

개발자 가이드

Amazon GameLift Streams



Amazon GameLift Streams: 개발자 가이드

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

Amazon GameLift Streams란 무엇입니까?	1
특성	1
Amazon GameLift Streams를 시작하는 방법	1
Amazon GameLift 스트림 액세스	2
이용 약관	3
설정	4
에 가입 AWS 계정	4
관리자 액세스 권한이 있는 사용자 생성	5
프로그래밍 방식 액세스 가져오기	6
웹 SDK 다운로드	6
다운로드 AWS CLI	7
결제 알림 설정	7
시작하기	8
구성 선택	8
시작점	9
비용 최적화	10
구성 결정	11
구성 선택이 다음 단계에 미치는 영향	12
다음 단계	12
구성 옵션	13
런타임 환경	13
스트림 클래스	14
첫 번째 스트림	19
사전 조건	20
1단계: Amazon S3 버킷에 애플리케이션 업로드	20
2단계: Amazon GameLift Streams용 애플리케이션 구성	22
3단계: Amazon GameLift Streams가 애플리케이션을 스트리밍하는 방법 관리	26
4단계: Amazon GameLift Streams에서 스트림 테스트	32
5단계: 정리(건너뛰지 않음)	33
스트림 관리	35
주요 개념	35
Applications	36
업로드하기 전에	37
Amazon S3 버킷에 애플리케이션 업로드	38

애플리케이션 만들기	39
애플리케이션 편집	45
애플리케이션 삭제	48
애플리케이션 로그 버킷 권한 정책	50
연결된 스트림 그룹	51
스트림 그룹	51
스트림 용량 정보	52
용량 및 서비스 할당량	54
위치 정보	55
스트림 그룹 생성	56
일반 설정 편집	67
용량 편집	68
용량 축소 동작	70
스트림 그룹에 위치 추가	70
스트림 그룹의 위치 제거	71
스트림 그룹 삭제	73
연결된 애플리케이션	74
스트림 그룹 수명 주기	74
스트림 그룹 유지 관리	75
다중 애플리케이션 스트림 그룹	75
제한 사항 및 요구 사항	76
기본 애플리케이션 정보	76
기본 애플리케이션 변경	77
애플리케이션 연결	78
애플리케이션 연결 해제	80
스트림 세션	82
스트림 세션 정보	82
콘솔에서 스트림 테스트	82
스트림 세션 수명 주기	83
스트림 세션에 영향을 미치는 제한 시간 값	86
스트림 세션 종료	87
스트림 세션에 다시 연결	87
스트림 세션 파일 내보내기	87
작동 방식	88
비용 영향	89
파일 내보내기(콘솔)	89

파일 내보내기(CLI)	89
VPC 연결	92
VPC 연결 작동 방식	92
요구 사항 및 고려 사항	93
요구 사항	93
추가 고려 사항	93
VPC 연결 구성	94
1단계: VPC 구성을 사용하여 스트림 그룹 생성	94
2단계: RAM 리소스 공유 수락	95
3단계: VPC 연결 생성	96
4단계: 라우팅 구성	96
(선택 사항) 5단계: 보안 그룹 업데이트	97
(선택 사항) 6단계: CIDR 블록 업데이트	98
연결 확인	98
Amazon GameLift Streams 백엔드 서비스 및 웹 클라이언트	100
지원되는 브라우저 및 입력	100
알려진 문제	101
제한 사항	102
IPv6 지원	102
필수 포트	103
웹 서버 및 클라이언트 설정	103
사전 조건	104
웹 SDK 다운로드	104
스트리밍 리소스 설정	104
백엔드 서버 설정	105
웹 클라이언트 시작	105
스트리밍 리소스 정리	106
스트림 모양 사용자 지정	108
화면 로드	108
로컬 기본 설정	108
마우스 이동 처리	109
마우스 입력 모드	109
포인터 잠금	110
모범 사례	111
데이터 채널 통신	111
Features	112

데이터 채널 사용	112
클라이언트 측에서	112
애플리케이션 측에서	113
시작 체크리스트	116
Amazon GameLift Streams 팀에 알림	116
호환성 및 성능 테스트	116
용량 예약	116
대규모 성능 테스트	117
사전 시작 설정	117
추가 팁	117
추가 지원이 필요하십니까?	117
보안	118
데이터 보호	119
저장 시 암호화	120
전송 중 암호화	121
최종 사용자 스트림 보호	121
Linux 스트림 클래스의 세션 격리	121
Windows 스트림 클래스의 세션 격리	121
암호화 키 관리	122
인터넷워크 트래픽 개인 정보 보호	122
자격 증명 및 액세스 관리	123
대상	123
ID를 통한 인증	123
정책을 사용하여 액세스 관리	125
Amazon GameLift Streams가 IAM과 작동하는 방식	126
ID 기반 정책 예시	131
문제 해결	134
규정 준수 확인	135
복원력	135
인프라 보안	135
재사용 및 다중 테넌시	136
인터페이스 VPC 엔드포인트	137
구성 및 취약성 분석	139
보안 모범 사례	139
Amazon GameLift 스트림 모니터링	140
CloudWatch를 사용하여 모니터링	140

스트림 그룹 용량 및 사용량	141
스트림 그룹 성능 및 리소스 사용률	142
스트림 상태	143
고객 참여	143
데이터 채널	144
CloudTrail을 사용하여 API 직접 호출 로깅	145
CloudTrail에서 Amazon GameLift Streams 데이터 이벤트	147
CloudTrail의 Amazon GameLift Streams 관리 이벤트	148
Amazon GameLift Streams 이벤트 예제	149
실시간 성능 통계	153
성능 통계 수신	153
성능 통계 참조	155
문제 해결	157
액세스 거부됨	157
애플리케이션 문제	158
예비 검사	158
양성자 문제	158
화면 해상도로 인한 애플리케이션 문제	159
스트림 세션 시작 시 애플리케이션이 종료됩니다.	159
Unreal Engine 애플리케이션이 충돌하거나 추가 종속성이 필요함	159
성능 문제	160
Amazon GameLift Streams에서 스트리밍할 때 게임 성능이 저하됩니다.	160
Windows 애플리케이션에서 로드 시간이 느리거나 문제가 발생함	161
스트림 연결 문제	166
스트림 입력 문제	167
일반 입력 문제 해결	167
게임패드 및 마이크 입력은 네이티브 Linux 애플리케이션에서 작동하지 않습니다.	167
MacOS 클라이언트에서 키 입력이 멈춘 것처럼 보임	167
OS UI 요소를 열 때 입력 고정	168
마우스 움직임은 Amazon GameLift Streams에서 다르게 동작합니다.	168
스트림 세션 문제	169
스트림 세션이 시작되지 않음	169
스트림 세션 종료됨	169
Proton과의 호환성	170
테스트 및 문제 해결을 위한 상위 수준 단계	170
Proton과 관련된 알려진 문제	170

로컬 시스템 설정	171
원격 시스템 설정	172
Proton 문제 해결	177
Unreal Engine 성능 프로파일링	181
리전, 할당량 및 제한 사항	184
AWS 리전 및 스트리밍 위치	184
서비스 엔드포인트	184
스트리밍 위치	185
스트림 클래스별로 지원되는 위치	186
Service Quotas	187
Service Quotas	187
API 속도 제한	200
기타 제한 사항	201
사용량 및 청구서	203
Amazon GameLift Streams 청구서 및 사용량 검토	203
Amazon GameLift Streams 비용 관리 모범 사례	204
결제 알림을 생성하여 사용량 모니터링	204
스트림 그룹을 0 용량으로 확장	204
원본 애플리케이션 파일 삭제	205
.....	ccvi

Amazon GameLift Streams란 무엇입니까?

Amazon GameLift Streams를 사용하면 게임 게시자 등이 전 세계 플레이어와 시청자에게 지연 시간이 짧은 온디맨드 스트리밍 경험을 제공할 수 있습니다. Amazon GameLift Streams는 AWS 글로벌 인프라와 결합된 자체 스트리밍 기술을 사용하여 대규모로 애플리케이션 스트리밍을 운영하고 유지합니다. 게시자는 온디맨드 스트리밍 리소스와 예약된 스트리밍 리소스를 모두 유연하게 프로비저닝하여 용량과 비용을 효과적으로 관리할 수 있습니다.

주제

- [특성](#)
- [Amazon GameLift Streams를 시작하는 방법](#)
- [Amazon GameLift 스트림 액세스](#)

특성

Amazon GameLift Streams는 다음과 같은 주요 기능을 제공합니다.

- AWS 글로벌 공간을 사용하는 브라우저가 있는 모든 디바이스에 player-to-cloud 지연 시간을 최소화 하면서 실시간 게임 플레이 경험을 제공하는 스트리밍 기술입니다.
- 다운로드 없이 고화질(1080p) 해상도와 60fps로 게임을 원활하게 플레이할 수 있으므로 브라우저 기반 디바이스를 강력한 게임 머신으로 바꿀 수 있습니다.
- 고객 수요에 맞게 스트리밍 용량을 조정하는 조정 도구입니다. 예를 들어 이러한 도구를 사용하면 새 플레이어를 스트림 세션에 빠르게 수용할 수 있는 충분한 용량을 유지하면서 게임 스트리밍 비용을 유지할 수 있습니다.
- Amazon GameLift Streams 콘솔을 사용하여 성능 분석을 스트리밍하여 지표를 추적하고, 스트림 로그를 보고, 스트림 리소스 사용에 대한 데이터를 검토합니다.
- Windows 및 Linux 기반 게임을 거의 또는 전혀 수정하지 않고 직접 스트리밍합니다.
- Amazon GameLift Streams SDK는 기존 ID 서비스, 스토어프론트 및 클라이언트 애플리케이션을 통합하는 데 도움이 됩니다.

Amazon GameLift Streams를 시작하는 방법

Amazon GameLift Streams를 처음 사용하는 경우 다음 주제로 시작하는 것이 좋습니다.

- [Amazon GameLift Streams를 개발자로 설정](#)에서는 사용자 액세스 권한이 AWS 계정 있는를 가져 오고 Amazon GameLift Streams를 사용하여 개발하는 데 필요한 소프트웨어를 설정하는 등 일회성 설정 작업을 다룹니다.
- [Amazon GameLift Streams에서 첫 번째 스트림 시작](#)는 콘텐츠 스트리밍 워크플로의 중요한 단계를 안내합니다. 게임 빌드와 같은 콘텐츠부터 시작하여 Amazon GameLift Streams 스트리밍 클라우드 리소스를 프로비저닝하고 스트리밍 세션을 시작합니다.

Amazon GameLift 스트림 액세스

다음 도구를 사용하여 애플리케이션 콘텐츠 및 스트리밍 리소스를 생성, 액세스 및 관리할 수 있습니다.

- AWS Management Console - Amazon GameLift Streams 애플리케이션 및 스트림 그룹을 생성하고 관리하는 데 사용할 수 있는 웹 인터페이스를 제공합니다.
- AWS Command Line Interface (AWS CLI) - 광범위한 AWS 서비스에 대한 명령을 제공하며 Windows, Mac 및 Linux에서 지원됩니다. 이 도구에 대한 자세한 내용은 [AWS Command Line Interface 페이지](#)를 참조하세요.
- AWS SDK - 언어별 APIs 제공하고 서명 계산, 요청 재시도 처리, 오류 처리와 같은 연결 세부 정보를 관리합니다. Amazon GameLift Streams 서비스 API에 대한 설명서는 [Amazon GameLift Streams API 참조](#)를 참조하세요. AWS SDK에 대한 자세한 내용은 [빌드 기반 도구를 참조하세요 AWS](#).

지원되는에 대한 자세한 내용은 단원을 AWS 리전참조하십시오 [리전, 할당량 및 제한 사항](#).

Amazon GameLift Streams 사용 약관

Amazon GameLift Streams를 사용하기 전에 스트리밍하려는 애플리케이션과 스트리밍하려는 위치에 적용되는 라이선스 조건을 포함하여 적용 가능한 모든 법적 요구 사항을 준수할 수 있는지 확인합니다.

- AWS 요구 사항에 대한 자세한 내용은 [AWS 서비스 약관](#)의 섹션 43을 참조하세요.
- 서비스 수준 계약에 대한 자세한 내용은 [Amazon GameLift Streams 서비스 수준 계약을 참조하세요](#).

Amazon GameLift Streams를 개발자로 설정

프로젝트에서 Amazon GameLift Streams 서비스 사용을 시작하려면 다음 기본 설정 작업을 완료하세요. Amazon GameLift Streams에서 사용하려는 계정의 AWS 계정 및 사용자가 이미 있는 경우 로 건너뛸 수 있습니다 [웹 SDK 다운로드](#).

로 수행할 수 있는 작업에 대한 자세한 내용은 [AWS 시작하기](#)를 AWS 계정참조하세요.

이러한 설정 작업을 완료한 후에는 웹 클라이언트에서 콘텐츠 스트리밍을 가져오기 위한 전체 워크플로를 다루는 자습서로 이동하여 [Amazon GameLift Streams에서 첫 번째 스트림 시작](#) 단계별로 진행하는 것이 좋습니다.

주제

- [에 가입 AWS 계정](#)
- [관리자 액세스 권한이 있는 사용자 생성](#)
- [프로그래밍 방식 액세스 가져오기](#)
- [Amazon GameLift Streams 웹 SDK 다운로드](#)
- [다운로드 AWS CLI](#)
- [결제 알림 설정](#)

에 가입 AWS 계정

이 없는 경우 다음 단계를 AWS 계정완료하여 생성합니다.

에 가입하려면 AWS 계정

1. <https://portal.aws.amazon.com/billing/signup>을 엽니다.
2. 온라인 지시 사항을 따릅니다.

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

에 가입하면 AWS 계정AWS 계정 루트 사용자인 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 작업](#)을 수행하는 것입니다.

AWS 는 가입 프로세스가 완료된 후 확인 이메일을 보냅니다. 언제든지 <https://aws.amazon.com/>으로 이동하고 내 계정을 선택하여 현재 계정 활동을 보고 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 소유자 [AWS Management Console](#)로 로그인합니다. 다음 페이지에서 비밀번호를 입력합니다.

루트 사용자를 사용하여 로그인하는 데 도움이 필요하다면 AWS 로그인 User Guide의 [루트 사용자 로 로그인](#)을 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 Center 설정](#)을 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 사용자 로 로그인

- IAM IDentity Center 사용자 로 로그인하려면 IAM Identity Center 사용자를 생성할 때 이메일 주소 로 전송된 로그인 URL을 사용합니다.

IAM Identity Center 사용자를 사용하여 로그인하는 데 도움이 필요하다면 AWS 로그인 사용 설명서의 [AWS 액세스 포털에 로그인](#)을 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 set](#)를 참조하세요.
2. 사용자를 그룹에 할당하고, 그룹에 Single Sign-On 액세스 권한을 할당합니다.
지침은 AWS IAM Identity Center 사용 설명서의 [Add groups](#)를 참조하세요.

프로그래밍 방식 액세스 가져오기

에 대한 사용자 로그인 자격 증명 외에도 AWS Command Line Interface ()로 작업 AWS Management Console할 때와 같이 프로그래밍 방식 액세스를 위한 자격 증명이 필요합니다AWS CLI. 프로그래밍 방식 자격 증명은 두 부분으로 구성된 액세스 키 세트로 구성됩니다. 다음 방법 중 하나를 사용하여 액세스 키를 생성합니다.

- 방법 1 - IAM Identity Center로 생성한 관리 사용자를 사용하는 경우 [AWS 리소스에 대한 단기 AWS CLI 액세스를 위한 임시 보안 자격 증명을 생성하기 위해 액세스를 위한 IAM 역할](#) 자격 증명 가져오기를 참조하세요. 이 지침을 따를 때는 관리 사용자 이름 및 암호(루트 사용자가 아님)를 사용하여 계정의 AWS 액세스 포털 URL을 통해 로그인해야 합니다.
- 방법 2 - 기존 IAM 사용자를 사용 중이고 아직 IAM Identity Center 사용으로 전환하지 않은 경우 [IAM 사용자의 액세스 키 관리\(콘솔\)](#)를 참조하여 사용자의 장기 자격 증명을 생성합니다.

Note

장기 액세스 키 대신 임시 자격 증명을 사용하는 것이 가장 좋습니다. 임시 보안 인증도 액세스 키 ID와 비밀 액세스 키로 구성되지만 보안 인증이 만료되는 시간을 나타내는 보안 토큰이 포함되어 있습니다. 자세한 내용은 [AWS 액세스 키 관리 모범 사례](#)를 참조하세요AWS 일반 참조.

