터베이스의 사용자 이름으로, *`DbEndpoint`*를 관리형 데이터베이스의 엔드포인트 주소로 변경합니다.

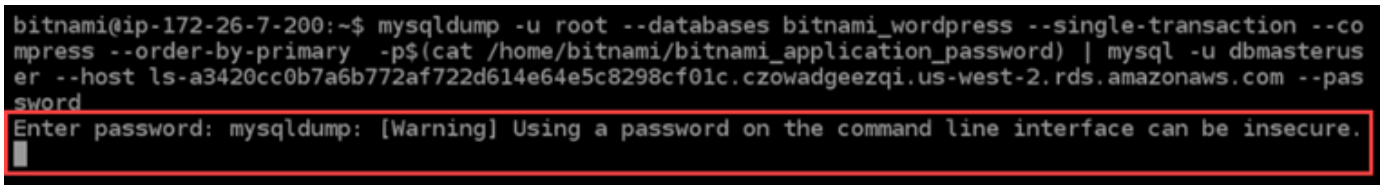
```
sudo mysqldump -u root --databases bitnami_wordpress --single-transaction --
compress --order-by-primary  -p$(cat /home/bitnami/bitnami_application_password) |
 sudo mysql -u DbUserName --host DbEndpoint --password
```

예

```
sudo mysqldump -u root --databases bitnami_wordpress --single-transaction --
compress --order-by-primary -p$(cat /home/bitnami/bitnami_application_password)
 | sudo mysql -u dbmasteruser --host ls-abc123exampleE67890.czowadgeezqi.us-
west-2.rds.amazonaws.com --password
```

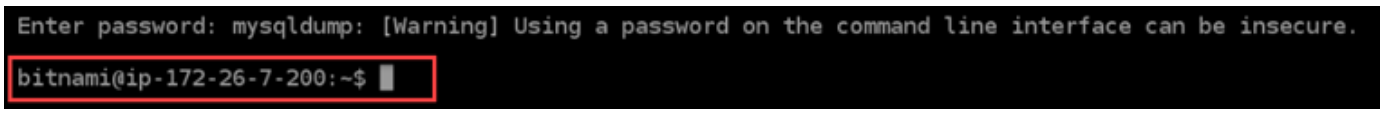
4. 프롬프트에서 MySQL 관리형 데이터베이스의 암호를 입력하고 Enter를 누릅니다.

   입력 중인 동안에는 암호를 볼 수 없습니다.



5. 데이터가 성공적으로 전송된 경우 다음 예와 유사한 응답이 표시됩니다.

   오류가 발생하면 사용 중인 데이터베이스 사용자 이름, 암호 또는 엔드포인트가 올바른지 확인하고 다시 시도합니다.



3단계: MySQL 관리형 데이터베이스에 연결되도록 WordPress 구성

다음 절차를 완료하여 웹 사이트가 MySQL 관리형 데이터베이스에 연결되도록 WordPress 구성 파일(`wp-config.php`)을 편집합니다.

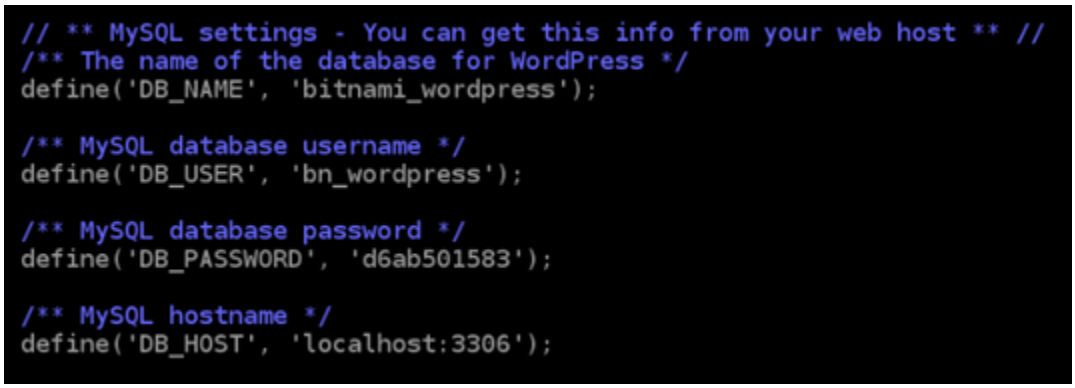
1. 문제가 발생할 경우를 대비해서 WordPress 인스턴스에 연결된 브라우저 기반 SSH 클라이언트에서 다음 명령을 입력하여 `wp-config.php` 파일의 백업을 생성합니다.

```
cp /opt/bitnami/wordpress/wp-config.php /opt/bitnami/wordpress/wp-config.php-backup
```

2. 다음 명령을 입력하여 Nano 텍스트 편집기로 `wp-config.php` 파일을 엽니다.

```
nano /opt/bitnami/wordpress/wp-config.php
```

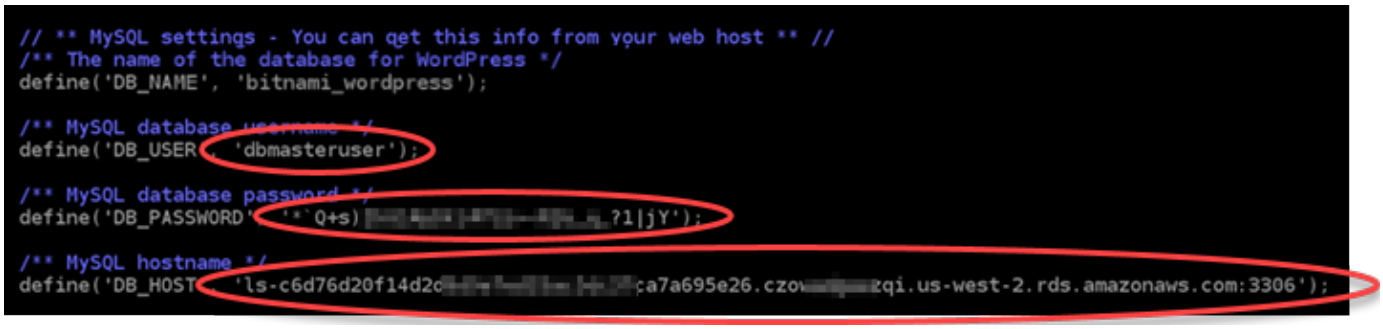
3. 다음 예와 같이 DB_USER, DB_PASSWORD 및 DB_HOST에 대한 값을 찾을 때까지 아래로 스크롤합니다.



4. 다음 값을 수정합니다.

   - DB_USER - MySQL 관리형 데이터베이스의 사용자 이름과 일치하도록 이 값을 편집합니다. Lightsail 관리형 데이터베이스의 기본 사용자 이름은 dbmasteruser입니다.
   - DB_PASSWORD - MySQL 관리형 데이터베이스의 강력한 암호와 일치하도록 이 값을 편집합니다. 자세한 내용은 데이터베이스 암호 관리를 참조하세요.
   - DB_HOST - MySQL 관리형 데이터베이스의 엔드포인트와 일치하도록 이 값을 편집합니다. 호스트 주소 끝에 반드시 :3306 포트 번호를 추가하십시오. 예: `ls-abc123exampleE67890.czowadgeezqi.us-west-2.rds.amazonaws.com:3306`.

   결과는 다음 예제와 같아야 합니다.



5. Ctrl+X 키를 눌러 Nano를 종료한 다음 Y 키와 Enter 키를 차례로 눌러 편집 내용을 저장합니다.
6. 다음 명령을 입력하여 인스턴스의 웹 서비스를 다시 시작합니다.

```
sudo /opt/bitnami/ctlscript.sh restart
```

서비스가 다시 시작되면 다음 예와 유사한 결과가 표시됩니다.

축하합니다! 이제 WordPress 사이트가 MySQL 관리형 데이터베이스를 사용하도록 구성되었습니다.

> **Note**
>
> 어떤 이유로든 원본 `wp-config.php` 파일을 복원해야 하는 경우에는 다음 명령을 입력하여 이 자습서 앞부분에서 생성한 백업을 통해 복원하면 됩니다.
>
> ```
> cp /opt/bitnami/wordpress/wp-config.php-backup /opt/bitnami/wordpress/wp-config.php
> ```

4단계: 다음 단계 완료

WordPress 웹 사이트를 MySQL 관리형 데이터베이스에 연결한 후에는 다음 추가 단계를 완료해야 합니다.

- WordPress 인스턴스의 스냅샷을 생성합니다. 자세한 내용은 Linux 또는 Unix 인스턴스의 스냅샷 생성을 참조하세요.
- MySQL 관리형 데이터베이스의 스냅샷을 생성합니다. 자세한 내용은 데이터베이스의 스냅샷 생성을 참조하세요.
- MySQL 관리형 데이터베이스의 퍼블릭 모드 및 데이터 가져오기 모드를 사용 중지합니다. 자세한 내용은 데이터베이스의 퍼블릭 모드 구성 및 데이터베이스의 데이터 가져오기 모드 구성을 참조하세요.

정적 콘텐츠를 위해 WordPress 인스턴스를 Lightsail 버킷에 연결

이 자습서에서는 Amazon Lightsail 인스턴스에서 실행되는 WordPress 웹 사이트를 Lightsail 버킷에 연결하는 데 필요한 단계를 설명합니다. 버킷을 사용하여 이미지 및 첨부 파일과 같은 정적 콘텐츠를 호스팅할 수 있습니다. 이렇게 하려면 WP Offload Media Lite 플러그 인을 WordPress 웹 사이트에 설치하고 Lightsail 버킷에 연결하도록 구성해야 합니다. 플러그 인이 구성되면 WordPress 웹 사이트에 업로드하는 모든 미디어가 인스턴스의 디스크가 아닌 버킷에 자동으로 추가됩니다.

목차

- 1단계: 필수 구성 요소 완성
- 2단계: 버킷 권한 수정
- 3단계: WordPress 웹 사이트에서 WP Offload Media Lite 플러그 인 설치
- 4단계: WordPress 웹 사이트와 Lightsail 버킷 간의 연결 테스트

1단계: 필수 구성 요소 완성

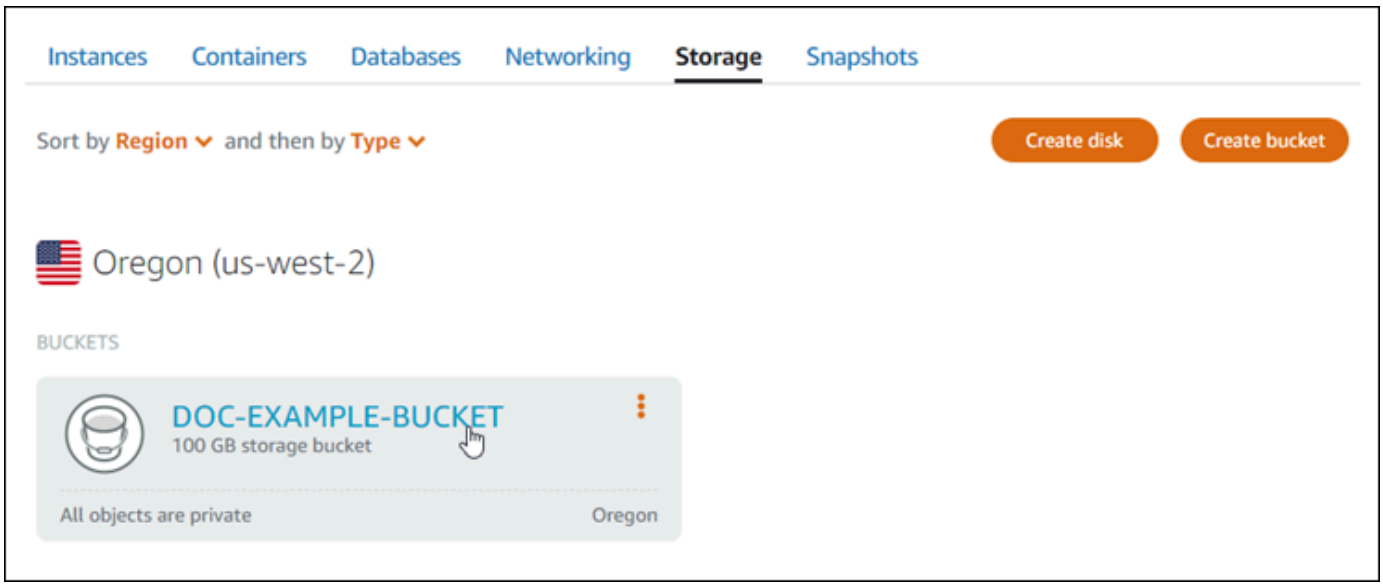
아직 수행하지 않은 경우 다음 사전 조건을 완료합니다.

- Lightsail에 WordPress 인스턴스를 생성합니다. 자세한 내용은 자습서: Amazon Lightsail에서 WordPress 인스턴스 시작 및 구성을 참조하세요.
- Lightsail 객체 스토리지 서비스에서 버킷을 생성합니다. 자세한 내용은 버킷 생성을 참조하세요.

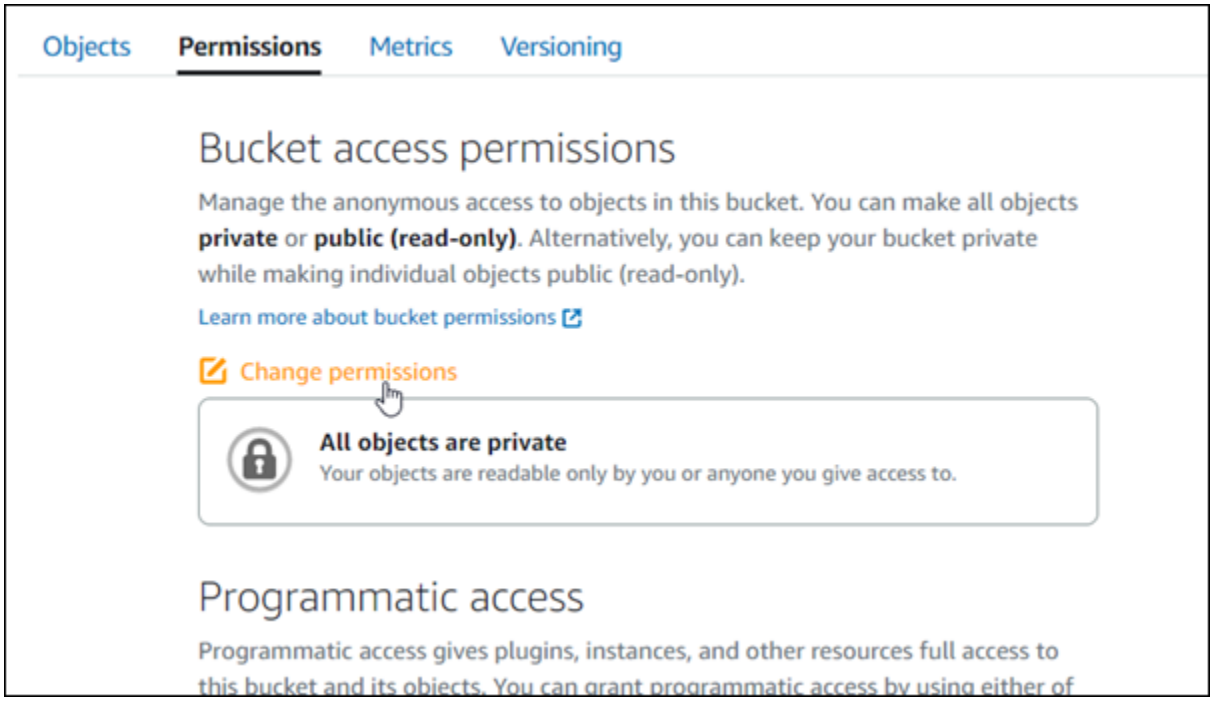
2단계: 버킷 권한 수정

버킷의 권한을 변경하여 WordPress 인스턴스 및 Offload Media Lite 플러그 인에 액세스할 권한을 부여하려면 다음 절차를 완료하세요. 버킷의 액세스 권한은 개별 객체 공개 가능(읽기 전용) (Individual objects can be made public (read-only))으로 설정되어야 합니다. 또한, 버킷의 액세스 역할에 WordPress 인스턴스를 연결해야 합니다. 버킷 권한에 대한 자세한 내용은 버킷 권한을 참조하세요.

1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 스토리지를 선택합니다.
3. WordPress 웹 사이트에서 사용하려는 버킷의 이름을 선택합니다.

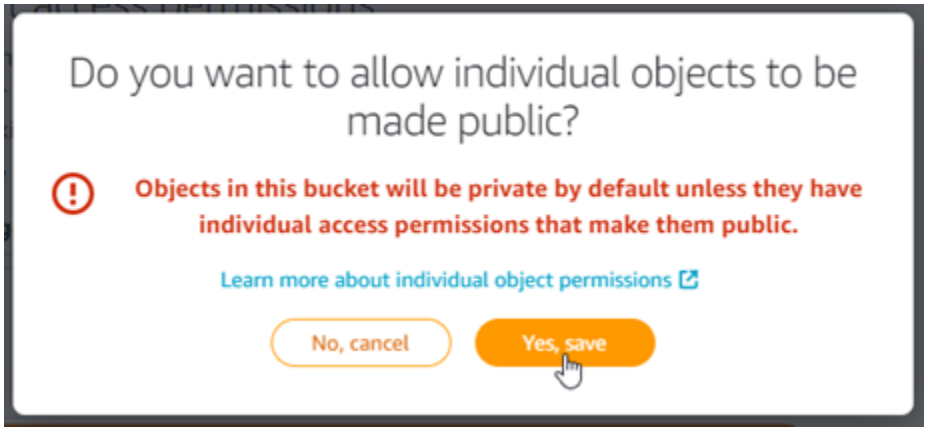


4. 버킷 관리(Bucket management) 페이지에서 권한(Permissions) 탭을 선택합니다.
5. 페이지의 버킷 액세스 권한(Bucket access permissions) 섹션에서 권한 변경(Change permissions)을 선택합니다.

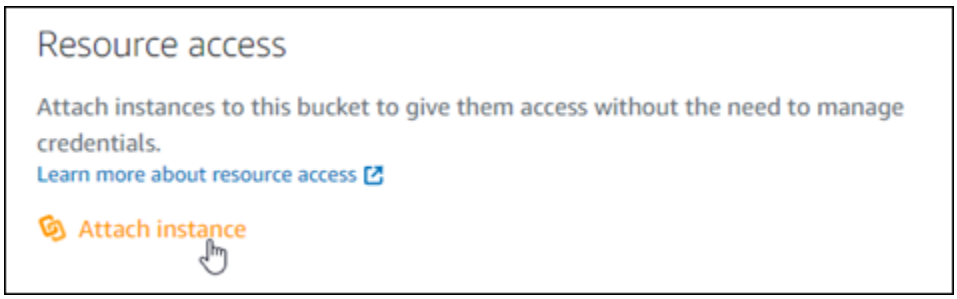


6. 개별 객체 공개 가능 및 읽기 전용(Individual objects can be made public and read only)을 선택합니다.

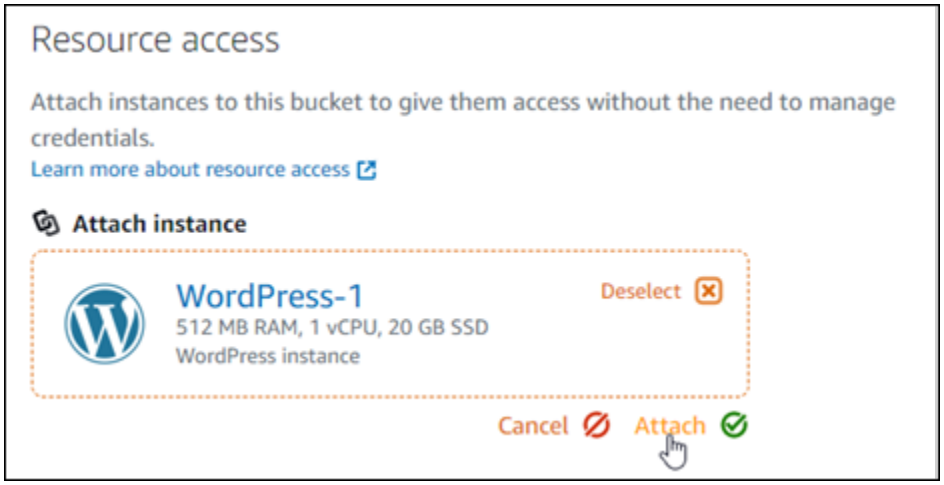
7. 저장을 선택합니다.
8. 표시되는 확인 프롬프트에서 예, 저장합니다(Yes, save)를 선택합니다.



   잠시 후 버킷이 개별 객체 액세스를 허용하도록 구성됩니다. 이렇게 하면 Offload Media Lite 플러그 인을 사용하여 WordPress 웹 사이트에서 버킷에 업로드된 객체를 고객이 읽을 수 있습니다.
9. 페이지의 리소스 액세스(Resource access) 섹션으로 스크롤하고 인스턴스 연결(Attach instance)을 선택합니다.



10. 표시되는 드롭다운 목록에서 WordPress 인스턴스의 이름을 선택한 후 연결(Attach)을 선택합니다.



    몇 분 정도 지나면 WordPress 인스턴스가 버킷에 연결됩니다. 이렇게 하면 WordPress 인스턴스에서 버킷과 해당 객체를 관리할 수 있습니다.

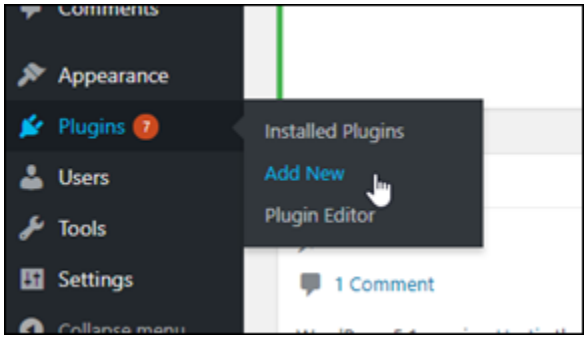
3단계: WordPress 웹 사이트에서 WP Offload Media Lite 플러그 인 설치

WP Offload Media Lite 플러그 인을 WordPress 웹 사이트에 설치하려면 다음 절차를 완료하세요. 이 플러그 인은 WordPress 미디어 업로더를 통해 추가된 이미지, 비디오, 문서 및 기타 미디어를 Lightsail 버킷에 자동으로 복사합니다. 자세한 내용은 WordPress 웹 사이트의 WP Offload Media Lite를 참조하세요.

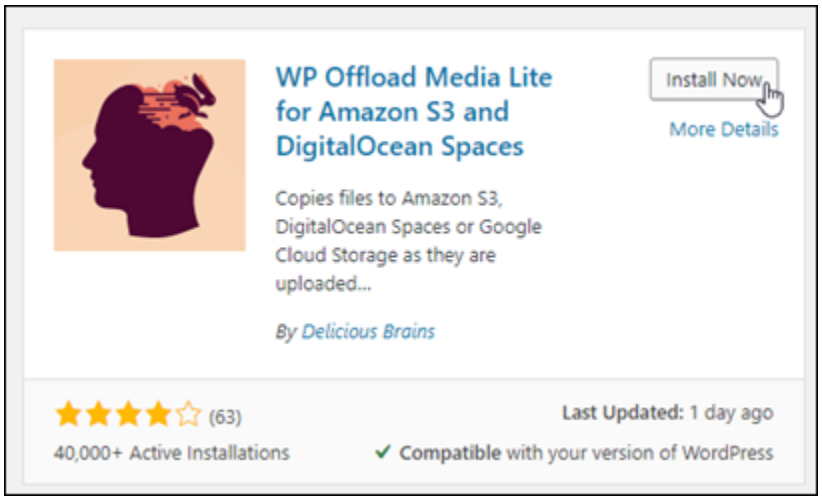
1. WordPress 웹 사이트의 대시보드에 관리자로 로그인합니다.

    자세한 내용은 Amazon Lightsail의 Bitnami 인스턴스에 대한 애플리케이션 사용자 이름과 암호 가져오기를 참조하세요.

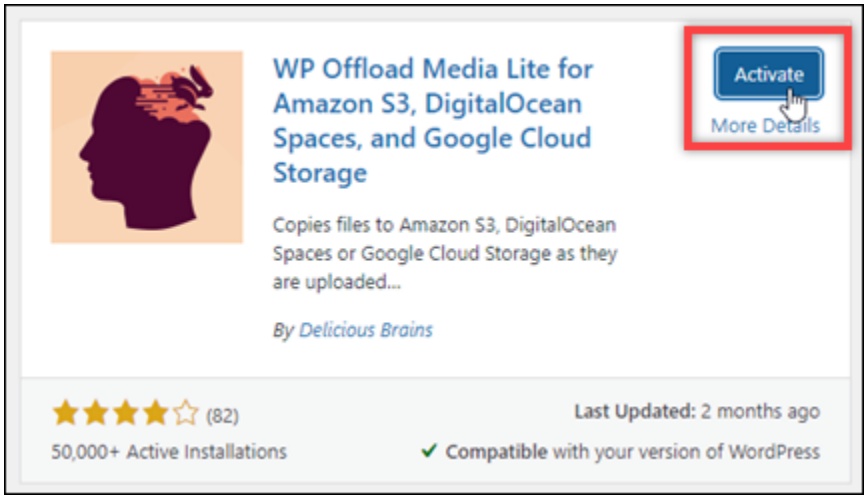
2. 왼쪽 탐색 메뉴에서 플러그 인(Plugins)을 일시 중지하고 새로 추가(Add New)를 선택합니다.



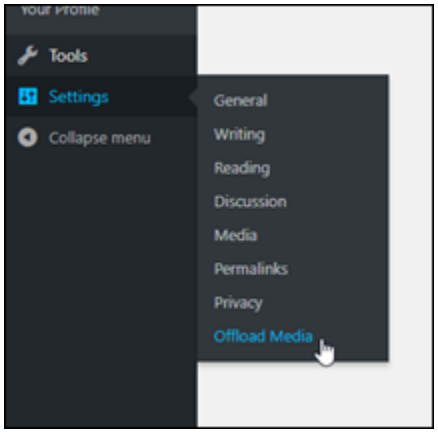
3. WP Offload Media Lite를 검색합니다.
4. 검색 결과에서 WP Offload Media 플러그 인 옆에 있는 지금 설치(Install Now)를 선택합니다.



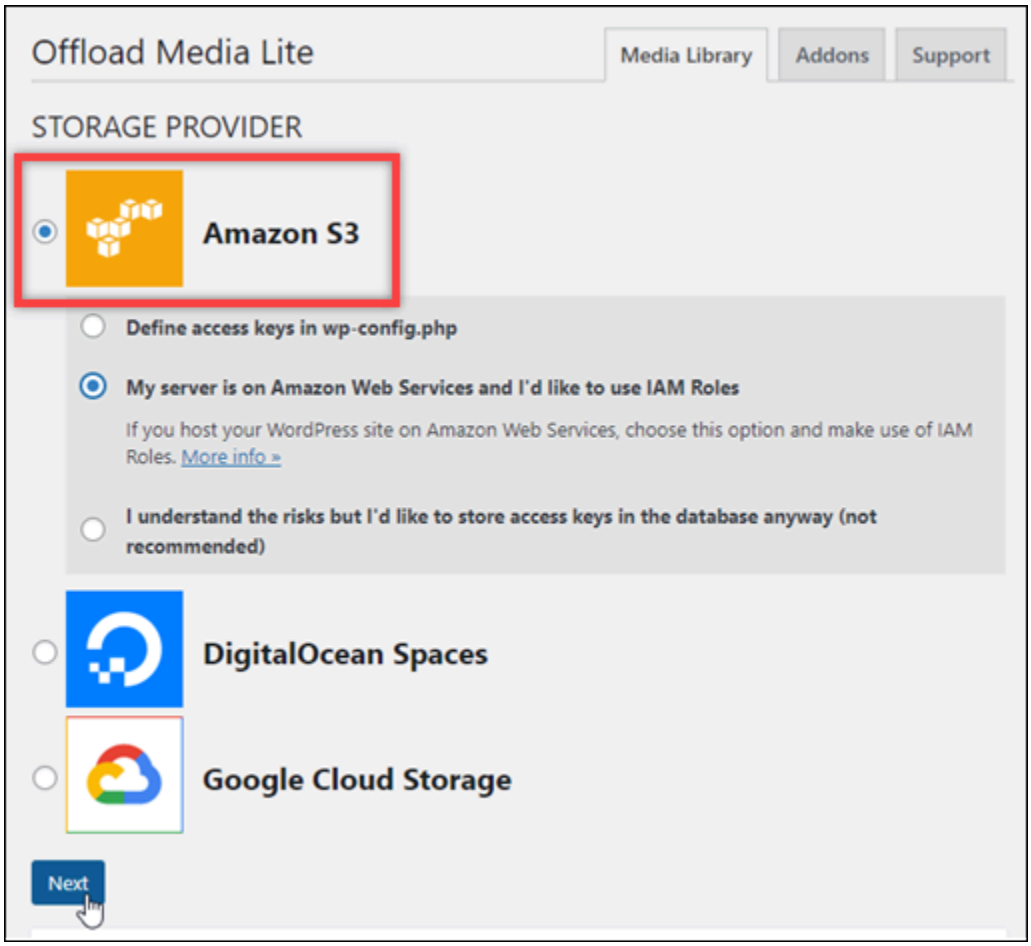
5. 플러그 인 설치가 끝나면 활성화(Activate)를 선택합니다.



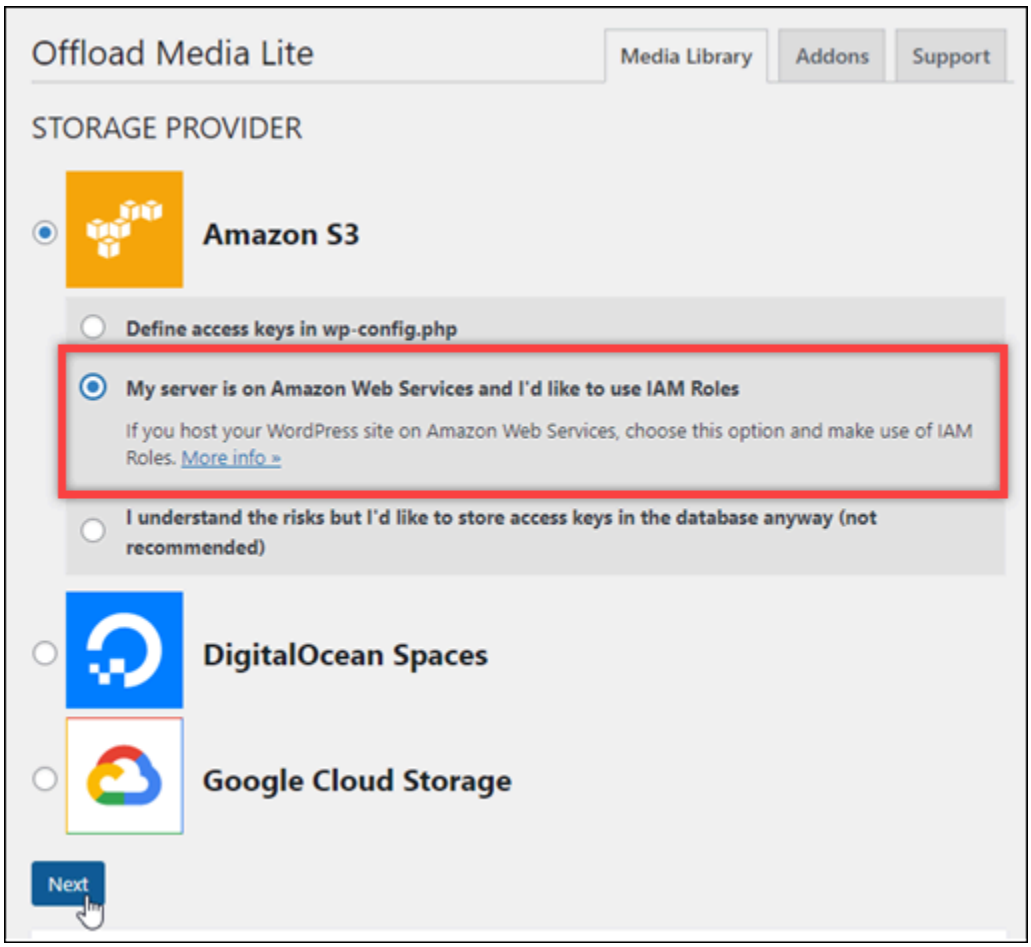
6. 왼쪽 탐색 메뉴에서 설정(Settings)을 선택한 후 Offload Media를 선택합니다.



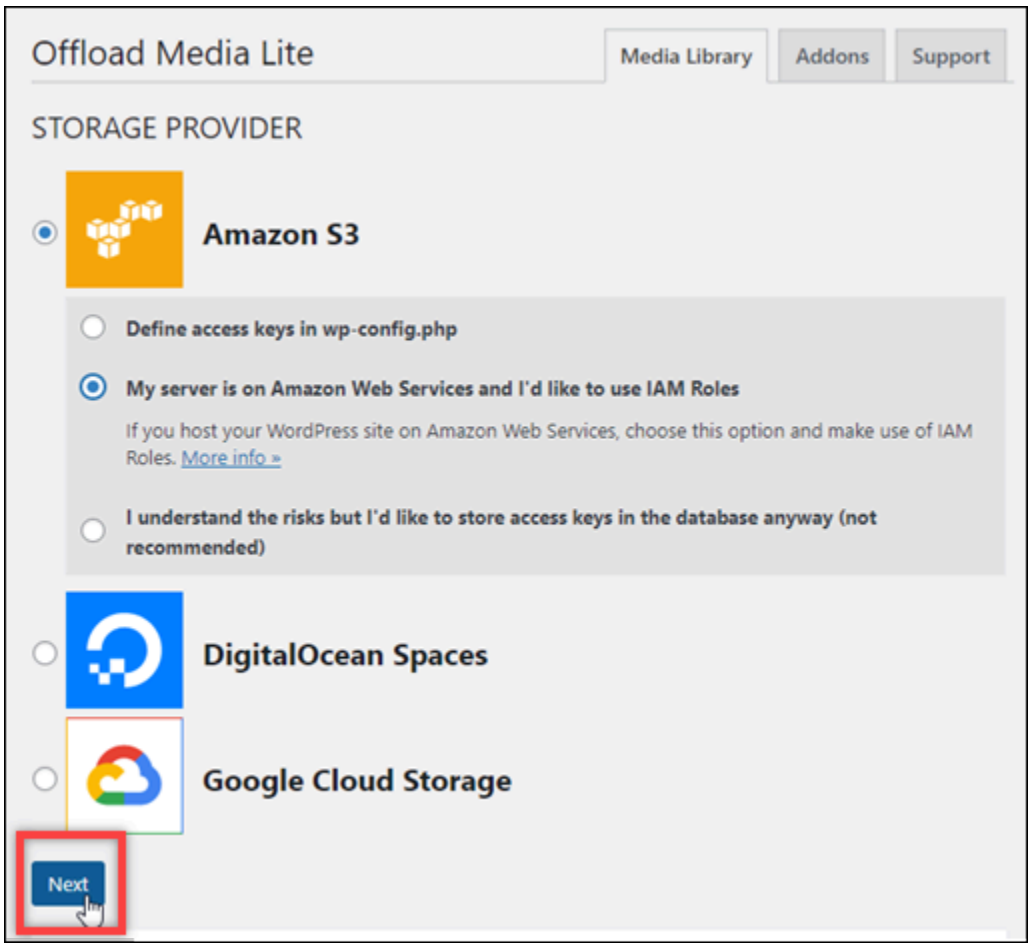
7. Offload Media 페이지에서 Amazon S3를 스토리지 공급자로 지정합니다.



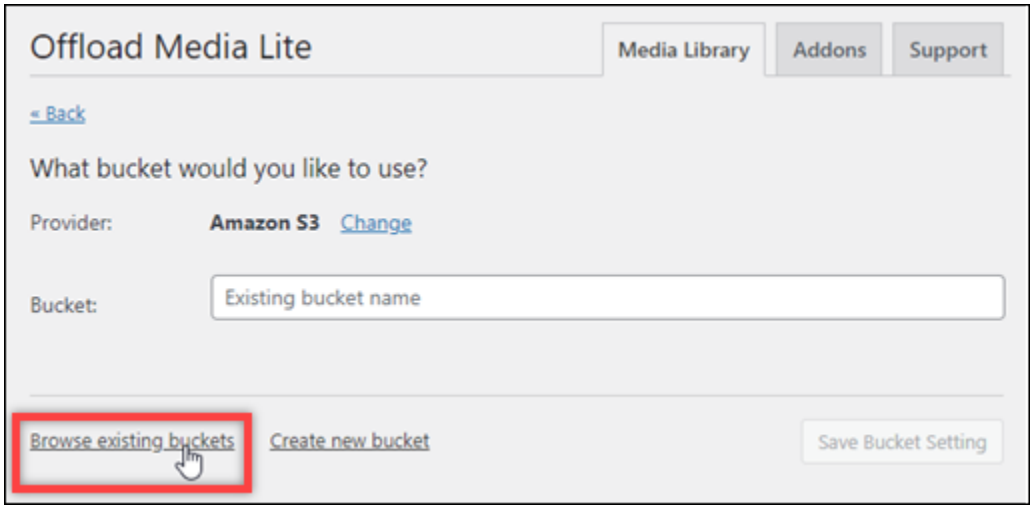
8. 서버가 Amazon Web Services에 있으며 IAM 역할을 사용하려고 합니다(My server is on Amazon Web Services and I'd like to use IAM Roles)를 선택합니다.


Offload Media Lite
Media Library
Addons
Support
STORAGE PROVIDER
Amazon S3
Define access keys in wp-config.php
My server is on Amazon Web Services and I'd like to use IAM Roles
If you host your WordPress site on Amazon Web Services, choose this option and make use of IAM Roles. More info »
I understand the risks but I'd like to store access keys in the database anyway (not recommended)
DigitalOcean Spaces
Google Cloud Storage
Next

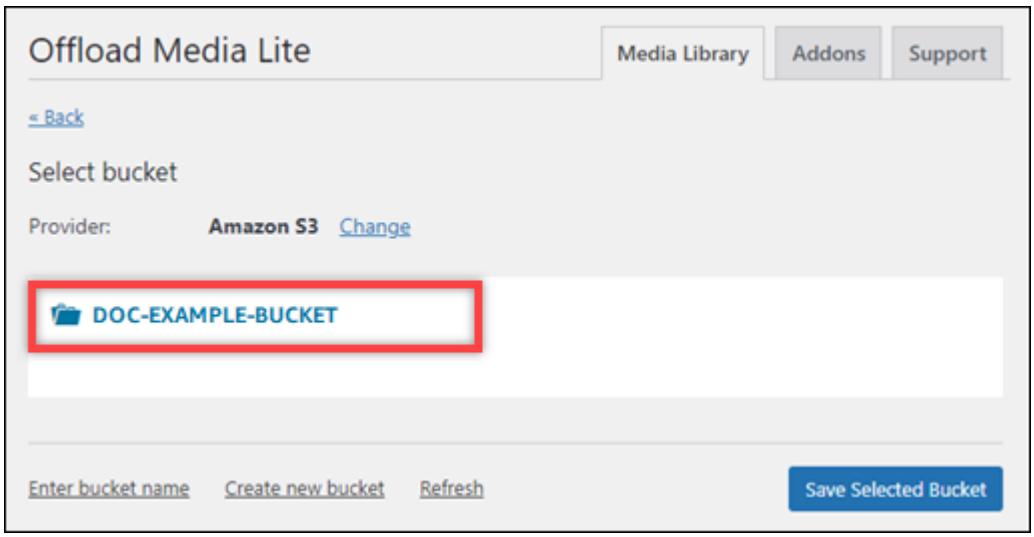
9. 다음을 선택합니다.



10. 표시되는 어느 버킷을 사용하시겠어요?(What bucket would you like to use?) 페이지에서 기존 버킷 찾아보기(Browse existing buckets)를 선택합니다.

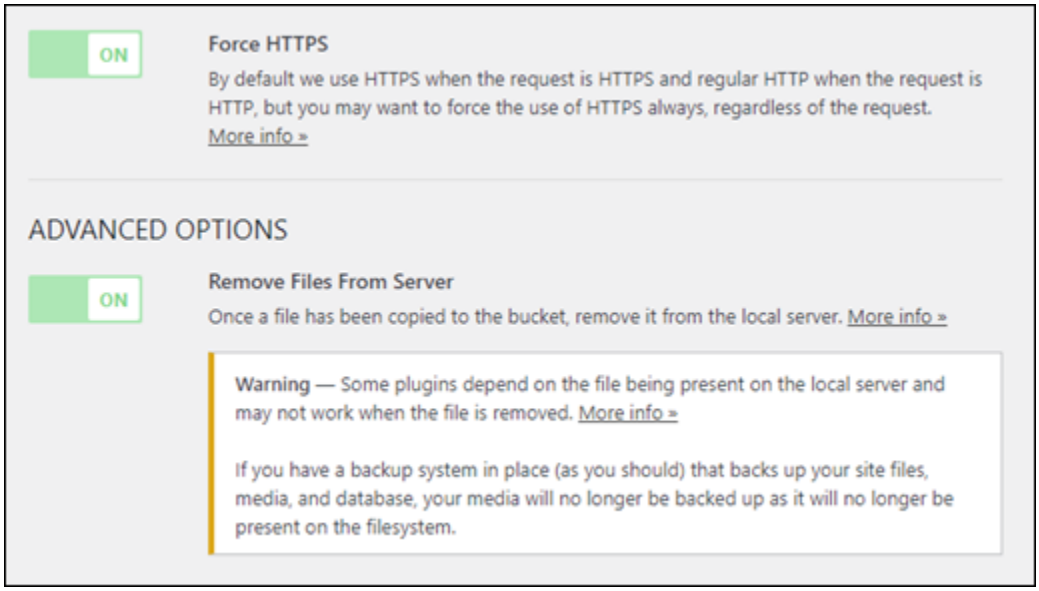


11. WordPress 인스턴스에서 사용하려는 버킷의 이름을 선택합니다.



12. Offload Media Lite 설정(Offload Media Lite Settings) 페이지가 표시되면 HTTPS 강제 실행(Force HTTPS) 및 서버에서 파일 제거(Remove Files From Server)를 선택하여 켭니다.

    - Lightsail 버킷은 기본적으로 HTTPS를 사용하여 미디어 파일을 제공하므로 HTTPS 강제 실행(Force HTTPS) 설정을 켜야 합니다. 이 기능을 켜지 않으면 WordPress 웹 사이트에서 Lightsail 버킷에 업로드된 미디어 파일이 웹 사이트 방문자에게 올바르게 제공되지 않습니다.
    - 서버에서 파일 제거(Remove Files From Server) 설정을 사용하면 Lightsail 버킷에 업로드된 미디어가 인스턴스의 디스크에도 저장되지 않습니다. 이 기능을 켜지 않으면 Lightsail 버킷에 업로드된 미디어 파일이 WordPress 인스턴스의 로컬 스토리지에도 저장됩니다.



13. 변경 사항 저장(Save Changes)을 선택합니다.

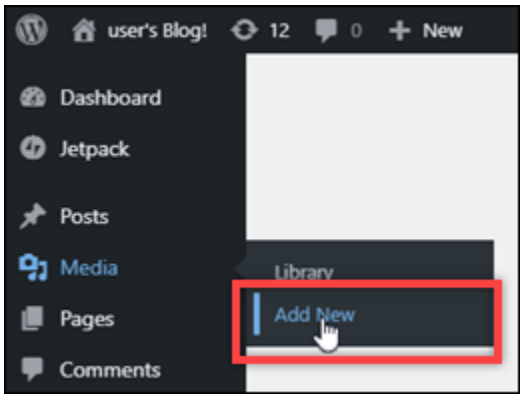
> **Note**
>
> 나중에 Offload Media Lite 설정(Offload Media Lite Settings) 페이지로 돌아가려면 왼쪽 탐색 메뉴에서 설정(Settings)을 일시 중지하고 Offload Media Lite를 선택하면 됩니다.

이제 WordPress 웹 사이트가 Media Lite 플러그 인을 사용하도록 구성되었습니다. 다음에 WordPress를 통해 미디어 파일을 업로드하면 해당 파일이 Lightsail 버킷에 자동으로 업로드되어 버킷에서 제공됩니다. 구성을 테스트하려면 이 자습서의 다음 섹션을 계속 진행합니다.

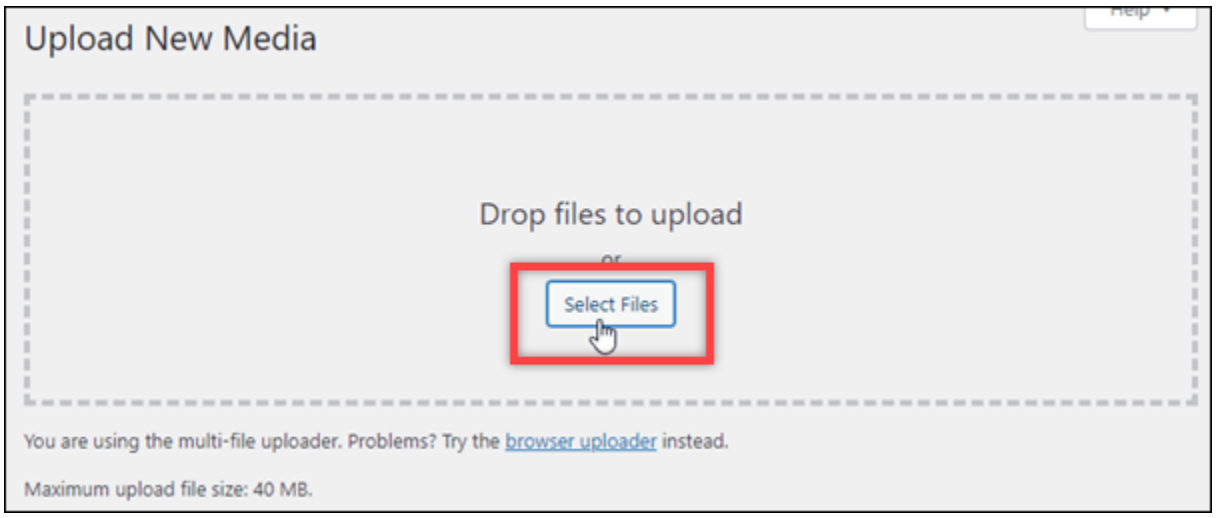
4단계: WordPress 웹 사이트와 Lightsail 버킷 간의 연결 테스트

다음 절차를 완료하여 WordPress 인스턴스에 미디어 파일을 업로드하고 Lightsail 버킷에 업로드되어 제공되고 있는지 확인합니다.

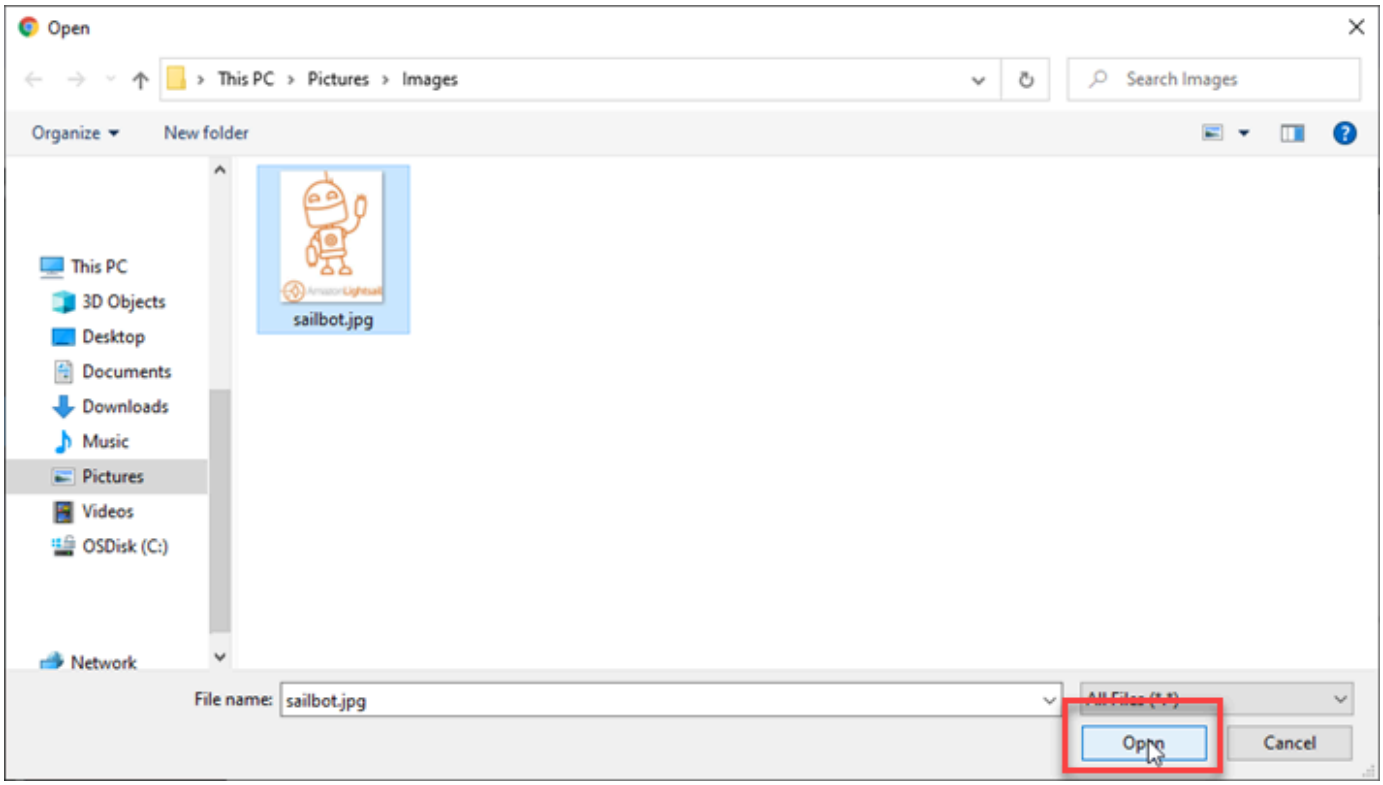
1. WordPress 대시보드의 왼쪽 탐색 메뉴에서 미디어(Media)를 일시 중지하고 새로 추가(Add New)를 선택합니다.



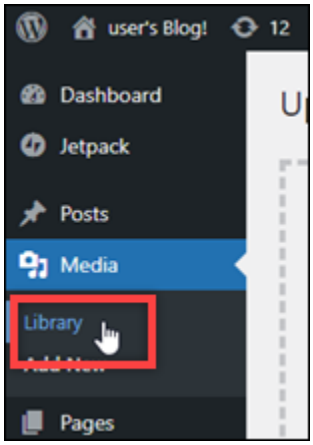
2. 표시되는 새 미디어 업로드(Upload New Media) 페이지에서 파일 선택(Select Files)을 선택합니다.



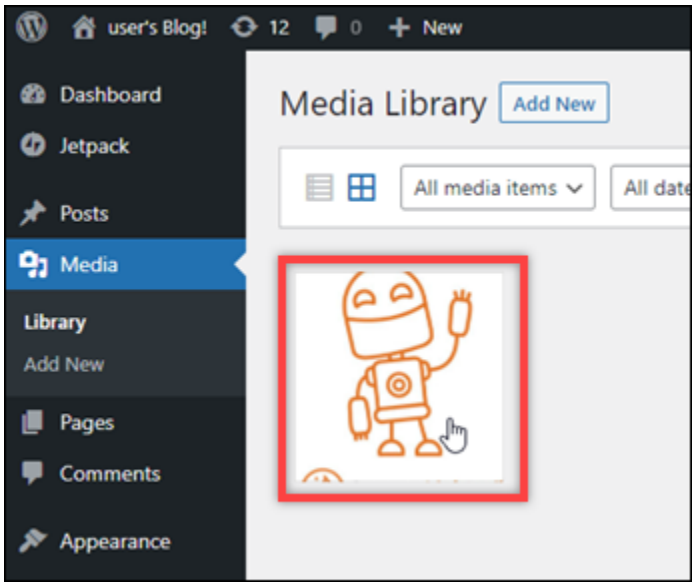
3. 로컬 컴퓨터에서 업로드할 미디어 파일을 선택하고 열기(Open)를 선택합니다.



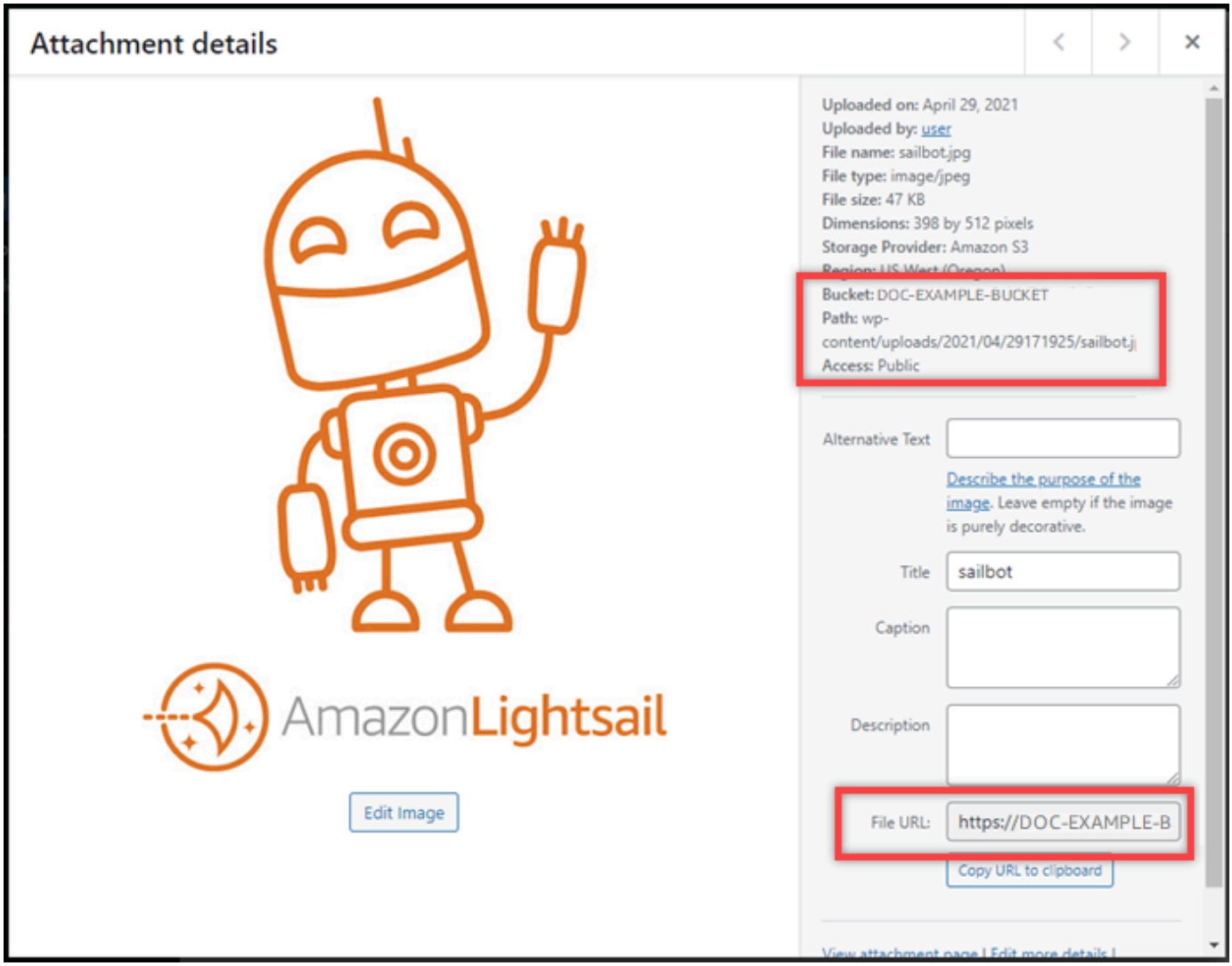
4. 파일 업로드가 완료되면 왼쪽 탐색 메뉴에서 미디어(Media) 아래의 라이브러리(Library)를 선택합니다.



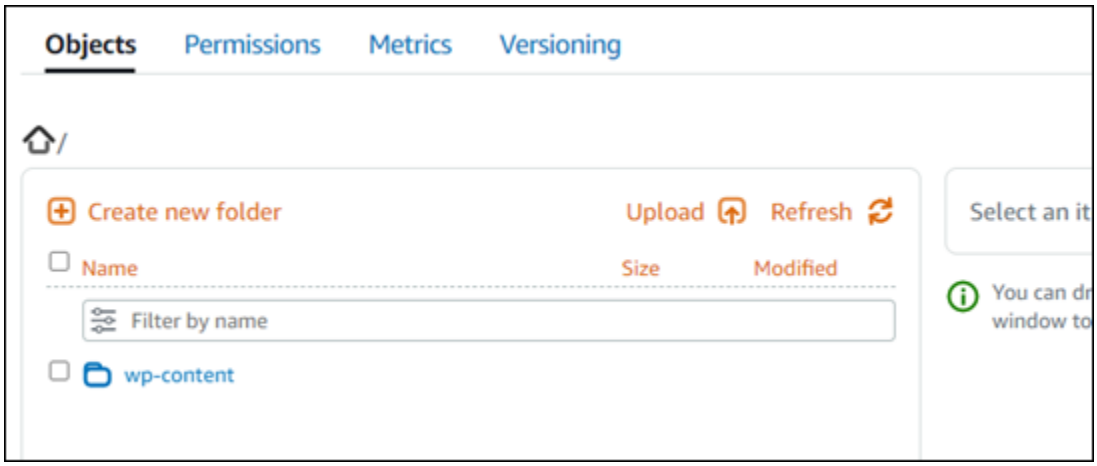
5. 최근에 업로드한 파일을 선택합니다.



6. 파일의 세부 정보 패널에 있는 버킷(Bucket) 및 파일 URL(File URL) 필드에 버킷 이름이 표시됩니다.



7. Lightsail 버킷 관리 페이지의 객체(Objects) 탭으로 이동하면 wp-content 폴더가 표시됩니다. 이 폴더는 Offload Media Lite 플러그 인에 의해 생성되며 업로드된 미디어 파일을 저장하는 데 사용됩니다.



버킷 및 객체 관리

다음은 Lightsail 객체 스토리지 버킷을 관리하는 일반적인 단계입니다.

1. Amazon Lightsail 객체 스토리지 서비스의 객체와 버킷에 대해 알아보세요. 자세한 내용은 Amazon Lightsail의 객체 스토리지를 참조하세요.
2. Amazon Lightsail에서 버킷에 지정할 수 있는 이름에 대해 알아봅니다. 자세한 내용은 Amazon Lightsail의 버킷 이름 지정 규칙을 참조하세요.
3. 버킷을 생성하여 Lightsail 객체 스토리지 서비스를 시작합니다. 자세한 내용은 Amazon Lightsail에서 버킷 생성을 참조하세요.
4. 버킷의 보안 모범 사례와 버킷에 구성할 수 있는 액세스 권한에 대해 알아봅니다. 버킷의 모든 객체를 공개 또는 비공개로 설정하거나 개별 객체를 공개로 설정하도록 선택할 수 있습니다. 액세스 키를 생성하고, 버킷에 인스턴스를 연결하고, 다른 AWS 계정에 액세스 권한을 부여하여 버킷에 대한 액세스 권한을 부여할 수도 있습니다. 자세한 내용은 Amazon Lightsail 객체 스토리지에 대한 보안 모범 사례 및 Amazon Lightsail의 버킷 권한 이해를 참조하세요.

   버킷 액세스 권한에 대해 알아본 후 다음 가이드를 참조하여 버킷에 대한 액세스 권한을 부여합니다.

   - Amazon Lightsail의 버킷에 대한 퍼블릭 액세스 차단
   - Amazon Lightsail에서 버킷 액세스 권한 구성
   - Amazon Lightsail에서 버킷의 개별 객체에 대한 액세스 권한 구성
   - Amazon Lightsail에서 버킷의 액세스 키 생성
   - Amazon Lightsail에서 버킷에 대한 리소스 액세스 구성
   - Amazon Lightsail에서 버킷의 크로스 계정 액세스 구성
5. 버킷에 대한 액세스 로깅을 활성화하는 방법과 액세스 로그를 사용하여 버킷의 보안을 감사하는 방법에 대해 알아봅니다. 자세한 내용은 다음 안내서를 참조하세요.
   - Amazon Lightsail 객체 스토리지 서비스의 버킷에 대한 액세스 로깅
   - Amazon Lightsail 객체 스토리지 서비스에서 버킷에 대한 액세스 로그 형식
   - Amazon Lightsail 객체 스토리지 서비스에서 버킷에 대한 액세스 로깅 활성화
   - Amazon Lightsail에서 버킷에 대한 액세스 로그 사용하여 요청을 식별
6. Lightsail에서 버킷을 관리할 수 있는 기능을 사용자에게 부여하는 IAM 정책을 생성합니다. 자세한 내용은 Amazon Lightsail에서 버킷을 관리하기 위한 IAM 정책을 참조하세요.
7. 버킷의 객체에 라벨을 지정하고 식별하는 방식에 대해 알아봅니다. 자세한 내용은 Amazon Lightsail에서 객체 키 이름 이해를 참조하세요.

8. 파일을 업로드하고 버킷의 객체를 관리하는 방법에 대해 알아봅니다. 자세한 내용은 다음 안내서를 참조하세요.
   - Amazon Lightsail에서 버킷으로 파일 업로드
   - 멀티파트 업로드를 사용하여 Amazon Lightsail에서 버킷으로 파일 업로드
   - Amazon Lightsail에서 버킷의 객체 확인
   - Amazon Lightsail에서 버킷의 객체 복사 또는 이동
   - Amazon Lightsail에서 버킷의 객체 다운로드
   - Amazon Lightsail에서 버킷의 객체 필터링
   - Amazon Lightsail에서 버킷의 객체 태그 지정
   - Amazon Lightsail에서 버킷의 객체 삭제
9. 객체 버전 관리를 활성화하면 버킷에 저장된 모든 버전의 객체를 모두 보존, 검색 및 복원할 수 있습니다. 자세한 내용은 Amazon Lightsail에서 버킷의 객체 버전 사용 설정 및 사용 중지를 참조하세요.
10. 객체 버전 관리를 활성화한 후 버킷에 있는 객체의 이전 버전을 복원할 수 있습니다. 자세한 내용은 Amazon Lightsail에서 버킷의 이전 객체 버전 복원을 참조하세요.
11. 버킷 사용률을 모니터링합니다. 자세한 내용은 Amazon Lightsail에서 버킷에 대한 지표 보기를 참조하세요.
12. 버킷 사용률이 임곗값을 초과할 때 알림을 받도록 버킷 지표에 대한 경보를 구성합니다. 자세한 내용은 Amazon Lightsail에서 버킷 지표 경보 생성을 참조하세요.
13. 스토리지 및 네트워크 전송이 부족한 경우 버킷의 스토리지 플랜을 변경합니다. 자세한 내용은 Amazon Lightsail에서 버킷 플랜 변경을 참조하세요.
14. 버킷을 다른 리소스에 연결하는 방법에 대해 알아봅니다. 자세한 내용은 다음 자습서를 참조하세요.
   - 자습서: WordPress 인스턴스를 Amazon Lightsail 버킷에 연결
   - 자습서: Lightsail 콘텐츠 전송 네트워크 배포와 함께 Amazon Lightsail 버킷 사용
15. 버킷을 더 이상 사용하지 않는 경우 삭제할 수 있습니다. 자세한 내용은 Amazon Lightsail에서 버킷 삭제를 참조하세요.

Lightsail 콘텐츠 전송 네트워크를 사용하여 WordPress 구성

이 가이드에서는 Amazon Lightsail 배포와 함께 사용할 수 있도록 WordPress 인스턴스를 구성하는 방법을 안내합니다.

모든 Lightsail 배포는 기본 도메인(예: `123456abcdef.cloudfront.net`)에 대해 HTTPS를 사용하도록 기본적으로 설정되어 있습니다. 배포 구성에 따라 배포와 인스턴스 간의 연결이 암호화되는지 여부가 결정됩니다.

- WordPress 웹 사이트에서 HTTP만 사용 - 웹 사이트에서 배포의 오리진으로 HTTP만 사용하고 HTTPS를 사용하도록 구성되지 않은 경우 SSL/TLS를 종료하고 암호화되지 않은 연결을 사용하여 모든 콘텐츠 요청을 인스턴스로 전달하도록 배포를 구성할 수 있습니다.
- WordPress 웹 사이트에서는 HTTP만 사용 - 웹 사이트가 HTTPS를 배포의 오리진으로 사용할 때 암호화된 연결을 사용하여 모든 콘텐츠 요청을 인스턴스로 전달하도록 배포를 구성할 수 있습니다. 이 구성을 엔드 투 엔드 암호화라고 합니다.

배포 생성

다음 단계를 완료하여 WordPress 인스턴스에 대한 Lightsail 배포를 구성합니다. 자세한 내용은 the section called “배포 생성” 단원을 참조하십시오.

전제 조건

the section called “WordPress”에 설명된 대로 WordPress 인스턴스를 생성하고 구성합니다.

WordPress 인스턴스에 대한 배포 생성

1. 왼쪽 탐색 창에서 네트워킹을 선택합니다.
2. 배포 생성을 선택합니다.
3. 오리진 선택에서 WordPress 인스턴스를 실행하는 리전을 선택한 다음 WordPress 인스턴스를 선택합니다. 인스턴스에 연결한 정적 IP 주소를 자동으로 사용합니다.
4. 캐싱 동작에서 WordPress에 가장 적합을 선택합니다.
5. (선택 사항) 종단 간 암호화를 구성하려면 오리진 프로토콜 정책을 HTTPS 전용으로 변경합니다. 자세한 내용은 the section called “오리진 프로토콜 정책” 섹션을 참조하세요.
6. 나머지 옵션을 구성하고 데이터베이스 생성을 선택합니다.
7. 사용자 지정 도메인 탭에서 인증서 생성을 선택합니다. 인증서의 고유한 이름을 입력하고 도메인 및 하위 도메인의 이름을 입력한 다음 인증서 생성을 선택합니다.
8. 인증서 연결(Attach certificate)을 선택합니다.
9. DNS 레코드 업데이트에서 이해함을 선택합니다.

DNS 레코드 업데이트

다음 단계를 완료하여 Lightsail DNS 영역의 DNS 레코드를 업데이트합니다.

배포에 대한 DNS 레코드 업데이트

1. 왼쪽 탐색 창에서 도메인 및 DNS를 선택합니다.
2. DNS 영역을 선택한 다음 DNS 레코드 탭을 선택합니다.
3. 인증서에 지정한 도메인의 A 및 AAAA 레코드를 삭제합니다.
4. 레코드 추가를 선택하고 배포를 위해 도메인을 도메인으로 확인하는 CNAME 레코드를 생성합니다(예: d2vbec9EXAMPLE.cloudfront.net).
5. 저장을 선택합니다.

배포에서 정적 콘텐츠를 캐싱하도록 허용

배포와 함께 사용할 수 있도록 WordPress 인스턴스의 `wp-config.php` 파일을 편집하려면 다음 절차를 완료하세요.

> **Note**
>
> 이 절차를 시작하기 전에 WordPress 인스턴스의 스냅샷을 생성하는 것이 좋습니다. 스냅샷은 문제가 발생할 경우 다른 인스턴스를 생성할 백업으로 사용할 수 있습니다. 자세한 내용은 Linux 또는 Unix 인스턴스의 스냅샷 생성을 참조하세요.

1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 WordPress 인스턴스 옆에 표시되는 브라우저 기반 SSH 클라이언트 아이콘을 선택합니다.
3. 인스턴스에 연결한 후 다음 명령을 입력하여 `wp-config.php` 파일의 백업을 생성합니다. 문제가 발생할 경우 백업을 사용하여 파일을 복원할 수 있습니다.

```
sudo cp /opt/bitnami/wordpress/wp-config.php /opt/bitnami/wordpress/wp-
config.php.backup
```

4. 다음 명령을 입력하여 Vim을 통해 `wp-config.php` 파일을 엽니다.

```
sudo vim /opt/bitnami/wordpress/wp-config.php
```

5. I를 눌러 Vim을 삽입 모드를 설정합니다.
6. 파일에서 다음 코드 행을 삭제합니다.

```
define('WP_SITEURL', 'http://' . $_SERVER['HTTP_HOST'] . '/');
define('WP_HOME', 'http://' . $_SERVER['HTTP_HOST'] . '/');
```

7. 사용 중인 WordPress 버전에 따라 파일에 다음 행의 코드를 추가합니다.

   - 버전 3.3 이하를 사용하는 경우 이전에 코드를 삭제한 파일에 다음 행의 코드를 추가합니다.

```
define('WP_SITEURL', 'https://' . $_SERVER['HTTP_HOST'] . '/');
define('WP_HOME', 'https://' . $_SERVER['HTTP_HOST'] . '/');
if (isset($_SERVER['HTTP_CLOUDFRONT_FORWARDED_PROTO'])
&& $_SERVER['HTTP_CLOUDFRONT_FORWARDED_PROTO'] === 'https') {
$_SERVER['HTTPS'] = 'on';
}
```

   - 버전 3.3.1~5 또는 그 이상을 사용하는 경우 이전에 코드를 삭제한 파일에 다음 행의 코드를 추가합니다.

```
define('WP_SITEURL', 'http://DOMAIN/');
define('WP_HOME', 'http://DOMAIN/');
if (isset($_SERVER['HTTP_CLOUDFRONT_FORWARDED_PROTO'])
&& $_SERVER['HTTP_CLOUDFRONT_FORWARDED_PROTO'] === 'https') {
$_SERVER['HTTPS'] = 'on';
}
```

8. Esc 키를 눌러 Vim에서 삽입 모드를 종료한 다음 `:wq!`를 입력하고 Enter 키를 눌러 편집한 내용을 저장(쓰기)하고 Vim을 종료합니다.
9. 다음 명령을 입력하여 인스턴스에서 Apache 서비스를 다시 시작합니다.

```
sudo /opt/bitnami/ctlscript.sh restart apache
```

10. Apache 서비스가 다시 시작될 때까지 잠시 기다린 후 배포에서 콘텐츠를 캐시하고 있는지 테스트합니다. 자세한 내용은 Amazon Lightsail 배포 테스트를 참조하세요.
11. 문제가 발생한 경우 브라우저 기반 SSH 클라이언트를 사용하여 인스턴스에 다시 연결합니다. 다음 명령을 실행하여 이 가이드의 앞부분에서 생성한 백업을 통해 `wp-config.php` 파일을 복원합니다.

```
sudo cp /opt/bitnami/wordpress/wp-config.php.backup /opt/bitnami/wordpress/wp-
config.php
```

파일을 복원한 후 다음 명령을 입력하여 Apache 서비스를 다시 시작합니다.

```
sudo /opt/bitnami/ctlscript.sh restart apache
```

배포에 대한 추가 정보

다음은 Lightsail에서 배포를 관리하는 데 도움이 되는 관련 문서입니다.

- 콘텐츠 전송 네트워크 배포
- 배포 생성
- 배포의 요청 및 응답 동작 이해
- 배포 테스트
- 배포의 오리진 변경
- 배포의 캐싱 동작 변경
- 배포의 캐시 재설정
- 배포 플랜 변경하기
- 배포용 사용자 지정 도메인 활성화
- 도메인을 배포로 연결하기
- 배포용 사용자 지정 도메인 변경
- 배포용 사용자 지정 도메인 사용 중지
- 배포 지표 확인
- 배포 삭제

Lightsail의 WordPress 인스턴스에서 이메일 활성화

Amazon Lightsail의 WordPress 인스턴스에서 이메일을 활성화할 수 있습니다. Amazon Simple Email Service(Amazon SES)에서 SMTP 서비스를 구성합니다. 그런 다음, 인스턴스에서 WP Mail SMTP 플러그인을 활성화 및 구성합니다. 이메일이 활성화되고 나면 WordPress 관리자는 사용자 프로필에 대해 암호 재설정을 요청할 수 있으며, 블로그 게시물, 웹사이트 업데이트 및 기타 플러그인 메시지

에 대해 이메일 알림을 받게 됩니다. 이 설명서에서는 Amazon SES를 사용하여 Amazon Lightsail의 WordPress 인스턴스에서 이메일을 활성화하는 방법을 보여줍니다.

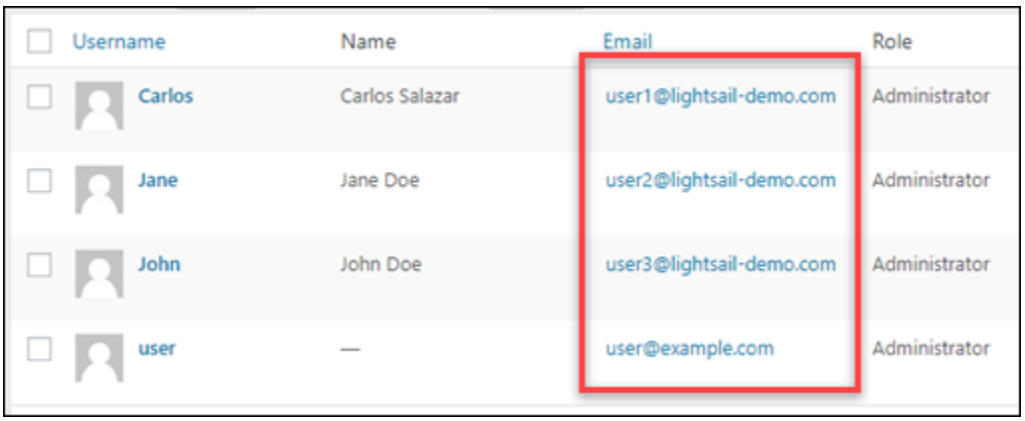
목차

- 1단계: 제한 검토
- 2단계: 사전 조건 완료
- 3단계: Amazon SES에서 SMTP 보안 인증 생성
- 4단계: Amazon SES에서 도메인 확인
- 5단계: Amazon SES에서 이메일 주소 확인
- 6단계: WordPress 인스턴스에서 WP Mail SMTP 플러그인 구성

자세한 내용은 Amazon SES 설명서의 Amazon SES SMTP 인터페이스를 사용하여 이메일 전송을 참조하세요.

1단계: 제한 검토

Amazon SES 샌드박스에 있는 새로운 Amazon Web Services(AWS) 계정은 확인된 주소 및 도메인으로만 이메일을 전송할 수 있습니다. 새 계정이 사용자의 계정인 경우에는 웹사이트의 도메인을 확인하고 WordPress 관리자의 이메일 주소를 확인하는 것이 좋습니다. 이메일 주소를 가져오려면 WordPress 웹사이트의 대시보드에 로그인하고 왼쪽 탐색 메뉴에서 사용자를 선택합니다. 다음 예제에서와 같이 이메일 열에 관리자 이메일 주소가 나열될 것입니다.



> **Note**
>
> 기본 `user` 프로필은 `user@example.com` 이메일 주소로 구성됩니다. 작동 중인 이메일 주소로 이를 변경해야 합니다. 자세한 내용은 WordPress 설명서의 사용자 프로필 화면을 참조하십시오.

어떤 주소 및 도메인으로든 이메일을 전송하려면 Amazon SES 샌드박스에서 계정을 가져오도록 요청해야 합니다. 자세한 내용은 Amazon SES 설명서의 Amazon SES 샌드박스 환경에서 나가기를 참조하세요.

2단계: 사전 조건 완료

WordPress 인스턴스에서 이메일을 활성화할 수 있으려면 먼저 이러한 작업들을 완료해야 합니다.

- Lightsail에 WordPress 인스턴스를 생성합니다. 자세한 내용은 자습서: Amazon Lightsail에서 WordPress 인스턴스 시작 및 구성을 참조하세요.
- Lightsail DNS 영역을 사용하여 WordPress 인스턴스에 대한 등록된 도메인을 지정합니다. 자세한 내용은 DNS 영역을 생성하여 도메인의 DNS 레코드 관리를 참조하세요.
- Amazon SES에 등록하여 서비스에 대해 자세히 알아보세요. Amazon SES에 가입하는 방법에 대한 자세한 내용은 Amazon SES 설명서의 Amazon SES 빠른 시작을 참조하세요. Amazon SES에 대한 자세한 내용은 다음 Amazon SES 설명서를 참조하세요.
  - Amazon SES 개발자 안내서
  - Amazon SES FAQ
  - Amazon SES 요금
  - Amazon SES Service Quotas

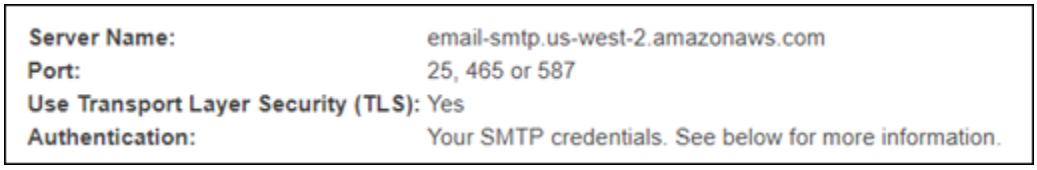
3단계: Amazon SES에서 SMTP 보안 인증 생성

WP Mail SMTP 플러그인을 구성하려면(이 설명서의 뒷부분에 설명) Amazon SES 계정에서 SMTP 보안 인증을 생성해야 합니다. 자세한 내용은 Amazon SES 설명서의 Amazon SES SMTP 보안 인증 받기를 참조하세요.

Amazon SES에서 SMTP 보안 인증 생성

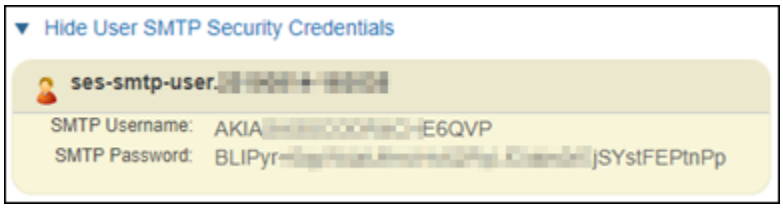
1. Amazon SES 콘솔에 로그인합니다.
2. 왼쪽 탐색 메뉴에서 SMTP settings(SMTP 설정)를 선택합니다.

SMTP settings(SMTP 설정) 페이지에 SMTP 서버 이름, 포트 및 TLS 설정이 표시됩니다. WordPress 인스턴스 상에서 WP Mail SMTP 플러그인을 구성할 때(이 안내서 뒷부분에 설명) 필요하기 때문에 이들 값을 적어둡니다.



3. SMTP 보안 인증 정보 생성을 선택합니다.
4. IAM 사용자 이름 텍스트 상자에서 기본 사용자 이름을 그대로 두고 생성을 선택합니다.

5. Show User SMTP Security Credentials(사용자 SMTP 보안 자격 증명 표시)를 선택하여 SMTP 사용자 이름 및 암호를 확인하거나, Download Credentials(자격 증명 다운로드)를 선택하여 동일한 정보가 포함된 CSV 파일을 다운로드합니다. WordPress 인스턴스 상에서 WP Mail SMTP 플러그인을 구성할 때 이러한 자격 증명이 필요합니다.



> **Note**
>
> Amazon SES 콘솔에서 생성된 보안 인증은 계정의 AWS Identity and Access Management (IAM)에 자동 추가됩니다.

4단계: Amazon SES에서 도메인 확인

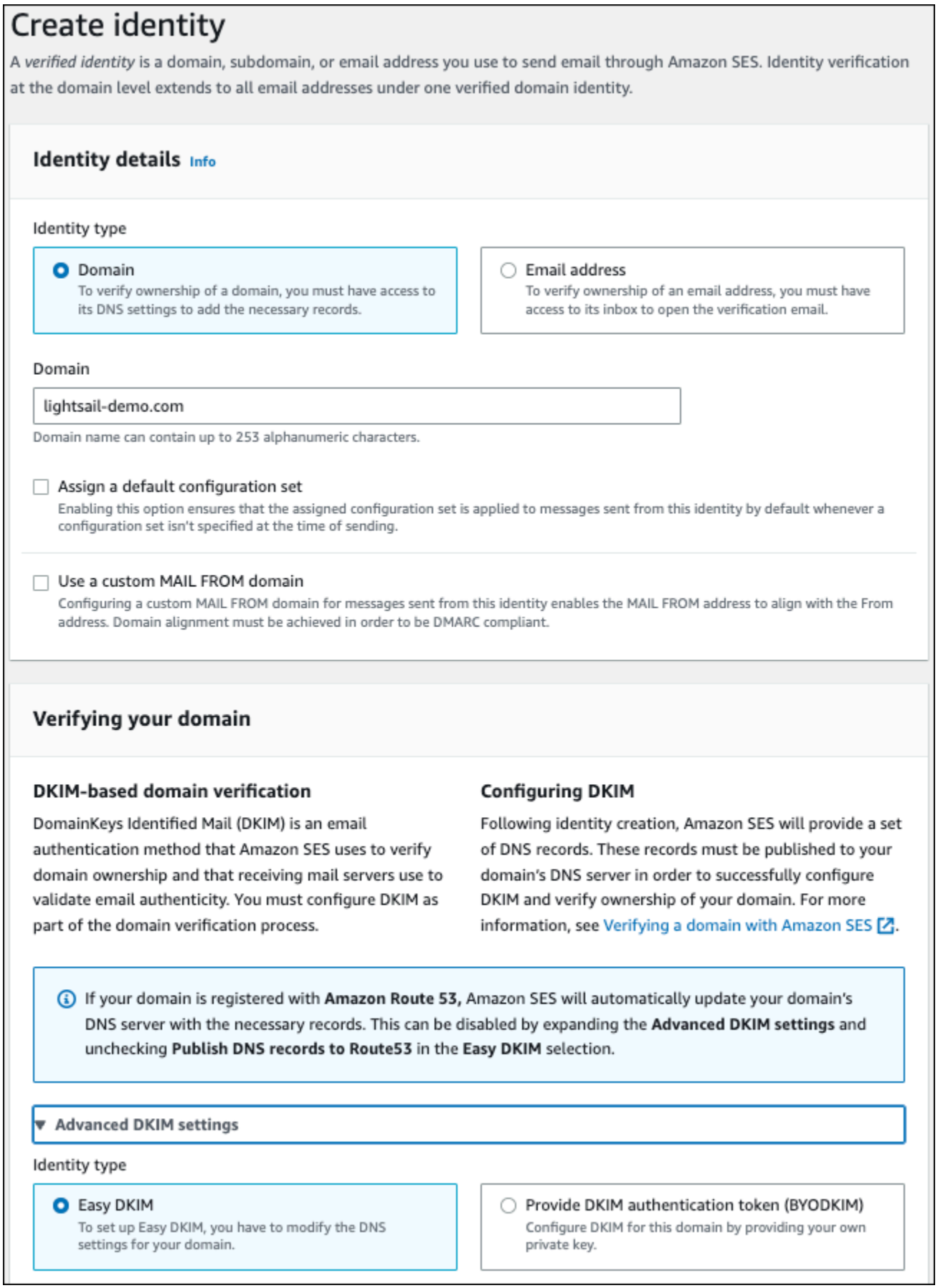
Amazon SES에서는 도메인이 본인의 소유인지 확인하도록 하여 다른 사용자의 도용을 방지해야 합니다. 도메인을 확인할 때 해당 도메인의 모든 이메일 주소를 확인하게 되므로 해당 도메인의 이메일 주소를 개별적으로 확인할 필요가 없습니다. 예를 들어 도메인 `example.com`를 확인하는 경우, `user1@example.com`, `user2@example.com` 또는 `example.com`의 기타 모든 사용자로부터 이메일을 전송할 수 있습니다. 자세한 내용은 Amazon SES 설명서의 Amazon SES에서 도메인 확인을 참조하세요.

Amazon SES에서 도메인 확인

1. Amazon SES 콘솔의 왼쪽 탐색 메뉴에서 확인된 ID를 선택합니다.
2. 자격 증명 생성(Create identity)을 선택합니다.
3. 확인하려는 도메인을 입력하고 보안 인증 정보 생성을 선택합니다.

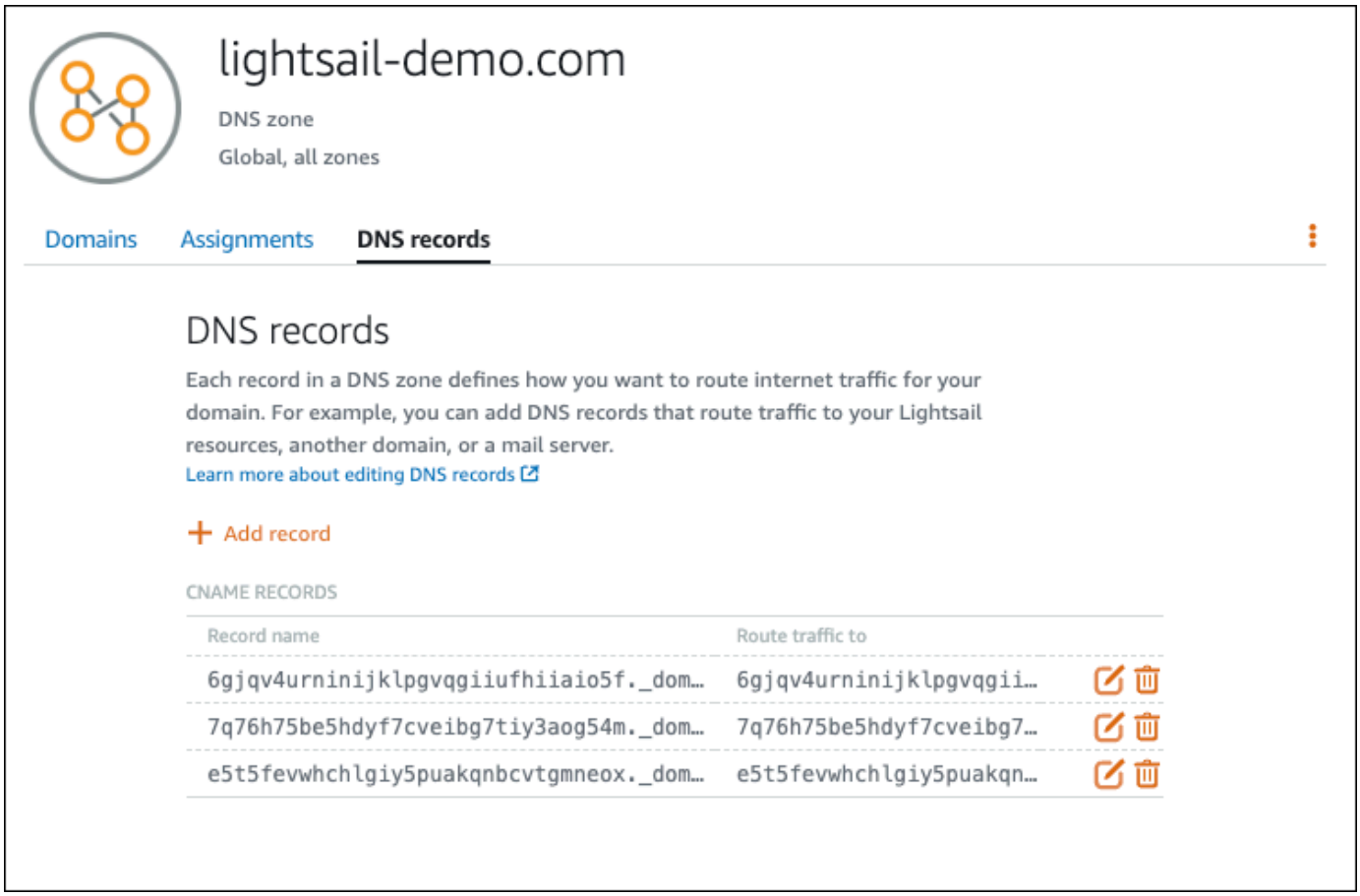
   사용자가 확인하는 도메인은 Lightsail의 WordPress 인스턴스에서 사용 중인 도메인과 동일해야 합니다.

> **Important**
>
> 레거시 TXT 레코드
> 이제 Amazon SES는 수신 메일 서버가 이메일의 신뢰성을 검증하는 데 사용하는 이메일 인증 표준인 도메인키 식별 메일(DKIM)에 기반하여 도메인 확인을 수행합니다. 도메인의 DNS 설정에서 DKIM을 구성하면 사용자가 보안 인증 정보 소유자임을 SES에 확인시켜 주므로 TXT 레코드가 필요하지 않습니다. TXT 레코드를 사용하여 확인된 도메인 보안 인증 정보는 다시 확인할 필요가 없습니다. 하지만 DKIM 호환 이메일 제공업체를 통한 메일 전달 가능성을 높이려면 DKIM 서명을 활성화하는 것이 좋습니다.


Create identity
A verified identity is a domain, subdomain, or email address you use to send email through Amazon SES. Identity verification at the domain level extends to all email addresses under one verified domain identity.
Identity details Info
Identity type
Domain
To verify ownership of a domain, you must have access to its DNS settings to add the necessary records.
Email address
To verify ownership of an email address, you must have access to its inbox to open the verification email.
Domain
lightsail-demo.com
Domain name can contain up to 253 alphanumeric characters.
Assign a default configuration set
Enabling this option ensures that the assigned configuration set is applied to messages sent from this identity by default whenever a configuration set isn't specified at the time of sending.
Use a custom MAIL FROM domain
Configuring a custom MAIL FROM domain for messages sent from this identity enables the MAIL FROM address to align with the From address. Domain alignment must be achieved in order to be DMARC compliant.
Verifying your domain
DKIM-based domain verification
DomainKeys Identified Mail (DKIM) is an email authentication method that Amazon SES uses to verify domain ownership and that receiving mail servers use to validate email authenticity. You must configure DKIM as part of the domain verification process.
Configuring DKIM
Following identity creation, Amazon SES will provide a set of DNS records. These records must be published to your domain's DNS server in order to successfully configure DKIM and verify ownership of your domain. For more information, see Verifying a domain with Amazon SES.
If your domain is registered with Amazon Route 53, Amazon SES will automatically update your domain's DNS server with the necessary records. This can be disabled by expanding the Advanced DKIM settings and unchecking Publish DNS records to Route53 in the Easy DKIM selection.
Advanced DKIM settings
Identity type
Easy DKIM
To set up Easy DKIM, you have to modify the DNS settings for your domain.
Provide DKIM authentication token (BYODKIM)
Configure DKIM for this domain by providing your own private key.

4. Easy DKIM을 사용하여 도메인 보안 인증 정보를 생성한 후에는 다음과 같이 생성된 CNAM 레코드를 복사하고 도메인의 DNS 제공업체에 게시하여 DKIM 인증으로 확인 프로세스를 완료해야 합니다. 이러한 레코드가 검색되려면 최대 72시간이 소요될 수 있습니다. 자세한 내용은 DKIM을 사용한 도메인 보안 인증 정보 확인 및 Easy DKIM을 참조하세요.
5. 새 브라우저 탭을 열고 Lightsail 콘솔로 이동합니다.
6. 왼쪽 탐색 창에서 도메인 및 DNS를 선택한 다음 도메인의 DNS 영역을 선택합니다.
7. Amazon SES 콘솔에서 DNS 레코드를 추가합니다. Lightsail에서 DNS 영역을 편집하는 방법에 대한 자세한 내용은 Amazon Lightsail에서 DNS 영역 편집을 참조하십시오.

   결과는 다음 예제와 같아야 합니다.

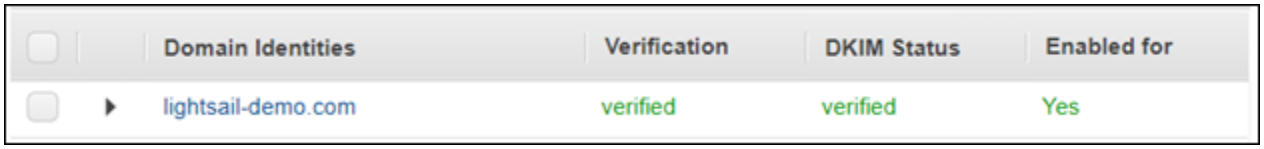


> **Note**
>
> MX 레코드에 도메인의 정점을 사용하려면 하위 도메인(Subdomain) 텍스트 상자에 @ 기호를 입력합니다. 뿐만 아니라 Amazon SES에서 제공되는 MX 레코드 값은 `10 inbound-smtp.us-west-2.amazonaws.com`입니다. 우선순위

(Priority)로 10을 입력하고 다음으로 매핑(Maps to) 도메인으로 inbound-smtp.us-west-2.amazonaws.com을 입력합니다.