Amazon GameLift Streams 웹 SDK 다운로드

콘솔 내 스트리밍 환경을 사용하여 추가 자료 없이 시작할 수 있습니다. 이를 시작점으로 사용하는 것이 좋습니다. 추가 인프라를 설정하지 않고도 Amazon GameLift Streams에서 애플리케이션이 어떻게 작동하는지 평가할 수 있기 때문입니다. 자세한 정보는 [Amazon GameLift Streams 시작하기](#) 섹션으로 이동하세요.

자체 Amazon GameLift Streams 통합을 구축할 준비가 되면 [시작하기 제품 페이지](#)의 리소스 섹션에 있는 Amazon GameLift Streams 웹 SDK를 다운로드합니다. Amazon GameLift Streams는 웹 애플리케이션에 통합되도록 구축되었습니다. JavaScript 기반 웹 SDK를 통합하여 웹 사이트 또는 브라우저 기반 애플리케이션에서 스트리밍을 설정해야 합니다. 다운로드에는 Amazon GameLift Streams 서비스를 사용하는 샘플 웹 서버와 스트림에 연결하기 위한 샘플 웹 클라이언트도 포함되어 있습니다.

자체 Amazon GameLift Streams 솔루션 설정에 대한 자세한 내용은 섹션을 참조하세요 [Amazon GameLift Streams 백엔드 서비스 및 웹 클라이언트](#).

다운로드 AWS CLI

Amazon GameLift Streams를 콘텐츠와 함께 사용하려면 AWS Command Line Interface ()를 가져오는 것이 좋습니다. AWS CLI는 터미널 프로그램에서 명령을 실행하여 동등한 AWS SDK 기능을 제공하는 오픈 소스 도구입니다.

1. AWS CLI 운영 체제용의 최신 버전을 다운로드하여 설치합니다. AWS Command Line Interface 사용 설명서의 다음 [설치 지침](#)을 참조하세요.
2. [설정에 AWS CLI](#) 설명된 대로 사용자 액세스 자격 증명 및 기타 기본 설정으로 도구를 구성합니다. 이 구성을 사용하면 모든 명령으로 자격 증명 및 기타 설정을 명시적으로 지정할 필요가 없습니다.
3. 다음 명령을 사용하여 설치를 확인하고 사용 가능한 Amazon GameLift Streams 명령 목록을 가져옵니다.

```
aws gameliftstreams help
```

결제 알림 설정

스트림 그룹에는 활성 스트림 용량당 초당 비용이 발생합니다. 비용 및 사용량이 예산 내에서 유지되도록 하려면 섹션을 참조하세요 [결제 알림을 생성하여 사용량 모니터링](#).

Amazon GameLift Streams 시작하기

이 섹션에서는 Amazon GameLift Streams를 통해 애플리케이션 및 게임 스트리밍을 성공적으로 시작하는 데 도움이 될 수 있습니다. 이 섹션의 주제에서는 애플리케이션을 Amazon GameLift Streams에 업로드하는 것부터 스트림에서 콘텐츠가 어떻게 작동하는지 테스트하는 것까지 end-to-end 프로세스를 다룹니다. 또한 적절한 런타임 및 스트림 클래스 구성을 선택하여 성능과 비용을 최적화하는 등 스트리밍을 준비하는 데 도움이 되는 중요한 단계도 다룹니다.

주제

- [Amazon GameLift Streams에서 구성 선택](#)
- [구성 옵션](#)
- [Amazon GameLift Streams에서 첫 번째 스트림 시작](#)

Amazon GameLift Streams에서 구성 선택

이 가이드는 Amazon GameLift Streams를 통해 애플리케이션 및 게임을 스트리밍하기 위한 최적의 런타임 환경 및 구성 설정을 선택하는 데 도움이 될 수 있습니다. 구성 설정은 콘텐츠의 성능과 Amazon GameLift Streams에서 콘텐츠 실행과 관련된 비용에 직접적인 영향을 미칩니다. 다양한 애플리케이션과 그래픽 충실도를 지원하는 몇 가지 옵션이 있습니다.

구성 옵션의 전체 목록은 [여기](#)에서 확인할 수 있습니다.

다음 주요 용어는 이러한 구성 옵션이 함께 작동하는 방식을 이해하는 데 도움이 될 수 있습니다.

- 런타임은 Amazon GameLift Streams에서 애플리케이션을 실행할 기본 운영 체제 및 소프트웨어 환경을 나타냅니다. 기본 런타임 환경 옵션은 Windows, Linux 및 Proton입니다.
- 스트림 클래스는 운영 체제, CPU, GPU, RAM 및 기타 사양에 따라 Amazon GameLift Streams 내에서 사용할 수 있는 다양한 리소스 구성을 나타냅니다. 스트림 클래스는 스트림 세션에 할당된 하드웨어 리소스와 테넌시 모델(단일 가상 머신에서 실행할 수 있는 동시 스트림 수)을 모두 정의하는 스트림 그룹의 구성 옵션입니다.
- 다중 테넌시를 사용하면 여러 사용자가 동일한 기본 하드웨어 리소스를 공유할 수 있으며, 이는 최대 하드웨어 기능이 필요하지 않은 애플리케이션에 비용 효율적인 옵션일 수 있습니다. 다중 테넌시가 있는 스트림 클래스는 하나의 리소스 비용으로 여러 스트림을 호스팅할 수 있습니다. "High" 스트림 클래스에는 1:2 테넌시가 있는 반면, "Ultra" 스트림 클래스에는 1 테넌시가 있습니다.

Amazon GameLift Streams 구성을 설정할 때 선택한 런타임 환경에 따라 호환되고 사용할 수 있는 특정 스트림 클래스 옵션이 결정됩니다. 애플리케이션의 요구 사항을 올바른 런타임 환경 및 스트림 클래스와 일치시키는 것은 Amazon GameLift Streams의 성능과 비용 효율성을 최적화하는 데 중요합니다.

스트리밍 비용은 스트림 클래스에 따라 다릅니다. 자세한 비용 목록은 Amazon GameLift 스트림 [요금 페이지](#)를 참조하세요.

시작점

애플리케이션에 따라 스트리밍을 시작하기에 좋은 시작점입니다. 나중에 다른 구성 옵션을 탐색하여 비용을 최적화할 수 있습니다.

Windows 애플리케이션의 경우

Windows 애플리케이션의 경우 Microsoft Windows Server 2022 Base 런타임 환경 및 gen6n_ultra_win2022 스트림 클래스로 시작하는 것이 좋습니다. 이러한 런타임 환경과 스트림 클래스의 조합은 Windows 기반 콘텐츠에 대한 다양한 그래픽 집약적 사용 사례에 가장 높은 호환성과 고성능을 갖춘 예측 가능하고 잘 지원되는 구성을 제공합니다.

다양한 가격 및 성능 옵션을 제공하는 다른 Windows 스트림 클래스 구성이 있습니다(참조 [Windows 스트림 클래스](#)). 애플리케이션에 가장 적합한 옵션을 찾기 위해 이러한 구성을 시도해 볼 수 있습니다.

Windows 런타임은 DirectX 11 또는 DirectX 12를 사용하는 게임 및 기타 3D 애플리케이션과 Unity 2022.3, Unreal Engine 4.27, Unreal Engine 5~5.6을 포함한 게임 엔진을 지원합니다. 스트리밍은 IPv4와 IPv6 모두에서 지원됩니다.

Linux 애플리케이션의 경우

Linux에서 기본적으로 실행되도록 구축된 애플리케이션에 Ubuntu 22.04 LTS 런타임 환경을 사용합니다. 성능을 최적화하려면 Pro 또는 Ultra 스트림 클래스 중 하나를 선택합니다(참조 [Linux 및 Proton 스트림 클래스](#)). 비용을 최적화하려면 소규모, 중간 또는 높은 스트림 클래스 중 하나를 선택합니다. 이는 여러 동시 스트림 세션이 동일한 컴퓨팅 리소스를 공유하는 비용 효율적인 옵션입니다.

Important

Amazon GameLift Streams의 Linux 런타임은 IPv6를 통한 스트리밍을 지원하지 않습니다. 클라이언트는 IPv4를 통해 애플리케이션을 스트리밍해야 합니다.

비용 최적화

시작점 권장 사항은 좋은 시작점이지만 우수한 성능을 유지하면서 비용을 최적화하기 위해 다른 구성 옵션을 고려하는 것이 좋습니다.

Proton 런타임 환경 사용

많은 Windows 애플리케이션이 Proton 런타임 환경에서 실행될 수 있습니다. Proton은 Linux에서 실행되는 게임 최적화 호환성 계층입니다. 이 런타임의 스트림 클래스 옵션에는 NVIDIA 하드웨어에서 실행되는 강력한 GPU 리소스가 포함되며, DirectX 11과 Proton 8.0-5, DirectX 12부터 지원됩니다. 이 옵션에 대한 자세한 내용은 [Proton wiki](#)를 참조하십시오. Proton에서 애플리케이션 실행을 탐색하려면 Proton 9.0-2를 사용하여 테스트를 시작하는 것이 좋습니다.

Important

Amazon GameLift Streams의 Proton 런타임은 IPv6를 통한 스트리밍을 지원하지 않습니다. 클라이언트는 IPv4를 통해 애플리케이션을 스트리밍해야 합니다.

Important

Proton 런타임 환경에서 Windows 애플리케이션의 호환성은 특정 애플리케이션 요구 사항에 따라 달라집니다. 예를 들어 Proton 9.0-2는 Unreal Engine 5에 대해 Proton 8.0-2c보다 더 나은 지원을 제공합니다. 일반적으로 최신 게임일수록 최신 버전의 Proton이 필요합니다. 최적의 성능을 보장하려면 로컬 환경에서 이 런타임을 철저히 테스트하는 것이 좋습니다. [Proton 문제 해결 가이드](#)를 사용하면 이러한 작업에 도움이 됩니다.

애플리케이션을 Linux에 컴파일

또 다른 비용 절감 옵션은 Linux에서 기본적으로 실행되도록 애플리케이션을 대상으로 지정하는 것입니다. 먼저 애플리케이션을 테스트하여 애플리케이션의 Linux 버전이 필요에 따라 작동하는지 확인합니다. 애플리케이션이 Linux에서 성공적으로 실행되면 Linux 애플리케이션에 대한 Amazon GameLift Streams 구성 옵션을 따를 수 있습니다.

Unreal Engine 애플리케이션을 Linux로 교차 컴파일하는 방법에 대한 자세한 내용은 Unreal Engine 개발자 안내서의 [교차 컴파일 도구 체인](#) 섹션을 참조하세요.

구성 결정

최상의 런타임 및 스트림 클래스 구성을 결정하려면 다음 주요 질문을 고려하세요.

1. 애플리케이션 또는 게임은 어떤 플랫폼을 위해 구축되나요? Windows 애플리케이션이 있는 경우 Windows 런타임 환경을 가장 간단하게 설정할 수 있습니다. 애플리케이션이 Linux용으로 빌드된 경우 Linux 런타임 환경이 가장 간단합니다. Windows 애플리케이션 스트리밍 비용을 절감하기 위해 Proton 런타임 환경을 탐색하거나 애플리케이션을 Linux로 컴파일할 수 있습니다.
2. 사용 사례에서 성능과 비용은 얼마나 중요합니까? Windows 런타임 환경은 최상의 성능을 제공할 수 있지만 실행 비용이 더 많이 들 수 있습니다. 반대로 Proton 런타임 환경은 약간 낮은 성능 또는 잠재적 호환성 문제가 발생할 수 있지만 더 비용 효율적입니다. 이는 Windows 기반 애플리케이션에 사용 가능한 Proton 런타임에서 아직 완전히 지원되지 않는 특정 기능이 필요할 수 있기 때문입니다. 따라서 Proton 환경에서 애플리케이션을 실행할 때 기능 또는 그래픽 차이가 발생할 수 있습니다. 다양한 런타임 환경 및 스트림 클래스에서 애플리케이션을 테스트하여 성능 및 비용 장단점을 평가하는 것이 좋습니다. 런타임 환경 옵션의 전체 목록은 [섹션을 참조하세요](#) [런타임 환경](#).
3. 애플리케이션의 그래픽 요구 사항은 무엇입니까? 애플리케이션의 그래픽 요구 사항은 가장 적합한 스트림 클래스 구성을 결정하는 데 도움이 될 수 있습니다. 애플리케이션에 고성능 GPUs 필요한 경우 비디오 메모리(VRAM) 및 시스템 메모리(RAM)가 더 많은 스트림 클래스를 사용하는 것이 좋습니다. 예를 들어 gen5n 및 gen6n 스트림 클래스는 gen4n 스트림 클래스에 비해 그래픽 집약적인 애플리케이션에서 최대 3배 더 나은 성능을 제공합니다. 애플리케이션에 최대 GPU 및 CPU 리소스가 필요한 경우 "pro" 스트림 클래스를 고려해야 합니다. 반대로 애플리케이션이 더 낮은 그래픽 충실도로 효과적으로 작동할 수 있는 경우 GPU를 공유하는 소형, 중형 또는 하이 스트림 클래스를 사용하여 비용을 절감할 수 있습니다. [스트림 클래스](#)을(를) 참조하세요.
4. 설정에 얼마나 투자할 의향이 있습니까? 애플리케이션을 설정하는 가장 간단한 방법은 기본적으로 Windows 또는 Linux 런타임을 사용하여 실행하는 것입니다. 애플리케이션을 out-of-the-box 사용할 수 있기 때문입니다. 반대로 Proton 런타임 환경에서는 필요에 맞는 최적의 Proton 구성을 식별하기 위해 더 많은 실습 테스트가 필요합니다. 런타임 환경을 결정할 때 설정 및 테스트 프로세스에 할당할 수 있는 시간과 리소스를 고려합니다.
5. 다양한 런타임 환경 및 스트림 클래스에서 애플리케이션을 테스트했나요? 다양한 런타임 환경 및 스트림 클래스에서 콘텐츠를 테스트하여 콘텐츠의 성능을 확인하는 것이 좋습니다. 이를 통해 안정성, 그래픽 품질, 기능 기능 및 입력 응답성과 같은 요인을 기반으로 최적의 적합성을 결정할 수 있습니다.

구성 선택이 다음 단계에 미치는 영향

선택한 구성은 스트리밍 환경 설정의 다음 단계에 직접적인 영향을 미칩니다. 구체적으로 설명하면 다음과 같습니다.

- Amazon GameLift Streams 애플리케이션 생성: 게임 또는 애플리케이션을 Amazon GameLift Streams에 업로드할 때 사용할 런타임 환경을 지정해야 합니다. 이 선택에 따라 사용할 수 있는 스트림 그룹의 유형이 결정됩니다.
- 스트림 그룹에 연결: 기존 스트림 그룹이 이미 있는 경우 런타임 환경 선택이 해당 그룹의 구성과 일치해야 합니다. 예를 들어 Windows 런타임을 선택하는 경우 애플리케이션을 Windows 애플리케이션용으로 설정된 스트림 그룹에만 연결할 수 있습니다.
- 스트림 그룹 생성: 새 스트림 그룹을 생성할 때 선택한 런타임과 호환되는 스트림 클래스를 선택해야 합니다. 선택한 스트림 클래스는 애플리케이션에 필요한 그래픽 요구 사항 및 컴퓨팅 성능과 일치해야 합니다.

선택한 구성 설정이 이러한 후속 단계에 어떤 영향을 미치는지 이해하면 전체 스트리밍 구현을 더 잘 계획하고 원활한 통합 프로세스를 보장할 수 있습니다.

다음 단계

선택한 구성에 따라 스트리밍을 위해 애플리케이션을 설정하는 데 사용할 수 있는 몇 가지 방법이 있습니다.

Windows 또는 Linux 런타임을 선택한 경우

Windows 또는 Linux 런타임의 경우 다음 단계는 Amazon GameLift Streams에서 스트리밍을 설정한 다음 스트림을 테스트하는 것입니다. 자세한 정보는 [Amazon GameLift Streams에서 첫 번째 스트림 시작](#) 섹션으로 이동하세요.

Proton 사용을 고려 중인 경우

애플리케이션의 Proton과의 호환성은 애플리케이션의 특정 요구 사항에 따라 달라집니다. 따라서 Amazon GameLift Streams로 가져오기 전에 다양한 Proton 버전에서 애플리케이션을 테스트하는 것이 좋습니다. 이를 통해 요구 사항에 가장 적합한 성능과 호환성을 제공하는 Proton 설정을 식별할 수 있습니다. Amazon GameLift Streams 외부에서 테스트하면 애플리케이션의 성능과 기능을 검증하고 런타임과 관련된 문제를 디버깅할 수 있습니다. 자세한 내용은 [Proton for Amazon GameLift Streams와의 호환성 테스트 및 문제 해결](#) 단원을 참조하세요.

특정 Proton 구성을 선택하면 Amazon GameLift Streams에서 스트리밍을 설정할 준비가 된 것입니다. 자세한 내용은 단원을 참조하십시오 [Amazon GameLift Streams에서 첫 번째 스트림 시작](#).

구성 옵션

런타임 환경

런타임은 Amazon GameLift Streams에서 애플리케이션을 실행하는 기본 운영 체제 및 소프트웨어 환경을 나타냅니다. 기본 런타임 옵션은 Windows, Linux 및 Proton입니다. 시작하기 워크플로 [2단계: Amazon GameLift Streams용 애플리케이션 구성](#)의에서 런타임 환경을 지정합니다.

[Proton](#)은 Linux 기반 환경에서 많은 Windows 애플리케이션을 실행할 수 있는 호환성 계층입니다. Proton을 사용하려는 경우 로컬 시스템에서 애플리케이션이 실행되는 방식을 테스트하는 것이 좋습니다. 자세한 정보는 [Proton for Amazon GameLift Streams와의 호환성 테스트 및 문제 해결](#) 섹션을 참조하세요.

런타임	설명
Microsoft Windows Server 2022 Base	Windows 애플리케이션과 호환됩니다. 스트림 세션에서 IPv4 및 IPv6 사용을 지원합니다.
Ubuntu 22.04 LTS	Linux 애플리케이션과 호환됩니다. 스트림 세션에서 IPv6 사용을 지원하지 않습니다.
Proton 9.0-2	Windows 애플리케이션과 호환됩니다. Proton experimental_9.0 브랜치를 기반으로 합니다. Proton과의 호환성 테스트를 시작하기 위한 권장 버전입니다. 스트림 세션에서 IPv6 사용을 지원하지 않습니다.
Proton 8.0-5	Windows 애플리케이션과 호환됩니다. Proton experimental_8.0 브랜치를 기반으로 합니다. 스트림 세션에서 IPv6 사용을 지원하지 않습니다.
Proton 8.0-2c	Windows 애플리케이션과 호환됩니다. Proton experimental_8.0 브랜치를 기반으로 합니다. 스트림 세션에서 IPv6 사용을 지원하지 않습니다.

제한 사항

Ubuntu 22.04 LTS에서는 게임패드 지원을 사용할 수 없습니다. 다른 런타임 환경은 최종 사용자의 운영 체제 및 브라우저에 따라 게임패드를 지원합니다. 자세한 내용은 [지원되는 브라우저 및 입력 단원을 참조](#)하십시오.

스트림 클래스

스트림 클래스는 CPU, GPU, RAM 및 기타 사양에 따라 Amazon GameLift Streams 내에서 사용할 수 있는 다양한 리소스 구성을 나타냅니다. 스트림 클래스는 스트림 세션에 할당된 하드웨어 리소스와 테넌시 모델(단일 가상 머신에서 실행할 수 있는 동시 스트림 수)을 모두 정의하는 스트림 그룹의 구성 옵션입니다. 시작하기 워크플로 [3단계: Amazon GameLift Streams가 애플리케이션을 스트리밍하는 방법 관리](#)의에서 스트림 클래스를 지정합니다.

Windows 스트림 클래스

스트림 클래스	Amazon EC2 구성	설명
gen6n_pro_win2022	g6.4xlarge Amazon EC2 인스턴스의 Windows 런타임	(NVIDIA, pro) 최대 리소스가 필요하며 3D 장면 복잡성이 매우 높음 수준인 애플리케이션을 지원합니다. Microsoft Windows Server 2022 Base에서 애플리케이션을 실행하고 DirectX 12를 지원합니다. Unreal Engine 5.6 버전 이하, 32비트 및 64비트 애플리케이션, 치트 방지 기술과 호환됩니다. NVIDIA L4 Tensor Core GPU를 사용합니다. 애플리케이션당 리소스: vCPU: 16 RAM: 64GB VRAM: 24GB 테넌시: 최대 1개의 동시 스트림 세션을 지원합니다.
gen6n_ultra_win2022	g6.2xlarge Amazon EC2 인스턴스의 Windows 런타임	(NVIDIA, ultra) 3D 장면 복잡도가 높음 수준인 애플리케이션을 지원합니다. Microsoft Windows Server 2022 Base에서 애플리케이션을 실행하고 DirectX 12를 지원합니다. Unreal Engine 5.6 버전 이하, 32비트 및 64비트 애플리케이션, 치트 방지 기술과 호환됩니다.

스트림 클래스	Amazon EC2 구성	설명
		<p>니다. NVIDIA L4 Tensor Core GPU를 사용합니다.</p> <p>애플리케이션당 리소스: vCPU: 8 RAM: 32GB VRAM: 24GB</p> <p>테넨시: 최대 1개의 동시 스트림 세션을 지원합니다.</p>
gen6n_medium_win2022	1:1 테넨시가 있는 g6f.2xlarge Amazon EC2 인스턴스의 Windows 런타임	<p>(NVIDIA, 스몰) 3D 장면 복잡성이 낮은 애플리케이션을 지원합니다. Microsoft Windows Server 2022 Base에서 애플리케이션을 실행합니다. NVIDIA L4 Tensor Core GPU를 사용합니다.</p> <p>애플리케이션당 리소스: vCPU: 2 RAM: 8GB VRAM: 3GB.</p> <p>테넨시: 최대 1개의 동시 스트림 세션을 지원합니다.</p>
gen6n_small_win2022	1:1 테넨시가 있는 g6f.large Amazon EC2 인스턴스의 Windows 런타임	<p>(NVIDIA, 스몰) 3D 장면 복잡성이 낮은 애플리케이션을 지원합니다. Microsoft Windows Server 2022 Base에서 애플리케이션을 실행합니다. NVIDIA L4 Tensor Core GPU를 사용합니다.</p> <p>애플리케이션당 리소스: vCPU: 2 RAM: 8GB VRAM: 3GB.</p> <p>테넨시: 최대 1개의 동시 스트림 세션을 지원합니다.</p>

스트림 클래스	Amazon EC2 구성	설명
gen5n_win 2022	g5.2xlarge Amazon EC2 인스턴스의 Windows 런타임	<p>(NVIDIA, ultra) 3D 장면 복잡도가 매우 높음 수준인 애플리케이션을 지원합니다. Microsoft Windows Server 2022 Base에서 애플리케이션을 실행하고 DirectX 12 및 DirectX 11을 지원합니다. Unreal Engine 버전 5.6 이하, 32비트 및 64비트 애플리케이션, 치트 방지 기술을 지원합니다. NVIDIA A10G Tensor Core GPU를 사용합니다.</p> <p>애플리케이션당 리소스: vCPU: 8 RAM: 32GB VRAM: 24GB</p> <p>테넌시: 동시 스트림 세션 1개를 지원합니다.</p>
gen4n_win 2022	g4dn.2xlarge Amazon EC2 인스턴스의 Windows 런타임	<p>(NVIDIA, ultra) 3D 장면 복잡도가 높음 수준인 애플리케이션을 지원합니다. Microsoft Windows Server 2022 Base에서 애플리케이션을 실행하고 DirectX 12 및 DirectX 11을 지원합니다. Unreal Engine 버전 5.6 이하, 32비트 및 64비트 애플리케이션, 치트 방지 기술을 지원합니다. NVIDIA T4 Tensor Core GPU를 사용합니다.</p> <p>애플리케이션당 리소스: vCPU: 8 RAM: 32GB VRAM: 16GB</p> <p>테넌시: 동시 스트림 세션 1개를 지원합니다.</p>

Linux 및 Proton 스트림 클래스

스트림 클래스	Amazon EC2 구성	설명
gen6n_pro	g6.4xlarge Amazon EC2 인스턴스의 Linux 런타임	(NVIDIA, pro) 최대 리소스가 필요하며 3D 장면 복잡성이 매우 높음 수준인 애플리케이션

스트림 클래스	Amazon EC2 구성	설명
		<p>션을 지원합니다. NVIDIA L4 Tensor Core GPU를 사용합니다.</p> <p>애플리케이션당 리소스: vCPU: 16 RAM: 64GB VRAM: 24GB</p> <p>테넨시: 최대 1개의 동시 스트림 세션을 지원합니다.</p>
gen6n_ultra	g6.2xlarge Amazon EC2 인스턴스의 Linux 런타임	<p>(NVIDIA, ultra) 3D 장면 복잡도가 높음 수준인 애플리케이션을 지원합니다. NVIDIA L4 Tensor Core GPU를 사용합니다.</p> <p>애플리케이션당 리소스: vCPU: 8 RAM: 32GB VRAM: 24GB</p> <p>테넨시: 최대 1개의 동시 스트림 세션을 지원합니다.</p>
gen6n_high	2:1 테넨시가 있는 g6.2xlarge Amazon EC2 인스턴스의 Linux 런타임	<p>(NVIDIA, high) 3D 장면 복잡도가 보통-높음 수준인 애플리케이션을 지원합니다. NVIDIA L4 Tensor Core GPU를 사용합니다.</p> <p>애플리케이션당 리소스: vCPU: 4 RAM: 16GB VRAM: 12GB</p> <p>테넨시: 최대 2개의 동시 스트림 세션을 지원합니다.</p>
gen6n_medium	4:1 테넨시를 사용하는 g6.2xlarge Amazon EC2 인스턴스의 Linux 런타임	<p>(NVIDIA, medium) 3D 장면 복잡도가 보통 수준인 애플리케이션을 지원합니다. NVIDIA L4 Tensor Core GPU를 사용합니다.</p> <p>애플리케이션당 리소스: vCPU: 2 RAM: 8GB VRAM: 6GB</p> <p>테넨시: 최대 4개의 동시 스트림 세션을 지원합니다.</p>

스트림 클래스	Amazon EC2 구성	설명
gen6n_small	12:1 테넌시가 있는 g6.4xlarge Amazon EC2 인스턴스의 Linux 런타임	<p>(NVIDIA, small) 3D 장면 복잡도가 경량 수준이며 CPU 사용량이 적은 애플리케이션을 지원합니다. NVIDIA L4 Tensor Core GPU를 사용합니다.</p> <p>애플리케이션당 리소스: vCPU: 1 RAM: 4GB VRAM: 2GB</p> <p>테넌시: 최대 12개의 동시 스트림 세션을 지원합니다.</p>
gen5n_ultra	g5.2xlarge Amazon EC2 인스턴스의 Linux 런타임	<p>(NVIDIA, ultra) 3D 장면 복잡도가 매우 높음 수준인 애플리케이션을 지원합니다. NVIDIA A10G Tensor Core GPU를 사용합니다.</p> <p>애플리케이션당 리소스: vCPU: 8 RAM: 32GB VRAM: 24GB</p> <p>테넌시: 동시 스트림 세션 1개를 지원합니다.</p>
gen5n_high	2:1 테넌시가 있는 g5.2xlarge Amazon EC2 인스턴스의 Linux 런타임	<p>(NVIDIA, high) 3D 장면 복잡도가 보통-높음 수준인 애플리케이션을 지원합니다. NVIDIA A10G Tensor Core GPU를 사용합니다.</p> <p>애플리케이션당 리소스: vCPU: 4 RAM: 16GB VRAM: 12GB</p> <p>테넌시: 최대 2개의 동시 스트림 세션을 지원합니다.</p>
gen4n_ultra	g4dn.2xlarge Amazon EC2 인스턴스의 Linux 런타임	<p>(NVIDIA, ultra) 3D 장면 복잡도가 높음 수준인 애플리케이션을 지원합니다. NVIDIA T4 Tensor Core GPU를 사용합니다.</p> <p>애플리케이션당 리소스: vCPU: 8 RAM: 32GB VRAM: 16GB</p> <p>테넌시: 동시 스트림 세션 1개를 지원합니다.</p>

스트림 클래스	Amazon EC2 구성	설명
gen4n_high	2:1 테넌시를 사용하는 g4dn.2xlarge Amazon EC2 인스턴스의 Linux 런타임	<p>(NVIDIA, high) 3D 장면 복잡도가 보통-높음 수준인 애플리케이션을 지원합니다. NVIDIA T4 Tensor Core GPU를 사용합니다.</p> <p>애플리케이션당 리소스: vCPU: 4 RAM: 16GB VRAM: 8GB</p> <p>테넌시: 최대 2개의 동시 스트림 세션을 지원합니다.</p>

Amazon GameLift Streams에서 첫 번째 스트림 시작

이 자습서에서는 Amazon GameLift Streams를 시작하여 애플리케이션 또는 게임을 스트리밍하는 단계를 안내합니다. Amazon GameLift Streams는 애플리케이션을 실행하고 최종 사용자의 웹 브라우저로 직접 스트리밍합니다. 스트리밍하려는 애플리케이션을 업로드하고 구성하는 방법과 Amazon GameLift Streams 스트림 방식을 관리하는 방법을 알아봅니다. 결국 Amazon GameLift Streams 콘솔에서 직접 상호 작용하여 애플리케이션이 Amazon GameLift Streams에서 스트리밍되는 방식을 테스트합니다.

⚠ 시작하기 전에 Amazon GameLift Streams 요금을 이해합니다.

[요금 페이지에서](#) Amazon GameLift 스트림 비용을 확인할 수 있습니다. 자세한 내용은 [Amazon GameLift Streams의 사용량 및 청구서 관리](#) 섹션을 참조하세요.

특히 다음과 같은 경우 Amazon GameLift Streams 사용 비용이 발생합니다.

- 에서 Amazon GameLift Streams 애플리케이션 생성 [2단계: Amazon GameLift Streams용 애플리케이션 구성](#)
- 에서 스트림 그룹 생성 [3단계: Amazon GameLift Streams가 애플리케이션을 스트리밍하는 방법 관리](#)

를 건너뛰지 마십시오 [5단계: 정리\(건너뛰지 않음\)](#). Amazon GameLift Streams 시도를 완료한 후 불필요한 요금이 부과되지 않도록 하려면 모든 리소스를 정리해야 합니다.

주제

- [사전 조건](#)
- [1단계: Amazon S3 버킷에 애플리케이션 업로드](#)
- [2단계: Amazon GameLift Streams용 애플리케이션 구성](#)
- [3단계: Amazon GameLift Streams가 애플리케이션을 스트리밍하는 방법 관리](#)
- [4단계: Amazon GameLift Streams에서 스트림 테스트](#)
- [5단계: 정리\(건너뛰지 않음\)](#)

사전 조건

자습서를 시작하기 전에 다음 작업을 완료합니다.

- 아직 관리 액세스 권한이 없는 경우 AWS 계정에 가입하고 관리 액세스 권한이 있는 사용자를 생성합니다. 이 작업에 대한 도움말은 이 가이드의 [설정](#) 주제를 참조하세요. AWS CLI 현재 Amazon GameLift Streams 웹 SDK를 다운로드하거나 설정할 필요가 없습니다. 를 사용하여 다음 단계를 완료합니다 AWS Management Console.
- 디지털 권한 관리(DRM) 없이 애플리케이션 콘텐츠 파일의 버전을 가져옵니다. 실행 파일 및 자산을 포함하여 애플리케이션을 실행하는 데 필요한 파일을 폴더로 수집하되 폴더를 압축하지 않습니다.

1단계: Amazon S3 버킷에 애플리케이션 업로드

Amazon GameLift Streams는 Amazon Simple Storage Service(Amazon S3)를 사용하여 애플리케이션 또는 게임 파일을 클라우드에 저장하고 스트리밍을 위해 액세스합니다. 이 단계에서는 애플리케이션 파일을 Amazon S3 버킷에 업로드합니다. Amazon S3 콘솔에서 이 단계를 완료합니다.

Note

Amazon GameLift Streams에 필요한 Amazon S3 스토리지 클래스가 기본 S3 표준입니다. S3 Intelligent-Tiering을 통해 Infrequent Access 또는 Archive Access로 이동하는 S3 Glacier 또는 객체와 같은 다른 스토리지 클래스는 Amazon GameLift Streams에서 지원되지 않습니다. S3 스토리지 비용을 최적화하려면 완료 [2단계: Amazon GameLift Streams용 애플리케이션 구성](#) 후 애플리케이션이 준비 상태인 경우 S3 버킷에서 애플리케이션을 삭제할 수 있습니다.