8. Amazon SES 콘솔에서 새 도메인 확인 페이지를 닫습니다.

   아래 예제에서와 같이 몇 분 후면 Amazon SES 콘솔에 나열된 도메인에서 확인됨으로 레이블이 지정되고 전송이 활성화됩니다.



   이제 Amazon SES의 SMTP 서비스에서 도메인에서 이메일을 전송할 준비가 완료되었습니다.

5단계: Amazon SES에서 이메일 주소 확인

새로운 Amazon SES 고객은 이메일을 전송하고자 하는 이메일 주소를 확인해야 합니다. Amazon SES 콘솔에서 이메일 주소를 추가하여 이를 수행할 수 있습니다. 자세한 내용은 Amazon SES 설명서의 Amazon SES에서 이메일 주소 확인을 참조하세요.

WordPress 웹사이트 관리자의 이메일 주소를 추가하는 것이 좋습니다. 이렇게 하면 사용자 프로필에 대해 암호 재설정을 요청할 수 있으며, 블로그 게시물, 웹사이트 업데이트 및 기타 플러그인 메시지에 대해 이메일 알림을 받게 됩니다.

> **Note**
>
> 확인 없이 어떤 주소이든 이메일을 전송하고 싶은 경우에는 Amazon SES 계정이 샌드박스 환경에서 나가도록 요청해야 합니다. 자세한 내용은 Amazon SES 설명서의 Amazon SES 샌드박스 환경에서 나가기를 참조하세요.

이메일 주소 자격 증명을 생성하려면

1. Amazon SES 콘솔의 왼쪽 탐색 메뉴에서 확인된 ID를 선택합니다.
2. 자격 증명 생성(Create identity)을 선택합니다.
3. 이메일 주소를 선택합니다. 확인하려는 이메일 주소를 입력합니다.
4. 자격 증명 생성(Create identity)을 선택합니다.

확인하려는 모든 이메일 주소에 대해 1 ~ 4단계를 반복합니다. 확인 이메일이 입력한 이메일 주소로 전송됩니다. 상태가 “확인 대기 중”으로 확인된 이메일 ID의 목록에 이 주소가 추가됩니다. 사용자가 이메일 메시지를 열어서 확인 프로세스를 완료하면 “확인됨”으로 표시가 됩니다.

이메일 주소 자격 증명을 확인하는 방법

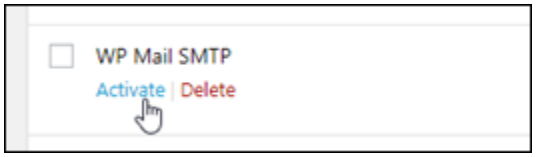
1. 자격 증명을 생상하는 데 사용한 이메일 주소의 받은 편지함에서 no-reply-aws@amazon.com에서 보낸 이메일이 있는지 확인합니다.
2. 이메일을 열고 링크를 클릭하여 해당 이메일 주소에 대한 확인 프로세스를 완료합니다. 작업이 완료되면 자격 증명 상태가 확인됨으로 업데이트됩니다.

6단계: WordPress 인스턴스에서 WP Mail SMTP 플러그인 구성

최종 단계는 WordPress 인스턴스에서 WP Mail SMTP 플러그인을 구성하는 것입니다. 이 설명서 앞부분에서 생성한 SMTP 보안 인증을 Amazon SES 콘솔에서 사용합니다.

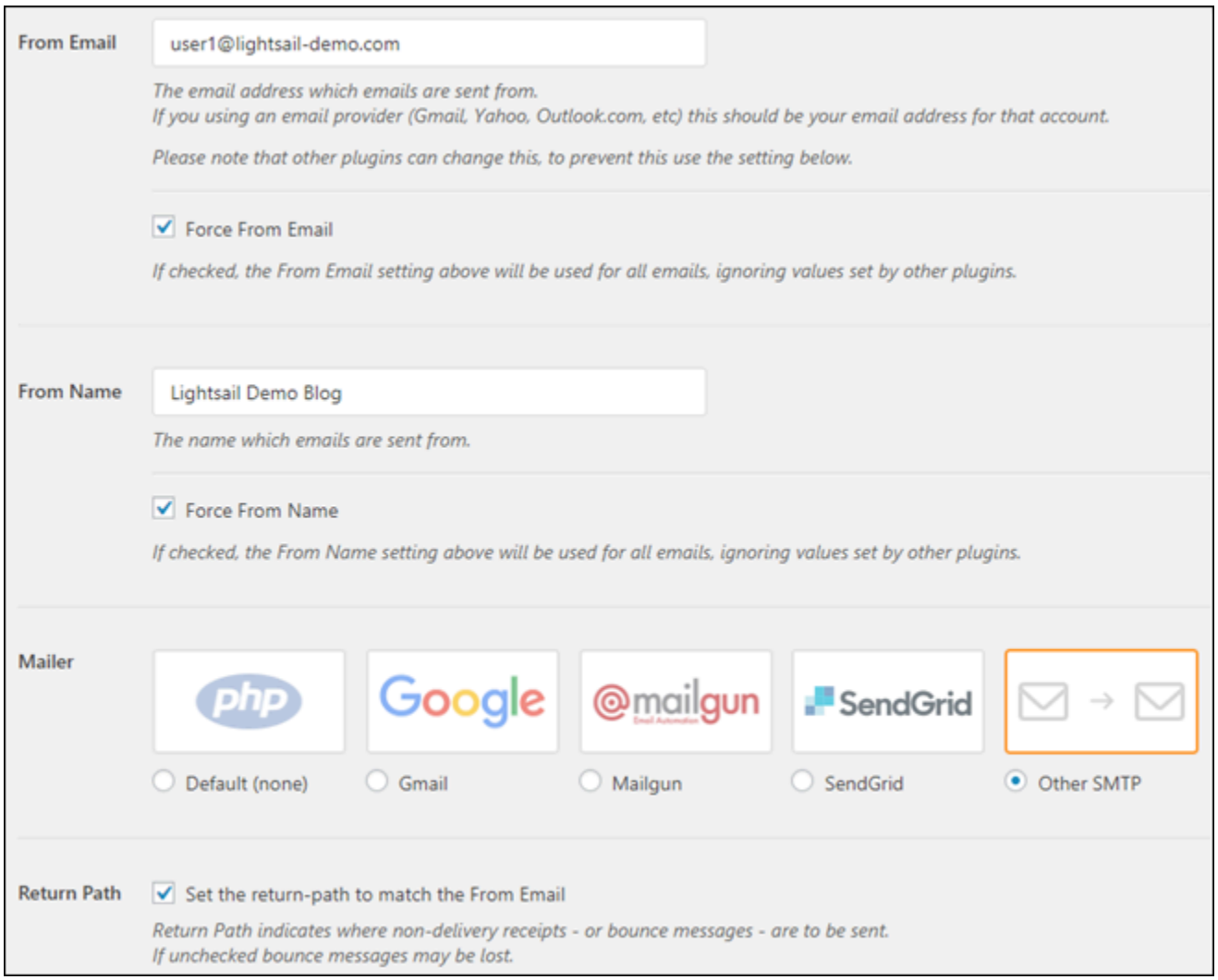
WordPress 인스턴스에서 WP Mail SMTP 플러그인을 구성하려면

1. WordPress 웹사이트의 대시보드에 관리자로 로그인합니다.
2. 왼쪽 탐색 메뉴에서 Plugins(플러그인)를 선택한 다음, Installed Plugins(설치된 플러그인)를 선택합니다.
3. 아래로 스크롤하여 WP Mail SMTP 플러그인을 찾은 다음, 활성화를 선택합니다. 새 버전의 플러그인이 있는 경우에는 다음 단계로 넘어가기 전에 이를 반드시 업데이트해야 합니다.

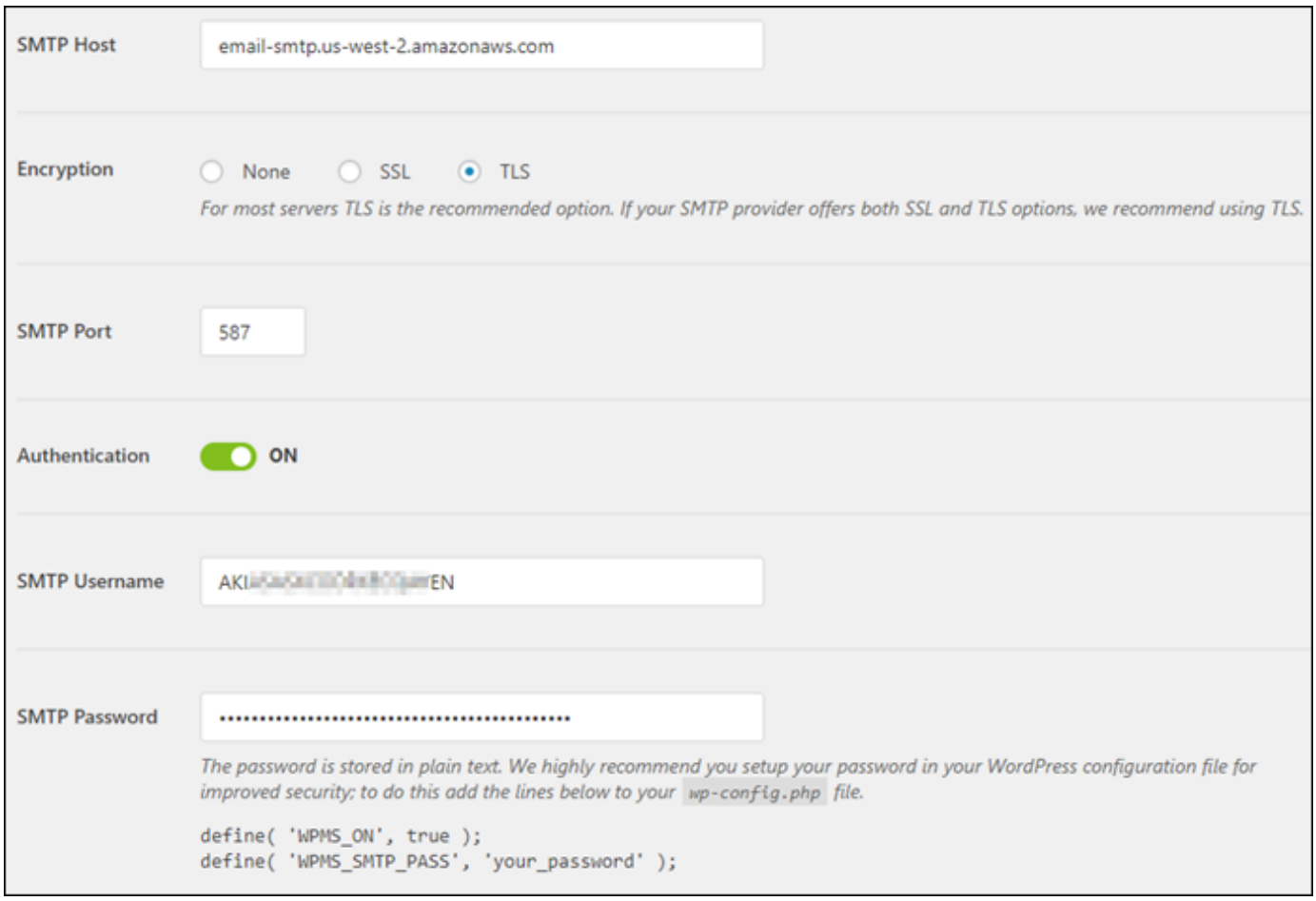


4. WP Mail SMTP 플러그인이 활성화되고 나면 설정을 선택합니다. 아래로 스크롤하여 플러그인을 찾아야 할 수 있습니다.

5. 발신 이메일 주소 텍스트 상자에 이메일 발신을 원하는 이메일 주소를 입력합니다. 입력한 이메일 주소는 이 설명서 앞부분에 나온 단계를 사용하여 Amazon SES에서 확인이 되어야 합니다.
6. Force From Email(이메일에서 강제 적용)을 선택하여 발신 이메일 주소 텍스트 상자에 입력하는 이메일 주소를 강제로 사용하고, 다른 플러그인에서 설정된 “발신 이메일 주소” 값을 무시합니다.
7. From Name(발산 이름) 텍스트 상자에 이메일 발신을 원하는 이름을 입력하거나 있는 그대로 둬서 WordPress 블로그의 이름을 사용합니다.
8. Force From Name(이메일에서 강제 적용)을 선택하여 From Name(발신 이름) 텍스트 상자에 입력한 이름을 강제로 사용합니다. 이 옵션을 선택하면 다른 플러그인에서 설정된 “발신 이름” 값을 무시하고, WordPress에게 From Name(발신 이름) 텍스트 상자에 입력한 이름을 사용하도록 강제합니다.
9. 페이지의 발신자 섹션에서 Other SMTP(기타 SMTP)를 선택합니다.
10. Set the return-path to match the From Email(발신 이메일과 일치하도록 반환 경로 설정)을 선택하여 From Email Address(발신 이메일 주소) 텍스트 상자에 입력한 이메일 주소로 송달 방지 영수증이 전송되도록 합니다.



11. SMTP 호스트 텍스트 상자에 이 설명서의 앞부분에서 Amazon SES 콘솔의 SMTP 설정 페이지로부터 가져온 SMTP 서버 이름을 입력합니다.
12. 페이지의 암호화 섹션에서 TLS를 선택하여 Amazon SES의 SMTP 서비스가 TLS 암호화를 사용하도록 지정합니다.
13. SMTP Port(SMTP 포트) 텍스트 상자에서 기본 값을 587로 그대로 둡니다.
14. 인증 토글을 활성으로 전환한 다음, 이 설명서 앞부분에서 Amazon SES 콘솔로부터 얻은 SMTP 사용자 이름 및 암호를 입력합니다.

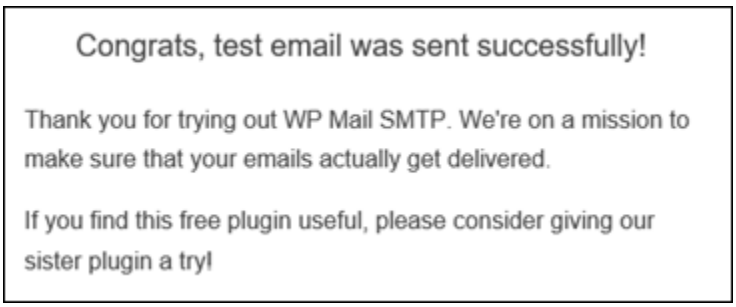


15. Save Settings(설정 저장)를 선택합니다. 설정이 성공적으로 저장되었음을 확인하는 프롬프트가 나타납니다.
16. Email Test(이메일 테스트) 탭을 선택합니다.

    다음 단계에서 이메일 서비스가 작동 중인지 확인하기 위한 테스트 이메일을 전송합니다.
17. 전송 대상 텍스트 상자에 이메일 주소를 입력한 다음 Send Email(이메일 전송)을 선택합니다. 입력한 이메일 주소는 이 설명서 앞부분에 나온 단계를 사용하여 Amazon SES에서 확인이 되어야 합니다.

    사용자가 보게 되는 결과는 두 가지입니다.

    - 성공 확인 메시지가 나타나면 이메일에 대해 WordPress 웹사이트가 활성화됩니다. 다음과 같은 테스트 이메일이 지정된 사서함에 도착했는지 확인합니다.



이제 WordPress 웹사이트의 대시보드에 대한 로그인 페이지에서 Lost your password?(암호를 분실한 경우)를 선택할 수 있습니다. WordPress 사용자 프로필의 이메일 주소를 Amazon SES에서 확인한 경우에는 새 암호가 이메일로 전송됩니다.

- 오류 알림이 표시되는 경우에는 WP Mail SMTP 플러그인에 입력한 SMTP 설정이 Amazon SES 계정의 SMTP 서비스 설정과 일치하는지 확인합니다. 또한 Amazon SES에서 확인한 이메일 주소를 사용하고 있는지도 확인합니다.

Lightsail에서 HTTPS로 WordPress 사이트 보호

WordPress 웹 사이트에 Hypertext Transfer Protocol Secure(HTTPS)를 활성화하면 방문자에게 웹 사이트가 안전하다는 것을 보장하고 암호화된 데이터를 보내고 받을 수 있습니다. 보안되지 않은 웹 사이트의 주소는 `http`로 시작(예: `http://example.com`)하는 반면, 보안 웹 사이트의 주소는 `https`(예: `https://example.com`)로 시작합니다. 주로 정보를 제공하는 웹 사이트인 경우에도 HTTPS를 활성화하는 것이 좋습니다. 대부분의 웹 브라우저가 HTTPS를 활성화하지 않으면 웹 사이트 방문자에게 웹 사이트가 안전하지 않다고 안내하며, 검색 엔진 결과에서 웹 사이트가 아래쪽에 표시됩니다.

> **Tip**
>
> Lightsail은 WordPress 인스턴스에서 SSL/TLS Let's Encrypt 인증서의 설치 및 구성을 자동화하는 안내 워크플로를 제공합니다. 이 자습서의 수동 단계를 따르지 않고 워크플로를 사용하는 것이 좋습니다. 자세한 내용은 WordPress 인스턴스 시작 및 구성을 참조하세요.

이 가이드에서는 Bitnami HTTPS 구성 도구(`bncert`)를 사용하여 Amazon Lightsail의 Bitnami 인증 WordPress 인스턴스에서 HTTPS를 활성화하는 방법을 안내합니다. 요청할 때 지정한 도메인 및 하위 도메인에 대해서만 인증서를 요청할 수 있습니다. 아니면 도메인용 인증서와 하위 도메인용 와일드카드 인증서를 요청할 수 있는 Certbot 도구를 사용할 수도 있습니다. 와일드카드 인증서는 도메인의 모든 하위 도메인에서 사용할 수 있으며, 트래픽을 인스턴스로 연결하는 데 사용할 하위 도메인을 모르는 경우 유용합니다. 그러나 Certbot은 `bncert` 도구처럼 인증서를 자동으로 갱신하지 않습니다. Certbot

을 사용할 경우 90일마다 수동으로 인증서를 갱신해야 합니다. Certbot을 사용하여 HTTPS를 활성화하는 방법에 대한 자세한 내용은 자습서: WordPress 인스턴스와 함께 Let's Encrypt SSL 인증서 사용을 참조하세요.

목차

- 1단계: 프로세스 알아보기
- 2단계: 사전 조건 완료
- 3단계: 인스턴스에 연결
- 4단계: 인스턴스에 bncert 도구 설치 여부 확인
- 5단계: WordPress 인스턴스에서 HTTPS 활성화
- 6단계: 웹 사이트에서 HTTPS 사용 여부 테스트

1단계: 프로세스 알아보기

> **Note**
>
> 이 섹션에서는 프로세스의 개괄적인 개요를 살펴봅니다. 이 프로세스를 수행하기 위한 구체적인 단계는 이 가이드의 후속 단계에 나와 있습니다.

WordPress 웹 사이트에 HTTPS를 활성화하려면 SSH를 사용하여 Lightsail 인스턴스에 연결한 다음, `bncert` 도구를 사용하여 Let's Encrypt 인증 기관에서 SSL/TLS 인증서를 요청합니다. 인증서를 요청할 때 웹 사이트의 기본 도메인(`example.com`)과 대체 도메인(`www.example.com`, `blog.example.com` 등)을 지정해야 합니다. Let's Encrypt는 도메인의 DNS에 TXT 레코드를 생성하도록 요청하거나 해당 도메인이 요청한 인스턴스의 퍼블릭 IP 주소로 트래픽을 이미 연결하고 있는지 확인하여 소유한 도메인을 검증합니다.

인증서가 검증되면 WordPress 웹 사이트를 구성해서 방문자를 HTTP에서 HTTPS로 자동 리디렉션(`http://example.com`을 `https://example.com`으로 리디렉션)하여 방문자가 암호화된 연결을 사용하도록 할 수 있습니다. `www` 하위 도메인을 도메인의 정점으로 자동 리디렉션(`https://www.example.com`을 `https://example.com`으로 리디렉션)하거나 그 반대로 리디렉션(`https://example.com`을 `https://www.example.com`으로 리디렉션)하도록 웹 사이트를 구성할 수도 있습니다. 이러한 리디렉션은 `bncert` 도구를 사용하여 구성됩니다.

Let's Encrypt를 사용하면 90일마다 인증서를 갱신해야 웹 사이트에서 HTTPS를 유지할 수 있습니다. `bncert` 도구는 인증서를 자동으로 갱신하므로, 사용자는 웹 사이트 자체에 더 많은 시간을 투자할 수 있습니다.

bncert 도구의 제한 사항

`bncert` 도구에는 다음과 같은 제한 사항이 있습니다.

- 생성 시 모든 Bitnami 인증 WordPress 인스턴스에 미리 설치되어 있지 않습니다. 이전에 Lightsail에서 생성된 WordPress 인스턴스를 사용하면 `bncert` 도구를 수동으로 설치해야 합니다. 이 가이드의 4단계에서는 도구가 인스턴스에 설치되어 있는지 확인하는 방법과 설치되어 있지 않은 경우 설치하는 방법을 설명합니다.
- 요청할 때 지정한 도메인 및 하위 도메인에 대해서만 인증서를 요청할 수 있습니다. 이는 Certbot 도구와는 다른데, Certbot 도구를 사용하면 도메인용 인증서와 하위 도메인용 와일드카드 인증서를 요청할 수 있습니다. 와일드카드 인증서는 도메인의 모든 하위 도메인에서 사용할 수 있으며, 트래픽을 인스턴스로 연결하는 데 사용할 하위 도메인을 모르는 경우 유용합니다. 그러나 Certbot은 `bncert` 도구처럼 인증서를 자동으로 갱신하지 않습니다. Certbot을 사용할 경우 90일마다 수동으로 인증서를 갱신해야 합니다. Certbot을 사용하여 HTTPS를 활성화하는 방법에 대한 자세한 내용은 자습서: Amazon Lightsail에서 WordPress 인스턴스와 함께 Let's Encrypt SSL 인증서 사용을 참조하세요.

2단계: 사전 조건 완료

아직 수행하지 않은 경우 다음 사전 조건을 완료하세요.

- Lightsail에서 WordPress 인스턴스를 생성하고 인스턴스에서 웹 사이트를 구성합니다. 자세한 내용은 Amazon Lightsail에서 Linux/Unix 기반 인스턴스 시작하기를 참조하세요.
- 고정 IP를 인스턴스에 연결합니다. 인스턴스를 중지했다가 시작하면 인스턴스의 퍼블릭 IP 주소가 변경됩니다. 인스턴스를 중지했다가 시작해도 고정 IP는 변경되지 않습니다. 자세한 내용은 고정 IP를 생성하여 Amazon Lightsail의 인스턴스에 연결을 참조하세요.
- WordPress 인스턴스 구성을 완료한 후 스냅샷을 생성하거나 자동 스냅샷을 활성화합니다. 스냅샷은 원본 인스턴스에 문제가 발생할 경우에 대비하여 다른 인스턴스를 생성할 수 있는 백업으로 사용할 수 있습니다. 자세한 내용은 Linux 또는 Unix 인스턴스의 스냅샷 생성 또는 Amazon Lightsail에서 인스턴스 또는 디스크의 자동 스냅샷 활성화 또는 비활성화를 참조하세요.
- 도메인의 정점(`example.com`) 및 `www` 하위 도메인(`www.example.com`)에 대한 트래픽을 Lightsail에 있는 WordPress 인스턴스의 퍼블릭 IP 주소로 연결하는 도메인의 DNS에 DNS 레코드를 추가합니다. 도메인의 현재 DNS 호스팅 공급자에서 이러한 작업을 완료할 수 있습니다. 도메인의 DNS 관

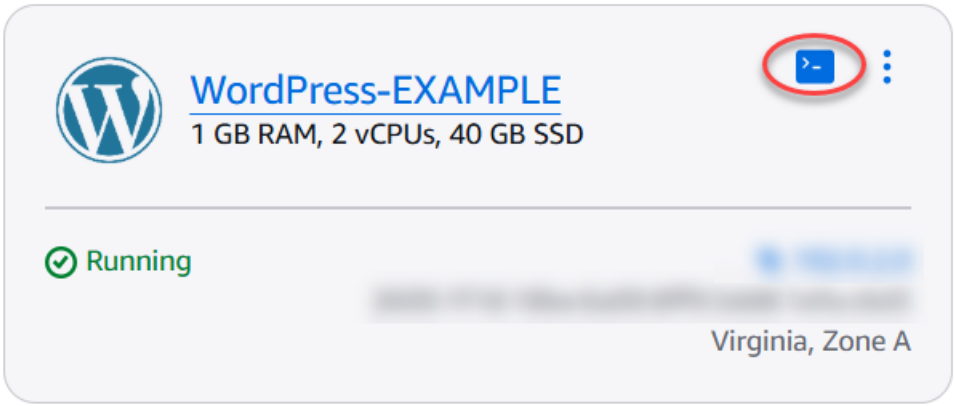
리를 Lightsail로 전송한 경우 Lightsail의 DNS 영역을 사용하여 작업을 완료할 수도 있습니다. 자세한 내용은 DNS를 참조하세요.

> **Important**
>
> WordPress 웹 사이트에서 사용할 모든 도메인의 DNS에 DNS 레코드를 추가합니다. 이러한 도메인은 모두 WordPress 웹 사이트의 퍼블릭 IP 주소로 트래픽을 라우팅해야 합니다. `bncert` 도구는 현재 WordPress 인스턴스의 퍼블릭 IP 주소로 트래픽을 연결하는 도메인에 대해서만 인증서를 발급합니다.

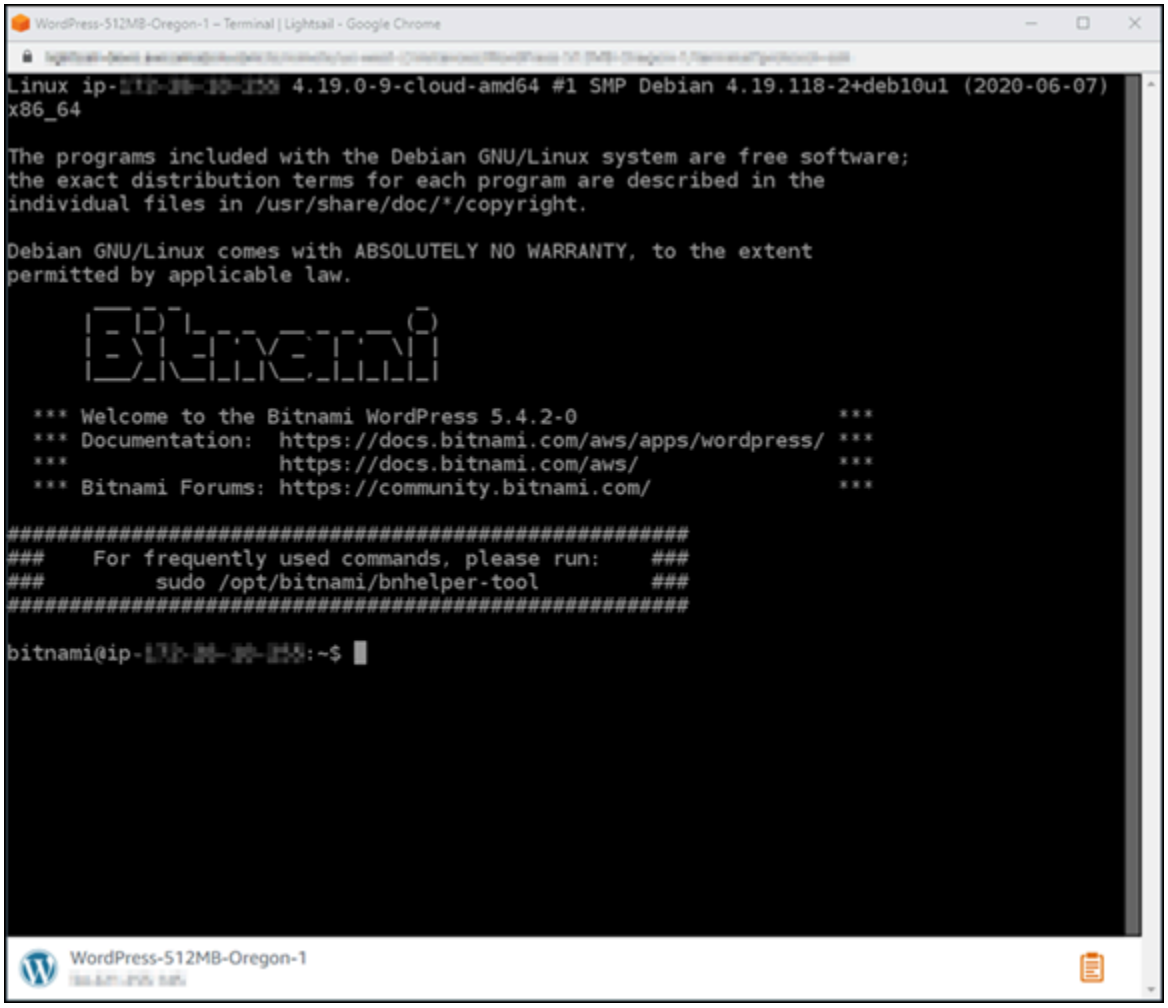
3단계: 인스턴스에 연결

Lightsail 콘솔에서 브라우저 기반 SSH 클라이언트를 사용하여 인스턴스에 연결하려면 다음 단계를 완료하세요.

1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 WordPress 인스턴스에 대한 SSH 빠른 연결 아이콘을 선택합니다.



브라우저 기반 SSH 클라이언트 터미널 창이 열립니다. 다음 예와 같이 Bitnami 로고가 표시되면 SSH를 통해 인스턴스에 성공적으로 연결된 것입니다.



4단계: 인스턴스에 bncert 도구 설치 여부 확인

Bitnami HTTPS 구성 도구(bncert)가 인스턴스에 설치되었는지 확인하려면 다음 단계를 완료하세요. 생성 시 모든 Bitnami 인증 WordPress 인스턴스에 미리 설치되어 있지 않습니다. 이전에 Lightsail에서 생성된 WordPress 인스턴스를 사용하면 bncert 도구를 수동으로 설치해야 합니다. 이 절차에는 도구가 설치되지 않은 경우 도구를 설치하는 단계가 포함되어 있습니다.

1. 다음 명령을 입력하여 bncert 도구를 실행합니다.

   ```
   sudo /opt/bitnami/bncert-tool
   ```

   - 다음 예와 같이 응답에 command not found가 표시되는 경우, bncert 도구가 인스턴스에 설치되지 않은 것입니다. 이 절차의 다음 단계를 계속 진행하여 인스턴스에 bncert 도구를 설치합니다.

> **Important**
>
> bncert 도구는 Bitnami 인증 WordPress 인스턴스에서만 사용할 수 있습니다. 또는 Certbot 도구를 사용하여 WordPress 인스턴스에서 HTTPS를 활성화할 수 있습니다. 자세한 내용은 자습서: WordPress 인스턴스와 함께 Let's Encrypt SSL 인증서 사용을 참조하세요.

```
bitnami@ip-[redacted]:~$ sudo /opt/bitnami/bncert-tool
sudo: /opt/bitnami/bncert-tool: command not found
bitnami@ip-[redacted]:~$
```

- 다음 예와 같이 응답에 Welcome to the Bitnami HTTPS configuration tool이 표시되는 경우, bncert 도구가 인스턴스에 설치된 것입니다. 이 가이드의 5단계: WordPress 인스턴스에서 HTTPS 활성화 섹션을 이어서 진행합니다.

```
bitnami@ip-[redacted]:~$ sudo /opt/bitnami/bncert-tool
--------------------------------------------------------------------------------
Welcome to the Bitnami HTTPS Configuration tool.

--------------------------------------------------------------------------------
Domains

Please provide a valid space-separated list of domains for which you wish to
configure your web server.

Domain list []:
```

2. 다음 명령을 입력하여 bncert 실행 파일을 인스턴스로 다운로드합니다.

```
wget -O bncert-linux-x64.run https://downloads.bitnami.com/files/bncert/latest/
bncert-linux-x64.run
```

3. 다음 명령을 입력하여 인스턴스에서 bncert 실행 파일의 디렉터리를 생성합니다.

```
sudo mkdir /opt/bitnami/bncert
```

4. 다음 명령을 입력하여 다운로드한 bncert 실행 파일을 생성한 새 디렉터리로 이동합니다.

```
sudo mv bncert-linux-x64.run /opt/bitnami/bncert/
```

5. 다음 명령을 입력하여 bncert에서 프로그램으로 실행할 수 있는 파일을 실행하도록 합니다.

```
sudo chmod +x /opt/bitnami/bncert/bncert-linux-x64.run
```

6. 다음 명령을 입력하여 sudo /opt/bitnami/bncert-tool 명령을 입력할 때 bncert 도구를 실행하는 심볼 링크를 생성합니다.

   ```
   sudo ln -s /opt/bitnami/bncert/bncert-linux-x64.run /opt/bitnami/bncert-tool
   ```

   이제 인스턴스에 bncert 도구를 설치했습니다. 이 가이드의 5단계: WordPress 인스턴스에서 HTTPS 활성화 섹션을 이어서 진행합니다.

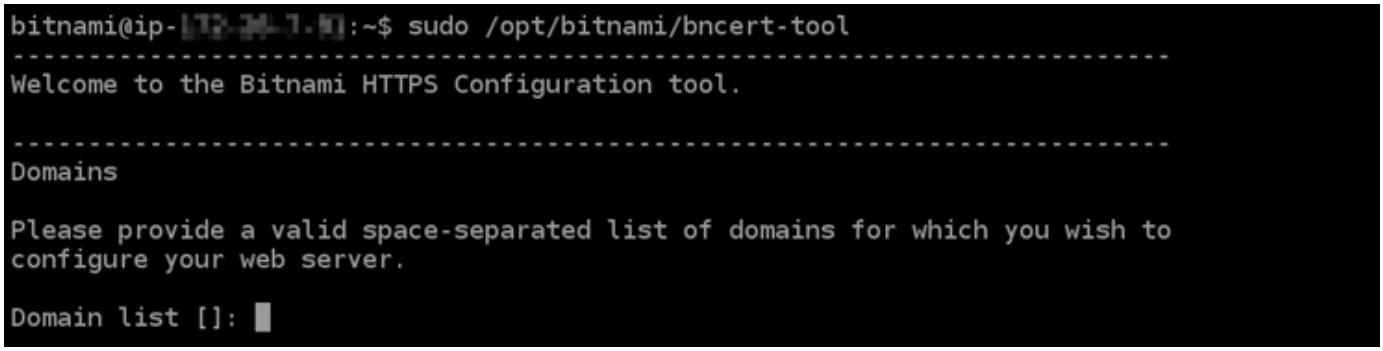
5단계: WordPress 인스턴스에서 HTTPS 활성화

bncert 도구가 사용자의 인스턴스에 설치되었는지 확인한 후 다음 절차를 완료하여 WordPress 인스턴스에서 HTTPS를 활성화합니다.

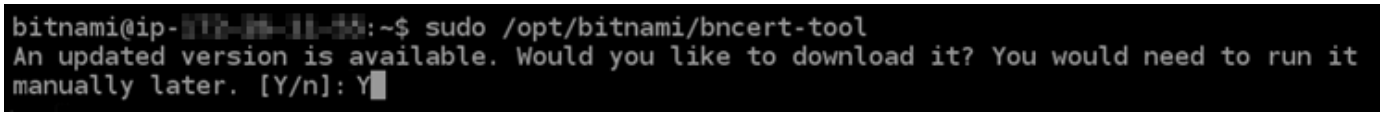
1. 다음 명령을 입력하여 bncert 도구를 실행합니다.

   ```
   sudo /opt/bitnami/bncert-tool
   ```

   다음 예와 비슷한 메시지가 나타나는 것을 볼 수 있습니다.

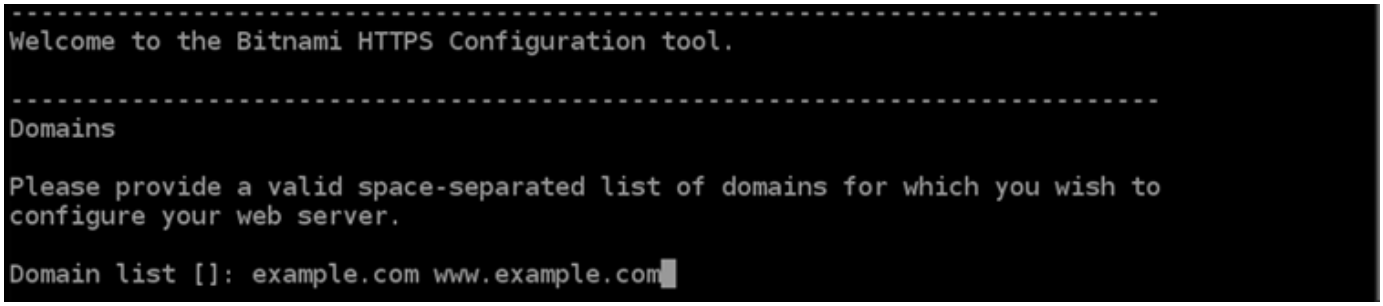
   

   bncert 도구가 일시적으로 인스턴스에 설치된 경우, 업데이트된 버전의 도구를 사용할 수 있다는 메시지가 표시될 수 있습니다. 다음 예와 같이 다운로드하도록 선택하고 sudo /opt/bitnami/bncert-tool 명령을 입력하여 bncert 도구를 다시 실행합니다.

2. 다음 예와 같이 기본 도메인 이름과 대체 도메인 이름을 공백으로 구분하여 입력합니다.

도메인이 트래픽을 인스턴스의 퍼블릭 IP 주소로 라우팅하도록 구성되지 않은 경우, bncert 도구에서 계속하기 전에 해당 구성을 설정하라는 메시지를 표시합니다. 도메인은 bncert 도구를 사용하여 인스턴스에서 HTTPS를 활성화한 인스턴스의 퍼블릭 IP 주소로 트래픽을 라우팅해야 합니다. 이렇게 해야 도메인을 소유하고 있음을 확인하고 인증서를 검증하는 역할을 할 수 있습니다.



3. bncert 도구는 웹 사이트의 리디렉션을 어떻게 구성할지 묻는 메시지를 표시합니다. 사용할 수 있는 옵션은 다음과 같습니다.

   - HTTP에서 HTTPS로 리디렉션 활성화(Enable HTTP to HTTPS redirection) - HTTP 버전 웹 사이트(예: http:/example.com)를 방문하는 사용자를 HTTPS 버전(예: https://example.com)으로 자동 리디렉션할지 지정합니다. 모든 방문자가 암호화된 연결을 사용하도록 강제하기 때문에 이 옵션을 활성화하는 것이 좋습니다. Y를 입력하고 Enter 키를 눌러 활성화합니다.
   - 비 www에서 www로 리디렉션 활성화(Enable non-www to www redirection) - 도메인의 정점(https://example.com)을 방문하는 사용자를 도메인의 www 하위 도메인(https://www.example.com)으로 자동 리디렉션할지 지정합니다. 이 옵션은 활성화하는 것이 좋습니다. 그러나 Google의 웹 마스터 도구와 같은 검색 엔진 도구에서 도메인의 정점을 기본 웹 사이트 주소로 지정했거나 정점이 IP를 직접 가리키고 www 하위 도메인이 CNAME 레코드를 통해 정점을 참조하는 경우, 이 옵션을 비활성화하고 대체 옵션(www에서 비 www로 리디렉션 활성화)을 활성화할 수 있습니다. Y를 입력하고 Enter 키를 눌러 활성화합니다.
   - www에서 비 www로 리디렉션 활성화(Enable www to non-www redirection) - 도메인의 www 하위 도메인(https://www.example.com)을 방문하는 사용자를 도메인의 정점(https://example.com)으로 자동 리디렉션할지 지정합니다. 비 www에서 www로 리디렉션을 활성화한 경우 이 옵션을 비활성화하는 것이 좋습니다. N를 입력하고 Enter 키를 눌러 비활성화합니다.

   선택한 내용은 다음 예와 같아야 합니다.

```
Enable/disable redirections

Please select the redirections you wish to enable or disable on your Bitnami
installation.



Enable HTTP to HTTPS redirection [Y/n]: Y



Enable non-www to www redirection [Y/n]: Y



Enable www to non-www redirection [y/N]: N
```

4. 적용될 변경 사항의 목록이 나열됩니다. Y를 입력하고 Enter 키를 눌러 확인하고 계속합니다.

```
Changes to perform

The following changes will be performed to your Bitnami installation:

1. Stop web server
2. Configure web server to use a free Let's Encrypt certificate for the domains:
example.com www.example.com
3. Configure a cron job to automatically renew the certificate each month
4. Configure web server name to: example.com
5. Enable HTTP to HTTPS redirection (example: redirect http://example.com to
https://example.com)
6. Enable non-www to www redirection (example: redirect example.com to
 www.example.com)
7. Start web server once all changes have been performed



Do you agree to these changes? [Y/n]: Y
```

5. Let's Encrypt 인증서에 연결할 이메일 주소를 입력하고 Enter 키를 누릅니다.

```
Create a free HTTPS certificate with Let's Encrypt

Please provide a valid e-mail address for which to associate your Let's Encrypt
certificate.

Domain list: example.com www.example.com

Server name: example.com

E-mail address []:
```

6. Let's Encrypt 구독자 계약을 검토합니다. Y를 입력하고 Enter 키를 눌러 계약을 수락하고 계속합니다.

```
The Let's Encrypt Subscriber Agreement can be found at:

https://letsencrypt.org/documents/LE-SA-v1.2-November-15-2017.pdf

Do you agree to the Let's Encrypt Subscriber Agreement? [Y/n]:
```

인증서 요청 및 지정한 리디렉션 구성을 비롯하여 인스턴스에서 HTTPS를 활성화하는 작업이 수행됩니다.

```
Performing changes to your installation

The Bitnami HTTPS Configuration Tool will perform any necessary actions to your
Bitnami installation. This may take some time, please be patient.
```

인증서가 성공적으로 발급되고 검증됩니다. 다음 예와 유사한 메시지가 표시되면 인스턴스에서 리디렉션이 성공적으로 구성된 것입니다.

```
Success

The Bitnami HTTPS Configuration Tool succeeded in modifying your installation.

The configuration report is shown below.

Backup files:
* /opt/bitnami/apache2/conf/httpd.conf.back.202005290035
* /opt/bitnami/apache2/conf/bitnami/bitnami-apps-prefix.conf.back.202005290035
* /opt/bitnami/apache2/conf/bitnami/bitnami.conf.back.202005290035

Find more details in the log file:

/tmp/bncert-202005290035.log

If you find any issues, please check Bitnami Support forums at:

https://community.bitnami.com

Press [Enter] to continue:
```

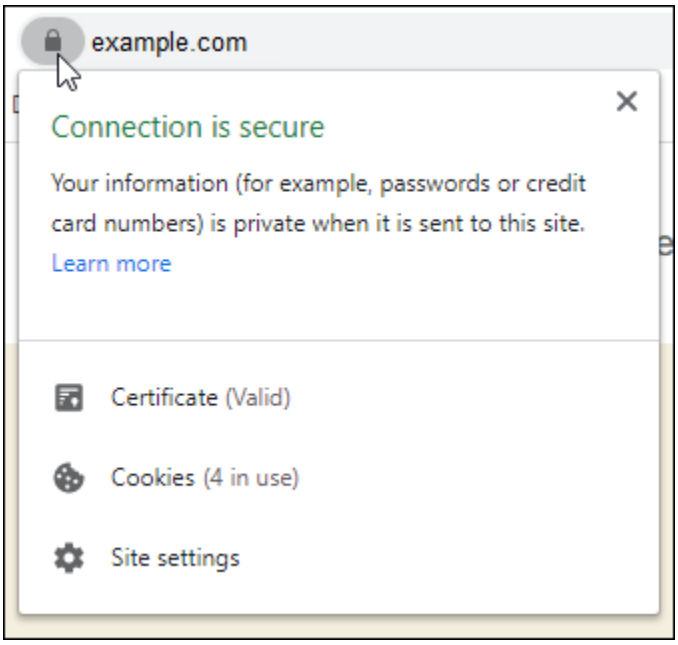
`bncert` 도구는 인증서가 만료되기 80일 전마다 인증서를 자동으로 갱신합니다. 인스턴스와 함께 추가 도메인 및 하위 도메인을 사용하고 이러한 도메인에서 HTTPS를 활성화하려면 위의 단계를 반복합니다.

이제 WordPress 인스턴스에서 HTTPS가 활성화되었습니다. 이 가이드의 6단계: 웹 사이트에서 HTTPS 사용 여부 테스트 섹션을 이어서 진행합니다.

6단계: 웹 사이트에서 HTTPS 사용 여부 테스트

WordPress 인스턴스에서 HTTPS를 활성화한 후에는 `bncert` 도구를 사용할 때 지정한 모든 도메인을 방문하여 웹 사이트에서 HTTPS를 사용하고 있는지 확인해야 합니다. 각 도메인을 방문하면 다음 예와 같이 도메인에서 보안 연결을 사용하는지 표시됩니다.

> **Note**
>
> 변경 내용을 확인하려면 브라우저의 캐시를 새로 고치고 지워야 할 수 있습니다.



`bncert` 도구를 실행할 때 선택한 옵션에 따라 비 `www` 주소가 도메인의 `www` 하위 도메인으로 리디렉션되거나 그 반대로 리디렉션될 수도 있습니다.

WordPress 블로그를 Lightsail로 마이그레이션

WordPress 호스팅 공급자를 변경하고 싶으신가요? Amazon Lightsail은 AWS에서 Wordpress 사이트를 실행하는 가장 쉬운 방법입니다.

Amazon의 요금제(매달 5 USD부터) 중 하나를 선택하면 플러그인, 테마 등을 비롯하여 WordPress 설치를 완벽히 관리할 수 있습니다.

몇 분이면 Lightsail WordPress 인스턴스를 생성할 수 있습니다. 이 자습서에 따라 기존의 WordPress 블로그를 백업하고 Lightsail에서 실행 중인 새로운 인스턴스로 가져옵니다.

다음은 그 프로세스에 대한 간략한 설명입니다.

시작하려면 계속 읽어 보세요.

사전 조건

시작하려면 다음이 필요합니다.

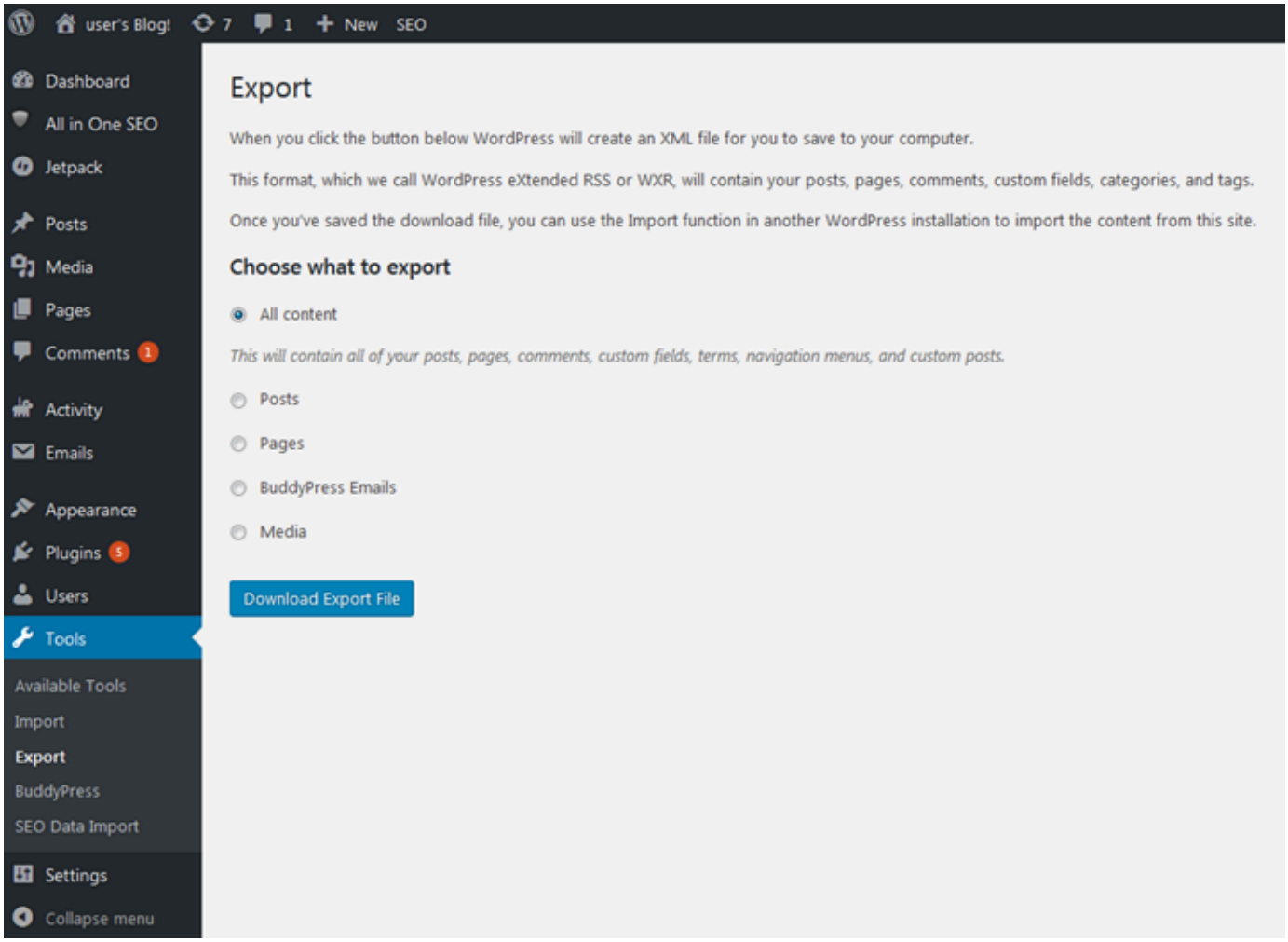
1. AWS 계정이 필요합니다. 계정이 이미 있는 경우 에 가입 AWS하거나 에 로그인 AWS합니다.
2. 계정이 Lightsail을 사용하도록 설정되어 있는지 확인합니다. 계정을 생성한 후 시간이 지났거나 아직 신용 카드를 제공하지 않은 경우 먼저에 로그인하여 계정을 AWS Management Console 업데이트해야 할 수 있습니다.

1단계: 기존 WordPress 블로그 백업

WordPress를 사용하여 기존 블로그를 백업할 수 있습니다. WordPress 관리 콘솔에 로그인하고 블로그를 관리할 수만 있으면 됩니다.

1. 블로그로 이동한 다음 관리를 선택합니다.

   관리(Manage) 배너가 표시되지 않는 경우 `http://<PublicIP>/wp-login.php`을 통해 로그인 페이지로 이동할 수 있습니다. *<PublicIP>*을(를) 인스턴스의 퍼블릭 IP 주소로 바꿉니다.
2. 사용자 이름과 암호를 입력하여 WordPress 관리 콘솔에 로그인합니다.
3. WordPress Dashboard(대시보드)에서 Tools(도구)와 Export(내보내기)를 차례대로 선택합니다.
4. Export(내보내기) 페이지에서 All content(모든 콘텐츠)를 선택하여 모든 콘텐츠를 XML 파일로 내보냅니다.



5. Download export file(내보내기 파일 다운로드)을 선택하여 이전의 블로그를 XML 파일로 다운로드합니다.

   찾기 쉬운 위치에 XML 파일을 저장합니다. 4단계에서 이 파일이 필요할 것입니다.

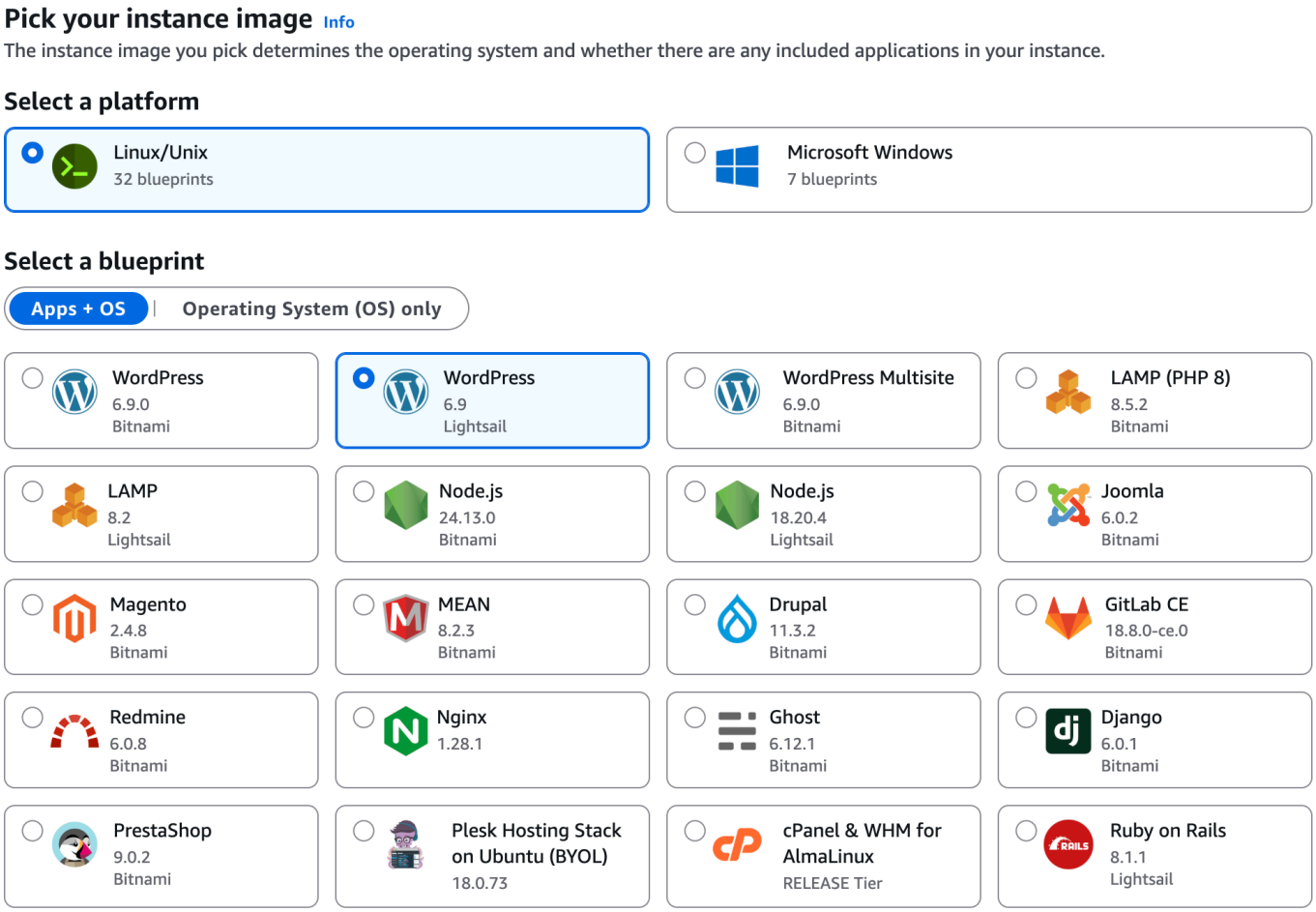
2단계: Lightsail에서 새 WordPress 인스턴스 만들기

단 몇 분 만에 Lightsail에서 새 WordPress 인스턴스를 만들 수 있습니다. 그 방법은 다음과 같습니다.

1. Lightsail 홈 페이지로 이동하여 로그인합니다.
2. 인스턴스 생성을 선택합니다.
3. 블로그를 생성할 AWS 리전 을 선택합니다.

   AWS 리전을 선택한 후 기본 가용 영역을 선택하거나 변경할 수 있습니다.
4. WordPress 블루프린트를 선택합니다.

Amazon Lightsail 또는 Bitnami 공급자를 선택할 수 있습니다.



5. 인스턴스 플랜(또는 번들)을 선택합니다.

   필요한 경우 나중에 Lightsail 플랜을 업그레이드할 수 있습니다. 자세한 내용은 Lightsail의 스냅샷에서 인스턴스 생성을 참조하세요.

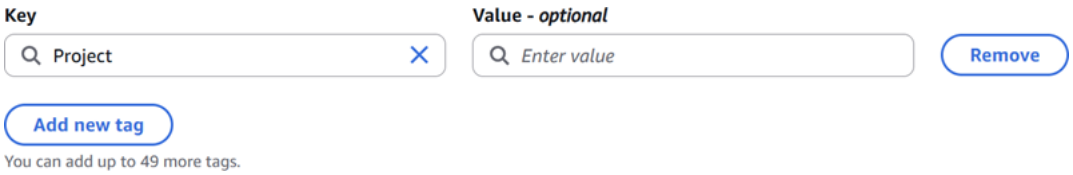
6. 인스턴스 이름을 입력합니다.

   리소스 이름:

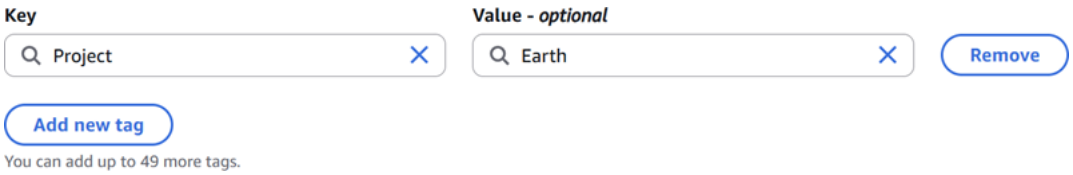
   - Lightsail 계정 AWS 리전 의 각 내에서 고유해야 합니다.
   - 2~255자로 구성되어야 합니다.
   - 영숫자 문자로 시작하고 끝나야 합니다.
   - 영숫자 문자, 마침표, 대시, 밑줄이 포함될 수 있습니다.

7. (선택 사항) 인스턴스에 태그를 추가하려면 새 태그 추가를 선택합니다. 필요에 따라 이 단계를 반복하여 모든 태그를 추가합니다. 태그 사용에 대한 자세한 내용은 태그를 참조하세요.

a. 키에는 태그 키를 입력합니다.



b. (선택 사항) 값에 태그 값을 입력합니다.



8. 인스턴스 생성을 선택합니다.

3단계: 새 Lightsail WordPress 블로그에 로그인

Lightsail에 새 블로그가 있으므로 WordPress Dashboard에 액세스하여 이전 블로그 데이터를 가져와야 합니다. WordPress 웹사이트의 관리 대시보드에 로그인하기 위한 기본 암호가 인스턴스에 저장됩니다. 암호를 가져오려면 다음 단계를 완료합니다.

Bitnami

WordPress 관리자의 기본 암호 가져오기

1. WordPress 인스턴스의 인스턴스 관리 페이지를 엽니다.
2. WordPress 패널에서 기본 암호 검색을 선택합니다. 이렇게 하면 페이지 하단의 액세스 기본 암호가 확장됩니다.

3. CloudShell 시작을 선택합니다. 이는 페이지 하단에 분할 창을 엽니다.
4. 복사를 선택한 다음 콘텐츠를 CloudShell 창에 붙여넣습니다. 커서를 CloudShell 프롬프트에 놓고 Ctrl+V를 누르거나 마우스 오른쪽 버튼을 클릭하여 메뉴를 열고 붙여넣기를 선택할 수 있습니다.
5. CloudShell 창에 표시된 암호를 기록해 둡니다. 나중에 WordPress 웹사이트의 관리 대시보드에 로그인하는 데 사용합니다.

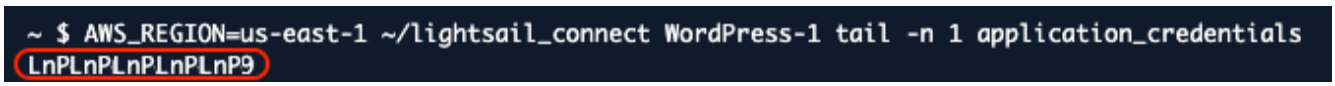
```
[cloudshell-user@ip-[redacted] ~]$ AWS_REGION=us-east-1 ~/lightsail_connect WordPress-1 cat bitnami_application_password
JKzhKzhKzhR!
```

Lightsail

WordPress 관리자의 기본 암호 가져오기

1. WordPress 인스턴스의 인스턴스 관리 페이지를 엽니다.
2. WordPress 패널에서 기본 암호 검색을 선택합니다. 이렇게 하면 페이지 하단의 액세스 기본 암호가 확장됩니다.

3. CloudShell 시작을 선택합니다. 이는 페이지 하단에 분할 창을 엽니다.
4. 복사를 선택한 다음 콘텐츠를 CloudShell 창에 붙여넣습니다. 커서를 CloudShell 프롬프트에 놓고 Ctrl+V를 누르거나 마우스 오른쪽 버튼을 클릭하여 메뉴를 열고 붙여넣기를 선택할 수 있습니다.
5. CloudShell 창에 표시된 암호를 기록해 둡니다. 나중에 WordPress 웹사이트의 관리 대시보드에 로그인하는 데 사용합니다.



이제 WordPress 웹사이트의 관리 대시보드에 대한 암호가 있으므로 로그인할 수 있습니다. 관리 대시보드에서 사용자 암호를 변경하고, 플러그인을 설치하고, 웹사이트의 테마를 변경하는 등의 작업을 할 수 있습니다.

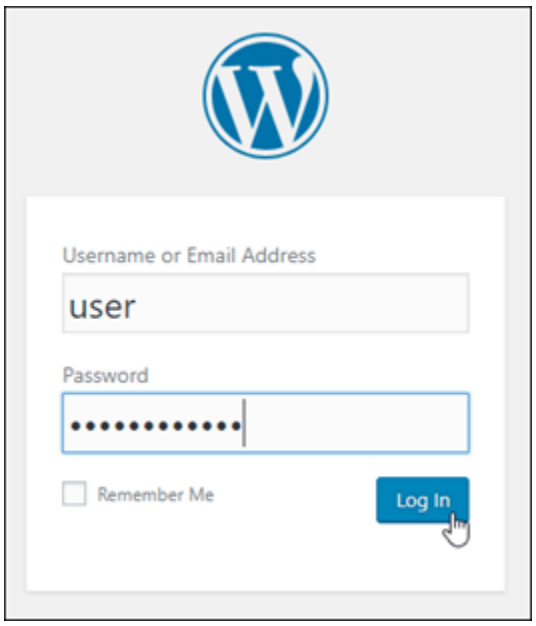
WordPress 웹사이트의 관리 대시보드에 로그인하려면 다음 단계를 완료합니다.

관리자 대시보드에 로그인

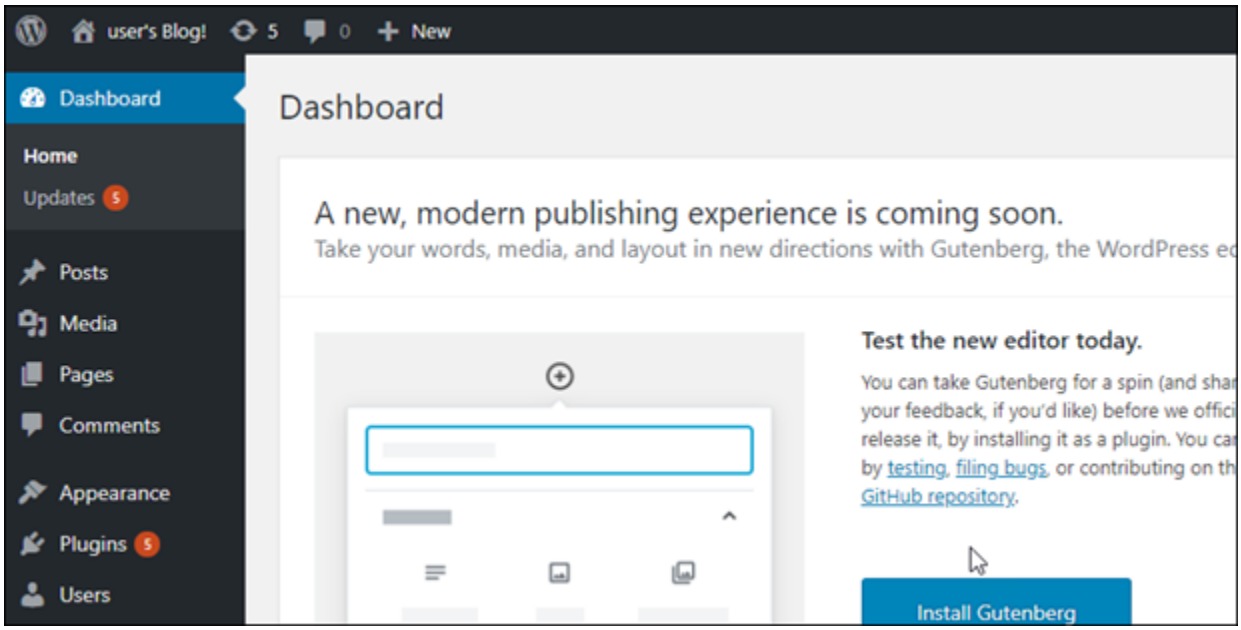
1. WordPress 인스턴스의 인스턴스 관리 페이지를 엽니다.
2. WordPress 패널에서 WordPress 관리자 액세스를 선택합니다.
3. WordPress 관리 대시보드 액세스 패널의 퍼블릭 IP 주소 사용에서 다음 형식의 링크를 선택합니다.

   http://*public-ipv4-address*./wp-admin
4. 사용자 이름 또는 이메일 주소 상자에 **user**를 입력합니다.

5. 암호에 이전 단계에서 얻은 암호를 입력합니다.
6. 그런 다음 로그인을 선택합니다.



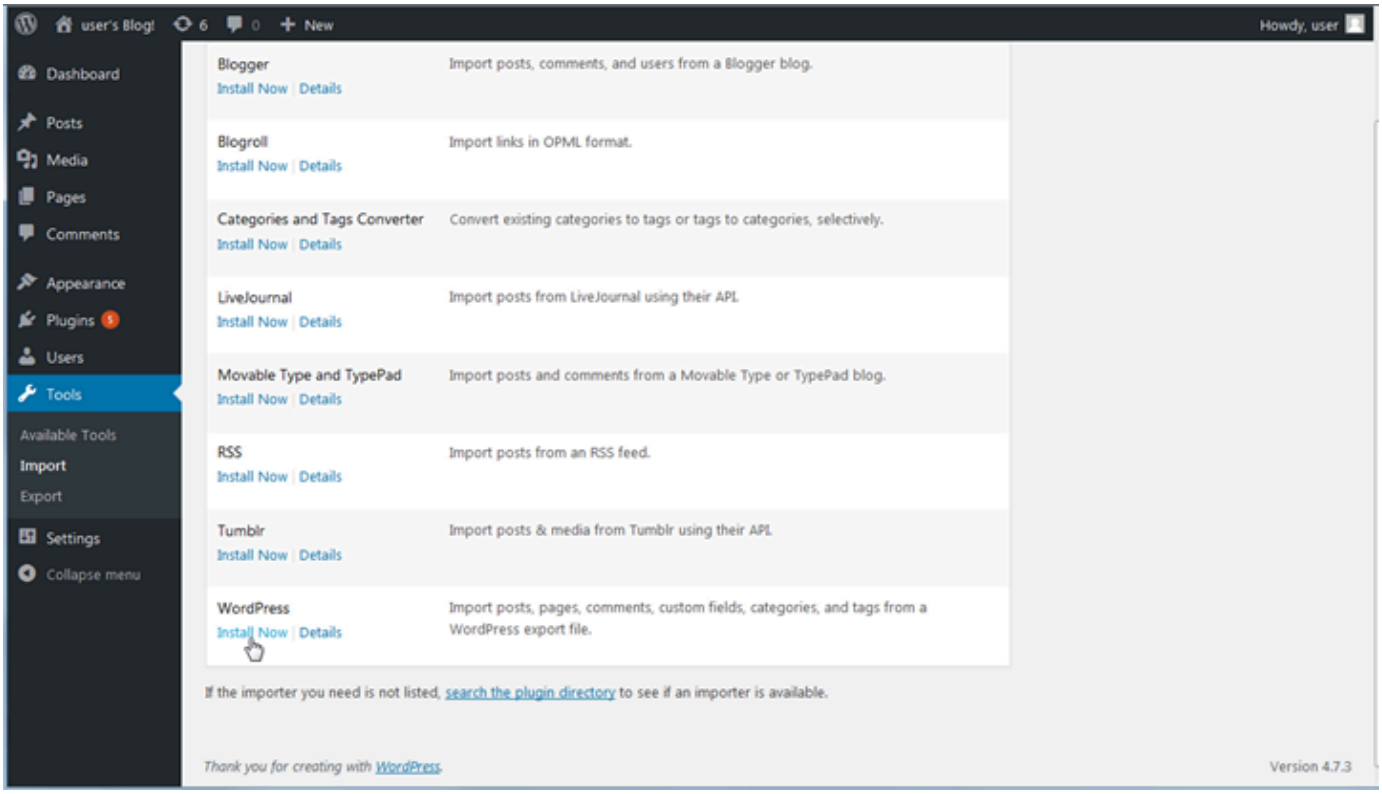
이제 관리 작업을 수행할 수 있는 WordPress 웹사이트의 관리 대시보드에 로그인되었습니다. WordPress 웹사이트 관리에 대한 자세한 내용은 WordPress 설명서의 WordPress Codex를 참조하세요.



4단계: XML 파일을 새 Lightsail 블로그로 가져오기

새 Lightsail 인스턴스에서 WordPress Dashboard에 성공적으로 로그인했으면 다음 단계에 따라 XML 파일을 새 Lightsail 블로그로 가져옵니다.

1. 새 Lightsail 인스턴스의 WordPress Dashboard에서 도구(Tools)를 선택합니다.
2. Import(가져오기)를 선택한 다음 Install Now(지금 설치)를 선택하여 WordPress 가져오기 도구를 설치합니다.



3. 도구 설치가 완료되면 Run Importer(임포터 실행)를 선택하여 가져오기 도구를 실행합니다.
4. Import WordPress(WordPress 가져오기) 페이지에서 Browse(찾아보기)를 선택합니다.
5. 1단계: 기존 WordPress 블로그 백업에서 저장한 XML 파일을 찾은 다음 Open(열기)을 선택합니다.
6. Upload file and import(파일 업로드 및 가져오기)를 선택합니다.

   나머지 기본값을 수락하고 Submit(제출)을 선택합니다.

다음 단계

홈 아이콘 옆의 블로그를 선택한 다음 WordPress 대시보드에서 Visit Site(사이트 방문)를 선택하여 작업한 모든 내용을 확인할 수 있습니다. 브라우저에 IP 주소를 입력해도 블로그를 볼 수 있습니다.

몇 가지 남은 다음 단계는 아래와 같습니다.

- 도메인 이름 서버가 블로그의 새 버전을 가리키도록 DNS를 마이그레이션합니다.
- 새 블로그의 모양을 사용자 지정하거나 WordPress 플러그인을 설치합니다.
- 에서 WordPress 인스턴스 시작 및 구성 Lightsail

단계별 지침에 따라 WordPress 인스턴스를 시작 및 구성하고, HTTPS로 보호하고, 외부 데이터베이스 또는 스토리지 서비스에 연결하고, 기존 블로그를 Lightsail로 마이그레이션합니다. 이 자습서에서는 WordPress 관리자 자격 증명 획득, 플러그인 설치, DNS 및 도메인 설정 구성, Amazon S3, Amazon Aurora, Amazon SES와 AWS 서비스 같은 다른와 통합과 같은 필수 작업을 다룹니다. 이 가이드를 따르면 Lightsail 플랫폼에서 안전하고 확장 가능하며 고성능인 WordPress 웹 사이트를 쉽게 설정하고 관리할 수 있습니다.

Lightsail에서 다중 사이트를 사용하여 여러 WordPress 사이트 관리

이 섹션에서는 Amazon Lightsail에서 WordPress Multisite 인스턴스의 블로그 관리와 관련된 다음 주제를 다룹니다.

주제

- Lightsail의 WordPress Multisite에 블로그를 도메인으로 추가
- Lightsail의 WordPress Multisite에 블로그를 하위 도메인으로 추가
- Lightsail의 WordPress Multisite 인스턴스에 대한 기본 도메인 정의

Lightsail의 WordPress Multisite에 블로그를 도메인으로 추가

Amazon Lightsail의 WordPress Multisite 인스턴스는 해당 인스턴스 내에 생성하는 각 블로그 사이트에 대해 여러 도메인 또는 하위 도메인을 사용하도록 설계되었습니다. 이 안내서에서는 WordPress Multisite 인스턴스에 기본 블로그의 기본 도메인과 다른 도메인을 사용하여 블로그 사이트를 추가하는 방법을 보여줍니다. 예를 들어, 기본 블로그의 기본 도메인이 `example.com`인 경우 동일한 인스턴스의 `another-example.com` 및 `third-example.com` 도메인을 사용하여 새 블로그 사이트를 생성할 수 있습니다.

> **Note**
>
> 또한 하위 도메인을 사용하여 WordPress Multisite 인스턴스에 사이트를 추가할 수 있습니다. 자세한 내용은 WordPress Multisite 인스턴스에 블로그를 하위 도메인으로 추가를 참조하세요.

사전 조건

다음 사전 조건을 표시된 순서로 완료합니다.

1. Lightsail에 WordPress Multisite 인스턴스를 생성합니다. 자세한 내용은 인스턴스 생성을 참조하세요.
2. 고정 IP를 생성하여 Lightsail의 WordPress Multisite 인스턴스에 연결합니다. 자세한 내용은 고정 IP를 생성하여 인스턴스에 연결을 참조하세요.
3. DNS 영역을 생성한 다음 WordPress Multisite 인스턴스에 연결된 고정 IP를 가리키도록 하여 도메인을 Lightsail에 추가합니다. 자세한 내용은 DNS 영역을 생성하여 도메인의 DNS 레코드 관리를 참조하세요.
4. WordPress Multisite 인스턴스에 대한 기본 도메인을 정의합니다. 자세한 내용은 WordPress Multisite 인스턴스에 대한 기본 도메인 정의를 참조하세요.

WordPress Multisite 인스턴스에 블로그를 도메인으로 추가

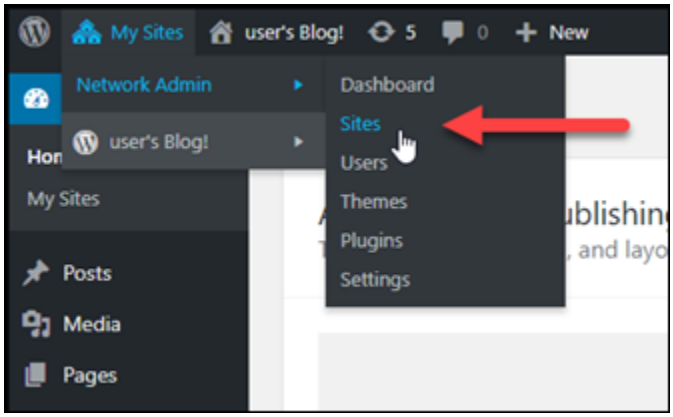
이 단계를 완료하여 WordPress Multisite 인스턴스에 기본 블로그의 기본 도메인과 다른 도메인을 사용하는 블로그 사이트를 생성합니다.

> **Important**
>
> 이 단계를 수행하기 전에 이 가이드의 사전 조건 섹션에 나열된 4단계를 완료해야 합니다.

1. WordPress Multisite 인스턴스의 관리 대시보드에 로그인합니다.

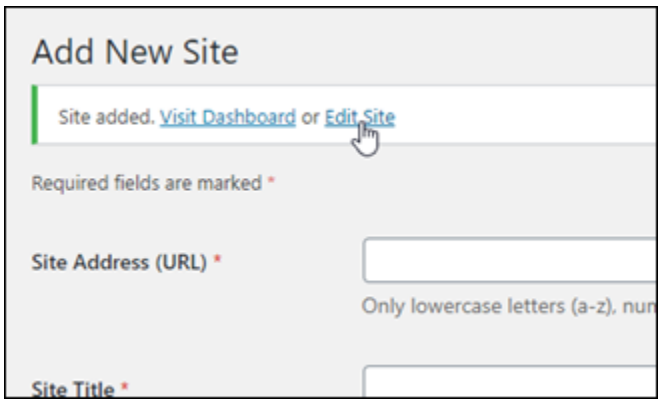
   > **Note**
   >
   > 자세한 내용은 Bitnami 인스턴스에 대한 애플리케이션 사용자 이름과 암호 가져오기를 참조하세요.

2. 상단 탐색 창에서 My Sites(내 사이트), Network Admin(네트워크 관리) 및 Sites(사이트)를 차례대로 선택합니다.

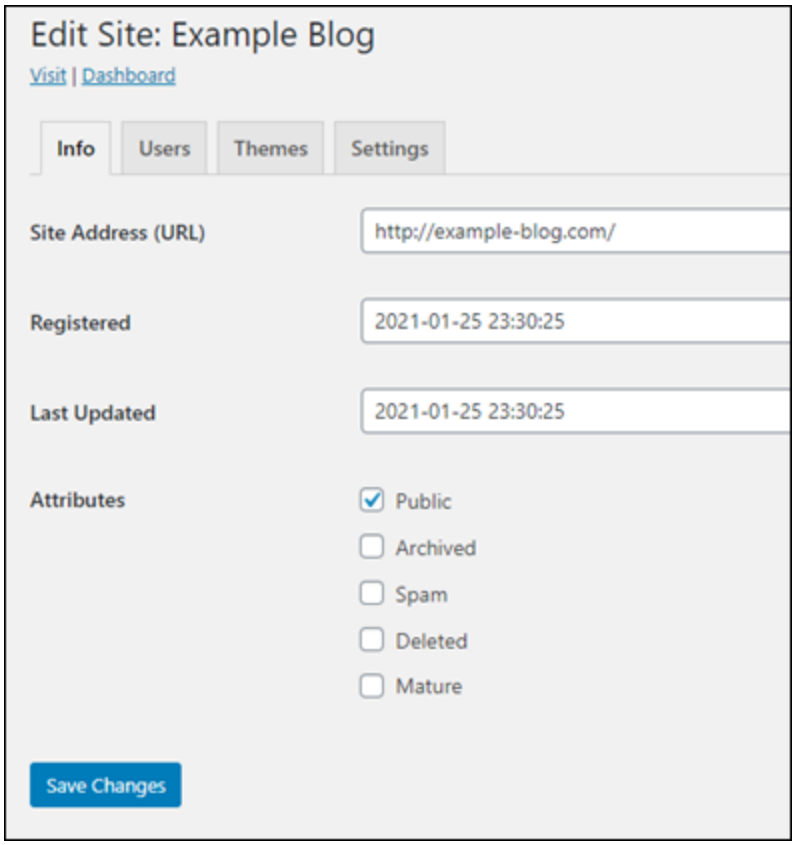


3. Add New(새로 추가)를 선택하여 새 블로그 사이트를 추가합니다.
4. 사이트 주소(URL)(Site Address (URL)) 텍스트 상자에 사이트 주소를 입력합니다. 이 주소는 새 블로그 사이트에 사용될 도메인입니다. 예를 들어, 새 블로그 사이트에서 `example-blog.com`을 도메인으로 사용할 경우 사이트 주소(URL)(Site Address (URL)) 텍스트 상자에 `example-blog`를 입력합니다. 페이지에 표시된 기본 도메인 접미사를 무시합니다.

5. 사이트 제목을 입력하고 사이트 언어를 선택한 다음 관리 이메일을 입력합니다.
6. Add Site(사이트 추가)를 선택합니다.
7. 페이지에 표시되는 확인 배너에서 사이트 편집(Edit Site)을 선택합니다. 그러면 최근에 생성한 사이트의 세부 정보를 편집하도록 리디렉션됩니다.

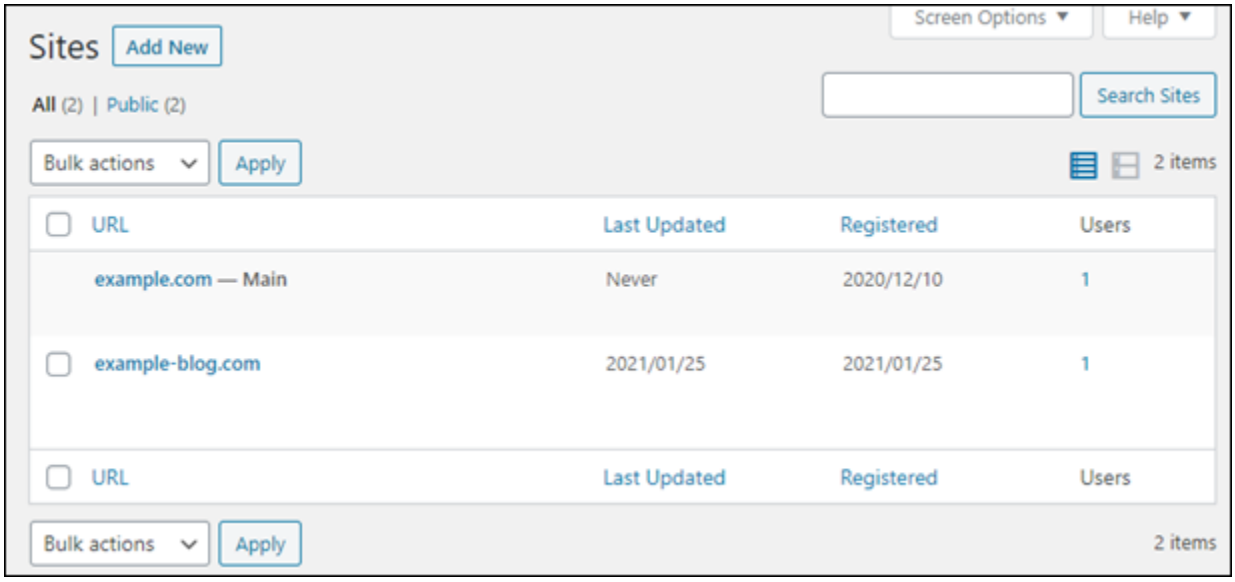


8. 사이트 편집(Edit Site) 페이지에서 사이트 주소(URL)(Site Address (URL)) 텍스트 상자에 나열된 하위 도메인을 사용할 정점 도메인으로 변경합니다. 이 예시에서는 `http://example-blog.com`을 지정합니다.



9. 변경 사항 저장(Save Changes)을 선택합니다.

   이때 새 블로그 사이트가 WordPress Multisite 인스턴스에 생성되지만, 도메인이 아직 새 블로그 사이트로 라우팅되도록 구성되지 않았습니다. 다음 단계로 이동하여 주소 레코드(A 레코드)를 도메인의 DNS 영역에 추가합니다.



주소 레코드(A 레코드)를 도메인의 DNS 영역에 추가

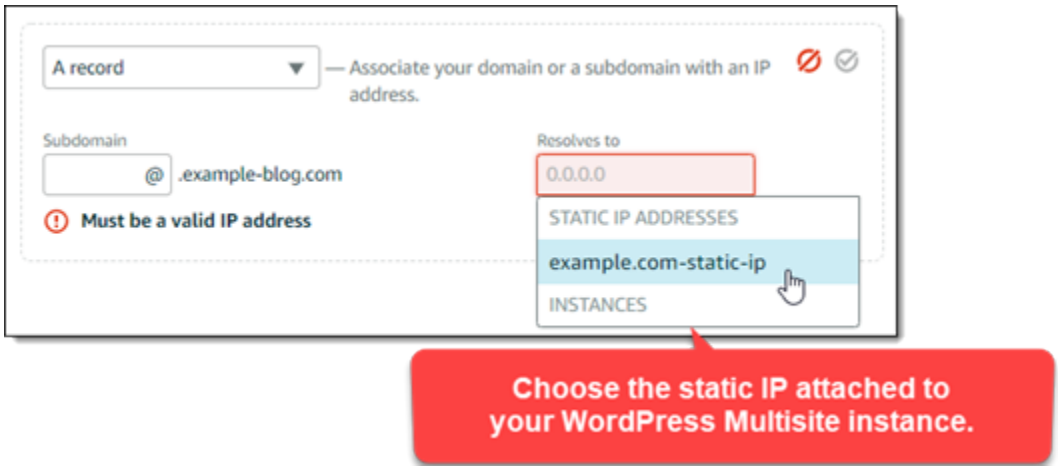
이 단계를 완료하여 새 블로그 사이트에 대한 도메인이 WordPress Multisite 인스턴스를 가리키도록 합니다. WordPress Multisite 인스턴스에 생성한 모든 블로그 사이트에 대해 이 단계를 수행해야 합니다.

설명을 위해 여기서는 Lightsail DNS 영역을 사용합니다. 하지만 이 단계는 일반적으로 도메인 등록 대행자가 호스팅한 다른 DNS 영역과 비슷할 수 있습니다.

> **Important**
>
> Lightsail 콘솔에 최대 6개의 DNS 영역을 생성할 수 있습니다. 추가 DNS 영역이 필요한 경우 Amazon Route 53을 사용하여 도메인의 DNS 레코드를 관리하는 것이 좋습니다. 자세한 내용은 Amazon Route 53를 기존 도메인의 DNS 서비스로 지정을 참조하세요.

1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 도메인 및 DNS를 선택합니다.
3. 페이지의 DNS 영역 섹션에서 새 블로그 사이트의 도메인에 대한 DNS 영역을 선택합니다.
4. DNS 영역 편집기에서 DNS records(DNS 레코드)를 선택합니다. 그런 다음 Add record(레코드 추가)를 선택합니다.

5. 레코드 유형 드롭다운 메뉴에서 A record(A 레코드)를 선택합니다.
6. Record name(레코드 이름) 텍스트 상자에 "at"(@) 기호를 입력하여 도메인 루트에 대한 레코드를 생성합니다.
7. Resolves to(확인) 텍스트 상자에서 WordPress Multisite 인스턴스에 연결된 고정 IP 주소를 선택합니다.



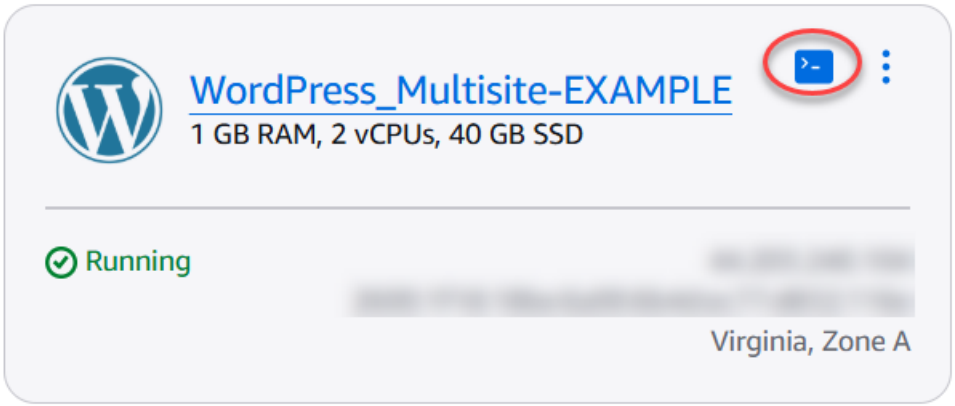
8. 저장 아이콘을 선택합니다.

   인터넷의 DNS를 통해 변경 내용을 전파한 후 도메인이 WordPress Multisite 인스턴스의 새 블로그 사이트로 트래픽 경로를 지정합니다.

쿠키 지원을 사용하여 블로그 사이트에 대한 로그인 허용

블로그 사이트를 WordPress Multisite 인스턴스에 도메인으로 추가할 때 쿠키 지원을 사용하려면 인스턴스의 WordPress 구성(`wp-config`) 파일을 업데이트해야 합니다. 쿠키 지원을 사용하지 않으면 사용자가 블로그 사이트의 WordPress 관리 대시보드에 로그인하려고 할 때 '오류: 쿠키가 차단되었거나 지원되지 않음' 오류가 발생할 수 있습니다.

1. Lightsail 콘솔에 로그인합니다.
2. Lightsail 홈 페이지에서 WordPress Multisite 인스턴스에 대한 SSH 빠른 연결 아이콘을 선택합니다.



3. Lightsail 브라우저 기반 SSH 세션이 연결되면 다음 명령을 입력하여 Vim을 통해 인스턴스의 `wp-config.php` 파일을 열고 편집합니다.

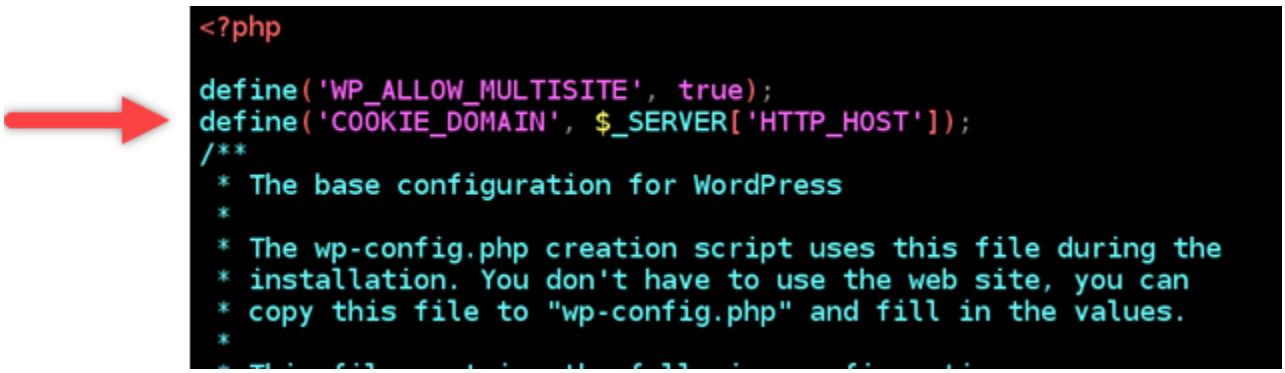
```
sudo vim /opt/bitnami/wordpress/wp-config.php
```

> **Note**
>
> 이 명령이 실패하면 이전 WordPress Multisite 인스턴스 버전을 사용하고 있는 것일 수 있습니다. 이 경우 대신 다음 명령을 실행합니다.
>
> ```
> sudo vim /opt/bitnami/wordpress/wp-config.php
> ```

4. I를 눌러 Vim을 삽입 모드를 설정합니다.
5. `define('WP_ALLOW_MULTISITE', true);` 텍스트 행 아래에 다음 텍스트 행을 추가합니다.

```
define('COOKIE_DOMAIN', $_SERVER['HTTP_HOST']);
```

완료되면 파일이 다음과 같이 표시됩니다.



6. Esc 키를 눌러 Vim에서 삽입 모드를 종료한 다음 `:wq!`를 입력하고 Enter 키를 눌러 편집한 내용을 저장(쓰기)하고 Vim을 종료합니다.
7. 다음 명령을 입력하여 WordPress 인스턴스의 기본 서비스를 다시 시작합니다.

```
sudo /opt/bitnami/ctlscript.sh restart
```

이제 WordPress Multisite 인스턴스에서 쿠키를 사용하도록 설정하며, 블로그 사이트에 로그인하려는 사용자를 대상으로 '오류: 쿠키가 차단되었거나 지원되지 않음' 오류가 발생하지 않습니다.

다음 단계

WordPress Multisite 인스턴스에 블로그를 도메인으로 추가한 후에는 WordPress Multisite 인스턴스 관리에 익숙해지도록 하는 것이 좋습니다. 자세한 내용은 WordPress 문서에서 Multisite 네트워크 관리를 참조하세요.

Lightsail의 WordPress Multisite에 블로그를 하위 도메인으로 추가

Amazon Lightsail의 WordPress Multisite 인스턴스는 해당 인스턴스 내에 생성하는 각 블로그 사이트에 대해 여러 도메인 또는 하위 도메인을 사용하도록 설계되었습니다. 이 안내서에서는 블로그 사이트를 WordPress Multisite 인스턴스의 하위 도메인으로 추가하는 방법을 보여줍니다. 예를 들어, 기본 블로그의 기본 도메인이 `example.com`인 경우 동일한 인스턴스의 `earth.example.com` 및 `moon.example.com` 하위 도메인을 사용하여 새 블로그 사이트를 생성할 수 있습니다.