애플리케이션 제한 사항

이름	기본값	조정 가능	설명
애플리케이션당 파일	파일 30,000개	예*	이 계정에서 애플리케이션 별로 사용할 수 있는 최대 파일 수입니다.
단일 파일 크기	80GiB	아니요	애플리케이션에서 단일 파일의 최대 크기입니다. 참고로 기비바이트(GiB)는 1024*1024*1024바이트입니다.
애플리케이션 크기	100GiB	예*	이 계정에서 Amazon GameLift Streams 애플리케이션의 최대 총 크기입니다. 참고로 기비바이트(GiB)는 1024*1024*1024 바이트입니다.

*증가를 요청하려면 로그인 AWS Management Console 하고 [Amazon GameLift Streams](#)에 대한 Service Quotas 콘솔을 엽니다. 그러면 적용된 계정 수준 할당량 값 열에서 현재 할당량을 검토하고 값 증가 요청을 제출할 수 있습니다.

Amazon S3에 애플리케이션을 업로드하려면

1. 에 로그인 AWS Management Console 하고 <https://console.aws.amazon.com/s3/> Amazon S3 콘솔을 엽니다.
2. Amazon S3 버킷을 생성합니다. 버킷 이름을 입력하고를 선택합니다 AWS 리전. 이 리전은 나중에 생성할 애플리케이션 및 스트림 그룹과 동일해야 합니다. Amazon GameLift Streams [AWS 리전 Amazon GameLift Streams에서 지원하는 및 스트리밍 위치](#)를 사용할 수 AWS 리전 있는 목록은 섹션을 참조하세요. 나머지 필드의 경우 기본 설정을 유지합니다.

자세한 지침은 Amazon Simple Storage Service 사용 설명서의 [버킷 생성](#)을 참조하세요.

3. 새 버킷을 열고 애플리케이션 파일과 함께 폴더를 업로드합니다.

⚠ Warning

애플리케이션 파일을 압축되지 않은 폴더로 업로드해야 합니다. .zip 폴더를 업로드하지 마십시오.

⚠ Warning

업로드한 애플리케이션 파일이 올바른 파일이고 애플리케이션 파일 크기 제한 내에 있는지 확인합니다. 나중에 파일을 업데이트하려면 반복해야 합니다. [2단계: Amazon GameLift Streams용 애플리케이션 구성](#)이 작업은 몇 분 정도 걸릴 수 있습니다.

2단계: Amazon GameLift Streams용 애플리케이션 구성

Amazon GameLift Streams의 애플리케이션이란 무엇입니까?

Amazon GameLift Streams 애플리케이션은 Amazon GameLift Streams 인프라에서 실행되고 클라우드 스트리밍을 통해 플레이어에게 게임 플레이 경험을 제공하는 게임 또는 대화형 애플리케이션을 포함하는 리소스입니다. 애플리케이션은 AWS 컴퓨팅 인스턴스에서 실행되고 인터넷을 통해 플레이어의 디바이스로 직접 스트리밍되는 게임 콘텐츠를 렌더링하므로 플레이어가 게임을 로컬에서 다운로드, 설치 또는 실행할 필요가 없습니다.

이 단계에서는 Amazon GameLift Streams 애플리케이션을 생성하여 Amazon GameLift Streams로 스트리밍할 애플리케이션을 구성합니다. Amazon GameLift Streams 애플리케이션을 생성할 때 Amazon S3 버킷에 업로드한 애플리케이션 폴더에 Amazon S3 URI를 제공하고 유효한 실행 파일 또는 스크립트 파일에 대한 상대 경로를 제공합니다. Amazon GameLift Streams 콘솔에서 이 단계를 완료합니다.

Amazon GameLift Streams 콘솔을 사용하여 Amazon GameLift Streams 애플리케이션을 생성하려면

1. [이](#) 로그인 AWS Management Console 하고 [Amazon GameLift Streams 콘솔](#)을 엽니다. 파일 세트를 업로드한 Amazon S3 버킷 AWS 리전 과 동일한을 선택합니다. 자세한 내용은 시작 안내서의 [리전 선택](#)을 참조하세요. AWS Management Console
2. 탐색 모음에서 애플리케이션을 선택한 다음 애플리케이션 생성을 선택합니다.
3. 런타임 설정에서 다음을 입력합니다.
 - 런타임 환경

애플리케이션을 실행할 런타임 환경입니다. Amazon GameLift 스트림은 Windows, Ubuntu 22.04 LTS 또는 [Proton](#)에서 실행할 수 있습니다.

생성 워크플로 이후에는 이 필드를 편집할 수 없습니다.

다음 런타임 환경 중 하나를 선택하거나.

- Linux 애플리케이션의 경우:
 - Ubuntu 22.04 LTS(UBUNTU, 22_04_LTS)
- Windows 애플리케이션의 경우:
 - Microsoft Windows Server 2022 Base(WINDOWS, 2022)
 - Proton 9.0-2(PROTON, 20250516)
 - Proton 8.0-5(PROTON, 20241007)
 - Proton 8.0-2c(PROTON, 20230704)

설명을 검토하고 비교 체크리스트를 사용하여 애플리케이션에 가장 적합한 런타임 환경을 선택할 수 있습니다.

4. 일반 설정에서 다음을 입력합니다.

a. 설명

사람이 읽을 수 있는 애플리케이션 레이블입니다. 이 값은 고유하지 않아도 됩니다. 모범 사례를 따라 애플리케이션에 의미 있는 설명, 이름 또는 레이블을 사용하세요. 이 필드는 언제든지 편집할 수 있습니다.

b. 기본 경로

Amazon S3 버킷 내 애플리케이션 루트 폴더의 Amazon S3 URI입니다. 폴더와 하위 폴더에는 빌드 실행 파일과 모든 지원 파일이 들어 있어야 합니다.

유효한 URI는 애플리케이션을 실행하고 스트리밍하는 데 필요한 모든 파일을 포함하는 버킷 접두사입니다. 예: mygamebuild라는 버킷에는 게임 빌드 파일의 전체 버전 3개가 각각 별도의 폴더에 들어 있습니다. mygamebuild-EN101 폴더에서 빌드를 스트리밍하려고 합니다. 이 예시에서 URI는 s3://amzn-s3-demo-bucket/mygamebuild-EN101입니다.

생성 워크플로 이후에는 이 필드를 편집할 수 없습니다.

c. 실행 가능한 시작 경로

Amazon GameLift Streams가 스트리밍할 실행 파일의 Amazon S3 URI입니다. 파일은 애플리케이션의 루트 폴더 내에 있어야 합니다. Windows 애플리케이션의 경우 파일은 유효한 Windows 실행 파일 또는 파일 이름이 .exe, .cmd 또는 .bat로 끝나는 배치 파일이어야 합니다. Linux 애플리케이션의 경우 파일은 유효한 Linux 바이너리 실행 파일이거나 shebang('#!')으로 시작하는 초기 인터프리터 줄이 포함된 스크립트여야 합니다.

생성 워크플로 이후에는 이 필드를 편집할 수 없습니다.

5. (선택 사항) 애플리케이션 로그 경로에 다음을 입력합니다.

a. 애플리케이션 로그 경로

저장하려는 로그가 들어 있는 애플리케이션 폴더 또는 파일의 경로입니다. 애플리케이션 기본 경로를 기준으로 각 로그 경로를 지정합니다. 이 기능을 사용하는 경우 모든 스트림 세션이 끝날 때 Amazon GameLift Streams는 사용자가 지정한 파일을 사용자가 지정한 이름의 Amazon S3 버킷에 복사합니다. 애플리케이션 폴더의 하위 폴더에서는 복사 작업이 재귀적으로 수행되지 않습니다.

로깅을 비활성화하려면 모든 애플리케이션 로그 경로를 제거하고 애플리케이션 로그 출력 대상을 지웁니다.

이 필드는 언제든지 편집할 수 있습니다.

b. 애플리케이션 로그 출력

Amazon GameLift Streams가 애플리케이션 로그 파일을 복사할 Amazon S3 버킷의 URI입니다. 애플리케이션 로그 경로를 지정하는 경우 이 필드는 필수입니다.

로깅을 비활성화하려면 모든 애플리케이션 로그 경로를 제거하고 애플리케이션 로그 출력 대상을 지웁니다.

이 필드는 언제든지 편집할 수 있습니다.

사용자를 대신하여 로그 파일을 저장하도록 Amazon GameLift Streams에 S3 버킷에 저장할 수 있는 권한을 부여해야 합니다. Amazon GameLift Streams에서 로깅용 버킷을 생성하도록 하면 생성 시 권한 정책이 자동으로 적용됩니다. 사용자가 자체 버킷을 제공하는 경우 권한 정책을 직접 적용해야 합니다.