> **Note**
>
> 또한 도메인을 사용하여 WordPress Multisite 인스턴스에 사이트를 추가할 수 있습니다. 자세한 내용은 WordPress Multisite 인스턴스에 블로그를 도메인으로 추가를 참조하세요.

사전 조건

다음 사전 조건을 표시된 순서로 완료합니다.

1. WordPress Multisite 인스턴스를 생성합니다. 자세한 내용은 인스턴스 생성을 참조하세요.
2. 고정 IP를 생성하여 WordPress Multisite 인스턴스에 연결합니다. 자세한 내용은 고정 IP를 생성하여 인스턴스에 연결을 참조하세요.

3. DNS 영역을 생성한 다음 WordPress Multisite 인스턴스에 연결된 고정 IP를 가리키도록 하여 도메인을 Lightsail에 추가합니다. 자세한 내용은 DNS 영역을 생성하여 도메인의 DNS 레코드 관리를 참조하세요.
4. WordPress Multisite 인스턴스에 대한 기본 도메인을 정의합니다. 자세한 내용은 WordPress Multisite 인스턴스에 대한 기본 도메인 정의를 참조하세요.

WordPress Multisite 인스턴스에 블로그를 하위 도메인으로 추가

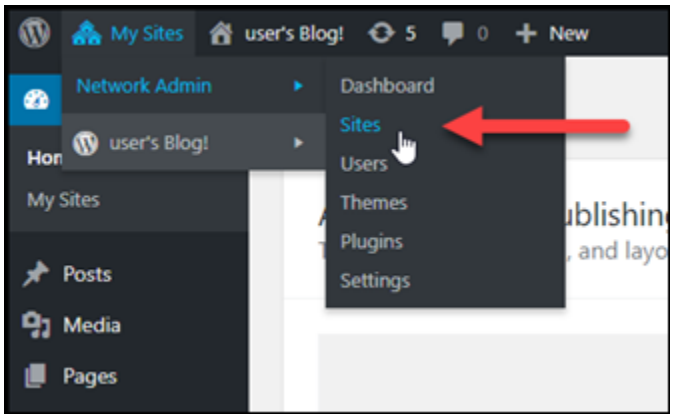
이 단계를 완료하여 WordPress Multisite 인스턴스에 기본 블로그에 대한 기본 도메인의 하위 도메인을 사용하는 새 블로그를 생성합니다.

> **Important**
>
> 이 단계를 수행하기 전에 이 가이드의 사전 조건 섹션에 나열된 4단계를 완료해야 합니다.

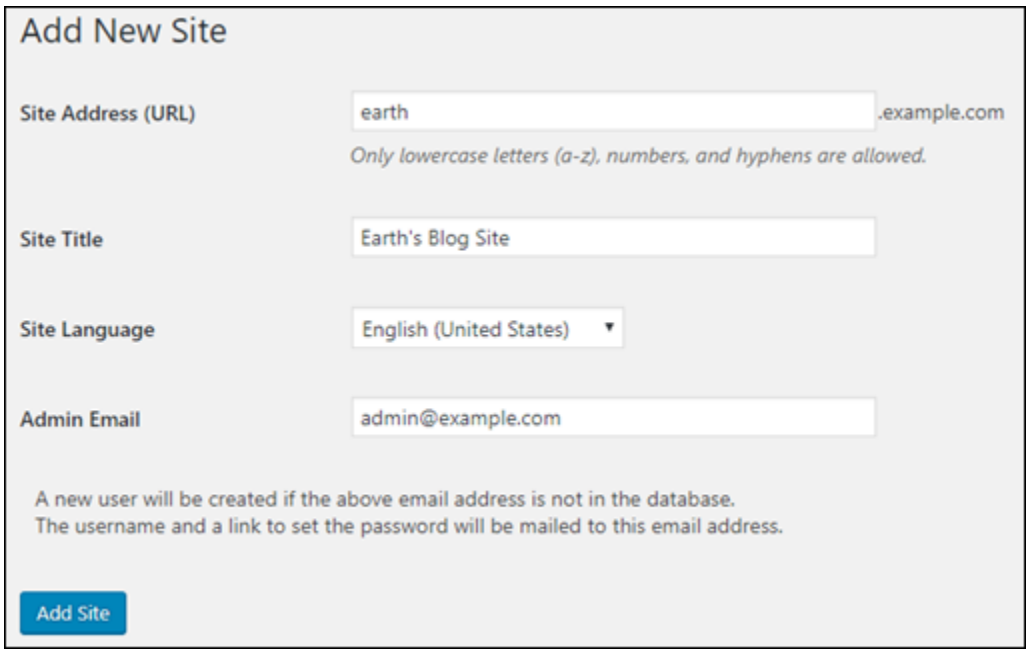
1. WordPress Multisite 인스턴스의 관리 대시보드에 로그인합니다.

   > **Note**
   >
   > 자세한 내용은 Bitnami 인스턴스에 대한 애플리케이션 사용자 이름과 암호 가져오기를 참조하세요.

2. 상단 탐색 창에서 My Sites(내 사이트), Network Admin(네트워크 관리) 및 Sites(사이트)를 차례대로 선택합니다.

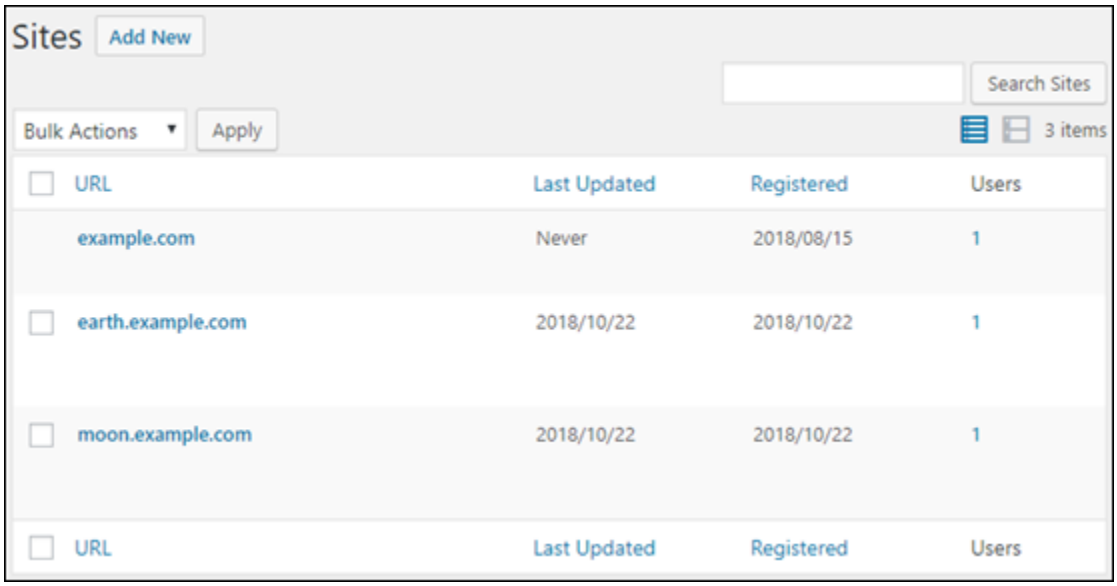


3. Add New(새로 추가)를 선택하여 새 블로그 사이트를 추가합니다.
4. 새 블로그 사이트에 사용될 하위 도메인인 사이트 주소를 입력합니다.



5. 사이트 제목을 입력하고 사이트 언어를 선택한 다음 관리 이메일을 입력합니다.
6. Add Site(사이트 추가)를 선택합니다.

   이때 새 블로그 사이트가 WordPress Multisite 인스턴스에 생성되지만, 하위 도메인이 아직 새 블로그 사이트로 라우팅되도록 구성되지 않았습니다. 다음 단계로 이동하여 주소 레코드(A 레코드)를 도메인의 DNS 영역에 추가합니다.

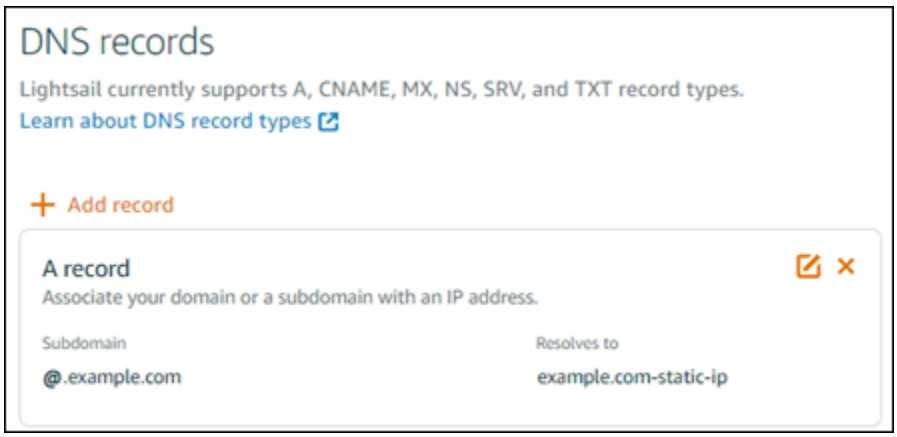


주소 레코드(A 레코드)를 도메인의 DNS 영역에 추가

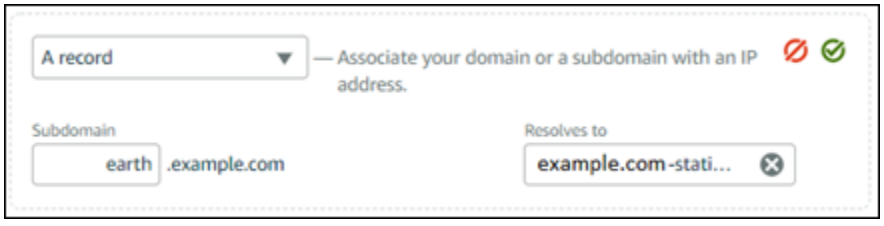
이 단계를 완료하여 새 블로그 사이트에 대한 하위 도메인이 WordPress Multisite 인스턴스를 가리키도록 합니다. WordPress Multisite 인스턴스에 생성한 모든 블로그 사이트에 대해 이 단계를 수행해야 합니다.

설명을 위해 여기서는 Lightsail DNS 영역을 사용합니다. 하지만 이 단계는 일반적으로 도메인 등록 대행자가 호스팅한 다른 DNS 영역과 비슷할 수 있습니다.

1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 도메인 및 DNS를 선택합니다.
3. 페이지의 DNS 영역 섹션에서 WordPress Multisite 인스턴스에 대한 기본 도메인으로 정의한 도메인에 대한 DNS 영역을 선택합니다.
4. DNS 영역 편집기에서 DNS records(DNS 레코드)를 선택합니다. 그런 다음 Add record(레코드 추가)를 선택합니다.



5. 레코드 유형 드롭다운 메뉴에서 A record(A 레코드)를 선택합니다.
6. Record name(레코드 이름) 텍스트 상자에 WordPress Multisite 인스턴스에 새 블로그 사이트를 생성할 때 사이트 주소로 지정된 하위 도메인을 입력합니다.
7. Resolves to(확인) 텍스트 상자에서 WordPress Multisite 인스턴스에 연결된 고정 IP 주소를 선택합니다.



8. 저장 아이콘을 선택합니다.

더 이상 수행할 작업이 없습니다. 인터넷의 DNS를 통해 전파를 변경한 후 도메인이 WordPress Multisite 인스턴스의 새 블로그 사이트로 리디렉션됩니다.

다음 단계

블로그를 WordPress Multisite 인스턴스에 하위 도메인으로 추가한 후에는 WordPress Multisite 관리에 익숙해지도록 하는 것이 좋습니다. 자세한 내용은 WordPress 문서에서 Multisite 네트워크 관리를 참조하세요.

Lightsail의 WordPress Multisite 인스턴스에 대한 기본 도메인 정의

Amazon Lightsail의 WordPress Multisite 인스턴스는 해당 인스턴스 내에 생성하는 각 블로그 사이트에 대해 여러 도메인 또는 하위 도메인을 사용하도록 설계되었습니다. 따라서 WordPress Multisite 인스턴스의 기본 블로그에 사용할 기본 도메인을 정의해야 합니다.

사전 조건

다음 사전 조건을 표시된 순서로 완료합니다.

1. Lightsail에 WordPress Multisite 인스턴스를 생성합니다. 자세한 내용은 인스턴스 생성을 참조하세요.
2. 고정 IP를 생성하여 Lightsail의 WordPress Multisite 인스턴스에 연결합니다. 자세한 내용은 고정 IP를 생성하여 인스턴스에 연결을 참조하세요.

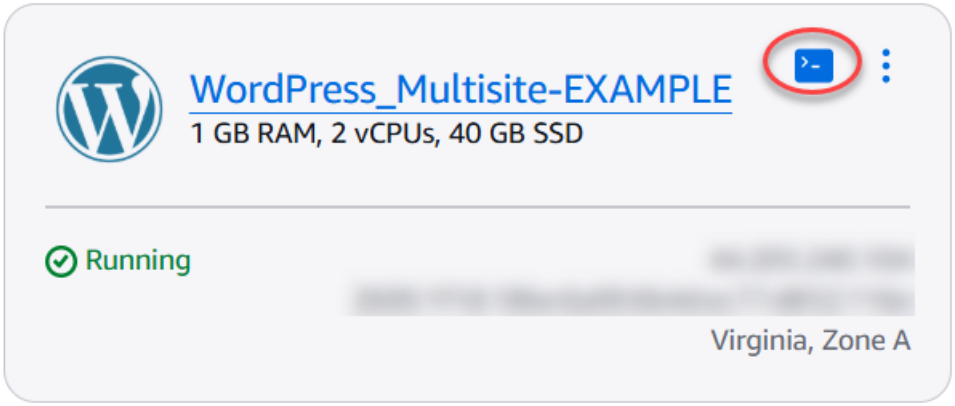
   > **Important**
   >
   > WordPress Multisite 인스턴스는 고정 IP를 첨부하면 재부팅해야 합니다. 그러면 연결된 새 고정 IP를 인스턴스에서 인식할 수 있습니다.

3. DNS 영역을 생성한 다음 WordPress Multisite 인스턴스에 연결된 고정 IP를 가리키도록 하여 도메인을 Lightsail에 추가합니다. 자세한 내용은 DNS 영역을 생성하여 도메인의 DNS 레코드 관리를 참조하세요.
4. 인터넷의 DNS를 통해 DNS 변경 사항이 전파될 때까지 기다립니다. 그런 다음 이 안내서의 WordPress Multisite 인스턴스에 대한 기본 도메인 정의 단원으로 계속 진행할 수 있습니다.

WordPress Multisite 인스턴스에 대한 기본 도메인 정의

이 단계를 완료하여 `example.com`과 같은 도메인이 WordPress Multisite 인스턴스의 기본 블로그로 리디렉션되도록 합니다.

1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 WordPress Multisite 인스턴스에 대한 SSH 빠른 연결 아이콘을 선택합니다.



3. 다음 명령을 입력하여 WordPress Multisite 인스턴스에 대한 기본 도메인 이름을 정의합니다. *<domain>*을 WordPress Multisite에 대한 올바른 도메인 이름으로 바꿔야 합니다.

```
sudo /opt/bitnami/configure_app_domain --domain <domain>
```

예제:

```
sudo /opt/bitnami/configure_app_domain --domain example.com
```

> **Note**
>
> 이 명령이 실패하면 이전 WordPress Multisite 인스턴스 버전을 사용하고 있는 것일 수 있습니다. 다음 명령을 실행하는 대신, *<domain>*을 WordPress Multisite에 대한 올바른 도메인 이름으로 바꿔야 합니다.
>
> ```
> cd /opt/bitnami/apps/wordpress
> sudo ./bnconfig --machine_hostname <domain>
> ```
>
> 해당 명령을 실행하고 나서 다음 명령을 입력하여 서버가 재시작될 때마다 bnconfig 도구가 자동으로 실행되지 않도록 합니다.

```
sudo mv bnconfig bnconfig.disabled
```

이때 정의한 도메인을 탐색하면 WordPress Multisite 인스턴스의 기본 블로그로 리디렉션되어야 합니다.

다음 단계

WordPress Multisite 인스턴스에 대한 기본 도메인을 정의한 후에는 다음 단계를 완료합니다.

- WordPress Multisite 인스턴스에 블로그를 하위 도메인으로 추가
- WordPress Multisite 인스턴스에 블로그를 도메인으로 추가

단계별 지침에 따라 별도의 도메인 또는 하위 도메인을 사용하여 새 블로그 사이트를 추가하는 방법과 WordPress Multisite 인스턴스에서 기본 블로그의 기본 도메인을 정의하는 방법을 알아봅니다.

이 가이드에서는 WordPress Multisite 인스턴스 생성, 고정 IP 연결, DNS 영역 생성, 기본 도메인 구성과 같은 사전 조건을 다룹니다. 그런 다음 블로그를 도메인 또는 하위 도메인으로 추가하고, DNS 레코드를 업데이트하고, 쿠키 지원을 활성화하고, 기타 필요한 구성을 수행하기 위한 세부 단계를 제공합니다. 이 가이드를 따르면 각 블로그 사이트에 대해 별도의 도메인 또는 하위 도메인을 사용하는 유연성을 활용하여 WordPress Multisite 인스턴스 내에서 여러 블로그를 효과적으로 관리하고 구성할 수 있습니다.

Let's Encrypt를 사용하여 Lightsail 리소스에 대해 암호화된 통신 활성화

이 가이드에서는 Amazon Lightsail의 Let's Encrypt와 관련된 다음 주제를 다룹니다. 시작하기 전에 다음 사전 조건을 완료해야 합니다.

사전 조건

- LAMP, Nginx 또는 WordPress를 실행하는 Lightsail 인스턴스를 생성합니다.
- 도메인 이름을 등록하고 DNS 레코드를 편집하는 데 필요한 액세스 권한을 얻습니다.
- Lightsail 브라우저 기반 SSH 터미널 또는 자체 SSH 클라이언트를 사용합니다.

주제

- Lightsail LAMP 인스턴스와 함께 Let's Encrypt SSL 인증서 보호
- Let's Encrypt SSL/TLS를 사용하여 Lightsail NGINX 웹 사이트 보호
- Lightsail WordPress 인스턴스와 함께 Let's Encrypt SSL 인증서 보호

Lightsail LAMP 인스턴스와 함께 Let's Encrypt SSL 인증서 보호

Amazon Lightsail을 사용하면 Lightsail 로드 밸런서를 사용하여 SSL/TLS를 사용하는 웹사이트 및 애플리케이션을 쉽게 보호할 수 있습니다. 그러나 Lightsail 로드 밸런서를 사용하는 것이 일반적으로 올바른 선택이 아닐 수도 있습니다. 사이트에서 로드 밸런서가 제공하는 확장성 또는 내결함성을 필요로 하지 않거나 비용 최적화를 원할 수도 있습니다.

후자의 경우에는 Let's Encrypt를 사용하여 무료 SSL 인증서를 얻을 수 있으며 그런 경우 아무런 문제가 되지 않습니다. 이러한 인증서를 Lightsail 인스턴스와 통합할 수 있습니다. 이 자습서에서는 Certbot을 사용하여 Let's Encrypt 와일드카드 인증서를 요청하고 이를 LAMP 인스턴스와 통합하는 방법을 보여줍니다.

Important

- Bitnami 인스턴스에서 사용하는 Linux 배포는 2020년 7월에 Ubuntu에서 Debian으로 변경되었습니다. 이러한 변화로 인스턴스의 Linux 배포에 따라 자습서의 일부 단계가 달라집니다. 변경 후 생성된 모든 Bitnami 블루프린트 인스턴스는 Debian Linux 배포를 사용합니다. 변경 전에 생성된 인스턴스는 Ubuntu Linux 배포를 계속 사용합니다. 인스턴스의 배포를 확인하려면 `uname -a` 명령을 실행합니다. 응답은 Ubuntu 또는 Debian을 인스턴스의 Linux 배포로 표시합니다.
- Bitnami는 많은 스택에 대한 파일 구조를 수정하는 중입니다. 이 자습서의 파일 경로는 Bitnami 스택이 네이티브 Linux 시스템 패키지(접근법 A)를 사용하는지 또는 자체 포함 설치(접근법 B)인지 여부에 따라 달라질 수 있습니다. Bitnami 설치 유형 및 따라야 할 접근 방식을 식별하려면 다음 명령을 실행합니다.

  ```
  test ! -f "/opt/bitnami/common/bin/openssl" && echo "Approach
  A: Using system packages." || echo "Approach B: Self-contained
  installation."
  ```

목차

- 1단계: 필수 구성 요소 완성
- 2단계: 인스턴스에 Certbot 설치
- 3단계: Let's Encrypt SSL 와일드카드 인증서 요청
- 4단계: 도메인의 DNS 영역에 TXT 레코드 추가
- 5단계: TXT 레코드가 전파되었는지 확인
- 6단계: Let's Encrypt SSL 인증서 요청 완료
- 7단계 Let's Encrypt 인증서 파일을 Apache 서버 디렉터리에 연결하는 링크 생성
- 8단계: 웹 애플리케이션에 대해 HTTP에서 HTTPS로의 리디렉션 구성
- 9단계: 90일마다 Let's Encrypt 인증서 갱신

1단계: 필수 구성 요소 완성

아직 수행하지 않은 경우 다음 사전 조건을 완료하세요.

- Lightsail에서 LAMP 인스턴스를 생성합니다. 자세한 내용은 인스턴스 생성을 참조하세요.
- DNS 레코드를 편집하려면 도메인 이름을 등록하고 관리 액세스 권한을 얻습니다. 자세한 내용은 Amazon Lightsail DNS를 참조하세요.

> **Note**
>
> Lightsail DNS 영역을 사용하여 도메인의 DNS 레코드를 관리하는 것이 좋습니다. 자세한 내용은 DNS 영역을 생성하여 도메인의 DNS 레코드 관리를 참조하세요.

- Lightsail 콘솔의 브라우저 기반 SSH 터미널을 사용하여 이 자습서의 단계를 수행합니다. 그러나 PuTTY와 같은 자체 SSH 클라이언트를 사용할 수도 있습니다. PuTTY 구성에 대한 자세한 내용은 SSH를 사용하여 연결할 PuTTY 다운로드 및 설정을 참조하세요.

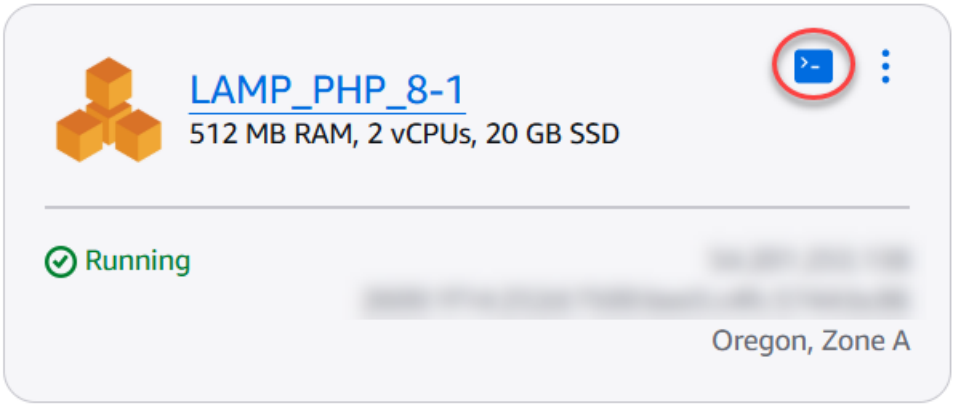
사전 조건을 완료한 후 이 자습서의 다음 단원으로 계속 진행합니다.

2단계: 인스턴스에 Certbot 설치

Certbot은 Let's Encrypt에서 인증서를 요청하고 이를 웹 서버에 배포하는 데 사용되는 클라이언트입니다. Let's Encrypt는 ACME 프로토콜을 사용하여 인증서를 발행하고 Certbot은 Let's Encrypt와 상호 작용하는 ACME 사용 클라이언트입니다.

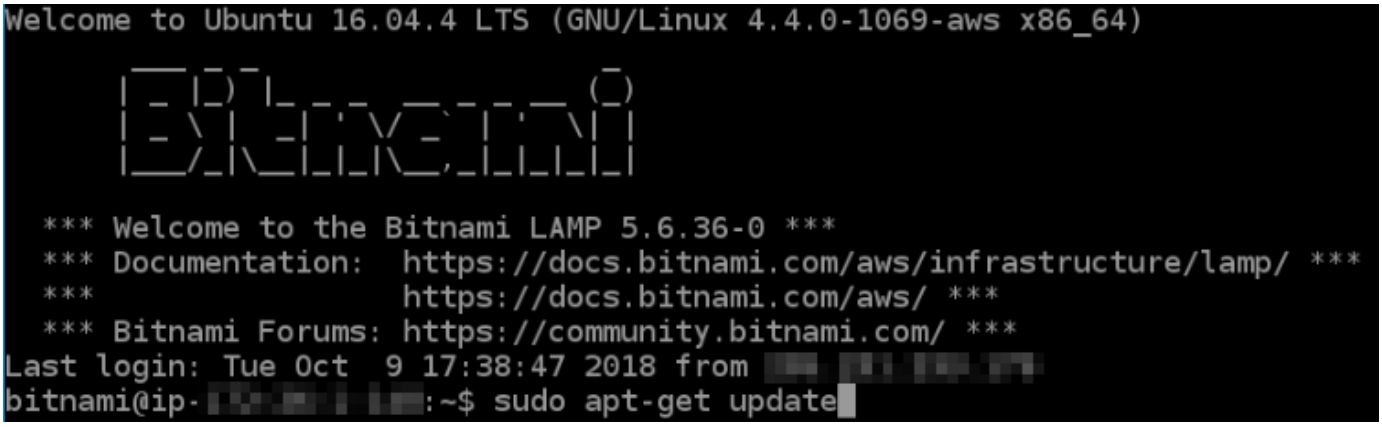
Lightsail 인스턴스에 Certbot을 설치하려면

1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 연결할 인스턴스에 대한 SSH 빠른 연결 아이콘을 선택합니다.



3. Lightsail 브라우저 기반 SSH 세션이 연결되면 다음 명령을 입력하여 인스턴스의 패키지를 업데이트합니다.

```
sudo apt-get update
```



4. 다음 명령을 입력하여 소프트웨어 속성 패키지를 설치합니다. Certbot의 개발자는 Personal Package Archive(PPA)를 사용하여 Certbot을 배포합니다. 소프트웨어 속성 패키지를 사용하면 PPA를 사용하여 보다 효율적으로 작업할 수 있습니다.

```
sudo apt-get install software-properties-common
```

> **Note**
>
> `sudo apt-get install` 명령을 실행할 때 `Could not get lock` 오류가 발생하면 약 15분 정도 기다린 후 다시 시도합니다. 이 오류는 Apt 패키지 관리 도구를 사용하여 무인 업그레이드를 설치하는 cron 작업으로 인해 발생할 수 있습니다.

5. 다음 명령을 입력하여 로컬 apt 리포지토리에 Certbot을 추가합니다.

> **Note**
>
> 5단계는 Ubuntu Linux 배포를 사용하는 인스턴스에만 적용됩니다. 인스턴스가 Debian Linux 배포를 사용하는 경우 이 단계를 건너뜁니다.

```
sudo apt-add-repository ppa:certbot/certbot -y
```

6. 다음 명령을 입력하여 새 리포지토리를 포함하도록 apt를 업데이트합니다.

```
sudo apt-get update -y
```

7. 다음 명령을 입력하여 Certbot을 설치합니다.

```
sudo apt-get install certbot -y
```

이제 Certbot이 Lightsail 인스턴스에 설치됩니다.

8. 브라우저 기반 SSH 터미널 창을 열린 상태로 유지하세요. 이 자습서의 뒷부분에서 다시 해당 세션으로 돌아옵니다. 이 자습서의 다음 섹션으로 계속 진행합니다.

3단계: Let's Encrypt SSL 와일드카드 인증서 요청

Let's Encrypt에서 인증서를 요청하는 프로세스를 시작합니다. Certbot을 사용하여 도메인 및 하위 도메인에 대해 단일 인증서를 사용할 수 있는 와일드카드 인증서를 요청합니다. 예를 들어 `example.com` 최상위 도메인과 `blog.example.com` 및 `stuff.example.com` 하위 도메인에 대해 단일 와일드카드 인증서가 작동합니다.

Let's Encrypt SSL 와일드카드 인증서를 요청하려면

1. 이 자습서의 2단계에서 사용된 것과 동일한 브라우저 기반 SSH 터미널 창에서 다음 명령을 입력하여 도메인에 대해 환경 변수를 설정합니다. 이제 명령을 더 효율적으로 복사하고 붙여 넣어 인증서를 얻을 수 있습니다.

   ```
   DOMAIN=Domain
   ```

   ```
   WILDCARD=*.$DOMAIN
   ```

   명령에서 *Domain*을 등록된 도메인 이름으로 바꿉니다.

   예제:

   ```
   DOMAIN=example.com
   ```

   ```
   WILDCARD=*.$DOMAIN
   ```

2. 다음 명령을 입력하여 변수가 올바른 값을 반환하는지 확인합니다.

   ```
   echo $DOMAIN && echo $WILDCARD
   ```

   다음과 유사한 결과가 출력되어야 합니다.

3. 다음 명령을 입력하여 대화형 모드에서 Certbot을 시작합니다. 이 명령은 DNS 문제에 대해 수동 권한 부여 방법을 사용하여 도메인 소유권을 확인하도록 Certbot에 지시하며, 최상위 도메인 및 하위 도메인에 대한 와일드카드 인증서를 요청합니다.

   ```
   sudo certbot -d $DOMAIN -d $WILDCARD --manual --preferred-challenges dns certonly
   ```

4. 메시지가 표시되면 갱신 및 보안 고지 사항에 사용될 이메일 주소를 입력합니다.

5. Let's Encrypt 서비스 계약 조건을 읽습니다. 모두 읽은 다음 동의하면 A를 누릅니다. 동의하지 않는 경우 Let's Encrypt 인증서를 얻을 수 없습니다.

6. 이메일 주소를 공유하라는 메시지와 IP 주소가 기록된다는 경고에 적절하게 응답합니다.
7. 이제 Let's Encrypt에 지정된 도메인을 소유하고 있는지 확인하라는 메시지가 표시됩니다. TXT 레코드를 도메인의 DNS 레코드에 추가하여 이 작업을 수행할 수 있습니다. 다음 예와 같이 한 세트의 TXT 레코드 값이 제공됩니다.

> **Note**
>
> 확인에 사용해야 하는 하나 또는 여러 개의 TXT 레코드를 Let's Encrypt에서 제공할 수 있습니다. 이 예에서는 확인에 사용할 두 개의 TXT 레코드가 제공되었습니다.

8. Lightsail 브라우저 기반 SSH 세션을 열린 상태로 유지하세요. 이 자습서의 뒷부분에서 다시 해당 세션으로 돌아옵니다. 이 자습서의 다음 섹션으로 계속 진행합니다.

4단계: 도메인의 DNS 영역에 TXT 레코드 추가

도메인의 DNS 영역에 TXT 레코드를 추가하면 도메인 소유권이 확인됩니다. 설명을 위해 Lightsail DNS 영역을 사용하겠습니다. 하지만 이 단계는 일반적으로 도메인 등록 대행자가 호스팅하는 다른 DNS 영역과 비슷할 수 있습니다.

> **Note**
>
> 도메인에 대해 Lightsail DNS 영역을 만드는 방법에 대한 자세한 내용은 DNS 영역을 생성하여 Lightsail에서 도메인의 DNS 레코드 관리를 참조하세요.

Lightsail의 도메인 DNS 영역에 TXT 레코드를 추가하려면

1. 왼쪽 탐색 창에서 도메인 및 DNS를 선택합니다.
2. 페이지의 DNS 영역 섹션에서 Certbot 인증서 요청에서 지정한 도메인의 DNS 영역을 선택합니다.
3. DNS 영역 편집기에서 DNS records(DNS 레코드)를 선택합니다.
4. 레코드 추가(Add record)를 선택합니다.
5. 레코드 유형(Record type) 드롭다운 메뉴에서 TXT 레코드를 선택합니다.
6. Let's Encrypt 인증서 요청에서 지정한 값을 레코드 이름(Record name) 및 응답(Responds with) 필드에 입력합니다.

> **Note**
>
> Lightsail 콘솔은 도메인의 정점 부분을 미리 채웁니다. 예를 들어, *_acme-challenge.example.com* 하위 도메인을 추가하려면 텍스트 상자에 *_acme-challenge*만 입력하면 되며, 레코드를 저장할 때 Lightsail이 `.example.com` 부분을 추가합니다.

7. 저장을 선택합니다.
8. 4~7단계를 반복하여 Let's Encrypt 인증서 요청에 지정된 두 번째 TXT 레코드 세트를 추가합니다.
9. Lightsail 콘솔 브라우저 창을 열린 상태로 유지하세요. 이 자습서의 뒷부분에서 다시 해당 세션으로 돌아옵니다. 이 자습서의 다음 섹션으로 계속 진행합니다.

5단계: TXT 레코드가 전파되었는지 확인

TXT 레코드가 인터넷의 DNS에 전파되었는지 확인하려면 MxToolbox 유틸리티를 사용합니다. DNS 레코드 전파는 DNS 호스팅 공급자 및 DNS 레코드에 대해 구성된 TTL(Time to Live)에 따라 다소 시간이 걸릴 수 있습니다. 이 단계를 완료하고 Certbot 인증서 요청을 계속하기 전에 TXT 레코드가 전파되었는지 확인하는 것이 중요합니다. 그렇지 않으면 인증서 요청이 실패합니다.

TXT 레코드가 인터넷 DNS에 전파되었는지 확인하려면

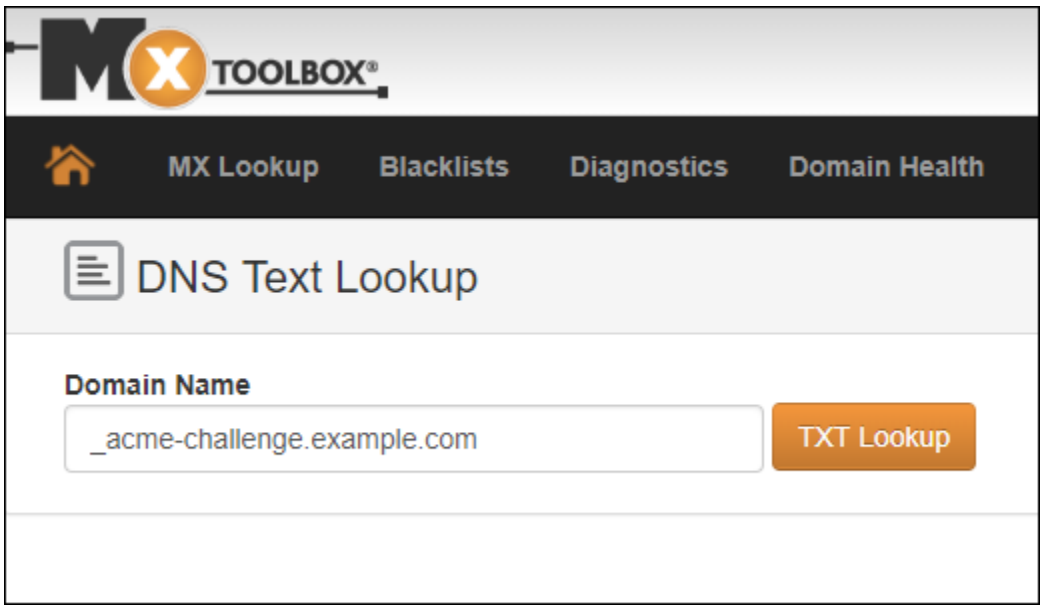
1. 새 브라우저 창을 열고 https://mxtoolbox.com/TXTLookup.aspx로 이동합니다.
2. 다음 텍스트를 텍스트 상자에 입력합니다.

   ```
   _acme-challenge.Domain
   ```

   *Domain*을 등록된 도메인 이름으로 바꿉니다.

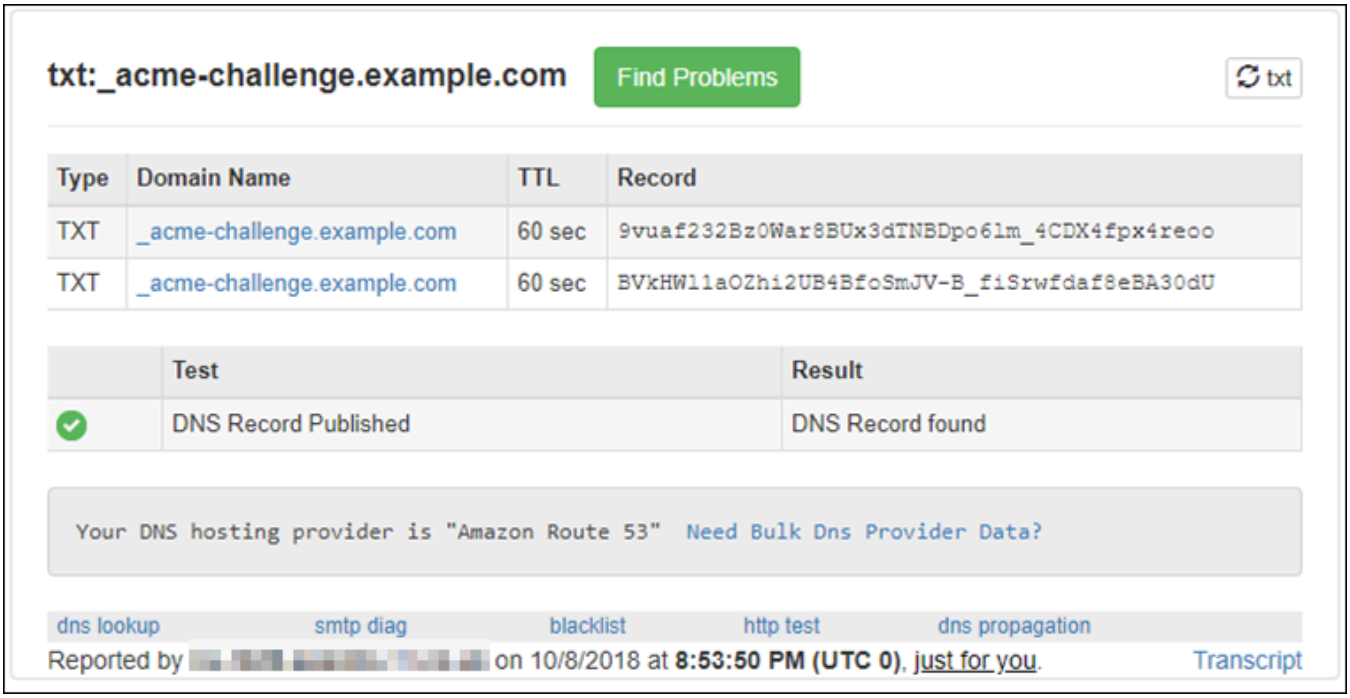
   예제:

   ```
   _acme-challenge.example.com
   ```



3. TXT 조회를 선택하여 점검을 실행합니다.
4. 다음 응답 중 하나가 발생합니다.

   - TXT 레코드가 인터넷 DNS에 전파되면 다음 스크린샷과 비슷한 응답이 표시됩니다. 브라우저 창을 닫고 이 자습서의 다음 단원으로 계속 진행합니다.



   - TXT 레코드가 인터넷의 DNS에 전파되지 않은 경우 DNS Record not found(DNS 레코드를 찾을 수 없음) 응답이 표시됩니다. 도메인의 DNS 영역에 올바른 DNS 레코드를 추가했는지 확인합니

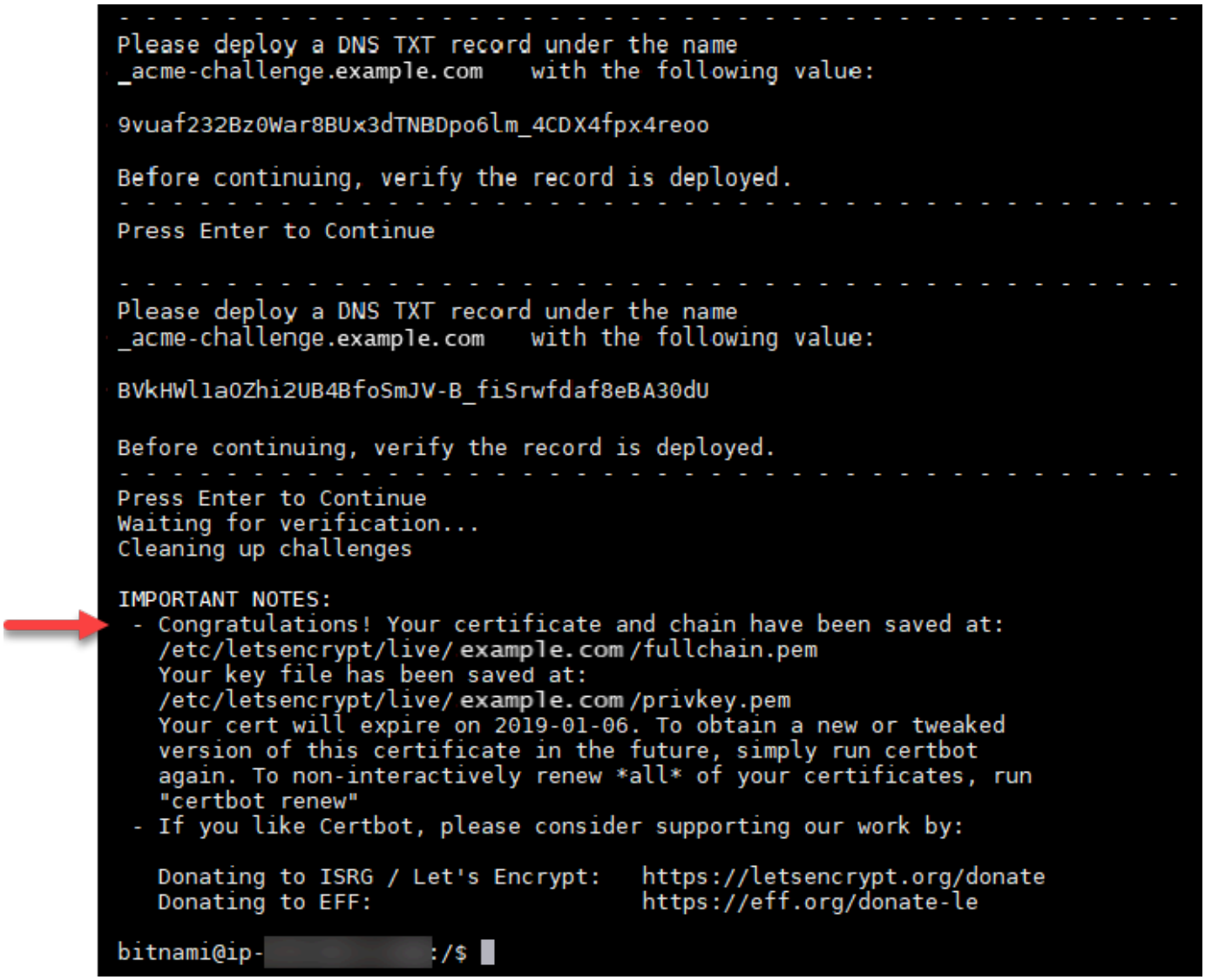
다. 올바른 레코드를 추가한 경우 도메인의 DNS 레코드가 전파되도록 잠시 기다렸다가 TXT 조회를 다시 실행합니다.

6단계: Let's Encrypt SSL 인증서 요청 완료

LAMP 인스턴스의 Lightsail 브라우저 기반 SSH 세션으로 돌아가서 Let's Encrypt 인증서 요청을 완료합니다. Certbot은 SSL 인증서, 체인 및 키 파일을 LAMP 인스턴스의 특정 디렉터리에 저장합니다.

Let's Encrypt SSL 인증서 요청을 완료하려면

1. LAMP 인스턴스의 Lightsail 브라우저 기반 SSH 세션에서 Enter 키를 눌러 Let's Encrypt SSL 인증서 요청을 계속 진행합니다. 성공하면 다음 스크린샷과 비슷한 응답이 나타납니다.



이 메시지는 인증서, 체인 및 키 파일이 /etc/letsencrypt/live/*Domain*/ 디렉터리에 저장되어 있음을 확인합니다. *Domain*은 등록된 도메인 이름(예: /etc/letsencrypt/live/*example.com*/)입니다.

2. 메시지에 지정된 만료 날짜를 적어 둡니다. 이 날짜를 사용하여 해당 날짜까지 인증서를 갱신합니다.

```
IMPORTANT NOTES:
 - Congratulations! Your certificate and chain have been saved at:
   /etc/letsencrypt/live/example.com/fullchain.pem
   Your key file has been saved at:
   /etc/letsencrypt/live/example.com/privkey.pem
   Your cert will expire on 2019-01-06. To obtain a new or tweaked
   version of this certificate in the future, simply run certbot
   again. To non-interactively renew *all* of your certificates, run
   "certbot renew"
 - If you like Certbot, please consider supporting our work by:

   Donating to ISRG / Let's Encrypt:   https://letsencrypt.org/donate
   Donating to EFF:                    https://eff.org/donate-le
```

3. 이제 Let's Encrypt SSL 인증서가 있으므로 이 자습서의 다음 단원으로 계속 진행합니다.

7단계 Let's Encrypt 인증서 파일을 Apache 서버 디렉터리에 연결하는 링크 생성

LAMP 인스턴스의 Apache 서버 디렉터리에 Let's Encrypt SSL 인증서 파일에 대한 링크를 만듭니다. 또한 필요할 때를 대비하여 기존 인증서를 백업합니다.

Let's Encrypt 인증서 파일을 Apache 서버 디렉터리에 연결하는 링크를 생성하려면

1. LAMP 인스턴스에 대한 Lightsail 브라우저 기반 SSH 세션에서 다음 명령을 입력하여 기본 LAMP 스택 서비스를 중지합니다.

```
sudo /opt/bitnami/ctlscript.sh stop
```

다음과 유사한 응답이 나타납니다.

```
bitnami@ip-[redacted]:~$ sudo /opt/bitnami/ctlscript.sh stop

Syntax OK
/opt/bitnami/apache2/scripts/ctl.sh : httpd stopped
/opt/bitnami/php/scripts/ctl.sh : php-fpm stopped
/opt/bitnami/mysql/scripts/ctl.sh : mysql stopped
bitnami@ip-[redacted]:~$
```

2. 다음 명령을 입력하여 도메인에 대한 환경 변수를 설정합니다.

```
DOMAIN=Domain
```

명령에서 *Domain*을 등록된 도메인 이름으로 바꿉니다.

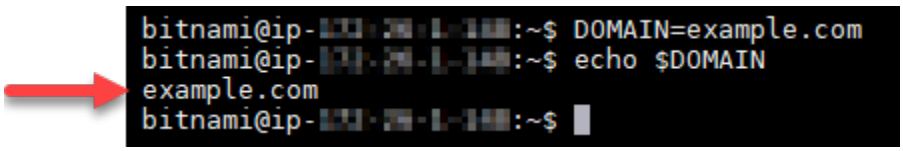
예제:

```
DOMAIN=example.com
```

3. 다음 명령을 입력하여 변수가 올바른 값을 반환하는지 확인합니다.

```
echo $DOMAIN
```

다음과 유사한 결과가 출력되어야 합니다.



4. 다음 명령을 개별적으로 입력하여 백업으로 기존 인증서 파일 이름을 다시 지정합니다. 다양한 배포 및 파일 구조에 대한 정보는 이 자습서의 시작 부분에 있는 중요 블록을 참조하세요.

   - Debian Linux 배포의 경우

     접근 방식 A(시스템 패키지를 사용한 Bitnami 설치):

     ```
     sudo mv /opt/bitnami/apache2/conf/bitnami/certs/server.crt /opt/bitnami/apache2/
     conf/bitnami/certs/server.crt.old
     ```

     ```
     sudo mv /opt/bitnami/apache2/conf/bitnami/certs/server.key /opt/bitnami/apache2/
     conf/bitnami/certs/server.key.old
     ```

     접근 방식 B(자체 포함 Bitnami 설치):

     ```
     sudo mv /opt/bitnami/apache2/conf/server.crt /opt/bitnami/apache2/conf/
     server.crt.old
     ```

     ```
     sudo mv /opt/bitnami/apache2/conf/server.key /opt/bitnami/apache2/conf/
     server.key.old
     ```

- Ubuntu Linux 배포를 사용하는 이전 인스턴스의 경우:

```
sudo mv /opt/bitnami/apache/conf/bitnami/certs/server.crt /opt/bitnami/apache/
conf/bitnami/certs/server.crt.old
```

```
sudo mv /opt/bitnami/apache/conf/bitnami/certs/server.key /opt/bitnami/apache/
conf/bitnami/certs/server.key.old
```

5. 다음 명령을 개별적으로 입력하여 apache2 server 디렉터리에 Let's Encrypt 인증서 파일에 대한 링크를 생성합니다. 다양한 배포 및 파일 구조에 대한 정보는 이 자습서의 시작 부분에 있는 중요 블록을 참조하세요.

   - Debian Linux 배포의 경우

     접근 방식 A(시스템 패키지를 사용한 Bitnami 설치):

```
sudo ln -sf /etc/letsencrypt/live/$DOMAIN/privkey.pem /opt/bitnami/apache2/conf/
bitnami/certs/server.key
```

```
sudo ln -sf /etc/letsencrypt/live/$DOMAIN/fullchain.pem /opt/bitnami/apache2/
conf/bitnami/certs/server.crt
```

     접근 방식 B(자체 포함 Bitnami 설치):

```
sudo ln -sf /etc/letsencrypt/live/$DOMAIN/privkey.pem /opt/bitnami/apache2/conf/
server.key
```

```
sudo ln -sf /etc/letsencrypt/live/$DOMAIN/fullchain.pem /opt/bitnami/apache2/
conf/server.crt
```

   - Ubuntu Linux 배포를 사용하는 이전 인스턴스의 경우:

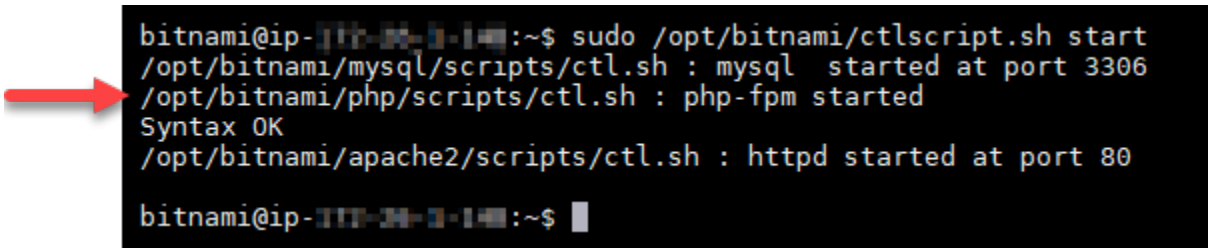
```
sudo ln -s /etc/letsencrypt/live/$DOMAIN/privkey.pem /opt/bitnami/apache/conf/
bitnami/certs/server.key
```

```
sudo ln -s /etc/letsencrypt/live/$DOMAIN/fullchain.pem /opt/bitnami/apache/conf/
bitnami/certs/server.crt
```

6. 다음 명령을 입력하여 이전에 중지한 기본 LAMP 스택 서비스를 시작합니다.

```
sudo /opt/bitnami/ctlscript.sh start
```

다음과 유사한 결과가 출력되어야 합니다.



이제 LAMP 인스턴스가 SSL 암호화를 사용하도록 구성되었습니다. 그러나 트래픽은 HTTP에서 HTTPS로 자동 리디렉션되지 않습니다.

7. 이 자습서의 다음 섹션으로 계속 진행합니다.

8단계: 웹 애플리케이션에 대해 HTTP에서 HTTPS로의 리디렉션 구성

LAMP 인스턴스에 대해 HTTP에서 HTTPS로 리디렉션을 구성할 수 있습니다. HTTP에서 HTTPS로 자동 리디렉션하면 HTTP를 사용하여 연결하는 경우에도 SSL을 사용하는 고객만 사이트에 액세스할 수 있습니다.

웹 애플리케이션에 대해 HTTP에서 HTTPS로의 리디렉션을 구성하려면

1. LAMP 인스턴스에 대한 Lightsail 브라우저 기반 SSH 세션에서 다음 명령을 입력하여 Vim 텍스트 편집기를 통해 Apache 웹 서버 구성 파일을 편집합니다.

```
sudo vim /opt/bitnami/apache2/conf/bitnami/bitnami.conf
```

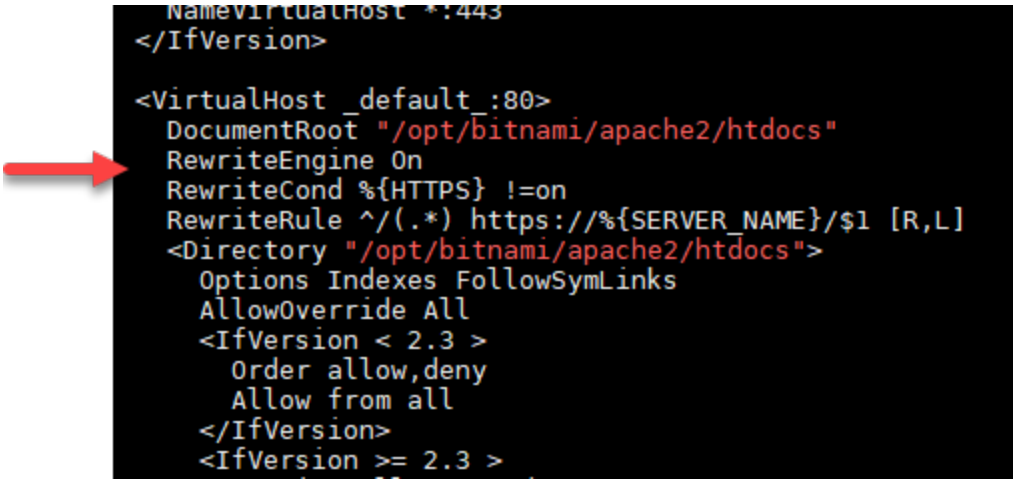
> **Note**
>
> 이 자습서에서는 Vim을 데모용으로 사용하지만 원하는 아무 텍스트 편집기나 이 단계에 사용할 수 있습니다.

2. Vim 편집기에서 `i`를 눌러 삽입 모드를 시작합니다.
3. 파일에서 `DocumentRoot "/opt/bitnami/apache2/htdocs"` 및 `<Directory "/opt/bitnami/apache2/htdocs">` 사이에 다음 텍스트를 입력합니다.

```
RewriteEngine On
```

```
RewriteCond %{HTTPS} !=on
RewriteRule ^/(.*) https://%{SERVER_NAME}/$1 [R,L]
```

결과는 다음과 같아야 합니다.



4. ESC 키를 누른 다음 `:wq`를 입력하여 편집 내용을 작성(저장)하고 Vim을 종료합니다.
5. 다음 명령을 입력하여 기본 LAMP 스택 서비스를 다시 시작하고 편집 사항을 적용합니다.

```
sudo /opt/bitnami/ctlscript.sh restart
```

이제 LAMP 인스턴스가 HTTP에서 HTTPS로의 연결을 자동으로 리디렉션하도록 구성되었습니다. 방문자가 `http://www.example.com`으로 이동하면 자동으로 암호화된 `https://www.example.com` 주소로 리디렉션됩니다.

9단계: 90일마다 Let's Encrypt 인증서 갱신

Let's Encrypt 인증서는 90일 동안 유효합니다. 인증서는 만료되기 30일 전에 갱신할 수 있습니다. Let's Encrypt 인증서를 갱신하려면 인증서를 가져오는 데 사용한 원래 명령을 실행합니다. 이 자습서의 Let's Encrypt SSL 와일드카드 인증서 요청 단원에 나오는 단계를 반복합니다.

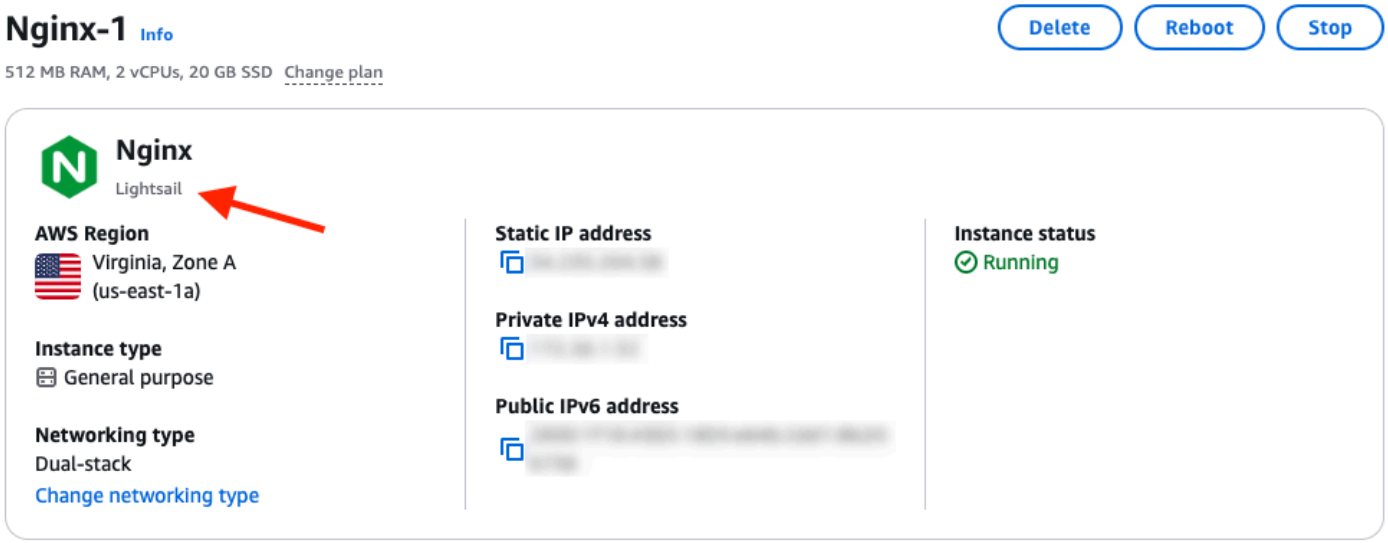
Let's Encrypt SSL/TLS를 사용하여 Lightsail NGINX 웹 사이트 보호

Amazon Lightsail을 사용하면 Lightsail 로드 밸런서를 사용하여 SSL/TLS를 사용하는 웹사이트 및 애플리케이션을 쉽게 보호할 수 있습니다. 그러나 Lightsail 로드 밸런서를 사용하는 것이 일반적으로 올바른 선택이 아닐 수도 있습니다. 사이트에서 로드 밸런서가 제공하는 확장성 또는 내결함성을 필요로 하지 않거나 비용 최적화를 원할 수도 있습니다.

후자의 경우에는 Let's Encrypt를 사용하여 무료 SSL 인증서를 얻을 수 있으며 그런 경우 아무런 문제가 되지 않습니다. 이러한 인증서를 Lightsail 인스턴스와 통합할 수 있습니다. 이 자습서에서는 Certbot을 사용하여 Let's Encrypt 와일드카드 인증서를 요청하고 이를 Nginx 인스턴스와 통합하는 방법을 보여줍니다.

Nginx 블루프린트 공급업체 식별

다음은에서 Nginx 인스턴스를 설치하고 실행한 후 시작해야 하는 몇 가지 단계입니다Amazon Lightsail. 시작하기 전에 인스턴스 관리 페이지에서 블루프린트 공급업체를 식별합니다.



Nginx 인스턴스에 적합한 가이드를 선택합니다.

Bitnami

> **Important**
>
> - Bitnami 인스턴스에서 사용하는 Linux 배포는 2020년 7월에 Ubuntu에서 Debian으로 변경되었습니다. 이러한 변화로 인스턴스의 Linux 배포에 따라 자습서의 일부 단계가 달라집니다. 변경 후 생성된 모든 Bitnami 블루프린트 인스턴스는 Debian Linux 배포를 사용합니다. 변경 전에 생성된 인스턴스는 Ubuntu Linux 배포를 계속 사용합니다. 인스턴스의 배포를 확인하려면 `uname -a` 명령을 실행합니다. 응답은 Ubuntu 또는 Debian을 인스턴스의 Linux 배포로 표시합니다.
> - Bitnami는 많은 스택에 대한 파일 구조를 수정하는 중입니다. 이 자습서의 파일 경로는 Bitnami 스택이 네이티브 Linux 시스템 패키지(접근법 A)를 사용하는지 또는 자체 포함

설치(접근법 B)인지 여부에 따라 달라질 수 있습니다. Bitnami 설치 유형 및 따라야 할 접근 방식을 식별하려면 다음 명령을 실행합니다.

```
test ! -f "/opt/bitnami/common/bin/openssl" && echo "Approach
A: Using system packages." || echo "Approach B: Self-contained
installation."
```

목차

- 1단계: 필수 구성 요소 완성
- 2단계: Lightsail 인스턴스에 Certbot 설치
- 3단계: Let's Encrypt SSL 와일드카드 인증서 요청
- 4단계: 도메인의 DNS 영역에 TXT 레코드 추가
- 5단계: TXT 레코드가 전파되었는지 확인
- 6단계: Let's Encrypt SSL 인증서 요청 완료
- 7단계: NGINX 서버 디렉터리에서 Let's Encrypt 인증서 파일에 대한 링크 생성
- 8단계: 웹 애플리케이션에 대해 HTTP에서 HTTPS로의 리디렉션 구성
- 9단계: 90일마다 Let's Encrypt 인증서 갱신

1단계: 필수 구성 요소 완성

아직 수행하지 않은 경우 다음 사전 조건을 완료하세요.

- Lightsail에서 Nginx 인스턴스를 생성합니다. 자세한 내용은 인스턴스 생성을 참조하세요.
- DNS 레코드를 편집하려면 도메인 이름을 등록하고 관리 액세스 권한을 얻습니다. 자세한 내용은 DNS를 참조하세요.

> **Note**
>
> Lightsail DNS 영역을 사용하여 도메인의 DNS 레코드를 관리하는 것이 좋습니다. 자세한 내용은 DNS 영역을 생성하여 도메인의 DNS 레코드 관리를 참조하세요.

- Lightsail 콘솔의 브라우저 기반 SSH 터미널을 사용하여 이 자습서의 단계를 수행합니다. 그러나 PuTTY와 같은 자체 SSH 클라이언트를 사용할 수도 있습니다. PuTTY 구성에 대한 자세한 내용은 Amazon Lightsail에서 SSH를 사용하여 연결할 PuTTY 다운로드 및 설정을 참조하세요.

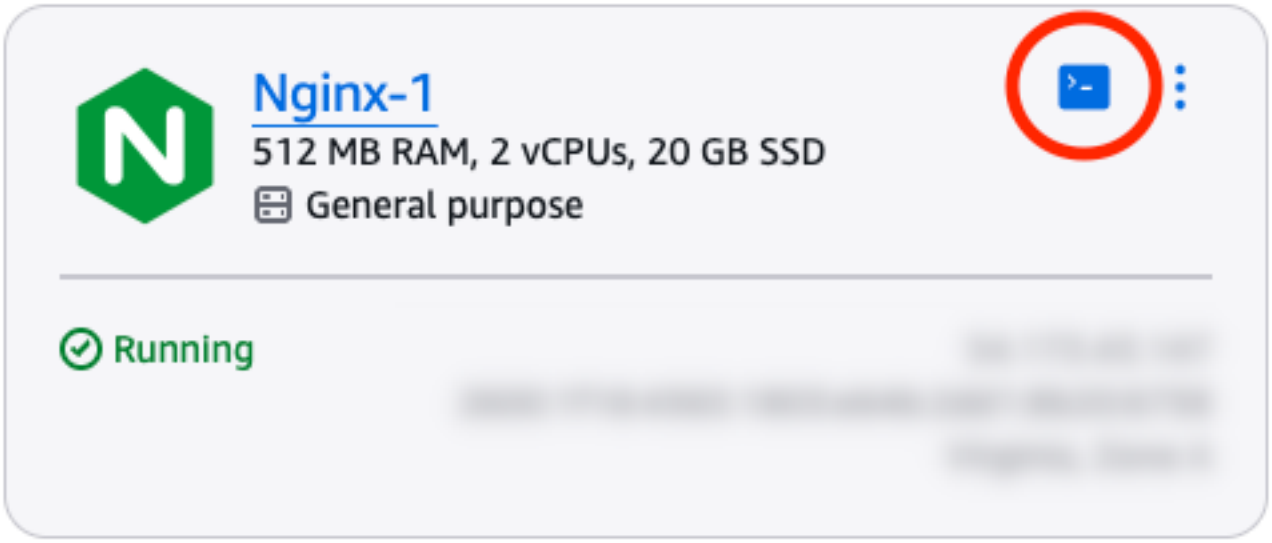
사전 조건을 완료한 후 이 자습서의 다음 단원으로 계속 진행합니다.

2단계: Lightsail 인스턴스에 Certbot 설치

Certbot은 Let's Encrypt에서 인증서를 요청하고 이를 웹 서버에 배포하는 데 사용되는 클라이언트입니다. Let's Encrypt는 ACME 프로토콜을 사용하여 인증서를 발행하고 Certbot은 Let's Encrypt와 상호 작용하는 ACME 사용 클라이언트입니다.

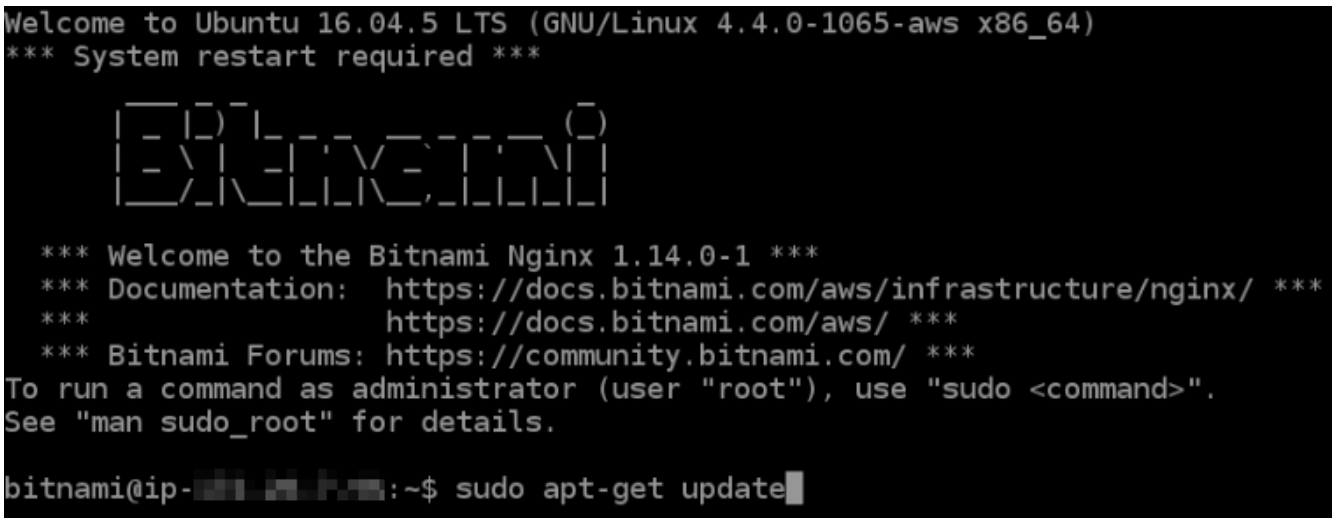
Lightsail 인스턴스에 Certbot을 설치하려면

1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 연결할 인스턴스에 대한 SSH 빠른 연결 아이콘을 선택합니다.



3. Lightsail 브라우저 기반 SSH 세션이 연결되면 다음 명령을 입력하여 인스턴스의 패키지를 업데이트합니다.

```
sudo apt-get update
```



4. 다음 명령을 입력하여 소프트웨어 속성 패키지를 설치합니다. Certbot의 개발자는 Personal Package Archive(PPA)를 사용하여 Certbot을 배포합니다. 소프트웨어 속성 패키지를 사용하면 PPA를 사용하여 보다 효율적으로 작업할 수 있습니다.

```
sudo apt-get install software-properties-common
```

> **Note**
>
> `sudo apt-get install` 명령을 실행할 때 `Could not get lock` 오류가 발생하면 약 15분 정도 기다린 후 다시 시도합니다. 이 오류는 Apt 패키지 관리 도구를 사용하여 무인 업그레이드를 설치하는 cron 작업으로 인해 발생할 수 있습니다.

5. 다음 명령을 입력하여 로컬 apt 리포지토리에 Certbot을 추가합니다.

> **Note**
>
> 5단계는 Ubuntu Linux 배포를 사용하는 인스턴스에만 적용됩니다. 인스턴스가 Debian Linux 배포를 사용하는 경우 이 단계를 건너뜁니다.

```
sudo apt-add-repository ppa:certbot/certbot -y
```

6. 다음 명령을 입력하여 새 리포지토리를 포함하도록 apt를 업데이트합니다.

```
sudo apt-get update -y
```

7. 다음 명령을 입력하여 Certbot을 설치합니다.

```
sudo apt-get install certbot -y
```

이제 Certbot이 Lightsail 인스턴스에 설치됩니다.

8. 브라우저 기반 SSH 터미널 창을 열린 상태로 유지하세요. 이 자습서의 뒷부분에서 다시 해당 세션으로 돌아옵니다. 이 자습서의 다음 섹션으로 계속 진행합니다.

3단계: Let's Encrypt SSL 와일드카드 인증서 요청

Let's Encrypt에서 인증서를 요청하는 프로세스를 시작합니다. Certbot을 사용하여 도메인 및 하위 도메인에 대해 단일 인증서를 사용할 수 있는 와일드카드 인증서를 요청합니다. 예를 들어 `example.com` 최상위 도메인과 `blog.example.com` 및 `stuff.example.com` 하위 도메인에 대해 단일 와일드카드 인증서가 작동합니다.

Let's Encrypt SSL 와일드카드 인증서를 요청하려면

1. 이 자습서의 2단계에서 사용된 것과 동일한 브라우저 기반 SSH 터미널 창에서 다음 명령을 입력하여 도메인에 대해 환경 변수를 설정합니다. 이제 명령을 더 효율적으로 복사하고 붙여 넣어 인증서를 얻을 수 있습니다. *domain*을 등록된 도메인 이름으로 바꿔야 합니다.

```
DOMAIN=domain
```

```
WILDCARD=*.$DOMAIN
```

예제:

```
DOMAIN=example.com
```

```
WILDCARD=*.$DOMAIN
```

2. 다음 명령을 입력하여 변수가 올바른 값을 반환하는지 확인합니다.

```
echo $DOMAIN && echo $WILDCARD
```

다음과 유사한 결과가 출력되어야 합니다.

3. 다음 명령을 입력하여 대화형 모드에서 Certbot을 시작합니다. 이 명령은 DNS 문제에 대해 수동 권한 부여 방법을 사용하여 도메인 소유권을 확인하도록 Certbot에 지시하며, 최상위 도메인 및 하위 도메인에 대한 와일드카드 인증서를 요청합니다.

```
sudo certbot -d $DOMAIN -d $WILDCARD --manual --preferred-challenges dns
 certonly
```

4. 메시지가 표시되면 갱신 및 보안 고지 사항에 사용될 이메일 주소를 입력합니다.
5. Let's Encrypt 서비스 계약 조건을 읽습니다. 모두 읽은 다음 동의하면 A를 누릅니다. 동의하지 않는 경우 Let's Encrypt 인증서를 얻을 수 없습니다.
6. 이메일 주소를 공유하라는 메시지와 IP 주소가 기록된다는 경고에 적절하게 응답합니다.
7. 이제 Let's Encrypt에 지정된 도메인을 소유하고 있는지 확인하라는 메시지가 표시됩니다. TXT 레코드를 도메인의 DNS 레코드에 추가하여 이 작업을 수행할 수 있습니다. 다음 예와 같이 한 세트의 TXT 레코드 값이 제공됩니다.

> **Note**
>
> 확인에 사용해야 하는 하나 또는 여러 개의 TXT 레코드를 Let's Encrypt에서 제공할 수 있습니다. 이 예에서는 확인에 사용할 두 개의 TXT 레코드가 제공되었습니다.

8. Lightsail 브라우저 기반 SSH 세션을 열린 상태로 유지하세요. 이 자습서의 뒷부분에서 다시 해당 세션으로 돌아옵니다. 이 자습서의 다음 섹션으로 계속 진행합니다.

4단계: 도메인의 DNS 영역에 TXT 레코드 추가

도메인의 DNS 영역에 TXT 레코드를 추가하면 도메인 소유권이 확인됩니다. 설명을 위해 Lightsail DNS 영역을 사용하겠습니다. 하지만 이 단계는 일반적으로 도메인 등록 대행자가 호스팅하는 다른 DNS 영역과 비슷할 수 있습니다.

> **Note**
>
> 도메인에 대해 Lightsail DNS 영역을 만드는 방법에 대한 자세한 내용은 DNS 영역을 생성하여 Lightsail에서 도메인의 DNS 레코드 관리를 참조하세요.

Lightsail의 도메인 DNS 영역에 TXT 레코드를 추가하려면

1. 왼쪽 탐색 창에서 도메인 및 DNS를 선택합니다.
2. 페이지의 DNS 영역 섹션에서 Certbot 인증서 요청에서 지정한 도메인의 DNS 영역을 선택합니다.
3. DNS 영역 편집기에서 DNS records(DNS 레코드)를 선택합니다.
4. 레코드 추가(Add record)를 선택합니다.
5. 레코드 유형(Record type) 드롭다운 메뉴에서 TXT 레코드를 선택합니다.
6. Let's Encrypt 인증서 요청에서 지정한 값을 레코드 이름(Record name) 및 응답(Responds with) 필드에 입력합니다.

   > **Note**
   >
   > Lightsail 콘솔은 도메인의 정점 부분을 미리 채웁니다. 예를 들어, *_acme-challenge.example.com* 하위 도메인을 추가하려면 텍스트 상자에 *_acme-challenge*만 입력하면 되며, 레코드를 저장할 때 Lightsail이 `.example.com` 부분을 추가합니다.

7. 저장을 선택합니다.
8. 4~7단계를 반복하여 Let's Encrypt 인증서 요청에 지정된 두 번째 TXT 레코드 세트를 추가합니다.
9. Lightsail 콘솔 브라우저 창을 열린 상태로 유지하세요. 이 자습서의 뒷부분에서 다시 해당 세션으로 돌아옵니다. 이 자습서의 다음 섹션으로 계속 진행합니다.

5단계: TXT 레코드가 전파되었는지 확인

TXT 레코드가 인터넷의 DNS에 전파되었는지 확인하려면 MxToolbox 유틸리티를 사용합니다. DNS 레코드 전파는 DNS 호스팅 공급자 및 DNS 레코드에 대해 구성된 TTL(Time to Live)에 따라 다소 시간이 걸릴 수 있습니다. 이 단계를 완료하고 Certbot 인증서 요청을 계속하기 전에 TXT 레코드가 전파되었는지 확인하는 것이 중요합니다. 그렇지 않으면 인증서 요청이 실패합니다.

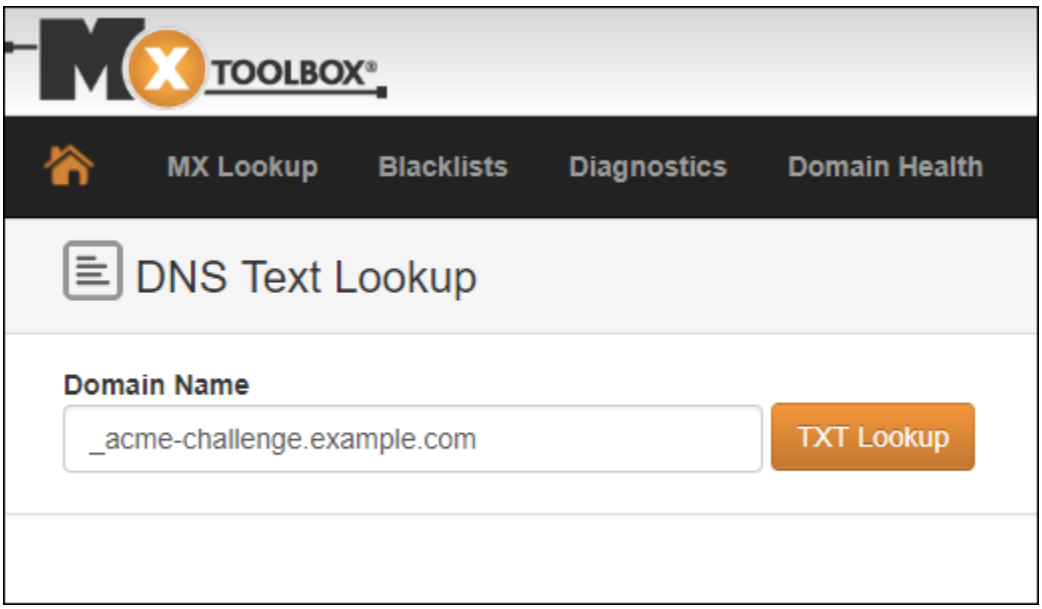
TXT 레코드가 인터넷의 DNS에 전파되었는지 확인하려면

1. 새 브라우저 창을 열고 https://mxtoolbox.com/TXTLookup.aspx로 이동합니다.
2. 다음 텍스트를 텍스트 상자에 입력합니다. *domain*을 도메인으로 바꿉니다.

```
_acme-challenge.domain
```

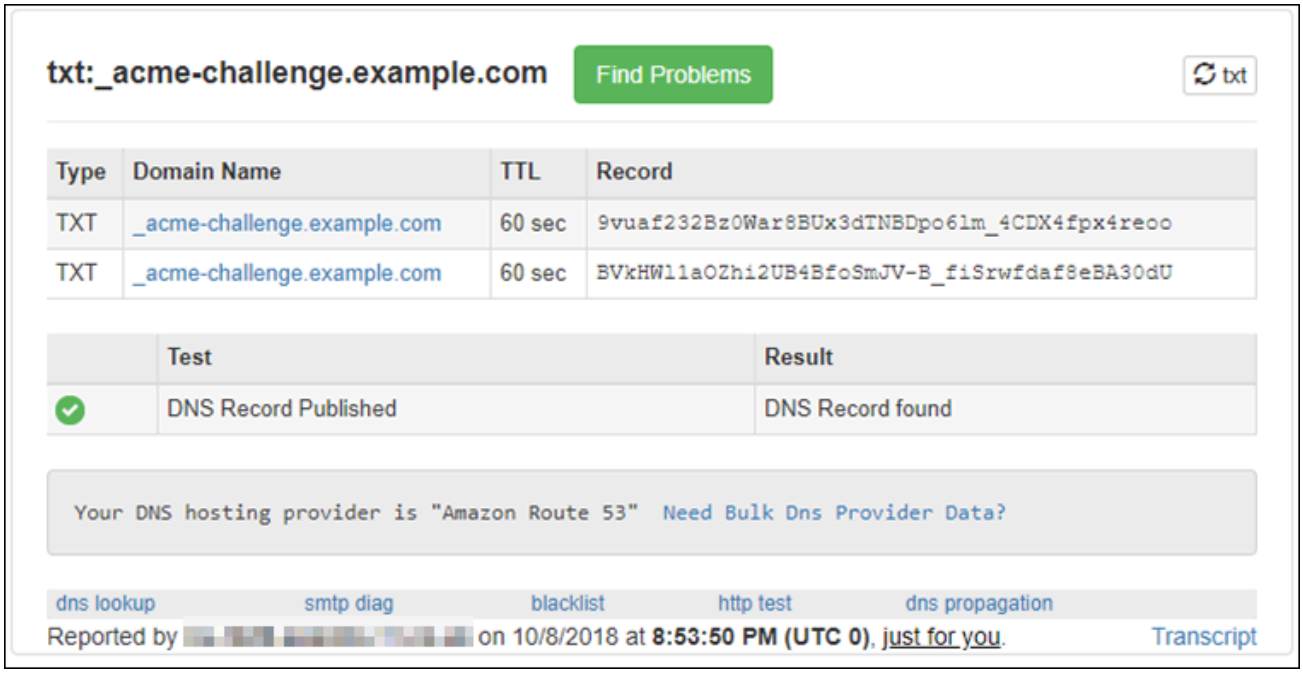
예제:

```
_acme-challenge.example.com
```



3. TXT 조회를 선택하여 점검을 실행합니다.
4. 다음 응답 중 하나가 발생합니다.

   - TXT 레코드가 인터넷의 DNS에 전파되면 다음 스크린샷과 비슷한 응답이 표시됩니다. 브라우저 창을 닫고 이 자습서의 다음 단원으로 계속 진행합니다.



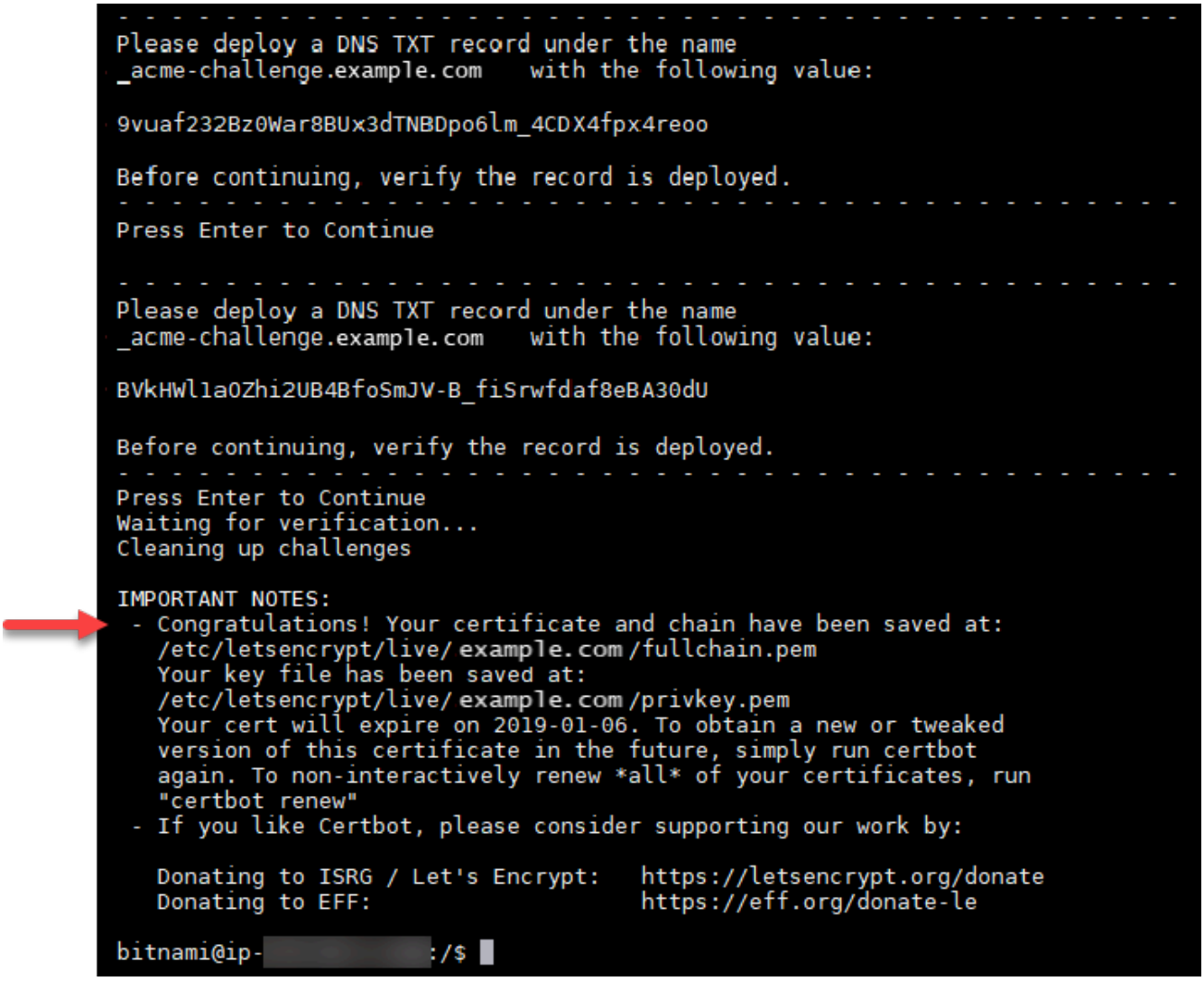
- TXT 레코드가 인터넷의 DNS에 전파되지 않은 경우 DNS Record not found(DNS 레코드를 찾을 수 없음) 응답이 표시됩니다. 도메인의 DNS 영역에 올바른 DNS 레코드를 추가했는지 확인합니다. 올바른 레코드를 추가한 경우 도메인의 DNS 레코드가 전파되도록 잠시 기다렸다가 TXT 조회를 다시 실행합니다.

6단계: Let's Encrypt SSL 인증서 요청 완료

Nginx 인스턴스의 Lightsail 브라우저 기반 SSH 세션으로 돌아가서 Let's Encrypt 인증서 요청을 완료합니다. Certbot은 SSL 인증서, 체인 및 키 파일을 Nginx 인스턴스의 특정 디렉터리에 저장합니다.

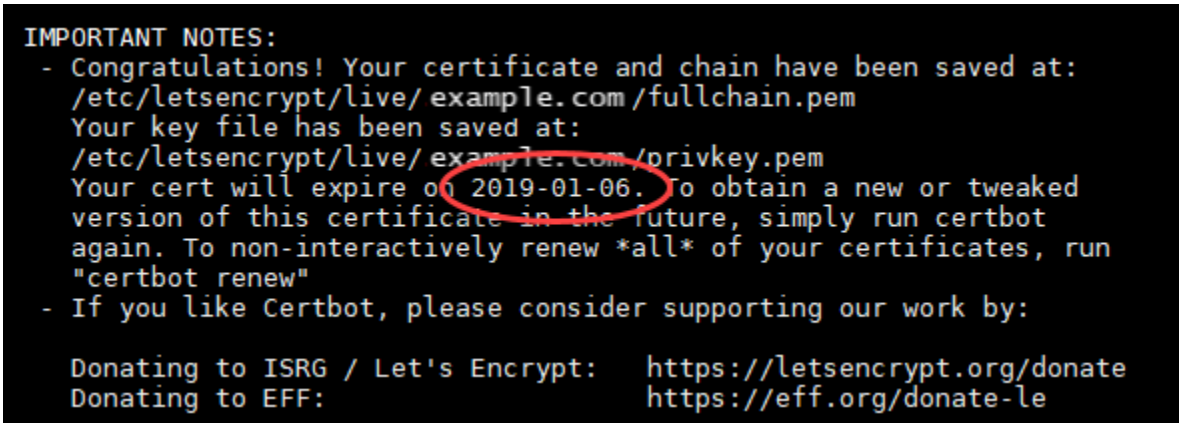
Let's Encrypt SSL 인증서 요청을 완료하려면

1. Nginx 인스턴스의 Lightsail 브라우저 기반 SSH 세션에서 Enter 키를 눌러 Let's Encrypt SSL 인증서 요청을 계속 진행합니다. 성공하면 다음 스크린샷과 비슷한 응답이 나타납니다.



이 메시지는 인증서, 체인 및 키 파일이 /etc/letsencrypt/live/*domain*/ 디렉터리에 저장되어 있음을 확인합니다. *domain*을 /etc/letsencrypt/live/*example.com*/과 같은 도메인으로 바꿔야 합니다.

2. 메시지에 지정된 만료 날짜를 적어 둡니다. 이 날짜를 사용하여 해당 날짜까지 인증서를 갱신합니다.



3. 이제 Let's Encrypt SSL 인증서가 있으므로 이 자습서의 다음 단원으로 계속 진행합니다.

7단계: NGINX 서버 디렉터리에서 Let's Encrypt 인증서 파일에 대한 링크 생성

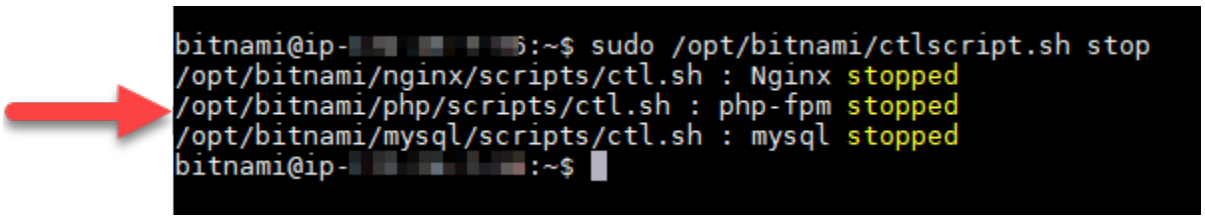
Nginx 인스턴스의 NGINX 서버 디렉터리에서 Let's Encrypt SSL 인증서 파일에 대한 링크를 생성합니다. 또한 필요할 때를 대비하여 기존 인증서를 백업합니다.

NGINX 서버 디렉터리에서 Let's Encrypt 인증서 파일에 대한 링크를 생성하려면

1. Nginx 인스턴스에 대한 Lightsail 브라우저 기반 SSH 세션에서 다음 명령을 입력하여 기본 서비스를 중지합니다.

```
sudo /opt/bitnami/ctlscript.sh stop
```

다음과 유사한 응답이 나타납니다.



2. 다음 명령을 입력하여 도메인에 대한 환경 변수를 설정합니다. 명령을 더 효율적으로 복사하고 붙여 넣어 인증서 파일을 연결할 수 있습니다. *domain*을 등록된 도메인 이름으로 바꿔야 합니다.

```
DOMAIN=domain
```

예제:

```
DOMAIN=example.com
```

3. 다음 명령을 입력하여 변수가 올바른 값을 반환하는지 확인합니다.

```
echo $DOMAIN
```

다음과 유사한 결과가 출력되어야 합니다.

4. 다음 명령을 개별적으로 입력하여 백업으로 기존 인증서 파일 이름을 다시 지정합니다. 다양한 배포 및 파일 구조에 대한 정보는 이 자습서의 시작 부분에 있는 중요 블록을 참조하세요.

   - Debian Linux 배포의 경우

     접근 방식 A(시스템 패키지를 사용한 Bitnami 설치):

```
sudo mv /opt/bitnami/nginx/conf/bitnami/certs/server.crt /opt/bitnami/nginx/
conf/bitnami/certs/server.crt.old
```

```
sudo mv /opt/bitnami/nginx/conf/bitnami/certs/server.key /opt/bitnami/nginx/
conf/bitnami/certs/server.key.old
```

     접근 방식 B(자체 포함 Bitnami 설치):

```
sudo mv /opt/bitnami/nginx/conf/server.crt /opt/bitnami/nginx/conf/
server.crt.old
```

```
sudo mv /opt/bitnami/nginx/conf/server.key /opt/bitnami/nginx/conf/
server.key.old
```

   - Ubuntu Linux 배포를 사용하는 이전 인스턴스의 경우:

```
sudo mv /opt/bitnami/nginx/conf/bitnami/certs/server.crt /opt/bitnami/nginx/
conf/bitnami/certs/server.crt.old
```

```
sudo mv /opt/bitnami/nginx/conf/bitnami/certs/server.key /opt/bitnami/nginx/
conf/bitnami/certs/server.key.old
```

5. 다음 명령을 개별적으로 입력하여 NGINX 서버 디렉터리에서 Let's Encrypt 인증서 파일에 대한 링크를 생성합니다. 다양한 배포 및 파일 구조에 대한 정보는 이 자습서의 시작 부분에 있는 중요 블록을 참조하십시오.

   - Debian Linux 배포의 경우

     접근 방식 A(시스템 패키지를 사용한 Bitnami 설치):

     ```
     sudo ln -sf /etc/letsencrypt/live/$DOMAIN/privkey.pem /opt/bitnami/nginx/conf/
     bitnami/certs/server.key
     ```

     ```
     sudo ln -sf /etc/letsencrypt/live/$DOMAIN/fullchain.pem /opt/bitnami/nginx/
     conf/bitnami/certs/server.crt
     ```

     접근 방식 B(자체 포함 Bitnami 설치):

     ```
     sudo ln -sf /etc/letsencrypt/live/$DOMAIN/privkey.pem /opt/bitnami/nginx/conf/
     server.key
     ```

     ```
     sudo ln -sf /etc/letsencrypt/live/$DOMAIN/fullchain.pem /opt/bitnami/nginx/
     conf/server.crt
     ```

   - Ubuntu Linux 배포를 사용하는 이전 인스턴스의 경우:

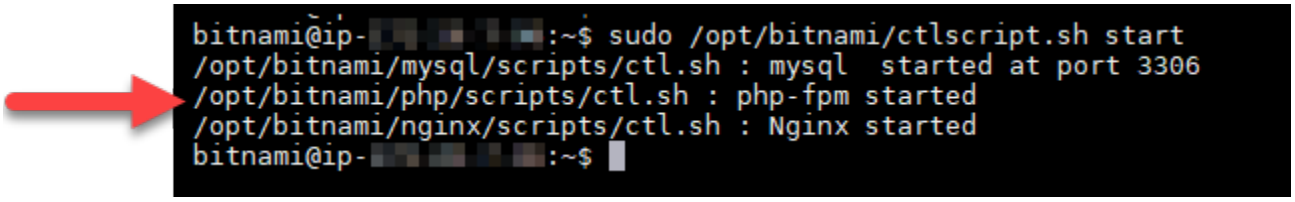
     ```
     sudo ln -s /etc/letsencrypt/live/$DOMAIN/privkey.pem /opt/bitnami/nginx/conf/
     bitnami/certs/server.key
     ```

     ```
     sudo ln -s /etc/letsencrypt/live/$DOMAIN/fullchain.pem /opt/bitnami/nginx/
     conf/bitnami/certs/server.crt
     ```

6. 다음 명령을 입력하여 이전에 중지한 기본 서비스를 시작합니다.

   ```
   sudo /opt/bitnami/ctlscript.sh start
   ```

   다음과 유사한 결과가 출력되어야 합니다.



이제 Nginx 인스턴스가 SSL 암호화를 사용하도록 구성되었습니다. 그러나 트래픽은 HTTP에서 HTTPS로 자동 리디렉션되지 않습니다.

7. 이 자습서의 다음 섹션으로 계속 진행합니다.

8단계: 웹 애플리케이션에 대해 HTTP에서 HTTPS로의 리디렉션 구성

Nginx 인스턴스에 대해 HTTP에서 HTTPS로 리디렉션을 구성할 수 있습니다. HTTP에서 HTTPS로 자동 리디렉션하면 HTTP를 사용하여 연결하는 경우에도 SSL을 사용하는 고객만 사이트에 액세스할 수 있습니다. 다양한 배포 및 파일 구조에 대한 정보는 이 자습서의 시작 부분에 있는 중요 블록을 참조하십시오.

이 자습서에서는 Vim을 데모용으로 사용하지만 원하는 텍스트 편집기를 사용할 수 있습니다.

Debian Linux 배포의 경우 - 웹 애플리케이션에 대해 HTTP에서 HTTPS로의 리디렉션 구성

1. Nginx 인스턴스에 대한 Lightsail 브라우저 기반 SSH 세션에서 다음 명령을 입력하여 서버 블록 구성 파일을 수정합니다. <ApplicationName>을 애플리케이션 이름으로 바꿉니다.

```
sudo vim /opt/bitnami/nginx/conf/server_blocks/<ApplicationName>-server-
block.conf
```

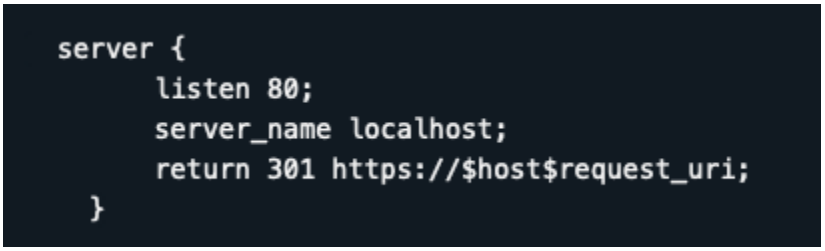
2. Vim 편집기에서 i를 눌러 삽입 모드를 시작합니다.
3. 다음 예제의 정보를 사용하여 파일을 편집합니다.

```
server {
    listen 80 default_server;
    root /opt/bitnami/APPNAME;
    return 301 https://$host$request_uri;
}
```

4. ESC 키를 누른 다음 :wq를 입력하여 편집 내용을 작성(저장)하고 Vim을 종료합니다.
5. 다음 명령을 입력하여 NGINX 구성 파일의 서버 섹션을 수정합니다.

```
sudo vim /opt/bitnami/nginx/conf/nginx.conf
```

6. Vim 편집기에서 i를 눌러 삽입 모드를 시작합니다.
7. 다음 예제의 정보를 사용하여 파일을 편집합니다.



8. ESC 키를 누른 다음 :wq를 입력하여 편집 내용을 작성(저장)하고 Vim을 종료합니다.
9. 다음 명령을 입력하여 기본 서비스를 다시 시작하고 편집 사항을 적용합니다.

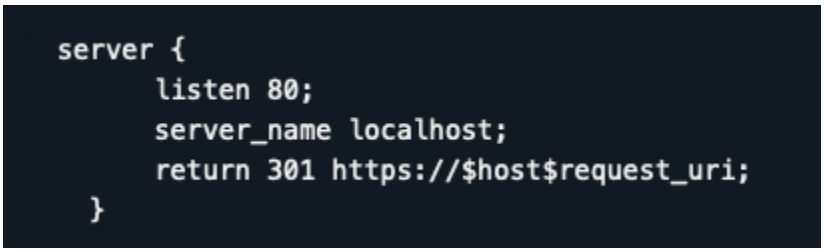
```
sudo /opt/bitnami/ctlscript.sh restart
```

접근 방식 B(자체 포함 Bitnami 설치):

1. Nginx 인스턴스의 Lightsail 브라우저 기반 SSH 세션에서 다음 명령을 입력하여 NGINX 구성 파일의 서버 섹션을 수정합니다.

```
sudo vim /opt/bitnami/nginx/conf/nginx.conf
```

2. Vim 편집기에서 i를 눌러 삽입 모드를 시작합니다.
3. 다음 예제의 정보를 사용하여 파일을 편집합니다.



4. ESC 키를 누른 다음 :wq를 입력하여 편집 내용을 작성(저장)하고 Vim을 종료합니다.
5. 다음 명령을 입력하여 기본 서비스를 다시 시작하고 편집 사항을 적용합니다.

```
sudo /opt/bitnami/ctlscript.sh restart
```

Ubuntu Linux 배포판을 사용하는 이전 인스턴스의 경우 - 웹 애플리케이션에 대해 HTTP에서 HTTPS로의 리디렉션 구성

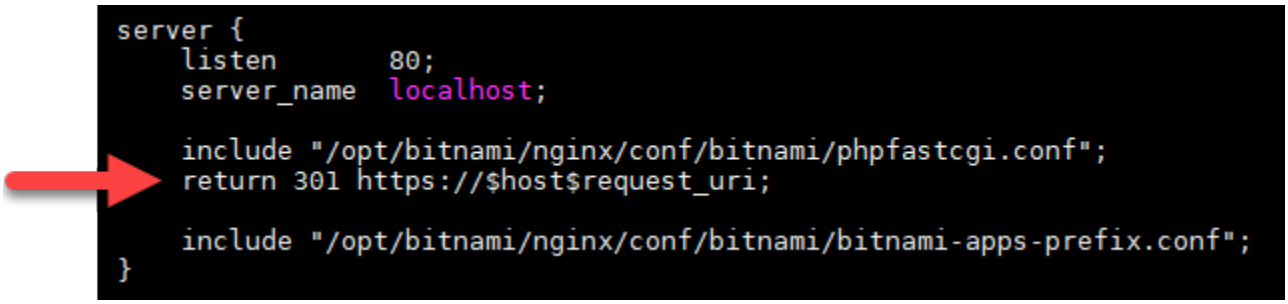
1. Nginx 인스턴스의 Lightsail 브라우저 기반 SSH 세션에서 다음 명령을 입력하여 Vim 텍스트 편집기를 사용하여 NGINX 웹 서버 구성 파일을 편집합니다.

```
sudo vim /opt/bitnami/nginx/conf/bitnami/bitnami.conf
```

2. Vim 편집기에서 `i`를 눌러 삽입 모드를 시작합니다.
3. 파일에서 `server_name localhost;` 및 `include "/opt/bitnami/nginx/conf/bitnami/bitnami-apps-prefix.conf";` 사이에 다음 텍스트를 입력합니다.

```
return 301 https://$host$request_uri;
```

결과는 다음과 같아야 합니다.



4. ESC 키를 누른 다음 `:wq`를 입력하여 편집 내용을 작성(저장)하고 Vim을 종료합니다.
5. 다음 명령을 입력하여 기본 서비스를 다시 시작하고 편집 사항을 적용합니다.

```
sudo /opt/bitnami/ctlscript.sh restart
```

이제 Nginx 인스턴스가 HTTP에서 HTTPS로의 연결을 자동으로 리디렉션하도록 구성되었습니다. 방문자가 `http://www.example.com`으로 이동하면 자동으로 암호화된 `https://www.example.com` 주소로 리디렉션됩니다.

9단계: 90일마다 Let's Encrypt 인증서 갱신

Let's Encrypt 인증서는 90일 동안 유효합니다. 인증서는 만료되기 30일 전에 갱신할 수 있습니다. Let's Encrypt 인증서를 갱신하려면 인증서를 가져오는 데 사용한 원래 명령을 실행합니다. 이 자습서의 Let's Encrypt SSL 와일드카드 인증서 요청 단원에 나오는 단계를 반복합니다.

Lightsail

목차

- 1단계: 사전 조건 완료
- 2단계: Lightsail 인스턴스에 Certbot 설치
- 3단계: Let's Encrypt SSL 와일드카드 인증서 요청
- 4단계: 도메인의 DNS 영역에 TXT 레코드 추가
- 5단계: TXT 레코드가 전파되었는지 확인
- 6단계: Let's Encrypt SSL 인증서 요청 완료
- 7단계: NGINX에서 SSL 구성 업데이트 및 HTTP에서 HTTPS로 트래픽 리디렉션
- 8단계: 90일마다 Let's Encrypt 인증서 갱신

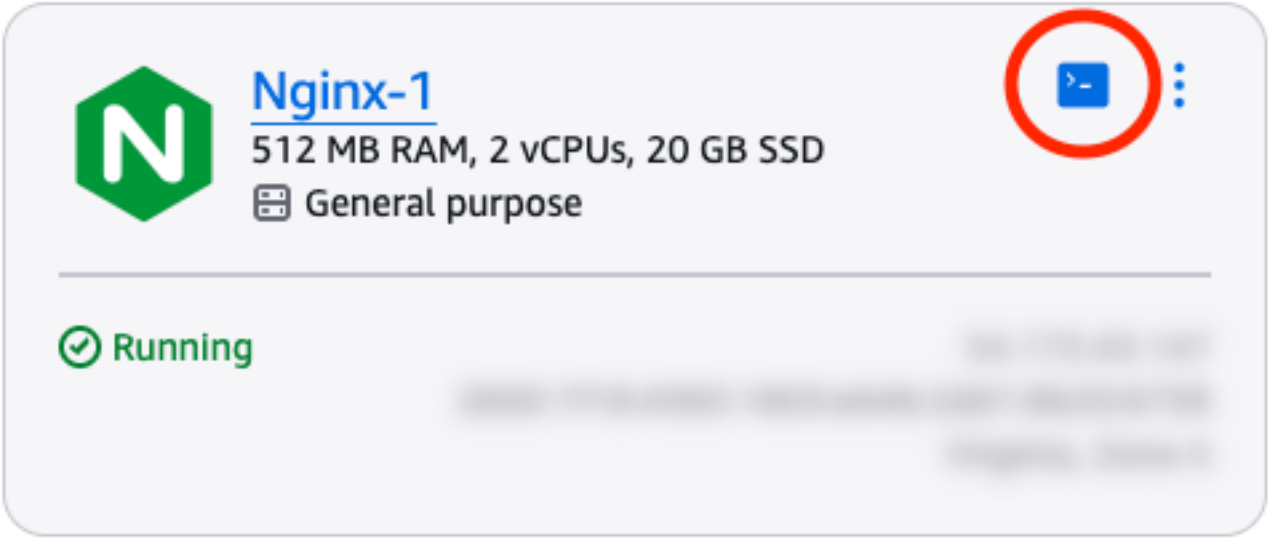
1단계: 필수 구성 요소 완성

아직 수행하지 않은 경우 다음 사전 조건을 완료하세요.

- Lightsail에서 Nginx 인스턴스를 생성합니다. 자세한 내용은 인스턴스 생성을 참조하세요.
- DNS 레코드를 편집하려면 도메인 이름을 등록하고 관리 액세스 권한을 얻습니다. 자세한 내용은 DNS를 참조하세요.

> **Note**
>
> Lightsail DNS 영역을 사용하여 도메인의 DNS 레코드를 관리하는 것이 좋습니다. 자세한 내용은 도메인의 DNS 레코드를 관리하기 위한 DNS 영역 생성을 참조하세요.

- Lightsail 콘솔에서 브라우저 기반 SSH 터미널을 사용하여이 자습서의 단계를 수행합니다.



> **Note**
>
> PuTTY와 같은 자체 SSH 클라이언트를 사용할 수도 있습니다. PuTTY 구성에 대한 자세한 내용은 Amazon Lightsail에서 SSH를 사용하여 연결하도록 PuTTY 다운로드 및 설정을 참조하세요.

2단계: Lightsail 인스턴스에 Certbot 설치

Certbot은 Let's Encrypt에서 인증서를 요청하고 이를 웹 서버에 배포하는 데 사용되는 클라이언트입니다. Let's Encrypt는 ACME 프로토콜을 사용하여 인증서를 발급하고 Certbot은 Let's Encrypt와 상호 작용하는 ACME 지원 클라이언트입니다.

Lightsail 인스턴스에 Certbot을 설치하려면

1. Lightsail 브라우저 기반 SSH 터미널과 같은 SSH 클라이언트를 사용하여 인스턴스에 연결합니다. 다음 명령을 입력하여 인스턴스의 패키지를 업데이트합니다.

   ```
   sudo apt-get update
   ```

2. 다음 명령을 입력하여 소프트웨어 속성 패키지를 설치합니다. Certbot의 개발자는 Personal Package Archive(PPA)를 사용하여 Certbot을 배포합니다. 소프트웨어 속성 패키지를 사용하면 PPA를 사용하여 보다 효율적으로 작업할 수 있습니다.

   ```
   sudo apt-get install software-properties-common -y
   ```

3. 다음 명령을 입력하여 새 리포지토리를 포함하도록 apt를 업데이트합니다.

```
sudo apt-get update -y
```

4. 다음 명령을 입력하여 Certbot을 설치합니다.

```
sudo apt-get install certbot -y
```

이제 Certbot이 Lightsail 인스턴스에 설치됩니다.

3단계: Let's Encrypt SSL 와일드카드 인증서 요청

Let's Encrypt에서 인증서를 요청하는 프로세스를 시작합니다. Certbot을 사용하여 도메인 및 하위 도메인에 대해 단일 인증서를 사용할 수 있는 와일드카드 인증서를 요청합니다. 예를 들어 `example.com` 최상위 도메인과 `blog.example.com` 및 `stuff.example.com` 하위 도메인에 대해 단일 와일드카드 인증서가 작동합니다.

Let's Encrypt SSL 와일드카드 인증서를 요청하려면

1. 이 자습서의 2단계에서 사용된 것과 동일한 브라우저 기반 SSH 터미널 창에서 다음 명령을 입력하여 도메인에 대해 환경 변수를 설정합니다. 이제 명령을 더 효율적으로 복사하고 붙여 넣어 인증서를 얻을 수 있습니다. *domain*을 등록된 도메인 이름으로 바꿔야 합니다.

```
DOMAIN=domain
WILDCARD=*.$DOMAIN
```

예제:

```
DOMAIN=example.com
WILDCARD=*.$DOMAIN
```

2. 다음 명령을 입력하여 변수가 올바른 값을 반환하는지 확인합니다.

```
echo $DOMAIN && echo $WILDCARD
```

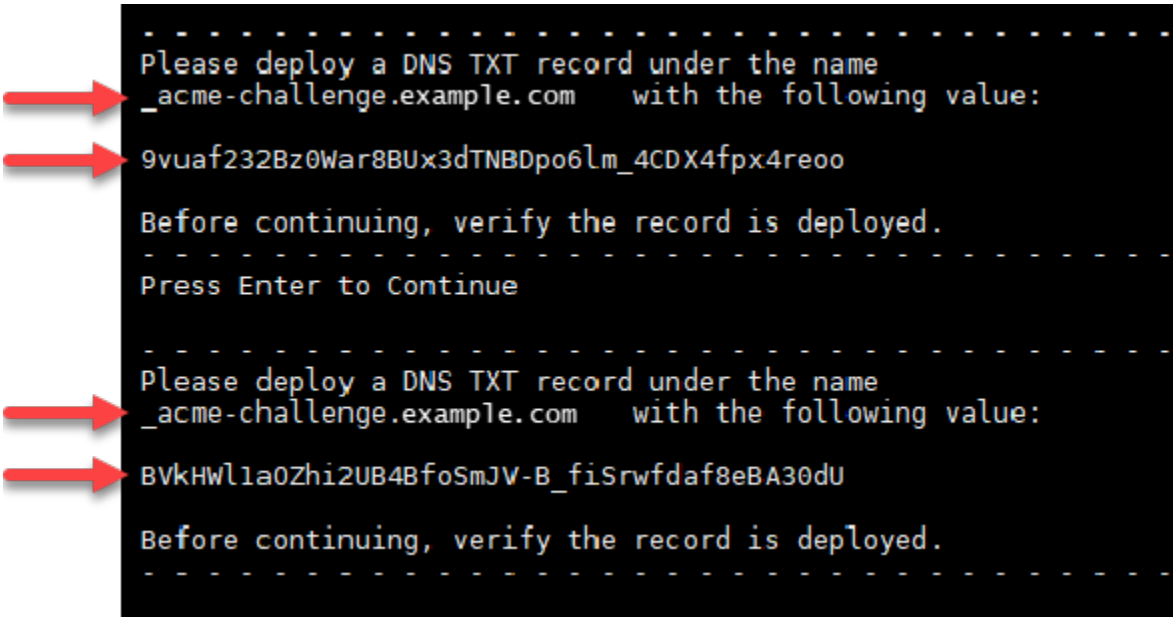
다음과 유사한 결과가 출력되어야 합니다.

3. 다음 명령을 입력하여 대화형 모드에서 Certbot을 시작합니다. 이 명령은 DNS 문제에 대해 수동 권한 부여 방법을 사용하여 도메인 소유권을 확인하도록 Certbot에 지시하며, 최상위 도메인 및 하위 도메인에 대한 와일드카드 인증서를 요청합니다.

```
sudo certbot -d $DOMAIN -d $WILDCARD --manual --preferred-challenges dns
 certonly
```

4. 메시지가 표시되면 갱신 및 보안 고지 사항에 사용될 이메일 주소를 입력합니다.
5. Let's Encrypt 서비스 계약 조건을 읽습니다. 완료되면 동의하면 Y를 누릅니다. 동의하지 않는 경우 Let's Encrypt 인증서를 얻을 수 없습니다.
6. 이메일 주소를 공유하라는 메시지와 IP 주소가 기록된다는 경고에 적절하게 응답합니다.
7. 이제 Let's Encrypt에 지정된 도메인을 소유하고 있는지 확인하라는 메시지가 표시됩니다. TXT 레코드를 도메인의 DNS 레코드에 추가하여 이 작업을 수행할 수 있습니다. 다음 예와 같이 한 세트의 TXT 레코드 값이 제공됩니다.

> **Note**
>
> Let's Encrypt는 확인에 사용해야 하는 단일 또는 여러 TXT 레코드를 제공할 수 있습니다. 이 예에서는 확인에 사용할 두 개의 TXT 레코드가 제공되었습니다.



8. Lightsail 브라우저 기반 SSH 세션을 열린 상태로 유지하세요. 이 자습서의 뒷부분에서 다시 해당 세션으로 돌아옵니다. 이 자습서의 다음 섹션으로 계속 진행합니다.

4단계: 도메인의 DNS 영역에 TXT 레코드 추가

도메인의 DNS 영역에 TXT 레코드를 추가하면 도메인 소유권이 확인됩니다. 설명을 위해 Lightsail DNS 영역을 사용하겠습니다. 하지만 이 단계는 일반적으로 도메인 등록 대행자가 호스팅하는 다른 DNS 영역과 비슷할 수 있습니다.

> **Note**
>
> 도메인의 Lightsail DNS 영역을 생성하는 방법에 대한 자세한 내용은 Lightsail에서 도메인의 DNS 레코드를 관리하기 위한 DNS 영역 생성을 참조하세요.

Lightsail의 도메인 DNS 영역에 TXT 레코드를 추가하려면

1. 왼쪽 탐색 창에서 도메인 및 DNS를 선택합니다.
2. 페이지의 DNS 영역 섹션에서 Certbot 인증서 요청에서 지정한 도메인의 DNS 영역을 선택합니다.
3. DNS 영역 편집기에서 DNS records(DNS 레코드)를 선택합니다.
4. 레코드 추가(Add record)를 선택합니다.

5. 레코드 유형(Record type) 드롭다운 메뉴에서 TXT 레코드를 선택합니다.
6. Let's Encrypt 인증서 요청에서 지정한 값을 레코드 이름(Record name) 및 응답(Responds with) 필드에 입력합니다.

   > **Note**
   >
   > Lightsail 콘솔은 도메인의 정점 부분을 미리 채웁니다. 예를 들어, *_acme-challenge.example.com* 하위 도메인을 추가하려면 텍스트 상자에 *_acme-challenge*만 입력하면 되며, 레코드를 저장할 때 Lightsail이 `.example.com` 부분을 추가합니다.

7. 저장을 선택합니다.
8. 4~7단계를 반복하여 Let's Encrypt 인증서 요청에 지정된 두 번째 TXT 레코드 세트를 추가합니다.
9. Lightsail 콘솔 브라우저 창을 열린 상태로 유지하세요. 이 자습서의 뒷부분에서 다시 해당 세션으로 돌아옵니다. 이 자습서의 다음 섹션으로 계속 진행합니다.

5단계: TXT 레코드가 전파되었는지 확인

TXT 레코드가 인터넷의 DNS에 전파되었는지 확인하려면 MxToolbox 유틸리티를 사용합니다. DNS 레코드 전파는 DNS 호스팅 공급자 및 DNS 레코드에 대해 구성된 TTL(Time to Live)에 따라 다소 시간이 걸릴 수 있습니다. 이 단계를 완료하고 Certbot 인증서 요청을 계속하기 전에 TXT 레코드가 전파되었는지 확인하는 것이 중요합니다. 그렇지 않으면 인증서 요청이 실패합니다.

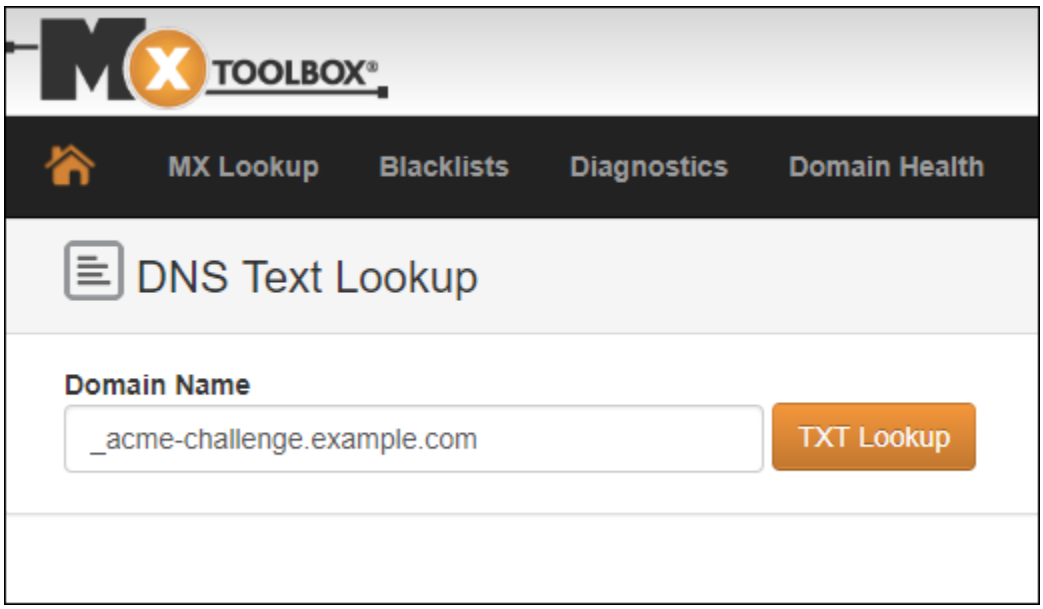
TXT 레코드가 인터넷의 DNS에 전파되었는지 확인하려면

1. 새 브라우저 창을 열고 https://mxtoolbox.com/TXTLookup.aspx로 이동합니다.
2. 다음 텍스트를 텍스트 상자에 입력합니다. *domain*을 도메인으로 바꿉니다.

   ```
   _acme-challenge.domain
   ```

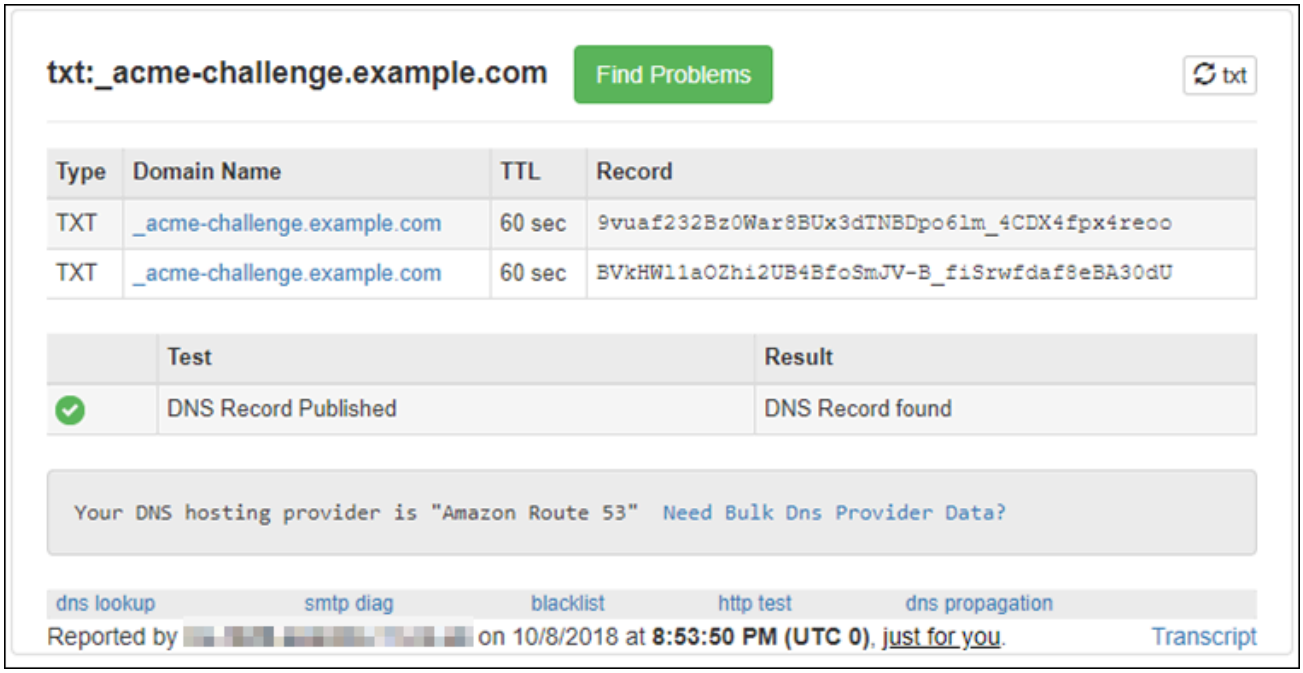
   예제:

   ```
   _acme-challenge.example.com
   ```



3. TXT 조회를 선택하여 점검을 실행합니다.
4. 다음 응답 중 하나가 발생합니다.

   - TXT 레코드가 인터넷의 DNS에 전파되면 다음 스크린샷과 비슷한 응답이 표시됩니다. 브라우저 창을 닫고 이 자습서의 다음 단원으로 계속 진행합니다.



   - TXT 레코드가 인터넷의 DNS에 전파되지 않은 경우 DNS Record not found(DNS 레코드를 찾을 수 없음) 응답이 표시됩니다. 도메인의 DNS 영역에 올바른 DNS 레코드를 추가했는지

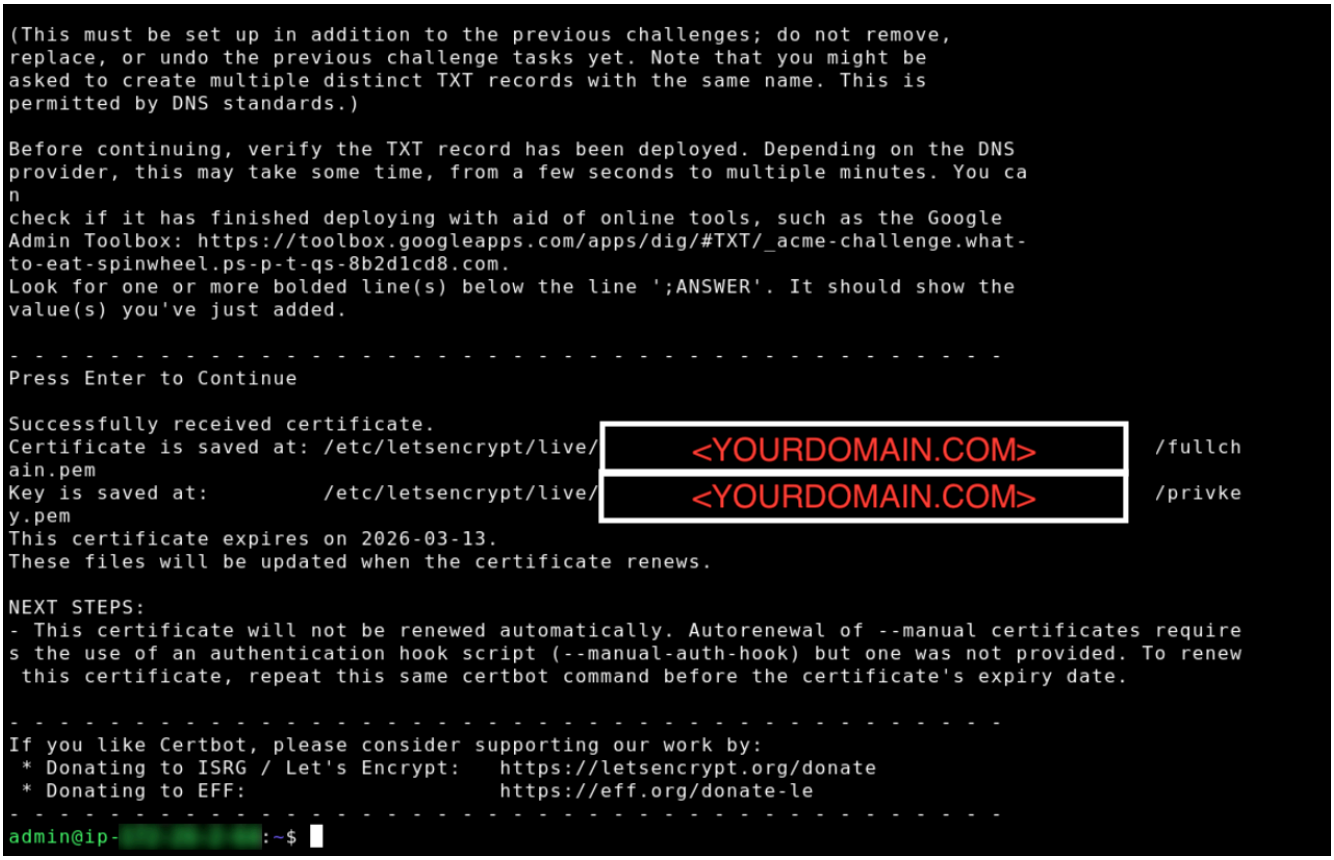
확인합니다. 올바른 레코드를 추가한 경우 도메인의 DNS 레코드가 전파되도록 잠시 기다렸다가 TXT 조회를 다시 실행합니다.

6단계: Let's Encrypt SSL 인증서 요청 완료

인스턴스의 Lightsail 브라우저 기반 SSH 세션으로 돌아가 Let's Encrypt 인증서 요청을 완료합니다. Certbot은 SSL 인증서, 체인 및 키 파일을 인스턴스의 특정 디렉터리에 저장합니다.

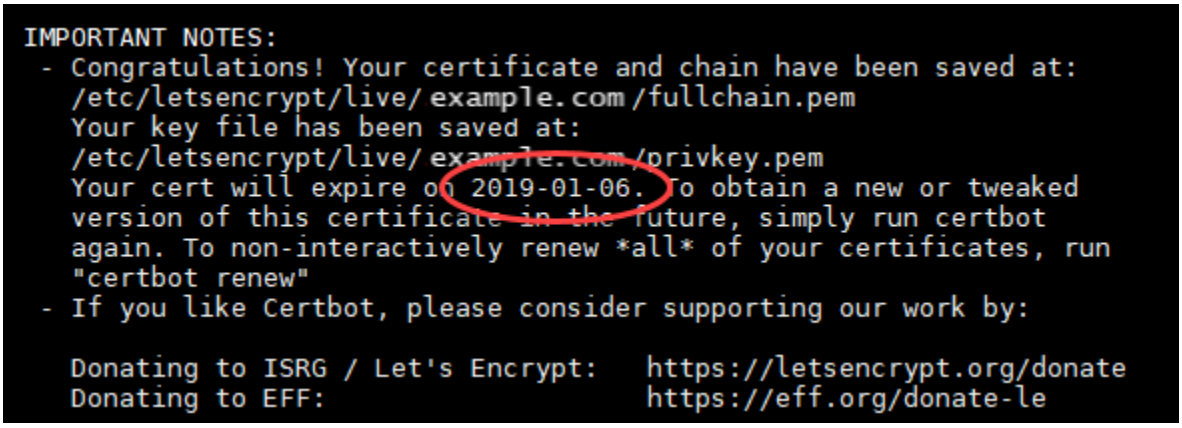
Let's Encrypt SSL 인증서 요청을 완료하려면

1. 인스턴스의 Lightsail 브라우저 기반 SSH 세션에서 Enter 키를 눌러 Let's Encrypt SSL 인증서 요청을 계속합니다. 성공하면 다음 스크린샷과 비슷한 응답이 나타납니다.



   이 메시지는 인증서, 체인 및 키 파일이 `/etc/letsencrypt/live/`*`domain`*`/` 디렉터리에 저장되어 있음을 확인합니다. *`domain`*을 `/etc/letsencrypt/live/`*`example.com`*`/`과 같은 도메인으로 바꿔야 합니다.

2. 메시지에 지정된 만료 날짜를 적어 둡니다. 이 날짜를 사용하여 해당 날짜까지 인증서를 갱신합니다.



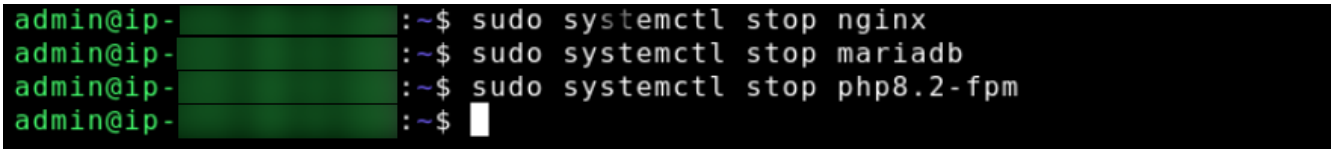
7단계: NGINX에서 SSL 구성 업데이트 및 HTTP에서 HTTPS로 트래픽 리디렉션

NGINX의 default.conf에서 SSL 구성을 업데이트하려면

1. Nginx 인스턴스의 SSH 세션에서 다음 명령을 입력하여 기본 서비스를 중지합니다.

```
sudo systemctl stop nginx
sudo systemctl stop mariadb
sudo systemctl stop php8.2-fpm
```

다음과 유사한 응답이 나타납니다.



2. 다음 명령을 입력하여 도메인에 대한 환경 변수를 설정합니다. 명령을 더 효율적으로 복사하고 붙여 넣어 인증서 파일을 연결할 수 있습니다. *domain*을 등록된 도메인 이름으로 바꿔야 합니다.

```
DOMAIN=domain
```

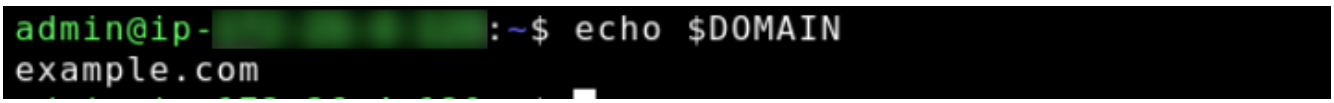
예제:

```
DOMAIN=example.com
```

3. 다음 명령을 입력하여 변수가 올바른 값을 반환하는지 확인합니다.

```
echo $DOMAIN
```

다음과 유사한 결과가 출력되어야 합니다.



4. 아래 명령을 실행하여 SSL 구성을 변경합니다.

```
sudo sed \
-i -e "s|ssl_certificate /etc/ssl/certs/ssl-cert-snakeoil.pem|ssl_certificate /
etc/letsencrypt/live/$DOMAIN/fullchain.pem|g" \
-i -e "s|ssl_certificate_key /etc/ssl/private/ssl-cert-snakeoil.key|
ssl_certificate_key /etc/letsencrypt/live/$DOMAIN/privkey.pem|g" \
/etc/nginx/conf.d/default.conf
```

5. `default.conf` 파일을 덮어쓴 후 아래 명령을 실행하여 구성을 확인하고 NGINX를 다시 시작합니다.

```
sudo nginx -t
sudo systemctl restart nginx
sudo systemctl restart mariadb
sudo systemctl restart php8.2-fpm
```

다음과 유사한 결과가 출력되어야 합니다.

이제 Nginx 인스턴스가 SSL 암호화를 사용하도록 구성되고 트래픽이 HTTP에서 HTTPS로 리디렉션됩니다.

8단계: 90일마다 Let's Encrypt 인증서 갱신

Let's Encrypt 인증서는 90일 동안 유효합니다. 인증서는 만료되기 30일 전에 갱신할 수 있습니다. Let's Encrypt 인증서를 갱신하려면 3단계: Let's Encrypt SSL 와일드카드 인증서 요청을 반복합니다.

Lightsail WordPress 인스턴스와 함께 Let's Encrypt SSL 인증서 보호

> **Tip**
>
> Amazon Lightsail은 WordPress 인스턴스에서 Let's Encrypt 인증서의 설치 및 구성을 자동화하는 안내 워크플로를 제공합니다. 이 자습서의 수동 단계를 따르지 않고 워크플로를 사용하는 것이 좋습니다. 자세한 내용은 WordPress 인스턴스 시작 및 구성을 참조하세요.

Lightsail을 사용하면 Lightsail 로드 밸런서를 사용하여 SSL/TLS를 사용하는 웹사이트 및 애플리케이션을 쉽게 보호할 수 있습니다. 그러나 Lightsail 로드 밸런서를 사용하는 것이 일반적으로 올바른 선택이 아닐 수도 있습니다. 사이트에서 로드 밸런서가 제공하는 확장성 또는 내결함성을 필요로 하지 않거나 비용 최적화를 원할 수도 있습니다. 후자의 경우에는 Let's Encrypt를 사용하여 무료 SSL 인증서를 얻을 수 있으며 그런 경우 아무런 문제가 되지 않습니다. 이러한 인증서를 Lightsail 인스턴스와 통합할 수 있습니다.

이 가이드에서는 Certbot을 사용하여 Let's Encrypt 와일드카드 인증서를 요청하고 Really Simple SSL 플러그인을 사용하여 WordPress 인스턴스와 통합하는 방법에 대해 배웁니다.

- Bitnami 인스턴스에서 사용하는 Linux 배포는 2020년 7월에 Ubuntu에서 Debian으로 변경되었습니다. 이러한 변화로 인스턴스의 Linux 배포에 따라 자습서의 일부 단계가 달라집니다. 변경 후 생성된 모든 Bitnami 블루프린트 인스턴스는 Debian Linux 배포를 사용합니다. 변경 전에 생성된 인스턴스는 Ubuntu Linux 배포를 계속 사용합니다. 인스턴스의 배포를 확인하려면 `uname -a` 명령을 실행합니다. 응답은 Ubuntu 또는 Debian을 인스턴스의 Linux 배포로 표시합니다.
- Bitnami가 많은 스택에 대한 파일 구조를 수정했습니다. 이 자습서의 파일 경로는 Bitnami 스택이 네이티브 Linux 시스템 패키지(접근법 A)를 사용하는지 또는 자체 포함 설치(접근법 B)인지 여부에 따라 달라질 수 있습니다. Bitnami 설치 유형 및 따라야 할 접근 방식을 식별하려면 다음 명령을 실행합니다.

  ```
  test ! -f "/opt/bitnami/common/bin/openssl" && echo "Approach A: Using
  system packages." || echo "Approach B: Self-contained installation."
  ```

목차

- 시작하기 전에
- 1단계: 필수 구성 요소 완성
- 2단계: Lightsail 인스턴스에 Certbot 설치

- 3단계: Let’s Encrypt SSL 와일드카드 인증서 요청
- 4단계: 도메인의 DNS 영역에 TXT 레코드 추가
- 5단계: TXT 레코드가 전파되었는지 확인
- 6단계: Let's Encrypt SSL 인증서 요청 완료
- 7단계 Let's Encrypt 인증서 파일을 Apache 서버 디렉터리에 연결하는 링크 생성
- 8단계: Really Simple SSL 플러그 인을 사용하여 SSL 인증서를 WordPress 사이트와 통합
- 9단계: 90일마다 Let's Encrypt 인증서 갱신

시작하기 전에

이 자습서를 시작하기 전에 다음을 고려해야 합니다.

Bitnami HTTPS 구성(**bncert**) 도구를 대신 사용

이 자습서에서 설명하는 단계에서는 수동 프로세스를 사용하여 SSL/TLS 인증서를 구현하는 방법을 보여줍니다. 하지만 Bitnami에서는 Bitnami HTTPS 구성(bncert) 도구를 사용하는 보다 자동화된 프로세스를 제공하며, 해당 도구는 일반적으로 Lightsail의 WordPress 인스턴스에 사전 설치되어 있습니다. 이 자습서의 수동 단계를 따르지 않고 해당 도구를 사용하는 것이 좋습니다. 이 자습서는 bncert 도구가 제공되기 전에 작성되었습니다. bncert 도구를 사용하는 방법에 대한 자세한 내용은 Amazon Lightsail의 WordPress 인스턴스에서 HTTPS 활성화를 참조하세요.

WordPress 인스턴스의 Linux 배포 식별

Bitnami 인스턴스에서 사용하는 Linux 배포는 2020년 7월에 Ubuntu에서 Debian으로 변경되었습니다. 변경 후 생성된 모든 Bitnami 블루프린트 인스턴스는 Debian Linux 배포를 사용합니다. 변경 전에 생성된 인스턴스는 Ubuntu Linux 배포를 계속 사용합니다. 이러한 변화로 인스턴스의 Linux 배포에 따라 자습서의 일부 단계가 달라집니다. 이 자습서에서 사용할 단계를 알 수 있도록 인스턴스의 Linux 배포를 식별해야 합니다. 인스턴스의 Linux 배포를 식별하려면 `uname -a` 명령을 실행합니다. 응답은 Ubuntu 또는 Debian을 인스턴스의 Linux 배포로 표시합니다.

인스턴스에 적용되는 자습서 접근 방식 식별

Bitnami는 많은 스택에 대한 파일 구조를 수정하는 중입니다. 이 자습서의 파일 경로는 Bitnami 스택이 네이티브 Linux 시스템 패키지(접근법 A)를 사용하는지 또는 자체 포함 설치(접근법 B)인지 여부에 따라 달라질 수 있습니다. Bitnami 설치 유형 및 따라야 할 접근 방식을 식별하려면 다음 명령을 실행합니다.

```
test ! -f "/opt/bitnami/common/bin/openssl" && echo "Approach A: Using
system packages." || echo "Approach B: Self-contained installation."
```

1단계: 필수 구성 요소 완성

아직 수행하지 않은 경우 다음 사전 조건을 완료하세요.

- Lightsail에 WordPress 인스턴스를 생성합니다. 자세한 내용은 인스턴스 생성을 참조하세요.
- DNS 레코드를 편집하려면 도메인 이름을 등록하고 관리 액세스 권한을 얻습니다. 자세한 내용은 DNS를 참조하세요.

  Lightsail DNS 영역을 사용하여 도메인의 DNS 레코드를 관리하는 것이 좋습니다. 자세한 내용은 DNS 영역을 생성하여 도메인의 DNS 레코드 관리를 참조하세요.
- Lightsail 콘솔의 브라우저 기반 SSH 터미널을 사용하여 이 자습서의 단계를 수행합니다. 그러나 PuTTY와 같은 자체 SSH 클라이언트를 사용할 수도 있습니다. PuTTY 구성에 대한 자세한 내용은 Amazon Lightsail에서 SSH를 사용하여 연결할 PuTTY 다운로드 및 설정을 참조하세요.

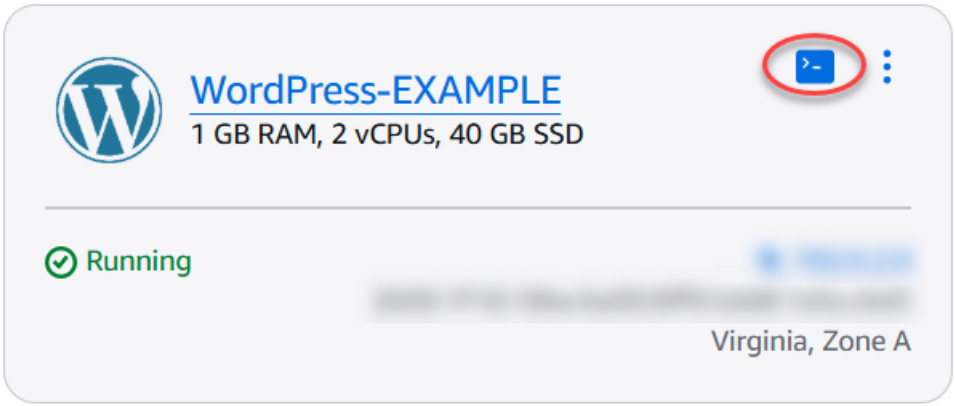
사전 조건을 완료한 후 이 자습서의 다음 단원으로 계속 진행합니다.

2단계: Lightsail 인스턴스에 Certbot 설치

Certbot은 Let's Encrypt에서 인증서를 요청하고 이를 웹 서버에 배포하는 데 사용되는 클라이언트입니다. Let's Encrypt는 ACME 프로토콜을 사용하여 인증서를 발행하고 Certbot은 Let's Encrypt와 상호 작용하는 ACME 사용 클라이언트입니다.

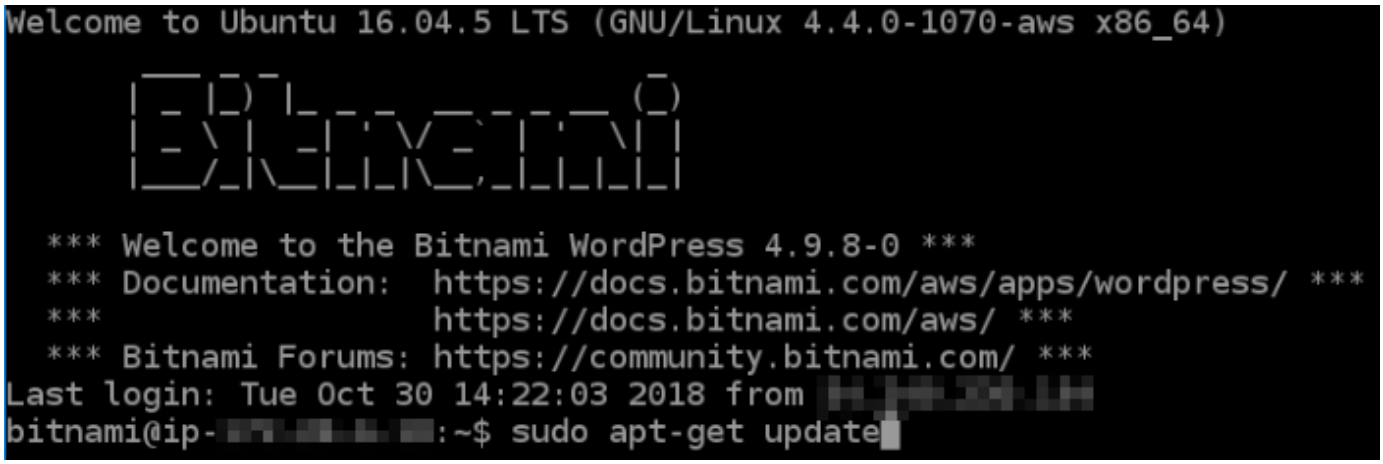
Lightsail 인스턴스에 Certbot을 설치하려면

1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 연결할 인스턴스에 대한 SSH 빠른 연결 아이콘을 선택합니다.



3. Lightsail 브라우저 기반 SSH 세션이 연결되면 다음 명령을 입력하여 인스턴스의 패키지를 업데이트합니다.

```
sudo apt-get update
```



4. 다음 명령을 입력하여 소프트웨어 속성 패키지를 설치합니다. Certbot의 개발자는 Personal Package Archive(PPA)를 사용하여 Certbot을 배포합니다. 소프트웨어 속성 패키지를 사용하면 PPA를 사용하여 보다 효율적으로 작업할 수 있습니다.

```
sudo apt-get install software-properties-common
```

> **Note**
>
> `sudo apt-get install` 명령을 실행할 때 `Could not get lock` 오류가 발생하면 약 15분 정도 기다린 후 다시 시도합니다. 이 오류는 Apt 패키지 관리 도구를 사용하여 무인 업그레이드를 설치하는 cron 작업으로 인해 발생할 수 있습니다.

5. 다음 명령을 입력하여 GPG 패키지를 설치하고 Certbot을 로컬 apt 리포지토리에 추가합니다.

> **Note**
>
> 5단계는 Ubuntu Linux 배포를 사용하는 인스턴스에만 적용됩니다. 인스턴스가 Debian Linux 배포를 사용하는 경우 이 단계를 건너뜁니다.

```
sudo apt-get install gpg -y
```

```
sudo apt-add-repository ppa:certbot/certbot -y
```

6. 다음 명령을 입력하여 새 리포지토리를 포함하도록 apt를 업데이트합니다.

```
sudo apt-get update -y
```

7. 다음 명령을 입력하여 Certbot을 설치합니다.

```
sudo apt-get install certbot -y
```

이제 Certbot이 Lightsail 인스턴스에 설치됩니다.

8. 브라우저 기반 SSH 터미널 창을 열린 상태로 유지하세요. 이 자습서의 뒷부분에서 다시 해당 세션으로 돌아옵니다. 이 자습서의 다음 섹션으로 계속 진행합니다.

3단계: Let's Encrypt SSL 와일드카드 인증서 요청

Let's Encrypt에서 인증서를 요청하는 프로세스를 시작합니다. Certbot을 사용하여 도메인 및 하위 도메인에 대해 단일 인증서를 사용할 수 있는 와일드카드 인증서를 요청합니다. 예를 들어 `example.com` 최상위 도메인과 `blog.example.com` 및 `stuff.example.com` 하위 도메인에 대해 단일 와일드카드 인증서가 작동합니다.

Let's Encrypt SSL 와일드카드 인증서를 요청하려면

1. 이 자습서의 2단계에서 사용된 것과 동일한 브라우저 기반 SSH 터미널 창에서 다음 명령을 입력하여 도메인에 대해 환경 변수를 설정합니다. 이제 명령을 더 효율적으로 복사하고 붙여 넣어 인증서를 얻을 수 있습니다. *domain*을 등록된 도메인 이름으로 바꿔야 합니다.

```
DOMAIN=domain
```

```
WILDCARD=*.$DOMAIN
```

예제:

```
DOMAIN=example.com
```

```
WILDCARD=*.$DOMAIN
```

2. 다음 명령을 입력하여 변수가 올바른 값을 반환하는지 확인합니다.

```
echo $DOMAIN && echo $WILDCARD
```

다음과 유사한 결과가 출력되어야 합니다.

3. 다음 명령을 입력하여 대화형 모드에서 Certbot을 시작합니다. 이 명령은 DNS 문제에 대해 수동 권한 부여 방법을 사용하여 도메인 소유권을 확인하도록 Certbot에 지시하며, 최상위 도메인 및 하위 도메인에 대한 와일드카드 인증서를 요청합니다.

```
sudo certbot -d $DOMAIN -d $WILDCARD --manual --preferred-challenges dns certonly
```

4. 메시지가 표시되면 갱신 및 보안 고지 사항에 사용될 이메일 주소를 입력합니다.
5. Let's Encrypt 서비스 계약 조건을 읽습니다. 모두 읽은 다음 동의하면 A를 누릅니다. 동의하지 않는 경우 Let's Encrypt 인증서를 얻을 수 없습니다.
6. 이메일 주소를 공유하라는 메시지와 IP 주소가 기록된다는 경고에 적절하게 응답합니다.
7. 이제 Let's Encrypt에 지정된 도메인을 소유하고 있는지 확인하라는 메시지가 표시됩니다. TXT 레코드를 도메인의 DNS 레코드에 추가하여 이 작업을 수행할 수 있습니다. 다음 예와 같이 한 세트의 TXT 레코드 값이 제공됩니다.

> **Note**
>
> 확인에 사용해야 하는 하나 또는 여러 개의 TXT 레코드를 Let's Encrypt에서 제공할 수 있습니다. 이 예에서는 확인에 사용할 두 개의 TXT 레코드가 제공되었습니다.

8. Lightsail 브라우저 기반 SSH 세션을 열린 상태로 유지하세요. 이 자습서의 뒷부분에서 다시 해당 세션으로 돌아옵니다. 이 자습서의 다음 섹션으로 계속 진행합니다.

4단계: 도메인의 DNS 영역에 TXT 레코드 추가

도메인의 DNS 영역에 TXT 레코드를 추가하면 도메인 소유권이 확인됩니다. 설명을 위해 Lightsail DNS 영역을 사용하겠습니다. 하지만 이 단계는 일반적으로 도메인 등록 대행자가 호스팅하는 다른 DNS 영역과 비슷할 수 있습니다.

> **Note**
>
> 도메인에 대해 Lightsail DNS 영역을 만드는 방법에 대한 자세한 내용은 DNS 영역을 생성하여 Lightsail에서 도메인의 DNS 레코드 관리를 참조하세요.

Lightsail의 도메인 DNS 영역에 TXT 레코드를 추가하려면

1. 왼쪽 탐색 창에서 도메인 및 DNS를 선택합니다.
2. 페이지의 DNS 영역 섹션에서 Certbot 인증서 요청에서 지정한 도메인의 DNS 영역을 선택합니다.
3. DNS 영역 편집기에서 DNS records(DNS 레코드)를 선택합니다.
4. 레코드 추가(Add record)를 선택합니다.
5. 레코드 유형(Record type) 드롭다운 메뉴에서 TXT 레코드를 선택합니다.
6. Let's Encrypt 인증서 요청에서 지정한 값을 레코드 이름(Record name) 및 응답(Responds with) 필드에 입력합니다.

   > **Note**
   >
   > Lightsail 콘솔은 도메인의 정점 부분을 미리 채웁니다. 예를 들어, *_acme-challenge.example.com* 하위 도메인을 추가하려면 텍스트 상자에 *_acme-challenge*만 입력하면 되며, 레코드를 저장할 때 Lightsail이 `.example.com` 부분을 추가합니다.

7. 저장을 선택합니다.
8. 4~7단계를 반복하여 Let's Encrypt 인증서 요청에 지정된 두 번째 TXT 레코드 세트를 추가합니다.
9. Lightsail 콘솔 브라우저 창을 열린 상태로 유지하세요. 이 자습서의 뒷부분에서 다시 해당 세션으로 돌아옵니다. 이 자습서의 다음 섹션으로 계속 진행합니다.

5단계: TXT 레코드가 전파되었는지 확인

TXT 레코드가 인터넷의 DNS에 전파되었는지 확인하려면 MxToolbox 유틸리티를 사용합니다. DNS 레코드 전파는 DNS 호스팅 공급자 및 DNS 레코드에 대해 구성된 TTL(Time to Live)에 따라 다소 시간이 걸릴 수 있습니다. 이 단계를 완료하고 Certbot 인증서 요청을 계속하기 전에 TXT 레코드가 전파되었는지 확인하는 것이 중요합니다. 그렇지 않으면 인증서 요청이 실패합니다.

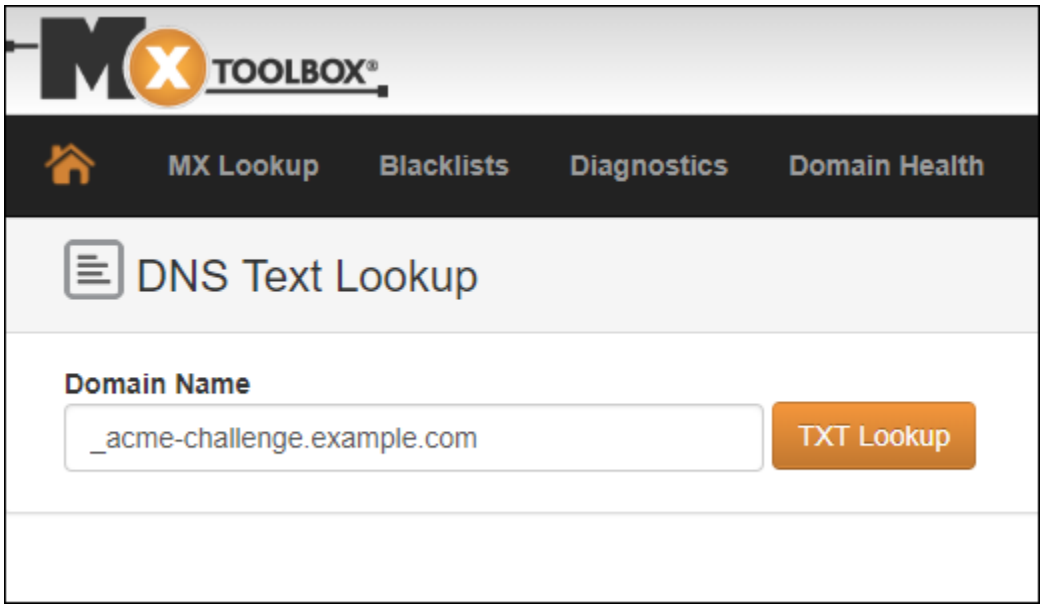
TXT 레코드가 인터넷의 DNS에 전파되었는지 확인하려면

1. 새 브라우저 창을 열고 https://mxtoolbox.com/TXTLookup.aspx로 이동합니다.
2. 다음 텍스트를 텍스트 상자에 입력합니다. *domain*을 도메인으로 바꿉니다.

   ```
   _acme-challenge.domain
   ```

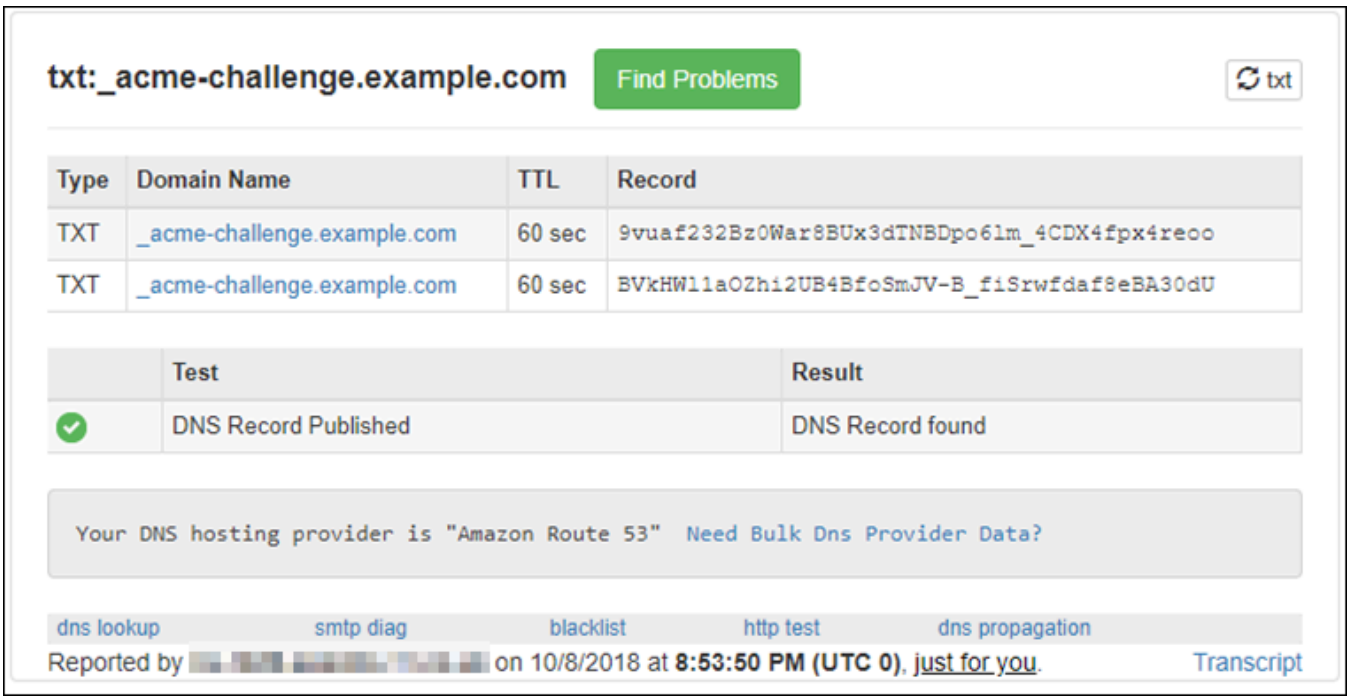
   예제:

   ```
   _acme-challenge.example.com
   ```

3. TXT 조회를 선택하여 점검을 실행합니다.
4. 다음 응답 중 하나가 발생합니다.

   - TXT 레코드가 인터넷의 DNS에 전파되면 다음 스크린샷과 비슷한 응답이 표시됩니다. 브라우저 창을 닫고 이 자습서의 다음 단원으로 계속 진행합니다.



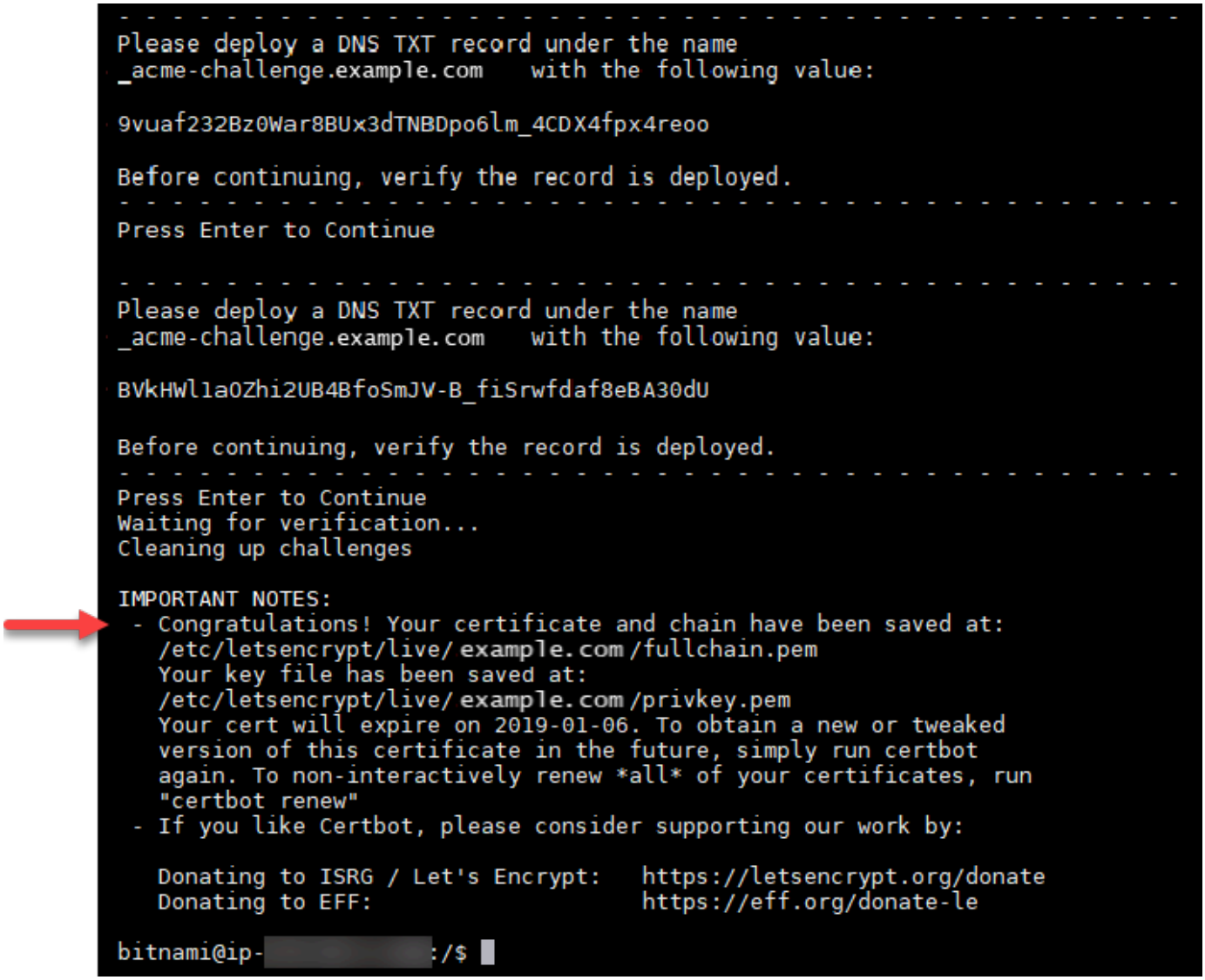
- TXT 레코드가 인터넷의 DNS에 전파되지 않은 경우 DNS Record not found(DNS 레코드를 찾을 수 없음) 응답이 표시됩니다. 도메인의 DNS 영역에 올바른 DNS 레코드를 추가했는지 확인합니다. 올바른 레코드를 추가한 경우 도메인의 DNS 레코드가 전파되도록 잠시 기다렸다가 TXT 조회를 다시 실행합니다.

6단계: Let's Encrypt SSL 인증서 요청 완료

WordPress 인스턴스의 Lightsail 브라우저 기반 SSH 세션으로 돌아가서 Let's Encrypt 인증서 요청을 완료합니다. Certbot은 SSL 인증서, 체인 및 키 파일을 WordPress 인스턴스의 특정 디렉터리에 저장합니다.

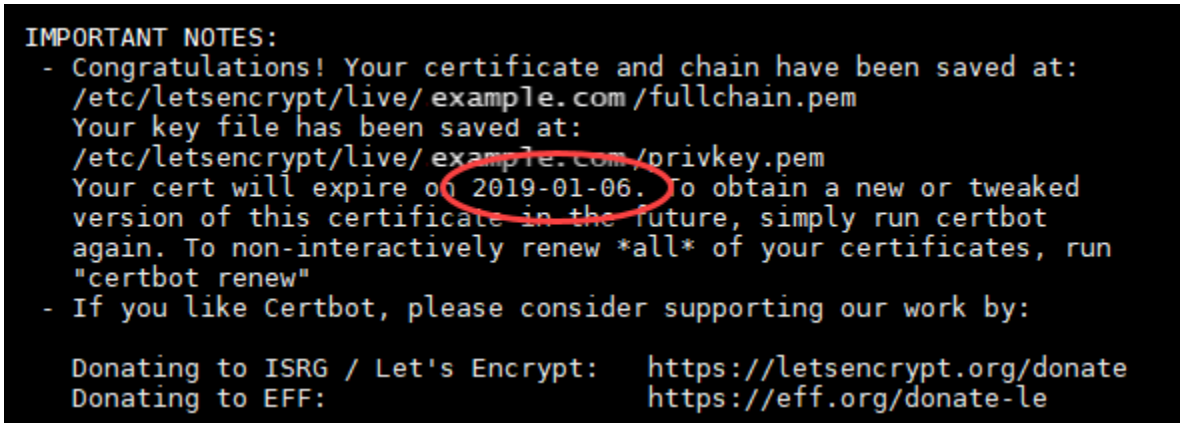
Let's Encrypt SSL 인증서 요청을 완료하려면

1. WordPress 인스턴스의 Lightsail 브라우저 기반 SSH 세션에서 Enter 키를 눌러 Let's Encrypt SSL 인증서 요청을 계속 진행합니다. 성공하면 다음 스크린샷과 비슷한 응답이 나타납니다.



이 메시지는 인증서, 체인 및 키 파일이 /etc/letsencrypt/live/*domain*/ 디렉터리에 저장되어 있음을 확인합니다. *domain*을 /etc/letsencrypt/live/*example.com*/과 같은 도메인으로 바꿔야 합니다.

2. 메시지에 지정된 만료 날짜를 적어 둡니다. 이 날짜를 사용하여 해당 날짜까지 인증서를 갱신합니다.



3. 이제 Let's Encrypt SSL 인증서가 있으므로 이 자습서의 다음 단원으로 계속 진행합니다.

7단계 Let's Encrypt 인증서 파일을 Apache 서버 디렉터리에 연결하는 링크 생성

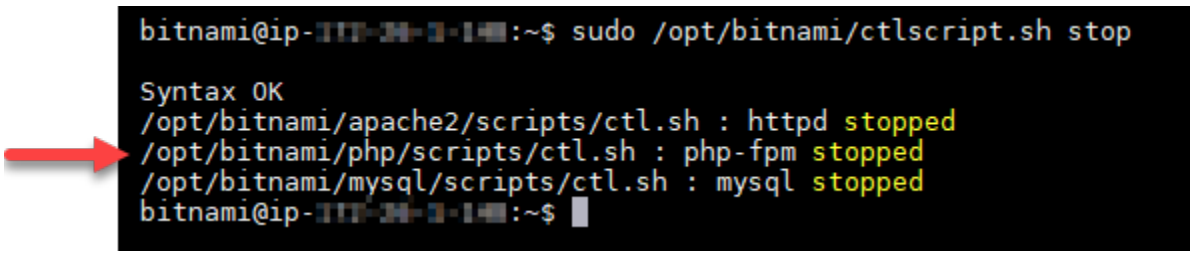
Let's Encrypt SSL 인증서 파일을 WordPress 인스턴스의 Apache 서버 디렉터리에 연결하는 링크를 생성합니다. 또한 필요할 때를 대비하여 기존 인증서를 백업합니다.

Let's Encrypt 인증서 파일을 Apache 서버 디렉터리에 연결하는 링크를 생성하려면

1. WordPress 인스턴스에 대한 Lightsail 브라우저 기반 SSH 세션에서 다음 명령을 입력하여 기본 서비스를 중지합니다.

```
sudo /opt/bitnami/ctlscript.sh stop
```

다음과 유사한 응답이 나타납니다.



2. 다음 명령을 입력하여 도메인에 대한 환경 변수를 설정합니다. 명령을 더 효율적으로 복사하고 붙여 넣어 인증서 파일을 연결할 수 있습니다. *domain*을 등록된 도메인 이름으로 바꿔야 합니다.

```
DOMAIN=domain
```

예제:

```
DOMAIN=example.com
```

3. 다음 명령을 입력하여 변수가 올바른 값을 반환하는지 확인합니다.

```
echo $DOMAIN
```

다음과 유사한 결과가 출력되어야 합니다.

4. 다음 명령을 개별적으로 입력하여 백업으로 기존 인증서 파일 이름을 다시 지정합니다. 다양한 배포 및 파일 구조에 대한 정보는 이 자습서의 시작 부분에 있는 중요 블록을 참조하세요.

   - Debian Linux 배포의 경우

     접근 방식 A(시스템 패키지를 사용한 Bitnami 설치):

     ```
     sudo mv /opt/bitnami/apache2/conf/bitnami/certs/server.crt /opt/bitnami/apache2/
     conf/bitnami/certs/server.crt.old
     ```

     ```
     sudo mv /opt/bitnami/apache2/conf/bitnami/certs/server.key /opt/bitnami/apache2/
     conf/bitnami/certs/server.key.old
     ```

     접근 방식 B(자체 포함 Bitnami 설치):

     ```
     sudo mv /opt/bitnami/apache2/conf/server.crt /opt/bitnami/apache2/conf/
     server.crt.old
     ```

     ```
     sudo mv /opt/bitnami/apache2/conf/server.key /opt/bitnami/apache2/conf/
     server.key.old
     ```

   - Ubuntu Linux 배포를 사용하는 이전 인스턴스의 경우:

     ```
     sudo mv /opt/bitnami/apache/conf/bitnami/certs/server.crt /opt/bitnami/apache/
     conf/bitnami/certs/server.crt.old
     ```

```
sudo mv /opt/bitnami/apache/conf/bitnami/certs/server.key /opt/bitnami/apache/
conf/bitnami/certs/server.key.old
```

```
sudo mv /opt/bitnami/apache/conf/bitnami/certs/server.csr /opt/bitnami/apache/
conf/bitnami/certs/server.csr.old
```

5. 다음 명령을 개별적으로 입력하여 Apache 디렉터리에 Let's Encrypt 인증서 파일에 대한 링크를 생성합니다. 다양한 배포 및 파일 구조에 대한 정보는 이 자습서의 시작 부분에 있는 중요 블록을 참조하세요.

   - Debian Linux 배포의 경우

     접근 방식 A(시스템 패키지를 사용한 Bitnami 설치):

     ```
     sudo ln -sf /etc/letsencrypt/live/$DOMAIN/privkey.pem /opt/bitnami/apache2/conf/
     bitnami/certs/server.key
     ```

     ```
     sudo ln -sf /etc/letsencrypt/live/$DOMAIN/fullchain.pem /opt/bitnami/apache2/
     conf/bitnami/certs/server.crt
     ```

     접근 방식 B(자체 포함 Bitnami 설치):

     ```
     sudo ln -sf /etc/letsencrypt/live/$DOMAIN/privkey.pem /opt/bitnami/apache2/conf/
     server.key
     ```

     ```
     sudo ln -sf /etc/letsencrypt/live/$DOMAIN/fullchain.pem /opt/bitnami/apache2/
     conf/server.crt
     ```

   - Ubuntu Linux 배포를 사용하는 이전 인스턴스의 경우:

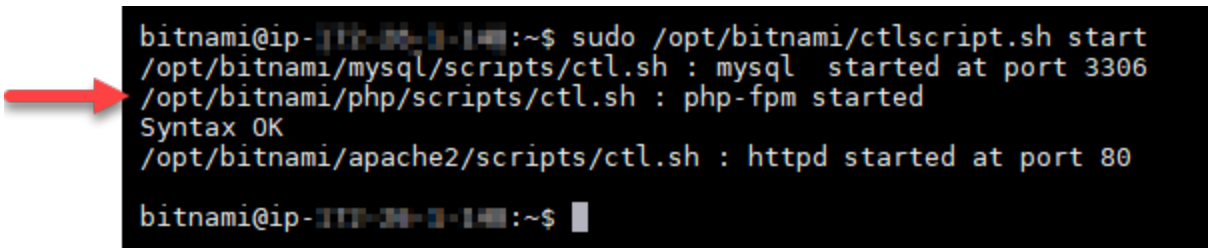
     ```
     sudo ln -s /etc/letsencrypt/live/$DOMAIN/privkey.pem /opt/bitnami/apache/conf/
     bitnami/certs/server.key
     ```

     ```
     sudo ln -s /etc/letsencrypt/live/$DOMAIN/fullchain.pem /opt/bitnami/apache/conf/
     bitnami/certs/server.crt
     ```

6. 다음 명령을 입력하여 이전에 중지한 기본 서비스를 시작합니다.

```
sudo /opt/bitnami/ctlscript.sh start
```

다음과 유사한 결과가 출력되어야 합니다.



이제 WordPress 인스턴스의 SSL 인증서 파일이 올바른 디렉터리에 있습니다.

7. 이 자습서의 다음 섹션으로 계속 진행합니다.

8단계: Really Simple SSL 플러그 인을 사용하여 SSL 인증서를 WordPress 사이트와 통합

Really Simple SSL 플러그 인을 WordPress 사이트에 설치하고 SSL 인증서를 통합하는 데 사용합니다. Really Simple SSL은 또한 사이트를 방문하는 사용자가 항상 HTTPS 연결을 유지하도록 HTTPS-HTTP 리디렉션을 구성합니다.

Really Simple SSL 플러그 인을 사용하여 SSL 인증서를 WordPress 사이트와 통합하려면

1. WordPress 인스턴스의 Lightsail 브라우저 기반 SSH 세션에서 다음 명령을 입력하여 `wp-config.php` 및 `htaccess.conf` 파일을 쓰기 가능하도록 설정합니다. Really Simple SSL 플러그 인에서 wp-config.php 파일에 기록하여 인증서를 구성합니다.

   - Debian Linux 배포를 사용하는 최신 인스턴스의 경우:

     ```
     sudo chmod 666 /opt/bitnami/wordpress/wp-config.php && sudo chmod 666 /opt/
     bitnami/apache/conf/vhosts/htaccess/wordpress-htaccess.conf
     ```

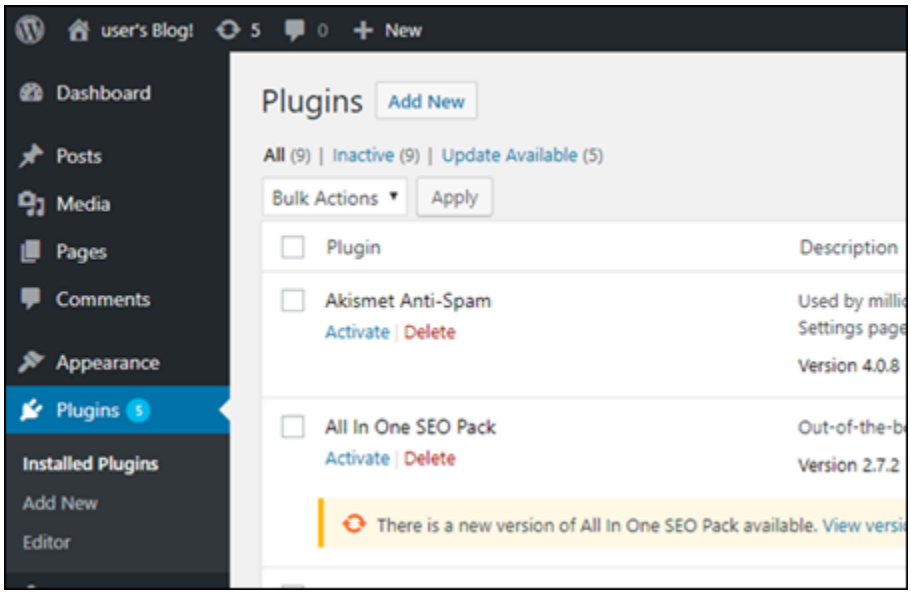
   - Ubuntu Linux 배포를 사용하는 이전 인스턴스의 경우:

     ```
     sudo chmod 666 /opt/bitnami/apps/wordpress/htdocs/wp-config.php && sudo chmod
      666 /opt/bitnami/apps/wordpress/conf/htaccess.conf
     ```

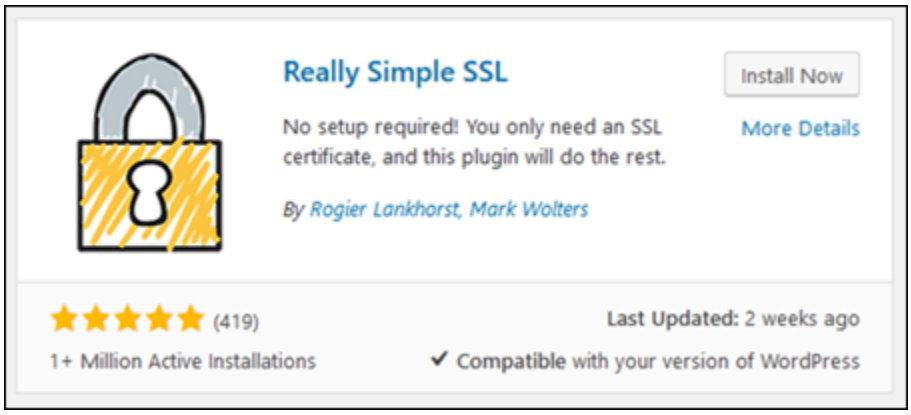
2. 새 브라우저 창을 열고 WordPress 인스턴스의 관리 대시보드에 로그인합니다.

> **Note**
>
> 자세한 내용은 Amazon Lightsail의 Bitnami 인스턴스에 대한 애플리케이션 사용자 이름과 암호 가져오기를 참조하세요.

3. 왼쪽 탐색 창에서 플러그인을 선택합니다.
4. 플러그인 페이지 상단에서 새로 추가를 선택합니다.



5. Really Simple SSL을 검색합니다.
6. 검색 결과에서 Really Simple SSL 플러그 인 옆에 있는 지금 설치(Install Now)를 선택합니다.



7. 설치가 완료되면 활성화를 선택합니다.
8. 나타나는 메시지에서 Go ahead, activate SSL!(SSL 활성화)을 선택합니다. WordPress 인스턴스의 관리 대시보드에 대한 로그인 페이지로 리디렉션될 수 있습니다.

이제 WordPress 인스턴스가 SSL 암호화를 사용하도록 구성되었습니다. 또한 이제 WordPress 인스턴스가 HTTP에서 HTTPS로의 연결을 자동으로 리디렉션하도록 구성되었습니다. 방문자가 `http://example.com`으로 이동하면 자동으로 암호화된 HTTPS 연결(즉, `https://example.com`)로 리디렉션됩니다.

9단계: 90일마다 Let's Encrypt 인증서 갱신

Let's Encrypt 인증서는 90일 동안 유효합니다. 인증서는 만료되기 30일 전에 갱신할 수 있습니다. Let's Encrypt 인증서를 갱신하려면 인증서를 가져오는 데 사용한 원래 명령을 실행합니다. 이 자습서의 Let's Encrypt SSL 와일드카드 인증서 요청 단원에 나오는 단계를 반복합니다.

특정 인스턴스 유형에 대한 단계별 지침을 따릅니다. 각 주제에서는 인스턴스의 Linux 배포(Ubuntu 또는 Debian) 및 Bitnami 설치 유형(시스템 패키지 또는 독립형)에 맞게 조정된 자세한 명령 및 구성 단계를 제공합니다. 이 주제를 따르면 Let's Encrypt의 무료 SSL/TLS 인증서로 Lightsail 웹사이트와 애플리케이션을 보호하여 암호화된 통신과 방문자의 보안 향상을 보장할 수 있습니다.

Lightsail 인스턴스 IPv6 네트워킹 구성

이 섹션에서는 Lightsail 인스턴스 블루프린트에서 IPv6 구성과 관련된 다음 주제를 다룹니다.

주제

- Lightsail의 cPanel 인스턴스에서 IPv6 구성
- Lightsail의 GitLab 인스턴스에서 IPv6 구성
- Lightsail의 Nginx 인스턴스에서 IPv6 구성
- Lightsail의 Plesk 인스턴스에서 IPv6 구성

Lightsail의 cPanel 인스턴스에서 IPv6 구성

Amazon Lightsail의 모든 인스턴스에는 기본적으로 퍼블릭 및 프라이빗 IPv4 주소가 할당되어 있습니다. 필요에 따라 인스턴스에서 IPv6를 사용하도록 설정하여 퍼블릭 IPv6 주소를 인스턴스에 할당할 수 있습니다. 자세한 내용은 Amazon Lightsail IP 주소 및 IPv6 사용 또는 사용 중지를 참조하세요.

cPanel & WHM 블루프린트를 사용하는 인스턴스에서 IPv6를 사용하도록 설정한 후에는 추가 단계를 수행하여 인스턴스가 IPv6 주소를 인식하도록 해야 합니다. 이 가이드에서는 cPanel & WHM 인스턴스에 대해 수행해야 하는 추가 단계를 안내합니다.

사전 조건

아직 수행하지 않은 경우 다음 사전 조건을 완료합니다.

- Lightsail에서 cPanel & WHM 인스턴스를 생성합니다. 자세한 내용은 인스턴스 생성을 참조하세요.
- cPanel & WHM 인스턴스를 구성합니다. 자세한 내용은 빠른 시작 안내서: Amazon Lightsail의 cPanel & WHM을 참조하세요.

> **Important**
>
> 이 가이드의 단계를 계속 진행하기 전에 모든 소프트웨어를 업데이트하고 필수 시스템을 재부팅해야 합니다.

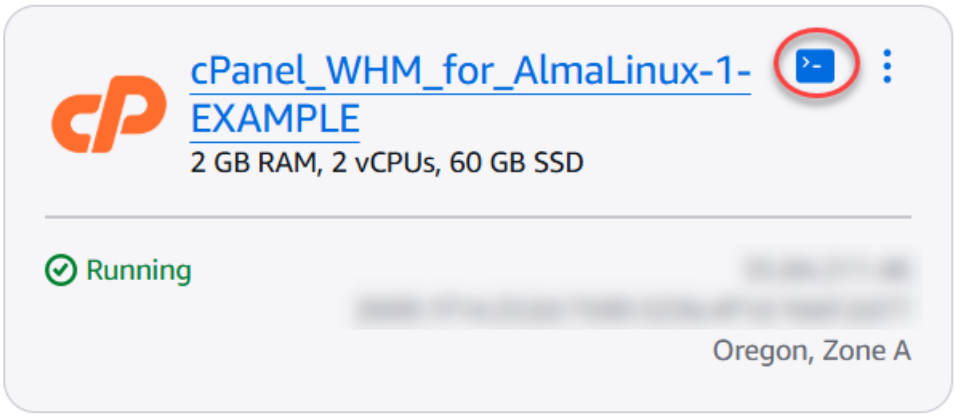
- cPanel & WHM 인스턴스에 대해 IPv6를 사용하도록 설정합니다. 자세한 내용은 IPv6 사용 또는 사용 중지를 참조하세요.

> **Note**
>
> 2021년 1월 12일 이후에 새로 생성된 cPanel & WHM 인스턴스는 Lightsail 콘솔에서 생성될 때 IPv6를 기본적으로 사용하도록 설정됩니다. 인스턴스를 생성할 때 기본적으로 IPv6를 사용하도록 설정되어 있더라도 이 가이드의 다음 단계를 완료해야 인스턴스에서 IPv6를 구성할 수 있습니다.

cPanel & WHM 인스턴스에 IPv6 구성

Lightsail에서 cPanel & WHM 인스턴스에 IPv6를 구성하려면 다음 절차를 수행하세요.

1. Lightsail 콘솔에 로그인합니다.
2. Lightsail 홈 페이지의 인스턴스 섹션에서 구성할 cPanel 및 WHM 인스턴스를 찾은 다음, 브라우저 기반 SSH 클라이언트 아이콘을 선택하여 SSH를 통해 연결합니다.



3. 인스턴스에 연결한 후 다음 명령을 입력하여 Nano를 사용하여 `ifcfg-eth0` 네트워크 인터페이스 구성 파일을 엽니다.

```
sudo nano /etc/sysconfig/network-scripts/ifcfg-eth0
```

4. 파일에 다음 텍스트 줄이 없는 경우 추가합니다.

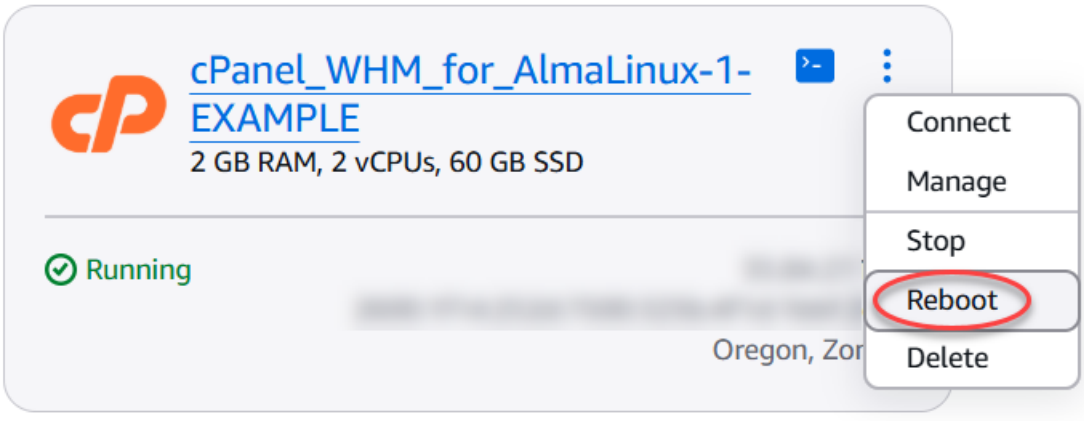
```
IPV6INIT=yes
IPV6_AUTOCONF=yes
DHCPV6C=yes
```

결과는 다음 예제와 같아야 합니다.

```
# Automatically generated by the vm import process
TYPE=Ethernet
PROXY_METHOD=none
BROWSER_ONLY=no
BOOTPROTO=dhcp
DEFROUTE=yes
IPV4_FAILURE_FATAL=no
NAME=eth0
DEVICE=eth0
ONBOOT=yes
IPV6INIT=yes
IPV6_FAILURE_FATAL=no
DHCPV6C=yes
IPV6_AUTOCONF=yes
```

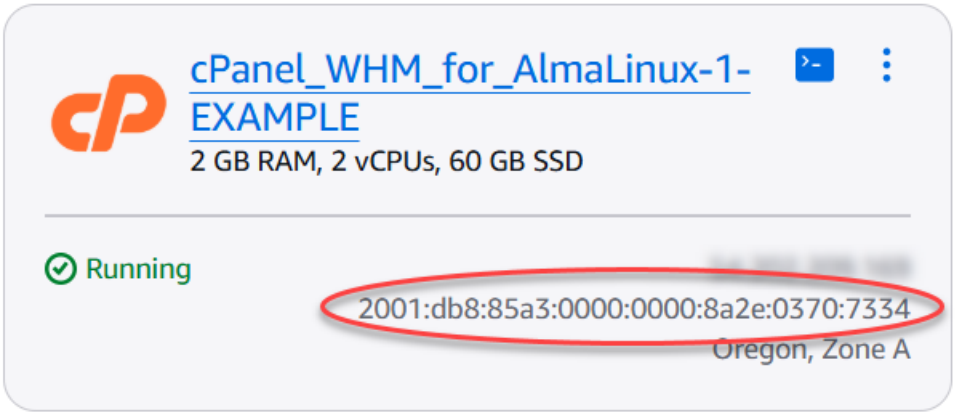
5. 파일을 종료하려면 키보드에서 Ctrl+C 키를 누릅니다.
6. 수정된 버퍼를 저장할지 묻는 메시지가 표시되면 Y 키와 Enter 키를 차례로 눌러 기존 파일에 저장합니다. 이렇게 하면 편집한 내용이 `ifcfg-eth0` 네트워크 인터페이스 구성 파일에 저장됩니다.
7. 브라우저 기반 SSH 창을 닫고 Lightsail 콘솔로 돌아갑니다.