버킷 권한 정책 템플릿

다음 정책 코드를 복사하여 애플리케이션 로그에 사용할 버킷에 적용합니다. `amzn-s3-demo-bucket`을 기존 S3 버킷의 이름으로 교체해야 합니다.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "PutPolicy",
      "Effect": "Allow",
      "Principal": {
        "Service": [
          "gameliftstreams.amazonaws.com"
        ]
      },
      "Action": "s3:PutObject",
      "Resource": "arn:aws:s3:::amzn-s3-demo-bucket/*",
      "Condition": {
        "StringEquals": {
          "aws:SourceAccount": "your 12-digit account id"
        }
      }
    }
  ]
}
```

6. (선택 사항) 태그에서이 애플리케이션에 태그를 할당합니다.

태그는 AWS 리소스를 구성하는 데 도움이 될 수 있는 레이블입니다. 자세한 내용은 [AWS 리소스 태그 지정](#)을 참조하세요.

예를 들어 애플리케이션 버전을 추적하려면와 같은 태그를 사용합니다 `application-version : my-game-1121`.

7. 애플리케이션 생성을 선택합니다.

Amazon GameLift Streams에서 애플리케이션을 준비하는 데 몇 분 정도 걸립니다. 애플리케이션 페이지에서 새 애플리케이션은 처리 중 상태입니다. 애플리케이션이 준비 상태이면 다음 단계인 로 이동할 수 있습니다 [3단계: Amazon GameLift Streams가 애플리케이션을 스트리밍하는 방법 관리](#).

요청이 오류를 반환하거나 애플리케이션이 생성되었지만 오류 상태인 경우 Amazon S3 및 Amazon GameLift Streams 모두에 대한 액세스를 포함하는 사용자 자격 증명으로 작업하고 있는지 확인합니다.

Note

애플리케이션이 준비 상태이면 새 애플리케이션에 영향을 주지 않고 Amazon S3 버킷의 애플리케이션 파일을 안전하게 삭제할 수 있습니다. 또한 스토리지 비용을 최적화하는 데도 도움이 됩니다. 자세한 내용은 [애플리케이션 삭제](#) 단원을 참조하십시오.

자세한 정보는 [Amazon GameLift Streams에서 애플리케이션 준비](#) 섹션을 참조하세요.

3단계: Amazon GameLift Streams가 애플리케이션을 스트리밍하는 방법 관리

스트림 그룹이란 무엇입니까?

스트림 그룹을 사용하여 Amazon GameLift Streams가 애플리케이션을 스트리밍하는 방법을 관리합니다. 스트림 그룹은 Amazon GameLift Streams가 애플리케이션을 최종 사용자에게 스트리밍하는 데 사용하는 컴퓨팅 리소스 모음입니다. 스트림 그룹을 생성할 때 게임을 실행할 하드웨어 구성(CPU, GPU, RAM), 게임을 실행할 수 있는 지리적 위치, 각 위치에서 동시에 실행할 수 있는 스트림 수(스트림 용량)를 지정합니다. 스트림 그룹을 생성할 때 애플리케이션을 연결하거나 이후까지 기다릴 수 있지만 스트림 그룹에서 스트리밍하려면 하나 이상의 애플리케이션을 연결해야 합니다. 스트림 그룹이 생성되면 Amazon GameLift Streams는 스트림 용량을 할당한 위치에 컴퓨팅 리소스를 할당합니다. 이 시점에서 스트림 그룹에 추가 애플리케이션을 연결하여 스트리밍할 애플리케이션을 선택할 수도 있습니다.

애플리케이션이 준비되면 Amazon GameLift Streams가 애플리케이션을 스트리밍하기 위한 컴퓨팅 리소스가 필요합니다. 이 단계에서는 스트림 그룹을 생성하여 Amazon GameLift Streams가 애플리케이션을 스트리밍하는 방법을 관리합니다. Amazon GameLift Streams 콘솔에서 이 단계를 완료합니다.

Amazon GameLift Streams 콘솔에서 스트림 그룹을 생성하려면

1. [에 로그인](#) AWS Management Console 하고 [Amazon GameLift Streams 콘솔](#)을 엽니다. 스트림 그룹을 생성할 AWS 리전을 선택합니다. 이 리전은 스트림 그룹으로 스트리밍하려는 애플리케이션의 리전과 동일해야 합니다. 자세한 내용은 시작 안내서의 [리전 선택](#)을 참조하세요. AWS Management Console

2. 생성 워크플로를 열려면 탐색 창에서 스트림 그룹을 선택한 다음 스트림 그룹 생성을 선택합니다.
3. 스트림 그룹 정의에 다음을 입력합니다.

- a. 설명

사람이 읽을 수 있는 스트림 그룹 레이블입니다. 이 값은 고유하지 않아도 됩니다. 모범 사례를 따라 스트림 그룹에 의미 있는 설명, 이름 또는 레이블을 사용하세요. 이 필드는 언제든지 편집할 수 있습니다.

- b. Tags

태그는 AWS 리소스를 구성하는 데 도움이 될 수 있는 레이블입니다. 자세한 내용은 [AWS 리소스 태그 지정](#)을 참조하세요.

4. 스트림 클래스 선택에서 스트림 그룹의 스트림 클래스를 선택합니다.

- 스트림 클래스 옵션

애플리케이션을 실행하고 스트리밍할 컴퓨팅 리소스의 유형입니다. 이 선택은 스트리밍 경험의 품질과 비용에 영향을 미칩니다. 스트림 그룹당 하나의 스트림 클래스만 지정할 수 있습니다. 애플리케이션에 가장 적합한 클래스를 선택합니다.

스트림 클래스	설명
gen6n_pro_win2022	<p>(NVIDIA, pro) 최대 리소스가 필요하며 3D 장면 복잡성이 매우 높음 수준인 애플리케이션을 지원합니다. Microsoft Windows Server 2022 Base에서 애플리케이션을 실행하고 DirectX 12를 지원합니다. Unreal Engine 5.6 버전 이하, 32비트 및 64비트 애플리케이션, 치트 방지 기술과 호환됩니다. NVIDIA L4 Tensor Core GPU를 사용합니다.</p> <p>애플리케이션당 리소스: vCPU: 16 RAM: 64GB VRAM: 24GB</p> <p>테넌시: 최대 1개의 동시 스트림 세션을 지원합니다.</p>
gen6n_pro	<p>(NVIDIA, pro) 최대 리소스가 필요하며 3D 장면 복잡성이 매우 높음 수준인 애플리케이션을 지원합니다. NVIDIA L4 Tensor Core GPU를 사용합니다.</p> <p>애플리케이션당 리소스: vCPU: 16 RAM: 64GB VRAM: 24GB</p>

스트림 클래스	설명
	테넌시: 최대 1개의 동시 스트림 세션을 지원합니다.
gen6n_ultra_win2022	<p>(NVIDIA, ultra) 3D 장면 복잡도가 높음 수준인 애플리케이션을 지원합니다. Microsoft Windows Server 2022 Base에서 애플리케이션을 실행하고 DirectX 12를 지원합니다. Unreal Engine 5.6 버전 이하, 32비트 및 64비트 애플리케이션, 치트 방지 기술과 호환됩니다. NVIDIA L4 Tensor Core GPU를 사용합니다.</p> <p>애플리케이션당 리소스: vCPU: 8 RAM: 32GB VRAM: 24GB</p> <p>테넌시: 최대 1개의 동시 스트림 세션을 지원합니다.</p>
gen6n_ultra	<p>(NVIDIA, ultra) 3D 장면 복잡도가 높음 수준인 애플리케이션을 지원합니다. NVIDIA L4 Tensor Core GPU를 사용합니다.</p> <p>애플리케이션당 리소스: vCPU: 8 RAM: 32GB VRAM: 24GB</p> <p>테넌시: 최대 1개의 동시 스트림 세션을 지원합니다.</p>
gen6n_high	<p>(NVIDIA, high) 3D 장면 복잡도가 보통-높음 수준인 애플리케이션을 지원합니다. NVIDIA L4 Tensor Core GPU를 사용합니다.</p> <p>애플리케이션당 리소스: vCPU: 4 RAM: 16GB VRAM: 12GB</p> <p>테넌시: 최대 2개의 동시 스트림 세션을 지원합니다.</p>
gen6n_medium	<p>(NVIDIA, medium) 3D 장면 복잡도가 보통 수준인 애플리케이션을 지원합니다. NVIDIA L4 Tensor Core GPU를 사용합니다.</p> <p>애플리케이션당 리소스: vCPU: 2 RAM: 8GB VRAM: 6GB</p> <p>테넌시: 최대 4개의 동시 스트림 세션을 지원합니다.</p>

스트림 클래스	설명
gen6n_small	<p>(NVIDIA, small) 3D 장면 복잡도가 경량 수준이며 CPU 사용량이 적은 애플리케이션을 지원합니다. NVIDIA L4 Tensor Core GPU를 사용합니다.</p> <p>애플리케이션당 리소스: vCPU: 1 RAM: 4GB VRAM: 2GB</p> <p>테넌시: 최대 12개의 동시 스트림 세션을 지원합니다.</p>
gen6n_medium_win2022	<p>(NVIDIA, 중간) 3D 장면 복잡성이 낮은 애플리케이션을 지원합니다. Microsoft Windows Server 2022 Base에서 애플리케이션을 실행합니다. NVIDIA L4 Tensor Core GPU를 사용합니다.</p> <p>애플리케이션당 리소스: vCPU: 8 RAM: 32GB VRAM: 6GB</p> <p>테넌시: 최대 1개의 동시 스트림 세션을 지원합니다.</p>
gen6n_small_win2022	<p>(NVIDIA, 스몰) 3D 장면 복잡성이 낮은 애플리케이션을 지원합니다. Microsoft Windows Server 2022 Base에서 애플리케이션을 실행합니다. NVIDIA L4 Tensor Core GPU를 사용합니다.</p> <p>애플리케이션당 리소스: vCPU: 2 RAM: 8GB VRAM: 3GB.</p> <p>테넌시: 최대 1개의 동시 스트림 세션을 지원합니다.</p>
gen5n_win2022	<p>(NVIDIA, ultra) 3D 장면 복잡도가 매우 높음 수준인 애플리케이션을 지원합니다. Microsoft Windows Server 2022 Base에서 애플리케이션을 실행하고 DirectX 12 및 DirectX 11을 지원합니다. Unreal Engine 버전 5.6 이하, 32비트 및 64비트 애플리케이션, 치트 방지 기술을 지원합니다. NVIDIA A10G Tensor Core GPU를 사용합니다.</p> <p>애플리케이션당 리소스: vCPU: 8 RAM: 32GB VRAM: 24GB</p> <p>테넌시: 동시 스트림 세션 1개를 지원합니다.</p>

스트림 클래스	설명
gen5n_high	<p>(NVIDIA, high) 3D 장면 복잡도가 보통-높음 수준인 애플리케이션을 지원합니다. NVIDIA A10G Tensor Core GPU를 사용합니다.</p> <p>애플리케이션당 리소스: vCPU: 4 RAM: 16GB VRAM: 12GB</p> <p>테넌시: 최대 2개의 동시 스트림 세션을 지원합니다.</p>
gen5n_ultra	<p>(NVIDIA, ultra) 3D 장면 복잡도가 매우 높음 수준인 애플리케이션을 지원합니다. NVIDIA A10G Tensor Core GPU를 사용합니다.</p> <p>애플리케이션당 리소스: vCPU: 8 RAM: 32GB VRAM: 24GB</p> <p>테넌시: 동시 스트림 세션 1개를 지원합니다.</p>
gen4n_win2022	<p>(NVIDIA, ultra) 3D 장면 복잡도가 높음 수준인 애플리케이션을 지원합니다. Microsoft Windows Server 2022 Base에서 애플리케이션을 실행하고 DirectX 12 및 DirectX 11을 지원합니다. Unreal Engine 버전 5.6 이하, 32비트 및 64비트 애플리케이션, 치트 방지 기술을 지원합니다. NVIDIA T4 Tensor Core GPU를 사용합니다.</p> <p>애플리케이션당 리소스: vCPU: 8 RAM: 32GB VRAM: 16GB</p> <p>테넌시: 동시 스트림 세션 1개를 지원합니다.</p>
gen4n_high	<p>(NVIDIA, high) 3D 장면 복잡도가 보통-높음 수준인 애플리케이션을 지원합니다. NVIDIA T4 Tensor Core GPU를 사용합니다.</p> <p>애플리케이션당 리소스: vCPU: 4 RAM: 16GB VRAM: 8GB</p> <p>테넌시: 최대 2개의 동시 스트림 세션을 지원합니다.</p>
gen4n_ultra	<p>(NVIDIA, ultra) 3D 장면 복잡도가 높음 수준인 애플리케이션을 지원합니다. NVIDIA T4 Tensor Core GPU를 사용합니다.</p> <p>애플리케이션당 리소스: vCPU: 8 RAM: 32GB VRAM: 16GB</p> <p>테넌시: 동시 스트림 세션 1개를 지원합니다.</p>

계속하려면 다음을 선택합니다.

5. 애플리케이션 연결에서 스트리밍할 애플리케이션을 선택하거나 "애플리케이션 없음"을 선택하여 나중에 하나를 선택합니다. 스트림 그룹을 생성한 후 편집하여 애플리케이션을 추가하거나 제거할 수 있습니다. Ready 상태이고 선택한 스트림 클래스와 호환되는 런타임이 있는 애플리케이션만 연결할 수 있습니다. 기본적으로 이러한 애플리케이션만 테이블에 표시됩니다. Ready 상태의 모든 애플리케이션을 보려면 드롭다운 목록에서 All runtimes를 선택합니다.

Note

애플리케이션이 나열되지 않으면 현재 AWS 리전 설정을 확인합니다. 애플리케이션을 동일한 리전에 있는 스트림 그룹에만 연결할 수 있습니다.

계속하려면 다음을 선택합니다.

6. 스트림 설정 구성의 위치 및 용량에서 스트림 그룹에 애플리케이션을 스트리밍할 용량이 있는 위치를 하나 이상 선택합니다. 기본적으로 기본 위치라고 하는 스트림 그룹을 생성하는 리전은 스트림 그룹에 이미 추가되었으며 제거할 수 없습니다. 추가하려는 각 위치 옆의 확인란을 선택하여 위치를 추가할 수 있습니다. 지연 시간을 줄이고 스트리밍 품질을 높이려면 사용자에게 더 가까운 위치를 선택해야 합니다.

각 위치에 대해 스트리밍 용량을 지정할 수 있습니다. 스트림 용량은 한 번에 활성화할 수 있는 동시 스트림 수를 나타냅니다. 각 스트림 그룹의 위치별로 스트림 용량을 설정합니다.

- 상시 가동 용량: 이 설정은 0이 아닌 경우 사용자에게 할당되고 서비스로 다시 릴리스되지 않는 최소 스트리밍 용량을 나타냅니다. 사용량이든 유휴 상태이든 관계없이 항상 이 기본 용량 수준에 대한 비용을 지불합니다.
- 최대 용량: 서비스가 할당할 수 있는 최대 용량을 나타냅니다. 새로 생성된 스트림을 시작하는데 몇 분 정도 걸릴 수 있습니다. 유휴 상태일 때 용량이 서비스로 다시 해제됩니다. 릴리스될 때까지 할당된 용량에 대해 비용을 지불합니다.
- 목표 유휴 용량: 향후 활동을 예상하여 서비스가 미리 할당하고 보관하는 유휴 용량을 나타냅니다. 이렇게 하면 용량 할당 지연으로부터 사용자를 보호할 수 있습니다. 의도적인 유휴 상태로 유지되는 이 용량에 대해 비용을 지불합니다.

어느 용량이든 조정하여 위치에 대한 사용자 수요의 변화를 충족하기 위해 언제든지 총 스트림 용량을 늘리거나 줄일 수 있습니다. Amazon GameLift Streams는 사용 가능한 경우 상시 작동 용량

풀의 유휴 사전 할당된 리소스를 사용하여 스트리밍 요청을 이행합니다. 모든 상시 작동 용량이 사용 중인 경우 Amazon GameLift Streams는 온디맨드 용량에 지정된 최대 수까지 추가 컴퓨팅 리소스를 프로비저닝합니다. 할당된 용량이 조정되면 변경 사항이 스트림 그룹의 총 비용에 반영됩니다.

연결된 애플리케이션은 활성화된 각 위치에 자동으로 복제됩니다. 원격 위치에서 스트림을 호스팅하려면 애플리케이션이 원격 위치에서 복제를 완료해야 합니다. 복제 상태를 확인하려면 스트림 그룹을 생성한 후 열고 연결된 애플리케이션 테이블의 복제 상태 열을 참조하세요. 현재 상태를 클릭하여 추가된 각 위치의 복제 상태를 확인합니다.

Note

애플리케이션 데이터는 이 스트림 그룹의 기본 위치를 포함하여 활성화된 모든 위치에 저장됩니다. 스트림 세션 데이터는 기본 위치와 스트리밍이 발생한 위치 모두에 저장됩니다.

7. 스트림 그룹 검토 및 생성에서 스트림 그룹 구성을 확인하고 필요에 따라 변경합니다. 모든 것이 올바르게 스트림 그룹 생성을 선택합니다.

자세한 정보는 [Amazon GameLift Streams 스트림 그룹을 사용하여 스트리밍 관리](#) 섹션을 참조하세요.

4단계: Amazon GameLift Streams에서 스트림 테스트

스트림 세션이란 무엇입니까?

스트림 자체를 나타냅니다. Amazon GameLift Streams가 서버에서 최종 사용자에게 전송하는 스트림의 인스턴스입니다. 스트림 세션은 스트림 그룹이 할당한 컴퓨팅 리소스 또는 스트림 용량에서 실행됩니다. 간단히 스트림이라고도 합니다.

Amazon GameLift Streams 콘솔에서 직접 실행하여 애플리케이션이 스트리밍되는 방식을 확인할 수 있습니다. 스트림을 시작하면 Amazon GameLift Streams는 스트림 그룹이 할당하는 컴퓨팅 리소스 중 하나를 사용합니다. 따라서 스트림 그룹에 사용 가능한 용량이 있어야 합니다.

Amazon GameLift Streams 콘솔에서 스트림을 테스트하려면

1. 에 로그인 AWS Management Console 하고 [Amazon GameLift Streams 콘솔](#)을 엽니다.
2. 여러 가지 방법으로 스트림을 테스트할 수 있습니다. 스트림 그룹 페이지 또는 스트림 테스트 페이지에서 시작하여 다음 단계를 따릅니다.

- a. 스트리밍에 사용할 스트림 그룹을 선택합니다.
 - b. 스트림 그룹 페이지에서 시작하는 경우 스트림 테스트를 선택합니다. 스트림 테스트 페이지에서 시작하는 경우 선택을 선택합니다. 그러면 선택한 스트림 그룹에 대한 스트림 구성 테스트 페이지가 열립니다.
 - c. 연결된 애플리케이션에서 애플리케이션을 선택합니다.
 - d. 위치에서 사용 가능한 용량이 있는 위치를 선택합니다.
 - e. (선택 사항) 프로그램 구성에서 시작 시 애플리케이션에 전달할 명령줄 인수 또는 환경 변수를 입력합니다.
 - f. 선택을 확인하고 스트림 테스트를 선택합니다.
3. 스트림이 로드된 후 스트림에서 다음 작업을 수행할 수 있습니다.
 - a. 마우스, 키보드, 게임패드(테스트 스트림에서 지원되지 않는 마이크 제외)와 같은 입력을 연결하려면 입력 연결을 선택합니다. 커서를 스트림 창으로 이동할 때 마우스를 자동으로 연결합니다.
 - b. 스트리밍 세션 중에 생성된 파일을 세션 종료 시 Amazon S3 버킷으로 내보내려면 파일 내보내기를 선택하고 버킷 세부 정보를 지정합니다. 내보낸 파일은 세션 페이지에서 찾을 수 있습니다.