8. Lightsail 홈 페이지의 인스턴스 섹션에서 cPanel & WHM 인스턴스의 작업 메뉴(⋮)를 선택하고 재부팅을 선택합니다.

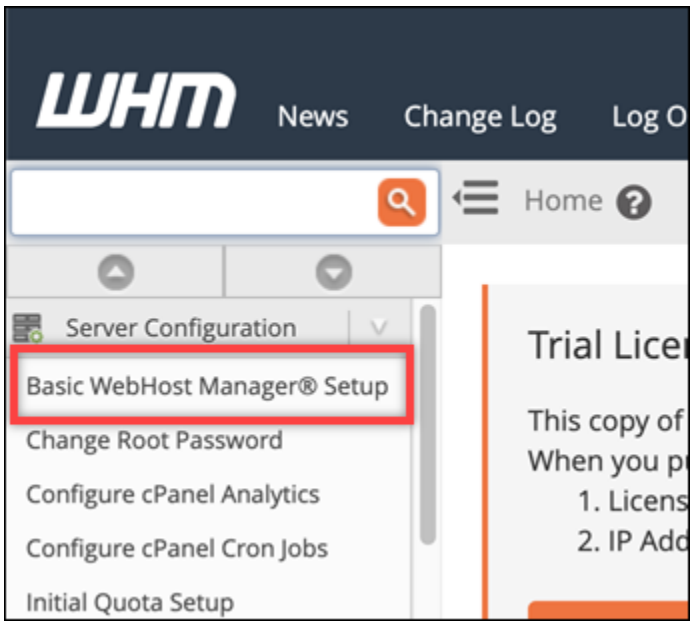


몇 분 정도 기다린 후 인스턴스가 재부팅되면 다음 단계를 수행합니다.

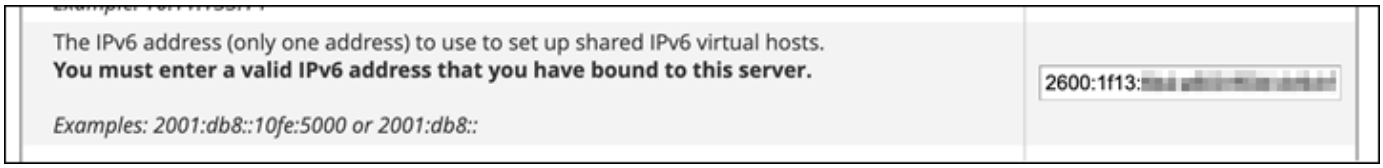
9. Lightsail 홈 페이지의 인스턴스 섹션에서 cPanel & WHM 인스턴스에 할당된 IPv6 주소를 기록해 둡니다.



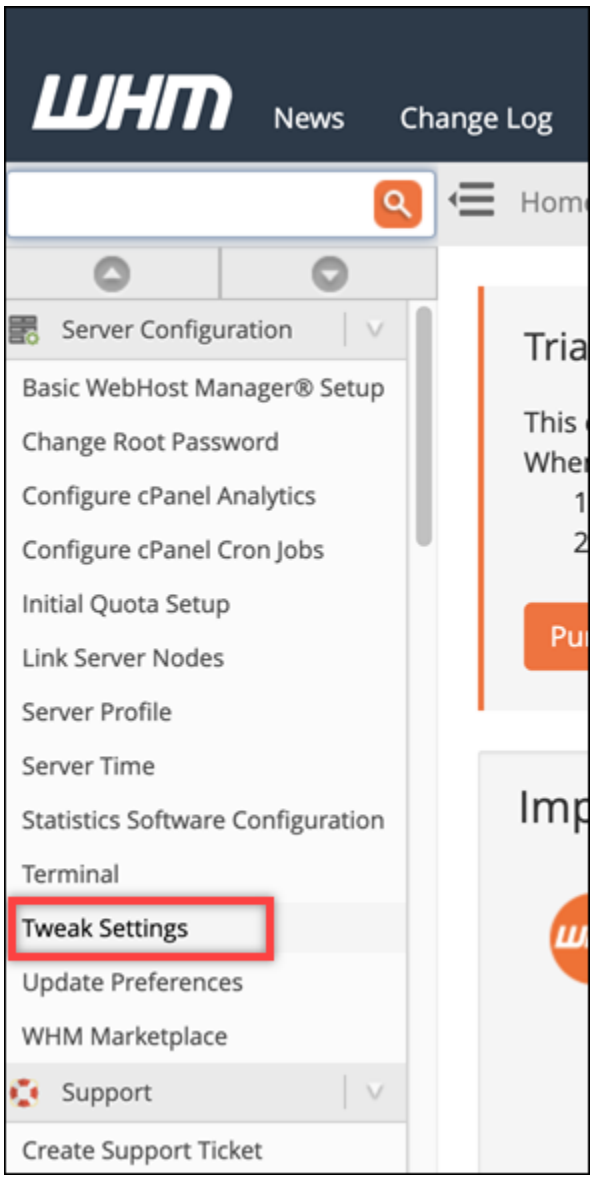
10. 새 브라우저 탭을 열고 cPanel & WHM 인스턴스의 웹 호스트 관리자(WHM)에 로그인합니다.
11. WHM 콘솔의 왼쪽 탐색 창에서 기본 웹 호스트 관리자 설정(Basic WebHost Manager Setup)을 선택합니다.



12. 모두(All) 탭에서 사용할 IPv6 주소 텍스트를 찾아 인스턴스에 할당된 IPv6 주소를 입력합니다. 이 절차의 9단계에서 인스턴스에 할당된 IPv6 주소를 기록해 두었어야 합니다.

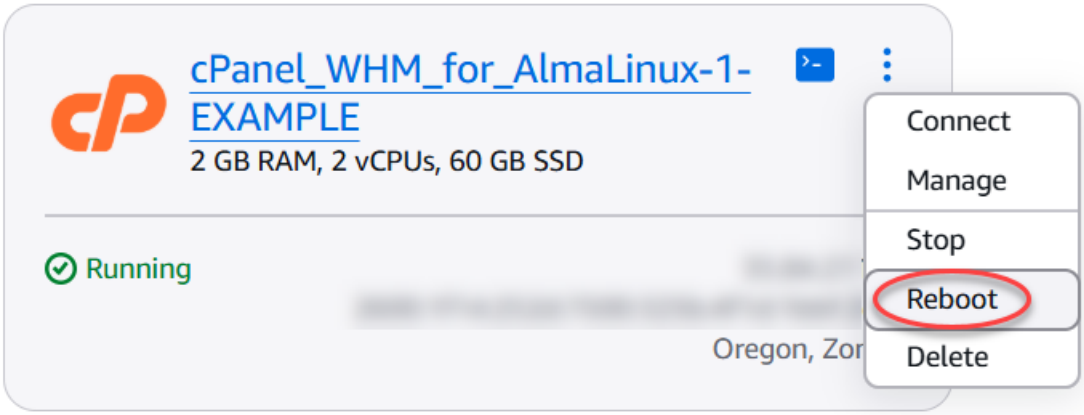


13. 페이지 맨 아래로 스크롤하고 변경 내용 저장(Save Changes)을 선택합니다.
14. WHM 콘솔의 왼쪽 탐색 창에서설정 수정(Tweak Settings)을 선택합니다.



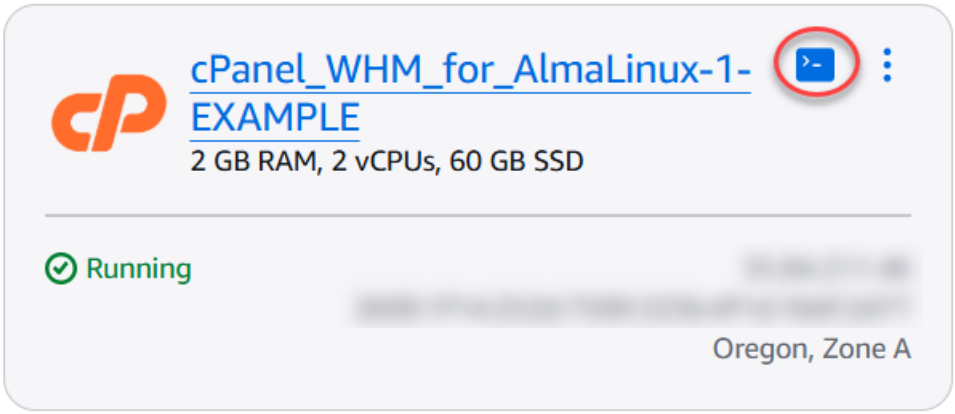
15. 모두(All) 탭에서 아래로 스크롤하여 IPv6 주소 수신 대기(Listen on IPv6 Addresses) 설정을 찾아 켬(On)으로 설정합니다.

16. 페이지의 하단으로 스크롤하고 저장(Save)을 선택합니다.
17. Lightsail 콘솔로 돌아갑니다.
18. Lightsail 홈 페이지의 인스턴스 섹션에서 cPanel & WHM 인스턴스의 작업 메뉴(:)를 선택하고 재부팅을 선택합니다.



몇 분 정도 기다린 후 인스턴스가 재부팅되면 다음 단계를 수행합니다.

19. cPanel & WHM 인스턴스가 SSH를 사용하여 연결할 브라우저 기반 SSH 클라이언트 아이콘을 선택합니다.



20. 인스턴스에 연결한 후 다음 명령을 입력하여 인스턴스에 구성된 IP 주소를 살펴보고, 할당된 IPv6 주소를 인식하는지 확인합니다.

```
ip addr
```

다음 예와 비슷한 응답이 나타나는 것을 볼 수 있습니다. 인스턴스가 IPv6 주소를 인식하는 경우 이 예와 같이 전역 범위 레이블과 함께 응답에 나열됩니다.

```
[centos@ ~]$ ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 9001 qdisc mq state UP group default qlen 1000
    link/ether 02:9b:51:92:50:45 brd ff:ff:ff:ff:ff:ff
    inet /20 brd  scope global dynamic eth0
    inet6 2600:1f13:  :6d59:4ac0/128 scope global dynamic
    inet6 fe80:  :5045/64 scope link
       valid_lft forever preferred_lft forever
```

21. 다음 명령을 입력하여 인스턴스가 IPv6 주소를 ping할 수 있는지 확인합니다.

```
ping6 ipv6.google.com -c 6
```

인스턴스가 IPv6 주소를 ping할 수 있음을 확인하는 다음 예와 같은 결과가 나타나는 것을 볼 수 있습니다.

```
[centos@ ~]$ ping6 ipv6.google.com
PING ipv6.google.com(sea15s12-in-x0e.1e100.net (2607:f8b0:400a:809::200e)) 56 data bytes
64 bytes from sea15s12-in-x0e.1e100.net (2607:f8b0:400a:809::200e): icmp_seq=1 ttl=103 tim
e=7.66 ms
64 bytes from sea15s12-in-x0e.1e100.net (2607:f8b0:400a:809::200e): icmp_seq=2 ttl=103 tim
e=7.70 ms
64 bytes from sea15s12-in-x0e.1e100.net (2607:f8b0:400a:809::200e): icmp_seq=3 ttl=103 tim
e=7.68 ms
64 bytes from sea15s12-in-x0e.1e100.net (2607:f8b0:400a:809::200e): icmp_seq=4 ttl=103 tim
e=7.69 ms
64 bytes from sea15s12-in-x0e.1e100.net (2607:f8b0:400a:809::200e): icmp_seq=5 ttl=103 tim
e=7.70 ms
64 bytes from sea15s12-in-x0e.1e100.net (2607:f8b0:400a:809::200e): icmp_seq=6 ttl=103 tim
e=7.68 ms
^C
--- ipv6.google.com ping statistics ---
6 packets transmitted, 6 received, 0% packet loss, time 5008ms
rtt min/avg/max/mdev = 7.667/7.690/7.702/0.052 ms
```

Lightsail의 GitLab 인스턴스에서 IPv6 구성

Amazon Lightsail의 모든 인스턴스에는 기본적으로 퍼블릭 및 프라이빗 IPv4 주소가 할당되어 있습니다. 필요에 따라 인스턴스에서 IPv6를 사용하도록 설정하여 퍼블릭 IPv6 주소를 인스턴스에 할당할 수 있습니다. 자세한 내용은 Amazon Lightsail IP 주소 및 IPv6 사용 또는 사용 중지를 참조하세요.

GitLab 블루프린트를 사용하는 인스턴스에서 IPv6를 사용하도록 설정한 후에는 추가 단계를 수행하여 인스턴스가 IPv6 주소를 인식하도록 해야 합니다. 이 가이드에서는 GitLab 인스턴스에 대해 수행해야 하는 추가 단계를 안내합니다.

사전 조건

아직 수행하지 않은 경우 다음 사전 조건을 완료합니다.

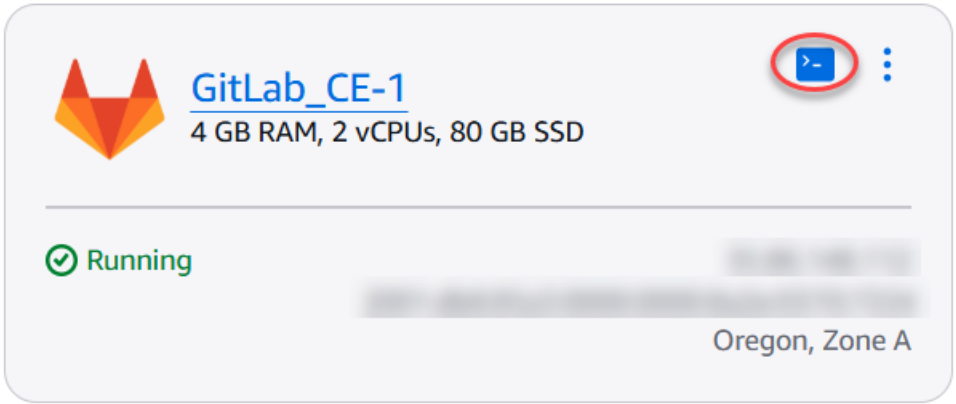
- Lightsail에서 GitLab 인스턴스를 생성합니다. 자세한 내용은 인스턴스 생성을 참조하세요.
- GitLab 인스턴스에서 IPv6를 사용하도록 설정합니다. 자세한 내용은 IPv6 사용 또는 사용 중지를 참조하세요.

> **Note**
>
> 2021년 1월 12일 이후에 새로 생성된 GitLab 인스턴스는 Lightsail 콘솔에서 생성될 때 IPv6를 기본적으로 사용하도록 설정됩니다. 인스턴스를 생성할 때 기본적으로 IPv6를 사용하도록 설정되어 있더라도 이 가이드의 다음 단계를 완료해야 인스턴스에서 IPv6를 구성할 수 있습니다.

GitLab 인스턴스에 IPv6 구성

Lightsail에서 GitLab 인스턴스에 IPv6를 구성하려면 다음 절차를 수행하세요.

1. Lightsail 콘솔에 로그인합니다.
2. Lightsail 홈 페이지의 인스턴스 섹션에서 구성할 GitLab 인스턴스를 찾은 다음 브라우저 기반 SSH 클라이언트 아이콘을 선택하여 SSH를 통해 연결합니다.

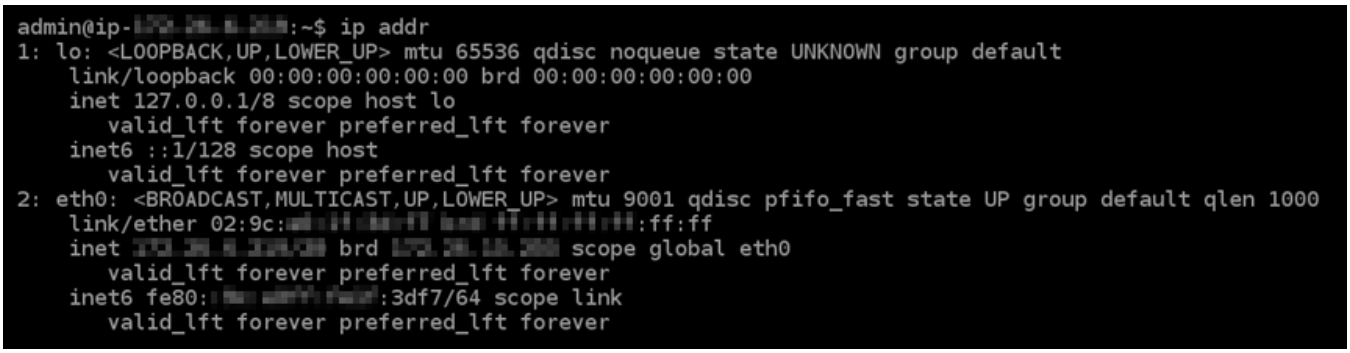


3. 인스턴스에 연결한 후 다음 명령을 입력하여 인스턴스에 구성된 IP 주소를 확인합니다.

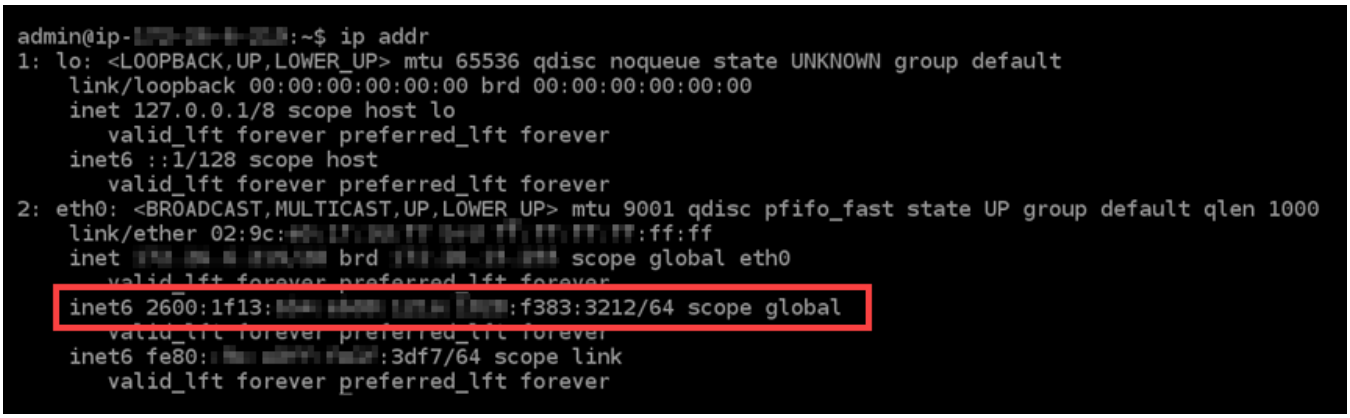
```
ip addr
```

다음 예 중 하나와 비슷한 응답이 나타나는 것을 볼 수 있습니다.

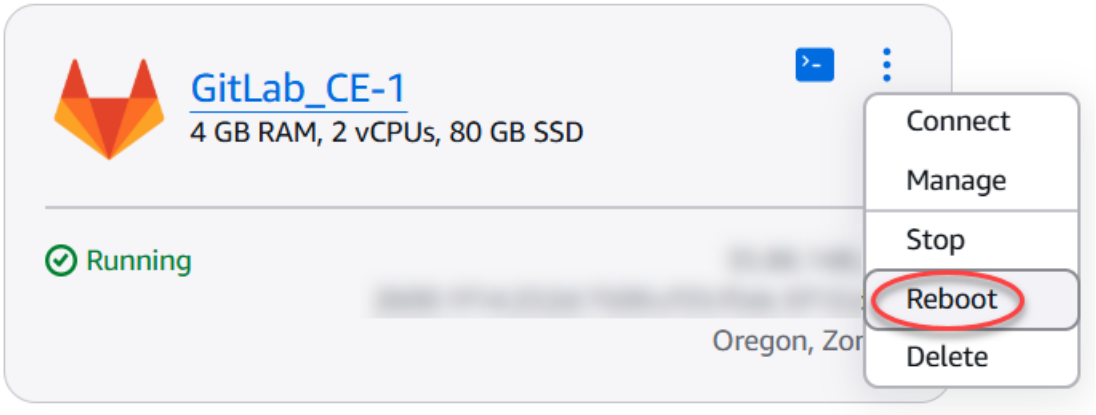
- 인스턴스가 IPv6 주소를 인식하지 못하는 경우 응답에 해당 주소가 나열되지 않습니다. 이 절차의 4단계부터 9단계까지를 이어서 완료해야 합니다.



- 인스턴스가 IPv6 주소를 인식하는 경우 이 예와 같이 `scope global`과 함께 응답에 나열됩니다. 여기서 중지해야 합니다. 인스턴스가 이미 IPv6 주소를 인식하도록 구성되어 있으므로 이 절차의 4~9단계를 완료할 필요가 없습니다.



4. Lightsail 콘솔로 돌아갑니다.
5. Lightsail 홈 페이지의 인스턴스 섹션에서 GitLab 인스턴스의 작업 메뉴(⋮)를 선택하고 재부팅을 선택합니다.

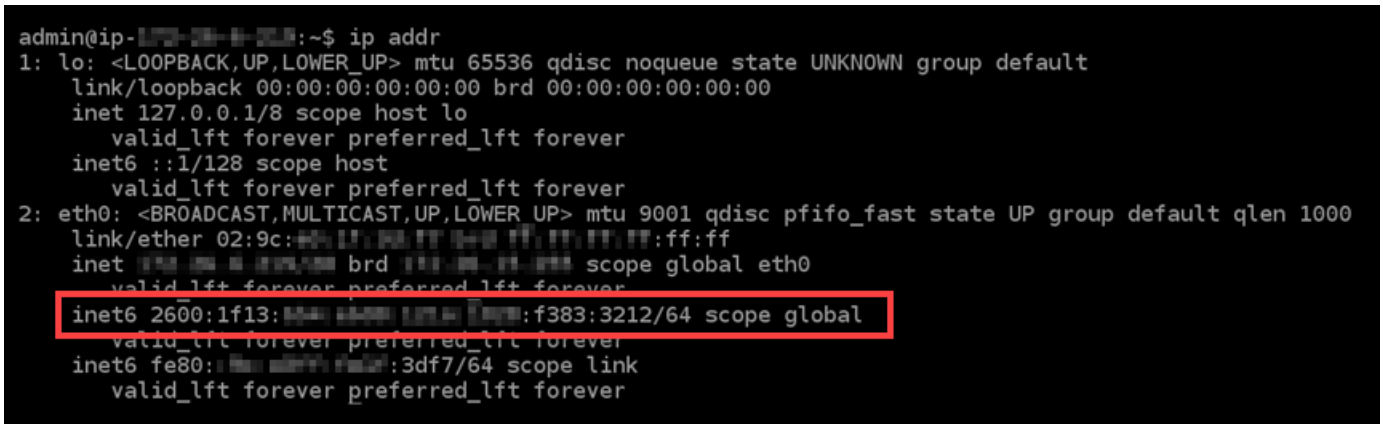


몇 분 정도 기다린 후 인스턴스가 재부팅되면 다음 단계를 수행합니다.

6. GitLab 인스턴스의 SSH 세션으로 돌아갑니다.
7. 다음 명령을 입력하여 인스턴스에 구성된 IP 주소를 살펴보고, 할당된 IPv6 주소를 인식하는지 확인합니다.

```
ip addr
```

다음 예와 비슷한 응답이 나타나는 것을 볼 수 있습니다. 인스턴스가 IPv6 주소를 인식하는 경우 이 예와 같이 `scope global` 레이블과 함께 응답에 나열됩니다.



Lightsail의 Nginx 인스턴스에서 IPv6 구성

Amazon Lightsail의 모든 인스턴스에는 기본적으로 퍼블릭 및 프라이빗 IPv4 주소가 할당되어 있습니다. 필요에 따라 인스턴스에서 IPv6를 사용하도록 설정하여 퍼블릭 IPv6 주소를 인스턴스에 할당할 수 있습니다. 자세한 내용은 Amazon Lightsail IP 주소 및 IPv6 사용 또는 사용 중지를 참조하세요.

Nginx 블루프린트를 사용하는 인스턴스에서 IPv6를 사용하도록 설정한 후에는 추가 단계를 수행하여 인스턴스가 IPv6 주소를 인식하도록 해야 합니다. 이 가이드에서는 Nginx 인스턴스에 대해 수행해야 하는 추가 단계를 안내합니다.

사전 조건

아직 수행하지 않은 경우 다음 사전 조건을 완료합니다.

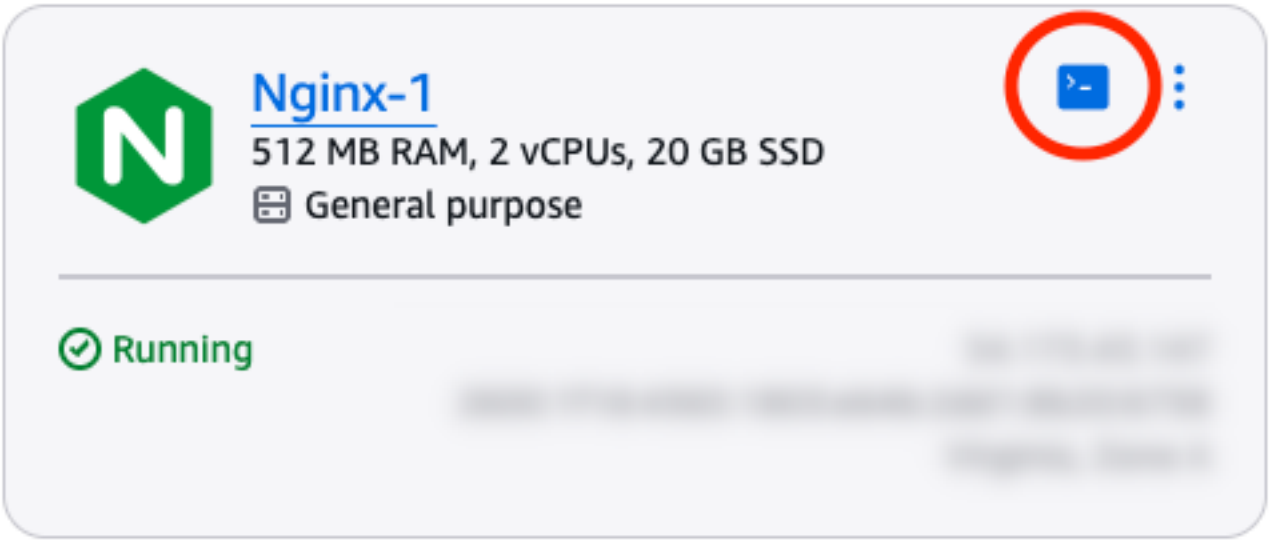
- Lightsail에서 Nginx 인스턴스를 생성합니다. 자세한 내용은 인스턴스 생성을 참조하세요.
- Nginx 인스턴스에서 IPv6를 사용하도록 설정합니다. 자세한 내용은 IPv6 사용 또는 사용 중지를 참조하세요.

> **Note**
>
> 2021년 1월 12일 이후에 새로 생성된 Nginx 인스턴스는 Lightsail 콘솔에서 생성될 때 IPv6를 기본적으로 사용하도록 설정됩니다. 인스턴스를 생성할 때 기본적으로 IPv6를 사용하도록 설정되어 있더라도 이 가이드의 다음 단계를 완료해야 인스턴스에서 IPv6를 구성할 수 있습니다.

Nginx 인스턴스에 IPv6 구성

Lightsail에서 Nginx 인스턴스에 IPv6를 구성하려면 다음 절차를 수행하세요.

1. Lightsail 콘솔에 로그인합니다.
2. Lightsail 홈 페이지의 인스턴스 섹션에서 구성할 Ubuntu 인스턴스를 찾은 다음 브라우저 기반 SSH 클라이언트 아이콘을 선택하여 SSH를 통해 연결합니다.



3. 인스턴스에 연결한 후 다음 명령을 입력하여 인스턴스가 포트 80을 통해 IPv6 요청을 수신 중인지 확인합니다. *<IPv6Address>*를 인스턴스에 할당된 IPv6 주소로 변경합니다.

```
curl -g -6 'http://[<IPv6Address>]'
```

예제:

```
curl -g -6 'http://[2001:0db8:85a3:0000:0000:8a2e:0370:7334]'
```

다음 예 중 하나와 비슷한 응답이 나타나는 것을 볼 수 있습니다.

- 인스턴스가 포트 80을 통해 IPv6 요청을 수신하지 않는 경우 연결 실패(Failed to connect) 오류 메시지와 함께 응답이 표시됩니다. 이 절차의 4단계부터 9단계까지를 이어서 완료해야 합니다.

```
bitnami@ip-[redacted]:~$ curl -g -6 'http://[2600:1f13:[redacted]:985b:25d9]:80'
curl: (7) Failed to connect to 2600:1f13:[redacted]:985b:25d9 port 80: Connection refused
```

- 인스턴스가 포트 80을 통해 IPv6 요청을 수신하는 경우 다음 예와 같이 인스턴스 홈 페이지의 HTML 코드가 있는 응답이 표시됩니다. 여기서 중지해야 합니다. 인스턴스가 이미 IPv6에 대해 구성되어 있으므로 이 절차의 4~9단계를 완료할 필요가 없습니다.

```
bitnami@ip-[redacted]:~$ curl -g -6 'http://[2600:[redacted]:985b:25d9]:80'
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8">
    <title>Bitnami NGINX Open Source</title>
    <meta name="description" content="Bitnami: Open Source. Simplified.">
    <meta name="author" content="Bitnami">
    <link rel="stylesheet" media="screen" href="//unpkg.com/@bitnami/hex/dist/hex.min.css">
  </head>
  <body>
    <main class="margin-t-huge">
      <section aria-labelledby="installation-title" aria-describedby="installation-desc" class="container container-tiny margin-b-gi
        <h1 id="installation-title">Congratulations!</h1>
        <div aria-hidden="true" style="height: 4px; width: 100%;" class="gradient-135-brand"></div>
        <p id="installation-desc" class="type-big">You are now running <strong>Bitnami NGINX Open Source 1.18.0</strong> in the Clou
      </section>
      <section aria-labelledby="links-title" aria-describedby="links-desc" class="bg-light padding-v-bigger margin-v-enormous">
        <div class="container container-tiny">
          <div class="row row-collapse-b-tablet align-center ">
            <div class="col-6">
              <h3 id="links-title" class="margin-t-reset">Useful Links</h3>
              <p id="links-desc" class="margin-b-reset">The following links will help you to understand better how to get started an
unched.</p>
```

4. 다음 명령을 입력하여 Vim을 통해 nginx.conf 구성 파일을 엽니다.

```
sudo vim /opt/bitnami/nginx/conf/nginx.conf
```

5. I를 눌러 Vim을 삽입 모드를 설정합니다.
6. 파일에 이미 있는 `listen 80;` 텍스트 아래에 다음 텍스트를 추가합니다. 텍스트를 추가해야 하는 섹션을 보려면 Vim에서 아래로 스크롤해야 할 수 있습니다.

```
listen [::]:80;
```

완료되면 파일이 다음과 같이 표시됩니다.

```
    server_tokens off;

    include  "/opt/bitnami/nginx/conf/server_blocks/*.conf";

    # HTTP Server
    server {
        # Port to listen on, can also be set in IP:PORT format
        listen  80;
        listen [::]:80;

        include  "/opt/bitnami/nginx/conf/bitnami/*.conf";

        location /status {
            stub_status on;
            access_log   off;
            allow 127.0.0.1;
            deny all;
        }
    }
```

7. Esc 키를 눌러 Vim에서 삽입 모드를 종료한 다음 `:wq!`를 입력하고 Enter 키를 눌러 편집한 내용을 저장(쓰기)하고 Vim을 종료합니다.
8. 다음 명령을 입력하여 인스턴스의 서비스를 다시 시작합니다.

```
sudo /opt/bitnami/ctlscript.sh restart
```

9. 다음 명령을 입력하여 인스턴스가 포트 80을 통해 IPv6 요청을 수신 중인지 확인합니다. *<IPv6Address>*를 인스턴스에 할당된 IPv6 주소로 변경합니다.

```
curl -g -6 'http://[<IPv6Address>]'
```

예제:

```
curl -g -6 'http://[2001:0db8:85a3:0000:0000:8a2e:0370:7334]'
```

다음 예와 비슷한 응답이 나타나는 것을 볼 수 있습니다. 인스턴스가 포트 80을 통해 IPv6 요청을 수신하는 경우 인스턴스 홈 페이지의 HTML 코드가 있는 응답이 표시됩니다.

Lightsail의 Plesk 인스턴스에서 IPv6 구성

Plesk 블루프린트를 사용하는 인스턴스가 IPv6 주소를 인식하도록 하려면 추가 단계를 수행해야 합니다. 이 가이드에서는 Plesk 인스턴스에 대해 수행해야 하는 추가 단계를 안내합니다.

사전 조건

아직 수행하지 않은 경우 다음 사전 조건을 완료합니다.

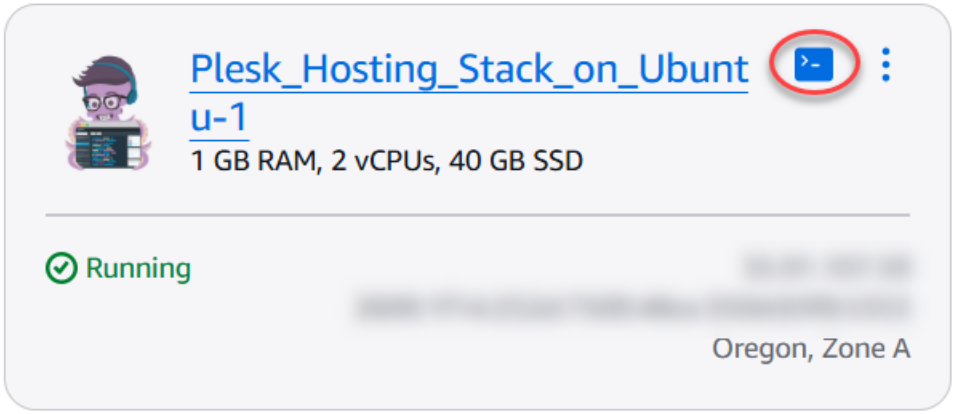
- Lightsail에서 Plesk 인스턴스를 생성합니다. 자세한 내용은 인스턴스 생성을 참조하세요.
- Plesk 인스턴스에서 IPv6를 사용하도록 설정합니다. 자세한 내용은 IPv6 사용 또는 사용 중지를 참조하세요.

> **Note**
>
> 2021년 1월 12일 이후에 생성된 Lightsail Plesk 인스턴스는 기본적으로 IPv6를 활성화하도록 설정되어 있습니다. 인스턴스를 생성할 때 기본적으로 IPv6를 사용하도록 설정되어 있더라도 이 가이드의 다음 단계를 완료해야 인스턴스에서 IPv6를 구성할 수 있습니다.

Plesk 인스턴스에 IPv6 구성

Lightsail에서 Plesk 인스턴스에 IPv6를 구성하려면 다음 절차를 수행하세요.

1. Lightsail 콘솔에 로그인합니다.
2. Lightsail 홈 페이지의 인스턴스 섹션에서 구성할 Plesk 인스턴스를 찾은 다음 브라우저 기반 SSH 클라이언트 아이콘을 선택하여 SSH를 통해 연결합니다.



3. 인스턴스에 연결한 후 다음 명령을 입력하여 인스턴스에 구성된 IP 주소를 확인합니다.

```
ip addr
```

다음 예 중 하나와 비슷한 응답이 나타나는 것을 볼 수 있습니다.

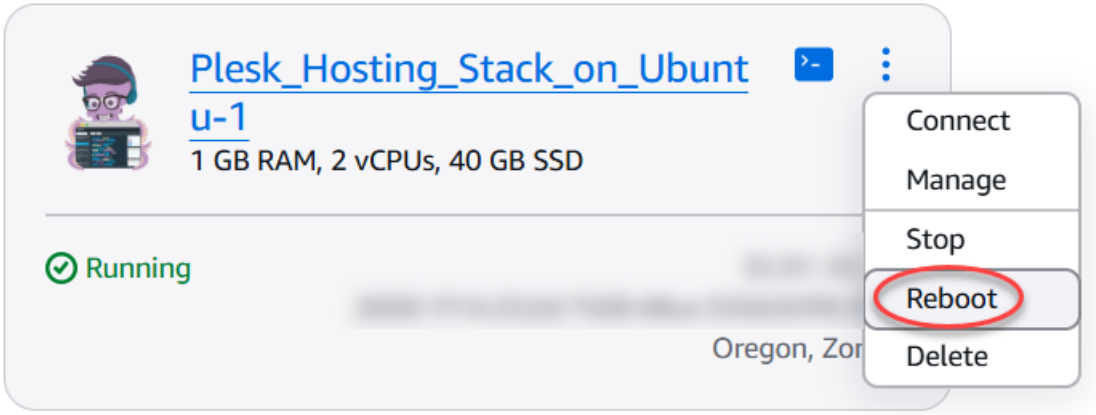
- 인스턴스가 IPv6 주소를 인식하지 못하는 경우 응답에 해당 주소가 나열되지 않습니다. 이 절차의 4단계부터 7단계까지를 이어서 완료해야 합니다.

```
admin@ip-[redacted]:~$ ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 9001 qdisc pfifo_fast state UP group default qlen 1000
    link/ether 02:9c:[redacted]:ff:ff
    inet [redacted] brd [redacted] scope global eth0
       valid_lft forever preferred_lft forever
    inet6 fe80:[redacted]:3df7/64 scope link
       valid_lft forever preferred_lft forever
```

- 인스턴스가 IPv6 주소를 인식하는 경우 이 예와 같이 `scope global`과 함께 응답에 나열됩니다. 여기서 중지해야 합니다. 인스턴스가 이미 IPv6 주소를 인식하도록 구성되어 있으므로 이 절차의 4~7단계를 완료할 필요가 없습니다.

```
admin@ip-[redacted]:~$ ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 9001 qdisc pfifo_fast state UP group default qlen 1000
    link/ether 02:9c:[redacted]:ff:ff
    inet [redacted] brd [redacted] scope global eth0
       valid_lft forever preferred_lft forever
    inet6 2600:1f13:[redacted]:f383:3212/64 scope global
       valid_lft forever preferred_lft forever
    inet6 fe80:[redacted]:3df7/64 scope link
       valid_lft forever preferred_lft forever
```

4. Lightsail 콘솔로 돌아갑니다.
5. Lightsail 홈 페이지의 인스턴스 섹션에서 Plesk 인스턴스의 작업 메뉴(⋮)를 선택하고 재부팅을 선택합니다.



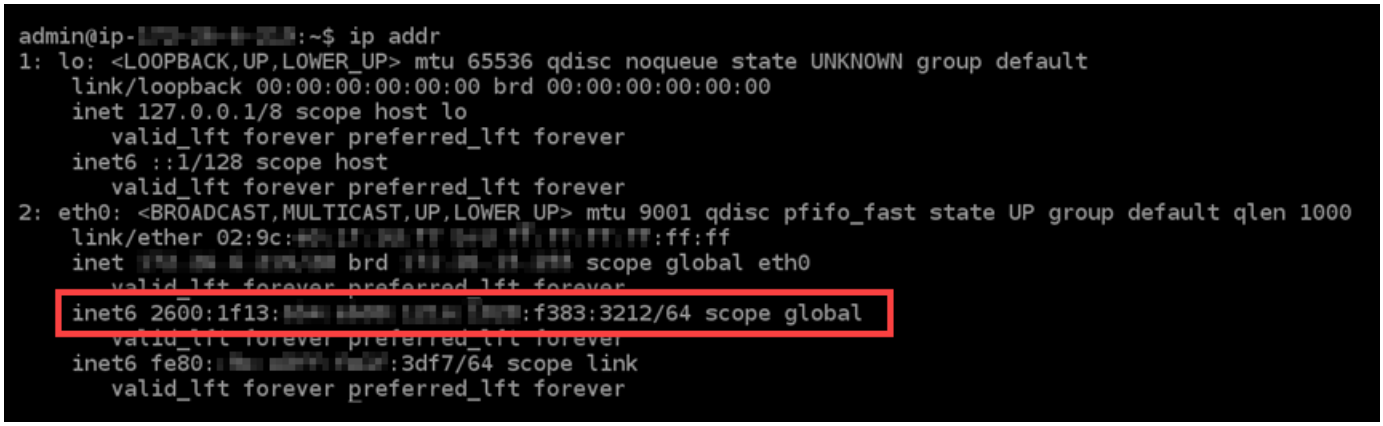
몇 분 정도 기다린 후 인스턴스가 재부팅되면 다음 단계를 수행합니다.

6. Plesk 인스턴스의 SSH 세션으로 돌아갑니다.

7. 다음 명령을 입력하여 인스턴스에 구성된 IP 주소를 살펴보고, 할당된 IPv6 주소를 인식하는지 확인합니다.

```
ip addr
```

다음 예와 비슷한 응답이 나타나는 것을 볼 수 있습니다. 인스턴스가 IPv6 주소를 인식하는 경우 이 예와 같이 `scope global` 레이블과 함께 응답에 나열됩니다.



단계별 지침에 따라 Lightsail 인스턴스 블루프린트에서 IPv6를 구성하는 방법을 알아봅니다.

이 가이드에서는 cPanel, GitLab, Nginx 및 Plesk을 비롯한 다양한 인스턴스 블루프린트를 다룹니다. 이 절차에는 SSH를 통해 인스턴스에 연결하고, 네트워크 구성 파일을 수정하고, 서비스를 다시 시작하고, 인스턴스가 할당된 IPv6 주소를 인식하는지 확인하는 작업이 포함됩니다. 이 안내서에 따라 IPv4 및 IPv6 주소를 모두 활용하도록 Lightsail 인스턴스가 올바르게 구성되어 있는지 확인하여 연결을 개선하고 인터넷의 미래를 대비할 수 있습니다.

Lightsail 작업을 AWS CLI 위한 설정 및 구성

AWS Command Line Interface (AWS CLI)는 고급 사용자와 개발자가 터미널(Linux 및 Unix) 또는 명령 프롬프트(Windows)에 명령을 입력하여 Amazon Lightsail 서비스를 제어할 수 있는 도구입니다. Lightsail 콘솔, 그래픽 사용자 인터페이스, Lightsail 애플리케이션 프로그램 인터페이스(API)를 사용하여 Lightsail을 제어할 수도 있습니다.

> **Tip**
>
> 명령줄 도구를 다운로드하거나 설치하지 않고도 AWS CLI 명령을 실행하여 AWS CloudShell를 사용하여 Lightsail 리소스를 관리할 수도 있습니다. CloudShell은 브라우저 기반의 사전 인

증된 셸로, Lightsail 콘솔에서 직접 실행할 수 있습니다. 자세한 내용은 를 사용하여 Lightsail 리소스 관리 AWS CloudShell 단원을 참조하십시오.

주제

- 1단계: 설치 AWS CLI
- 2단계: 새 액세스 키 생성
- 3단계: 구성 AWS CLI
- 다음 단계

1단계: 설치 AWS CLI

로컬 데스크톱 AWS CLI 에를 설치하거나 Lightsail 인스턴스에 설치할 수 있습니다. 에 대한 자세한 내용은 AWS Command Line Interface 사용 설명서를 AWS CLI참조하세요.

- 로컬 데스크톱 AWS CLI 에를 설치하려면 AWS Command Line Interface 설명서의 설치를 AWS CLI 참조하세요.
- Ubuntu 기반 Lightsail 인스턴스 AWS CLI 에를 설치하려면 인스턴스에 연결하고를 입력합니다`sudo apt-get -y install awscli`.

> **Note**
>
> 는 Amazon Linux Lightsail 인스턴스에 이미 설치되어 있어야 AWS CLI 합니다. 어떤 이유로 다시 설치해야 할 경우에는 인스턴스에 연결하고 `sudo yum install aws-cli`를 입력합니다.

를 설치 AWS CLI한 후에는 액세스 키를 생성한 다음 이를 사용하도록를 AWS CLI 구성해야 합니다.

2단계: 새 액세스 키 생성

Lightsail API 또는 AWS Command Line Interface (AWS CLI)를 사용하려면 새 액세스 키를 생성해야 합니다. 액세스 키는 Access Key ID(액세스 키 아이디)와 Secret Access Key(보안 액세스 키)로 구성되어 있습니다. 다음 절차에 따라 키를 생성합니다.

1. IAM 콘솔에 로그인합니다.

2. 액세스 키를 생성하려는 사용자 이름을 선택합니다. 선택한 사용자는 Lightsail 작업에 대한 전체 액세스 권한 또는 특정 액세스 권한이 있어야 합니다.
3. 보안 자격 증명(Security credentials) 탭을 선택합니다.
4. 페이지의 액세스 키(Access keys) 섹션에서 액세스 키 생성(Create access key)을 선택합니다.

> **Note**
>
> 사용자당 1번에 최대 2개의 액세스 키(활성 또는 비활성)를 보유할 수 있습니다. 2개의 액세스 키가 이미 있는 경우, 키를 새로 생성하기 전에 해당 키 중 하나를 삭제해야 합니다. 액세스 키를 삭제하기 전에 액세스 키가 현재 사용되고 있지 않은지 확인합니다.

5. 액세스 키 ID 및 비밀 액세스 키를 기록해 둡니다. 비밀 액세스 키를 보려면 비밀 액세스 키(Secret access key) 열 아래의 표시(Show)를 선택합니다.

   이 화면에서 키를 복사하거나 키 파일 다운로드를 선택하여 액세스 키 ID와 시크릿 액세스 키가 들어 있는 `.csv` 파일을 다운로드할 수 있습니다.

> **Important**
>
> 액세스 키를 안전한 위치에 저장하세요. 나중에 찾느라 애먹지 않도록 `MyLightsailKeys.csv`와 같은 이름으로 파일 이름을 지정하세요. IAM 콘솔에서 CSV 파일을 다운로드한 경우, 다음 단계를 완료한 후 해당 파일을 삭제해야 합니다. 필요한 경우 나중에 새 액세스 키를 생성할 수 있습니다.

3단계: 구성 AWS CLI

액세스 키를 AWS CLI 사용할 수 있도록 액세스 키를 사용하도록를 구성해야 합니다.

1. 터미널 창 또는 명령 프롬프트를 엽니다.
2. `aws configure`를 입력합니다.
3. 이전 단계에서 생성한 `.csv` 파일에서 AWS 액세스 키 ID를 붙여넣습니다.
4. 메시지가 표시되면 AWS 시크릿 액세스 키를 붙여넣습니다.
5. 리소스가 AWS 리전 있는를 입력합니다. 예를 들어, 리소스가 주로 오하이오에 있는 경우 기본 리전 이름을 묻는 메시지가 표시되면 `us-east-2`를 선택합니다.

   옵션 사용에 AWS CLI `--region` 대한 자세한 내용은 AWS CLI 참조의 일반 옵션을 참조하세요.

6. Default output format(기본 출력 형식)을 선택합니다(예: `json`).

이제를 사용하여 Lightsail 프로그래밍 방식으로와 상호 작용할 수 있습니다 AWS CLI. AWS CLI 명령 레퍼런스에서 Amazon Lightsail 명령을 찾아볼 수 있습니다.

다음 단계

다음 리소스는 언어별 AWS SDK를 설치하고 Lightsail API에 익숙해지는 데 도움이 될 수 있습니다.

- 언어별 AWS SDKs 설치
- Lightsail API 레퍼런스 검토

를 사용하여 Lightsail 리소스 관리 AWS CloudShell

AWS CloudShell 는 브라우저 기반의 사전 인증된 셸로, Amazon Lightsail 콘솔에서 직접 시작할 수 있습니다. CloudShell을 사용하여 명령줄 인터페이스에서 Lightsail 리소스를 관리할 수 있습니다. Bash, PowerShell 또는 Z 쉘과 같은 원하는 쉘을 사용하여 AWS Command Line Interface (AWS CLI) 명령을 실행할 수 있습니다. 명령줄 도구를 다운로드하거나 설치할 필요 없이 이 작업을 수행할 수 있습니다. 자세한 내용은 란 무엇입니까? AWS CloudShell를 참조하십시오.

CloudShell을 시작하면 Amazon Linux 2를 기반으로 하는 컴퓨팅 환경이 생성됩니다. 이 환경 내에서는 AWS CLI와 같이 사전 설치된 광범위한 개발 도구에 액세스할 수 있습니다. 사전 설치된 도구의 전체 목록은 CloudShell 사용 설명서의 사전 설치된 소프트웨어를 참조하세요.

영구 스토리지

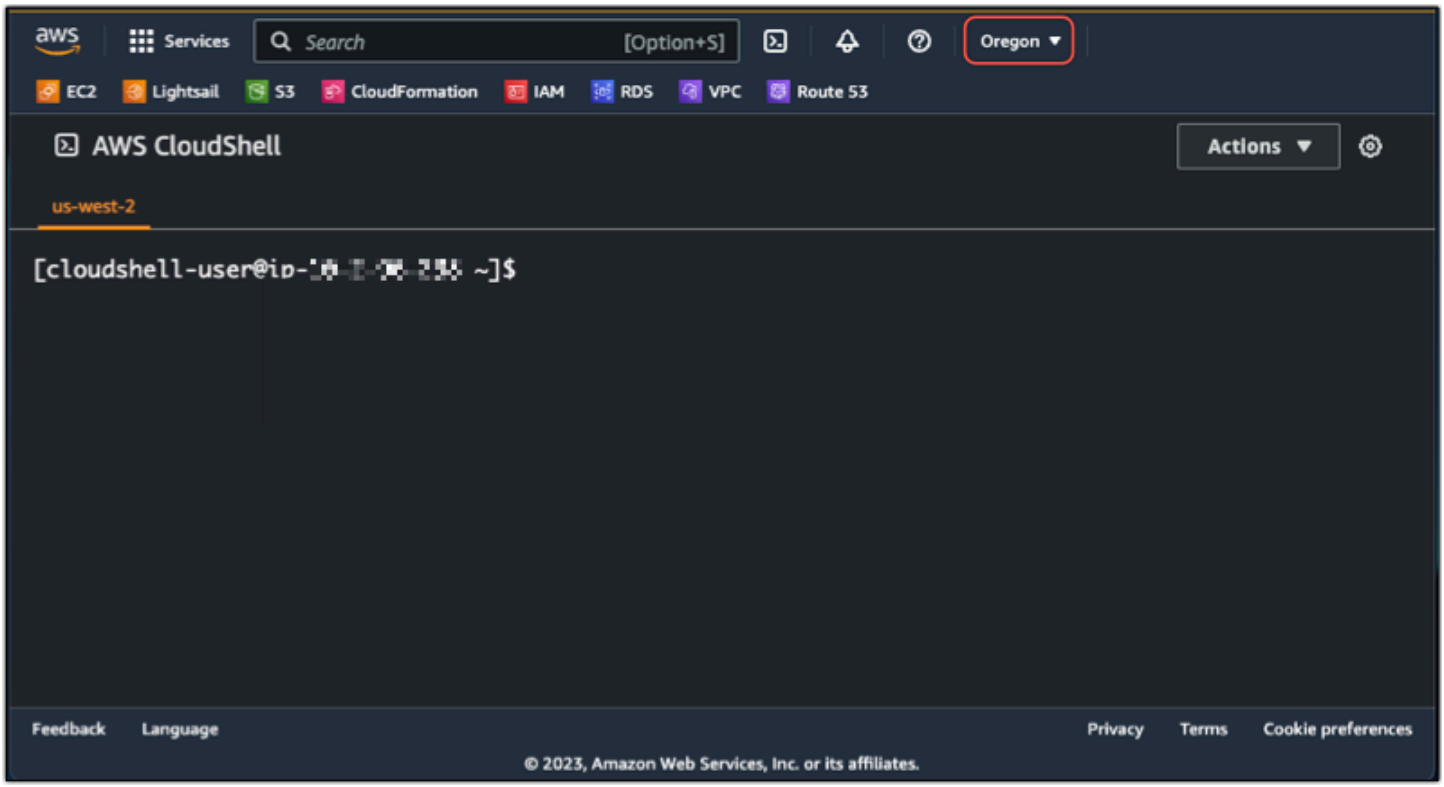
AWS CloudShell를 사용하면 추가 비용 없이 각각 최대 1GB의 영구 스토리지 AWS 리전 를 사용할 수 있습니다. 영구 스토리지는 홈 디렉터리($HOME)에 있으며 사용자만 이용할 수 있습니다. 각 쉘 세션이 종료된 후 삭제되는 임시 환경 리소스와 달리 홈 디렉터리의 데이터는 세션 간에 유지됩니다.

AWS CloudShell 에서 사용을 중지하면 AWS 리전데이터는 마지막 세션 종료 후 120일 동안 해당 리전의 영구 스토리지에 보관됩니다. 120일 경과 후 조치를 취하지 않으면, 해당 리전의 영구 스토리지에서 데이터가 자동으로 삭제됩니다. AWS 리전에서 AWS CloudShell 을(를) 다시 시작하면 삭제를 방지할 수 있습니다. 영구 스토리지의 데이터 보존에 대한 자세한 내용은 CloudShell 사용 설명서의 영구 스토리지를 참조하세요.

AWS 리전

에서는 물리적 위치에 대한 지연 시간을 최소화 AWS 리전 하는 Lightsail CloudShell 세션이에서 열립니다. 즉,는 세션 간에 변경될 AWS 리전 수 있습니다. 1GB 영구 스토리지 AWS 리전를 사용할 수 있도록에 있는 CloudShell 세션을 기록해 둡니다. 세션의 AWS 리전을 변경하려면 새 브라우저 탭에서 열기 아이콘을 선택합니다. 그러면 새 브라우저 창에서 CloudShell 세션에 액세스할 수 있는 옵션이 제공됩니다.

새 브라우저 탭의 탐색 모음에서 현재 표시된 AWS 리전 의 이름을 선택합니다. 그런 다음 전환할 AWS 리전 를 선택합니다.



CloudShell에 대한 자세한 내용은 CloudShell 사용 설명서를 참조하세요.

시작 및 사용 AWS CloudShell

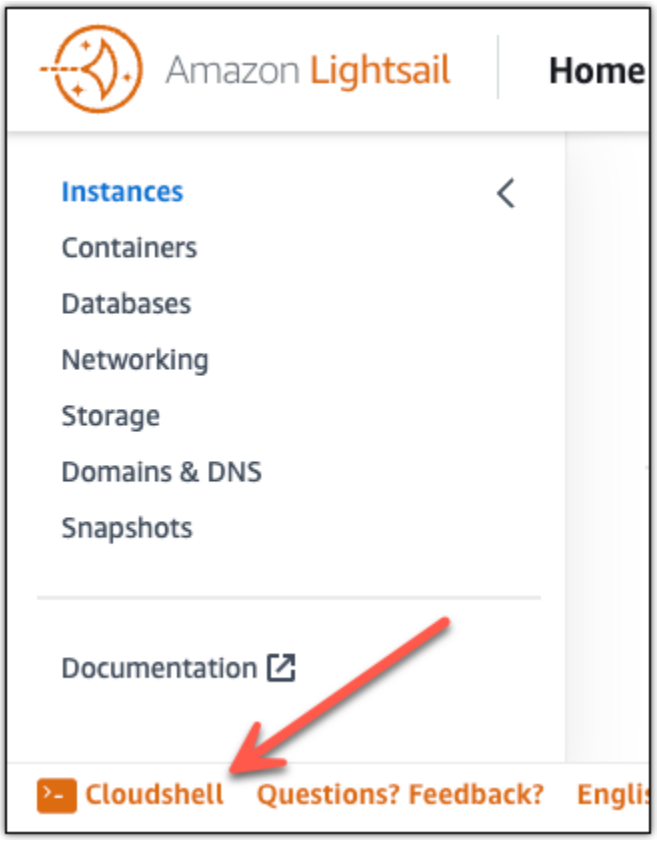
내에서 AWS CloudShell 세션을 시작하고 사용하는 방법을 알아봅니다Lightsail. CloudShell을 실행할 권한이 없는 경우 사용 중인 AWS Identity and Access Management (IAM) 자격 증명에 `arn:aws:iam::aws:policy/AWSCloudShellFullAccess` 정책을 추가해야 합니다. `arn:aws:iam::aws:policy/AdministratorAccess` 정책이 이미 연결되어 있으면 CloudShell에 액세스할 수 있습니다. 자세한 내용은 ??? 단원을 참조하십시오.

시작 AWS CloudShell

Amazon Lightsail 콘솔에서 CloudShell을 시작할 수 있습니다. 세션이 시작된 후 `Bash`, `PowerShell`, `Z shell` 등의 선호하는 쉘로 전환할 수 있습니다.

다음 단계를 완료하여에서 새 AWS CloudShell 세션을 시작합니다. Lightsail

1. https://lightsail.aws.amazon.com/ Lightsail 콘솔에 로그인합니다.
2. 콘솔 왼쪽 하단의 콘솔 도구 모음에서 CloudShell을 선택합니다. 명령 프롬프트가 표시되면 셸이 상호 작용할 준비가 된 것입니다.



3. (선택 사항) 사전 설치된 쉘을 선택하여 작업하려면 명령줄 프롬프트에 다음 프로그램 이름 중 하나를 입력합니다.

   Bash: **bash**

   Bash로 전환하면 명령 프롬프트의 기호가 $로 업데이트됩니다. Bash는의 기본 셸입니다 AWS CloudShell.

   PowerShell: **pwsh**

   PowerShell로 전환하면 명령 프롬프트의 기호가 PS>로 업데이트됩니다.

   Z 쉘: **zsh**

   Z 쉘로 전환하면 명령 프롬프트의 기호가 %로 업데이트됩니다.

Example의 Lightsail API 명령 예제 AWS CloudShell

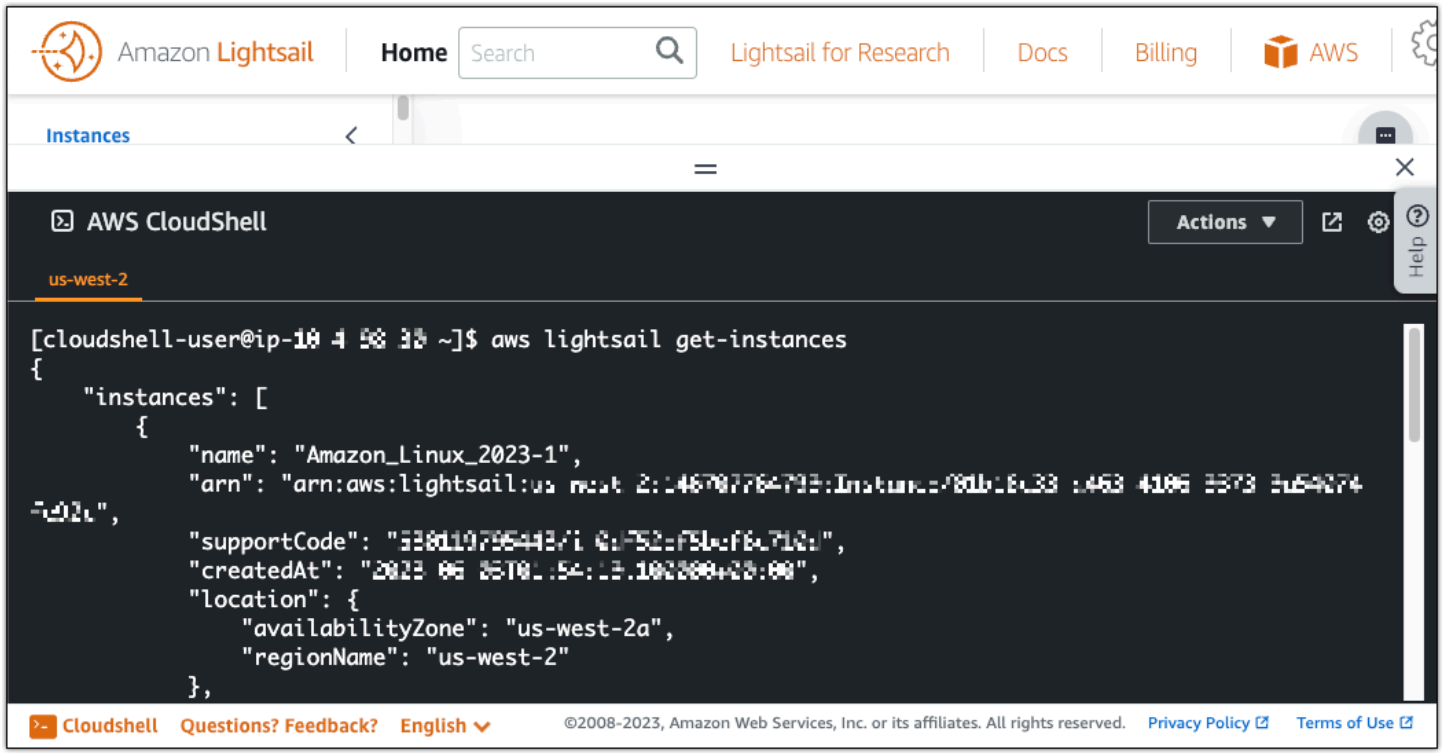
CloudShell 세션에는 사용할 수 있는 여러 명령줄 도구가 사전 설치되어 있습니다. 이 예에서는 Lightsail `GetInstances` API 작업을 사용하여 Lightsail 계정에 있는 인스턴스를 봅니

다. `GetInstances` API 작업에 대해 자세히 알아보려면 Amazon Lightsail API 레퍼런스의 GetInstances를 참조하세요.

1. https://lightsail.aws.amazon.com/ Lightsail 콘솔에 로그인합니다.
2. 콘솔 왼쪽 하단의 콘솔 도구 모음에서 CloudShell을 선택합니다.
3. AWS CloudShell 프롬프트 뒤에 다음 명령을 입력합니다.

```
aws lightsail get-instances
```

이제 Lightsail 계정에 있는 전체 인스턴스 목록이 표시됩니다.



추가 정보

다음에 대한 자세한 내용은 다음 설명서를 참조하세요 AWS CloudShell.

- Amazon Lightsail API 레퍼런스
- 에서 자주 묻는 질문 AWS CloudShell
- 에서 지원되는 브라우저 AWS CloudShell
- 의 문제 해결 AWS CloudShell

- AWS 서비스 에서 작업 AWS CloudShell

Lightsail LAMP 인스턴스에 PHP 애플리케이션 배포

Amazon Lightsail은 가상 프라이빗 서버만 필요한 경우에 Amazon Web Services(AWS)를 가장 쉽게 시작할 수 있는 방법입니다. Lightsail에는 프로젝트를 빠르게 시작하는 데 필요한 모든 것, 즉 가상 머신, SSD 기반 스토리지, 데이터 전송, DNS 관리, 고정 IP가 포함되어 있으며 고객은 이러한 서비스를 저렴하고 예측 가능한 요금으로 사용할 수 있습니다.

이 자습서에서는 Lightsail에서 LAMP 인스턴스를 시작하고 구성하는 방법을 보여줍니다. 여기에는 SSH를 통해 인스턴스에 연결하고, 인스턴스의 애플리케이션 암호를 가져오고, 고정 IP를 생성하여 인스턴스에 연결하고, DNS 영역을 만들고 도메인을 매핑하는 단계가 포함됩니다. 이 자습서를 마치면 Lightsail에서 인스턴스를 실행하기 위한 기본 지식을 얻게 됩니다.

목차

- 1단계: AWS에 가입
- 2단계: LAMP 인스턴스 생성
- 3단계: SSH를 통해 인스턴스에 연결하고 LAMP 인스턴스에 대한 애플리케이션 암호 가져오기
- 4단계: LAMP 인스턴스에 애플리케이션 설치
- 5단계: 고정 IP 주소 생성 및 LAMP 인스턴스에 연결
- 6단계: DNS 영역 생성 및 LAMP 인스턴스에 도메인 매핑
- 다음 단계

1단계: AWS에 가입

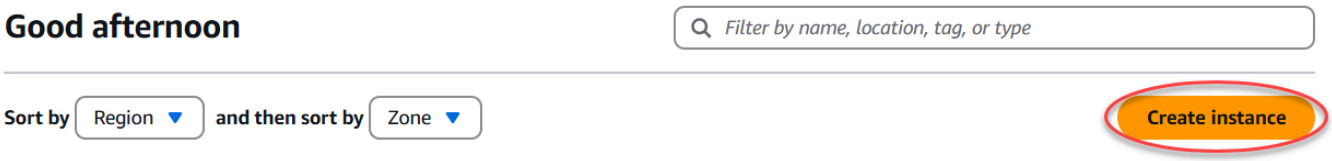
이 자습서에는 AWS 계정이 필요합니다. 계정이 이미 있는 경우 에 가입 AWS하거나 에 로그인 AWS합니다.

2단계: LAMP 인스턴스 생성

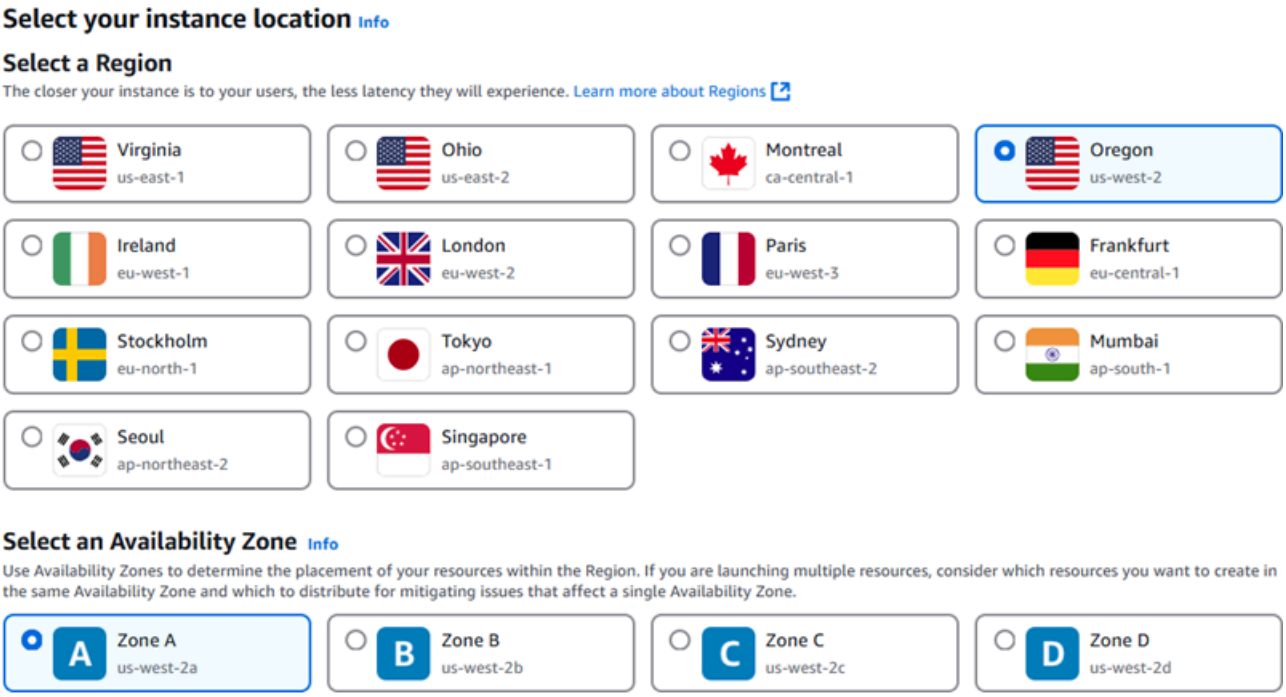
Lightsail에서 LAMP 인스턴스를 시작하고 실행합니다. Lightsail에서의 인스턴스 생성에 대한 자세한 내용은 Lightsail에서 Amazon Lightsail 인스턴스 생성 설명서를 참조하세요.

1. Lightsail 콘솔에 로그인합니다.

2. Lightsail 홈 페이지의 인스턴스(Instances) 섹션에서 인스턴스 생성(Create instance)을 선택합니다.



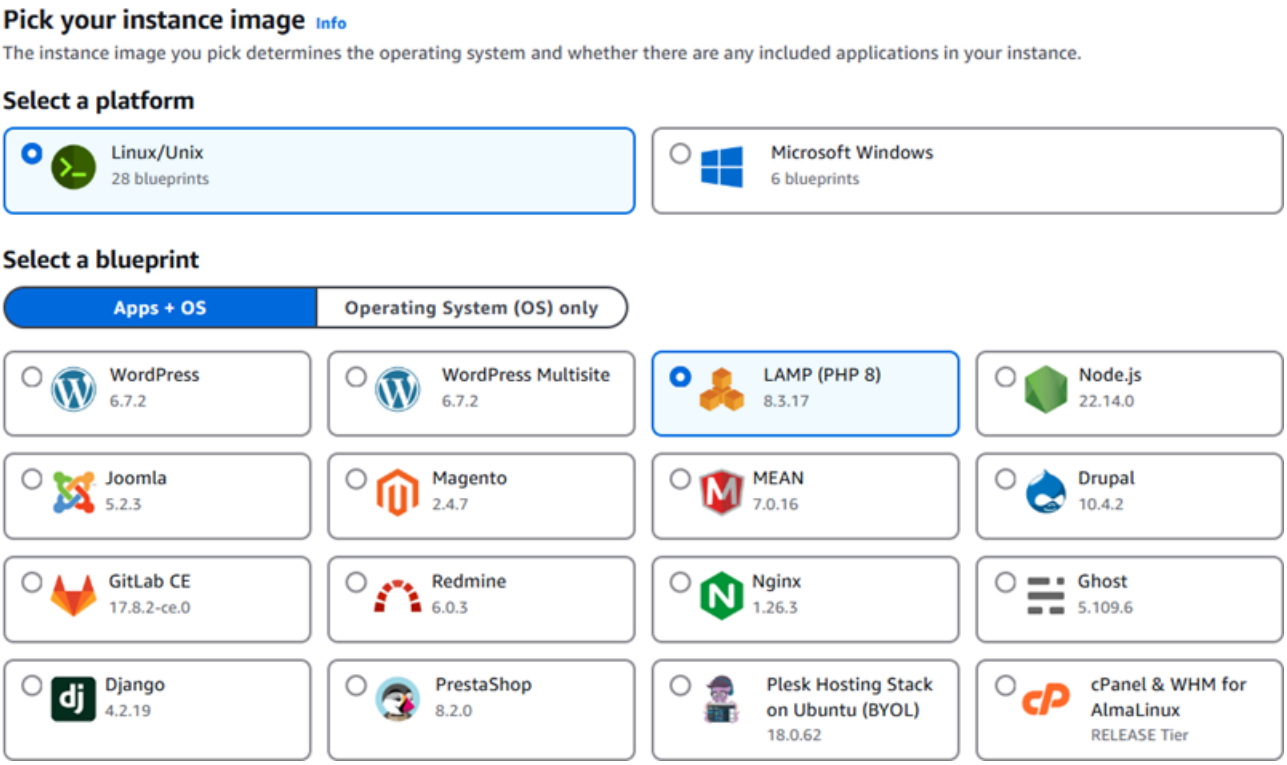
3. 인스턴스의 AWS 리전 및 가용 영역을 선택합니다.



4. 인스턴스 이미지를 선택합니다.

   a. Linux/Unix를 플랫폼으로 선택합니다.

   b. LAMP(PHP 8)를 청사진으로 선택합니다.



5. 인스턴스 플랜을 선택합니다.

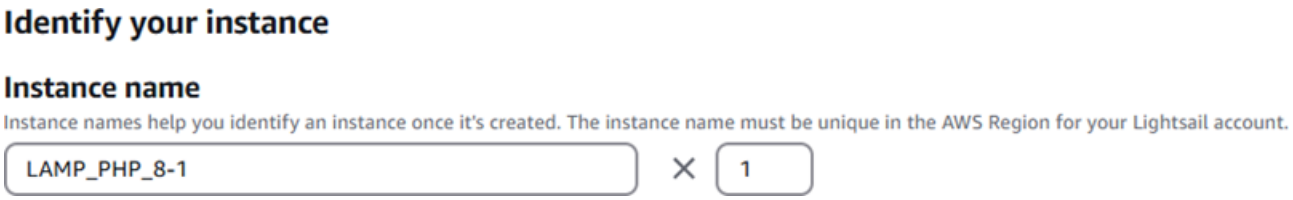
   플랜은 예측 가능하고 저렴한 비용으로 머신 구성(RAM, SSD, vCPU)과 데이터 전송 허용량을 포함합니다. 1개월(최대 750시간) 동안 무료로 5 USD Lightsail 플랜을 사용해 볼 수 있습니다.는 계정에 1개월 무료 AWS 크레딧을 제공합니다.

   > **Note**
   >
   > AWS 프리 티어의 일부로 일부 인스턴스 번들에서 Amazon Lightsail를 무료로 시작할 수 있습니다. 자세한 내용은 Amazon Lightsail 요금 페이지의 AWS 프리 티어를 참조하세요.

6. 인스턴스 이름을 입력합니다.

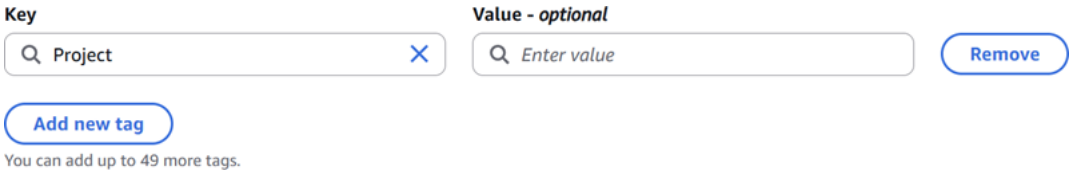
   리소스 이름:

   - Lightsail 계정 AWS 리전 의 각 내에서 고유해야 합니다.
   - 2~255자의 문자로 구성되어야 합니다.
   - 영숫자 문자 또는 숫자로 시작하고 끝나야 합니다.
   - 영숫자 문자, 숫자, 마침표, 대시, 밑줄이 포함될 수 있습니다.

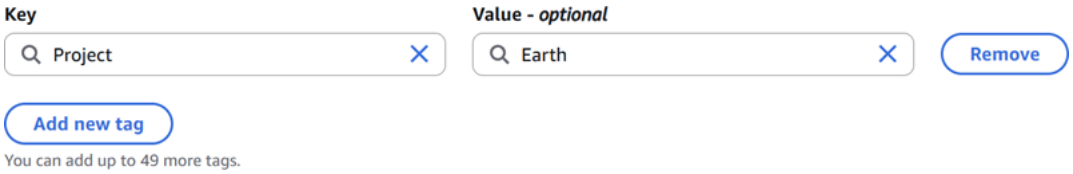


7. (선택 사항) 인스턴스에 태그를 추가하려면 새 태그 추가를 선택합니다. 필요에 따라 이 단계를 반복하여 모든 태그를 추가합니다. 태그 사용에 대한 자세한 내용은 태그를 참조하세요.

   a. 키에는 태그 키를 입력합니다.

   b. (선택 사항) 값에 태그 값을 입력합니다.

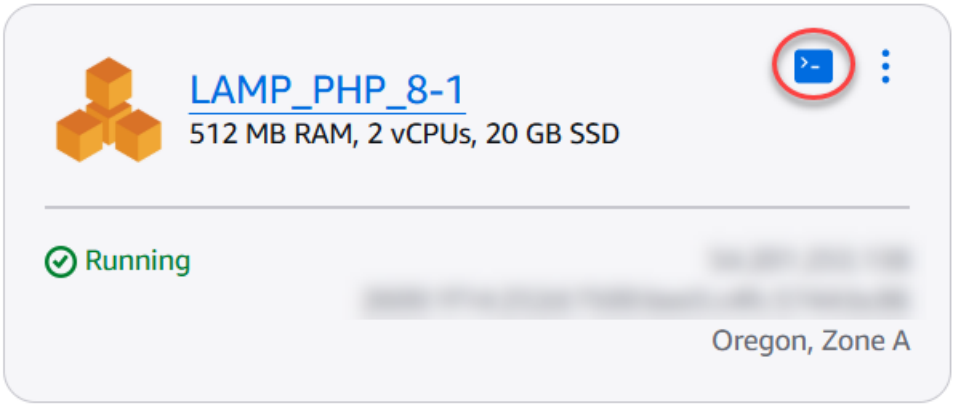
   

8. 인스턴스 생성을 선택합니다.

3단계: SSH를 통해 인스턴스에 연결하고 LAMP 인스턴스에 대한 애플리케이션 암호 가져오기

LAMP에서 데이터베이스에 로그인하기 위한 기본 암호는 인스턴스에 저장됩니다. Lightsail 콘솔에서 브라우저 기반 SSH 터미널을 사용하여 인스턴스에 연결하고 특수 명령을 실행하여 암호를 가져옵니다. 자세한 내용은 Amazon Lightsail의 Bitnami 인스턴스에 대한 애플리케이션 사용자 이름과 암호 가져오기를 참조하세요.

1. Lightsail 홈 페이지의 인스턴스 섹션에서 LAMP 인스턴스에 대한 SSH 빠른 연결 아이콘을 선택합니다.



2. 브라우저 기반 SSH 클라이언트 창이 열리면 다음 명령을 입력하여 기본 애플리케이션 암호를 검색합니다.

```
cat bitnami_application_password
```

> **Note**
>
> 사용자 홈 디렉터리가 아닌 다른 디렉터리에 있는 경우 `cat $HOME/bitnami_application_password`를 입력합니다.

3. 화면에 표시되는 암호를 기록해 둡니다. 나중에 이 암호를 사용하여 인스턴스에 Bitnami 애플리케이션을 설치하거나 사용자 이름이 `root`인 MySQL 데이터베이스에 액세스합니다.

4단계: LAMP 인스턴스에 애플리케이션 설치

LAMP 인스턴스에 PHP 애플리케이션을 배포하거나 Bitnami 애플리케이션을 설치합니다. PHP 애플리케이션을 배포하기 위한 기본 디렉터리는 `/opt/bitnami/apache2/htdocs`입니다. PHP 애플리케이션 파일을 해당 디렉터리에 복사하고 인스턴스의 퍼블릭 IP 주소로 이동하여 애플리케이션에 액세스합니다.

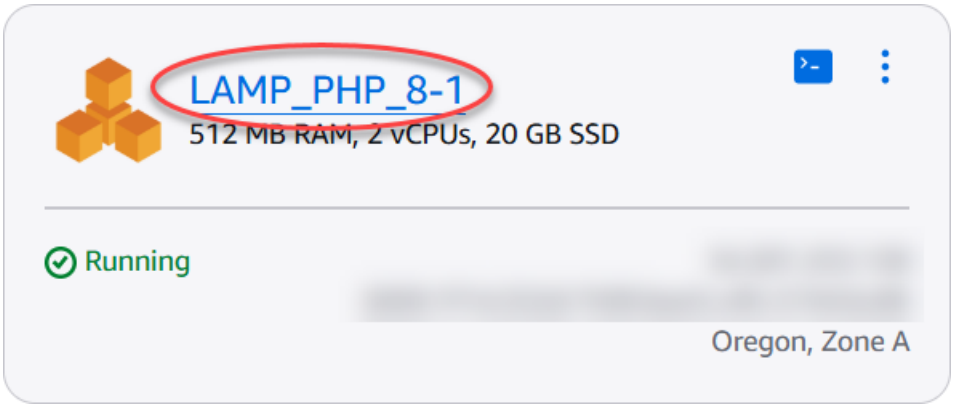
모듈 설치 관리자를 사용하여 Bitnami 애플리케이션을 설치할 수도 있습니다. Bitnami 웹사이트에서 여러 애플리케이션 중 WordPress, Drupal, Magento, Moodle을 다운로드하여 서버의 기능을 확장합니다. Bitnami 애플리케이션 설치에 대한 자세한 내용은 Bitnami 설명서의 시작하기를 참조하세요.

5단계: 고정 IP 주소 생성 및 LAMP 인스턴스에 연결

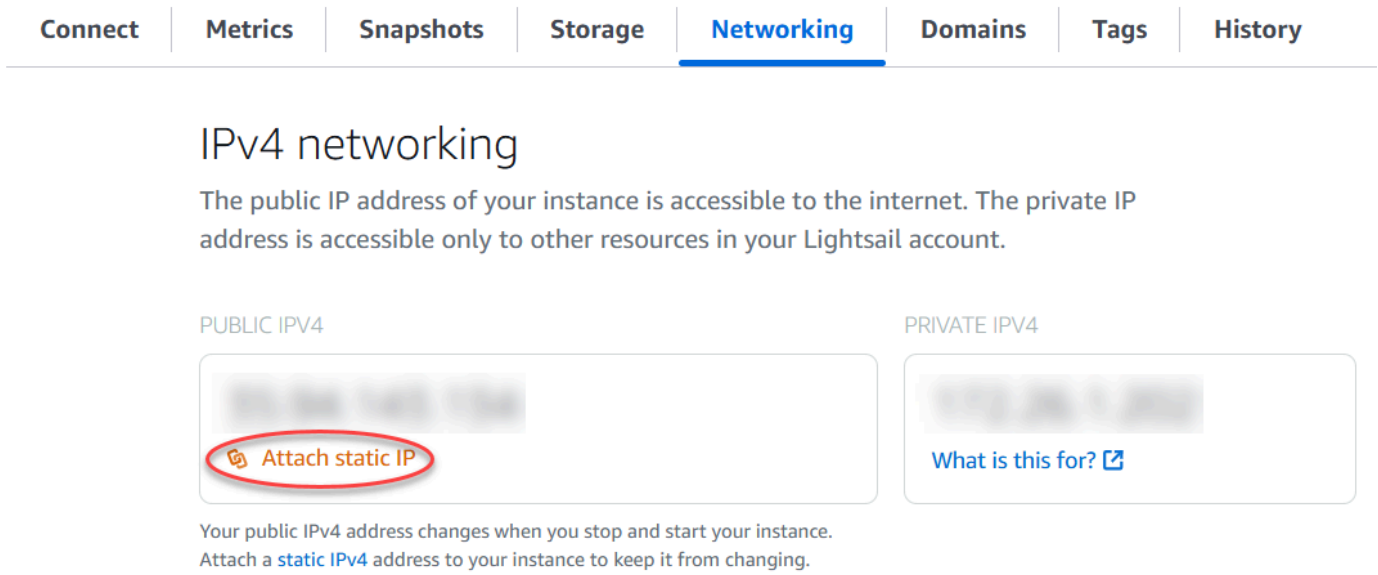
인스턴스를 중지했다가 시작하면 LAMP 인스턴스의 기본 퍼블릭 IP가 변경됩니다. 인스턴스에 연결된 고정 IP 주소는 인스턴스를 중지했다가 시작한 경우에도 동일하게 유지됩니다.

고정 IP 주소를 생성하고 LAMP 인스턴스에 연결합니다. 자세한 내용은 Lightsail 설명서의 고정 IP를 생성하여 인스턴스에 연결을 참조하세요.

1. Lightsail 홈 페이지의 인스턴스 섹션에서 실행 중인 LAMP 인스턴스를 선택합니다.



2. 네트워킹 탭을 선택한 후 고정 IP 연결을 선택합니다.



3. 고정 IP의 이름을 지정한 다음 생성 및 연결(Create and attac)을 선택합니다.

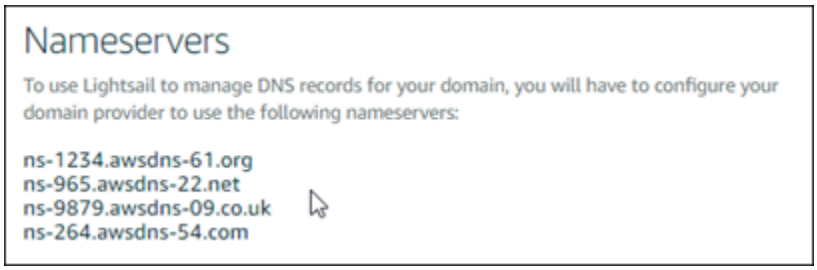
6단계: DNS 영역 생성 및 LAMP 인스턴스에 도메인 매핑

도메인의 DNS 레코드 관리를 Lightsail에 이전합니다. 이렇게 하면 도메인을 LAMP 인스턴스에 보다 쉽게 매핑하고 Lightsail 콘솔을 사용하여 웹 사이트의 모든 리소스를 관리할 수 있습니다. 자세한 내용은 DNS 영역을 생성하여 도메인의 DNS 레코드 관리를 참조하세요.

1. 홈 페이지의 도메인 및 DNSLightsail 섹션에서 DNS 영역 생성을 선택합니다.

2. 도메인을 입력하고 DNS 영역 생성을 선택합니다.
3. 페이지에 나열된 이름 서버 주소를 기록합니다.

   이 이름 서버 주소를 도메인 이름의 등록 기관에 추가하여 도메인의 DNS 레코드 관리를 Lightsail로 이전합니다.



4. 도메인의 DNS 레코드 관리가 Lightsail로 이전된 후 다음과 같이 A 레코드를 추가하여 도메인의 정점을 LAMP 인스턴스로 가리킵니다.

   a. 할당(Assignments) 탭에서 할당 추가(Add assignment)를 선택합니다.
   b. 도메인 선택(Select a domain) 필드에서 도메인 또는 하위 도메인을 선택합니다.
   c. 리소스 선택(Select a resource) 드롭다운에서 이 자습서의 앞부분에서 만든 LAMP 인스턴스를 선택합니다.
   d. 할당(Assign)을 선택합니다.

   도메인이 LAMP 인스턴스로 트래픽을 라우팅하기 전에 인터넷 DNS를 통해 변경 사항이 전파될 때까지 기다립니다.

다음 단계

다음은 Amazon Lightsail에서 LAMP 인스턴스를 실행한 후 수행할 수 있는 몇 가지 추가 단계입니다.

- Linux 또는 Unix 인스턴스의 스냅샷 생성
- 블록 스토리지 디스크를 추가로 생성하고 Linux 기반의 인스턴스에 연결

Lightsail LAMP 인스턴스를 Aurora 데이터베이스에 연결

게시물, 페이지 및 사용자에 대한 애플리케이션 데이터가 Amazon Lightsail의 LAMP 인스턴스에서 실행 중인 MariaDB 데이터베이스에 저장되어 있습니다. 인스턴스에 장애가 발생하면 데이터가 복구 불가능한 상태가 될 수 있습니다. 이러한 상황을 방지하려면 MySQL 관리형 데이터베이스로 애플리케이션 데이터를 전송해야 합니다.

Amazon Aurora는 클라우드용으로 구축된 MySQL 및 PostgreSQL 호환 관계형 데이터베이스입니다. 이는 기존 엔터프라이즈 데이터베이스의 성능 및 가용성과 오픈 소스 데이터베이스의 단순성 및 비용 효율성을 결합합니다. Aurora는 Amazon Realtional Database Service(RDS)의 일부로 제공됩니다. Amazon RDS는 클라우드에서 관계형 데이터베이스를 더 쉽게 설치, 운영 및 크기 조정할 수 있는 관리형 데이터베이스 서비스입니다. 자세한 내용은 https://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/Amazon Relational Database Service 사용 설명서와 https://docs.aws.amazon.com/AmazonRDS/latest/AuroraUserGuide/Amazon Aurora - Aurora 사용 설명서를 참조하세요.

이 자습서에서는 Lightsail의 LAMP 인스턴스의 애플리케이션 데이터베이스를 Amazon RDS의 Aurora 관리형 데이터베이스에 연결하는 방법을 보여줍니다.

목차

- 1단계: 필수 구성 요소 완성
- 2단계: Aurora 데이터베이스를 위한 보안 그룹 구성
- 3단계: Lightsail 인스턴스에서 Aurora 데이터베이스로 연결
- 4단계: LAMP 인스턴스에서 Aurora 데이터베이스로 MariaDB 데이터베이스 전송
- 5단계: Aurora 관리형 데이터베이스에 연결하도록 애플리케이션 구성

1단계: 필수 구성 요소 완성

시작하기 전에 다음 사전 조건을 완료합니다.

1. Lightsail에 LAMP 인스턴스를 생성하고 여기에 애플리케이션을 구성합니다. 계속하기 전에 인스턴스가 실행 중인 상태여야 합니다. 자세한 내용은 자습서: Lightsail에서 LAMP 인스턴스 시작 및 구성을 참조하세요.
2. Lightsail 계정에서 VPC 피어링을 켭니다. 자세한 내용은 외부 AWS 리소스로 작업하도록 Amazon VPC 피어링 설정을 참조하세요Lightsail.
3. Amazon RDS에 Aurora 관리형 데이터베이스를 생성합니다. 데이터베이스는 LAMP 인스턴스와 동일한 AWS 리전 에 있어야 합니다. 계속하기 전에 데이터베이스도 실행 중인 상태여야 합니다. 자세한 내용은 Amazon Aurora - Aurora 사용 설명서의 Amazon Aurora 시작하기를 참조하세요.

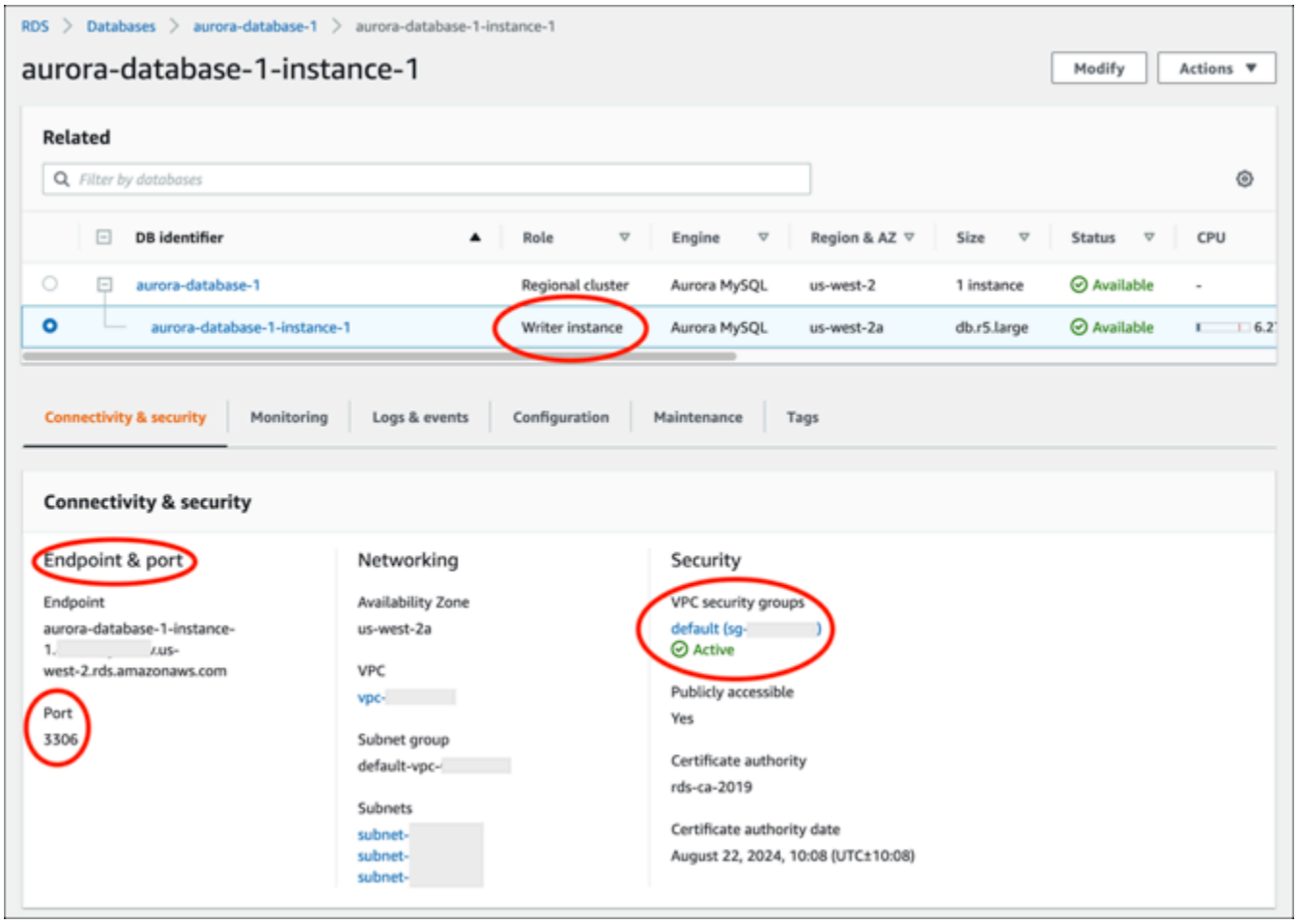
2단계: Aurora 데이터베이스를 위한 보안 그룹 구성

AWS 보안 그룹은 AWS 리소스의 가상 방화벽 역할을 합니다. 이 보안 그룹은 Amazon RDS의 Aurora 데이터베이스에 연결할 수 있는 수신 및 발신 트래픽을 제어합니다. 보안 그룹에 대한 자세한 내용은

Amazon Virtual Private Cloud 사용 설명서의 보안 그룹을 사용하여 리소스에 대한 트래픽 제어를 참조하세요.

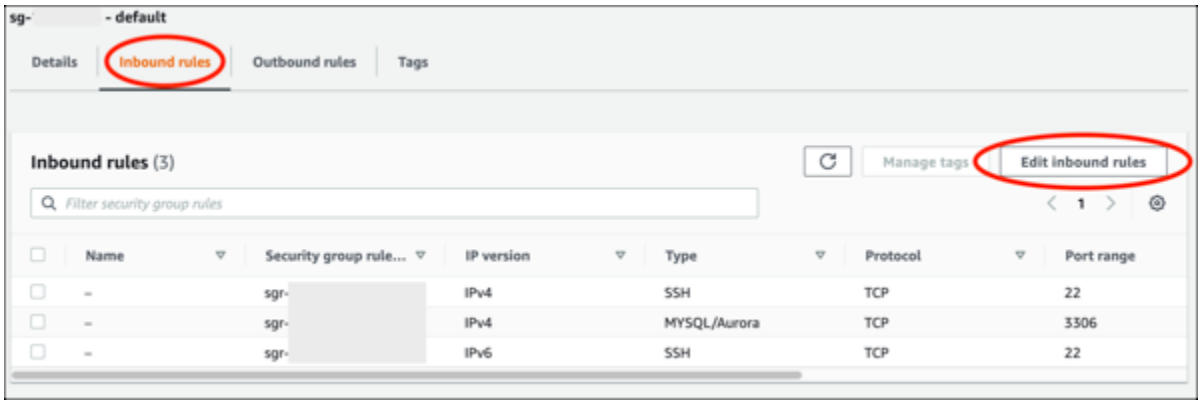
다음 절차를 완료하여 LAMP 인스턴스가 Aurora 데이터베이스에 연결을 설정할 수 있도록 보안 그룹을 구성합니다.

1. Amazon RDS 콘솔에 로그인합니다.
2. 탐색 창에서 [Databases(데이터베이스)]를 선택합니다.
3. LAMP 인스턴스가 연결할 Aurora 데이터베이스의 라이터 인스턴스를 선택합니다.
4. 연결 및 보안(Connectivity & security) 탭을 선택합니다.
5. 엔드포인트 및 포트(Endpoint & port) 섹션에서라이터 인스턴스(Writer instance)의 엔드포인트 이름(Endpoint name)과 포트(Port)를 기록해 둡니다. 이 정보는 나중에 데이터베이스에 연결하기 위해 Lightsail 인스턴스를 구성할 때 필요합니다.
6. 보안(Security) 섹션에서 활성 VPC 보안 그룹 링크를 선택합니다. 데이터베이스의 보안 그룹으로 리디렉션됩니다.



7. Aurora 데이터베이스에 대한 보안 그룹이 선택되어 있는지 확인합니다.

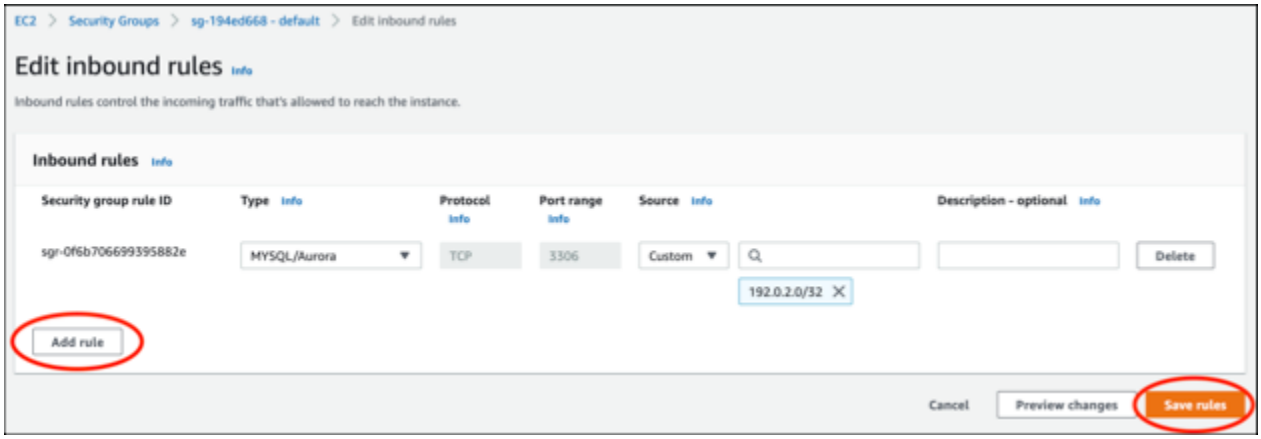
8. 인바운드 규칙 탭을 선택합니다.
9. 인바운드 규칙 편집을 선택합니다.



10. 인바운드 규칙 편집(Edit inbound rules) 페이지에서 규칙 추가(Add rule)를 선택합니다.
11. 다음 단계 중 하나를 완료합니다.

    - 기본 MySQL 포트 3306을 사용하는 경우, 유형(Type) 드롭다운 메뉴에서 MySQL/Aurora를 선택합니다.
    - 데이터베이스에 사용자 지정 포트를 사용하는 경우, 유형(Type) 드롭다운 메뉴에서 사용자 지정 TCP(Custom TCP)를 선택하고 포트 범위(Port Range)텍스트 상자에 포트 번호를 입력합니다.

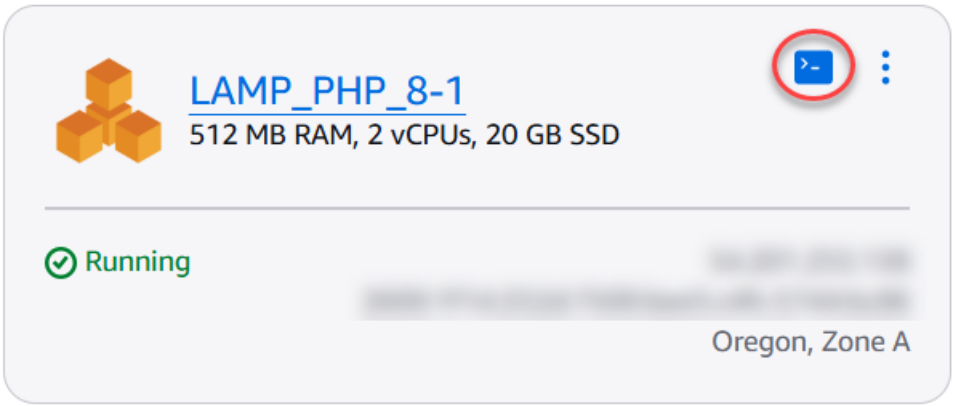
12. 소스(Source) 텍스트 상자에 LAMP 인스턴스의 프라이빗 IP 주소를 추가합니다. IP 주소는 CIDR 표기법으로 입력해야 합니다. 즉, `/32`를 추가해야 합니다. 예를 들어, `192.0.2.0`을 허용하려면 `192.0.2.0/32`를 입력합니다.
13. 규칙 저장을 선택합니다.



3단계: Lightsail 인스턴스에서 Aurora 데이터베이스로 연결

다음 절차를 완료하여 Lightsail 인스턴스에서 Aurora 데이터베이스에 연결할 수 있는지 확인합니다.

1. Lightsail 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 인스턴스를 선택합니다.
3. LAMP 인스턴스가 SSH를 사용하여 연결할 브라우저 기반 SSH 클라이언트 아이콘을 선택합니다.



4. 인스턴스에 연결한 후 다음 명령을 입력하여 Aurora 데이터베이스에 연결합니다. 명령에서 *DatabaseEndpoint*는 Aurora 데이터베이스의 엔드포인트 주소로, *Port*는 데이터베이스의 포트로 바꿉니다. *MyUserName*은 데이터베이스를 생성할 때 입력했던 사용자 이름으로 바꿉니다.

```
mysql -h DatabaseEndpoint -P Port -u MyUserName -p
```

인스턴스가 Aurora 데이터베이스에 액세스 및 연결할 수 있음을 확인해 주는 다음 예와 유사한 응답이 표시되어야 합니다.

```
bitnami@ip-            $ mysql -h database.cluster-            .us-west-2.rds.amazonaws.
com -P 3306 -u admin -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MySQL connection id is 215
Server version: 5.6.10 MySQL Community Server (GPL)

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MySQL [(none)]>
```

이 응답이 표시되지 않거나 오류 메시지가 나타나면 Lightsail 인스턴스의 프라이빗 IP 주소가 연결할 수 있도록 데이터베이스의 보안 그룹을 구성해야 할 수 있습니다. 자세한 내용은 이 설명서의 Aurora 데이터베이스에 대한 보안 그룹 구성을 참조하세요.

4단계: LAMP 인스턴스에서 Aurora 데이터베이스로 MariaDB 데이터베이스 전송

인스턴스에서 데이터베이스에 연결할 수 있다는 것을 확인했으므로 이제 LAMP 인스턴스 데이터베이스에서 Aurora 데이터베이스로 데이터로 마이그레이션해야 합니다. 자세한 내용은 Amazon Aurora - Aurora 사용 설명서의 Amazon Aurora MySQL DB 클러스터로 데이터 마이그레이션을 참조하세요.

5단계: Aurora 관리형 데이터베이스에 연결하도록 애플리케이션 구성

애플리케이션 데이터를 Aurora 데이터베이스로 전송한 후 Aurora 데이터베이스에 연결하기 위해 LAMP 인스턴스에서 실행 중인 애플리케이션을 구성해야 합니다. SSH를 사용하여 LAMP 인스턴스에 연결하고 애플리케이션의 데이터베이스 구성 파일에 액세스합니다. 구성 파일에서 Aurora 데이터베이스의 엔드포인트 주소, 데이터베이스 사용자 이름 및 암호를 정의합니다. 다음은 구성 파일의 예입니다.

```
bitnami@ip-            :~/htdocs$ cat connectvalues.php
<?php
$host           = 'database.cluster-            .us-west-2.rds.amazonaws.com';
$username       = 'admin';
$password       = 'Password1';
```

Lightsail에서 Windows Server 2016 인스턴스 시작 및 구성

Amazon Lightsail은 가상 프라이빗 서버만 필요한 경우에 Amazon Web Services(AWS)를 가장 쉽게 시작할 수 있는 방법입니다. Lightsail에는 프로젝트를 빠르게 시작하는 데 필요한 모든 것, 즉 가상 머신, SSD 기반 스토리지, 데이터 전송, DNS 관리, 고정 IP가 포함되어 있으며 고객은 이러한 서비스를 저렴하고 예측 가능한 요금으로 사용할 수 있습니다.

이 자습서에서는 Lightsail에서 Windows Server 2016 인스턴스를 시작하고 구성하는 방법을 보여줍니다. 여기에는 RDP를 통해 인스턴스에 연결하고, 고정 IP를 생성하여 인스턴스에 연결하고, DNS 영역을 만들고 도메인을 매핑하는 단계가 포함됩니다. 이 자습서를 마치면 Lightsail에서 인스턴스를 실행하기 위한 기본 지식을 얻게 됩니다.

목차

- 1단계: AWS에 가입
- 2단계: Windows Server 2016 인스턴스 생성
- 3단계: RDP로 Windows Server 2016 인스턴스에 연결
- 4단계: 고정 IP 주소 생성 및 Windows Server 2016 인스턴스 연결
- 5단계: DNS 영역 생성 및 Windows Server 2016 인스턴스에 도메인 매핑

- 다음 단계

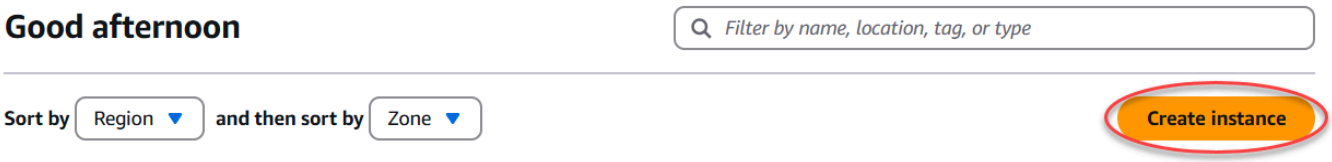
1단계: AWS에 가입

이 자습서에는 AWS 계정이 필요합니다. 이미 계정이 있는 경우 에 가입 AWS하거나 에 로그인 AWS합니다.

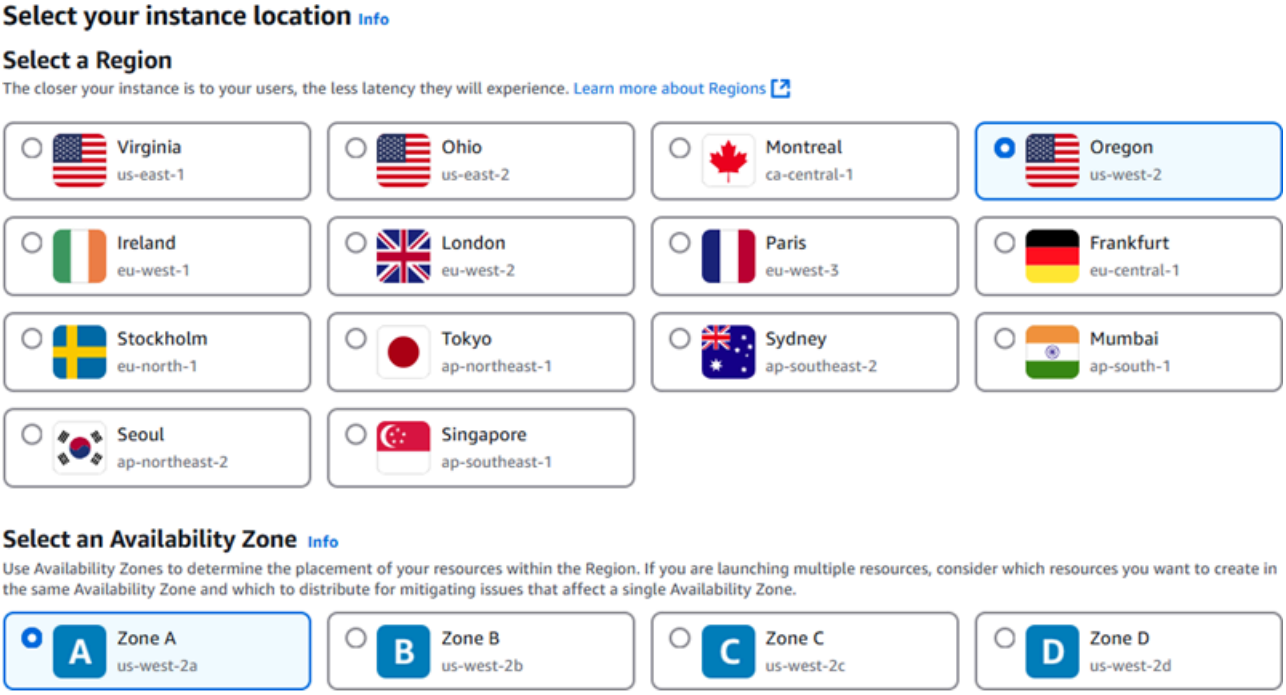
2단계: Lightsail에서 Windows Server 2016 인스턴스 만들기

Lightsail에서 Windows Server 2016 인스턴스를 시작하고 실행합니다. 자세한 내용은 Windows Server 기반 인스턴스 시작하기를 참조하세요.

1. Lightsail 콘솔에 로그인합니다.
2. Lightsail 홈 페이지의 인스턴스(Instances) 섹션에서 인스턴스 생성(Create instance)을 선택합니다.



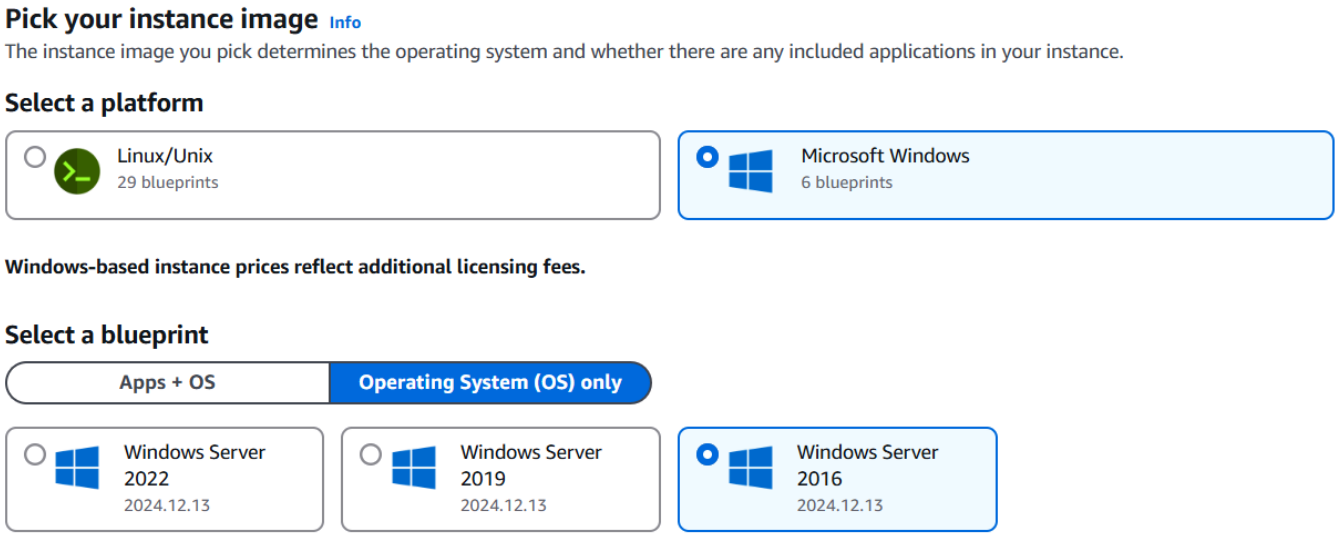
3. 인스턴스의 AWS 리전 및 가용 영역을 선택합니다.



4. 인스턴스 이미지를 선택합니다.

a. Microsoft Windows를 플랫폼으로 선택합니다.

b. OS 전용을 선택한 다음 Windows Server 2016을 블루프린트로 선택합니다.



5. 인스턴스 플랜을 선택합니다.

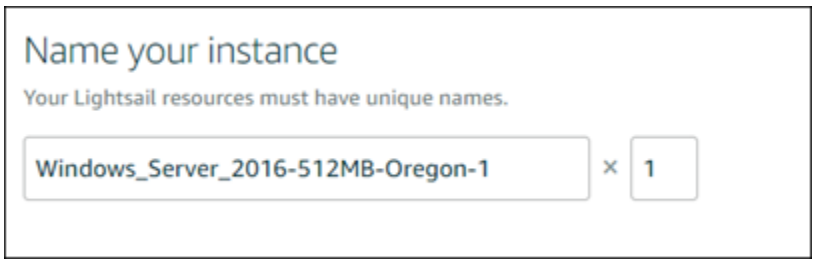
   플랜은 예측 가능하고 저렴한 비용으로 머신 구성(RAM, SSD, vCPU)과 데이터 전송 허용량을 포함합니다. 1개월(최대 750시간) 동안 무료로 9.50 USD Lightsail 플랜을 사용해 볼 수 있습니다.는 계정에 1개월 무료 AWS 크레딧을 제공합니다.

   > **Note**
   >
   > AWS 프리 티어의 일부로 일부 인스턴스 번들에서 Amazon Lightsail를 무료로 시작할 수 있습니다. 자세한 내용은 Amazon Lightsail 요금 페이지의 AWS 프리 티어를 참조하세요.

6. 인스턴스 이름을 입력합니다.

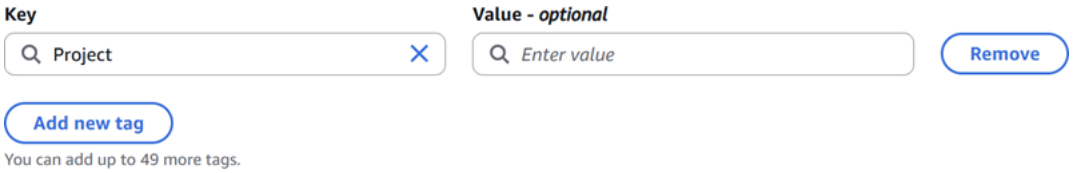
   리소스 이름:

   - Lightsail 계정 AWS 리전 의 각 내에서 고유해야 합니다.
   - 2~255자의 문자로 구성되어야 합니다.
   - 영숫자 문자 또는 숫자로 시작하고 끝나야 합니다.
   - 영숫자 문자, 숫자, 마침표, 대시, 밑줄이 포함될 수 있습니다.

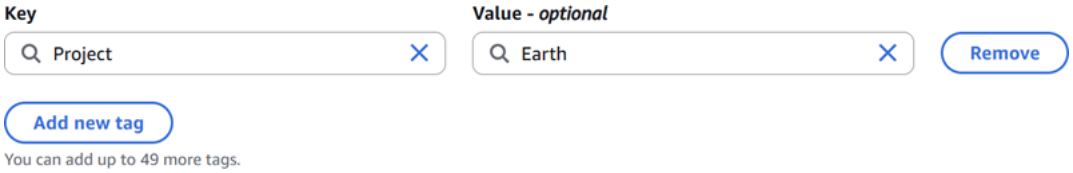


7. (선택 사항) 인스턴스에 태그를 추가하려면 새 태그 추가를 선택합니다. 필요에 따라 이 단계를 반복하여 모든 태그를 추가합니다. 태그 사용에 대한 자세한 내용은 태그를 참조하세요.

   a. 키에는 태그 키를 입력합니다.

   b. (선택 사항) 값에 태그 값을 입력합니다.

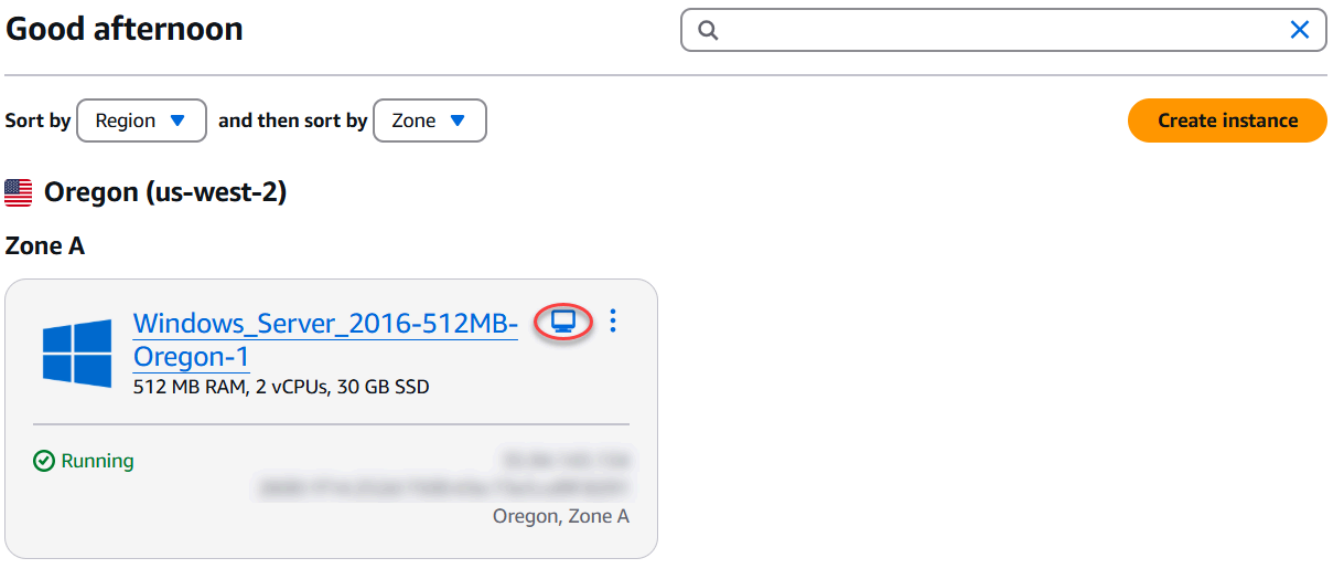
   

8. 인스턴스 생성을 선택합니다.

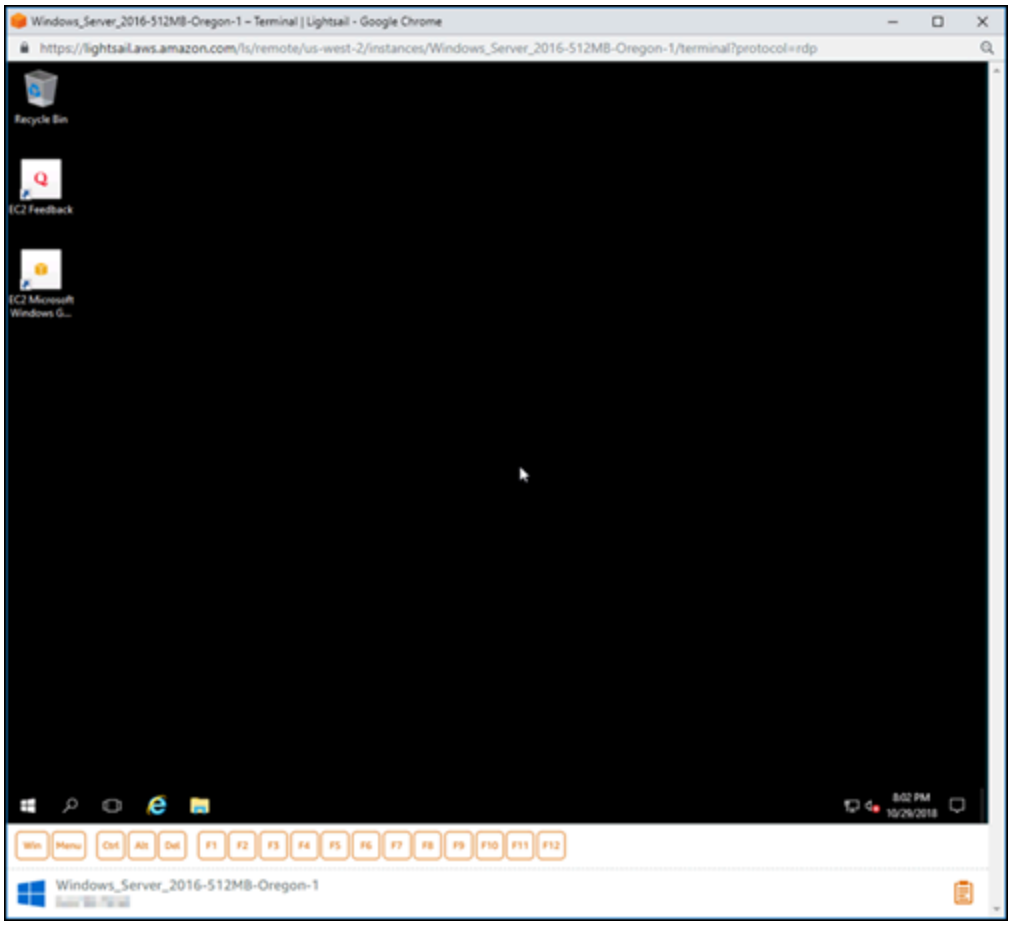
3단계: RDP로 Windows Server 2016 인스턴스에 연결

Lightsail 콘솔에서 브라우저 기반 RDP 클라이언트를 사용하여 Windows Server 2016 인스턴스에 연결합니다. 자세한 내용은 Windows 인스턴스에 연결을 참조하세요.

1. Lightsail 홈 페이지의 인스턴스 섹션에서 Windows Server 2016 인스턴스의 RDP 빠른 연결 아이콘을 선택합니다.



2. 브라우저 기반 RDP 클라이언트 창이 열리면 Windows Server 2016 인스턴스를 구성할 수 있습니다.

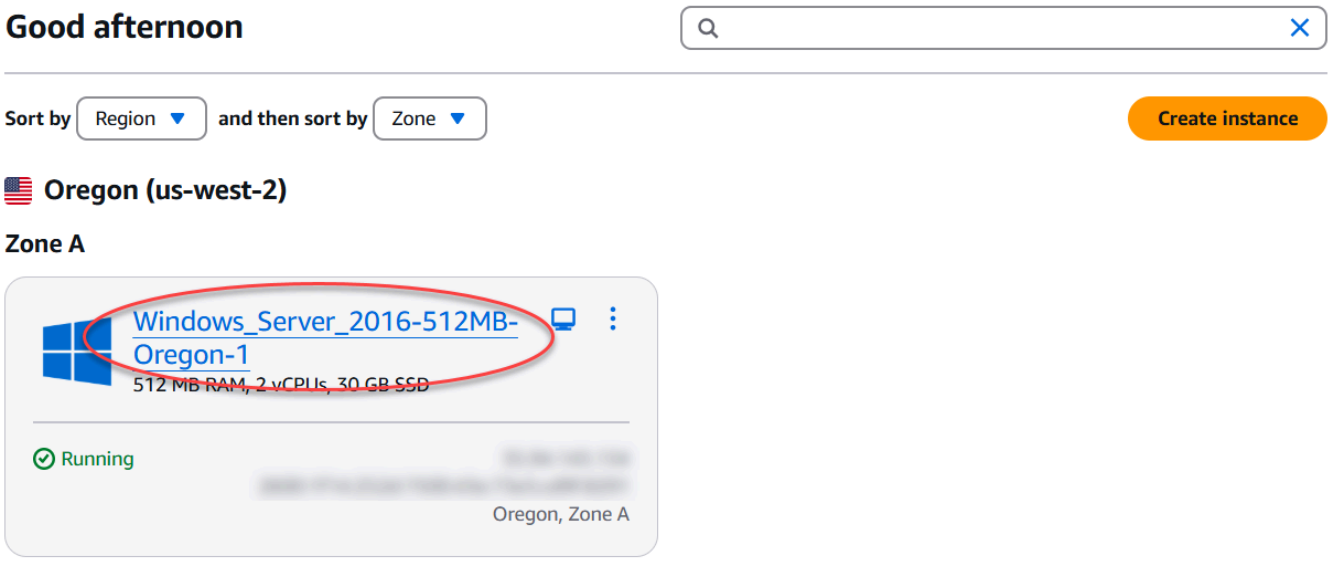


4단계: 고정 IP 주소 생성 및 Windows Server 2016 인스턴스 연결

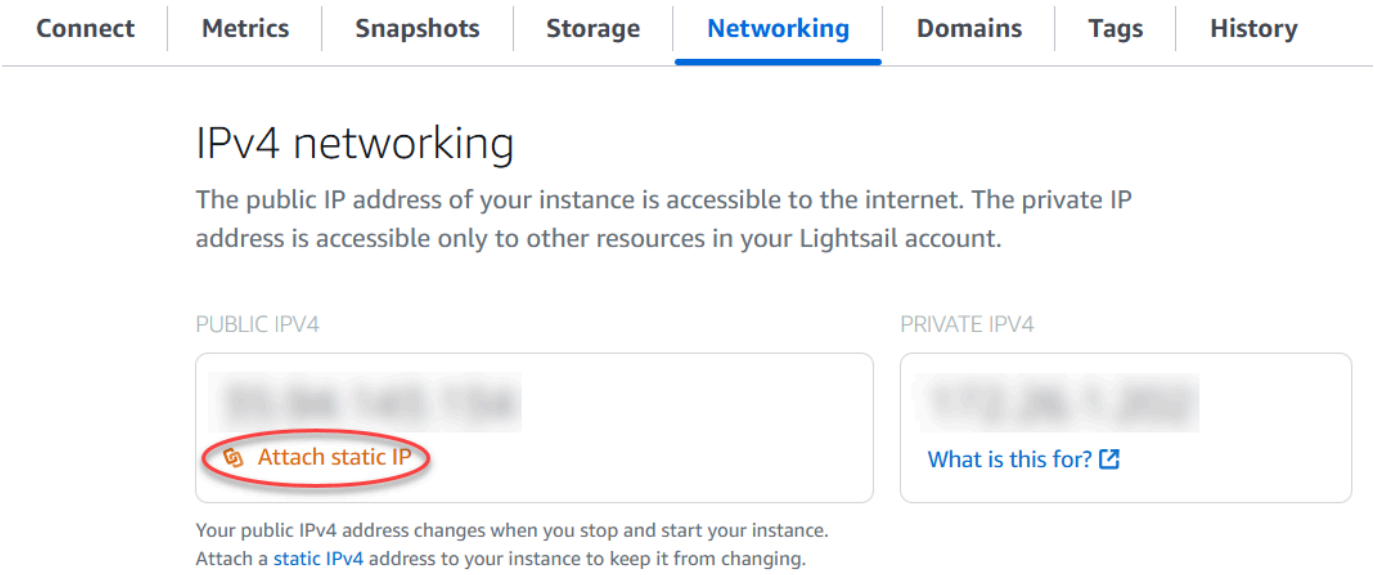
인스턴스를 중지했다가 시작하면 Windows Server 2016 인스턴스의 기본 퍼블릭 IP가 변경됩니다. 인스턴스에 연결된 고정 IP 주소는 인스턴스를 중지했다가 시작한 경우에도 동일하게 유지됩니다.

고정 IP 주소를 생성하고 Windows Server 2016 인스턴스에 연결합니다. 자세한 내용은 Lightsail 설명서의 고정 IP를 생성하여 인스턴스에 연결을 참조하세요.

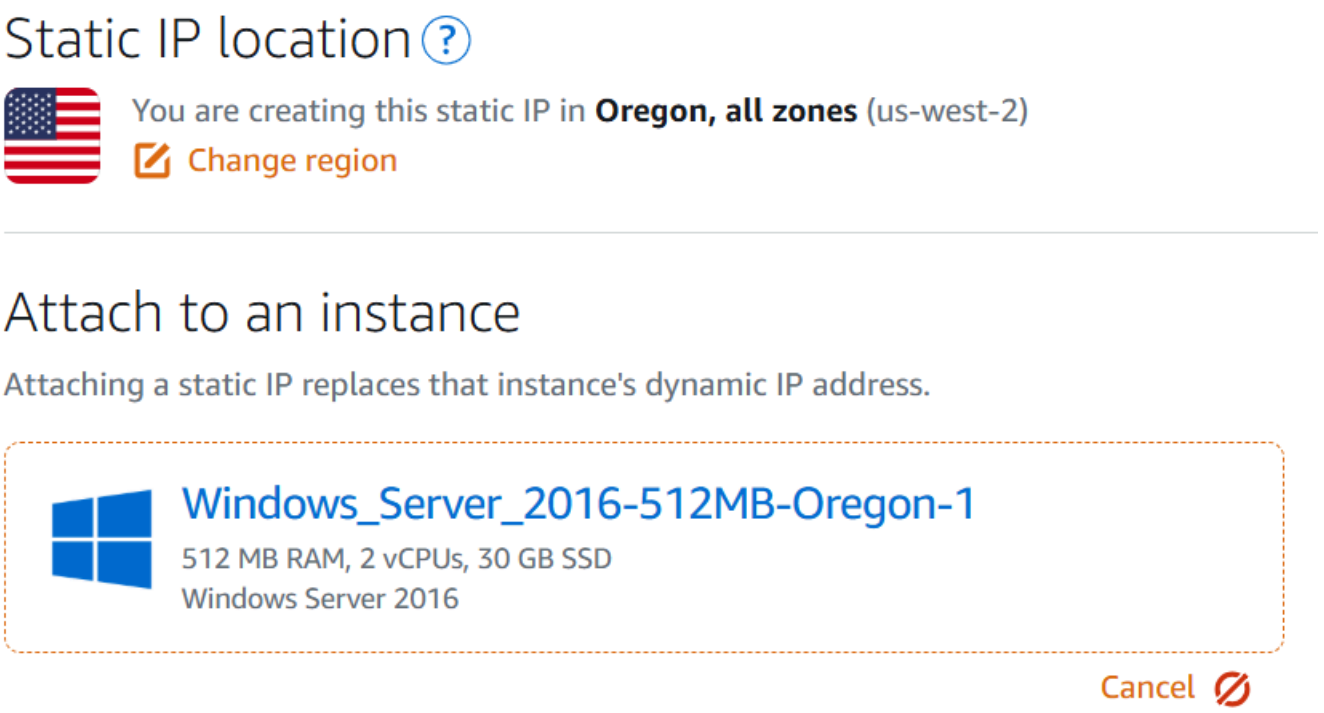
1. Lightsail 홈 페이지의 인스턴스 섹션에서 실행 중인 Windows Server 2016 인스턴스를 선택합니다.



2. 네트워킹 탭을 선택한 후 고정 IP 생성을 선택합니다.



3. 고정 IP 위치 및 연결된 인스턴스는 이 자습서의 앞부분에서 선택한 인스턴스를 기반으로 사전 선택됩니다.



4. 고정 IP의 이름을 입력합니다.

   리소스 이름:

   - Lightsail 계정 AWS 리전 의 각 내에서 고유해야 합니다.
   - 2~255자의 문자로 구성되어야 합니다.
   - 영숫자 문자 또는 숫자로 시작하고 끝나야 합니다.
   - 영숫자 문자, 숫자, 마침표, 대시, 밑줄이 포함될 수 있습니다.

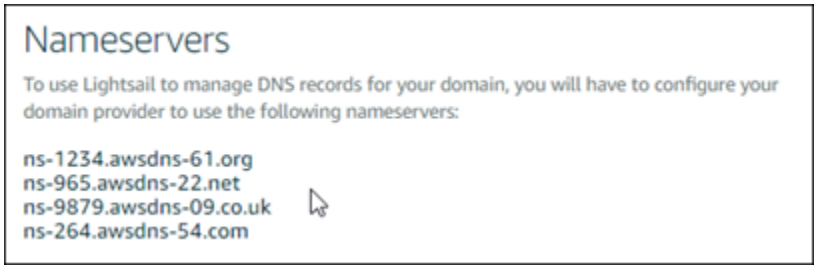
5. 생성(Create)을 선택합니다.

5단계: DNS 영역 생성 및 Windows Server 2016 인스턴스에 도메인 매핑

도메인의 DNS 레코드 관리를 Lightsail에 이전합니다. 이렇게 하면 도메인을 Windows Server 2016 인스턴스에 보다 쉽게 매핑하고 Lightsail 콘솔을 사용하여 웹 사이트의 모든 리소스를 관리할 수 있습니다. 자세한 내용은 Lightsail 설명서의 DNS 영역을 생성하여 도메인의 DNS 레코드 관리를 참조하세요.

1. 홈 페이지의 도메인 및 DNSLightsail 섹션에서 DNS 영역 생성을 선택합니다.
2. 도메인을 입력하고 DNS 영역 생성을 선택합니다.
3. 페이지에 나열된 이름 서버 주소를 기록합니다.

   이 이름 서버 주소를 도메인 이름의 등록 기관에 추가하여 도메인의 DNS 레코드 관리를 Lightsail로 이전합니다.



4. 도메인의 DNS 레코드 관리가 Lightsail로 이전된 후 다음과 같이 A 레코드를 추가하여 도메인의 정점을 LAMP 인스턴스로 가리킵니다.

   a. 할당(Assignments) 탭에서 할당 추가(Add assignment)를 선택합니다.

b. 도메인 선택(Select a domain) 필드에서 도메인 또는 하위 도메인을 선택합니다.

c. 리소스 선택(Select a resource) 드롭다운에서 이 자습서의 앞부분에서 만든 LAMP 인스턴스를 선택합니다.

d. 할당(Assign)을 선택합니다.

도메인이 LAMP 인스턴스로 트래픽을 라우팅하기 전에 인터넷 DNS를 통해 변경 사항이 전파될 때까지 기다립니다.

다음 단계

다음은 Amazon Lightsail에서 Windows Server 2016 인스턴스를 실행한 후 수행할 수 있는 몇 가지 추가 단계입니다.

- Windows Server 인스턴스의 스냅샷 생성
- Windows Server 기반 Lightsail 인스턴스의 보안 유지를 위한 모범 사례
- 블록 스토리지 디스크 생성 및 Windows Server 인스턴스에 연결
- Windows Server 인스턴스의 스토리지 공간 확장

를 사용하여 Lightsail API 활동 모니터링 AWS CloudTrail

Amazon Lightsail는에서 사용자 AWS CloudTrail, 역할 또는 서비스가 수행한 작업에 대한 레코드를 제공하는 AWS 서비스와 통합됩니다Lightsail. CloudTrail은 Lightsail에 대한 모든 API 직접 호출을 이벤트로 캡처합니다. 캡처되는 직접 호출에는 Lightsail 콘솔로부터의 직접 호출과 Lightsail API 작업에 대한 코드 직접 호출이 포함됩니다. 추적을 생성하면 Lightsail 이벤트를 포함한 CloudTrail 이벤트를 지속적으로 Amazon S3 버킷에 배포할 수 있습니다. 추적을 구성하지 않은 경우에도 CloudTrail 콘솔의 이벤트 기록에서 최신 이벤트를 볼 수 있습니다. CloudTrail에서 수집한 정보를 사용하여 Lightsail에 수행된 요청, 요청이 수행된 IP 주소, 요청을 수행한 사람, 요청이 수행된 시간 및 추가 세부 정보를 확인할 수 있습니다.

CloudTrail에 대한 자세한 내용은 AWS CloudTrail 사용 설명서를 참조하세요.

CloudTrail의 Lightsail 정보

AWS 계정을 생성할 때 계정에서 CloudTrail이 활성화됩니다. 에서 활동이 발생하면 Lightsail해당 활동이 이벤트 기록의 다른 AWS 서비스 이벤트와 함께 CloudTrail 이벤트에 기록됩니다. AWS 계정에서 최

근 이벤트를 보고 검색하고 다운로드할 수 있습니다. 자세한 설명은 CloudTrail 이벤트 기록으로 이벤트 보기를 참조하세요.

에 대한 이벤트를 포함하여 AWS 계정에 이벤트를 지속적으로 기록하려면 추적을 Lightsail생성합니다. CloudTrail은 추적을 사용하여 Amazon S3 버킷으로 로그 파일을 전송할 수 있습니다. 콘솔에서 추적을 생성하면 기본적으로 모든 리전에 추적이 적용됩니다. 추적은 AWS 파티션의 모든 리전에서 이벤트를 로깅하고 지정한 Amazon S3 버킷으로 로그 파일을 전송합니다. 추가적으로, CloudTrail 로그에서 수집된 이벤트 데이터를 추가 분석 및 처리하도록 다른 AWS 서비스를 구성할 수 있습니다. 자세한 내용은 다음 자료를 참조하세요.

- 추적 생성 개요
- CloudTrail 지원 서비스 및 통합
- CloudTrail에서 Amazon SNS 알림 구성
- 여러 리전으로부터 CloudTrail 로그 파일 받기 및 여러 계정으로부터 CloudTrail 로그 파일 받기

모든 Lightsail 작업은 CloudTrail에서 로깅되고 Amazon Lightsail API 참조에 기록됩니다. 예를 들어, GetInstance, AttachStaticIp 및 RebootInstance 섹션을 호출하면 CloudTrail 로그 파일에 항목이 생성됩니다.

모든 이벤트 또는 로그 항목에는 요청을 생성했던 사용자에 관한 정보가 포함됩니다. ID 정보를 이용하면 다음을 쉽게 판단할 수 있습니다.

- 요청이 루트 또는 AWS Identity and Access Management (IAM) 사용자 자격 증명으로 이루어졌는지 여부입니다.
- 역할 또는 페더레이션 사용자의 임시 자격 증명을 사용하여 요청이 생성되었는지 여부.
- 요청이 다른 AWS 서비스에서 이루어졌는지 여부입니다.

자세한 설명은 CloudTrail userIdentity 요소를 참조하십시오.

Lightsail 로그 파일 항목 이해

추적이란 지정한 Amazon S3 버킷에 이벤트를 로그 파일로 입력할 수 있게 하는 구성입니다. CloudTrail 로그 파일에는 하나 이상의 로그 항목이 포함될 수 있습니다. 이벤트는 모든 소스로부터의 단일 요청을 나타내며 요청 작업, 작업 날짜와 시간, 요청 파라미터 등에 대한 정보가 들어 있습니다. CloudTrail 로그 파일은 퍼블릭 API 간접 호출의 주문 스택 트레이스가 아니므로 특정 순서로 표시되지 않습니다.

Lightsail 문제를 해결하기 위한 HAR 파일 생성

Amazon Lightsail 콘솔 또는 Lightsail 가상 프라이빗 서버(VPS)에 문제가 있는 경우 웹 브라우저에서 HAR 파일을 제출하도록 요청할 지원 수 있습니다. HAR 파일에는 일반적이고 진단하기 어려운 문제를 해결하는 데 도움이 되는 중요한 정보가 포함되어 있습니다. 또한 HAR 파일을 사용하면가 이러한 문제를 조사하거나 복제 지원 할 수 있습니다.

> **Important**
>
> HAR 파일은 사용자 이름, 암호 및 키와 같은 민감한 정보를 캡처할 수 있습니다. 공유하기 전에 HAR 파일에서 민감한 정보를 모두 제거해야 합니다.

이 안내서에서는 웹 브라우저에서 HAR 파일을 생성하는 방법을 알아봅니다. HTTP 아카이브(HAR) 파일은 브라우저에서 기록한 최신 네트워크 활동을 포함하는 JSON 파일입니다. 단계별 절차를 따라 HAR 파일을 생성합니다.

목차

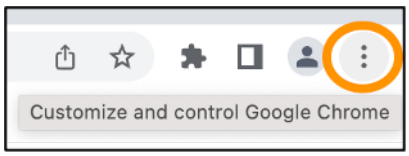
- 1단계: 브라우저에서 HAR 파일 생성
- 2단계: HAR 파일을 편집하여 민감한 정보 제거
- 3단계: 검토를 위해 HAR 파일 제출

1단계: 브라우저에서 HAR 파일 생성

> **Note**
>
> 이 지침은 Google Chrome 버전 101.0.4951.64, Microsoft Edge(Chromium) 버전 101.0.1210.47, Mozilla Firefox 버전 91.9에서 마지막으로 테스트되었습니다. 이러한 브라우저는 타사 제품이므로 이러한 지침은 최신 버전이나 사용 중인 버전의 환경과 일치하지 않을 수 있습니다. 레거시 Microsoft Edge(EdgeHTML) 또는 macOS용 Apple Safari와 같은 다른 브라우저에서는 HAR 파일을 생성하는 프로세스가 비슷할 수 있지만 단계는 다를 수 있습니다.

Google Chrome

1. 브라우저의 오른쪽 상단에서 Customize and control Google Chrome(Google Chrome 맞춤설정 및 제어)을 선택합니다.

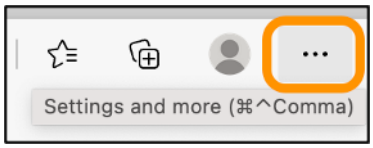


2. More tools(추가 도구)에서 일시 중지한 다음 Developer tools(개발자 도구)를 선택합니다.
3. 브라우저에서 DevTools를 연 상태에서 Network(네트워크) 패널을 선택합니다.
4. Preserve log(로그 보존) 확인란을 선택합니다.
5. 현재 네트워크 요청을 모두 지우려면 Clear(지우기)를 선택합니다.
6. 발생하고 있는 문제 재현
7. DevTools에서 네트워크 요청의 컨텍스트 메뉴(마우스 오른쪽 버튼 클릭)를 엽니다.
8. Save all as HAR with content(콘텐츠를 포함하여 모두 HAR 파일로 저장)를 선택한 다음 파일을 저장합니다.

자세한 내용은 Google 개발자 웹 사이트에서 Chrome DevTools 열기 및 모든 네트워크 요청을 HAR 파일에 저장을 참조하세요.

Microsoft Edge(Chromium)

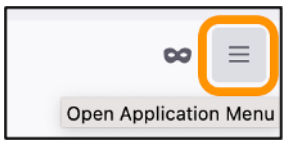
1. 브라우저의 오른쪽 상단에서 Settings and more(설정 및 기타)를 선택합니다.



2. More tools(추가 도구)에서 일시 중지한 다음 Developer tools(개발자 도구)를 선택합니다.
3. 브라우저에서 DevTools를 연 상태에서 Network(네트워크) 패널을 선택합니다.
4. Preserve log(로그 보존) 확인란을 선택합니다.
5. 현재 네트워크 요청을 모두 지우려면 Clear(지우기)를 선택합니다.
6. 발생하고 있는 문제 재현
7. DevTools에서 네트워크 요청의 컨텍스트 메뉴(마우스 오른쪽 버튼 클릭)를 엽니다.
8. Save all as HAR with content(콘텐츠를 포함하여 모두 HAR 파일로 저장)를 선택한 다음 파일을 저장합니다.

Mozilla Firefox

1. 브라우저의 오른쪽 상단에서 Open Application Menu(애플리케이션 메뉴 열기)를 선택합니다.



2. More tools(추가 도구)에서 Web Developer tools(웹 개발자 도구)를 선택합니다.
3. Web Developer(웹 개발자) 메뉴에서 Network(네트워크)를 선택합니다. (일부 Firefox 버전에서는 Web Developer(웹 개발자) 메뉴가 Tools(도구) 메뉴에 있습니다.)
4. 톱니바퀴 아이콘을 선택한 다음 Persist Logs(로그 지속)를 선택합니다.
5. 현재 네트워크 요청을 모두 지우려면 휴지통 아이콘(Clear(지우기))을 선택합니다.
6. 발생하고 있는 문제를 재현합니다.
7. Network Monitor에서 요청 목록의 네트워크 요청에서 컨텍스트 메뉴(마우스 오른쪽 버튼 클릭)를 엽니다.
8. Save All As HAR(모두 HAR 파일로 저장)을 선택한 다음 파일을 저장합니다.

2단계: HAR 파일을 편집하여 민감한 정보 제거

1. 텍스트 편집기 애플리케이션에서 HAR 파일을 엽니다.
2. 텍스트 편집기의 찾기 및 바꾸기 도구를 사용하여 HAR 파일에 캡처된 모든 민감한 정보를 식별하고 바꿉니다. 여기에는 파일을 생성할 때 브라우저에 입력한 모든 사용자 이름, 암호 및 키가 포함됩니다.
3. 민감한 정보가 제거된 HAR 파일을 저장합니다.

3단계: 검토를 위해 HAR 파일 제출

1. AWS Support Center Console의 지원 사례 열기에서 지원 사례를 선택합니다.
2. 지원 사례에서 원하는 연락처 옵션을 선택하고 편집한 HAR 파일을 첨부한 다음 제출합니다.

Lightsail에서 Prometheus를 사용하여 시스템 리소스 및 앱 모니터링

Prometheus는 다양한 시스템 리소스 및 애플리케이션을 관리하기 위한 오픈 소스 시계열 모니터링 툴입니다. 툴은 다차원 데이터 모델, 수집된 데이터 쿼리 기능, Grafana를 통한 상세한 보고 및 데이터 시각화를 제공합니다.

기본적으로 Prometheus는 설치된 서버에서 메트릭을 수집할 수 있습니다. 노드 내보내기의 도움을 받아 웹 서버, 컨테이너, 데이터베이스, 맞춤형 애플리케이션 및 기타 서드파티 시스템 등의 다른 리소

스에서 메트릭을 수집합니다. 이 튜토리얼에서는 Lightsail 인스턴스에서 노드 내보내기를 사용하여 Prometheus를 설치하고 구성하는 방법을 설명합니다. 사용 가능한 전체 내보내기 목록은 Prometheus 문서에서 내보내기 및 통합을 참조합니다.

목차

- 1단계: 필수 구성 요소 완성
- 2단계: Lightsail 인스턴스에 사용자와 로컬 시스템 디렉터리 추가
- 3단계: Prometheus 바이너리 패키지 다운로드
- 4단계: Prometheus 구성
- 5단계: Prometheus 시작
- 6단계: Node Exporter 시작
- 7단계: Node Exporter 데이터 수집기로 Prometheus 구성

1단계: 필수 구성 요소 완성

Amazon Lightsail 인스턴스에 Prometheus 설치하기 전에 다음을 진행해야 합니다.

- Lightsail에서 인스턴스를 생성합니다. 인스턴스를 위해서는 Ubuntu 20.04 LTS 블루프린트 사용을 권장합니다. 자세한 내용은 Amazon Lightsail에서 인스턴스 생성을 참조합니다.
- 고정 IP 주소를 생성하고 새 인스턴스에 연결합니다. 자세한 내용은 Amazon Lightsail에서 고정 IP 생성을 참조합니다.
- 새 인스턴스의 방화벽에서 9090 및 9100 포트를 엽니다. Prometheus를 사용하려면 포트 9090 및 9100이 열려 있어야 합니다. 자세한 내용은 Amazon Lightsail의 인스턴스 방화벽 규칙 추가 및 편집을 참조합니다.

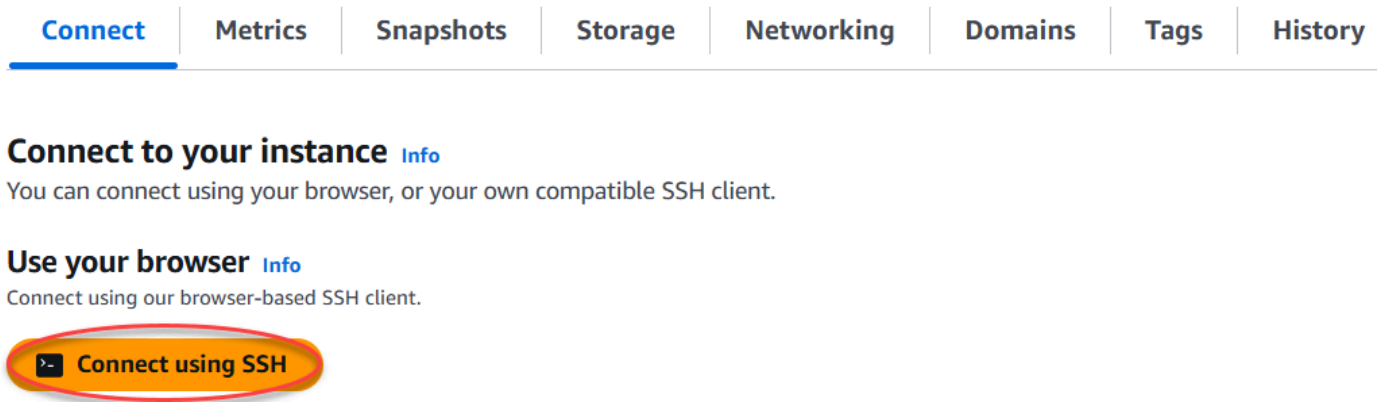
2단계: Lightsail 인스턴스에 사용자와 로컬 시스템 디렉터리 추가

다음 절차를 완료하면 SSH로 Lightsail 인스턴스에 연결하고, 사용자 디렉터리와 시스템 디렉터리를 추가합니다. 이 절차는 다음 Linux 사용자 계정을 생성합니다.

- `prometheus` - 이 계정은 서버 환경을 설치하고 구성하는 데 사용됩니다.
- `exporter` - 이 계정은 `node_exporter` 확장을 구성하는 데 사용합니다.

이러한 사용자 계정은 관리 목적으로만 생성되므로 이 설정 범위를 벗어나는 추가 사용자 서비스나 권한이 필요하지 않습니다. 이 절차에서는 Prometheus가 리소스 모니터링에 사용하는 파일, 서비스 설정, 데이터를 저장하고 관리하기 위한 디렉터리도 생성합니다.

1. Lightsail 콘솔에 로그인합니다.
2. 인스턴스 관리 페이지의 연결(Connect) 탭에서 SSH로 연결(Connect using SSH)을 선택합니다.



3. 연결한 후 다음 명령을 하나씩 입력하여 `prometheus`와 `exporter` Linux 사용자 계정 2개를 생성합니다.

```
sudo useradd --no-create-home --shell /bin/false prometheus
```

```
sudo useradd --no-create-home --shell /bin/false exporter
```

4. 다음 명령을 하나씩 입력하여 로컬 시스템 디렉터리를 생성합니다.

```
sudo mkdir /etc/prometheus /var/lib/prometheus
```

```
sudo chown prometheus:prometheus /etc/prometheus
```

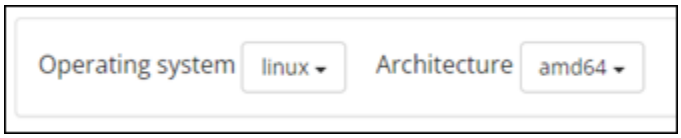
```
sudo chown prometheus:prometheus /var/lib/prometheus
```

3단계: Prometheus 바이너리 패키지 다운로드

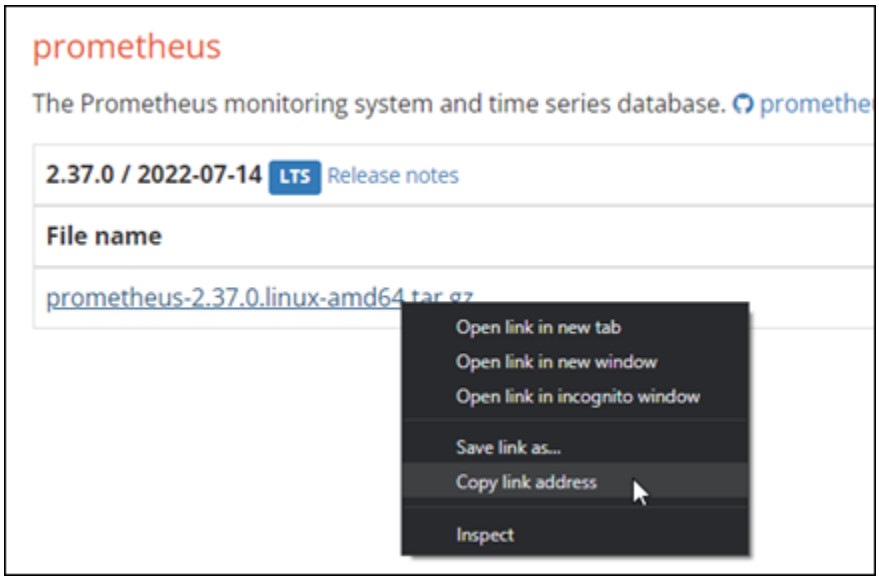
다음 절차를 완료하면 Lightsail 인스턴스에 Prometheus 바이너리 패키지를 다운로드합니다.

1. 로컬 컴퓨터에서 웹 브라우저를 열고 Prometheus 다운로드 페이지로 들어갑니다.

2. 페이지 상단의 운영 체제(Operating system) 드롭다운에서 Linux를 선택합니다. 아키텍처에서 amd64를 선택합니다.



3. Prometheus 다운로드 링크를 선택하거나 마우스 오른쪽 버튼으로 클릭하고, 컴퓨터의 텍스트 파일에 링크 주소를 복사합니다. 보이는 node_exporter 다운로드 링크에 대해서도 똑같이 반복합니다. 이 절차의 뒷부분에서 두 개의 복사한 주소를 모두 사용합니다.



4. SSH로 Lightsail 인스턴스에 연결합니다.
5. 다음 명령을 입력하여 디렉터리를 홈 디렉터리로 변경합니다.

```
cd ~
```

6. 다음 명령을 입력하여 인스턴스에 Prometheus 바이너리 패키지를 다운로드합니다.

```
curl -LO prometheus-download-address
```

*prometheus-download-address*에 이 절차에서 앞서 복사한 주소를 덮어씌웁니다. 명령은 주소를 추가했을 때 다음 예시 출력과 비슷하게 출력됩니다.

```
curl -LO https://github.com/prometheus/prometheus/releases/download/v2.37.0/
prometheus-2.37.0.linux-amd64.tar.gz
```

7. 다음 명령을 입력하여 node_exporter 실행 파일을 인스턴스로 다운로드합니다.

```
curl -LO node_exporter-download-address
```

*node_exporter-download-address*에 이 절차에서 앞서 복사한 주소를 덮어씌웁니다. 명령은 주소를 추가했을 때 다음 예시 출력과 비슷하게 출력됩니다.

```
curl -LO https://github.com/prometheus/node_exporter/releases/download/v1.3.1/
node_exporter-1.3.1.linux-amd64.tar.gz
```

8. 다음 명령을 하나씩 실행하여 다운로드한 Prometheus 및 Node Exporter 파일의 내용을 추출합니다.

```
tar -xvf prometheus-2.37.0.linux-amd64.tar.gz
```

```
tar -xvf node_exporter-1.3.1.linux-amd64.tar.gz
```

다운로드한 파일의 내용을 추출하고 나면 하위 디렉터리가 여러 개 만들어집니다.

9. 다음 명령을 하나씩 입력하여 압축을 푼 prometheus과 promtool 파일을 /usr/local/bin 프로그램 디렉터리에 복사합니다.

```
sudo cp -p ./prometheus-2.37.0.linux-amd64/prometheus /usr/local/bin
```

```
sudo cp -p ./prometheus-2.37.0.linux-amd64/promtool /usr/local/bin
```

10. 다음 명령을 입력하여 prometheus과 promtool파일의 소유권을 이 튜토리얼에서 앞서 생성한 prometheus 사용자로 변경합니다.

```
sudo chown prometheus:prometheus /usr/local/bin/prom*
```

11. 다음 명령을 하나씩 입력하여/etc/prometheus의 하위 디렉터리인 consoles과 console_libraries를 복사합니다. -r 옵션은 계층 내 모든 디렉터리의 재귀 복사를 수행합니다.

```
sudo cp -r ./prometheus-2.37.0.linux-amd64/consoles /etc/prometheus
```

```
sudo cp -r ./prometheus-2.37.0.linux-amd64/console_libraries /etc/prometheus
```

12. 다음 명령을 하나씩 입력하여 복사한 파일의 소유권을 이 튜토리얼에서 앞서 생성한 prometheus 사용자로 변경합니다. 이-R 옵션은 계층 내의 모든 파일과 디렉터리에 대해 재귀적인 소유권 변경을 수행합니다.

```
sudo chown -R prometheus:prometheus /etc/prometheus/consoles
```

```
sudo chown -R prometheus:prometheus /etc/prometheus/console_libraries
```

13. 다음 명령을 하나씩 입력하여 구성 파일 prometheus.yml을 /etc/prometheus 디렉터리에 복사하고 복사한 파일의 소유권을 이 튜토리얼에서 앞서 생성한 prometheus 사용자로 변경합니다.

```
sudo cp -p ./prometheus-2.37.0.linux-amd64/prometheus.yml /etc/prometheus
```

```
sudo chown prometheus:prometheus /etc/prometheus/prometheus.yml
```

14. 다음 명령을 입력하여 ./node_exporter* 하위 디렉터리의 node_exporter 파일을 /usr/local/bin 프로그램 디렉터리로 복사합니다.

```
sudo cp -p ./node_exporter-1.3.1.linux-amd64/node_exporter /usr/local/bin
```

15. 다음 명령을 입력하여 파일의 소유권을 이 튜토리얼에서 앞서 생성한 exporter 사용자로 변경합니다.

```
sudo chown exporter:exporter /usr/local/bin/node_exporter
```

4단계: Prometheus 구성

다음 절차를 완료하면 Prometheus를 구성합니다. 이 절차에서는 Prometheus 툴의 다양한 설정을 포함한 prometheus.yml 파일을 열고 편집합니다. Prometheus는 파일에 구성한 설정을 기반으로 모니터링 환경을 설정합니다.

1. SSH로 Lightsail 인스턴스에 연결합니다.
2. 다음 명령을 입력하여 prometheus.yml 파일을 열고 편집하기 전에 백업 사본을 생성합니다.

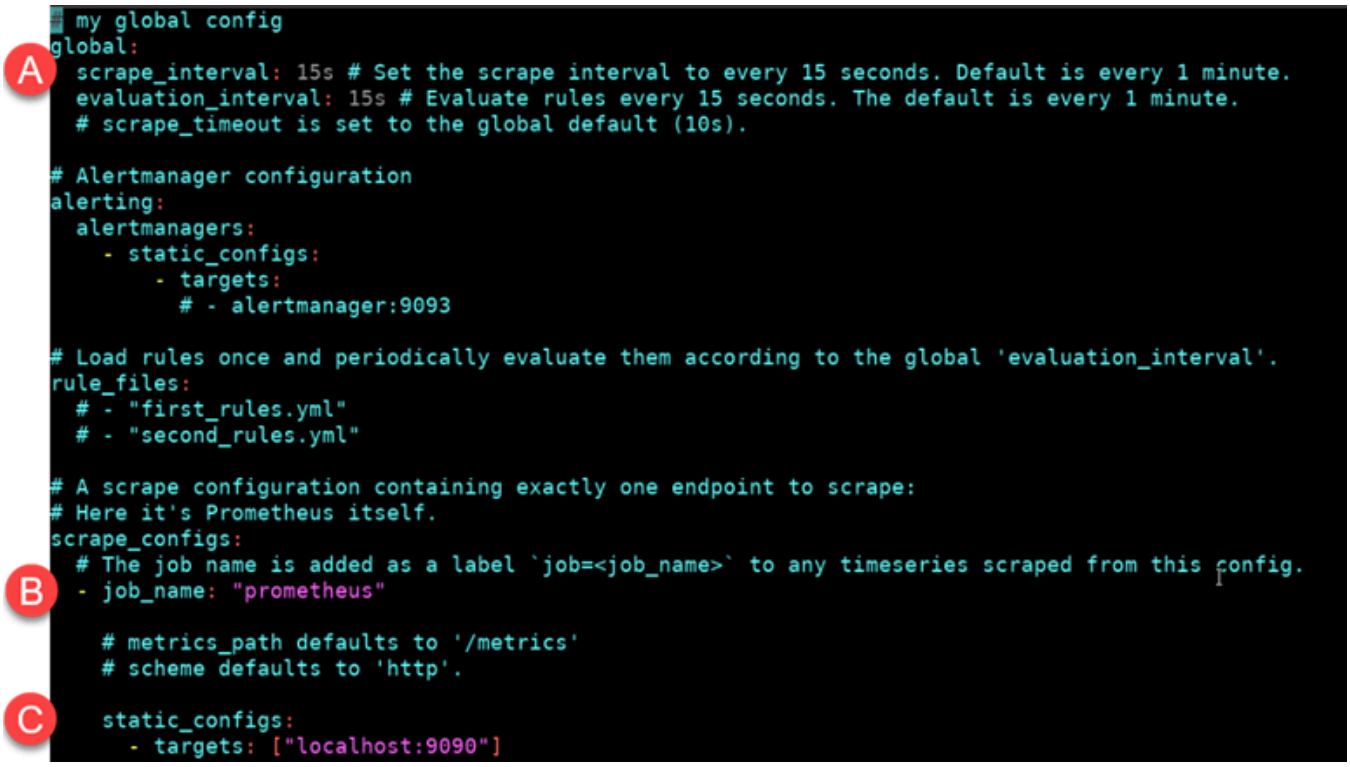
```
sudo cp /etc/prometheus/prometheus.yml /etc/prometheus/prometheus.yml.backup
```

3. 다음 명령을 입력하여 Vim에서 `prometheus.yml` 파일을 엽니다.

```
sudo vim /etc/prometheus/prometheus.yml
```

다음은 `prometheus.yml` 파일에서 구성할 수 있는 몇 가지 중요한 매개 변수입니다.

- `scrape_interval` - `global` 헤더 아래에 있는 매개 변수는 Prometheus가 주어진 대상의 메트릭 데이터를 어떤 빈도로 수집 또는 스크랩하는지 지정된 시간 간격(초 단위)을 정의합니다. `global` 태그가 가리키는 바와 같이 이 설정은 Prometheus가 모니터링하는 모든 리소스에 범용으로 적용됩니다. 이 설정은 개별 내보내기가 전역 값을 재정의하는 다른 값을 제공하지 않는 한 내보내기에도 적용됩니다. 이 매개 변수는 현재 값인 15초로 설정을 유지할 수 있습니다.
- `job_name` - `scrape_configs` 헤더 이래에 있는 이 매개 변수는 데이터 쿼리 또는 시각적 표시의 결과 집합에서 내보내기를 식별하는 레이블입니다. 작업 환경에서 모니터링되는 리소스를 가장 잘 반영하도록 작업 이름의 값을 지정할 수 있습니다. 예를 들어, 웹 사이트 관리 작업에 `business-web-app`이라는 레이블을 지정하거나, 데이터베이스에 `mysql-db-1`이라는 레이블을 지정할 수 있습니다. 이 초기 설정에서는 Prometheus 서버만 모니터링하므로 현재 `prometheus` 값을 유지할 수 있습니다.
- `targets` - `static_configs` 헤더 아래에 있는 `targets` 설정은 `ip_addr:port` 키-값 페어를 사용하여 지정된 내보내기가 실행 중인 위치를 식별합니다. 절차의 4~7단계에서 기본 설정을 변경합니다.



> **Note**
>
> 이 초기 설정의 경우에는 `alerting`과 `rule_files` 매개 변수를 구성할 필요가 없습니다.

4. Vim으로 연 `prometheus.yml` 파일에서 I를 눌러 Vim을 삽입 모드를 시작합니다.
5. 스크롤하여 `static_configs` 헤더 아래에 있는 `targets` 매개 변수를 찾습니다.
6. 기본 설정을 *<ip_addr>*:9090(으)로 변경합니다. *<ip_addr>*에서 인스턴스의 고정 IP 주소로 변경합니다. 수정한 매개 변수는 다음 예와 같은 형식이어야 합니다.

```
static_configs:
  - targets: ["192.0.2.0:9090"]
```

7. Esc키를 눌러 삽입 모드를 종료하고, :wq!를 입력하여 변경 사항 저장 후 Vim을 종료합니다.
8. (선택 사항) 문제가 발생한 경우 다음 명령을 입력하여 `prometheus.yml` 파일과 이 절차에서 앞서 생성한 백업 파일을 교체합니다.

```
sudo cp /etc/prometheus/prometheus.yml.backup /etc/prometheus/prometheus.yml
```

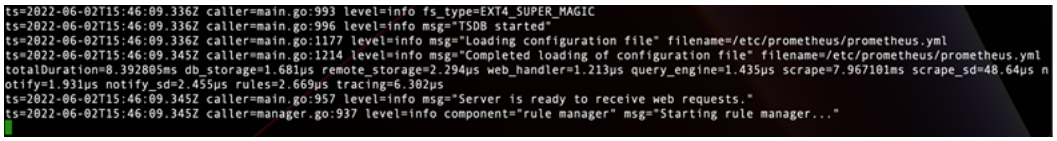
5단계: Prometheus 시작

다음 절차를 완료하면 인스턴스에서 Prometheus 서비스를 시작합니다.

1. SSH로 Lightsail 인스턴스에 연결합니다.
2. 다음 명령을 입력하여 Prometheus 서비스를 시작합니다.

```
sudo -u prometheus /usr/local/bin/prometheus --config.file /etc/prometheus/
prometheus.yml --storage.tsdb.path /var/lib/prometheus --web.console.templates=/
etc/prometheus/consoles --web.console.libraries=/etc/prometheus/console_libraries
```

명령은 시작 프로세스 및 기타 서비스에 대한 세부 정보를 출력합니다. 또한 서비스가 포트 9090에서 수신 대기 중임을 나타내야 합니다.



서비스가 시작되지 않는다면, 이 튜토리얼의 1단계: 필수 구성 요소 완성 섹션에서 이 포트의 트래픽을 허용하는 인스턴스 방화벽 규칙 생성에 대한 정보 참조합니다. 기타 오류는 `prometheus.yml` 파일에 구문 오류가 없는지 검토하여 확인합니다.

3. 실행 중인 서비스가 검증되면 Ctrl+C를 눌러 이를 중단합니다.
4. 다음 명령을 입력하여 Vim에서 `systemd` 구성 파일을 엽니다. 이 파일은 프로메테우스를 시작하는 데 사용합니다.

```
sudo vim /etc/systemd/system/prometheus.service
```

5. 다음 라인을 파일에 삽입합니다.

```
[Unit]
Description=PromServer
Wants=network-online.target
After=network-online.target

[Service]
User=prometheus
Group=prometheus
Type=simple
ExecStart=/usr/local/bin/prometheus \
--config.file /etc/prometheus/prometheus.yml \
--storage.tsdb.path /var/lib/prometheus/ \
--web.console.templates=/etc/prometheus/consoles \
--web.console.libraries=/etc/prometheus/console_libraries

[Install]
WantedBy=multi-user.target
```

위 지침은 Linux `systemd`서비스 매니저에서 서버의 Prometheus를 시작하기 위해 사용하는 지침입니다. Prometheus를 호출하면 Prometheus는 `prometheus` 사용자로 실행되고, 구성 설정을 로드, 파일`/var/lib/prometheus` 디렉터리에 시계열 데이터 저장을 위해 `prometheus.yml` 파일을 참조합니다. 명령줄에서 `man systemd`를 실행하여 서비스에 대한 추가 정보를 볼 수 있습니다.

6. Esc키를 눌러 삽입 모드를 종료하고, :wq!를 입력하여 변경 사항 저장 후 Vim을 종료합니다.
7. 다음 명령을 입력하여 `systemd` 서비스 매니저로 정보를 로드합니다.

```
sudo systemctl daemon-reload
```

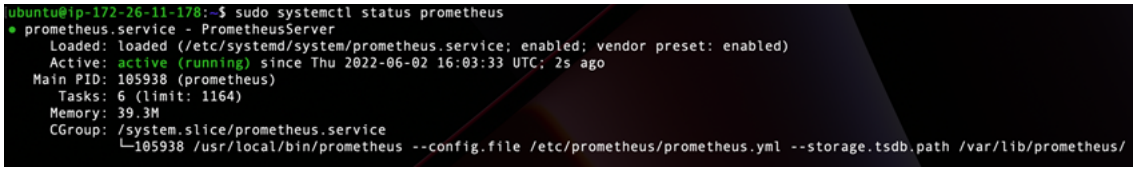
8. Prometheus을 설정하려면 다음 명령을 입력합니다.

```
sudo systemctl start prometheus
```

9. 다음 명령을 입력하여 Prometheus 서비스의 상태를 확인합니다.

```
sudo systemctl status prometheus
```

서비스가 제대로 실행된다면 다음 예제와 비슷한 출력이 표시됩니다.



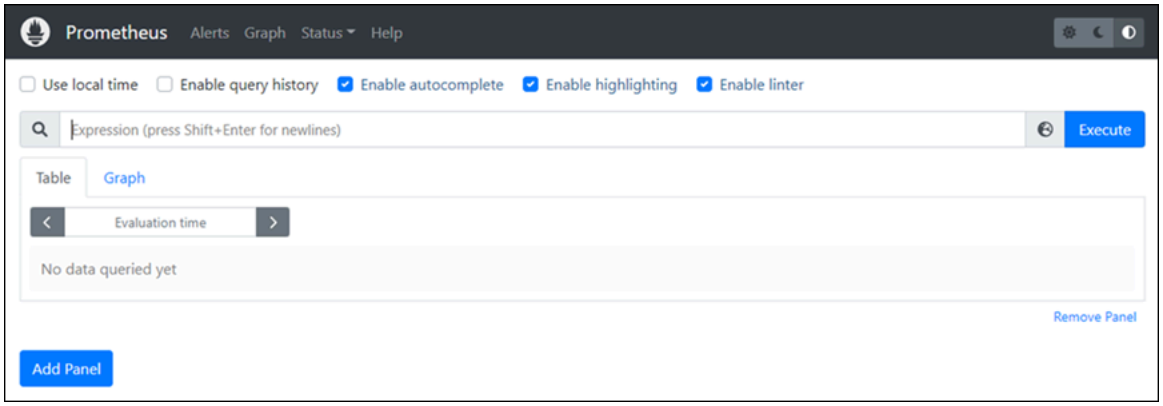
10. Q를 눌러 상태 명령을 종료합니다.
11. 다음 명령을 입력하여 인스턴스가 부팅되었을 때 Prometheus를 활성화합니다.

```
sudo systemctl enable prometheus
```

12. 로컬 컴퓨터에서 웹 브라우저를 열고 다음 웹 주소로 이동하여 Prometheus 관리 인터페이스를 확인합니다.

```
http:<ip_addr>:9090
```

*<ip_addr>*를 Lightsail 인스턴스의 고정 IP 주소로 변경합니다. 다음 예와 비슷한 대시보드를 볼 수 있습니다.



6단계: Node Exporter 시작

다음 절차를 완료하면 Node Exporter 서비스를 시작합니다.

1. SSH로 Lightsail 인스턴스에 연결합니다.
2. 다음 명령을 입력하여 Vim에서 node_exporter에 대한 systemd 서비스 파일을 생성합니다.

```
sudo vim /etc/systemd/system/node_exporter.service
```

3. Vim에서 I 키를 눌러 삽입 모드를 시작합니다.
4. 파일에 다음 텍스트 행을 추가합니다. 그러면 CPU 로드, 파일 시스템 사용량 및 메모리 리소스에 대한 모니터링 수집기를 갖춘 node_exporter를 구성합니다.

```
[Unit]
Description=NodeExporter
Wants=network-online.target
After=network-online.target

[Service]
User=exporter
Group=exporter
Type=simple
ExecStart=/usr/local/bin/node_exporter --collector.disable-defaults \
--collector.meminfo \
--collector.loadavg \
--collector.filesystem

[Install]
WantedBy=multi-user.target
```

> **Note**
>
> 이 지침은 Node Exporter 기본 머신 메트릭을 비활성화합니다. Ubuntu에서 사용 가능한 전체 지표 목록은 Ubuntu 문서에서 Prometheus node_exporter 매뉴얼 페이지를 참조합니다.

5. Esc키를 눌러 삽입 모드를 종료하고, :wq!를 입력하여 변경 사항 저장 후 Vim을 종료합니다.
6. 다음 명령을 입력하여 systemd프로세스를 다시 로드합니다.

```
sudo systemctl daemon-reload
```

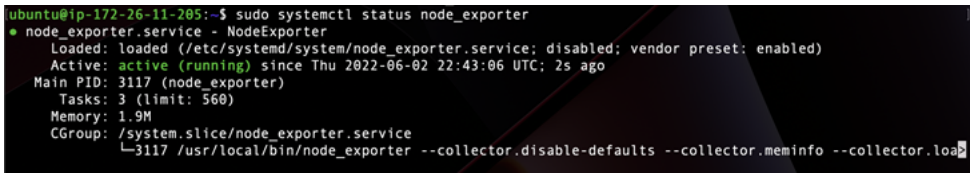
7. 다음 명령을 입력하여 node_exporter 서비스를 시작합니다.

```
sudo systemctl start node_exporter
```

8. 다음 명령을 입력하여 node_exporter 서비스의 상태를 확인합니다.

```
sudo systemctl status node_exporter
```

이 명령이 성공적으로 실행된다면 다음과 비슷한 출력이 표시됩니다.



9. Q를 눌러 상태 명령을 종료합니다.
10. 다음 명령을 입력하여 인스턴스가 부팅되었을 때 Node Exporter의 시작을 활성화합니다.

```
sudo systemctl enable node_exporter
```

7단계: Node Exporter 데이터 수집기로 Prometheus 구성

다음 절차를 완료하면 Node Exporter 데이터 수집기로 Prometheus를 구성합니다. 구성은 job_name에 대한 매개 변수에서prometheus.yml 파일에서 node_exporter에 대한 새로운 매개 변수 추가를 통해 설정합니다.

1. SSH로 Lightsail 인스턴스에 연결합니다.
2. 다음 명령을 입력하여 Vim에서 prometheus.yml 파일을 엽니다.

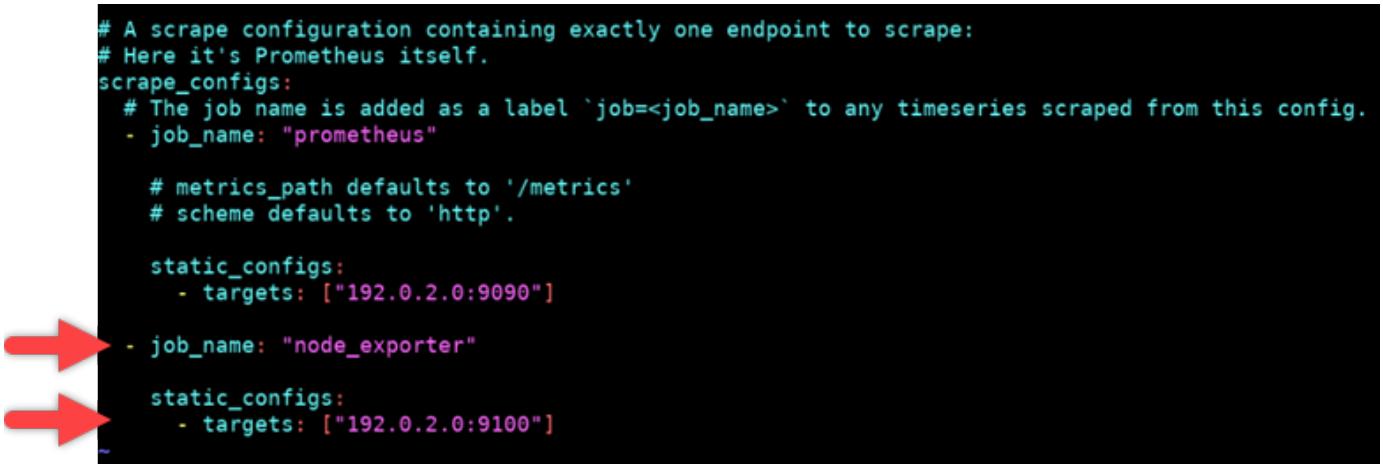
```
sudo vim /etc/prometheus/prometheus.yml
```

3. Vim에서 I 키를 눌러 삽입 모드를 시작합니다.
4. 파일에 다음 텍스트 줄을 기존 - targets: ["*<ip_addr>*:9090"] 매개 변수 아래 추가합니다.

```
- job_name: "node_exporter"
```

```
static_configs:
- targets: ["<ip_addr>:9100"]
```

prometheus.yml 파일에서 수정된 매개 변수는 다음 예와 같은 형식이어야 합니다.



다음을 참조합니다.

- Node Exporter 데이터를 스크랩하는 prometheus 서버에 대한 9100 포트를 수신합니다. 이 튜토리얼의 1단계: 필수 구성 요소 완성 섹션의 설명대로 인스턴스 방화벽 규칙을 생성하는 단계를 따랐는지 확인합니다.
- prometheus job_name의 구성과 마찬가지로 교체<ip_addr>를 Lightsail 인스턴스에 연결된 고정 IP 주소로 변경합니다.

5. Esc키를 눌러 삽입 모드를 종료하고, :wq!를 입력하여 변경 사항 저장 후 Vim을 종료합니다.
6. 다음 명령을 입력하여 구성 파일의 변경 사항을 적용할 수 있도록 Prometheus 서비스를 다시 시작합니다.

```
sudo systemctl restart prometheus
```

7. 다음 명령을 입력하여 Prometheus 서비스의 상태를 확인합니다.

```
sudo systemctl status prometheus
```

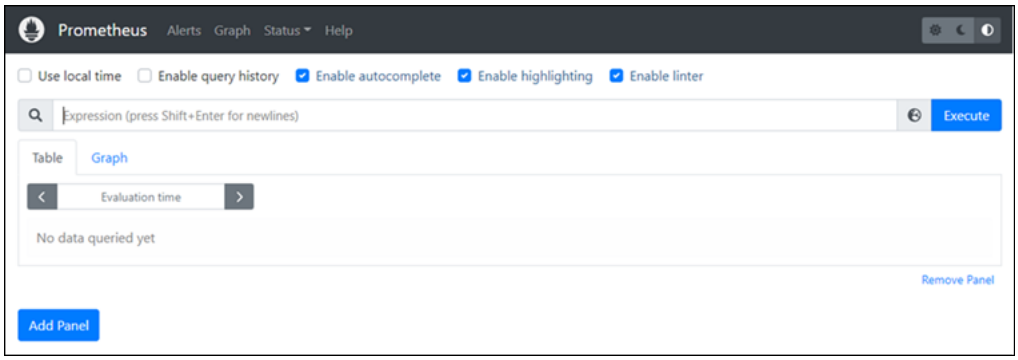
서비스가 제대로 다시 시작한다면 다음과 비슷한 출력이 표시됩니다.

```
ubuntu@ip-172-26-11-178:~$ sudo systemctl status prometheus
● prometheus.service - PrometheusServer
     Loaded: loaded (/etc/systemd/system/prometheus.service; enabled; vendor preset: enabled)
     Active: active (running) since Thu 2022-06-02 16:03:33 UTC; 2s ago
   Main PID: 105938 (prometheus)
      Tasks: 6 (limit: 1164)
     Memory: 39.3M
     CGroup: /system.slice/prometheus.service
             └─105938 /usr/local/bin/prometheus --config.file /etc/prometheus/prometheus.yml --storage.tsdb.path /var/lib/prometheus/
```

8. Q를 눌러 상태 명령을 종료합니다.
9. 로컬 컴퓨터에서 웹 브라우저를 열고 다음 웹 주소로 이동하여 Prometheus 관리 인터페이스를 확인합니다.

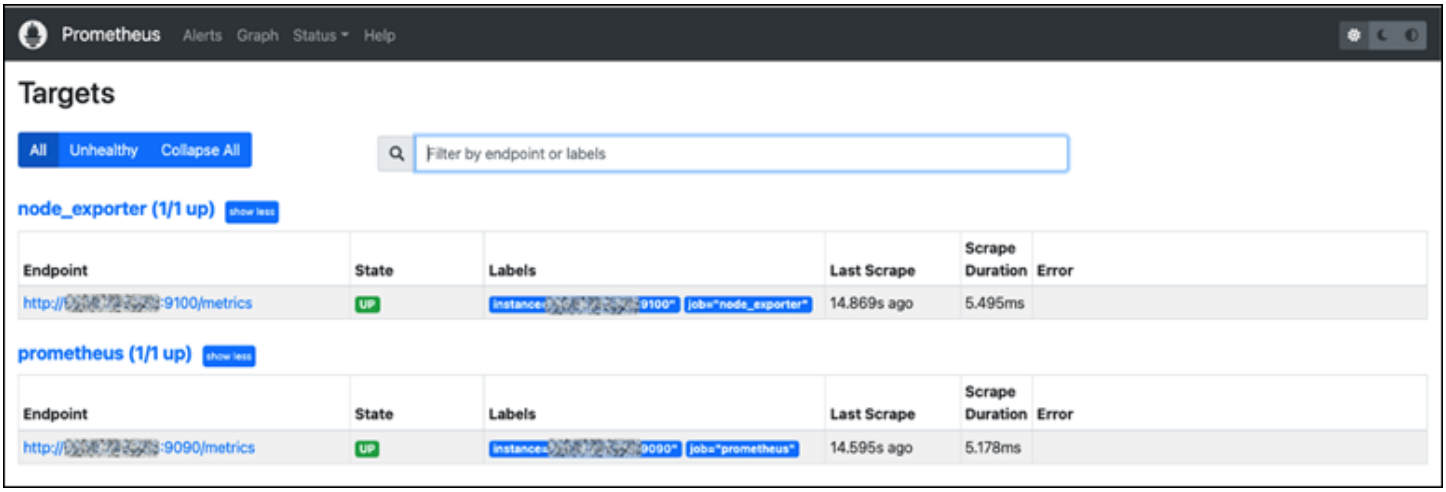
   ```
   http:<ip_addr>:9090
   ```

   *<ip_addr>*를 Lightsail 인스턴스의 고정 IP 주소로 변경합니다. 다음 예와 비슷한 대시보드를 볼 수 있습니다.



10. 기본 메뉴의 상태(Status) 드롭다운에서 대상(Targets)을 선택합니다.

다음 화면에는 두 개의 대상이 보입니다. 첫 번째 대상은node_exporter 메트릭 수집기 작업이고, 두 번째 대상은prometheus작업입니다.



이제 메트릭을 수집하고 서버를 모니터링할 수 있는 환경이 올바르게 설정되었습니다.

Lightsail에서 scp를 사용하여 Linux 인스턴스 간에 파일 전송

Linux의 보안 복사(scp) 명령을 사용하여 로컬 컴퓨터에서 Linux 또는 Unix 인스턴스로 파일을 전송하고 Amazon Lightsail의 한 인스턴스에서 다른 인스턴스로 파일을 전송합니다. scp 명령에 대한 자세한 내용은 man7 웹 사이트의 scp(1) - Linux 설명서 페이지를 참조하세요.

이 자습서에서는 한 인스턴스에서 다른 Lightsail 인스턴스로 파일을 복사하는 단계를 안내합니다.

내용

- 사전 조건
- 1단계: 프라이빗 키(.pem) 파일을 로컬 컴퓨터에 저장
- 2단계: 프라이빗 키의 권한 변경
- 3단계: 프라이빗 키를 인스턴스로 전송
- 4단계: Lightsail Linux 인스턴스와 Unix 인스턴스 간에 파일을 안전하게 전송

사전 조건

- 두 Lightsail 인스턴스의 퍼블릭 IP 주소와 함께 두 개의 인스턴스가 실행 중입니다. 인스턴스의 퍼블릭 IP 주소 가져오기 Lightsail 콘솔에 로그인한 다음, 인스턴스 옆에 표시되는 퍼블릭 IP 주소를 복사합니다.
- SSH 키 페어를 사용하여 두 인스턴스에 모두 액세스할 수 있습니다. 자세한 내용은 Linux 인스턴스에 연결 섹션을 참조하세요.

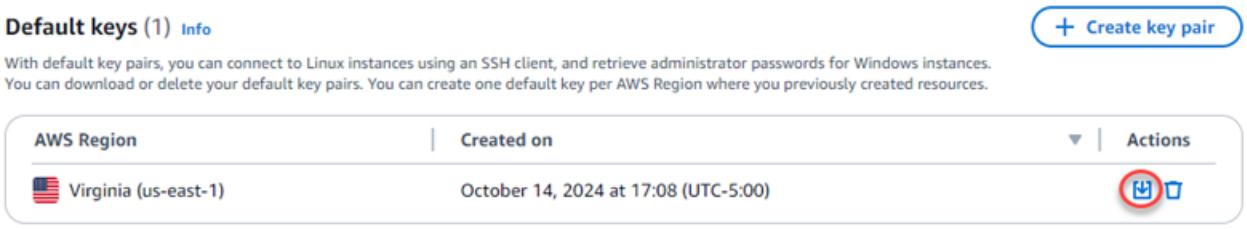
1단계: 프라이빗 키(.pem) 파일을 로컬 컴퓨터에 저장

프라이빗 키(.pem) 파일을 로컬 컴퓨터에 저장하려면 다음 단계를 완료합니다. 대상 인스턴스의 프라이빗 키 파일은 한 인스턴스에서 다른 인스턴스로 파일을 안전하게 전송하는 데 사용됩니다. 동일한 AWS 리전의 인스턴스 간에 파일을 복사하려면 해당 리전의 기본 키를 사용합니다. 서로 다른 리전의 인스턴스 간에 파일을 복사하려면 대상 인스턴스가 있는 리전의 기본 키를 사용합니다. 키 페어에 대한 자세한 내용은 SSH와 인스턴스에 대한 연결 섹션을 참조하세요.

> **Note**
>
> 자체 키 페어를 사용 중이거나 Lightsail 콘솔을 사용하여 키 페어를 생성한 경우에는 자체 프라이빗 키를 찾아서 이를 사용해 인스턴스에 연결합니다. 사용자가 자체 키를 업로드하거나 Lightsail 콘솔을 사용해 키 페어를 생성한 경우에는 Lightsail가 프라이빗 키를 저장하지 않습니다. 프라이빗 키가 없는 scp를 사용하면 인스턴스에 파일을 전송할 수 없습니다.

프라이빗 키(.pem)를 로컬 컴퓨터에 저장

1. Lightsail 콘솔에 로그인합니다.
2. 위쪽 탐색 모음에서 사용자 이름을 선택한 후 드롭다운에서 계정을 선택합니다.
3. SSH 키 탭을 선택합니다.
4. 페이지의 아래로 스크롤하여 기본 키(Default keys) 섹션으로 이동합니다.
5. 파일을 전송할 인스턴스가 위치한 AWS 리전 의 기본 프라이빗 키 옆에 있는 다운로드를 선택합니다.



6. 로컬 드라이브의 안전한 위치에 프라이빗 키를 저장합니다.