 - c. 스트림을 전체 화면으로 보려면 전체 화면을 선택합니다. 이 작업을 되돌리려면 이스케이프를 누릅니다.
 4. 스트림을 종료하려면 세션 종료를 선택합니다. 스트림 연결이 끊어지면 스트림 용량을 사용하여 다른 스트림을 시작할 수 있게 됩니다.

Note

Amazon GameLift Streams 콘솔의 스트림 테스트 기능은 마이크를 지원하지 않습니다.

5단계: 정리(건너뛰지 않음)

불필요한 비용 방지

스트림 그룹에 용량이 할당되면 해당 용량이 사용되지 않더라도 비용이 발생합니다. 불필요한 비용을 방지하려면 스트림 그룹 용량을 필요한 크기로 조정하세요. 개발 중에는 사용하지

않을 때 상시 가동 용량을 0으로 조정하는 것이 좋습니다. 자세한 정보는 [Amazon GameLift Streams 비용 관리 모범 사례](#) 섹션을 참조하세요.

자습서를 완료하고 더 이상 애플리케이션을 스트리밍할 필요가 없는 경우 다음 단계에 따라 Amazon GameLift Streams 리소스를 정리합니다.

Amazon GameLift Streams 콘솔을 사용하여 스트림 그룹을 삭제하려면

1. 에 로그인 AWS Management Console 하고 [Amazon GameLift Streams 콘솔](#)을 엽니다.
2. 기존 스트림 그룹 목록을 보려면 탐색 창에서 스트림 그룹을 선택합니다.
3. 삭제할 스트림 그룹의 이름을 선택합니다.
4. 스트림 그룹 세부 정보 페이지에서 삭제를 선택합니다.
5. 삭제 대화 상자에서 삭제 작업을 확인합니다.

Amazon GameLift Streams는 컴퓨팅 리소스를 해제하고 스트림 그룹을 삭제하기 시작합니다. 이 시간 동안 스트림 그룹은 삭제 중 상태입니다. Amazon GameLift Streams가 스트림 그룹을 삭제한 후에는 더 이상 검색할 수 없습니다.

Amazon GameLift Streams 콘솔을 사용하여 애플리케이션을 삭제하려면

1. 에 로그인 AWS Management Console 하고 [Amazon GameLift Streams 콘솔](#)을 엽니다.
2. 탐색 모음에서 애플리케이션을 선택하여 기존 애플리케이션 목록을 봅니다. 삭제할 애플리케이션을 선택합니다.
3. 애플리케이션 세부 정보 페이지에서 삭제를 선택합니다.
4. 삭제 대화 상자에서 삭제 작업을 확인합니다.

Amazon GameLift Streams가 애플리케이션 삭제를 시작합니다. 이 시간 동안 애플리케이션은 Deleting 상태입니다. Amazon GameLift Streams가 애플리케이션을 삭제한 후에는 더 이상 검색할 수 없습니다.

자세한 정보는 [스트림 그룹 삭제](#) 및 [애플리케이션 삭제](#) 섹션을 참조하세요.

Amazon GameLift Streams를 사용하여 스트림 관리

이 섹션에서는 Amazon GameLift Streams로 스트리밍하는 방법에 대한 자세한 정보를 제공합니다. 스트리밍 리소스(애플리케이션 및 스트림 그룹), 스트리밍 규모를 조정할 속성(스트림 용량 및 위치), 스트림 자체(스트림 세션)에 대해 알아봅니다. Amazon GameLift Streams 콘솔 또는 Amazon GameLift Streams CLI 명령을 사용하여 Amazon GameLift Streams로 스트리밍을 설정하는 데 필요한 모든 작업을 처리할 수 있습니다. GameLift

Amazon GameLift Streams를 처음 사용하는 경우 전체 워크플로를 안내 [Amazon GameLift Streams에서 첫 번째 스트림 시작](#)하는 섹션을 참조하세요.

주제

- [주요 개념](#)
- [Amazon GameLift Streams에서 애플리케이션 준비](#)
- [Amazon GameLift Streams 스트림 그룹을 사용하여 스트리밍 관리](#)
- [다중 애플리케이션 스트림 그룹 개요](#)
- [Amazon GameLift Streams로 스트림 세션 시작](#)
- [스트림 세션 파일 내보내기](#)

주요 개념

애플리케이션

Amazon GameLift Streams 애플리케이션은 Amazon GameLift Streams 인프라에서 실행되고 클라우드 스트리밍을 통해 플레이어에게 게임 플레이 경험을 제공하는 게임 또는 대화형 애플리케이션을 포함하는 리소스입니다. 애플리케이션은 AWS 컴퓨팅 인스턴스에서 실행되고 인터넷을 통해 플레이어의 디바이스로 직접 스트리밍되는 게임 콘텐츠를 렌더링하므로 플레이어가 게임을 로컬에서 다운로드, 설치 또는 실행할 필요가 없습니다.

다중 애플리케이션 스트림 그룹

여러 애플리케이션에 연결된 스트림 그룹입니다. 이 many-to-one 관계를 사용하면 단일 스트림 그룹에 설정한 것과 동일한 구성을 사용하여 여러 애플리케이션을 스트리밍할 수 있습니다. 스트림 세션을 시작할 때 연결된 애플리케이션을 지정합니다. 그런 다음 Amazon GameLift Streams는 이 스트림 그룹에서 사용 가능한 스트림 용량을 사용하여 해당 애플리케이션을 스트리밍합니다.

다중 위치 스트림 그룹

기본 위치(스트림 그룹을 AWS 리전 생성한 위치) 외에도 여러 위치에서 애플리케이션 및 스트림 세션을 호스팅하도록 구성된 스트림 그룹입니다. 각 위치의 용량을 관리합니다.

멀티테넌시

테넌시는 Amazon GameLift Streams의 단일 컴퓨팅 리소스에서 지원할 수 있는 동시 스트림 수를 나타냅니다. 다중 테넌시는 여러 사용자가 동일한 기본 하드웨어 리소스를 공유할 수 있는 기능이며, 이는 최대 하드웨어 기능이 필요하지 않은 애플리케이션에 비용 효율적인 옵션일 수 있습니다. 다중 테넌시가 있는 스트림 클래스는 하나의 리소스 비용으로 여러 스트림을 호스팅할 수 있습니다. "High" 스트림 클래스는 다중 테넌시를 지원하므로 단일 컴퓨팅 리소스에서 두 애플리케이션을 동시에 실행할 수 있지만 "Ultra" 스트림 클래스는 다중 테넌시를 지원하지 않습니다.

스트림 그룹

스트림 그룹을 사용하여 Amazon GameLift Streams가 애플리케이션을 스트리밍하는 방법을 관리합니다. 스트림 그룹은 Amazon GameLift Streams가 애플리케이션을 최종 사용자에게 스트리밍하는 데 사용하는 컴퓨팅 리소스 모음입니다. 스트림 그룹을 생성할 때 게임을 실행할 하드웨어 구성(CPU, GPU, RAM), 게임을 실행할 수 있는 지리적 위치, 각 위치에서 동시에 실행할 수 있는 스트림 수(스트림 용량)를 지정합니다. 스트림 그룹을 생성할 때 애플리케이션을 연결하거나 이후까지 기다릴 수 있지만 스트림 그룹에서 스트리밍하려면 하나 이상의 애플리케이션을 연결해야 합니다. 스트림 그룹이 생성되면 Amazon GameLift Streams는 스트림 용량을 할당한 위치에 컴퓨팅 리소스를 할당합니다. 이 시점에서 스트림 그룹에 추가 애플리케이션을 연결하여 스트리밍할 애플리케이션을 선택할 수도 있습니다.

스트림 용량

한 번에 활성화할 수 있는 동시 스트림 수를 나타냅니다. 각 스트림 그룹의 위치별로 스트림 용량을 설정합니다. 상시 가동 용량과 최대 용량을 구성합니다. 최대 용량은 상시 가동 용량과 필요에 따라 프로비저닝된 추가 용량을 결합하여 가능한 총 스트림을 나타냅니다.

스트림 세션

스트림 자체를 나타냅니다. Amazon GameLift Streams가 서버에서 최종 사용자에게 전송하는 스트림의 인스턴스입니다. 스트림 세션은 스트림 그룹이 할당한 컴퓨팅 리소스 또는 스트림 용량에서 실행됩니다. 간단히 스트림이라고도 합니다.

Amazon GameLift Streams에서 애플리케이션 준비

Amazon GameLift Streams로 스트리밍을 설정하려면 먼저 스트리밍하려는 게임 또는 기타 애플리케이션을 업로드한 다음 Amazon GameLift Streams 내에서 애플리케이션 리소스를 구성하여 게임에 대

한 메타데이터를 정의합니다. Amazon GameLift Streams 애플리케이션은 업로드한 파일(실행 파일 및 지원 파일)과 스트리밍 시 실행할 실행 파일을 Amazon GameLift Streams에 지시하는 구성으로 구성됩니다.

각 Amazon GameLift Streams 애플리케이션은 콘텐츠의 단일 버전을 나타냅니다. 버전이 여러 개인 경우 각 버전에 대해 별도의 애플리케이션을 생성해야 합니다. 애플리케이션을 생성한 후에는 파일을 업데이트할 수 없습니다. 실행 파일 또는 지원 파일을 업데이트해야 하는 경우 새 Amazon GameLift Streams 애플리케이션을 생성해야 합니다.

업로드하기 전에

Amazon GameLift Streams 애플리케이션을 생성하기 전에 게임이 다음 제한 사항을 준수하는지 확인합니다.

명칭	기본값	조정 가능	Description
애플리케이션당 파일	파일 30,000개	예*	이 계정의 애플리케이션에서 보유할 수 있는 최대 파일 수입니다.
단일 파일 크기	80GiB	아니요	애플리케이션에서 단일 파일의 최대 크기입니다. 참고로 기비바이트(GiB)는 1024*1024*1024바이트와 같습니다.
애플리케이션 크기	100GiB	예*	이 계정에서 Amazon GameLift Streams 애플리케이션의 최대 총 크기입니다. 참고로 기비바이트(GiB)는 1024*1024*1024 바이트와 같습니다.

*증가를 요청하려면 로그인 AWS Management Console 하고 Service Quotas 콘솔을 [Amazon GameLift Streams](#)로 엽니다. 그러면 적용된 계정 수준 할당량 값 열에서 현재 할당량을 검토하고 값 증가 요청을 제출할 수 있습니다.

Note

시간과 노력을 절약하려면 업로드할 준비가 된 파일이 애플리케이션의 올바른 버전인지 확인합니다. 나중에 새 버전을 업로드할 수 있지만 각 버전에 대해 [애플리케이션 만들기](#) 단계를 반복해야 합니다.

Amazon S3 버킷에 애플리케이션 업로드

이제 Amazon GameLift Streams용 게임을 준비했으므로 AWS 계정의 Amazon Simple Storage Service(Amazon S3) 버킷에 업로드할 차례입니다.

Note

Amazon GameLift Streams에 필요한 Amazon S3 스토리지 클래스가 기본 S3 표준입니다. S3 Intelligent-Tiering을 통해 Infrequent Access 또는 Archive Access로 이동하는 S3 Glacier 또는 객체와 같은 다른 스토리지 클래스는 Amazon GameLift Streams에서 지원되지 않습니다. S3 스토리지 비용을 최적화하려면 완료 [애플리케이션 만들기](#) 후 애플리케이션이 준비 상태인 경우 S3 버킷에서 애플리케이션을 삭제할 수 있습니다.

Amazon S3에 애플리케이션을 업로드하려면

1. 에 로그인 AWS Management Console 하고 <https://console.aws.amazon.com/s3/> Amazon S3 콘솔을 엽니다.
2. Amazon S3 버킷을 생성합니다. 버킷 이름을 입력하고를 선택합니다 AWS 리전. 이 리전은 나중에 생성할 애플리케이션 및 스트림 그룹과 동일해야 합니다. [Amazon GameLift Streams AWS 리전 Amazon GameLift Streams에서 지원하는 및 스트리밍 위치](#)를 사용할 수 AWS 리전 있는 목록은 섹션을 참조하세요. 나머지 필드의 경우 기본 설정을 유지합니다.

자세한 지침은 Amazon Simple Storage Service 사용 설명서의 [버킷 생성](#)을 참조하세요.

3. 새 버킷을 열고 애플리케이션 파일과 함께 폴더를 업로드합니다.

⚠ Warning

애플리케이션 파일을 압축되지 않은 폴더로 업로드해야 합니다. .zip 폴더를 업로드하지 마십시오.

애플리케이션 만들기

Amazon GameLift Streams 애플리케이션은 Amazon GameLift Streams 인프라에서 실행되고 클라우드 스트리밍을 통해 플레이어에게 게임 플레이 경험을 제공하는 게임 또는 대화형 애플리케이션을 포함하는 리소스입니다. 애플리케이션은 AWS 컴퓨팅 인스턴스에서 실행되고 인터넷을 통해 플레이어의 디바이스로 직접 스트리밍되는 게임 콘텐츠를 렌더링하므로 플레이어가 게임을 로컬에서 다운로드, 설치 또는 실행할 필요가 없습니다.

Amazon GameLift Streams 애플리케이션을 생성할 때 Amazon S3 버킷에 업로드한 애플리케이션 폴더에 Amazon S3 URI를 제공하고 유효한 실행 파일 또는 스크립트 파일에 대한 상대 경로를 제공합니다.

Amazon GameLift Streams는 애플리케이션 파일을 Amazon S3 버킷의 파일과 동기화하지 않습니다. Amazon GameLift Streams 애플리케이션의 파일을 업데이트하려면 새 Amazon GameLift Streams 애플리케이션을 생성해야 합니다.

Console

Amazon GameLift Streams 콘솔을 사용하여 Amazon GameLift Streams 애플리케이션을 생성하려면

1. 에 로그인 AWS Management Console 하고 [Amazon GameLift Streams 콘솔](#)을 엽니다. 파일 세트를 업로드한 Amazon S3 버킷 AWS 리전 과 동일한을 선택합니다. 자세한 내용은 시작 안내서의 [리전 선택](#)을 참조하세요. AWS Management Console
2. 탐색 모음에서 애플리케이션을 선택한 다음 애플리케이션 생성을 선택합니다.
3. 런타임 설정에서 다음을 입력합니다.
 - 런타임 환경

애플리케이션을 실행할 런타임 환경입니다. Amazon GameLift 스트림은 Windows, Ubuntu 22.04 LTS 또는 [Proton](#)에서 실행할 수 있습니다.

생성 워크플로 이후에는 이 필드를 편집할 수 없습니다.

다음 런타임 환경 중 하나를 선택하거나.

- Linux 애플리케이션의 경우:
 - Ubuntu 22.04 LTS(UBUNTU, 22_04_LTS)
- Windows 애플리케이션의 경우:
 - Microsoft Windows Server 2022 Base(WINDOWS, 2022)
 - Proton 9.0-2(PROTON, 20250516)
 - Proton 8.0-5(PROTON, 20241007)
 - Proton 8.0-2c(PROTON, 20230704)

설명을 검토하고 비교 체크리스트를 사용하여 애플리케이션에 가장 적합한 런타임 환경을 선택할 수 있습니다.

4. 일반 설정에서 다음을 입력합니다.

a. 설명

사람이 읽을 수 있는 애플리케이션 레이블입니다. 이 값은 고유하지 않아도 됩니다. 모범 사례를 따라 애플리케이션에 의미 있는 설명, 이름 또는 레이블을 사용하세요. 이 필드는 언제든지 편집할 수 있습니다.

b. 기본 경로

Amazon S3 버킷 내 애플리케이션 루트 폴더의 Amazon S3 URI입니다. 폴더와 하위 폴더에는 빌드 실행 파일과 모든 지원 파일이 들어 있어야 합니다.

유효한 URI는 애플리케이션을 실행하고 스트리밍하는 데 필요한 모든 파일을 포함하는 버킷 접두사입니다. 예: 라는 버킷mygamebuild에는 게임 빌드 파일의 세 가지 전체 버전이 포함되어 있으며, 각 버전은 별도의 폴더에 있습니다. mygamebuild-EN101 폴더에서 빌드를 스트리밍하려고 합니다. 이 예시에서 URI는 s3://amzn-s3-demo-bucket/mygamebuild-EN101입니다.

생성 워크플로 이후에는 이 필드를 편집할 수 없습니다.

c. 실행 가능한 시작 경로

Amazon GameLift Streams가 스트리밍할 실행 파일의 Amazon S3 URI입니다. 파일은 애플리케이션의 루트 폴더 내에 있어야 합니다. Windows 애플리케이션의 경우 파일은 유효한 Windows 실행 파일 또는 파일 이름이 .exe, .cmd 또는 .bat로 끝나는 배치 파일이어야 합니다. Linux 애플리케이션의 경우 파일은 유효한 Linux 바이너리 실행 파일이거나 shebang('#!')으로 시작하는 초기 인터프리터 줄이 포함된 스크립트여야 합니다.

생성 워크플로 이후에는 이 필드를 편집할 수 없습니다.

5. (선택 사항) 애플리케이션 로그 경로에 다음을 입력합니다.

a. 애플리케이션 로그 경로

저장하려는 로그가 들어 있는 애플리케이션 폴더 또는 파일의 경로입니다. 애플리케이션 기본 경로를 기준으로 각 로그 경로를 지정합니다. 이 기능을 사용하는 경우 모든 스트림 세션이 끝날 때 Amazon GameLift Streams는 사용자가 지정한 파일을 사용자가 지정한 이름의 Amazon S3 버킷에 복사합니다. 애플리케이션 폴더의 하위 폴더에서는 복사 작업이 재귀적으로 수행되지 않습니다.

로깅을 비활성화하려면 모든 애플리케이션 로그 경로를 제거하고 애플리케이션 로그 출력 대상을 지웁니다.

이 필드는 언제든지 편집할 수 있습니다.

b. 애플리케이션 로그 출력

Amazon GameLift Streams가 애플리케이션 로그 파일을 복사할 Amazon S3 버킷의 URI입니다. 애플리케이션 로그 경로를 지정하는 경우 이 필드는 필수입니다.

로깅을 비활성화하려면 모든 애플리케이션 로그 경로를 제거하고 애플리케이션 로그 출력 대상을 지웁니다.

이 필드는 언제든지 편집할 수 있습니다.

사용자를 대신하여 로그 파일을 저장하도록 Amazon GameLift Streams에 S3 버킷에 저장할 수 있는 권한을 부여해야 합니다. Amazon GameLift Streams에서 로깅용 버킷을 생성하도록 하면 생성 시 권한 정책이 자동으로 적용됩니다. 사용자가 자체 버킷을 제공하는 경우 권한 정책을 직접 적용해야 합니다.

버킷 권한 정책 템플릿

다음 정책 코드를 복사하여 애플리케이션 로그에 사용할 버킷에 적용합니다. `amzn-s3-demo-bucket`을 기존 S3 버킷의 이름으로 교체해야 합니다.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "PutPolicy",
      "Effect": "Allow",
      "Principal": {
        "Service": [
          "gameliftstreams.amazonaws.com"
        ]
      },
      "Action": "s3:PutObject",
      "Resource": "arn:aws:s3:::amzn-s3-demo-bucket/*",
      "Condition": {
        "StringEquals": {
          "aws:SourceAccount": "your 12-digit account id"
        }
      }
    }
  ]
}
```