   다운로드한 키를 사용자의 홈 디렉터리에 있는 '키(Keys)' 폴더와 같이 모든 SSH 키를 저장하는 디렉터리로 옮길 수 있습니다. 이 가이드의 다음 섹션에서 프라이빗 키가 저장된 디렉터리를 참조해야 합니다. 프라이빗 키가 `.pem`이 아닌 다른 형식으로 저장하려고 할 경우 저장하기 전에 수동으로 형식을 `.pem`으로 변경해야 합니다.

2단계: 프라이빗 키의 권한 변경

다음 절차에서는 프라이빗 키 파일의 권한을 사용자만 읽고 쓸 수 있도록 변경합니다.

프라이빗 키 파일의 권한을 변경

1. 로컬 시스템에서 터미널 창을 엽니다.
2. 다음 명령을 입력하여 키 페어의 프라이빗 키를 사용자만 읽고 쓸 수 있도록 합니다. 이는 일부 운영 체제에서 요구하는 보안 모범 사례입니다.

   ```
   sudo chmod 400 /path/to/private-key.pem
   ```

   명령에서 */path/to/private-key*을 인스턴스에서 사용 중인 키 페어의 프라이빗 키를 저장한 디렉터리 경로로 바꿉니다.

   예:

   ```
   sudo chmod 400 /Users/user/Keys/LightsailDefaultKey-us-west-2.pem
   ```

3단계: 프라이빗 키를 인스턴스로 전송

다음 절차에서는 로컬 컴퓨터에서 scp 명령을 실행하여 프라이빗 키를 소스 인스턴스로 전송합니다.

scp를 사용하여 컴퓨터에서 소스 인스턴스로 프라이빗 키 전송

1. 컴퓨터의 프라이빗 키 파일 위치와 인스턴스의 대상 경로를 확인합니다. 다음 예제에서 프라이빗 키 파일의 이름은 *private-key.pem*이고, 소스 인스턴스의 사용자 이름은 *ec2-user*이고, 소스 인스턴스의 IPv4 주소는 *public-ipv4-address*이고, 소스 인스턴스의 IPv6 주소는 *public-ipv6-address*입니다. *destination-path/*는 프라이빗 키를 전송하는 소스 인스턴스의 위치입니다.

   > **Note**
   >
   > 인스턴스에서 사용하는 블루프린트에 따라 다음 사용자 이름 중 하나를 지정할 수 있습니다.
   >
   > - AlmaLinux OS 9, Amazon Linux 2 및 Amazon Linux 2023, CentOS Stream 9, FreeBSD 및 openSUSE 인스턴스의 경우: `ec2-user`

- Debian 인스턴스: admin
- Ubuntu 인스턴스: ubuntu
- Bitnami 인스턴스: bitnami
- Plesk 인스턴스: ubuntu
- cPanel 및 WHM 인스턴스: centos

- (IPv4) 인스턴스로 프라이빗 키 파일을 전송하려면 컴퓨터에서 다음 명령을 입력합니다.

```
scp -i /path/private-key.pem /path/private-key.pem ec2-user@public-ipv4-address:path/
```

- (IPv6) 인스턴스에 IPv6 주소만 있는 경우 인스턴스로 프라이빗 키 파일을 전송하려면 컴퓨터에서 다음 명령을 입력합니다. IPv6 주소는 이스케이프된([ ]) 대괄호(\)로 묶어야 합니다.

```
scp -i /path/private-key.pem /path/private-key.pem ec2-user@\[public-ipv6-address\]:path/
```

2. SSH를 사용하여 인스턴스에 아직 연결하지 않은 경우 다음과 같은 응답이 표시됩니다.

```
The authenticity of host 'ec2-198-51-100-1.compute-1.amazonaws.com (10.254.142.33)'
can't be established.
RSA key fingerprint is 1f:51:ae:28:bf:89:e9:d8:1f:25:5d:37:2d:7d:b8:ca:9f:f5:f1:6f.
Are you sure you want to continue connecting (yes/no)?
```

yes를 입력합니다.

3. 전송이 성공한 경우 다음과 유사한 응답이 표시됩니다.

```
Warning: Permanently added 'ec2-198-51-100-1.compute-1.amazonaws.com' (RSA)
to the list of known hosts.
private-key.pem                                 100%   480     24.4KB/s   00:00
```

이제 프라이빗 키를 소스 인스턴스로 전송했으므로 대상 인스턴스에 안전하게 연결하고 대상 인스턴스로 파일을 전송할 수 있습니다. 다음 단계로 진행하여 방법을 알아봅니다.

4단계: Lightsail Linux 인스턴스와 Unix 인스턴스 간에 파일을 안전하게 전송

다음 절차에서는 한 인스턴스(소스 인스턴스)에서 scp 명령을 실행하여 파일을 다른 인스턴스(대상 인스턴스)로 전송합니다.

scp를 사용하여 인스턴스 간에 파일을 전송

1. SSH를 이용해 소스 인스턴스에 연결합니다. 로컬 컴퓨터에서 터미널 프로그램을 사용하거나 Lightsail에서 브라우저 기반 SSH 클라이언트를 사용하여 연결할 수 있습니다. 자세한 내용은 Linux 인스턴스에 연결 단원을 참조하십시오.
2. 컴퓨터의 소스 인스턴스의 파일 위치와 대상 인스턴스의 대상 경로를 확인합니다. 다음 예제에서 프라이빗 키 파일의 이름은 *private-key.pem*이고, 인스턴스의 사용자 이름은 *ec2-user*이고, 인스턴스의 IPv4 주소는 *public-ipv4-address*이고, 인스턴스의 IPv6 주소는 *public-ipv6-address*입니다. *destination-path/*는 파일을 전송할 대상 인스턴스의 위치입니다.
   - (IPv4) 소스 인스턴스에서 대상 인스턴스로 파일을 전송하려면 소스 인스턴스에서 다음 명령을 입력합니다.

     ```
     scp -i /path/private-key.pem /path/my-file.txt ec2-user@public-ipv4-address:destination-path/
     ```

   - (IPv6) 소스 인스턴스에서 대상 인스턴스로 파일을 전송하려면 소스 인스턴스에서 다음 명령을 입력합니다. IPv6 주소는 이스케이프된([ ]) 대괄호(\)로 묶어야 합니다.

     ```
     scp -i /path/private-key.pem /path/my-file.txt ec2-user@\[public-ipv6-address\]:destination-path/
     ```

3. SSH를 사용하여 타겟 인스턴스에 아직 연결하지 않은 경우 다음과 같은 응답이 표시됩니다.

   ```
   The authenticity of host 'ec2-198-51-100-1.compute-1.amazonaws.com (10.254.142.33)'
   can't be established.
   RSA key fingerprint is 1f:51:ae:28:bf:89:e9:d8:1f:25:5d:37:2d:7d:b8:ca:9f:f5:f1:6f.
   Are you sure you want to continue connecting (yes/no)?
   ```

   yes를 입력합니다.
4. 전송이 성공한 경우 다음과 유사한 응답이 표시됩니다.

   ```
   Warning: Permanently added 'ec2-198-51-100-1.compute-1.amazonaws.com' (RSA)
   to the list of known hosts.
   my-file.txt                                   100%   480     24.4KB/s   00:00
   ```

Lightsail 다른 AWS 서비스와 VPC 피어링 통합

Amazon Lightsail는 Amazon EC2 및와 같은 집중된 AWS 서비스 세트를 AWS Identity and Access Management 사용하여 더 쉽게 시작할 수 있습니다. 그렇다고 해서 해당 서비스로 제한되는 것은 아닙니다.

VPC 피어링을 통해 Lightsail 리소스를 다른 AWS 서비스와 통합할 수 있습니다. VPC 피어링을 활성화한 후에는 피어링 연결을 통해 연결하려는 리소스가 필요한 인바운드 트래픽을 수락하는지 확인해야 합니다. 자세한 내용은 VPC 피어링을 사용하여 AWS 서비스에 Lightsail 리소스 연결을 참조하세요.

Amazon Simple Storage Service, Amazon CloudFront, Amazon DynamoDB와 같은 일부 AWS 리소스에서는 VPC 피어링을 활성화할 필요가 없습니다. 다른 AWS 서비스에 대해 자세히 알아보려면 아래 링크를 따르세요.

가상 머신(가상 프라이빗 서버)

Amazon EC2

Amazon Elastic Compute Cloud(Amazon EC2)는 클라우드에서 확장 가능한 컴퓨팅 용량을 제공하는 웹 서비스입니다. 개발자가 보다 쉽게 웹 규모 클라우드 컴퓨팅 작업을 할 수 있도록 설계되었습니다.

Amazon EC2를 사용하면 마찰을 최소화하면서 용량을 확보하고 구성할 수 있습니다. 이를 통해 컴퓨팅 리소스를 완벽하게 제어하고 Amazon의 검증된 컴퓨팅 환경에서 실행할 수 있습니다. Amazon EC2는 새 서버 인스턴스를 확보하고 부팅하는 데 필요한 시간을 몇 분으로 단축하므로 컴퓨팅 요구 사항의 변화에 따라 용량을 빠르게 확장하고 축소할 수 있습니다. Amazon EC2는 실제로 사용한 용량에 대해서만 비용을 지불할 수 있도록 하여 컴퓨팅의 경제성을 변화시킵니다. Amazon EC2는 개발자에게 장애 복원력이 뛰어난 애플리케이션을 구축하고 일반적인 장애 시나리오로부터 스스로를 격리할 수 있는 도구를 제공합니다.

Amazon EC2에 대해 자세히 알아보세요.

Amazon VPC

Amazon Virtual Private Cloud(Amazon VPC)를 사용하면 정의한 가상 네트워크에서 AWS 리소스를 시작할 수 있도록 AWS 클라우드의 논리적으로 격리된 섹션을 프로비저닝할 수 있습니다. 고유의 IP 주소 범위, 서브넷 생성, 라우팅 테이블 및 네트워크 게이트웨이 구성 선택 등 가상 네트워킹 환경을 완벽하게 제어할 수 있습니다.

Amazon VPC의 네트워크 구성을 손쉽게 사용자 지정할 수 있습니다. 예를 들어, 인터넷에 액세스하는 웹 서버에 대해 공용 페이싱 서브넷을 생성하고 인터넷 액세스가 없는 개인 페이싱 서브넷의

애플리케이션 서버나 데이터베이스 등의 백엔드 시스템을 배치할 수 있습니다. 보안 그룹 및 네트워크 액세스 제어 목록을 포함한 다중 보안 계층을 활용하여 각 서브넷에서 Amazon EC2 인스턴스에 대한 액세스를 제어하도록 지원할 수 있습니다.

또한, 고객의 회사 데이터 센터와 VPC 사이에 하드웨어 VPN(가상 프라이빗 네트워크) 연결을 생성하여 AWS 클라우드를 사내 데이터 센터의 확장으로 활용할 수 있습니다.

Amazon VPC에 대해 자세히 알아보세요.

서버리스 컴퓨팅

AWS Lambda

AWS Lambda 를 사용하면 서버를 프로비저닝하거나 관리하지 않고도 코드를 실행할 수 있습니다. 사용한 컴퓨팅 시간에 대해서만 요금을 지불하면 되고 코드가 실행되지 않을 때는 요금이 부과되지 않습니다. Lambda에서는 사실상 모든 유형의 애플리케이션이나 백엔드 서비스에 대한 코드를 별도의 관리 없이 실행할 수 있습니다. 코드를 업로드하기만 하면 고가용성을 유지한 채로 코드를 실행하고 확장하는 데 필요한 모든 것을 Lambda가 알아서 처리해 줍니다. 코드가 다른 AWS 서비스에서 자동으로 트리거되도록 설정하거나 어떤 웹 또는 모바일 앱에서도 코드를 직접 호출할 수 있습니다.

에 대해 자세히 알아봅니다 AWS Lambda.

Amazon API Gateway

Amazon API Gateway는 어떤 규모에서든 개발자가 API를 손쉽게 생성, 게시, 유지 관리, 모니터링 및 보호할 수 있도록 지원하는 완전관리형 서비스입니다. AWS Management Console에서 몇 번만 클릭하면 "프런트 도어"의 역할을 하는 API를 생성할 수 있습니다. 애플리케이션은 이 도어를 통해 백엔드 서비스로부터 데이터, 비즈니스 로직 또는 기능에 액세스할 수 있습니다. 여기에는 Amazon EC2에서 실행되는 워크로드, Lambda에서 실행되는 코드 또는 모든 웹 애플리케이션이 포함됩니다. Amazon API Gateway는 최대 수십만 개의 동시 API 호출 허용 및 처리에 관련된 모든 작업을 다룹니다. 여기에는 트래픽 관리, 권한 부여 및 액세스 제어, 모니터링, API 버전 관리가 포함됩니다. Amazon API Gateway에는 최소 수수료나 시작 비용이 없습니다. 수신한 API 호출과 전송한 데이터 양에 대해서만 비용을 지불하면 됩니다.

Amazon API Gateway에 대해 자세히 알아보세요.

데이터베이스 수

Amazon DynamoDB

Amazon DynamoDB는 규모와 관계없이 10밀리초 미만의 지연 시간이 일관되게 요구되는 모든 애플리케이션을 위한 빠르고 유연성이 뛰어난 NoSQL 데이터베이스 서비스입니다. 이 서비스는 완벽하게 관리되는 클라우드 데이터베이스로, 문서와 키-값 스토어 모델을 모두 지원합니다. 유연한 데이터 모델과 안정적인 성능 덕분에 모바일, 웹, 게임, 광고 기술, IoT 및 기타 여러 애플리케이션에 아주 적합합니다.

DynamoDB에 대해 자세히 알아보세요.

Amazon RDS

Amazon Relational Database Service(RDS)를 사용하여 클라우드에서 쉽게 관계형 데이터베이스를 설정, 운영 및 확장할 수 있습니다. 시간이 많이 소요되는 데이터베이스 관리 작업을 관리하는 동시에 비용 효율적이고 크기 조정이 가능한 용량을 제공하므로 사용자는 애플리케이션과 비즈니스에 집중할 수 있습니다. Amazon RDS는 Amazon Aurora, PostgreSQL, MySQL, MariaDB, Oracle 및 Microsoft SQL Server를 포함하여 선택할 수 있는 6개의 친숙한 데이터베이스 엔진을 제공합니다.

Amazon RDS에 대해 자세히 알아보세요.

Amazon Aurora

Amazon Aurora는 고성능 상용 데이터베이스의 속도와 가용성에 오픈 소스 데이터베이스의 간편성과 비용 효율성을 결합한 MySQL 호환 관계형 데이터베이스 엔진입니다. Aurora는 MySQL보다 최대 다섯 배 뛰어난 성능에 더하여 상용 데이터베이스의 보안성, 가용성 및 안정성을 1/10의 비용으로 제공합니다.

Amazon Aurora에 대해 자세히 알아보세요.

로드 밸런서

Elastic Load Balancing

Elastic Load Balancing은 수신 애플리케이션 트래픽을 여러 Amazon EC2 인스턴스에 자동으로 분산합니다. 따라서 애플리케이션의 내결함성을 달성하고, 애플리케이션 트래픽을 라우팅하는 데 필요한 로드 밸런싱 용량을 원활하게 제공할 수 있습니다.

Elastic Load Balancing은 두 가지 유형의 로드 밸런서를 지원합니다. 두 유형 모두 고가용성, 자동 조정 및 강력한 보안이 특징입니다. 여기에는 애플리케이션 또는 네트워크 수준 정보에 따라 트래픽을 라우팅하는 Classic Load Balancer와 요청 콘텐츠를 포함하는 고급 애플리케이션 수준 정보에 따라 트래픽을 라우팅하는 Application Load Balancer가 포함됩니다. Classic Load Balancer는 여러 Amazon EC2 인스턴스에 걸쳐 트래픽을 간편하게 로드 밸런싱하기에 이상적입니다. Application Load Balancer는 고급 라우팅 기능, 마이크로서비스 및 컨테이너 기반 아키텍처가 필요한 애플리케이션에 적합합니다. Application Load Balancer는 트래픽을 여러 서비스로 라우팅하거나 동일한 Amazon EC2 인스턴스의 여러 포트에 걸쳐 로드 밸런싱하는 기능을 제공합니다.

Elastic Load Balancing에 대해 자세히 알아보세요.

Application Load Balancer

Application Load Balancer는 애플리케이션 계층에서 작동하는 Elastic Load Balancing 서비스를 위한 로드 밸런싱 옵션으로, 이를 통해 하나 이상의 Amazon EC2 인스턴스에서 작동하는 여러 서비스 또는 컨테이너에 걸쳐 콘텐츠를 기반으로 라우팅 규칙을 정의할 수 있습니다.

Application Load Balancer에 대해 자세히 알아보세요.

빅 데이터

Amazon Kinesis 서비스

Amazon Kinesis 서비스를 사용하면 AWS 클라우드에서 실시간 스트리밍 데이터를 쉽게 사용할 수 있습니다. Amazon 서비스에는 대량의 스트리밍 데이터를 AWS로 쉽게 로드하는 Amazon Kinesis Data Firehose, 표준 SQL을 사용하여 스트리밍 데이터를 분석하는 Amazon Managed Service for Apache Flink, 스트리밍 데이터를 처리하거나 분석하는 맞춤형 애플리케이션을 구축하는 Amazon Kinesis Data Streams가 포함됩니다.

Amazon Kinesis 서비스에 대해 자세히 알아보세요.

Amazon EMR

Amazon EMR은 동적 확장 가능 Amazon EC2 인스턴스 전반에 걸쳐 대량의 데이터를 쉽고 빠르고 비용 효율적으로 처리할 수 있도록 지원하는 관리형 Hadoop 프레임워크를 제공합니다. 또한 Amazon EMR에서 Apache Spark, HBase, Presto 및 Flink와 같이 널리 사용되는 분산 프레임워크를 실행하고 Amazon S3, DynamoDB 등의 다른 AWS 데이터 스토어의 데이터와 상호 작용할 수 있습니다.

Amazon EMR은 로그 분석, 웹 인덱싱, 데이터 변환(ETL), 기계 학습, 금융 분석, 과학적 시뮬레이션 및 생물 정보학을 비롯하여 광범위한 빅 데이터 사용 사례를 안전하고 안정적으로 처리합니다.

Amazon EMR에 대해 자세히 알아보세요.

Amazon Redshift

Amazon Redshift는 속도가 빠른 페타바이트 규모의 완전 관리형 데이터 웨어하우스로, 간편하고 비용 효율적으로 모든 데이터를 기존 비즈니스 인텔리전스 도구를 사용하여 분석할 수 있게 해줍니다.

Amazon Redshift에 대해 자세히 알아보세요.

스토리지

Amazon Simple Storage Service(Amazon S3)

Amazon S3는 개발자와 IT 팀에 안전하고 내구성과 확장성이 뛰어난 클라우드 스토리지를 제공합니다. Amazon S3는 웹 어디에서나 원하는 양의 데이터를 저장하고 검색할 수 있는 간단한 웹 서비스 인터페이스를 갖춘 사용하기 쉬운 객체 스토리지입니다. Amazon S3에서는 사용한 스토리지에 대해서만 비용을 지불합니다. 최소 요금이나 설치 비용이 없습니다.

Amazon S3는 자주 액세스하는 데이터의 범용 스토리지용 Amazon S3 Standard, 자주 액세스하지 않지만 수명이 긴 데이터를 위한 Amazon S3 Standard - Infrequent Access(Standard - IA), 장기 아카이브를 위한 Amazon Glacier를 비롯해 여러 사용 사례에 맞게 설계된 다양한 스토리지 클래스를 제공합니다. 또한 Amazon S3는 수명 주기 전반에 걸쳐 데이터를 관리하기 위한 구성 가능한 수명 주기 정책을 제공합니다. 정책이 설정되면 애플리케이션 변경 없이 데이터가 자동으로 가장 적합한 스토리지 클래스로 마이그레이션됩니다.

Amazon S3는 단독으로 사용하거나 Amazon EC2 및 IAM 등의 다른 AWS 서비스는 물론 초기 또는 지속적인 데이터 수집을 위한 클라우드 데이터 마이그레이션 서비스 및 게이트웨이와 함께 사용할 수 있습니다. Amazon S3는 백업 및 복구, 니어라인 아카이브, 빅 데이터 분석, 재해 복구, 클라우드 애플리케이션, 콘텐츠 배포 등 다양한 사용 사례에 대해 비용 효율적인 객체 스토리지를 제공합니다.

Amazon S3에 대해 자세히 알아보세요.

Amazon Elastic Block Store(Amazon EBS)

Amazon EBS는 AWS 클라우드의 Amazon EC2 인스턴스에 사용할 영구적인 블록 스토리지 볼륨을 제공합니다. 각 Amazon EBS 볼륨은 가용 영역 내에 자동으로 복제되어 구성요소 장애로부터 보호해주고, 고가용성 및 내구성을 제공합니다. Amazon EBS 볼륨은 워크로드 실행에 필요한 지연 시간이 짧고 일관된 성능을 제공합니다. Amazon EBS를 사용하면 프로비저닝하는 것에 대해서만 적은 요금을 지불하면서 몇 분 내에 사용량을 늘리거나 줄일 수 있습니다.

Amazon EBS에 대해 자세히 알아보세요.

모니터링 및 경보

Amazon CloudWatch

Amazon CloudWatch는 AWS 클라우드 리소스 및 AWS에서 실행하는 애플리케이션을 모니터링하는 서비스입니다. CloudWatch를 사용하여 지표를 수집 및 추적하고, 로그 파일을 수집 및 모니터링하며, 경보를 설정하고, AWS 리소스 변경에 자동으로 대응할 수 있습니다. CloudWatch는 Amazon EC2 인스턴스, Amazon DynamoDB 테이블, Amazon RDS DB 인스턴스와 같은 AWS 리소스뿐만 아니라 애플리케이션 및 서비스에서 생성된 사용자 지정 지표와 애플리케이션이 생성하는 모든 로그 파일을 모니터링할 수 있습니다. CloudWatch를 사용하여 시스템 전체의 리소스 사용률, 애플리케이션 성능 및 운영 상태를 파악할 수 있습니다. 상황에 대처하고 애플리케이션을 원활하게 운영하는 데 이러한 정보를 사용할 수 있습니다.

Amazon CloudWatch에 대해 자세히 알아보세요.

애플리케이션 배포

AWS Elastic Beanstalk

AWS Elastic Beanstalk 는 Apache, Nginx, Passenger 및 IIS와 같은 친숙한 서버에서 Java, .NET, PHP, Node.js, Python, Ruby, Go 및 Docker로 개발된 웹 애플리케이션 및 서비스를 배포하고 확장하기 위한 easy-to-use 서비스입니다.

코드를 업로드하기만 하면 Elastic Beanstalk가 용량 프로비저닝, 로드 밸런싱, Auto Scaling부터 애플리케이션 상태 모니터링에 이르기까지 배포를 자동으로 처리합니다. 이뿐만 아니라 애플리케이션을 실행하는 데 필요한 AWS 리소스를 완벽하게 제어할 수 있으며 언제든지 기본 리소스에 액세스할 수 있습니다.

Elastic Beanstalk에 대해 자세히 알아보세요.

애플리케이션 컨테이너

Amazon Elastic Container Service(Amazon ECS)

고도의 확장성과 우수한 성능을 갖춘 컨테이너 관리 서비스인 Amazon ECS는 Docker 컨테이너를 지원하며, 관리 대상인 Amazon EC2 인스턴스 클러스터에서 손쉽게 애플리케이션을 실행할 수 있

습니다. Amazon ECS를 사용하면 자체 클러스터 관리 인프라를 설치, 운영 및 확장할 필요가 없습니다. 간단한 API 호출을 통해 Docker를 사용하는 애플리케이션을 시작 및 중지하고, 클러스터의 전체 상태를 쿼리하고, 보안 그룹, Elastic Load Balancing, Amazon EBS 볼륨 및 IAM 역할 등 여러 가지 익숙한 기능을 액세스할 수 있습니다. 리소스와 필요와 가용성 요구 사항에 따라 클러스터 전체에 컨테이너를 배치할 일정을 수립하는 데에도 Amazon ECS를 사용할 수 있습니다. 또한 자신의 스케줄러 또는 타사 스케줄러를 통합하여 해당 비즈니스 또는 애플리케이션에 따른 요구 사항을 충족할 수 있습니다.

Amazon ECS에 대해 자세히 알아보세요.

보안 및 사용자 로그인

AWS Identity and Access Management (IAM)

IAM은 사용자의 AWS 서비스와 리소스에 대한 액세스 권한을 안전하게 제어할 수 있게 해 줍니다. IAM을 사용하여 AWS 사용자 및 그룹을 만들고 관리하며 AWS 리소스에 대한 액세스를 허용 및 거부할 수 있습니다.

IAM에 대해 자세히 알아보세요.

Amazon Cognito 사용자 풀

Amazon Cognito를 사용하여 손쉽게 사용자 가입 정보를 추가하고 사용 중인 모바일 및 웹 앱에 로그인할 수 있습니다. 또한 Amazon Cognito에는 Facebook, Twitter, Amazon과 같은 사회적 자격 증명 공급자, SAML 자격 증명 솔루션 또는 자체 자격 증명 시스템을 통해 사용자를 인증할 수 있는 옵션이 있습니다. 뿐만 아니라 Amazon Cognito를 통해 사용자의 디바이스에 데이터를 로컬 저장할 수 있어 디바이스가 오프라인 상태일 때에도 애플리케이션이 작동하도록 할 수 있습니다. 또한 사용자의 디바이스 전반에 걸쳐 데이터를 동기화하여 앱 사용 환경이 디바이스에 상관없이 일관되도록 할 수 있습니다.

Amazon Cognito를 사용하면 사용자 관리, 인증 및 디바이스 간 동기화를 처리하는 솔루션의 구축, 보안 및 확장에 대해 걱정하는 대신 뛰어난 앱 경험을 만드는 데 집중할 수 있습니다.

Amazon Cognito에 대해 자세히 알아보세요.

소스 제어 및 애플리케이션 수명 주기 관리

AWS CodeCommit

AWS CodeCommit 는 기업이 안전하고 확장성이 뛰어난 프라이빗 Git 리포지토리를 쉽게 호스팅할 수 있도록 하는 완전 관리형 소스 제어 서비스입니다.는 자체 소스 제어 시스템을 운영하거나 인프라 확장에 대해 걱정할 필요가 없도록 AWS CodeCommit 합니다. AWS CodeCommit 를 사용하여 소스 코드에서 바이너리에 이르기까지 모든 것을 안전하게 저장할 수 있으며 기존 Git 도구와 원활하게 작동합니다.

AWS CodeCommit에 대해 자세히 알아보세요.

대기열 및 메시징

Amazon SQS

Amazon Simple Queue Service(Amazon SQS)는 빠르고 안정적이며 확장 가능한 완전관리형 메시지 대기열 서비스입니다. Amazon SQS를 사용하면 클라우드 애플리케이션의 구성 요소를 간단하고 비용 효율적으로 분리할 수 있습니다. Amazon SQS를 사용하면 데이터 손실 위험 또는 다른 서비스를 항상 가용 상태로 유지할 필요 없이 모든 데이터 볼륨을 전송할 수 있습니다. Amazon SQS에는 높은 처리량과 최소 1회 처리 기능을 갖춘 표준 대기열과 선입선출(FIFO) 전송 및 정확히 1회 처리를 제공하는 FIFO 대기열이 포함되어 있습니다.

Amazon SQS를 통해 고가용성 메시징 클러스터를 운영하고 확장하는 관리적 부담에서 벗어나고 사용한 부분에 대해서만 저렴한 비용을 지불합니다.

Amazon SQS에 대해 자세히 알아보세요.

Amazon SNS

Amazon Simple Notification Service(SNS)는 빠르고 유연한 완전관리형 푸시 알림 서비스로서, 이 서비스를 사용하면 개별 메시지를 전송하거나 대규모의 수신자에게 메시지를 전송할 수 있습니다. Amazon SNS를 사용하면 간편하고 비용 효과적으로 모바일 디바이스 사용자와 이메일 수신자에게 푸시 알림을 보내거나 다른 배포된 서비스에도 메시지를 보낼 수 있습니다.

또한 Amazon SNS를 통해 APNS(Apple Push Notification Service), GCM(Google Cloud Messaging), Fire OS 및 Windows 디바이스는 물론 Baidu Cloud Push를 통해 중국의 Android 디바이스에도 알림을 전송할 수 있습니다. Amazon SNS를 사용하여 전 세계 모바일 디바이스 사용자에게 SMS 메시지를 전송할 수 있습니다.

이 엔드포인트뿐 아니라 Amazon SNS에서 Amazon SQS, AWS Lambda 함수 또는 모든 HTTP 엔드포인트로 메시지를 전송할 수 있습니다.

Amazon SNS에 대해 자세히 알아보세요.

Amazon SES

Amazon Simple Email Service(Amazon SES) 비용 효과적인 이메일 서비스로서, Amazon.com이 자체 고객층을 지원하기 위해 개발한 안정적이고 확장 가능 인프라에 구축되었습니다. Amazon SES를 사용하면 필수적인 최소 약정 없이 이메일을 보내고 받을 수 있습니다. 사용하는 것에 대해서만 사용할 때 지불합니다.

Amazon SES에 대해 자세히 알아보세요.

워크플로

Amazon Simple Workflow Service(Amazon SWF)

Amazon SWF는 개발자가 병렬적으로 또는 순차적으로 진행되는 백그라운드 작업을 빌드, 실행 및 규모 조정할 수 있게 해줍니다. Amazon SWF는 클라우드에서 완전 관리형 상태 추적기 및 작업 조정자의 기능을 한다고 볼 수 있습니다.

앱에서 절차를 완료하는 데 500밀리초가 넘게 걸리는 경우 처리 상태를 추적할 필요가 있으며, 작업이 실패하는 경우에는 복구하거나 재시도할 필요가 있습니다. Amazon SWF가 도움이 될 수 있습니다.

Amazon SWF에 대해 자세히 알아보세요.

스트리밍 애플리케이션

Amazon 앱Stream

Amazon AppStream을 사용하여 어느 디바이스에나 Windows 애플리케이션을 제공할 수 있습니다.

Amazon AppStream을 사용하면 코드 수정 없이 클라우드에서 기존의 Windows 애플리케이션을 스트리밍함으로써 더 많은 디바이스, 더 많은 사용자에게 스트리밍 서비스를 제공할 수 있습니다. Amazon AppStream을 사용하면 애플리케이션이 AWS 인프라에 배포 및 렌더링되고 출력이 개인용 컴퓨터, 태블릿, 휴대폰과 같은 대량 시장 디바이스로 스트리밍됩니다. 애플리케이션이 클라우드에서 실행되므로 고객이 어떤 디바이스를 사용하든지 관계없이 방대한 컴퓨팅 및 스토리지 요구

를 처리할 수 있도록 애플리케이션을 확장할 수 있습니다. Amazon AppStream은 클라우드에서 애플리케이션을 스트리밍하기 위한 SDK를 제공합니다. 자체 사용자 지정 클라이언트, 구독, 자격 증명 및 스토리지 솔루션을 Amazon AppStream과 통합하여 귀사의 요구를 충족하는 사용자 지정 스트리밍 솔루션을 빌드할 수 있습니다.

Amazon AppStream에 대해 자세히 알아보세요.

를 사용하여 Lightsail 리소스 생성 AWS CloudFormation

Amazon Lightsail는 AWS 리소스 및 인프라를 생성하고 관리하는 데 소요되는 시간을 줄일 수 있도록 리소스를 모델링하고 설정하는 데 도움이 되는 AWS CloudFormation서비스와 통합됩니다. 원하는 모든 AWS 리소스(예: 인스턴스 및 디스크)를 설명하는 템플릿을 생성하고 해당 리소스를 CloudFormation 프로비저닝하고 구성합니다.

를 사용하면 템플릿을 재사용하여 Lightsail 리소스를 일관되고 반복적으로 설정할 CloudFormation수 있습니다. 리소스를 한 번 설명한 다음 여러 AWS 계정 및 리전에서 동일한 리소스를 반복적으로 프로비저닝합니다.

Lightsail 및 CloudFormation 템플릿

Lightsail 및 관련 서비스에 대한 리소스를 프로비저닝하고 구성하려면 CloudFormation 템플릿을 이해해야 합니다. 템플릿은 JSON 또는 YAML로 서식 지정된 텍스트 파일입니다. 이러한 템플릿은 CloudFormation 스택에서 프로비저닝하려는 리소스를 설명합니다. JSON 또는 YAML에 익숙하지 않은 경우 CloudFormation Designer를 사용하여 템플릿을 시작할 CloudFormation 수 있습니다. 자세한 내용은 AWS CloudFormation 사용 설명서의 CloudFormation Designer란 무엇입니까?를 참조하세요.

Lightsail는 AWS CloudFormation에서 인스턴스 및 디스크 생성을 지원합니다. 자세한 내용은 AWS CloudFormation 사용 설명서의 Lightsail 리소스 유형 참조를 참조하세요.

에 대해 자세히 알아보기 CloudFormation

에 대해 자세히 알아보려면 다음 리소스를 CloudFormation참조하세요.

- AWS CloudFormation
- AWS CloudFormation 사용 설명서
- CloudFormation API Reference
- AWS CloudFormation 명령줄 인터페이스 사용 설명서

앱 배포를 위한 Lightsail 리소스 살펴보기

다음 목록에는 Amazon Lightsail에 대한 추가 정보 중 Lightsail 사용 설명서에 게시되지 않은 정보에 대한 링크가 포함되어 있습니다.

내용

- 블로그
- 자습서
- 동영상

블로그

- Monitoring the health of Amazon Lightsail instances with Datadog

  2022년 3월 30일 - Datadog를 사용한 Lightsail 워크로드 모니터링이 어떻게 애플리케이션 성능을 보장하고 비용을 제어하는 데 도움이 되는지 알아보세요.
- How to set up Galaxy for research on AWS using Amazon Lightsail

  2022년 1월 13일 - 과학 워크플로, 데이터 통합 및 디지털 보존 플랫폼인 Galaxy를 Lightsail에 배포합니다.
- 브라우저에 URL을 입력하면 어떻게 됩니까

  2021년 8월 26일 - 브라우저에 URL을 입력하고 Enter 키를 누르면 어떻게 될까요?
- Amazon Lightsail에서 메모리 사용량 모니터링

  2021년 6월 14일 - 모니터링, 경보 및 알림을 위해 Lightsail 인스턴스에서 메모리 사용량을 Amazon CloudWatch로 전송하도록 구성합니다.
- 을 사용하여 컨테이너화된 ASP.NET 웹 앱의 원활한 호스팅Amazon Lightsail

  2021년 6월 10일 - PostgreSQL 데이터베이스에 연결하는 컨테이너화된 ASP.NET 웹 애플리케이션을 가져와서 Lightsail에 배포하는 방법입니다.
- Amazon Lightsail 컨테이너를 사용하여 WordPress 웹 사이트 시작

  2021년 4월 5일 - Lightsail 컨테이너 및 Lightsail 데이터베이스를 사용하여 WordPress 웹 사이트를 시작합니다.
- Lightsail 컨테이너: 클라우드에서 컨테이너를 쉽게 실행할 수 있는 방법

2020년 11월 13일 - Lightsail에 컨테이너기반 워크로드를 배포합니다.

- Migrating web services from Amazon Lightsail to Amazon EC2

  2020년 10월 16일 - Amazon EC2에 프로덕션 환경을 설정하고 웹 서비스를 Lightsail의 해당 환경으로 마이그레이션합니다.

- Amazon Lightsail 인스턴스에서 실행할 Graylog 서버 구축

  2020년 7월 28일 - Lightsail에 Graylog 서버를 구축하는 방법입니다.

- Lightsail 콘텐츠 전송 네트워크를 사용한 웹 사이트 성능 향상

  2020년 7월 23일 - WordPress 외에도 표준 웹 서버와 함께 작동하는 Lightsail 배포를 구성합니다.

- Amazon Lightsail 인스턴스에서 사전 예방적으로 시스템 성능 모니터링

  2020년 6월 4일 - 시스템 성능 문제가 사용자에게 영향을 미치기 전에 방지할 수 있도록 버스트 가능한 용량 알림을 구성합니다.

- 새로운 Lightsail 방화벽 기능으로 사이트 보안 강화

  2020년 5월 7일 - SSH를 사용한 원격 액세스를 단일 소스 IP 주소로 제한합니다.

- Using CodeDeploy and CodePipeline to deploy applications to Amazon Lightsail

  2020년 4월 23일 - Lightsail이 CodeDeploy 및 CodePipeline와 작동하도록 구성하여 변경 사항을 GitHub에 푸시할 때마다 애플리케이션을 자동으로 배포(또는 업데이트)합니다.

- 에서 로드 밸런서 사용Amazon Lightsail

  2020년 4월 21일 - Amazon Lightsail 로드 밸런서를 사용하여 간단한 Node.js 웹 애플리케이션을 로드 밸런싱하는 방법입니다.

- Ghost를 사용하여 Amazon Lightsail에 사진 일기 만들기

  2020년 3월 23일 - Lightsail에서 Ghost를 사용하여 사진 일기를 시작합니다.

- Amazon Lightsail 데이터베이스 팁 및 요령

  2020년 3월 23일 - Amazon Relational Database Service(RDS)에서 제공하는 고급 기능을 사용합니다.

- Configuring and using monitoring and Notifications

  2020년 2월 27일 - 알림 연락처 생성, 새 경보 생성 및 리소스 모니터링을 통한 알림 테스트.

- Deploying a highly-available WordPress site on Amazon Lightsail, Part 1: Implementing a highly-available Lightsail database with WordPress

  2019년 10월 22일 - Lightsail에 고가용성 WordPress 사이트 구축, 파트 1입니다.
- Deploying a highly-available WordPress site on Amazon Lightsail, Part 2: Using Amazon S3 with WordPress to securely deliver media files

  2019년 10월 31일 - Lightsail에 고가용성 WordPress 사이트 구축, 파트 2입니다.
- Deploying a highly-available WordPress site on Amazon Lightsail, Part 3: Increasing security and performance using Amazon CloudFront

  2019년 11월 7일 - Lightsail에 고가용성 WordPress 사이트 구축, 파트 3.
- Deploying a highly-available WordPress site on Amazon Lightsail, Part 4: Increasing performance and scalability with a Lightsail load balancer

  2019년 11월 14일 - Lightsail에 고가용성 WordPress 사이트 구축, 파트 4.
- Building a pocket platform-as-a-service with Amazon Lightsail

  2019년 10월 8일 - Lightsail에 포켓 플랫폼을 구축합니다.
- Deploying a Nginx-based HTTP/HTTPS load balancer with Amazon Lightsail

  2019년 7월 8일 - Lightsail 인스턴스 내부에 NGINX 기반 로드 밸런서를 설정합니다.
- AWS 클라우드를 처음 사용하시나요? Amazon Lightsail이 도움이 될 수 있습니다.

  2019년 3월 27일 - Amazon Lightsail 시작하기.
- New – Managed databases for Amazon Lightsail

  2018년 10월 16일 - 몇 번의 클릭만으로 관리형 데이터베이스를 생성합니다.
- Amazon Lightsail 업데이트: 더 많은 인스턴스 크기 및 가격 인하

  2018년 8월 23일 -Lightsail 인스턴스 개요입니다.
- Amazon Lightsail: The power of AWS, the simplicity of a VPS

  2016년 11월 30일 - Lightsail 시작 안내입니다.

튜토리얼

탑 5 실습용 자습서:

1. 로드 밸런서 WordPress 웹 사이트 만들기

   2021년 9월 8일 - Lightsail을 사용하여 고가용성 WordPress 웹 사이트를 시작합니다.

2. 을 사용하여 WordPress 웹 사이트 마이그레이션 및 관리Amazon Lightsail

   2021년 2월 22일 – Seahorse 소프트웨어를 사용하여 WordPress 웹 사이트의 클론을 Lightsail에서 시작합니다.

3. Linux 가상 머신 시작

   2020년 9월 11일 - Lightsail을 사용하여 Linux 인스턴스를 시작, 구성 및 연결합니다.

4. Windows 가상 머신 시작

   2020년 9월 11일 - Lightsail을 사용하여 Windows 인스턴스를 시작, 구성 및 연결합니다.

5. 에서 cPanel 및 WHM 인스턴스 시작Amazon Lightsail

   2020년 7월 27일 - 이 자습서에서는 cPanel 및 WHM 인스턴스를 Lightsail에 설치하고 실행한 후 수행할 수 있는 몇 가지 단계를 안내합니다.

- 에 Magento를 설정 및 구성하는 방법Amazon Lightsail

  2021년 8월 11일 - 전자 상거래 사이트를 구축하고 실행합니다.

- WordPress 사이트를 객체 스토리지 버킷에 연결하는 방법

  2021년 7월 14일 - Lightsail에 WordPress 사이트를 설정하고 웹 사이트를 Lightsail 버킷에 연결합니다.

- 객체 스토리지 버킷 생성

  2021년 7월 14일 - Amazon Lightsail에 객체 스토리지 버킷을 생성합니다.

- Amazon Lightsail 버컷 및 배포에 WordPress 웹 사이트 연결

  2021년 7월 14일 - Lightsail 콘텐츠 전송 네트워크(CDN) 배포의 오리진으로 Lightsail 버킷을 구성합니다.

- Plesk 설정 및 구성 방법

  2021년 4월 22일 - Lightsail에 Plesk 호스팅 스택을 구축하고 실행합니다.

- How to Setup a Prestashop e–commerce site

2021년 4월 1일 - Bitnami 블루프린트의 인증을 받은 PrestaShop을 사용하여 Lightsail 인스턴스를 시작 및 구성합니다.

- How to Use Amazon EFS with Amazon Lightsail

  2021년 3월 15일 - VPC 피어링을 사용하여 Lightsail 인스턴스에서 Amazon EFS 파일 시스템을 생성하고 연결합니다.

- Nginx 리버스 프록시를 설정하는 방법

  2021년 2월 10일 - Lightsail 컨테이너를 사용하여 Nginx 리버스 프록시를 설정합니다.

- 플라스크 pp를 제공하는 방법

  2021년 2월 3일 -Lightsail 컨테이너를 사용하여 Flask 애플리케이션을 제공하는 방법에 대해 알아봅니다.

- 를 사용하여 컨테이너 이미지 생성, 푸시 및 배포Amazon Lightsail

  2020년 11월 11일 - Dockerfile을 사용하여 로컬 시스템에 컨테이너 이미지를 생성합니다.

- Drupal 웹 사이트 구축

  2020년 9월 11일 - Lightsail에 프로덕션용 Drupal 웹 사이트를 배포 및 호스팅합니다.

- LAMP 스택 웹 앱 구축

  2020년 9월 9일 - Lightsail에 고가용성 PHP 웹 애플리케이션을 시작 및 실행합니다.

- Configure your WordPress instance to work with your distribution

  2020년 7월 16일 - Lightsail 배포와 함께 사용할 수 있도록 WordPress 인스턴스를 구성합니다.

- WordPress 웹 사이트 시작

  2020년 3월 23일 - Lightsail 가상 머신에 설치된 WordPress를 사용하여 웹 사이트를 구축하고 실행합니다.

- NET 애플리케이션 호스팅

  2020년 3월 20일 - Lightsail을 사용하여 .NET 애플리케이션을 구축하고 배포합니다.

- Map your domain at Amazon Route 53 to your Lightsail resources

  example.com과 같은 도메인에 대한 트래픽을 Lightsail 리소스로 라우팅합니다.

동영상

- Amazon Lightsail 자습서: Django 앱 배포

  2021년 7월 14일 - 이 자습서에서는 Django 애플리케이션을 생성합니다.

- Amazon Lightsail 자습서: Flask 앱 배포

  2021년 7월 14일 - 이 자습서에서는 Flask 애플리케이션을 생성합니다.

- Amazon Lightsail 자습서: NGINX 리버스 프록시 배포

  2021년 7월 14일 - Flask 애플리케이션을 생성하고 Docker 컨테이너를 구축하고 Lightsail에 컨테이너 서비스를 생성한 다음 애플리케이션을 배포합니다.

- Amazon Lightsail 자습서: 전자 상거래 사이트 배포

  2021년 7월 14일 - Bitnami 블루프린트의 인증을 받은 PrestaShop을 사용하여 Lightsail 인스턴스를 시직하고 구성합니다.

- 에 컨테이너화된 애플리케이션 배포Amazon Lightsail

  2020년 12월 29일 - Lightsail에 컨테이너화된 애플리케이션을 배포하는 방법에 대해 알아봅니다.

- Amazon Lightsail 자습서: Drupal 웹 사이트 구축

  2020년 8월 31일 - Drupal 인스턴스를 시작하고 구성합니다.

- Amazon Lightsail 자습서: LAMP 스택 앱 배포

  2020년 8월 31일 - LAMP(Linux Apache MySQL PHP) 스택 애플리케이션을 단일 Lightsail 인스턴스에 배포합니다.

- Amazon Lightsail 자습서: Linux 인스턴스 시작

  2020년 8월 31일 - Linux 인스턴스를 시작하는 방법에 대해 알아봅니다.

- Amazon Lightsail 자습서: Windows 인스턴스 시작

  2020년 8월 31일 - Windows 인스턴스를 시작하는 방법에 대해 알아봅니다.

- Amazon Lightsail 자습서: 전용 Minecraft 서버 실행

  2020년 8월 31일 - 전용 Minecraft 서버를 설정하는 방법에 대해 알아봅니다.

- Amazon Lightsail 자습서 소개

  2020년 8월 31일 - Lightsail을 사용하여 지금 클라우드 여정을 시작합니다.

- Amazon Lightsail: 를 가장 쉽게 시작하는 방법AWS

  2020년 3월 20일 - Lightsail은 AWS를 가장 쉽게 시작하는 방법입니다. 가상 서버, 스토리지, 데이터베이스 및 네트워킹뿐만 아니라 비용 효과적인 월 요금제를 제공합니다.

- 에 Plesk 인스턴스 구성Amazon Lightsail

  2019년 3월 27일 - Lightsail에 Plesk 인스턴스를 구성하는 방법에 대해 알아봅니다.

- 에 WordPress 멀티사이트 구성Amazon Lightsail

  2019년 1월 15일 - Lightsail에 WordPress 멀티사이트 인스턴스를 구성하는 방법에 대해 알아봅니다.

- 관리Lightsail

  2018년 10월 9일 - Lightsail 주요 기능을 간단히 살펴봅니다.

- 에 MEAN 스택 앱 배포Amazon Lightsail

  2018년 6월 5일 - Lightsail의 MEAN 블루프린트를 사용하여 사용자 지정 애플리케이션을 클라우드에 배포합니다.

- 에 WordPress 인스턴스 배포Amazon Lightsail

  2018년 6월 5일 - Lightsail에 WordPress 인스턴스를 배포합니다.

Lightsail 세부 결제 및 사용량 보기

Amazon Lightsail의 결제는 Amazon Web Services(AWS) 결제를 통해 처리됩니다. Lightsail 청구서를 보려면 AWS 결제 및 비용 관리 대시보드로 이동하거나 Lightsail 콘솔 상단 탐색 모음에서 청구를 선택합니다. 요금에 대한 자세한 내용은 Lightsail 요금 페이지를 참조하세요.

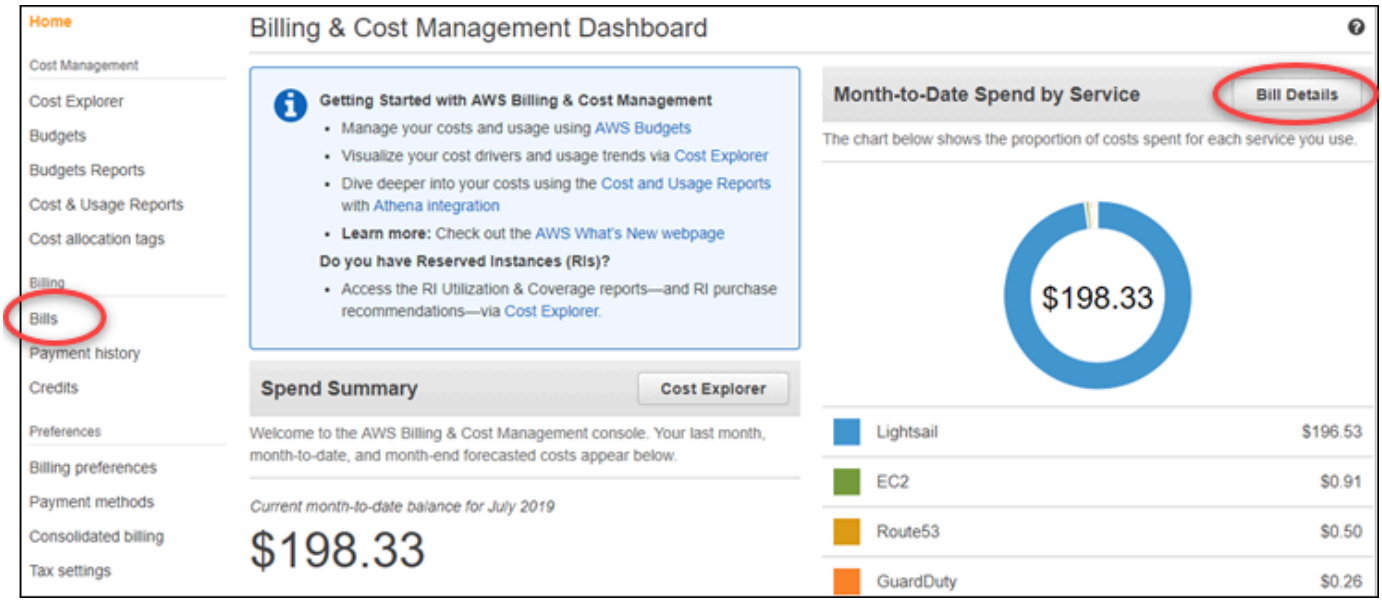
세부 Lightsail 청구서 보기

월별 Lightsail 청구서의 세부 분석을 보려면

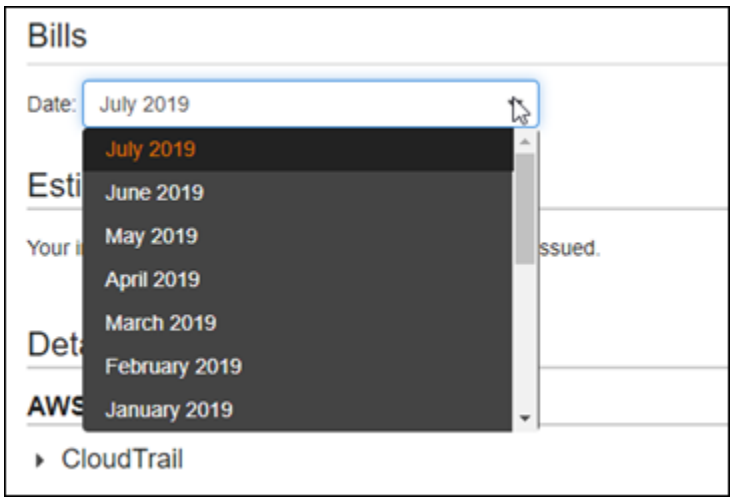
1. AWS 결제 및 비용 관리 대시보드에 로그인합니다.

   결제 대시보드 홈 페이지에 이번달 현재까지의 청구액 상위 수준 분석이 표시됩니다.

2. 대시보드 홈 페이지에서 청구서 세부 정보를 선택하거나, 왼쪽 탐색 창에서 청구서를 선택하여 자세한 월별 청구서를 봅니다.



3. 이번 달이 아닌 달을 선택하려면 날짜 드롭다운 메뉴를 선택합니다.



4. 청구서 페이지에서 아래로 스크롤하고 Lightsail 항목을 확장하여 리전별 세부 사용량을 확인합니다.

결제 사용 유형

다음 목록은 Lightsail 결제 및 사용 보고서에 표시되는 사용 유형을 설명합니다. 이 사용 유형을 통해 Lightsail 리소스의 월별 청구서에서 요금을 파악할 수 있습니다.

> **Note**
>
> 리전 코드를 지정하는 다음 사용 유형에 대해서는 이 설명서의 청구서의 리전 코드 섹션을 참조하여 해당하는 AWS 리전을 식별하세요.

- Amazon Lightsail Bundle:{Size}GB: 사용된 Linux 또는 Unix 인스턴스 플랜(시간). {Size}는 사용된 인스턴스 계획의 메모리 사양을 정의합니다. 예를 들어, 4GB 메모리가 지정되면 24 USD/월 Linux 또는 Unix 인스턴스 플랜으로 청구된 시간이 표시됩니다.
- Amazon Lightsail Bundle:{Size}GB(Windows): 사용된 Windows 인스턴스 플랜(시간). {Size}는 사용된 인스턴스 계획의 메모리 사양을 정의합니다. 예를 들어, 4GB 메모리가 지정되면 44 USD/월 Windows 인스턴스 플랜으로 청구된 시간이 표시됩니다.
- Amazon Lightsail MemoryOptimizedBundle:{Size}GB: 사용된 Linux 또는 Unix 메모리 최적화 인스턴스 계획(시간). {Size}는 사용된 메모리 최적화 인스턴스 계획의 메모리 사양을 정의합니다. 예를 들어 16GB 메모리를 지정하면 월 74 USD Linux 또는 Unix 메모리 최적화 인스턴스 플랜에 대해 청구된 시간이 표시됩니다.
- Amazon Lightsail MemoryOptimizedBundle:{Size}GB(Windows): 사용된 Windows 메모리 최적화 인스턴스 계획(시간). {Size}는 사용된 메모리 최적화 인스턴스 계획의 메모리 사양을 정의합니다. 예를 들어 16GB 메모리를 지정하면 월 134 USD의 Windows 메모리 최적화 인스턴스 플랜에 대해 청구된 시간이 표시됩니다.
- Amazon Lightsail RelationalDatabase:{Size}GB: 사용된 표준 데이터베이스 계획(시간). {Size}는 사용된 데이터베이스 계획의 메모리 사양을 정의합니다. 예를 들어, 4GB 메모리가 지정되면 $60 USD/월 표준 데이터베이스 플랜으로 청구된 시간이 표시됩니다.
- Amazon Lightsail RelationalDatabase:{Size}GB(고가용성): 사용된 고가용성 데이터베이스 플랜(시간). {Size}는 사용된 데이터베이스 계획의 메모리 사양을 정의합니다. 예를 들어, 4GB 메모리가 지정되면 $120 USD/월 고가용성 데이터베이스 플랜으로 청구된 시간이 표시됩니다.
- Amazon Lightsail Region-DiskUsage: 사용한 블록 스토리지 디스크의 양(월별 기가바이트 단위)
- Amazon Lightsail DNS-Queries: 월별 DNS 쿼리 수
- Amazon Lightsail Load Balancer: 사용한 로드 밸런서의 양(시간 단위)
- Amazon Lightsail Region-SnapshotUsage: 저장된 스냅샷 데이터의 양(월별 기가바이트 단위)
- Amazon Lightsail Region-UnusedStaticIP: 연결되지 않은 고정 IP의 양(시간 단위)
- Amazon Lightsail Region-TotalDataXfer-In-Bytes: 내부로 전송된 데이터의 총량(기가바이트 단위)
- Amazon Lightsail Region-TotalDataXfer-Out-Bytes: 외부로 전송된 데이터의 총량(기가바이트 단위)
- Amazon Lightsail Region-DataXfer-Out-Overage-Bytes: 사용한 인스턴스 또는 데이터베이스 플랜의 한도를 초과하여 인터넷 또는 퍼블릭 IP로 전송된 데이터의 양(기가바이트 단위)

청구서의 리전 코드

Lightsail 결제 및 사용 보고서는 코드와 약어를 사용합니다. 예를 들어, 사용 유형에서 리전은 다음 약어 중 하나로 바뀝니다.

- APN1: 아시아 태평양(도쿄)(ap-northeast-1)
- APN2: 아시아 태평양(서울)(ap-northeast-2)
- APS1: 아시아 태평양(싱가포르)(ap-southeast-1)
- APS2: 아시아 태평양(시드니)(ap-southeast-2)
- APS3: 아시아 태평양(뭄바이)(ap-south-1)
- APS4: 아시아 태평양(자카르타)(ap-southeast-3)
- CAN1: 캐나다(중부)(ca-central-1)
- EU: EU(아일랜드)(eu-west-1)
- EUC1: EU(프랑크푸르트)(eu-central-1)
- EUW2: EU(런던)(eu-west-2)
- EUW3: EU(파리)(eu-west-3)
- EUN1: EU(스톡홀름)(eu-north-1)
- USE1: 미국 동부(버지니아 북부)(us-east-1)
- USE2: 미국 동부(오하이오)(us-east-2)
- USW2: 미국 서부(오레곤)(us-west-2)

Lightsail에 대한 FAQ의 답변을 얻습니다.

이 섹션에서는 다음 범주로 구성된 Lightsail과 관련된 일반적인 질문과 답변을 다룹니다.

주제

- Lightsail 및 글로벌 가용성에 대해 알아보기
- 결제 및 계정 관리
- Lightsail에서의 데이터 전송
- 블록 스토리지(디스크)
- 인증서
- 연락처 및 모니터링 알림
- 컨테이너 서비스 수
- 콘텐츠 전송 네트워크 배포
- 데이터베이스 수
- 도메인
- Amazon Elastic Compute Cloud(Amazon EC2)로 Lightsail 리소스 내보내기
- 인스턴스
- 로드 밸런서
- 수동 및 자동 스냅샷
- 리소스 상태 지표 및 경보
- 네트워킹
- 객체 스토리지 및 버킷
- Lightsail의 태그

각 범주에 제공된 링크를 따라 Lightsail에 대해 자주 묻는 질문에 대한 자세한 답변을 찾아보세요.

Lightsail 및 글로벌 가용성에 대해 알아보기

Amazon Lightsail이란 무엇인가요?

Amazon Lightsail는 개발자, 중소기업, 학생 및 클라우드에서 웹 사이트와 웹 애플리케이션을 구축하고 호스팅하는 솔루션이 필요한 기타 사용자를 AWS 위해를 시작하는 가장 쉬운 방법입니다.는 개발자에

게 컴퓨팅, 스토리지 및 네트워킹 용량을 Lightsail 제공합니다. Lightsail 에는 가상 머신, 컨테이너, 데이터베이스, CDN, 로드 밸런서, DNS 관리 등 프로젝트를 빠르게 시작하는 데 필요한 모든 것이 포함되어 있으며, 저렴하고 예측 가능한 월별 가격으로 제공됩니다.

Lightsail으로 할 수 있는 작업은 무엇입니까?

애플리케이션을 쉽게 배포하고 관리할 수 있는 모든 사항이 포함된 사전 구성된 가상 프라이빗 서버(인스턴스)를 생성하거나 기본 인프라와 운영 체제의 보안 및 상태를 Lightsail에서 관리하는 데이터베이스를 생성할 수 있습니다. Lightsail은 몇 개의 인스턴스가 필요한 프로젝트 및 간단한 관리 인터페이스를 선호하는 개발자에 가장 적합합니다. Lightsail의 일반 사용 사례로는 웹 사이트, 웹 애플리케이션, 비즈니스 소프트웨어, 블로그, 전자 상거래 사이트 등의 실행이 있습니다. 프로젝트가 성장함에 따라 인스턴스와 함께 로드 밸런서 및 연결된 블록 스토리지를 사용하여 중복성과 가동 시간을 늘리고 수십 개의 다른 AWS 서비스에 액세스하여 새로운 기능을 추가할 수 있습니다.

Lightsail이 API를 제공하나요?

예. Lightsail 콘솔에서 수행하는 모든 작업은 공개적으로 사용 가능한 API의 지원을 받습니다. Lightsail CLI 및 API의 설치 및 사용 방법을 알아봅니다.

Lightsail에 가입해야 하나요?

Lightsail 사용을 시작하려면 시작하기를 선택하고 로그인하세요. Amazon Web Services 계정을 사용하여 Lightsail에 액세스합니다. 아직 계정이 없는 경우, 계정을 생성하라는 메시지가 표시됩니다.

어떤에서 AWS 리전 를 Lightsail 사용할 수 있나요?

Lightsail은 전 세계 다양한 리전에서 사용할 수 있습니다. 어느 리전을 사용할 수 있는지에 대한 자세한 내용은 Lightsail의 리전 및 가용 영역 섹션을 참조하세요.

가용 영역이란?

가용 영역은 물리적으로 구분된 독립 인프라에서 실행되고 높은 안정성을 보장하도록 설계된 데이터 센터의 모음입니다. 발전기 및 냉각 장비 등에 발생하는 일반적인 장애 사항은 가용 영역 간에 공유되지 않습니다. 또한, 가용 영역은 물리적으로 분리되어 있으므로 화재, 토네이도 또는 홍수와 같이 특별 재해가 발생할 경우에도 하나의 가용 영역만 영향을 받을 수 있습니다.

Lightsail 서비스 할당량이란 무엇인가요?

늘릴 수 있는 할당량을 포함한 최신 Lightsail 서비스 할당량의 경우, AWS 일반 참조의 Lightsail 서비스 할당량을 참조하세요. 서비스 할당량을 늘리려면 지원로 사례를 엽니다.

어떻게 더 많은 도움을 구할 수 있습니까?

Lightsail의 상황별 도움말(Help) 패널은 콘솔에서 작업하는 데 즉시 활용할 수 있는 유용한 팁을 제공합니다. 도움말 패널을 열려면 Lightsail 콘솔의 오른쪽 상단 모서리에서 도움말 패널 아이콘 ⓘ를 선택합니다. Lightsail 콘솔에서 시작 안내서, 개요 및 방법 관련 항목으로 구성된 라이브러리에 액세스할 수도 있습니다. API를 사용하려는 경우 Lightsail 또는 AWS CLI Lightsail 에는 지원되는 모든 프로그래밍 언어에 대한 전체 API 참조가 있습니다. Lightsail 지원 리소스를 사용할 수도 있습니다.

계정이나 결제와 관련된 문제가 있는 경우 지원에 온라인으로 문의하세요. Lightsail 계정을 통해 무료로 상시 액세스할 수 있습니다.

Lightsail 사용 방법에 관한 일반적인 질문 사항은 Lightsail 설명서와 지원 포럼을 검색하세요.

또한는 개별 요구 사항을 충족할 수 있는 다양한 유료 플랜을 지원 제공합니다.

결제 및 계정 관리

> **Note**
>
> Lightsail의 데이터 전송에 대한 주제가 Lightsail에서의 데이터 전송 섹션으로 재배치되었습니다.

Lightsail 요금제에 따른 가격은 얼마나 되나요?

Lightsail 플랜은 온디맨드 시간 요율에 따라 요금이 청구되므로 사용하는 만큼만 결제하면 됩니다. 고객이 이용하는 모든 Lightsail 플랜에 대해 월간 최대 요금까지 고정된 시간당 가격으로 계산해 요금이 청구됩니다. 가장 저렴한 Lightsail 플랜은 시간당 0.0067 USD(월 5 USD)부터 시작합니다. Windows Server 라이선스가 포함된 Lightsail 플랜은 시간당 0.0127 USD(월 9.50 USD)부터 시작합니다.

어떤 요금제에 가입한 경우 언제 요금이 청구됩니까?

Lightsail 인스턴스 및 관리형 데이터베이스는 삭제될 때까지 요금이 발생합니다. 이러한 리소스는 중지 상태인 경우에도 요금이 발생합니다. 월말 이전에 Lightsail 인스턴스 또는 관리형 데이터베이스를 삭제한 경우 해당 월의 Lightsail 인스턴스 또는 관리형 데이터베이스를 사용한 총 시간을 기준으로 비례 할당으로 계산된 요금만 청구됩니다. 예를 들어, 가장 요금이 낮은 Lightsail 인스턴스 플랜을 한 달에 100시간 동안 사용하는 경우 46센트(100*0.0046)가 부과됩니다.

Lightsail 인스턴스를 무료로 사용해 볼 수 있나요?

예. 기존 고객이든 신규 AWS 고객이든 상관없이 5 USD Lightsail 플랜을 750시간 동안 무료로 사용할 수 있습니다. 또한 9.50 USD Windows 플랜을 사용하면 Windows Server 라이선스가 포함된 Lightsail 플랜을 무료로 사용해 볼 수 있습니다. 원하는 만큼 많은 인스턴스에서 750시간의 주어진 무료 시간 동안 사용할 수 있습니다. 예를 들어, 한 달 내내 Lightsail 인스턴스 하나를 실행하거나 10개의 Lightsail 인스턴스를 75시간 동안 실행할 수 있습니다. 무료 평가판 혜택은 Lightsail을 사용하기 위해 등록한 시점으로부터 최초 역월 내에 사용하는 경우에만 적용됩니다. 계정이 조직에 연결된 경우(AWS Organizations 아래) 조직 내 하나의 계정만 AWS 프리 티어 제안의 이점을 누릴 수 있습니다.

인스턴스 플랜에는 데이터 전송 허용량이 포함됩니다. 인스턴스의 데이터 송수신은 모두 데이터 전송 허용량 계산에 포함됩니다. 데이터 전송 허용량을 초과하면 무료 평가 기간 내의 인스턴스를 포함한 인스턴스에는 외부로 송신되는 초과 데이터에 대해서만 요금이 부과됩니다. 데이터 전송 요금에 대한 자세한 내용은 데이터 전송 비용은 얼마나 됩니까? 섹션을 참조하세요.

> **Note**
>
> AWS 프리 티어의 일부로 일부 인스턴스 번들에서 Amazon Lightsail를 무료로 시작할 수 있습니다. 자세한 내용은 Amazon Lightsail 요금 페이지의 AWS 프리 티어를 참조하세요.

Lightsail 무료 체험판은 언제 시작되나요?

Lightsail 무료 평가판 혜택은 첫 번째 무료 평가판 대상 리소스가 출시될 때 시작됩니다.

인스턴스 및 데이터베이스에 대한 연장된 90일 무료 평가판은 일부 플랜(번들)에만 적용됩니다. 이 제안은 2021년 7월 8일 또는 그 이후에 사용을 시작한 신규 Lightsail 또는 기존 AWS 계정에 적용됩니다. 자세한 내용은 Lightsail 요금 페이지를 참조하세요.

Lightsail 관리형 데이터베이스 비용은 어떻게 되나요?

Lightsail 관리형 데이터베이스는 4개의 플랜 규모로 제공되며, 요금은 40GB의 SSD 스토리지와 100GB의 데이터 전송 허용량이 있는 1GB RAM 데이터베이스 인스턴스에 대해 최저 $15 USD/월입니다. 고가용성 플랜은 중복성을 위해 다른 가용 영역에 추가 데이터베이스 인스턴스와 스토리지 디스크를 실행하므로 해당 플랜 요금은 표준 플랜 요금의 두 배입니다.

Lightsail 관리형 데이터베이스를 무료로 사용해 볼 수 있나요?

예! 새 Lightsail 고객은 1개월 동안 $15 USD Lightsail 플랜을 무료로 이용할 수 있습니다.

Lightsail 블록 스토리지의 요금은 어떻게 되나요?

Lightsail 블록 스토리지의 요금은 GB당 $0.10 USD/월입니다.

Lightsail 로드 밸런서 요금은 어떻게 되나요?

Lightsail 로드 밸런서 요금은 $18 USD/월입니다.

인증서 관리의 비용은 어떻게 됩니까?

Lightsail 로드 밸런서를 사용하는 경우 Lightsail 인증서 및 인증서 관리는 무료입니다.

Lightsail 고정 IPv4 주소의 사용 요금은 얼마인가요?

고정 IP 주소가 Lightsail 인스턴스에 연결되어 있을 때에는 관련 비용이 발생하지 않습니다. 정적 IP는 IPv6 전용 인스턴스에 연결할 수 없습니다. IPv4 주소는 한정된 리소스이고 Lightsail은 퍼블릭 IP의 효율적 사용을 지원하고자 최선을 다하기 때문에, 1시간 이상 인스턴스에 연결되지 않는 고정 IP에 대해서는 0.005 USD/시간의 소액 요금을 부과합니다.

Lightsail 도메인 비용은 어떻게 되나요?

링크로 연결된 .pdf 파일에 나열된 요금은 2021년 12월 22일 기준으로 신규 도메인 이름 등록, 기존 도메인 이름 등록 갱신에 적용됩니다. 모든 요금에는 DNS 영역과 개인 정보 보호가 포함됩니다. 도메인 등록 요금에 대한 자세한 내용은 Amazon Route 53 도메인 등록 요금과 도메인 등록을 참조하세요.

Lightsail DNS 관리 비용은 얼마나 되나요?

Lightsail 내에서 DNS 관리 요금은 무료입니다. 최대 6개의 DNS 영역과 각 DNS 영역에 대해 원하는 만큼 많은 레코드를 만들 수 있습니다. 또한, 이러한 영역에 대해 매달 300만 회의 DNS 쿼리가 허용됩니다. 처음 주어진 월 300만 개의 쿼리 횟수를 초과하면 DNS 쿼리 100만 개당 $0.40 USD의 요금이 청구됩니다.

Lightsail 스냅샷 비용은 얼마나 되나요?

Lightsail 스냅샷(수동 및 자동) 비용은 월 $0.05 USD/GB입니다. 즉, 28GB의 공간을 사용하는 인스턴스의 스냅샷을 생성하고 한 달 동안 보관하면 해당 월에 $1.40 USD의 요금이 청구됩니다.

같은 인스턴스의 스냅샷을 연속으로 여러 개 생성하면 Lightsail이 스냅샷 요금을 자동으로 최적화합니다. 즉, 새 스냅샷을 생성할 때마다 변경된 데이터 부분에 대해서만 요금이 부과됩니다. 위 예의 인스턴스 데이터에서 2GB만 변경되었다면 두 번째 인스턴스 스냅샷 요금은 매달 $0.10 USD에 불과합니다.

내 AWS 계정을 관리하려면 어떻게 해야 하나요?

Lightsail는 AWS 서비스이며 AWS 클라우드 인프라에서 실행됩니다. 동일한 AWS 계정과 자격 증명을 사용하여 Lightsail과 AWS Management Console에 로그인합니다.

AWS 계정 암호, 사용자 이름, 연락 방법, 옵트인 리전(기본적으로 비활성화된 리전) 또는 AWS 결제 및 비용 관리 콘솔에서 결제 정보 변경을 포함하여 AWS 계정을 관리할 수 있습니다.

활성화 및 비활성화된 옵트인 리전은 어떻게 관리할 수 있나요?

옵트인 리전(기본적으로 비활성화된 리전)은 활성화하거나 비활성화할 수 있습니다. 옵트인 리전을 사용하려면 먼저 활성화해야 합니다. 사용 가능한 리전에 대한 자세한 내용과 옵트인 리전을 관리하는 방법은 Lightsail의 리전 및 가용 영역 섹션을 참조하세요.

비활성화된 옵트인 리전의 리소스는 어떻게 되나요?

비활성화된 옵트인 리전의 모든 리소스는 계속 실행되며 정상 요율로 요금이 발생합니다. 리전이 비활성화, Lightsail API AWS CLI또는 SDKs인 동안에는 비활성화된 옵트인 리전의 리소스를 Lightsail 콘솔로 관리할 수 없습니다. 그러한 리소스를 삭제하려면 먼저 리전을 일시적으로 다시 활성화해야 관리할 수 있습니다. 자세한 내용은 비활성화된 옵트인 리전에서 리소스는 어떻게 삭제할 수 있나요? 단원을 참조하십시오.

비활성화된 옵트인 리전에서 리소스는 어떻게 삭제할 수 있나요?

리소스를 삭제하기 전에 옵트인 리전을 비활성화한 경우 해당 리전을 일시적으로 다시 활성화하여 해당 리소스를 삭제할 수 있습니다. 자세한 내용은 Lightsail에 대한 옵트인 리전 비활성화 섹션을 참조하세요.

Lightsail의 법적 사용 약관은 무엇인가요?

Lightsail은 Amazon 웹 서비스이므로, Lightsail을 사용하려면 먼저 AWS 고객 계약 및 서비스 약관에 동의해야 합니다. Lightsail 인스턴스를 생성할 때는 소프트웨어 사용이 판매자의 최종 사용자 라이선스 계약에도 따른다는 점에 추가로 동의하게 되며, 이 약관은 인스턴스 생성 페이지에서 살펴볼 수 있습니다.

Lightsail 요금은 어떻게 결제할 수 있나요?

AWS Billing and Cost Management 콘솔을 통해 청구서를 결제하고 관리할 수 있습니다.는 대부분의 주요 신용 카드를 AWS 허용합니다. 결제 방법 관리에 대한 자세한 내용은 여기에서 확인할 수 있습니다.

Lightsail에서의 데이터 전송

내 데이터 전송 플랜 허용량을 초과할 경우에는 어떻게 됩니까?

대다수의 고객은 각자에게 할당된 한도로 충분히 사용량이 감당되어 추가 요금이 발생하지 않도록 데이터 전송 요금제를 설계했습니다. 인스턴스가 플랜의 데이터 전송 허용량을 초과할 경우, 사용한 데이터 전송량에 대해 GB당 초과 요금이 부과됩니다(인터넷으로 송신되는 데이터만 해당).

인스턴스가 플랜의 데이터 전송 허용량을 초과하더라도 많은 유형의 데이터 전송이 무료입니다. Lightsail 인스턴스 및 데이터베이스로 수신되는 데이터는 항상 무료입니다. 프라이빗 IP 주소를 사용하는 경우 Lightsail 인스턴스에서 다른 Lightsail 인스턴스로, Lightsail 인스턴스와 Lightsail 관리형 데이터베이스 간에 또는 동일한 리전의 AWS 리소스로 데이터 전송도 무료입니다.

인스턴스에는 어떤 유형의 데이터 전송에 대해 요금이 청구됩니까?

인스턴스 플랜의 월별 무료 데이터 전송 허용량을 초과하면 퍼블릭 IP 주소를 사용할 때 Lightsail 인스턴스에서 인터넷으로 또는 다른 로 AWS 리전 또는 동일한 리전의 AWS 리소스로 전송되는 데이터에 대해 요금이 부과됩니다. 무료 허용량을 초과하는 이러한 유형의 데이터 전송 요금은 다음과 같습니다.

- 미국 동부(오하이오) (us-east-2): 0.09 USD/GB
- 미국 동부(버지니아 북부) (us-east-1): 0.09 USD/GB
- 미국 서부(오리건) (us-west-2): 0.09 USD/GB
- 아시아 태평양(뭄바이) (ap-south-1): 0.13 USD/GB
- 아시아 태평양(서울) (ap-northeast-2): 0.13 USD/GB
- 아시아 태평양(싱가포르) (ap-southeast-1): 0.12 USD/GB
- 아시아 태평양(시드니) (ap-southeast-2): 0.17 USD/GB
- 아시아 태평양(자카르타)(ap-southeast-3): 0.132 USD/GB
- 아시아 태평양(도쿄) (ap-northeast-1): 0.14 USD/GB
- 캐나다(중부) (ca-central-1): 0.09 USD/GB
- EU(프랑크푸르트) (eu-central-1): 0.09 USD/GB
- EU(아일랜드) (eu-west-1): 0.09 USD/GB
- EU(런던) (eu-west-2): 0.09 USD/GB
- EU(파리) (eu-west-3): 0.09 USD/GB

- EU(스톡홀름) (eu-north-1): 0.09 USD/GB

서로 다른 가용 영역에서 생성된 인스턴트들은 영역 간에 무료, 비공개로 통신할 수 있고 동시에 통신 성능이 저하될 가능성이 훨씬 낮아집니다. 가용 영역을 사용하면 데이터 전송 비용이 늘거나 애플리케이션의 보안을 훼손시키지 않고 고가용성 애플리케이션과 웹사이트를 빌드할 수 있습니다.

인스턴스의 내 데이터 전송 허용량은 어떤 식으로 운영됩니까?

모든 Lightsail 인스턴스 플랜에는 데이터 전송 허용량이 제공됩니다. 인스턴스의 데이터 송수신은 모두 데이터 전송 허용량에 가산됩니다. 데이터 전송 허용량을 초과하는 경우 Lightsail 인스턴스의 퍼블릭 IP 주소를 사용하여 인스턴스에서 인터넷 또는 AWS 리소스로 전송되는 초과 데이터 전송에 대해서만 요금이 부과됩니다. 허용량을 초과하는 데이터 전송에 대한 이 추가 요금은 무료 평가 기간 내에 있는 리소스에 대해서도 지불해야 합니다. 데이터 전송 허용량은 매달 재설정되므로, 인스턴스에서 해당 월 내의 허용량을 필요할 때마다 사용할 수 있습니다.

Lightsail 인스턴스로의 초과 데이터 전송 IN에 대해서는 요금이 부과되지 않습니다(예제 1 참조). 리전에서 동일한 번들(bundleId)의 인스턴스에 대해 데이터 전송 허용량이 집계됩니다(예제 2 및 예제 3 참조). 동일한 크기의 IPv4 및 IPv6 인스턴스에 대해서도 데이터 전송 허용량이 집계됩니다(예제 4 참조). 인스턴스를 삭제하고 새 인스턴스를 생성해도 데이터 전송 허용량은 재설정되지 않습니다(예제 5 참조). 새 인스턴스를 생성해도 기존 데이터 전송 초과는 상쇄되지 않습니다(예제 6 참조). Lightsail 번들에 대한 자세한 내용은 Amazon Lightsail API 레퍼런스의 번들을 참조하세요.

- 예제 1 - 월별 데이터 전송 허용량이 1TB인 월별 5 USD 인스턴스 번들(bundleId `nano_3_0`)이 1개 있습니다. 500GB의 데이터를 인터넷(데이터 전송 OUT)으로 보내고 400GB의 데이터를 인스턴스(데이터 전송 IN)로 전송하는 경우 1TB 허용량 중 900GB를 사용하게 됩니다. 인터넷으로 200GB의 데이터를 추가로 전송하는 경우 허용량 100GB를 초과하고 100GB에 대한 데이터 전송 초과 요금이 부과됩니다. 다음으로 인스턴스에 200GB의 데이터를 전송하는 경우 초과에 대한 요금이 부과되지 않습니다.
- 예제 2 - 한 리전에서 한 달 동안 월 5 USD의 인스턴스 번들(bundleId `nano_3_0`)이 2개 있고, 각각 월 1TB의 데이터 전송 허용량이 있는 경우 총 2TB의 데이터 전송 허용량이 발생합니다. 첫 번째 인스턴스를 사용하여 1.5TB의 데이터를 인터넷으로 보내고 두 번째 인스턴스를 사용하여 100GB의 데이터를 인터넷으로 전송하는 경우에도 총 허용량인 2TB의 400GB는 그대로 유지되며 데이터 전송 초과 요금이 부과되지 않습니다.
- 예제 3 - 두 세트의 인스턴스 번들을 생성합니다. 두 세트 모두 미국 서부(오리건) 지역에 있는 인스턴스 번들(bundleId `nano_3_0`)을 월별 5 USD로 두 개로 설정하고 B를 월별 인스턴스 번들(bundleId `micro_3_0`)을 7 USD로 세 개로 설정합니다. 이를 통해 집합 A에 대한 데이터 전송 허용 한도 2TB와 집합 B에 대한 데이터 전송 허용 한도 6TB가 제공됩니다. 집합 A 인스턴스를 통해 인터넷으로

3TB의 데이터를 전송하고 집합 B 인스턴스를 통해 인터넷으로 4TB의 데이터를 전송하는 경우 집합 A 인스턴스에 대한 데이터 전송 허용 한도를 초과하고 1TB에 대한 데이터 전송 초과 요금이 부과됩니다. 세트 B 인스턴스에 대한 허용 한도는 2TB 이내입니다.

- 예제 4 - 결제 월의 첫 20일 이내에 월별 3.50 USD IPv6 인스턴스 번들(bundleId nano_ipv6_3_0)에 대한 총 1TB 데이터 전송 허용량 중 600GB를 사용했습니다. 인스턴스의 네트워킹 유형을 21일차에 듀얼 스택(bundleId nano_3_0은 월 요금 5 USD 부과)으로 전환하기로 결정합니다. 해당 월의 데이터 전송 사용률은 재설정되지 않으며 600GB로 유지되고 400GB 허용량이 남습니다. 결제 월의 나머지 기간 동안 500GB의 데이터를 인터넷으로 전송하는 경우 100GB에 대한 데이터 전송 초과 요금이 발생합니다.
- 예제 5 - 월별 5 USD 인스턴스 번들(bundleIdnano_3_0) 3개에 대해 월별 1TB의 데이터 전송 허용량이 있습니다. 결제 월 내에 총 3TB 데이터 전송 허용량 중 1TB를 소비했다고 가정해 보겠습니다. 이렇게 하면 남은 데이터 전송 허용량은 2TB가 됩니다. 모든 인스턴스를 삭제하고 동일한 결제 월 내에 동일한 리전에서 동일한 번들(bundleId nano_3_0)의 새 인스턴스 3개를 생성하는 경우 데이터 전송 사용률은 여전히 1TB이고 나머지 데이터 전송 허용량은 여전히 2TB입니다. 데이터 전송 OUT 초과 요금 발생을 시작하기 전 같은 달 내에 인스턴스를 통해 2TB의 데이터를 더 전송할 수 있습니다.
- 예제 6 - 결제 월의 처음 20일 동안 월 5 USD 인스턴스 번들(bundleId nano_3_0)에 대한 월 1TB 데이터 전송 허용량을 사용한 후 추가로 100GB의 데이터를 인터넷으로 전송했습니다. 이 100GB에 대해서는 데이터 전송 초과 요금이 발생합니다. 이제 동일한 번들(bundleId nano_3_0)의 다른 새 인스턴스를 생성하는 경우에도 이전에 발생한 데이터 전송 초과 요금이 계속 청구됩니다. 이러한 인스턴스를 통한 추가 데이터 전송에는 계속해서 데이터 전송 초과 요금이 발생합니다.

인스턴스의 데이터 전송 허용량은 AWS 리전에 따라 어떻게 달라지나요?

Lightsail 인스턴스에 대한 리전 데이터 전송 허용량은 Amazon Lightsail 요금에 나와 있습니다. 허용량은 아시아 태평양(뭄바이 AWS 리전, 시드니, 자카르타) 리전을 제외하고 모든에 대해 동일합니다. 뭄바이, 시드니 및 자카르타 리전의 플랜에는 다른 리전의 데이터 전송 허용량의 절반이 포함됩니다.

Lightsail 관리형 데이터베이스에 대한 데이터 전송 허용량은 모든 AWS 리전에서 동일합니다.

데이터 전송 비용은 얼마나 됩니까?

인스턴스, 데이터베이스 및 콘텐츠 전송 네트워크(CDN) 배포 플랜에는 데이터 전송 허용량이 제공됩니다.

Lightsail 인스턴스의 경우 인스턴스의 데이터 송수신은 모두 데이터 전송 허용량에 가산됩니다. 데이터 전송 허용량을 초과하는 경우 인스턴스의 퍼블릭 IP 주소를 사용하여 Lightsail 인스턴스에서 인터넷 또

는 AWS 리소스로 전송되는 초과 데이터 전송에 대해서만 요금이 부과됩니다. Lightsail 인스턴스로의 초과 데이터 전송 IN에 대해서는 요금이 부과되지 않습니다. 인스턴스의 프라이빗 IP 주소를 사용하여 데이터를 전송할 때 Lightsail 인스턴스의 데이터 수신 및 송신은 모두 데이터 전송 할당량 계산에 포함됩니다. 그러나 프라이빗 IP 주소를 사용하여 전송된 데이터는 데이터 전송 허용량을 초과하더라도 무료입니다.

Lightsail 관리형 데이터베이스의 경우 데이터 송신만 허용량에 가산됩니다. 데이터 전송 허용량을 초과할 경우, Lightsail 관리형 데이터베이스에서 인터넷으로 송신되는 데이터에 대한 요금만 청구됩니다.

Lightsail CDN 배포의 경우 배포 외부로 전송되는 모든 데이터는 허용량에 가산됩니다. 배포 데이터 전송 허용량을 초과하면 배포 외부로 전송되는 모든 데이터에는 요금이 부과됩니다.

내 데이터 전송 허용량은 로드 밸런서에서 어떤 식으로 운영됩니까?

로드 밸런서는 데이터 전송 허용량을 사용하지 않습니다. 로드 밸런서와 대상 인스턴스 또는 배포 간 트래픽이 측정되어 해당 인스턴스 또는 배포의 데이터 전송 허용량에 가산됩니다. 이는 인터넷으로 송수신되는 트래픽이 로드 밸런서를 사용하지 않는 Lightsail 인스턴스의 데이터 전송 허용량에 가산되는 것과 같은 방식입니다. 로드 밸런서에서 인터넷으로 송수신되는 트래픽은 인스턴스의 데이터 전송 허용량에 포함되지 않습니다.

내 데이터 전송 허용량은 객체 스토리지에서 어떻게 사용하나요?

다음의 경우를 제외하고 Lightsail 객체 스토리지를 오가는 데이터에 데이터 전송 허용량을 사용할 수 있습니다.

- 인터넷에서 Lightsail 객체 스토리지로 전송되는 데이터
- Lightsail 객체 스토리지 리소스 간 데이터 전송
- Lightsail 객체 스토리지에서 동일한의 다른 Lightsail 리소스로 전송된 데이터 AWS 리전 (다른 AWS 계정의 리소스 포함) AWS 리전
- Lightsail 객체 스토리지에서 Lightsail CDN 배포로 전송된 데이터

배포에는 어떤 유형의 데이터 전송에 대해 요금이 청구됩니까?

Lightsail CDN 배포 플랜의 데이터 전송 허용량을 초과하면 모든 데이터 송신에 요금이 부과됩니다. 배포 허용량을 초과하는 데이터 전송 요금은 다음과 같습니다.

- 아시아 태평양: 0.13 USD/GB

- 캐나다: 0.09 USD/GB
- 유럽: 0.09 USD/GB
- 인도: 0.13 USD/GB
- 일본: 0.14 USD/GB
- 중동: 0.11 USD/GB
- 남아프리카공화국: 0.11 USD/GB
- 남아메리카: 0.11 USD/GB
- 미국: 0.09 USD/GB

Lightsail의 인스턴스 데이터 전송 할당량과 배포 데이터 전송 할당량의 차이점은 무엇입니까?

배포 허용량을 초과하는 데이터 전송에 대한 요금은 Lightsail 인스턴스와 다릅니다. 데이터 수신 및 송신은 인스턴스의 데이터 전송 할당량에 포함되지만, 배포 할당량에는 오리진 및 최종 사용자에게 전달되는 데이터 송신만이 포함됩니다. 또한, 배포 할당량을 초과하는 모든 데이터 송신에는 초과 요금이 부과됩니다. 단, 일부 데이터 송신 유형은 인스턴스에 대해 무료로 제공됩니다. 마지막으로, Lightsail 배포는 다른 리전 초과 요금 모델을 사용하지만, 비율은 대부분 인스턴스 초과 요금에 대해 부과되는 것과 동일합니다.

컨테이너 서비스 내/외부로 데이터를 전송할 때 요금이 부과되나요?

모든 컨테이너 서비스에는 데이터 전송 할당량(월 500GB)이 제공됩니다. 이 할당량은 서비스 내/외부로의 데이터 전송에 모두 적용됩니다. 할당량을 초과하면 퍼블릭 IP 주소를 사용할 때 Lightsail 컨테이너 서비스에서 인터넷으로 또는 다른 로 AWS 리전 또는 동일한 리전의 AWS 리소스로 전송되는 데이터에 대한 요금이 부과됩니다. 무료 허용량을 초과하는 이러한 유형의 데이터 전송 요금은 다음과 같습니다.

- 미국 동부(오하이오) (us-east-2): 0.09 USD/GB
- 미국 동부(버지니아 북부) (us-east-1): 0.09 USD/GB
- 미국 서부(오리건) (us-west-2): 0.09 USD/GB
- 아시아 태평양(뭄바이) (ap-south-1): 0.13 USD/GB
- 아시아 태평양(서울) (ap-northeast-2): 0.13 USD/GB
- 아시아 태평양(싱가포르) (ap-southeast-1): 0.12 USD/GB
- 아시아 태평양(시드니) (ap-southeast-2): 0.17 USD/GB

- 아시아 태평양(도쿄) (ap-northeast-1): 0.14 USD/GB
- 캐나다(중부) (ca-central-1): 0.09 USD/GB
- EU(프랑크푸르트) (eu-central-1): 0.09 USD/GB
- EU(아일랜드) (eu-west-1): 0.09 USD/GB
- EU(런던) (eu-west-2): 0.09 USD/GB
- EU(파리) (eu-west-3): 0.09 USD/GB
- EU(스톡홀름) (eu-north-1): 0.09 USD/GB

블록 스토리지(디스크)

Lightsail 블록 스토리지로 어떤 작업을 수행할 수 있나요?

Lightsail 블록 스토리지는 개별 하드 드라이브처럼 Lightsail 인스턴스에 연결할 수 있는 추가 스토리지 볼륨(Lightsail에서는 "연결된 디스크"라고 함)을 제공합니다. 연결된 디스크는 핵심 서비스로부터 특정 데이터를 분리하고 인스턴스 및 시스템 디스크에 장애 또는 다른 문제가 발생하는 경우 애플리케이션 데이터를 보호해야 하는 애플리케이션 또는 소프트웨어에 유용합니다. 연결된 디스크는 저장된 데이터에 자주 액세스하는 애플리케이션 또는 소프트웨어에 필요한 일관성 성능과 짧은 지연 시간을 제공합니다.

Lightsail 블록 스토리지 디스크는 SSD(Solid State Drive)를 사용합니다. 이러한 유형의 블록 스토리지는 저렴한 비용과 우수한 성능 간 균형을 유지하며, Lightsail에서 실행되는 워크로드 대부분을 지원하도록 설계되어 있습니다. 일관된 IOPS 성능과 디스크당 높은 처리량이 필요하거나 MongoDB, Cassandra 등과 같은 대규모 데이터베이스를 실행하는 애플리케이션을 보유한 고객의 경우, Lightsail 대신 GP2 또는 프로비저닝된 IOPS SSD 스토리지가 탑재된 Amazon EC2를 사용하는 것이 좋습니다.

연결된 디스크는 내 Lightsail 플랜에 포함된 스토리지와 어떻게 다른가요?

Lightsail 플랜에 포함된 시스템 디스크는 인스턴스의 루트 디바이스입니다. 인스턴스를 종료하면, 시스템 디스크도 삭제됩니다. 인스턴스 장애가 발생하는 경우, 시스템 디스크가 영향을 받을 수 있습니다. 또한, 시스템 디스크를 분리하거나 인스턴스와 별도로 백업할 수 없습니다. 연결된 디스크에 저장된 데이터는 인스턴스와 독립적으로 유지됩니다. 연결된 디스크를 분리하고 인스턴스 사이에 이동할 수 있습니다. 디스크의 수동 스냅샷을 생성하여 인스턴스와 독립적으로 백업할 수 있습니다. 데이터를 보호하려면 Lightsail 인스턴스의 시스템 디스크를 임시 데이터에만 사용하는 것이 좋습니다. 더 높은 수준의 안정성이 필요한 데이터의 경우, 연결된 디스크를 사용하고 디스크 또는 인스턴스 스냅샷을 사용하여 디스크를 정기적으로 백업하는 것이 좋습니다.

내 연결 디스크의 최대 크기는 어떻게 됩니까?

연결된 각 디스크는 최대 16TB까지 가능하고, Lightsail 계정 내 연결된 블록 스토리지의 총 크기는 20TB를 넘지 않아야 합니다.

Lightsail 인스턴스당 몇 개의 디스크를 연결할 수 있나요?

Lightsail 인스턴스에 최대 15개의 디스크를 연결할 수 있습니다.

하나의 디스크를 2개 이상의 인스턴스에 연결할 수 있습니까?

아니요. 디스크는 한 번에 하나의 인스턴스에만 연결될 수 있습니다.

내 디스크를 인스턴스에 연결해야 합니까?

아니요. 디스크를 인스턴스에 연결하지 않도록 선택할 수 있습니다. 디스크는 연결이 안 된 상태로 계정에 남아있게 됩니다. 디스크가 인스턴스에 연결되지 않더라고 비용에는 차이가 없습니다.

내 연결된 디스크의 크기를 늘릴 수 있습니까?

예. 디스크 스냅샷을 생성한 후 해당 스냅샷에서 더 큰 디스크를 새로 생성하는 방법으로 디스크의 크기를 늘릴 수 있습니다.

Lightsail 블록 스토리지는 암호화를 제공하나요?