6. (선택 사항) 태그에서이 애플리케이션에 태그를 할당합니다.

태그는 AWS 리소스를 구성하는 데 도움이 될 수 있는 레이블입니다. 자세한 내용은 [AWS 리소스 태그 지정](#)을 참조하세요.

예를 들어 애플리케이션 버전을 추적하려면와 같은 태그를 사용합니다 `application-version : my-game-1121`.

7. 애플리케이션 생성을 선택합니다.

CLI

사전 조건

사용자 자격 증명과 선택한 AWS CLI 사용자를 구성해야 합니다 AWS 리전. 설정 지침은 단원을 참조하십시오 [다운로드 AWS CLI](#).

를 사용하여 애플리케이션을 생성하려면 AWS CLI

에서 콘텐츠에 맞게 사용자 지정된 [CreateApplication](#) 명령을 AWS CLI 사용합니다.

```
aws gameliftstreams create-application \
  --description "MyGame v1" \
  --runtime-environment '{"Type":"PROTON", "Version":"20241007"}' \
  --executable-path "launcher.exe" \
  --application-source-uri "s3://amzn-s3-demo-bucket/example"
```

여기서 각 항목은 다음과 같습니다.

- **description:**

사람이 읽을 수 있는 애플리케이션 레이블입니다. 이 값은 고유하지 않아도 됩니다. 모범 사례를 따라 애플리케이션에 의미 있는 설명, 이름 또는 레이블을 사용하세요. 이 필드는 언제든지 편집할 수 있습니다.

- **runtime-environment:**

애플리케이션을 실행할 런타임 환경입니다. Amazon GameLift 스트림은 Windows, Ubuntu 22.04 LTS 또는 [Proton](#)에서 실행할 수 있습니다.

생성 워크플로 이후에는 이 필드를 편집할 수 없습니다.

다음 런타임 환경 중 하나를 선택하거나.

- Linux 애플리케이션의 경우:

- Ubuntu 22.04 LTS(Type=UBUNTU, Version=22_04_LTS)

- Windows 애플리케이션의 경우:

- Microsoft Windows Server 2022 Base(Type=WINDOWS, Version=2022)
- Proton 9.0-2(Type=PROTON, Version=20250516)
- Proton 8.0-5(Type=PROTON, Version=20241007)
- Proton 8.0-2c(Type=PROTON, Version=20230704)

- **application-source-uri:**

Amazon S3 버킷 내 애플리케이션 루트 폴더의 Amazon S3 URI입니다. 폴더와 하위 폴더에는 빌드 실행 파일과 모든 지원 파일이 들어 있어야 합니다.

유효한 URI는 애플리케이션을 실행하고 스트리밍하는 데 필요한 모든 파일을 포함하는 버킷 접두사입니다. 예: 라는 버킷mygamebuild에는 게임 빌드 파일의 세 가지 전체 버전이 포함되어 있으며, 각 버전은 별도의 폴더에 있습니다. mygamebuild-EN101 폴더에서 빌드를 스트리밍하려고 합니다. 이 예시에서 URI는 s3://amzn-s3-demo-bucket/mygamebuild-EN101입니다.

생성 워크플로 이후에는 이 필드를 편집할 수 없습니다.

- executable-path:

Amazon GameLift Streams가 스트리밍할 실행 파일의 상대 경로 및 파일 이름입니다. 를 기준으로 경로를 지정합니다application-source-uri. 파일은 애플리케이션의 루트 폴더 내에 있어야 합니다. Windows 애플리케이션의 경우 파일은 유효한 Windows 실행 파일 또는 파일 이름이 .exe, .cmd 또는 .bat로 끝나는 배치 파일이어야 합니다. Linux 애플리케이션의 경우 파일은 유효한 Linux 바이너리 실행 파일이거나 shebang('#!')으로 시작하는 초기 인터프리터 줄이 포함된 스크립트여야 합니다.

생성 워크플로 이후에는 이 필드를 편집할 수 없습니다.

요청이 성공하면 Amazon GameLift Streams는 다음과 유사한 응답을 반환합니다.

```
{
  "Arn": "arn:aws:gameliftstreams:us-west-2:111122223333:application/a-9ZY8X7Wv6",
  "Description": "MyGame v1",
  "RuntimeEnvironment": {
    "Type": "PROTON",
    "Version": "20241007"
  },
  "ExecutablePath": "launcher.exe",
  "ApplicationSourceUri": "s3://amzn-s3-demo-bucket/example",
  "Id": "a-9ZY8X7Wv6",
  "Status": "PROCESSING",
  "CreatedAt": "2022-11-18T15:47:11.924000-08:00",
  "LastUpdatedAt": "2022-11-18T15:47:11.924000-08:00"
}
```

애플리케이션의 상태를 확인하려면 다음 예제와 같이 [GetApplication](#) 명령을 호출합니다.

```
aws gameliftstreams get-application /
  --identifier a-9ZY8X7Wv6
```

Amazon GameLift Streams에서 애플리케이션을 준비하는 데 몇 분 정도 걸립니다. 이 시간 동안 새 애플리케이션은 처리 중 상태입니다. 애플리케이션이 준비 상태이면 다음 단계인 로 이동할 수 있습니다. [다스트림 그룹 생성](#).

요청이 오류를 반환하거나 애플리케이션이 생성되었지만 오류 상태에 있는 경우 Amazon S3 및 Amazon GameLift Streams 모두에 대한 액세스를 포함하는 사용자 자격 증명으로 작업하고 있는지 확인합니다.

Note

애플리케이션이 준비 상태이면 Amazon GameLift Streams가 애플리케이션 파일을 프라이빗 Amazon S3 버킷에 성공적으로 복사한 것입니다. 새 애플리케이션에 영향을 주지 않고 원래 애플리케이션 파일을 삭제할 수 있습니다. 또한 스토리지 비용을 최적화하는 데도 도움이 됩니다. 자세한 내용은 [애플리케이션 삭제](#) 단원을 참조하십시오.

애플리케이션 편집

준비 상태에서 모든 애플리케이션의 설정을 업데이트할 수 있습니다. 기존 애플리케이션을 변경하면 이러한 변경 사항이 새 스트림 그룹과 기존 스트림 그룹의 스트리밍 동작에 모두 영향을 미칩니다.

Console

Amazon GameLift Streams 콘솔에서 애플리케이션을 편집하려면

1. 탐색 모음에서 애플리케이션을 선택하여 기존 애플리케이션 목록을 봅니다. 편집할 애플리케이션을 선택합니다.
2. 애플리케이션 세부 정보 페이지에서 변경하려는 설정이 포함된 섹션을 찾아 그에 따라 태그 편집 또는 관리를 선택합니다.
3. 다음과 같은 설정을 변경할 수 있습니다.

간단한 설명

사람이 읽을 수 있는 애플리케이션 레이블입니다. 이 값은 고유하지 않아도 됩니다. 모범 사례를 따라 애플리케이션에 의미 있는 설명, 이름 또는 레이블을 사용하세요. 이 필드는 언제든지 편집할 수 있습니다.

애플리케이션 로그 경로

저장하려는 로그가 들어 있는 애플리케이션 폴더 또는 파일의 경로입니다. 애플리케이션 기본 경로를 기준으로 각 로그 경로를 지정합니다. 이 기능을 사용하는 경우 모든 스트림 세션이 끝날 때 Amazon GameLift Streams는 사용자가 지정한 파일을 사용자가 지정한 이름의 Amazon S3 버킷에 복사합니다. 애플리케이션 폴더의 하위 폴더에서는 복사 작업이 재귀적으로 수행되지 않습니다.

로깅을 비활성화하려면 모든 애플리케이션 로그 경로를 제거하고 애플리케이션 로그 출력 대상을 지웁니다.

이 필드는 언제든지 편집할 수 있습니다.

애플리케이션 로그 출력

Amazon GameLift Streams가 애플리케이션 로그 파일을 복사할 Amazon S3 버킷의 URI입니다. 애플리케이션 로그 경로를 지정하는 경우 이 필드는 필수입니다.

로깅을 비활성화하려면 모든 애플리케이션 로그 경로를 제거하고 애플리케이션 로그 출력 대상을 지웁니다.

이 필드는 언제든지 편집할 수 있습니다.

사용자를 대신하여 로그 파일을 저장하도록 Amazon GameLift Streams에 S3 버킷에 저장할 수 있는 권한을 부여해야 합니다. Amazon GameLift Streams에서 로깅용 버킷을 생성하도록 하면 생성 시 권한 정책이 자동으로 적용됩니다. 사용자가 자체 버킷을 제공하는 경우 권한 정책을 직접 적용해야 합니다.

자세한 내용은 [애플리케이션 로그 버킷 권한 정책](#) 단원을 참조하십시오.

태그

태그는 AWS 리소스를 구성하는 데 도움이 될 수 있는 레이블입니다. 자세한 내용은 [AWS 리소스 태그 지정](#)을 참조하십시오.

예를 들어 애플리케이션 버전을 추적하려면와 같은 태그를 사용합니다application-version : my-game-1121.

4. 변경 사항 저장을 선택합니다. Amazon GameLift Streams 콘솔이 애플리케이션 세부 정보 페이지로 돌아가고 업데이트된 설정이 표시됩니다.

CLI

사전 조건

사용자 자격 증명과 선택한 AWS CLI 사용 여부를 구성해야 합니다 AWS 리전. 설정 지침은 단원을 참조하십시오 [다운로드 AWS CLI](#).

를 사용하여 애플리케이션을 편집하려면 AWS CLI

에서 콘텐츠에 맞게 사용자 지정된 [UpdateApplication](#) 명령을 AWS CLI 사용합니다.

```
aws gameliftstreams update-application \
  --identifier a-9ZY8X7Wv6 \
  --description "MyGame v2" \
  --application-log-paths '[".\logs"]' \
  --application-log-output-uri "s3://amzn-s3-demo-bucket/mygame"
```

여기서 각 항목은 다음과 같습니다.

- `identifier`: 편집할 애플리케이션입니다.

이 값은 애플리케이션 [리소스를 고유하게 식별하는 Amazon 리소스 이름\(ARN\)](#) 또는 ID입니다.

ARN 예제: `arn:aws:gameliftstreams:us-west-2:111122223333:application/a-9ZY8X7Wv6`

ID 예제: `a-9ZY8X7Wv6`

- `description`:

사람이 읽을 수 있는 애플리케이션 레이블입니다. 이 값은 고유하지 않아도 됩니다. 모범 사례를 따라 애플리케이션에 의미 있는 설명, 이름 또는 레이블을 사용하세요. 이 필드는 언제든지 편집할 수 있습니다.

- `application-log-paths`:

저장하려는 로그가 들어 있는 애플리케이션 폴더 또는 파일의 경로입니다. 애플리케이션 기본 경로를 기준으로 각 로그 경로를 지정합니다. 이 기능을 사용하는 경우 모든 스트림 세션이 끝날 때 Amazon GameLift Streams는 사용자가 지정한 파일을 사용자가 지정한 이름의 Amazon S3 버킷에 복사합니다. 애플리케이션 폴더의 하위 폴더에서는 복사 작업이 재귀적으로 수행되지 않습니다.

로깅을 비활성화하려면 모든 애플리케이션 로그 경로를 제거하고 애플리케이션 로그 출력 대상을 지웁니다.

이 필드는 언제든지 편집할 수 있습니다.

- `application-log-output-uri`:

Amazon GameLift Streams가 애플리케이션 로그 파일을 복사할 Amazon S3 버킷의 URI입니다. 애플리케이션 로그 경로를 지정하는 경우 이 필드는 필수입니다.

로깅을 비활성화하려면 모든 애플리케이션 로그 경로를 제거하고 애플리케이션 로그 출력 대상을 지웁니다.

이 필드는 언제든지 편집할 수 있습니다.

사용자를 대신하여 로그 파일을 저장하도록 Amazon GameLift Streams에 S3 버킷에 저장할 수 있는 권한을 부여해야 합니다. Amazon GameLift Streams에서 로깅용 버킷을 생성하도록 하면 생성 시 권한 정책이 자동으로 적용됩니다. 사용자가 자체 버킷을 제공하는 경우 권한 정책을 직접 적용해야 합니다.

자세한 내용은 [애플리케이션 로그 버킷 권한 정책](#) 단원을 참조하십시오.

애플리케이션 삭제

더 이상 필요하지 않은 애플리케이션은 삭제합니다. 이 작업은 Amazon GameLift Streams에 저장된 애플리케이션 콘텐츠 파일을 포함하여 애플리케이션을 영구적으로 삭제합니다. 그러나 Amazon S3 버킷에 업로드한 원본 파일은 삭제되지 않습니다. Amazon GameLift Streams가 Amazon S3 버킷에 GameLift 액세스하는 유일한 시간인 애플리케이션을 생성한 후에는 언제든지 삭제할 수 있습니다.

다음 조건을 충족하는 애플리케이션만 삭제할 수 있습니다.

- 애플리케이션이 준비 또는 오류 상태입니다.
- 애플리케이션이 진행 중인 스트림 세션에서 스트리밍되지 않습니다. 클라이언트가 스트림 세션을 종료할 때까지 기다리거나 Amazon GameLift Streams API에서 [TerminateStreamSession](#)을 호출하여 스트림을 종료해야 합니다.

애플리케이션이 스트림 그룹에 연결된 경우 연결된 모든 스트림 그룹에서 연결을 해제해야 삭제할 수 있습니다. 콘솔에서 대화 상자가 이 프로세스를 안내합니다.

Console

Amazon GameLift Streams 콘솔을 사용하여 애플리케이션을 삭제하려면

1. 에 로그인 AWS Management Console 하고 [Amazon GameLift Streams 콘솔](#)을 엽니다.
2. 탐색 모음에서 애플리케이션을 선택하여 기존 애플리케이션 목록을 봅니다. 삭제할 애플리케이션을 선택합니다.
3. 애플리케이션 세부 정보 페이지에서 삭제를 선택합니다.
4. 삭제 대화 상자에서 삭제 작업을 확인합니다.

CLI

사전 조건

사용자 자격 증명과 선택한 AWS CLI 사용자를 구성해야 합니다 AWS 리전. 설정 지침은 단원을 참조하십시오 [다운로드 AWS CLI](#).

를 사용하여 애플리케이션을 삭제하려면 AWS CLI

에서 콘텐츠에 맞게 사용자 지정된 [DeleteApplication](#) 명령을 AWS CLI 사용합니다.

```
aws gameliftstreams delete-application \
  --identifier arn:aws:gameliftstreams:us-west-2:111122223333:application/a-9ZY8X7Wv6
```

여기서 각 항목은 다음과 같습니다.

- `identifier`: 삭제할 애플리케이션입니다.

이 값은 애플리케이션 [리소스를 고유하게 식별하는 Amazon 리소스 이름\(ARN\)](#) 또는 ID입니다.

ARN 예제: `arn:aws:gameliftstreams:us-west-2:111122223333:application/a-9ZY8X7Wv6`

ID 예제: `a-9ZY8X7Wv6`

Amazon GameLift Streams가 애플리케이션 삭제를 시작합니다. 이 시간 동안 애플리케이션은 `Deleting` 상태입니다. Amazon GameLift Streams가 애플리케이션을 삭제한 후에는 더 이상 검색할 수 없습니다.

애플리케이션 로그 버킷 권한 정책

자체 애플리케이션 로그 Amazon S3 버킷을 제공하는 경우 Amazon GameLift Streams가 로그 파일을 버킷에 저장할 수 있도록 버킷에 권한 정책을 적용해야 합니다. 다음 템플릿을 사용하여 Amazon S3의 권한을 업데이트합니다.