예. 데이터를 안전하게 유지하는 데 도움이 되도록 기본적으로 모든 Lightsail의 연결된 디스크와 디스크 스냅샷은 사용자 대신 Lightsail에서 관리하는 키를 사용하여 유휴 상태에서 암호화됩니다. 또한 Lightsail은 Lightsail 인스턴스와 연결된 디스크 간에 데이터가 이동할 때도 데이터 암호화를 제공합니다.

Lightsail 블록 스토리지에서 기대할 수 있는 가용성 수준은 어떻게 되나요?

Lightsail 블록 스토리지는 뛰어난 가용성과 안정성을 제공하도록 설계되었습니다. 각 연결된 디스크는 가용 영역 내에서 자동으로 복제되어 구성요소 장애로부터 사용자를 보호합니다. Lightsail 블록 스토리지 디스크는 99.99% 가용성으로 설계되었습니다. 또한 Lightsail은 정기적인 데이터 백업을 허용하는 디스크 스냅샷을 지원합니다.

내 연결된 디스크를 백업하려면 어떻게 해야 합니까?

디스크의 수동 스냅샷을 생성하여 디스크를 백업할 수 있습니다. 인스턴스의 수동 스냅샷을 생성하거나 디스크가 연결된 인스턴스의 자동 스냅샷을 활성화하여 전체 인스턴스 및 연결된 디스크를 백업할 수도 있습니다. 인스턴스에 연결된 디스크는 인스턴스 수동 및 자동 스냅샷에 포함됩니다.

인증서

Lightsail에서 제공하는 인증서를 어떻게 사용할 수 있나요?

SSL/TLS 인증서는 웹 사이트 또는 애플리케이션의 자격 증명을 설정하고 브라우저와 웹 사이트 간 연결을 보호하는 데 사용됩니다. Lightsail에서는 로드 밸런서에서 사용할 수 있는 서명된 인증서를 제공하며, 로드 밸런서는 확인된 트래픽을 안전한 AWS 네트워크를 통해 대상 인스턴스로 라우팅하기 전에 SSL/TLS를 종료합니다. Lightsail 인증서는 Lightsail 로드 밸런서에서만 사용할 수 있고 개별 Lightsail 인스턴스에서는 사용할 수 없습니다.

내 인증서를 검증하려면 어떻게 해야 합니까?

Lightsail 인증서는 도메인 인증 인증서입니다. 즉, 인증 기관에서 인증서를 프로비저닝할 수 있으려면 먼저 사용자가 웹 사이트 도메인을 소유하고 있거나 액세스 권한이 있음을 검증하여 자격 증명에 대한 증거를 제공해야 합니다. 새 인증서를 요청하면 Lightsail에서 자동으로 인증서 검증을 시도합니다 인증서를 자동으로 검증할 수 없는 경우 Lightsail에서 검증 대상 도메인의 DNS 영역에 CNAME 레코드를 추가하라는 메시지를 표시합니다. Lightsail DNS 관리 또는 외부 DNS 호스팅 공급자 중 어디에서 현재 DNS 영역을 관리하든 72시간 이내에 CNAME 레코드를 추가할 수 있습니다.

내 도메인을 검증할 수 없는 경우에는 어떻게 됩니까?

보안을 위해서는 사용자가 도메인을 소유하고 있음을 검증할 수 있어야 합니다. 즉, 사용자 또는 조직 내 누군가가 어떤 이유로든 인증서를 검증하기 위해 DNS 레코드를 추가할 수 없을 경우 Lightsail에서 HTTPS 지원 로드 밸런서를 사용할 수 없습니다.

내 인증서에는 몇 개의 도메인과 하위 도메인을 추가할 수 있습니까?

인증서당 최대 10개의 도메인 또는 하위 도메인을 추가할 수 있습니다. Lightsail은 현재 와일드카드 도메인을 지원하지 않습니다.

내 인증서에 연결된 도메인을 변경하려면 어떻게 해야 합니까?

인증서에 연결된 도메인을 변경하려면, 인증서를 다시 제출하고 도메인의 소유권을 다시 검증해야 합니다. 인증서 관리 화면에 나온 단계를 따라 인증서를 다시 생성하고 메시지가 표시되면 도메인을 추가 또는 삭제합니다.

내 인증서를 갱신하려면 어떻게 해야 합니까?

Lightsail은 SSL/TLS 인증서에 대해 관리형 갱신을 제공합니다. 즉, Lightsail은 인증서가 만료되기 전에 자동으로 이를 갱신하려고 시도하며 사용자가 수행해야 할 작업은 없습니다. Lightsail 인증서를 자동으로 갱신하려면 먼저 해당 인증서가 로드 밸런서에 연결되어 있어야 합니다.

내 로드 밸런서를 삭제하면 대상 인증서는 어떻게 됩니까?

로드 밸런서가 삭제되면, 인증서도 삭제됩니다. 나중에 같은 도메인에 대한 인증서를 사용해야 하는 경우, 새로운 인증서를 요청하고 검증해야 합니다.

Lightsail에서 제공한 내 인증서를 다운로드할 수 있나요?

아니요. Lightsail 인증서는 Lightsail 계정에 연결되어 있으며, Lightsail 외부에서 제거할 수도 없고 사용할 수도 없습니다.

연락처 및 모니터링 알림

알림이란 무엇입니까?

인스턴스, 데이터베이스 또는 로드 밸런서의 지표가 지정된 임계값을 초과하면 알림을 받도록 Lightsail에 경보를 구성할 수 있습니다. 알림은 Lightsail 콘솔에 표시되는 배너, 지정한 주소로 전송되는 이메일 또는 지정한 휴대폰 번호로 전송되는 SMS 문자 메시지 형태로 사용할 수 있습니다. 이메일 및 SMS 문자 메시지로 알림을 받으려면 리소스를 모니터링하려는 각 AWS 리전의 알림 연락처로 이메일 주소 및 휴대폰 번호를 추가해야 합니다. 알림에 대한 자세한 내용은 알림을 참조하세요.

연락처를 몇 개까지 추가할 수 있습니까?

리소스를 모니터링하려는 각 AWS 리전에서 이메일 주소 하나와 휴대폰 번호 하나를 추가할 수 있습니다. SMS 문자 메시징은 Lightsail 리소스를 생성할 수 있는 AWS 리전 중 일부에서는 지원되지 않으며, 일부 국가 및 리전에는 문자 메시지를 전송할 수 없습니다. 알림에 대한 자세한 내용은 알림을 참조하세요.

컨테이너 서비스 수

Lightsail 컨테이너 서비스로 할 수 있는 작업은 무엇인가요?

Lightsail 컨테이너 서비스는 클라우드에서 컨테이너형 애플리케이션을 쉽게 실행할 수 있는 방법을 제공합니다. 간단한 웹 앱에서 멀티 티어 마이크로 서비스에 이르기까지 컨테이너 서비스에서 다양한 애플리케이션을 실행할 수 있습니다. 컨테이너 서비스에 필요한 컨테이너 이미지, 성능(CPU, RAM) 및 규모(노드 수)만 지정하면 Lightsail에서 기본 인프라를 관리할 필요 없이 컨테이너 서비스를 실행할 수 있습니다. Lightsail에서는 컨테이너 서비스에서 실행 중인 애플리케이션에 액세스할 수 있는 로드 밸런싱 TLS 엔드포인트를 제공합니다.

Lightsail 컨테이너 서비스에서 Docker 컨테이너를 실행할 수 있나요?

예, Lightsail은 Linux 기반 Docker 컨테이너를 지원합니다. 현재 Windows 컨테이너는 지원되지 않습니다.

퍼블릭 컨테이너 이미지를 Lightsail 컨테이너 서비스와 함께 사용하려면 어떻게 해야 하나요?

Amazon ECR 퍼블릭 레지스트리와 같은 온라인 퍼블릭 레지스트리의 컨테이너 이미지를 사용하거나 AWS CLI를 사용하여 몇 가지 간단한 단계를 거쳐 자체 사용자 지정 이미지를 구축하고 Lightsail에 푸시할 수 있습니다. 자세한 내용은 컨테이너 이미지 푸시 및 관리를 참조하세요.

프라이빗 컨테이너 레지스트리에서 컨테이너 이미지를 가져올 수 있나요?

현재 Lightsail 컨테이너 서비스에서는 퍼블릭 컨테이너 레지스트리만 지원됩니다. 또는 로컬 시스템에서 Lightsail로 사용자 지정 컨테이너 이미지를 푸시하여 프라이빗으로 유지할 수 있습니다.

온디맨드 방식으로 서비스의 성능 및 규모를 변경할 수 있나요?

예, 컨테이너 서비스 성능과 규모는 서비스를 생성한 후에도 언제든지 변경할 수 있습니다.

Lightsail 컨테이너 서비스에서 생성한 HTTPS 엔드포인트 이름을 사용자 지정할 수 있나요?

Lightsail은 *<service-name>*.*<random-guid>*.*<aws-region-name>*.cs.amazonlightsail.com 형식으로 모든 컨테이너 서비스에 HTTPS 엔드포인트를 제공합니다. 서비스 이름만 사용자 지정할 수 있습니다. 아니면 사용자 지정 도메인 이름을 사용할 수 있습니다. 자세한 내용은 사용자 지정 도메인 활성화 및 관리를 참조하세요.

Lightsail 컨테이너 서비스의 HTTPS 엔드포인트에 사용자 지정 도메인을 사용할 수 있나요?

예. Lightsail에서 사용자 지정 도메인 이름이 포함된 SSL/TLS 인증서를 생성하여 컨테이너 서비스에 연결할 수 있습니다. 인증서는 도메인 검증을 받아야 합니다. 도메인의 DNS가 Lightsail DNS 영역을 사용하는 경우 도메인(`example.com`) 또는 하위 도메인(`www.example.com`)의 Apex에 대한 트래픽을 컨테이너 서비스로 라우팅할 수 있습니다. 또는 ALIAS 레코드 추가를 지원하는 DNS 호스팅 공급자를 사용하여 도메인의 Apex(`example.com`)를 Lightsail 컨테이너 서비스의 기본 도메인(퍼블릭 DNS)에 매핑할 수 있습니다. 자세한 내용은 사용자 지정 도메인 활성화 및 관리를 참조하세요.

Lightsail 컨테이너 서비스의 사용 요금은 얼마인가요?

Lightsail 컨테이너 서비스는 온디맨드 시간 요율에 따라 요금이 청구되므로, 사용하는 만큼만 결제하면 됩니다. 사용자가 이용하는 모든 Lightsail 컨테이너 서비스에 대해 월간 최대 서비스 요금까지 시간당 고정 요금이 부과됩니다. 월간 최대 서비스 요금은 서비스 성능의 기본 요금에 서비스 규모를 곱하여 계산합니다. 예를 들어, 마이크로 성능과 규모 2의 서비스는 매월 최대 10 USD*2=20 USD의 비용이 듭니다. 가장 저렴한 Lightsail 컨테이너 서비스는 시간당 0.0094 USD(7 USD/월)부터 시작합니다. 각 서비스에서 월 500GB의 무료 할당량 이상을 사용하는 경우, 추가 데이터 전송 요금이 부과될 수 있습니다. 자세한 내용은 컨테이너 서비스 내/외부로 데이터를 전송할 때 요금이 부과되나요? 단원을 참조하십시오.

컨테이너 서비스를 며칠만 사용해도 한 달 요금이 부과되나요?

Lightsail 컨테이너 서비스가 실행 중이거나 비활성화된 상태일 때만 요금이 부과됩니다. 월말 이전에 Lightsail 컨테이너 서비스를 삭제하면 Lightsail 컨테이너 서비스를 사용한 총 시간을 기준으로 비례 할당으로 계산된 비용이 부과됩니다. 예를 들어, 마이크로 성능과 규모 1의 Lightsail 컨테이너 서비스를 한 달에 100시간 사용할 경우 1.34 USD(0.0134 USD*100)의 요금이 부과됩니다.

컨테이너 서비스를 중지하는 것과 삭제하는 것에는 어떤 차이가 있나요?

컨테이너 서비스를 비활성화하면 컨테이너 노드가 비활성화 상태가 되고 서비스의 퍼블릭 엔드포인트가 HTTP 상태 코드 '503'을 반환합니다. 서비스를 활성화하면 마지막 활성 배포로 복원됩니다. 성능과 규모 구성도 그대로 유지됩니다. 다시 활성화한 후에 퍼블릭 엔드포인트 이름이 변경되지 않습니다. 배포 기록과 컨테이너 이미지도 보존됩니다.

컨테이너 서비스를 삭제하면 안전하지 않은 작업을 수행하는 것입니다. 이렇게 하면 서비스의 모든 컨테이너 노드가 영구적으로 삭제됩니다. 서비스와 연결된 HTTPS 퍼블릭 엔드포인트 주소, 컨테이너 이미지, 배포 기록 및 로그도 영구적으로 삭제됩니다. 엔드포인트 주소는 복구할 수 없습니다.

컨테이너 서비스가 비활성화된 상태인 경우에도 요금이 청구되나요?

예, 컨테이너 서비스가 비활성화 상태인 경우에도 컨테이너 서비스의 성능 및 규모 구성에 따라 요금이 부과됩니다.

Lightsail 콘텐츠 전송 네트워크(CDN) 배포의 오리진으로 컨테이너 서비스를 사용할 수 있나요?

컨테이너 서비스는 현재 Lightsail CDN 배포의 오리진으로 지원되지 않습니다.

컨테이너 서비스를 Lightsail 로드 밸런서의 대상으로 사용할 수 있나요?

아니요, 컨테이너 서비스는 현재 Lightsail 로드 밸런서의 대상으로 사용할 수 없습니다. 그러나 컨테이너 서비스의 퍼블릭 엔드포인트에는 로드 밸런싱 기능이 내장되어 있습니다.

HTTP 요청을 HTTPS로 리디렉션하도록 컨테이너 서비스의 퍼블릭 엔드포인트를 구성할 수 있나요?

Lightsail 컨테이너 서비스 퍼블릭 엔드포인트는 콘텐츠가 안전하게 제공되도록 모든 HTTP 요청을 HTTPS로 자동 리디렉션합니다.

컨테이너 서비스가 모니터링 및 알림을 지원하나요?

컨테이너 서비스는 서비스 노드 전체의 CPU 사용률 및 메모리 사용률 지표를 제공합니다. 현재 이러한 지표를 기반으로 하는 알림은 지원되지 않습니다.

Lightsail 컨테이너 서비스는 IPv6를 지원하나요?

Lightsail 컨테이너 서비스 HTTPS 엔드포인트는 IPv4 및 IPv6를 모두 지원합니다. IPv6는 컨테이너 서비스에서 비활성화할 수 없습니다.

콘텐츠 전송 네트워크 배포

Lightsail CDN 배포로 수행할 수 있는 작업은 무엇인가요?

Lightsail콘텐츠 전송 네트워크(CDN) 배포를 사용하면 Amazon CloudFront를 기반으로 하는 Amazon의 글로벌 전송 네트워크에 콘텐츠를 저장하고 제공하여 Lightsail 리소스에서 호스팅된 콘텐츠를 쉽고 빠르게 전송할 수 있습니다. 또한, 배포를 통해 SSL 인증서를 간단하게 생성 및 호스팅하여 웹 사이트

가 HTTPS 트래픽을 지원하도록 할 수 있습니다. 마지막으로, 배포는 Lightsail 리소스에 가해지는 부하를 줄이고 웹 사이트가 대규모 트래픽 스파이크를 처리하는 데 도움이 됩니다. Lightsail의 모든 기능과 마찬가지로 몇 번의 클릭만으로 설정을 완료할 수 있으며, 편리한 월별 요금이 제공됩니다.

배포의 오리진으로 사용할 수 있는 리소스 유형은 무엇인가요?

Lightsail 배포를 사용하면 Lightsail 인스턴스와 로드 밸런서를 오리진으로 사용할 수 있습니다. Lightsail 컨테이너는 현재 오리진으로 지원되지 않습니다. S3 버킷과 같은 Lightsail 외부의 리소스는 지원되지 않습니다.

고정 IPv4 주소를 Lightsail 배포의 오리진으로 사용하려면 Lightsail 인스턴스에 연결해야 하나요?

예, 고정 IPv4 주소를 오리진으로 지정된 인스턴스에 연결해야 합니다. Lightsail 배포는 현재 IPv6를 지원하지 않습니다.

WordPress 웹 사이트에서 Lightsail 배포를 설정하려면 어떻게 해야 하나요?

배포를 생성하고 WordPress 인스턴스를 오리진으로 선택한 후 플랜을 선택하면 됩니다. Lightsail 배포는 대부분의 WordPress 구성에 맞춰 성능을 최적화하도록 배포 설정을 자동으로 구성합니다.

여러 오리진을 연결할 수 있나요?

Lightsail 배포에 여러 오리진을 연결할 수는 없지만, Lightsail 로드 밸런서에 여러 인스턴스를 연결하고 배포의 오리진으로 지정할 수 있습니다.

Lightsail 배포에서 인증서 생성을 지원하나요?

예, Lightsail 배포를 사용하면 배포의 관리 페이지에서 바로 인증서를 쉽게 만들고 확인하고 연결할 수 있습니다.

인증서가 필요하나요?

배포에 사용자 지정 도메인 이름을 사용하려는 경우에만 인증서가 필요합니다. 모든 Lightsail 배포는 HTTPS가 활성화된 고유한 Amazon CloudFront 도메인 이름으로 생성됩니다. 그러나 배포에서 사용자 지정 도메인을 사용하려면 사용자 지정 도메인의 인증서를 배포에 연결해야 합니다.

생성할 수 있는 인증서 수에 제한이 있나요?

예, 자세한 내용은 Lightsail 서비스 할당량을 참조하세요.

HTTP 요청을 HTTPS로 리디렉션하도록 배포를 구성하려면 어떻게 해야 하나요?

Lightsail 배포는 콘텐츠가 안전하게 제공되도록 모든 HTTP 요청을 HTTPS로 자동 리디렉션합니다.

Apex 도메인이 내 Lightsail 배포를 가리키도록 구성하려면 어떻게 해야 하나요?

Apex 도메인이 CDN 배포를 가리키도록 하려면 도메인의 도메인 이름 시스템(DNS)에서 Apex 도메인을 배포의 기본 도메인에 매핑하는 ALIAS 레코드를 생성해야 합니다. DNS 호스팅 공급자가 ALIAS 레코드를 지원하지 않는 경우, Lightsail DNS 영역을 사용하면 배포의 도메인을 가리키도록 Apex 도메인을 쉽게 구성할 수 있습니다.

배포와 연결된 플랜을 변경할 수 있나요?

예, 배포 플랜은 한 달에 한 번 변경할 수 있습니다. 플랜을 다시 변경하려면 다음 달 초까지 기다려야 합니다.

배포가 작동하는지 어떻게 알 수 있나요?

Lightsail 배포는 배포에서 받은 총 요청 수, 배포에서 클라이언트 및 오리진으로 보낸 데이터 양, 오류가 발생한 요청 비율 등 배포의 성능을 추적하는 다양한 지표를 제공합니다. 또한, 배포 지표와 관련된 알림을 생성할 수 있습니다.

Lightsail 배포에서 캐시된 콘텐츠를 삭제할 수 있나요?

특정 파일이나 폴더가 아닌 모든 캐시된 콘텐츠를 삭제할 수 있습니다.

Lightsail 배포와 Amazon CloudFront 배포는 언제 사용해야 하나요?

Lightsail 배포는 인스턴스 및 로드 밸런서와 같은 Lightsail 리소스에서 웹 사이트나 웹 애플리케이션을 호스팅하는 사용자를 위해 특별히 설계되었습니다. 에서 다른 서비스를 사용하여 웹 사이트 또는 앱을 AWS 호스팅하거나, 구성 요구 사항이 복잡하거나, 초당 요청 수가 많거나 비디오 스트리밍 양이 많은 워크로드가 있는 경우 Amazon CloudFront를 사용하는 것이 좋습니다.

Lightsail 콘텐츠 전송 네트워크(CDN) 배포를 Amazon CloudFront로 이동할 수 있나요?

예, Amazon CloudFront에서 유사하게 구성된 배포를 생성하여 Lightsail 배포를 이동할 수 있습니다. Lightsail 배포에서 구성할 수 있는 모든 설정은 CloudFront 배포에서도 구성할 수 있습니다. 배포를 CloudFront로 이동하려면 다음 단계를 완료하세요.

Lightsail 배포를 CloudFront로 이동하는 방법

- 배포의 오리진으로 구성된 Lightsail 인스턴스의 스냅샷을 생성합니다. 스냅샷을 Amazon EC2로 내보낸 다음, Amazon EC2의 스냅샷에서 새 인스턴스를 생성합니다. 자세한 내용은 스냅샷을 Amazon EC2로 내보내기를 참조하세요.

  > **Note**
  >
  > 웹 사이트나 웹 애플리케이션을 로드 밸런싱해야 하는 경우, Elastic Load Balancing에서 Application Load Balancer를 생성합니다. 자세한 내용은 Elastic Load Balancing User Guide를 참조하세요.

- Lightsail 배포에 연결된 인증서를 분리하려면 사용자 지정 도메인을 비활성화하면 됩니다. 자세한 내용은 Amazon Lightsail 배포용 사용자 지정 도메인 사용 중지를 참조하세요.
- AWS Command Line Interface (AWS CLI)를 사용하여 get-distributions 명령을 실행하여 Lightsail 배포 설정 목록을 가져옵니다. 자세한 내용은 AWS CLI 참조에서 get-distributions를 참조하세요.
- CloudFront 콘솔에 로그인하고 Lightsail 배포와 동일한 구성 설정으로 배포를 생성합니다. 자세한 내용은 Amazon CloudFront 개발자 안내서의 배포 생성을 참조하세요.
- CloudFront 배포에 연결할 인증서를 AWS Certificate Manager c(ACM)로 생성합니다. 자세한 내용은 ACM 사용 설명서의 공인 인증서 요청을 참조하세요.
- 생성한 ACM 인증서를 사용하도록 CloudFront 배포를 업데이트합니다. 자세한 내용은 CloudFront 사용 설명서의 CloudFront 배포 업데이트를 참조하세요.

Lightsail CDN의 용도는 무엇인가요?

Lightsail CDN 배포는 고정 요금 데이터 전송 번들을 사용하여 생성되어 서비스 사용 비용을 단순하고 예측 가능하게 만듭니다. 배포 번들은 한 달분 사용량을 충당할 수 있도록 고안되었습니다. 번들을 자주 업그레이드 또는 다운그레이드하거나 단일 오리진으로 지나치게 많은 배포를 사용하는 등 초과 요금을 피할 목적으로 배포 번들을 사용하는 행위는 정상적인 사용 범위를 벗어나는 것이며 허용되지 않

습니다. 또한, 초당 요청 수가 많거나 대량의 비디오 스트리밍이 필요한 워크로드는 지원되지 않습니다. 이러한 동작으로 인해 데이터 서비스 또는 계정이 스로틀링되거나 일시 정지될 수 있습니다. 자세한 내용은 데이터 전송 비용은 얼마나 됩니까? 단원을 참조하십시오.

Lightsail CDN 배포는 IPv6를 지원하나요?

모든 Lightsail CDN 배포에는 기본적으로 IPv6가 활성화되어 있습니다. 배포 호스트 이름에서 IPv4 및 IPv6 주소를 확인할 수 있습니다. CDN 관리 페이지의 네트워킹(Networking) 탭에서 토글을 사용하여 IPv6를 비활성화할 수 있습니다.

Lightsail CDN 배포를 사용하려면 오리진에서 IPv6를 활성화해야 하나요?

아니요, CDN 배포는 IPv6 및 IPv4 트래픽을 모두 수락하며 백엔드의 오리진과 통신할 때는 IPv4로 원활하게 전환됩니다. 따라서 배포 뒤에 있는 오리진은 듀얼 스택 또는 IPv4 전용 중 하나일 수 있습니다.

데이터베이스 수

Lightsail 관리형 데이터베이스란 무엇인가요?

Lightsail 관리형 데이터베이스는 웹 서버, 메일 서버 등 다른 워크로드 대신 데이터베이스를 전용으로 실행하는 인스턴스입니다. 관리형 데이터베이스에는 사용자가 생성한 여러 데이터베이스가 포함될 수 있으며, 독립 실행형 데이터베이스에 사용하는 동일한 도구 및 애플리케이션을 사용해 액세스할 수 있습니다. Lightsail은 데이터베이스의 기본 인프라와 운영 체제에 대한 보안 및 상태를 유지 관리하므로 인프라 관리에 대한 전문 지식이 없어도 데이터베이스를 실행할 수 있습니다.

일반적인 Lightsail 인스턴스와 같이 Lightsail 관리형 데이터베이스도 시간이 지남에 따라 확장할 수 있는 플랜의 고정된 메모리 양, 컴퓨팅 파워 및 SSD 기반 스토리지와 함께 제공됩니다. Lightsail은 생성 시 선택한 데이터베이스를 자동으로 설치 및 구성합니다.

Lightsail 관리형 데이터베이스로 무엇을 할 수 있나요?

Lightsail 관리형 데이터베이스는 낮은 유지 관리 비용으로 쉽게 클라우드에 데이터를 저장하는 방법을 제공합니다. 관리형 데이터베이스를 새 데이터베이스로 실행하거나, 기존 온프레미스 또는 호스팅된 데이터베이스에서 Lightsail로 마이그레이션하여 실행할 수 있습니다.

또한 데이터베이스를 전용 인스턴스로 분리하여 대량의 트래픽과 더 집약적인 로드를 수용하도록 애플리케이션을 확장할 수도 있습니다. Lightsail 관리형 데이터베이스는 WordPress 및 가장 일반적인

CMS와 같이 단일 인스턴스 이상으로 확장할 때 데이터를 동기화된 상태로 유지해야 하는 상태 저장 애플리케이션에 특히 유용합니다. 관리형 데이터베이스는 Lightsail 로드 밸런서 및 둘 이상의 Lightsail 인스턴스와 연결하여 강력하고 확장된 애플리케이션을 생성할 수 있습니다. Lightsail 고가용성 관리형 데이터베이스 플랜을 사용하면 데이터베이스에 중복성을 추가로 생성하여 애플리케이션에 대한 높은 가동 시간을 보장할 수도 있습니다.

Lightsail에서 무엇을 자동으로 관리해 주나요?

Lightsail은 관리형 데이터베이스와 기본 인프라에 대한 다양한 유지 관리 활동 및 보안을 관리합니다. Lightsail은 데이터베이스를 자동으로 백업하고 데이터베이스 복원 도구를 통해 지난 7일 중 특정 시점으로 복원을 허용하여 데이터 손실이나 구성 요소 장애를 방지합니다. 또한 Lightsail은 보안 강화를 위해 유휴 데이터 및 이동 중인 데이터를 자동으로 암호화하고, 데이터베이스에 쉽고 안전하게 연결하기 위해 데이터베이스 암호를 저장합니다. 유지 관리 측면에서 Lightsail은 설정된 유지 관리 기간 동안 데이터베이스에서 유지 관리를 실행합니다. 이러한 유지 관리에는 최신 마이너 데이터베이스 버전으로의 자동 업그레이드와 기본 인프라 및 운영 체제의 모든 관리가 포함됩니다.

Lightsail은 어떤 종류와 버전의 데이터베이스를 지원하나요?

Lightsail 관리형 데이터베이스는 MySQL 및 PostgreSQL의 최신 메이저 버전을 지원합니다. 현재 MySQL 5.7, MySQL 8.0, PostgreSQL 9, PostgreSQL 10, PostgreSQL 11 및 PostgreSQL 12가 지원됩니다. Lightsail은 각 메이저 버전 옵션에 대한 최신 마이너 버전만 제공합니다.

Lightsail이 제공하는 관리형 데이터베이스 플랜은 무엇인가요?

Lightsail는 표준 및 고가용성 플랜에서 6가지 크기의 관리형 데이터베이스를 제공합니다. 각 플랜은 고정된 스토리지 양 및 월간 데이터 전송 허용량과 함께 제공됩니다. 또한 시간이 지남에 따라 필요한 대로 더 큰 플랜으로 확장하고, 표준 및 고가용성 플랜 간에 전환할 수도 있습니다. 고가용성 플랜은 표준 플랜과 동일한 리소스를 미러링하고, 중복성을 위해 기본 데이터베이스의 별도 가용 영역에서 실행 중인 예비 데이터베이스를 추가로 포함합니다. 데이터 전송 요금에 대한 자세한 내용은 데이터 전송 비용은 얼마나 됩니까? 섹션을 참조하세요.

고가용성 플랜이란 무엇입니까?

Lightsail 관리형 데이터베이스는 표준 및 고가용성 플랜에서 사용할 수 있습니다. 표준 및 고가용성 플랜에는 메모리, 스토리지, 데이터 전송 허용량을 비롯해 동일한 플랜 리소스가 있습니다. 고가용성 플랜은 기본 데이터베이스의 별도 가용 영역에 예비 데이터베이스를 자동으로 생성하고, 데이터를 예비 데이터베이스에 동기식으로 복제하고, 유지 관리 기간 동안 및 인프라 장애 시 예비 데이터베이스에 장애 조치를 제공하여 데이터베이스에 중복성 및 내구성을 추가합니다. 이를 통해 데이터베이스가

Lightsail에서 자동으로 업그레이드/유지 관리되는 경우에도 가동 시간을 보장합니다. 높은 가동 시간이 필요한 실행 중인 프로덕션 애플리케이션 또는 소프트웨어에는 고가용성 플랜을 사용하세요.

Lightsail 관리형 데이터베이스는 어떻게 확장 또는 축소하나요?

데이터베이스의 스냅샷을 생성하고 스냅샷에서 새로운 더 큰 데이터베이스 플랜을 생성하거나, 비상 복원 기능을 사용하여 새로운 더 큰 데이터베이스를 생성하여 Lightsail 관리형 데이터베이스를 확장할 수 있습니다. 두 방법 중 하나를 사용하여 표준 플랜과 고가용성 플랜 간에 전환할 수도 있으며, 그 반대의 경우도 마찬가지입니다. 데이터베이스를 축소할 수는 없습니다. 자세한 내용은 Lightsail의 스냅샷에서 데이터베이스 생성을 참조하세요.

Lightsail 관리형 데이터베이스는 어떻게 백업할 수 있나요?

Lightsail은 데이터를 자동으로 백업하며 새 데이터베이스로 특정 시점의 이 데이터를 복원할 수 있습니다. 자동 백업은 데이터베이스에 대한 무료 서비스이지만 최근 7일 간의 데이터만 저장합니다. 데이터베이스를 삭제하면 자동 백업 레코드가 모두 삭제되며 특정 시점 복원 기능을 더 이상 사용할 수 없습니다. 데이터베이스를 삭제한 후에도 데이터의 백업을 유지하거나 지난 7일 이상의 백업을 복원하려면 수동 스냅샷을 사용하세요.

데이터베이스 관리 페이지에서 Lightsail 관리형 데이터베이스에 대한 수동 스냅샷을 생성할 수 있습니다. 수동 스냅샷에는 데이터베이스의 모든 데이터가 포함되며 영구적으로 저장하려는 데이터에 대한 백업으로 사용할 수 있습니다. 수동 스냅샷을 사용하여 새로운 더 큰 데이터베이스를 생성하거나 표준 및 고가용성 플랜 간에 전환할 수도 있습니다. 수동 스냅샷은 삭제할 때까지 저장되며 매월 $0.05 USD/GB의 요금이 청구됩니다.

Lightsail 관리형 데이터베이스를 삭제하면 데이터는 어떻게 되나요?

Lightsail 관리형 데이터베이스를 삭제하면 데이터베이스 자체와 모든 자동 백업이 삭제됩니다. 데이터베이스를 삭제하기 전에 수동 스냅샷을 생성하지 않으면 이 데이터를 복구할 수 없습니다. 데이터베이스 삭제 중에 Lightsail은 실수로 데이터가 손실되지 않도록 원할 경우 수동 스냅샷을 생성할 수 있는 원클릭 옵션을 제공합니다. 삭제 전에 수동 스냅샷을 생성하는 것은 선택 사항이지만 적극 권장됩니다. 저장된 데이터가 더 이상 필요하지 않은 경우 나중에 수동 스냅샷을 삭제할 수 있습니다.

다른 AWS 리전 가용 영역에서 실행 중인 Lightsail 관리형 데이터베이스에 인스턴스(들)를 연결할 수 있나요?

다른 AWS 리전에서 실행되는 인스턴스에는 Lightsail 관리형 데이터베이스를 사용할 수 없습니다. 하지만 인스턴스의 서로 다른 가용 영역에 대해서는 데이터베이스를 사용할 수 있습니다.

내 Lightsail 관리형 데이터베이스에 데이터를 어떻게 로드하나요?

Lightsail 관리형 데이터베이스로 데이터를 로드하려면 먼저 데이터 가져오기 모드를 활성화해야 합니다. 데이터 가져오기 모드를 활성화하면 원하는 데이터베이스 클라이언트를 사용하여 데이터를 수동으로 업로드하는 단계로 진행할 수 있습니다. 데이터 로드를 완료하면 데이터베이스에 대한 자동 백업 및 로깅 기능이 다시 시작될 수 있도록 데이터 가져오기 모드를 꺼야 합니다. 자세한 내용은 MySQL 데이터베이스로 데이터 가져오기 및 PostgreSQL 데이터베이스로 데이터 가져오기를 참조하세요.

내 Lightsail 관리형 데이터베이스의 데이터에 어떻게 액세스하나요?

표준 SQL 클라이언트 애플리케이션을 사용하여 데이터베이스에 연결하고 데이터를 쿼리할 수 있습니다. GUI 기반 관리 및 쿼리에는 MySQL Workbench를 사용하는 것이 좋습니다. 데이터베이스에 대한 데이터베이스 관리 화면에서 엔드포인트 URL 및 DNS 이름을 비롯한 연결 데이터를 확인할 수 있습니다. 자세한 내용은 MySQL 데이터베이스에 연결 또는 Amazon Lightsail에서 PostgreSQL 데이터베이스에 연결을 참조하세요.

Lightsail 관리형 데이터베이스는 Lightsail 인스턴스와 어떻게 연동되나요?

Lightsail 관리형 데이터베이스를 생성하면 Lightsail 인스턴스를 웹 서버 또는 앱에 대한 다른 전용 워크로드로 사용하여 애플리케이션에 데이터베이스를 즉시 사용할 수 있습니다. Lightsail 인스턴스를 데이터베이스에 연결하려면 데이터베이스 엔드포인트를 사용하고 안전하게 저장된 암호를 참조하여 데이터베이스를 애플리케이션의 코드에서 데이터 스토어로 구성합니다. 데이터베이스 관리 화면에서 연결 데이터를 확인할 수 있습니다. 데이터베이스 구성 파일의 파일 이름과 위치는 애플리케이션마다 다릅니다. 동일한 테이블을 사용하거나 다른 테이블을 사용하여 여러 인스턴스를 데이터베이스 하나에 연결할 수 있습니다.

내 AWS 계정에서 실행 중인 EC2 인스턴스에 Lightsail 관리형 데이터베이스를 연결하려면 어떻게 해야 하나요?

퍼블릭 인터넷을 통해 연결하여 EC2 인스턴스에 Lightsail 관리형 데이터베이스를 연결할 수 있습니다. 모든 AWS 서비스에 연결하면 데이터베이스 데이터 전송 허용량이 소비되고, 퍼블릭 인터넷을 통해 데이터 전송 허용량을 초과하는 AWS 서비스로 전송되는 데이터에는 초과 요금이 발생합니다. Lightsail 관리형 데이터베이스와 EC2 인스턴스 간에는 VPC 피어링을 사용할 수 없습니다. 데이터 전송 요금에 대한 자세한 내용은 데이터 전송 비용은 얼마나 됩니까? 섹션을 참조하세요.

내 Lightsail 관리형 데이터베이스의 퍼블릭 모드와 프라이빗 모드의 차이는 무엇인가요?

기본적으로 Lightsail 관리형 데이터베이스는 프라이빗 모드로 생성됩니다. 이 모드는 Lightsail 인스턴스를 통해서만 액세스할 수 있도록 하여 데이터베이스를 보호합니다. 퍼블릭 인터넷을 통해 소프트웨어 또는 서비스에 연결해야 할 경우 데이터베이스 퍼블릭 모드를 설정할 수 있습니다. 데이터의 보안을 위해 퍼블릭 모드를 장기간 활성화하는 것은 권장되지 않습니다. 데이터베이스 관리 화면에서 언제든지 퍼블릭 모드와 프라이빗 모드 간에 변경할 수 있습니다.

내 Lightsail 관리형 데이터베이스에서 사용하는 포트를 관리할 수 있나요?

아니요. 보안을 위해 Lightsail에서 포트를 자동으로 관리합니다. 퍼블릭 모드에서는 모든 Lightsail 관리형 데이터베이스용 MySQL에 대해 포트 3306을 엽니다. 데이터베이스가 프라이빗 모드인 경우 데이터베이스는 인터넷 네트워크를 통해 Lightsail 계정에서 실행되는 리소스에만 열립니다.

Lightsail 관리형 데이터베이스 서비스는 IPv6를 지원하나요?

Lightsail 관리형 데이터베이스 서비스는 IPv6를 지원하지 않습니다.

도메인

Lightsail 도메인에서 무엇을 할 수 있나요?

Lightsail 도메인을 사용하면 웹 사이트 또는 애플리케이션의 도메인을 등록하고 관리할 수 있습니다. 다른 공급자에 등록된 도메인이 있는 경우 해당 도메인의 관리를 Lightsail로 이전할 수 있습니다. 해당 도메인을 Lightsail 리소스로 지정할 수도 있습니다.

어떤 최상위 도메인(TLD)을 사용할 수 있습니까?

Lightsail은 Amazon Route 53과 동일한 일반 TLD를 사용합니다. 지리적 도메인을 등록하려면 Route 53 콘솔을 사용하는 것이 좋습니다. Route 53를 사용하여 등록된 후 Lightsail 콘솔에서 지리적 도메인을 사용할 수 있습니다. Lightsail에서 지원하는 TLD에 대한 자세한 내용은 Amazon Route 53 개발자 안내서의 Amazon Route 53에 등록할 수 있는 도메인을 참조하세요.

Lightsail을 기존 도메인에 대한 DNS 서비스로 설정할 수 있나요?

다른 DNS 서비스 공급자를 사용하여 등록한 도메인의 DNS 관리를 Lightsail로 이전할 수 있습니다. 자세한 내용은 DNS 영역을 생성하여 도메인의 DNS 레코드 관리를 참조하세요.

Lightsail에서 도메인 등록을 어떻게 시작하나요?

Lightsail에 로그인한 후 Lightsail 콘솔을 사용하여 도메인을 생성하고 관리할 수 있습니다. 자세한 내용은 도메인 등록을 참조하세요.

Lightsail 또는 Route 53 중에서 도메인 등록 선택의 기준은 무엇인가요?

도메인 등록, DNS 영역 생성, 도메인의 트래픽을 Lightsail 리소스로 라우팅하는 등의 작업은 Lightsail에서 실시됩니다. 도메인 등록 연장, 도메인 이전(트래픽 정책 포함), 프라이빗 호스팅 영역 생성과 같은 고급 작업은 Route 53를 사용하는 것이 좋습니다.

도메인을 Lightsail로 이전할 수 있나요?

도메인을 Route 53로 이전할 수 있습니다. 도메인 이전이 완료되면 Lightsail 콘솔에서 도메인을 사용할 수 있습니다. 자세한 내용은 Amazon Route 53에서 Lightsail 도메인 관리를 참조하세요.

도메인에 사용할 수 있는 Lightsail 리소스에는 무엇이 있나요?

Lightsail에 도메인을 등록한 후 도메인을 Lightsail 인스턴스, 컨테이너, 로드 밸런서, 고정 IP 또는 콘텐츠 배포 네트워크(CDN)로 지정할 수 있습니다.

Amazon Elastic Compute Cloud(Amazon EC2)로 Lightsail 리소스 내보내기

Amazon EC2로 내보내기란?

Amazon EC2로 내보내기는 Lightsail 인스턴스의 복사본을 Amazon EC2에 생성할 수 있는 기능입니다. Amazon EC2로 내보낼 때 Amazon EC2에서 제공하는 광범위한 인스턴스 유형, 구성 및 요금 모델 중에서 선택할 수 있으며, 네트워킹, 스토리지 및 컴퓨팅 환경을 훨씬 더 세부적으로 제어할 수 있습니다.

Amazon EC2로 내보내는 이유는 무엇입니까?

Lightsail에서는 예측 가능하며 낮은 가격으로 광범위한 클라우드 기반 애플리케이션을 번들로 간단하게 실행하고 조정할 수 있는 방법을 제공합니다. 또한 Lightsail은 네트워킹 및 액세스 관리와 같은 클라우드 환경 구성을 자동으로 설정합니다.

Amazon EC2로 내보내기를 사용하면 향상된 CPU 성능, 메모리 및 네트워킹 기능이 있는 가상 머신부터 FPGA 및 GPU 기능이 있는 특수한 또는 가속화된 인스턴스에 이르기까지 광범위한 인스턴스 유형에 대해 애플리케이션을 실행할 수 있습니다. 또한 Amazon EC2에서는 자동 관리 및 설정 작업이 적으므로 VPC와 같은 클라우드 환경을 구성하는 방식을 더 잘 제어할 수 있습니다.

Amazon EC2로 내보내기는 어떻게 작동하나요?

시작하려면 Lightsail 인스턴스 또는 블록 스토리지 디스크의 스냅샷을 내보내야 합니다. Amazon EC2 사용에 익숙한 고객은 기존 EC2 AMI 또는 EBS 볼륨에서와 마찬가지로 Amazon EC2 생성 마법사 또는 API를 사용하여 새 Amazon EC2 인스턴스 또는 Amazon EBS 볼륨을 생성할 수 있습니다. 또는 Lightsail은 새 EC2 인스턴스를 쉽게 생성할 수 있도록 안내 Lightsail 콘솔 환경도 제공합니다.

> **Note**
>
> cPanel 및 WHM(CentOS 7) 인스턴스의 스냅샷은 Amazon EC2로 내보낼 수 없습니다.

요금은 어떻게 청구됩니까?

Amazon EC2로 내보내기 기능은 무료로 사용할 수 있습니다. 수동 스냅샷을 Amazon EC2로 내보내면 Lightsail 수동 스냅샷 외에 Amazon EC2 이미지에 대해서도 별도로 요금이 부과됩니다. 또한 Amazon EBS 스토리지 볼륨 및 데이터 전송을 포함하여 시작한 새 Amazon EC2 인스턴스는 Amazon EC2에서 요금을 청구합니다. 새 인스턴스 및 리소스에 대한 요금 세부 정보는 Amazon EC2 요금 페이지를 참조하세요. Lightsail 계정에서 계속 실행되는 Lightsail 리소스는 삭제될 때까지 일반 요금이 부과됩니다.

관리형 데이터베이스나 디스크 스냅샷을 내보낼 수 있습니까?

내보내기 기능을 사용하면 수동 Lightsail 디스크 스냅샷을 내보낼 수 있지만, 현재 관리형 데이터베이스의 수동 스냅샷은 지원되지 않습니다. Amazon EC2 콘솔 또는 API에서 디스크 스냅샷을 Amazon EBS 볼륨으로 리하이드레이션할 수 있습니다.

어떤 Lightsail 리소스를 내보낼 수 있나요?

Amazon EC2로 Lightsail 내보내기 기능은 Linux 및 Windows 인스턴스 스냅샷을 Amazon EC2로 내보낼 수 있도록 설계되었습니다. 또한, 블록 스토리지 디스크 스냅샷을 Amazon EBS로 내보내는 작업을 지원합니다. 현재는 데이터베이스, 컨테이너 서비스, 콘텐츠 전송 네트워크(CDN) 배포, 로드 밸런서, 고정 IP 및 DNS 레코드의 내보내기를 지원하지 않습니다. 또한, 현재로서는 Django, Ghost 및 cPanel & WHM 인스턴스의 스냅샷을 Amazon EC2로 내보낼 수 없습니다.

인스턴스

Lightsail 인스턴스란 무엇인가요?

Lightsail 인스턴스는 AWS 클라우드에 있는 가상 프라이빗 서버(VPS)입니다. Lightsail 인스턴스를 사용하여 데이터를 저장하고 코드를 실행하고 웹 기반 애플리케이션 또는 웹 사이트를 빌드합니다. 인스턴스는 퍼블릭(인터넷) 및 프라이빗(VPC) 네트워킹 모두를 통해 상호 간에 연결하고 다른 AWS 리소스에 연결할 수 있습니다. Lightsail 콘솔에서 바로 손쉽게 인스턴스를 생성 및 관리하고 다른 인스턴스에 연결할 수 있습니다.

Lightsail 요금제란 무엇인가요?

번들이라고도 하는 Lightsail 플랜에는 고정된 양의 메모리(RAM)와 컴퓨팅(vCPU), SSD 기반 스토리지(디스크), 무료 데이터 전송 허용량이 지원되는 가상 서버가 포함됩니다. Lightsail 플랜은 고정 IPv4 주소 및 DNS 관리 기능도 제공합니다. Lightsail 플랜은 온디맨드 방식으로 시간당 요금이 부과되므로, 사용할 때만 요금을 지불하면 됩니다. 데이터 전송 요금에 대한 자세한 내용은 데이터 전송 비용은 얼마나 됩니까? 섹션을 참조하세요.

내 인스턴스에서는 어떤 소프트웨어를 실행할 수 있습니까?

Lightsail은 다양한 운영 체제와 새 Lightsail 인스턴스 생성 시 자동으로 설치되는 애플리케이션 템플릿을 제공합니다. 애플리케이션 템플릿으로는 WordPress, WordPress Multisite, cPanel & WHM, PrestaShop, Django, Drupal, Ghost, Joomla!, Magento, Redmine, LAMP, Nginx(LEMP), MEAN 및 Node.js가 있습니다.

브라우저 내 SSH 또는 자체적인 SSH 클라이언트를 사용하여 인스턴스에 소프트웨어를 추가로 설치할 수 있습니다.

어떤 운영 체제에서 Lightsail을 사용할 수 있나요?

Lightsail은 현재 AlmaLinux OS 9, Amazon Linux 2, Amazon Linux 2,023, CentOS, Debian, FreeBSD, OpenSUSE, Ubuntu 등 7개의 Linux 또는 Unix 유형 배포와 Windows Server 버전 3가지(2,016, 2,019, 2,022)를 지원합니다.

Lightsail 인스턴스를 사용하려면 자체 라이선스를 사용해야 하나요?

cPanel 및 WHM 블루프린트를 제외하고 Lightsail에서 사용 가능한 모든 인스턴스 블루프린트에는 라이선스가 포함됩니다. 이 블루프린트에는 15일 평가판 라이센스가 포함되어 있습니다. 자세한 내용은

빠른 시작 안내서: Amazon Lightsail의 cPanel & WHM을 참조하세요. 다른 모든 인스턴스 블루프린트의 경우 자체 라이선스를 가져올(BYOL) 필요가 없습니다.

Lightsail 인스턴스는 어떻게 만들 수 있나요?

Lightsail에 로그인한 후 Lightsail 콘솔, CLI(명령줄 인터페이스) 또는 API를 사용하여 인스턴스를 생성하고 관리할 수 있습니다.

콘솔에 처음 로그인할 때 인스턴스 생성을 선택합니다. 인스턴스 생성 페이지에서 인스턴스의 위치, 이름, 소프트웨어를 선택할 수 있습니다. 생성을 선택하면 몇 분 내에 새 인스턴스가 자동으로 가동됩니다.

Lightsail 인스턴스는 어떻게 실행되나요?

Lightsail 인스턴스는 웹 서버, 개발자 환경 및 소규모 데이터베이스 사용 사례를 AWS 위해에서 특별히 설계되었습니다. 이러한 워크로드는 CPU의 최대 성능을 자주 또는 지속적으로 사용하지는 않지만 가끔 성능 버스트가 필요합니다. Lightsail은 기본 수준의 CPU 성능을 발휘하면서 기본 수준 이상으로 버스트하는 추가적인 기능을 제공하여 버스트 성능을 발휘할 수 있는 인스턴스를 사용합니다. 이런 설계를 통해 필요한 성능을 필요한 시점에 얻는 동시에, 성능이 변하는 상황이나 다른 환경에서 과다 구독 시 흔히 겪을 수 있는 다른 일반적인 부작용으로부터 사용자를 보호할 수 있습니다.

비디오 인코딩과 같은 애플리케이션이나 HPC 애플리케이션을 위해 일관되게 높은 CPU 성능을 보유하면서도 고도로 구성 가능한 환경과 인스턴스가 필요하다면 Amazon EC2를 사용하는 것이 좋습니다.

인스턴스가 버스팅 중인지 어떻게 알 수 있습니까?

인스턴스의 CPU 사용률 지표 그래프에는 지속 가능 영역과 버스트 가능 영역이 표시됩니다. Lightsail 인스턴스는 시스템 운영에 영향을 주지 않고 지속 가능 영역에서 무기한으로 작동할 수 있습니다. 부하가 큰 경우 인스턴스가 버스트 가능 영역에서 작동하기 시작할 수 있습니다. 버스트 가능 영역에서 작동하는 동안에는 인스턴스가 상대적으로 더 많은 CPU 주기를 사용합니다. 따라서 제한된 기간 동안만 이 영역에서 작동할 수 있습니다. 자세한 내용은 Amazon Lightsail의 인스턴스 지표 보기를 참조하세요.

인스턴스의 CPU 사용률이 지속 가능 영역에서 버스팅 영역으로 전환될 때 알림을 받으려면 지표 경보를 추가합니다. 자세한 내용은 Amazon Lightsail의 인스턴스 지표 경보 생성을 참조하세요.

Lightsail 인스턴스에 어떻게 연결하나요?

Lightsail은 브라우저에서 클릭 한 번으로 인스턴스의 터미널에 바로 안전하게 연결되는 기능을 제공하며, Linux/Unix 기반 인스턴스에 대한 SSH 액세스와 Windows 기반 인스턴스에 대한 RDP 액세스도

지원합니다. 클릭 한 번으로 연결하려면 인스턴스 관리 화면을 시작하고 SSH를 사용하여 연결 또는 RDP를 사용하여 연결을 선택합니다. 새 브라우저 창이 열리면서 해당 인스턴스에 자동으로 연결됩니다.

자체 클라이언트를 사용하여 Linux/Unix 기반 인스턴스에 연결하는 방법을 선호할 경우, Lightsail은 SSH 키 저장 및 관리 작업을 자동으로 수행하고 SSH 클라이언트에서 사용할 보안 키를 제공합니다.

내 인스턴스는 어떻게 백업할 수 있습니까?

데이터를 백업하려는 경우 Lightsail 콘솔 또는 API를 사용하여 인스턴스의 수동 스냅샷을 생성하거나, 자동 스냅샷을 활성화하여 Lightsail이 매일 스냅샷을 생성하도록 할 수 있습니다. 장애가 발생하거나 불량 코드가 배포된 경우 이후에 인스턴스 스냅샷을 사용하여 새로운 인스턴스를 만들 수 있습니다. 자세한 내용은 스냅샷을 참조하세요.

내 요금제를 업그레이드할 수 있습니까?

예. 인스턴스의 스냅샷을 사용하여 더 큰 인스턴스를 새로 만들 수 있습니다. 자세한 내용은 스냅샷을 참조하세요.

내 AWS 계정의 다른 리소스에 Lightsail 인스턴스를 연결하려면 어떻게 해야 하나요?

VPC 피어링을 사용하여 Lightsail 인스턴스를 AWS 계정의 Amazon VPC 리소스에 비공개로 연결할 수 있습니다. Lightsail 계정 페이지에서 VPC 피어링 활성화(Enable VPC peering)를 선택하면 됩니다. Lightsail에서 이 작업을 수행할 수 있습니다. VPC 피어링이 활성화되면 프라이빗 IPs를 사용하여 기본 Amazon VPC의 다른 AWS 리소스를 처리할 수 있습니다. 지침은 여기에서 확인하세요.

> **Note**
>
> 와 VPC 피어링이 2013년 12월 이전에 생성된 work. AWS accountsLightsail에 기본 VPC가 없으므로 AWS 계정에 기본 Amazon VPC를 설정해야 합니다. 기본 VPC의 설정에 대한 자세한 내용은 여기에서 확인할 수 있습니다.

내 인스턴스를 중지하는 것과 삭제하는 것은 어떤 차이가 있나요?

인스턴스를 중지하면 현재 상태에서 전원이 꺼지며 언제든 다시 시작할 수 있습니다. 인스턴스를 중지하면 퍼블릭 IPv4 주소가 해제되므로, 중지 및 시작 후에도 동일한 IP를 유지해야 하는 인스턴스에는

고정 IPv4 주소를 사용하는 것이 좋습니다. 인스턴스에 연결된 퍼블릭 IPv6 주소는 인스턴스가 중지 및 시작되더라도 변경되지 않습니다.

인스턴스를 삭제하는 것은 안전하지 않은 작업을 수행하는 것입니다. 인스턴스 스냅샷을 생성하지 않았다면 모든 인스턴스가 손실되어 다시는 복구할 수 없게 됩니다. 자동 스냅샷도 수동 스냅샷으로 복사하여 보관하지 않으면 인스턴스와 함께 삭제됩니다. 인스턴스의 퍼블릭 및 프라이빗 IP 주소 역시 해제됩니다. 해당 인스턴스에 고정 IPv4 주소를 사용할 경우, 고정 IPv4 주소는 분리되지만 계정에는 그대로 유지됩니다.

로드 밸런서

Lightsail 로드 밸런서로 어떤 작업을 수행할 수 있나요?

Lightsail 로드 밸런서를 사용하면 가용성이 뛰어난 웹 사이트와 애플리케이션을 빌드할 수 있습니다. 서로 다른 가용 영역에 있는 인스턴스 전체로 트래픽을 분산하고 정상적인 대상 인스턴스로만 트래픽을 보냄으로써 Lightsail 로드 밸런서는 인스턴스 장애나 데이터 센터 가동 중단으로 인해 애플리케이션이 중단되는 위험을 줄일 수 있습니다. Lightsail 로드 밸런서와 복수의 대상 인스턴스를 사용하면 웹 사이트 또는 애플리케이션에서 피크 로드 기간에 웹 트래픽 증가를 수용하고 방문자에게 계속 좋은 성능을 제공할 수 있습니다.

또한, Lightsail 로드 밸런서를 사용하여 안전한 애플리케이션을 빌드하고 HTTPS 트래픽을 허용할 수 있습니다. Lightsail은 SSL/TLS 인증서를 요청, 프로비저닝 및 유지 관리하는 복잡성을 덜어줍니다. 내장된 인증서 관리에서 사용자를 대신해 자동으로 인증서를 요청하고 갱신하며 이를 로드 밸런서에 추가합니다.

서로 다른 AWS 리전 가용 영역에 있는 인스턴스에서 로드 밸런서를 사용할 수 있나요?

다른 AWS 리전에서 실행되는 인스턴스에는 로드 밸런서를 사용할 수 없습니다. 하지만 서로 다른 가용 영역에 있는 대상 인스턴스에는 로드 밸런서를 사용할 수 있습니다. 실제로 대상 인스턴스를 여러 가용 영역에 분산하여 애플리케이션의 가용성을 극대화하는 것이 좋습니다.

Lightsail 로드 밸런서는 트래픽 스파이크를 어떻게 처리하나요?

Lightsail 로드 밸런서는 사용자가 수동으로 조정하지 않아도 애플리케이션의 트래픽 급증을 처리하도록 자동으로 조정합니다. 애플리케이션에서 트래픽이 일시적으로 급증할 경우 Lightsail 로드 밸런서는 자동으로 조정하여 Lightsail 인스턴스로 트래픽을 계속 효율적으로 보냅니다. Lightsail 로드 밸런서는

트래픽 급증을 손쉽게 관리할 수 있도록 설계되었지만, 계속해서 높은 트래픽 볼륨 레벨을 겪고 있는 애플리케이션은 성능이 저하되거나 스로틀링될 수 있습니다. 애플리케이션이 데이터를 시간당 5GB 이상 지속적으로 관리하거나, 애플리케이션에 여러 연결(시간당 새 연결 400k 초과, 활성화 및 동시 연결 15k 초과)이 있을 경우, 애플리케이션 로드 밸런싱이 있는 Amazon EC2 사용을 권장합니다.

Lightsail 로드 밸런서는 어떻게 트래픽을 내 대상 인스턴스로 라우팅하나요?

Lightsail 로드 밸런서는 라운드 로빈 알고리즘에 따라 트래픽을 정상적인 대상 인스턴스로 보냅니다.

Lightsail은 내 대상 인스턴스가 정상인지 어떻게 알 수 있나요?

로드 밸런서를 생성하고 인스턴스를 연결하면 Lightsail에서 상태 확인 요청을 웹 애플리케이션의 루트로 보냅니다. Lightsail이 ping을 실행하는 경로(일반 파일 또는 웹 페이지 URL)를 지정하여 위치를 사용자 지정할 수 있습니다. 이 경로를 통해 대상 인스턴스에 도달할 수 있으면, Lightsail에서 트래픽을 해당 인스턴스로 라우팅합니다. 대상 인스턴스 중 하나가 응답이 없으면 상태 확인이 실패하고, Lightsail이 해당 인스턴스로 트래픽을 라우팅하지 않습니다. 상태 확인에 대해 자세히 알아보기

내 로드 밸런서에 인스턴스를 몇 개나 연결할 수 있습니까?

Lightsail 계정 인스턴스 할당량 내에서 원하는 수만큼 대상 인스턴스를 로드 밸런서에 추가할 수 있습니다.

하나의 인스턴스를 여러 로드 밸런서에 할당할 수 있습니까?

예. Lightsail에서는 필요한 경우 인스턴스를 둘 이상의 로드 밸런서에 대상 인스턴스로 추가할 수 있습니다.

내 로드 밸런서를 삭제하면 대상 인스턴스는 어떻게 됩니까?

로드 밸런서를 삭제하면 연결된 대상 인스턴스는 계속 정상적으로 실행되고 Lightsail 콘솔에 정규 Lightsail 인스턴스로 표시됩니다. 로드 밸런서를 삭제한 후에 예전의 대상 인스턴스 중 하나로 트래픽을 보내려면 DNS 레코드를 업데이트해야 한다는 점에 유의하세요.

세션 지속성이란 무엇입니까?

세션 지속성을 사용하면 로드 밸런서가 방문자의 세션을 특정 대상 인스턴스에 바인딩할 수 있습니다. 이렇게 하면 세션 중에 사용자로부터 수신되는 모든 요청이 동일한 대상 인스턴스로 전송됩니다. Lightsail에서는 데이터 일관성을 위해 방문자가 동일한 대상 인스턴스에 도달해야 하는 애플리케이션

에 대해 세션 지속성을 지원합니다. 예를 들어 사용자 인증이 필요한 많은 애플리케이션에서 세션 지속성 기능을 활용할 수 있습니다. 로드 밸런서를 생성한 후 로드 밸런서 관리 화면에서 특정 로드 밸런서에 대한 세션 지속성 기능을 활성화할 수 있습니다. 자세한 내용은 로드 밸런서에 대한 세션 지속성 활성화를 참조하세요.

Lightsail 로드 밸런서에는 어떤 종류의 연결을 지원하나요?

Lightsail 로드 밸런서는 HTTP 및 HTTPS 연결을 지원합니다.

Lightsail 로드 밸런서가 IPv6를 지원하나요?

2021년 1월 12일 이후에 생성된 Lightsail 로드 밸런서는 기본적으로 듀얼 스택 모드로 운영됩니다. 즉, IPv4 및 IPv6 프로토콜을 통한 클라이언트 트래픽을 모두 수락합니다. 로드 밸런서 관리 페이지의 네트워킹 탭에서 토글 버튼을 통해 이 날짜 이전에 생성된 로드 밸런서의 IPv6를 활성화할 수 있습니다. 이 토글 버튼을 사용하여 어떤 로드 밸런서에서도 IPv6를 비활성화할 수 있습니다.

IPv6를 활성화한 로드 밸런서를 사용하려면 로드 밸런서 뒤에 인스턴스에 IPv6를 활성화해야 하나요?

아니요, 로드 밸런서는 IPv4 및 IPv6 트래픽을 모두 수락하고 백엔드의 인스턴스와 통신할 때는 IPv4로 원활하게 전환됩니다. 따라서 로드 밸런서 뒤의 인스턴스는 듀얼 스택 또는 IPv4 전용 중 하나일 수 있습니다.

수동 및 자동 스냅샷

스냅샷이란 무엇입니까?

스냅샷은 인스턴스, 데이터베이스 또는 블록 스토리지 디스크의 특정 시점 백업입니다. 언제라도 리소스의 스냅샷을 생성할 수 있습니다. 또는 Lightsail에서 자동으로 스냅샷을 생성하도록 인스턴스 및 디스크에서 자동 스냅샷을 활성화할 수 있습니다. 스냅샷을 기준선으로 사용하여 새 리소스를 생성하거나 데이터를 백업할 수 있습니다. 스냅샷은 스냅샷을 생성한 시점부터 리소스를 복원하는 데 필요한 모든 데이터를 포함합니다. 스냅샷에서 리소스를 생성하여 리소스를 복원하는 경우 새 리소스는 스냅샷을 생성하는 데 사용된 원래 리소스와 정확히 동일한 복제본으로 시작됩니다.

Lightsail 인스턴스, 디스크 및 데이터베이스의 스냅샷을 수동으로 생성할 수 있습니다. 또는 자동 스냅샷을 사용하여 Lightsail에서 인스턴스 및 디스크의 일일 스냅샷을 자동으로 생성할 수 있습니다. 자세한 내용은 스냅샷을 참조하세요.

자동 스냅샷이란 무엇입니까?

자동 스냅샷은 Amazon Lightsail에서 Linux/Unix 인스턴스의 일일 스냅샷을 예약하는 기능입니다. 하루 중 시간을 선택할 수 있습니다. 그러면, Lightsail은 매일 선택한 시간에 자동으로 스냅샷을 생성하며, 항상 최대 7개의 자동 스냅샷을 보관합니다. 스냅샷은 무료로 활성화할 수 있으며, 스냅샷이 사용하는 실제 스토리지에 대해서만 비용이 청구됩니다.

수동 스냅샷과 자동 스냅샷의 차이는 무엇입니까?

자동 스냅샷은 Amazon EC2로 직접 내보내거나 태그를 지정할 수 없습니다. 그러나 자동 스냅샷을 복사하여 수동 스냅샷으로 전환할 수 있습니다. 자동 스냅샷을 수동 스냅샷으로 복사하려면 자동 스냅샷의 컨텍스트 메뉴에서 보관을 선택하여 수동 스냅샷으로 복사합니다.

어떤 리소스가 스냅샷을 지원합니까?

인스턴스, 데이터베이스, 디스크의 수동 스냅샷을 생성할 수 있습니다.

Lightsail 콘솔, Lightsail API 또는를 사용하는 Linux 또는 Unix 인스턴스 AWS CLI와 Lightsail API만 사용하는 디스크에 대해 자동 스냅샷을 활성화할 수 있습니다 AWS CLI. 현재 자동 스냅샷은 Windows 인스턴스 또는 관리형 데이터베이스에 지원되지 않습니다.

스냅샷을 얼마나 오래 저장할 수 있습니까?

수동 스냅샷은 사용자가 삭제할 때까지 저장됩니다. 자세한 내용은 Amazon Lightsail의 스냅샷 삭제를 참조하십시오.

자동 스냅샷은 새로운 자동 스냅샷으로 교체될 때까지 저장됩니다. Lightsail은 가장 오래된 스냅샷을 삭제하고 최신 스냅샷으로 교체하기 전에 최신 자동 스냅샷 7개를 저장합니다. 그러나 특정 자동 스냅샷을 수동 스냅샷으로 복사하여 유지할 수 있습니다. 자세한 내용은 Amazon Lightsail에서 인스턴스 또는 디스크의 자동 스냅샷 유지를 참조하십시오. 계정에 저장된 자동 스냅샷에 대해서는 스냅샷 스토리지 요금이 청구됩니다.

자동 스냅샷을 어떻게 활성화합니까?

Linux 또는 Unix 인스턴스를 생성할 때 또는 인스턴스가 실행되기 시작한 이후에 Lightsail 콘솔, Lightsail API 또는 AWS CLI 를 사용하여 자동 스냅샷을 활성화할 수 있습니다.

디스크를 생성할 때 또는 생성한 이후에 디스크의 자동 스냅샷을 활성화할 수도 있습니다. 그러나 Lightsail API 또는 AWS CLI만 사용해야 합니다.

자세한 내용은 Amazon Lightsail에서 인스턴스 또는 디스크의 자동 스냅샷 활성화 또는 비활성화를 참조하십시오.

언제 자동 스냅샷이 생성됩니까?

자동 스냅샷을 활성화하면 기본 시간은 리소스가 있는 AWS 리전 에 따라 설정됩니다. 자동 스냅샷을 원하는 시간(시간 단위)으로 변경할 수 있습니다. 자세한 내용은 Amazon Lightsail에서 인스턴스 또는 디스크의 자동 스냅샷 시간 변경을 참조하십시오.

몇 개의 스냅샷을 저장할 수 있습니까?

원하는 만큼 수동 스냅샷을 저장할 수 있습니다. 그러나 가장 오래된 스냅샷이 최신 스냅샷으로 교체되기 전에 가장 최근의 7개 자동 스냅샷만 저장됩니다.

스냅샷 요금은 어떻게 청구됩니까?

Lightsail 계정에 저장된 스냅샷 비용만 지불하면 됩니다. Lightsail 스냅샷(수동 및 자동) 저장 비용은 월 $0.05 USD/GB입니다.

자동 스냅샷을 비활성화하면 스냅샷이 손실됩니까?

아니요. 자동 스냅샷을 비활성화하면 Lightsail이 일별 스냅샷 생성을 중지하고, 기존의 자동 스냅샷이 유지됩니다. 자동 스냅샷을 다시 활성화하면 Lightsail이 일별 스냅샷 생성을 재개하여 가장 오래된 스냅샷을 삭제하고 최신 스냅샷으로 교체합니다.

자동 스냅샷이 교체되지 않도록 하려면 어떻게 해야 합니까?

특정 자동 스냅샷을 수동 스냅샷으로 복사하여 유지할 수 있습니다. 자세한 내용은 Amazon Lightsail에서 인스턴스 또는 디스크의 자동 스냅샷 유지를 참조하십시오.

자동 스냅샷을 삭제할 수 있습니까?

자동 스냅샷의 컨텍스트 메뉴에서 삭제를 선택하면 언제든 자동 스냅샷을 삭제할 수 있습니다. 자세한 내용은 자동 인스턴스 스냅샷 삭제를 참조하세요.

스냅샷을 사용하려면 어떻게 해야 합니까?

스냅샷을 기준으로 사용하거나, 원래 리소스에 문제가 있는 경우 새 리소스를 생성할 수 있습니다. 자세한 내용은 스냅샷을 참조하세요.

스냅샷을 Amazon EC2로 내보내 해당 서비스 내에 새로운 리소스를 생성할 수도 있습니다. 자세한 내용은 스냅샷을 Amazon EC2로 내보내기를 참조하세요.

리소스 상태 지표 및 경보

지표란 무엇입니까?

Lightsail은 인스턴스, 데이터베이스 및 로드 밸런서에 대한 지표 데이터를 보고합니다. 일부 지표에는 인스턴스의 CPU 사용률, 인바운드 및 아웃바운드 네트워크 트래픽 양, 시스템 및 인스턴스 오류 수, 데이터베이스 디스크 대기열 깊이, 데이터베이스 여유 스토리지 공간, 로드 밸런서 오류 수, 로드 밸런서 응답 시간 등이 포함됩니다. 지표를 통해 리소스의 안정성, 가용성 및 성능을 모니터링하고 유지할 수 있습니다. 리소스에서 지표 데이터를 정기적으로 모니터링하고 수집하면 다중 지점 오류가 발생할 경우 보다 쉽게 디버깅할 수 있습니다. 자세한 내용은 리소스 지표를 참조하세요.

경보란 무엇입니까?

Lightsail에서 인스턴스, 데이터베이스 및 로드 밸런서에 대한 지표를 감시하는 경보를 생성할 수 있습니다. 지정한 임계값에 상대적인 지표 값을 기준으로 경보 알림을 구성할 수 있습니다. 자세한 내용은 경보 섹션을 참조하세요.

알림은 Lightsail 콘솔에 표시되는 배너, 이메일 주소로 전송되는 이메일, 휴대폰 번호로 전송되는 SMS 문자 메시지 형태로 사용할 수 있습니다. 알림에 대한 자세한 내용은 알림을 참조하세요.

경보를 몇 개까지 추가할 수 있습니까?

인스턴스, 데이터베이스 및 로드 밸런서에 사용할 수 있는 각 지표에 대해 두 개의 경보를 구성할 수 있습니다. 자세한 내용은 경보 섹션을 참조하세요.

네트워킹

Lightsail에서 IP 주소를 사용하려면 어떻게 해야 하나요?

각 Lightsail 인스턴스는 자동으로 프라이빗 IPv4 주소, 퍼블릭 IPv4 주소 또는 퍼블릭 IPv6 주소를 가져옵니다. 2021년 1월 12일 이전에 생성된 인스턴스의 경우 IPv6를 수동으로 활성화해야 합니다. 프라이빗 IP를 사용하여 Lightsail 인스턴스와 AWS 리소스 간에 데이터를 비공개로 무료로 전송할 수 있습니다. 퍼블릭 IP를 사용하면 인터넷(예: 등록된 도메인 이름이나 로컬 컴퓨터의 SSH 또는 RDP 연결 사용)에서 인스턴스에 연결할 수 있습니다. 정적 IPv4 주소를 인스턴스에 연결하여 인스턴스가 중지되고

시작되더라도 변경되지 않는 IPv4 주소를 퍼블릭 IPv4 주소 대신 사용할 수도 있습니다. 인스턴스에 할당된 IPv6 주소는 인스턴스가 삭제되거나 인스턴스에서 IPv6를 비활성화하여 IPv6 주소를 수동으로 해제할 때까지 변경되지 않습니다.

Lightsail은 IPv6 전용 인스턴스를 지원하나요?

네, Lightsail 인스턴스는 듀얼 스택(IPv4 및 IPv6) 및 IPv6 전용 구성을 지원합니다.

고정 IP란 무엇입니까?

고정 IP는 Lightsail 계정 전용으로 사용되는 고정된 퍼블릭 IP 주소입니다. 고정 IPv4 주소를 인스턴스에 할당하여 퍼블릭 IPv4를 대체할 수 있습니다. 인스턴스를 다른 인스턴스로 바꾸려는 경우 고정 IP를 새 인스턴스에 다시 할당할 수 있습니다. 이런 식으로, 인스턴스를 바꾸고 싶을 때마다 새 IP 주소를 지정하려고 DNS 레코드 같은 외부 시스템을 다시 구성할 필요가 없습니다. Lightsail은 현재 IPv4 전용 고정 IP를 지원합니다. 고정 IPv6 주소는 사용할 수 없습니다. 그러나 인스턴스에 할당된 IPv6 주소는 인스턴스가 삭제되거나 인스턴스에서 IPv6를 비활성화하여 IPv6 주소를 수동으로 해제할 때까지 변경되지 않습니다.

인스턴스에 연결할 수 있는 고정 IP는 몇 개인가요?

한 번에 한 인스턴스에 1개의 고정 IP만 연결할 수 있습니다.

DNS 레코드란?

DNS는 `www.example.com`과 같이 사람이 읽을 수 있는 이름을 컴퓨터 간 연결에 사용되는 `192.0.2.1`과 같은 영숫자 IP 주소로 변환하도록 전역으로 분산된 서비스입니다. Lightsail을 사용하면 `photos.example.com`과 같이 등록된 도메인 이름을 Lightsail 인스턴스의 퍼블릭 IP로 손쉽게 매핑할 수 있습니다. 이런 식으로, 사용자가 브라우저에 `example.com`과 같이 사람이 읽을 수 있는 이름을 입력하면 Lightsail에서 입력된 주소를 사용자를 이동시키고 싶은 인스턴스의 IP로 자동 변환합니다. 이런 변환을 각각 DNS 쿼리라고 합니다.

Lightsail에서 도메인을 사용하기 위해서는 먼저 도메인을 등록해야 한다는 점을 알아야 합니다. Lightsail 또는 원하는 DNS 등록 대행자를 사용하여 새 도메인을 등록할 수 있습니다.

내 인스턴스에 대한 방화벽 설정을 관리할 수 있습니까?

예. Lightsail 방화벽을 사용하여 인스턴스의 데이터 트래픽을 제어할 수 있습니다. Lightsail 콘솔에서 다양한 유형의 트래픽에 대해 공개적으로 액세스할 수 있는 인스턴스 포트에 대한 규칙을 설정할 수 있습니다.

객체 스토리지 및 버킷

Lightsail 객체 스토리지로 할 수 있는 작업은 무엇인가요?

Lightsail 객체 스토리지 서비스의 버킷에 이미지, 동영상 및 HTML 파일과 같은 정적 콘텐츠를 저장할 수 있습니다. 버킷에 저장된 객체를 웹 사이트 및 애플리케이션과 함께 사용할 수 있습니다. 몇 번의 클릭만으로 Lightsail CDN 배포에 Lightsail 객체 스토리지를 연결할 수 있으므로, 쉽고 빠르게 전 세계 고객에게 콘텐츠를 제공할 수 있습니다. 또한, 저렴한 비용의 안전한 백업 솔루션으로 사용할 수 있습니다. 자세한 내용은 객체 스토리지를 참조하세요.

Lightsail 객체 스토리지의 비용은 어떻게 됩니까?

Lightsail 객체 스토리지에는를 사용할 수 있는 모든 AWS 리전에 세 가지 고정 가격 번들Lightsail이 있습니다. 첫 번째 번들은 한 달에 1 USD이며 처음 12개월 동안은 무료입니다. 이 번들에서는 5GB의 스토리지 용량과 25GB의 데이터 전송량을 제공합니다. 두 번째 번들은 월 3 USD이며, 100GB의 스토리지 용량과 250GB의 데이터 전송량을 제공합니다. 마지막 세 번째 번들은 월 5 USD이며 250GB의 스토리지 용량과 500GB의 데이터 전송량을 제공합니다. 번들 데이터 전송 허용량은 버킷 외부로의 데이터 전송만 가산하므로, Lightsail 객체 스토리지를 사용하면 버킷으로 데이터를 무제한 전송할 수 있습니다.

Lightsail 버킷과 연결된 플랜을 변경할 수 있나요?

예, 월별 AWS 결제 주기 내에 개별 Lightsail 버킷의 스토리지 플랜을 한 번 변경할 수 있습니다.

Lightsail 객체 스토리지에서 Amazon S3로 객체를 복사할 수 있나요?

예, Lightsail 객체 스토리지에서 Amazon S3로 복사할 수 있습니다. 자세한 내용은 AWS Premium Support 지식 센터의 한 Amazon S3 버킷에서 다른 버킷으로 모든 객체를 복사하려면 어떻게 해야 하나요?를 참조하세요.

Lightsail 객체 스토리지를 시작하려면 어떻게 해야 하나요?

Lightsail 객체 스토리지를 사용하려면 먼저 데이터를 저장하는 데 사용할 버킷을 생성해야 합니다. 자세한 내용은 버킷 생성을 참조하세요. 버킷을 설정하여 실행한 후에는 Lightsail 콘솔을 사용하여 파일을 업로드하거나 로그 또는 기타 애플리케이션 데이터와 같은 콘텐츠를 버킷에 넣도록 애플리케이션을 구성하여 버킷에 객체를 추가할 수 있습니다. 또는 AWS Command Line Interface (AWS CLI)를 통해 Lightsail 객체 스토리지를 사용하기 시작할 수도 있습니다.

버킷에 객체를 업로드하려면 어떻게 해야 하나요?

이미지나 다른 정적 파일과 같은 객체를 버킷에 업로드하려면 상단 탐색 탭 '객체(Objects)'에서 '업로드(Upload)'를 선택하고 컴퓨터에서 올바른 파일 또는 디렉터리를 선택합니다. 아니면 데스크톱에서 파일과 디렉터리를 Lightsail 객체 스토리지 콘솔의 표시된 영역으로 끌어다 놓을 수도 있습니다.

내 버킷에 대한 퍼블릭 액세스를 차단할 수 있나요?

Lightsail 버킷 및 객체는 기본적으로 비공개로 설정되므로, 적절한 권한이 있는 사용자만 버킷과 객체에 액세스할 수 있습니다. 사용자는 이 기본 설정을 변경하여 프라이빗 버킷에 있는 개별 객체를 공개 및 읽기 전용으로 설정하거나 전체 버킷을 공개 및 읽기 전용으로 설정할 수 있습니다. 사용자가 버킷이나 객체를 공개하면 전 세계 모든 사람이 버킷의 내용을 읽을 수 있습니다. 자세한 내용은 버킷 권한을 참조하세요.

버킷에 프로그래밍 방식으로 액세스하려면 어떻게 해야 하나요?

버킷에 프로그래밍 방식으로 액세스하려면 액세스 키나 역할을 사용하면 됩니다. 먼저 Lightsail 콘솔에서 프로그래밍 방식으로 연결할 버킷을 선택합니다. 둘째, 권한 탭에서 액세스 키를 생성하거나 Lightsail 인스턴스에 역할을 할당한 다음, 버킷을 사용하도록 웹 사이트 또는 애플리케이션 코드를 구성합니다. 이 동작은 웹 사이트 또는 애플리케이션에서 객체 스토리지를 사용하는 방법에 따라 달라질 수 있습니다. 자세한 내용은 버킷 권한을 참조하세요.

버킷을 다른 AWS 계정과 공유하려면 어떻게 해야 하나요?

Lightsail를 사용하면 버킷 관리 페이지의 교차 계정 액세스 섹션에서 지정한 AWS 계정 ID와 버킷에 대한 액세스를 공유할 수 있으므로 교차 계정 공유가 쉬워집니다. AWS 계정 ID를 지정하면 해당 계정은 버킷에 대한 읽기 전용 액세스 권한을 갖게 됩니다. 자세한 내용은 버킷 권한을 참조하세요.

버전 관리란 무엇인가요?

버전 관리를 사용하면 버킷에 있는 모든 객체 스토리지의 버전 전체를 보존, 검색 및 복원할 수 있으므로 실수로 덮어쓰거나 삭제하는 상황에 대비해서 보호 수준을 높일 수 있습니다. 자세한 내용은 버킷의 객체 버전 사용 설정 및 사용 중지를 참조하세요.

Lightsail 버킷을 Lightsail CDN 배포에 연결하려면 어떻게 해야 하나요?

몇 번의 클릭만으로 Lightsail CDN 배포에 Lightsail 객체 스토리지를 연결할 수 있으므로, 쉽고 빠르게 전 세계 고객에게 콘텐츠를 제공할 수 있습니다. 이렇게 하려면 Lightsail CDN 배포를 생성하고 Lightsail CDN 배포의 오리진으로 Lightsail 버킷을 선택하면 됩니다. 자세한 내용은 Lightsail 콘텐츠 전송 네트워크 배포와 함께 Amazon Lightsail 버킷 사용을 참조하세요.

Lightsail 객체 스토리지 서비스에는 어떤 제한이 있나요?

Lightsail 객체 스토리지 서비스에서 계정당 최대 20개의 버킷을 생성할 수 있습니다. 버킷에 저장할 수 있는 객체 수에는 제한이 없습니다. 모든 객체를 하나의 버킷에 저장하거나, 여러 버킷에 저장할 수 있습니다.

Lightsail 객체 스토리지가 모니터링 및 알림을 지원하나요?

Lightsail 객체 스토리지를 사용하면 버킷 내에서 사용된 총 공간과 버킷 내의 객체 수에 대한 지표를 쉽게 확인할 수 있습니다. 이러한 지표를 기반으로 하는 알림도 지원됩니다. 자세한 내용은 Amazon Lightsail에서 버킷의 지표 보기 및 버킷 지표 경보 생성을 참조하세요.

Lightsail의 태그

태그란 무엇입니까?

태그는 Lightsail 리소스에 할당하는 레이블입니다. 각 태그는 사용자가 정의하는 키와 값으로 구성됩니다. 태그 값은 선택 사항이므로 Lightsail 콘솔에서 리소스를 필터링하기 위해 "키 전용" 태그를 생성하도록 선택할 수 있습니다.

Lightsail에서 태그를 사용하려면 어떻게 해야 하나요?

태그를 사용하여 Lightsail 콘솔 및 API에서 리소스를 그룹화 및 필터링하고, 청구서의 비용을 추적하고, 액세스 관리 규칙을 통해 리소스를 보거나 수정할 수 있는 사용자를 규제할 수 있습니다. 리소스에 태그를 지정하여 다음 작업을 수행할 수 있습니다.

- 구성 - Lightsail 콘솔 및 API 필터를 사용하여 지정한 태그에 따라 리소스를 보고 관리합니다. 이 기능은 지정한 태그에 따라 특정 리소스를 빠르게 식별할 수 있으므로 동일한 유형의 리소스가 많을 때 유용합니다.
- 비용 할당 - 리소스에 태그를 지정하고 결제 콘솔에 "비용 할당 태그"를 생성하여 다양한 프로젝트 또는 사용자에 대해 비용을 추적하고 할당합니다. 예를 들어, 프로젝트 또는 클라이언트별로 결제를 분할하고 비용을 이해할 수 있습니다.
- 액세스 관리 - AWS 계정에 액세스할 수 있는 사용자가 AWS Identity and Access Management 정책을 사용하여 Lightsail 리소스를 편집, 생성 및 삭제할 수 있는 방법을 제어합니다. 이렇게 하면 다른 사용자에게 Lightsail 리소스에 대한 모든 액세스 권한을 제공하지 않고 다른 사용자와 더 쉽게 공동 작업할 수 있습니다.

Lightsail에서 태그 사용에 대한 자세한 내용은 태그를 참조하세요.

어떤 리소스에 태그를 지정할 수 있습니까?

Lightsail은 현재 다음 리소스에 대한 태그 지정을 지원합니다.

- 인스턴스(Linux 및 Windows)
- 컨테이너 서비스 수
- 블록 스토리지 디스크
- 로드 밸런서
- 데이터베이스 수
- DNS 영역
- 인스턴스, 디스크 및 데이터베이스의 수동 스냅샷

수동 스냅샷은 태그를 지원하지만 Lightsail API 또는를 사용하여 스냅샷 AWS CLI 에 태그를 지정해야 합니다. Lightsail 콘솔을 사용하여 태그가 지정된 인스턴스, 디스크 또는 데이터베이스의 수동 스냅샷을 생성하면 원본 리소스와 동일한 태그가 수동 스냅샷에 자동으로 부여됩니다. Lightsail 콘솔을 사용하여 태그가 지정된 수동 스냅샷에서 새 리소스를 생성할 때 이 태그를 편집할 수 있습니다.

자동 스냅샷에는 태그를 지정할 수 없습니다.

Lightsail 스냅샷에 태그를 지정하려면 어떻게 해야 하나요?

Lightsail 콘솔에서는 수동 스냅샷에 원본 리소스와 동일한 태그가 자동으로 지정됩니다. Lightsail API 또는를 사용하여 스냅샷을 AWS CLI 생성하는 경우 스냅샷에 대한 태그를 직접 선택할 수 있습니다.

> **Important**
>
> 데이터베이스의 수동 스냅샷 태그는 현재 결제 보고서(비용 할당 태그)에 포함되지 않습니다.

키-값 태그와 키 전용 태그의 차이점은 무엇입니까?

Lightsail 태그는 키-값 페어이며, 다양한 범주의 인스턴스와 같이 리소스를 구성할 수 있습니다(예: 프로젝트:블로그, 프로젝트:게임, 프로젝트:테스트). 이렇게 하면 리소스 구성, 결제 보고, 액세스 관리 등 모든 사용 사례를 완벽하게 제어할 수 있습니다. 또한 Lightsail 콘솔에서도 빠른 필터링을 위해 리소스에 키 전용 태그를 지정할 수 있습니다.

Amazon Lightsail에 대한 문서 기록

다음 표는 Lightsail에 대한 설명서 변경 사항 및 업데이트를 설명합니다. 이 표에 새 항목을 추가될 때 알림을 받으려면 RSS 리더를 사용하여 다음 URL을 구독하세요.

```
https://docs.aws.amazon.com/lightsail/latest/userguide/document_history.rss
```

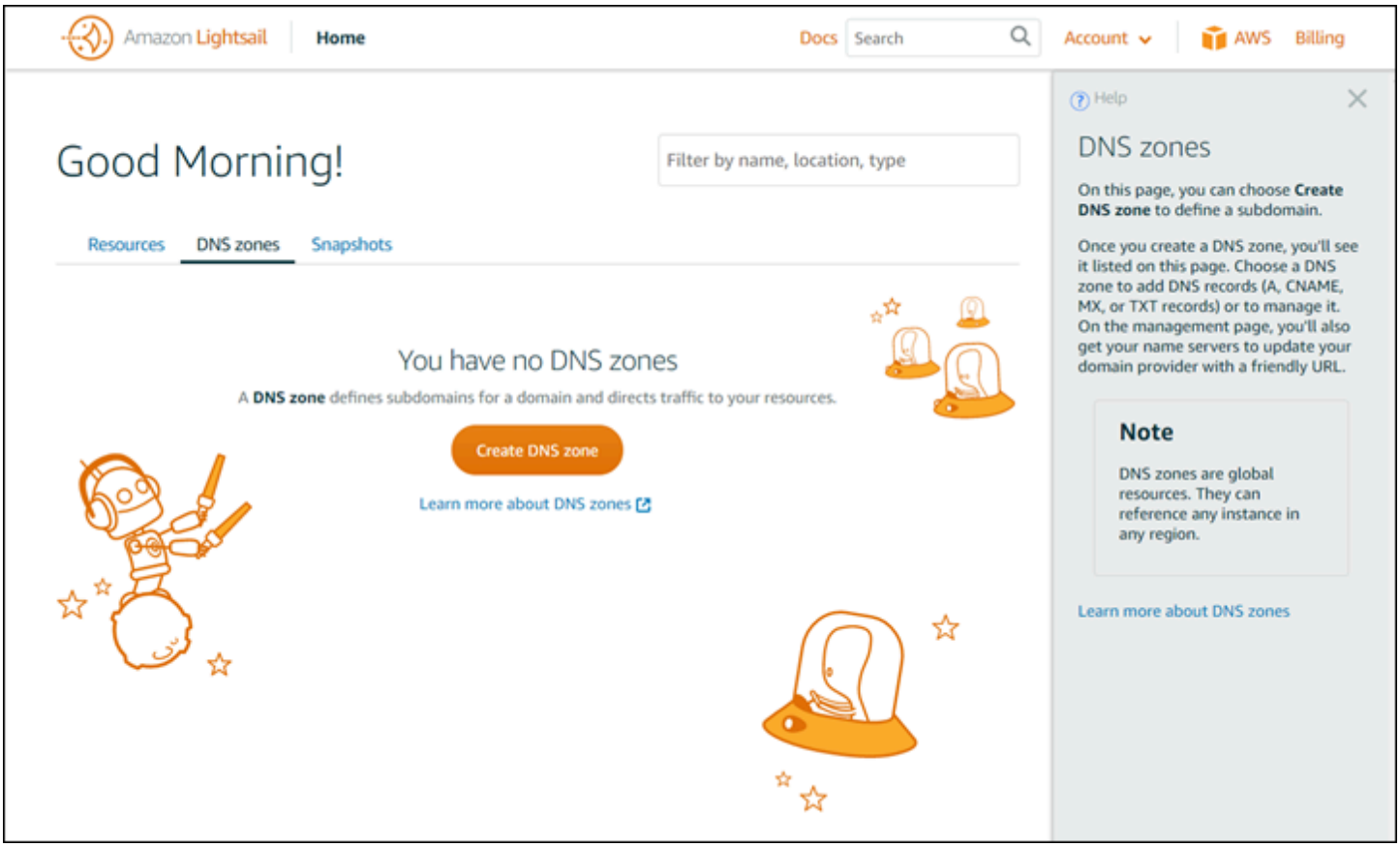
변경 사항	설명	날짜
Plesk 빠른 시작 안내서 업데이트	Lightsail에서 Plesk 인스턴스를 생성하는 데 대한 빠른 시작 안내서를 업데이트했습니다.	2025년 10월 31일
새 인스턴스 번들	Lightsail에 Linux/Unix 및 Windows 인스턴스 모두에 대한 더 큰 인스턴스 번들이 추가되었습니다.	2025년 10월 31일
인스턴스 번들 페이지	이 설명서에 인스턴스 번들 정보를 자세히 설명하는 새 페이지가 추가되었습니다.	2025년 10월 31일
Lightsail에 CORS 지원 추가	Lightsail은 이제 Lightsail 버킷에 대한 COR 지원을 제공합니다.	2025년 10월 15일
블루프린트 빠른 시작 안내서 업데이트	개정된 스냅샷 및 정적 IP 관련 콘텐츠로 여러 빠른 시작 안내서를 업데이트했습니다.	2025년 10월 2일
Lightsail AWS 리전확장	Lightsail(은)는 이제 아시아 태평양(자카르타) 리전에서 사용할 수 있습니다.	2025년 7월 31일
Lightsail에서 설명서 업데이트 추적 시작	이 문서 기록 페이지의 최초 게시.	2025년 7월 30일

Lightsail에 유용한 리소스 찾기

Amazon Lightsail에서는 여러 가지 방법으로 도움말을 찾을 수 있습니다.

상황에 맞는 도움말 패널

Lightsail에서는 콘솔의 각 페이지에 상황에 맞는 도움말 패널이 있으며, 이 패널에는 현재 페이지에 관련된 특정 정보와 추가 팁이 포함되어 있습니다. 페이지에 있는 내용에 대한 의문점이 있을 때 언제든 도움말 패널을 열어서 확인하고 계속 진행하려면 패널을 닫습니다. 어떤 페이지에서든 도움말을 선택하거나 사용자 인터페이스의 곳곳에 있는 작은 물음표를 선택하여 도움말 패널을 열 수 있습니다.



사용 설명서 정보

Amazon Lightsail 사용 설명서에는 Lightsail에서의 작업에 도움이 되는 방법 관련 항목과 개념적 개요가 수록되어 있습니다. 예를 들어, 인스턴스 생성, 인스턴스에 연결 또는 도메인 관리를 수행할 수 있습니다.

검색 사용

Lightsail의 어느 페이지에서든 각 페이지 상단에 있는 검색 상자를 사용하여 문서를 검색할 수 있습니다. 문서 검색 페이지에서 다시 검색을 수행하여 검색을 구체화할 수 있습니다.

찾으려는 문서를 찾지 못하셨습니까? 피드백을 보내주시면 저희가 해결해 드리겠습니다. Lightsail의 모든 페이지에서 피드백 제공을 선택하고 피드백을 제출하여 제안을 할 수 있습니다.

Lightsail CLI 및 API 사용

AWS Command Line Interface (AWS CLI) 또는 Lightsail REST API를 사용하여 Lightsail 리소스를 생성, 읽기, 업데이트 및 삭제할 수 있습니다. REST API 외에도 Java, Ruby, JavaScript(Node.js), Go, PHP, Python, .NET(C#) 및 C++를 포함한 여러 언어로 된 SDK가 있습니다. Lightsail API에 대한 자세한 내용은 Lightsail API 레퍼런스를 참조하세요.

> **Note**
>
> Lightsail API를 사용하려면 액세스 키를 생성해야 합니다. Lightsail API를 사용하기 위한 액세스 키 설정에 대해 자세히 알아보기

AWS CLI 는 Lightsail 리소스를 사용할 때 유용합니다. AWS에서 AWS CLI`aws lightsail help`를 입력하여 사용 가능한 명령에 대해 알아봅니다. 특정 CLI 명령에 대한 도움말을 보려면 명령 이름 다음에 `help`를 입력하면 명령의 파라미터와 예외에 대한 자세한 내용을 확인할 수 있습니다. 자세한 내용은 Lightsail CLI 참조를 참조하세요.

AWS 포럼 및 기타 커뮤니티 리소스

AWS 토론 포럼인 AWS 포럼에 질문을 게시할 수도 있습니다.

기계 번역으로 제공되는 번역입니다. 제공된 번역과 원본 영어의 내용이 상충하는 경우에는 영어 버전이 우선합니다.