버킷 권한 정책 템플릿

다음 정책 코드를 복사하여 애플리케이션 로그에 사용할 버킷에 적용합니다. `amzn-s3-demo-bucket`을 기존 S3 버킷의 이름으로 교체해야 합니다.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "PutPolicy",
      "Effect": "Allow",
      "Principal": {
        "Service": [
          "gameliftstreams.amazonaws.com"
        ]
      },
      "Action": "s3:PutObject",
      "Resource": "arn:aws:s3:::amzn-s3-demo-bucket/*",
      "Condition": {
        "StringEquals": {
          "aws:SourceAccount": "your 12-digit account id"
        }
      }
    }
  ]
}
```

Note

Amazon GameLift Streams는 교차 계정 리소스 액세스를 허용하지 않습니다. Amazon S3 버킷은 애플리케이션 리소스와 동일한 AWS 계정이 소유해야 합니다. 이는 서비스에서 강력하게 적용되지만 서비스에 권한을 부여할 때 [혼동된 대리자 문제를](#) 방지하기 위해 항상 `aws:SourceAccount` 또는 `aws:SourceArn` 조건을 포함하는 것이 가장 좋습니다 AWS .

연결된 스트림 그룹

동일한 컴퓨팅 리소스 풀을 사용하여 여러 애플리케이션을 스트리밍하려는 경우 여러 애플리케이션을 동일한 스트림 그룹에 연결할 수 있습니다. 마찬가지로 다양한 컴퓨팅 리소스 세트를 사용하여 애플리케이션을 스트리밍하려는 경우 애플리케이션을 여러 스트림 그룹에 연결할 수 있습니다.

애플리케이션을 스트림 그룹에 연결하는 방법에 대한 자세한 내용은 [섹션을 참조하세요](#) [다중 애플리케이션 스트림 그룹 개요](#).

Amazon GameLift Streams 스트림 그룹을 사용하여 스트리밍 관리

Amazon GameLift Streams 애플리케이션을 설정한 후에는 컴퓨팅 리소스를 관리하고 배포하여 애플리케이션을 실행하고 스트리밍할 준비가 되었습니다. Amazon GameLift Streams 스트림 그룹은 이러한 컴퓨팅 리소스의 모음을 나타냅니다. 스트림 용량을 조정하여 지원할 최대 동시 스트림 수를 지정합니다.

Amazon GameLift Streams는 스트림 그룹을 생성하는 AWS 리전에 컴퓨팅 리소스를 할당합니다. 스트림 그룹에 원격 위치를 추가하고 위치당 용량을 관리할 수도 있습니다. 최종 사용자와 지리적으로 가까운 위치에서 스트림 세션을 호스팅하는 것이 가장 좋습니다. 이를 통해 지연 시간을 최소화하고 스트림 품질을 개선할 수 있습니다. 자세한 정보는 [AWS 리전 Amazon GameLift Streams에서 지원하는 및 스트리밍 위치](#) 섹션을 참조하세요.

스트림 그룹에서 스트림 그룹이 스트리밍할 수 있는 Amazon GameLift Streams 애플리케이션을 하나 이상 지정할 수 있습니다. 단일 애플리케이션은 여러 스트림 그룹에 있을 수 있으므로 동일한 애플리케이션을 스트리밍하도록 다양한 구성 또는 유형의 컴퓨팅 리소스를 설정할 수 있습니다. 예를 들어 애플리케이션을 스트리밍하기 위한 두 가지 그래픽 품질 옵션을 제공하기 위해 서로 다른 스트림 클래스 구성으로 두 개의 스트림 그룹을 설정하고 동일한 애플리케이션에 연결할 수 있습니다.

반대로 단일 스트림 그룹에는 스트림 그룹을 생성할 때 설정할 수 있는 기본 애플리케이션과 연결된 추가 애플리케이션 등 여러 애플리케이션이 있을 수 있습니다. 자세한 정보는 [다중 애플리케이션 스트림 그룹 개요](#) 섹션을 참조하세요.

스트림 그룹과 애플리케이션을 연결하는 방법은 사용 사례에 따라 다르지만 관계는 many-to-many일 수 있습니다.

스트림 그룹은 중요한 서비스 업데이트 및 수정 사항을 선택하기 위해 3~4주마다 다시 생성해야 합니다. 자세한 정보는 [스트림 그룹 수명 주기](#) 섹션을 참조하세요.

주제

- [스트림 용량 정보](#)
- [용량 및 서비스 할당량](#)
- [위치 정보](#)
- [스트림 그룹 생성](#)
- [일반 설정 편집](#)
- [용량 편집](#)
- [용량 축소 동작](#)
- [스트림 그룹에 위치 추가](#)
- [스트림 그룹의 위치 제거](#)
- [스트림 그룹 삭제](#)
- [연결된 애플리케이션](#)
- [스트림 그룹 수명 주기](#)
- [스트림 그룹 유지 관리](#)

스트림 용량 정보

스트림 그룹의 용량 또는 스트림 용량을 설정하여 최종 사용자에게 동시에 제공할 수 있는 스트림 수를 관리합니다. 스트림 용량은 스트림 그룹이 지원할 수 있는 동시 스트림 세션 수를 나타냅니다. 각 위치에서 구성됩니다.

- **상시 가동 용량:** 이 설정은 0이 아닌 경우 사용자에게 할당되고 서비스로 다시 릴리스되지 않는 최소 스트리밍 용량을 나타냅니다. 사용량이든 유휴 상태이든 관계없이 항상 이 기본 용량 수준에 대한 비용을 지불합니다.
- **최대 용량:** 서비스가 할당할 수 있는 최대 용량을 나타냅니다. 새로 생성된 스트림을 시작하는 데 몇 분 정도 걸릴 수 있습니다. 유휴 상태일 때 용량이 서비스로 다시 해제됩니다. 릴리스될 때까지 할당된 용량에 대해 비용을 지불합니다.
- **목표 유휴 용량:** 향후 활동을 예상하여 서비스가 미리 할당하고 보관하는 유휴 용량을 나타냅니다. 이렇게 하면 용량 할당 지연으로부터 사용자를 보호할 수 있습니다. 의도적인 유휴 상태로 유지되는 이 용량에 대해 비용을 지불합니다.

한 위치에 최대 용량이 100으로 설정된 스트림 그룹이 있는 경우 스트림 그룹에는 해당 위치에서 100명의 최종 사용자에게 동시에 스트리밍할 수 있는 충분한 리소스가 있습니다. 사용자 수요 변화에 맞게 언제든지 각 위치(현재 할당량까지)에서 스트림 용량을 늘리거나 줄일 수 있습니다.

Amazon GameLift Streams는 먼저 이미 할당된 유휴 용량을 사용하여 새 세션 요청을 이행하려고 시도합니다. 이로 인해 유휴 용량이 목표 유휴 용량 아래로 떨어지면 새 용량이 비동기적으로 할당됩니다. 유휴 용량을 사용할 수 없는 경우 새 용량이 온디맨드로 할당되는 동안 스트림 그룹의 최대 용량까지 요청이 일시 중지됩니다. 최대값에 도달했지만 여전히 사용 가능한 유휴 용량이 없는 경우 세션 요청은 기존 세션이 종료되고 여유 용량이 확보될 때까지 기다립니다.

세션이 종료되면 해당 용량이 유휴 상태로 표시됩니다. 목표 유휴 값보다 유휴 용량이 많으면 잠시 지연된 후 초과 용량이 할당 해제되고 서비스로 반환됩니다. 구성된 최소 용량(0일 수 있음) 미만으로 용량 수준이 떨어지면 서비스가 유휴 용량을 할당 해제하지 않습니다.

다중 테넌트 스트림 클래스(컴퓨팅 리소스당 1개 이상의 세션을 스트리밍할 수 있음)가 있는 스트림 그룹에서 스트림 용량을 지정하는 경우 용량은 테넌시의 배수여야 합니다. 예를 들어 gen6n_high 스트림 클래스의 멀티테넌시는 2입니다. 즉, 스트림 그룹에 할당된 각 컴퓨팅 리소스는 2개의 클라이언트로 스트리밍할 수 있습니다. 따라서 요청하는 용량은 2의 배수여야 합니다.

용량 조정은 스트림 그룹의 총 비용에 반영됩니다. Amazon GameLift Streams 비용을 관리하기 위해 결제 알림을 설정해야 합니다. 자세한 내용은 [결제 알림을 생성하여 사용량 모니터링](#) 항목을 참조하세요.

스트림 그룹 용량을 변경하려면 스트림 그룹 설정을 편집하고 용량 설정에 새 값을 입력합니다. 상시 작동 용량을 변경하면 Amazon GameLift Streams는 새 리소스를 프로비저닝하거나 기존 리소스를 종료하여 새 값과 일치하도록 할당된 리소스를 조정합니다. 리소스를 즉시 사용할 수 없는 경우 상시 가동 용량을 늘리는 데 몇 분 이상 걸릴 수 있습니다. 상시 가동 용량을 줄이려면 할당된 리소스의 프로비저닝을 해제하는 데 몇 분 정도 걸립니다.

예: 스트림 용량 구성

다음 예제에서는 다양한 사용 사례에 대한 일반적인 스트림 용량 구성을 보여줍니다.

1. 비용 중심의 개발 단계: 비용을 절감하려는 개발자입니다. Minimum (always-on) capacity = 0, Maximum capacity = 10, Target Idle (pre-warmed) capacity = 1로 설정합니다. 이렇게 하면 빠른 시작을 위해 하나 이상의 세션을 사용할 수 있습니다.
2. 수요가 고정된 계획된 이벤트: 수요가 알려진 계획된 이벤트에 대해 빠른 세션 시작을 원합니다. Minimum (always-on) capacity = 200, Maximum capacity = 200, Target Idle (pre-warmed) capacity = 0으로 설정합니다. 200개 용량만 지불하면 됩니다. 수요가 알려져 있기 때문에 규모 조정 지연이 발생하지 않습니다.
3. 버스트 용량이 있는 대규모 이벤트: 피크 시간에 분당 100개의 새 세션이 있는 1,000명의 사용자를 계획하고 있습니다. Minimum = 0, Maximum = 1,000, Target Idle = 100으로 설정합니다. 이렇

게 하면 유휴 시 비용을 절약할 수 있습니다. 이렇게 하면 빠른 시작을 위해 최소 100개의 세션을 사용할 수 있습니다.

Note

OnDemandCapacity 입력 파라미터는 더 이상 사용되지 않습니다. MaximumCapacity 대신 API를 통해 용량을 구성할 때를 사용합니다.

용량 및 서비스 할당량

Amazon GameLift Streams 사용에는 계정에서 스트리밍을 위해 구성할 수 있는 총 GPU(컴퓨팅 리소스) 수를 제한하는 서비스 할당량이 적용됩니다. 기본 할당량 및 할당량 사용률은 GameLift Streams용 Service Quota 콘솔에서 확인할 수 있습니다. 이러한 할당량이 스트림 용량과 상호 작용하는 방식을 이해하면 스트리밍 인프라를 계획하고 용량 제한을 방지하는 데 도움이 됩니다.

보다 구체적으로, GPU 서비스 할당량은 계정의 모든 스트림 그룹에 걸쳐 위치당 요청할 수 있는 특정 스트림 클래스 패밀리 최대 GPU 수를 지정합니다. 예를 들어 계정에서의 gen6n GPU 한도가 5개인 경우 모든 스트림 그룹에 us-west-2 대해 us-west-2의 총 스트림 용량을 제공하는 데 필요한 gen6n GPU 합계는 5보다 작거나 같아야 합니다. 여기에는 상시 가동 용량과 온디맨드 용량 모두에 대한 GPU 포함됩니다.

Amazon GameLift Streams는 할당된 GPU 합계를 기준으로 서비스 할당량을 측정합니다. 일부 스트림 클래스(예: gen6n_high 또는 gen6n_small)는 동시 세션 간에 GPU를 공유한다는 점을 기억해야 합니다. gen6n_ultra 및와 같은 다른 스트림 클래스는 동시 세션당 하나의 전체 GPU를 gen6n_ultra_win2022 사용합니다. 따라서 10GPUs를 gen6n_ultra 스트림 그룹의 MaximumCapacity 10 또는 gen6n_medium 스트림 클래스의 MaximumCapacity 40으로 할당할 수 있습니다.

예: 할당량이 용량에 미치는 영향

다음 예제에서는 서비스 할당량이 여러 스트림 그룹 및 위치에서 스트림 용량과 상호 작용하는 방법을 보여줍니다. 이 예제에서는 계정에 위치당 10개의 gen6n GPU.

1. 단일 테넌트 스트림 그룹 생성:에서 총 용량(항상 켜짐 및 온디맨드)이 5개인 스트림 클래스를 사용하여 gen6n_ultra 스트림 그룹을 생성합니다us-east-2. 이 스트림 클래스에는 1:1 테넌시(GPU 당 스트림 1개)가 있으므로 총 5개 용량에 대해 GPU 5개가 필요합니다. 이렇게 하면에 남은 GPU가 5개 남습니다us-east-2.

- 다중 테넌트 스트림 그룹 생성:에서 총 용량이 6인 스트림 클래스를 사용하여 다른 gen6n_high 스트림 그룹을 생성합니다us-east-2. 이 스트림 클래스에는 1:2 테넌시(GPU당 스트림 2개)가 있으므로 총 6개 용량에 대해 GPUs 3개만 필요합니다. 이렇게 하면에 남은 GPUs가 2개 남습니다us-east-2.
- 다른 위치에 용량 추가: 이러한 스트림 그룹을 생성한 후에 2GPUs가 us-east-2있지만 us-west-2 또는와 같은 다른 위치에서는 여전히 10GPUs를 사용할 수 있습니다eu-west-1. 이러한 위치를 이전에 생성한 스트림 그룹 중 하나에 추가하거나 이러한 위치가 있는 새 스트림 그룹을 생성할 수 있습니다.

이 예제는 할당량이 위치별 및 모든 스트림 그룹에 적용되므로 서비스 제한을 유지하면서 여러 지리적 리전에 스트리밍 용량을 분산할 수 있음을 보여줍니다.

Note

GameLift 스트림을 AWS 서비스로 선택하여 Service Quotas 콘솔에서 해당 할당량 사용률을 포함한 적용된 계정 수준 또는 기본 할당량을 볼 수 있습니다. 자세한 내용은 [Amazon GameLift Streams 서비스 할당량](#) 단원을 참조하십시오.

위치 정보

위치는 Amazon GameLift Streams가 컴퓨팅 리소스를 할당하여 애플리케이션을 호스팅하고 사용자에게 스트리밍하는 위치입니다. 지연 시간을 줄이고 품질을 높이려면 사용자에게 더 가까운 위치를 선택해야 합니다. 기본적으로 기본 위치라고 하는 스트림 그룹을 AWS 리전 생성한에서 스트리밍할 수 있습니다. 또한 스트림 그룹은 적용 범위를 확장하여 원격 위치라고 하는 지원되는 다른 위치에서 스트리밍할 수 있습니다.

지원되는 위치의 전체 목록은 섹션을 참조하세요 [AWS 리전 및 스트리밍 위치](#).

다중 위치 스트림 그룹

기본 위치(스트림 그룹을 AWS 리전 생성한 위치) 외에도 여러 위치에서 애플리케이션 및 스트림 세션을 호스팅하도록 구성된 스트림 그룹입니다. 각 위치의 용량을 관리합니다.

스트림 그룹 생성

Console

Amazon GameLift Streams 콘솔에서 스트림 그룹을 생성하려면

1. 에 로그인 AWS Management Console 하고 [Amazon GameLift Streams 콘솔](#)을 엽니다. 스트림 그룹을 생성할 AWS 리전을 선택합니다. 이 리전은 스트림 그룹으로 스트리밍하려는 애플리케이션의 리전과 동일해야 합니다. 자세한 내용은 시작 안내서의 [리전 선택](#)을 참조하세요.
AWS Management Console
2. 생성 워크플로를 열려면 탐색 창에서 스트림 그룹을 선택한 다음 스트림 그룹 생성을 선택합니다.
3. 스트림 그룹 정의에 다음을 입력합니다.

a. 설명

사람이 읽을 수 있는 스트림 그룹 레이블입니다. 이 값은 고유하지 않아도 됩니다. 모범 사례를 따라 스트림 그룹에 의미 있는 설명, 이름 또는 레이블을 사용하세요. 이 필드는 언제든지 편집할 수 있습니다.

b. Tags

태그는 AWS 리소스를 구성하는 데 도움이 될 수 있는 레이블입니다. 자세한 내용은 [AWS 리소스 태그 지정](#)을 참조하세요.

4. 스트림 클래스 선택에서 스트림 그룹의 스트림 클래스를 선택합니다.

• 스트림 클래스 옵션

애플리케이션을 실행하고 스트리밍할 컴퓨팅 리소스의 유형입니다. 이 선택은 스트리밍 경험의 품질과 비용에 영향을 미칩니다. 스트림 그룹당 하나의 스트림 클래스만 지정할 수 있습니다. 애플리케이션에 가장 적합한 클래스를 선택합니다.

스트림 클래스	설명
gen6n_pro _win2022	(NVIDIA, pro) 최대 리소스가 필요하며 3D 장면 복잡성이 매우 높음 수준인 애플리케이션을 지원합니다. Microsoft Windows Server 2022 Base에서 애플리케이션을 실행하고 DirectX 12를 지원합니다. Unreal Engine 5.6 버전 이하, 32비트 및 64비트 애

스트림 클래스	설명
	<p>플리케이션, 치트 방지 기술과 호환됩니다. NVIDIA L4 Tensor Core GPU를 사용합니다.</p> <p>애플리케이션당 리소스: vCPU: 16 RAM: 64GB VRAM: 24GB</p> <p>테넌시: 최대 1개의 동시 스트림 세션을 지원합니다.</p>
gen6n_pro	<p>(NVIDIA, pro) 최대 리소스가 필요하며 3D 장면 복잡성이 매우 높음 수준인 애플리케이션을 지원합니다. NVIDIA L4 Tensor Core GPU를 사용합니다.</p> <p>애플리케이션당 리소스: vCPU: 16 RAM: 64GB VRAM: 24GB</p> <p>테넌시: 최대 1개의 동시 스트림 세션을 지원합니다.</p>
gen6n_ultra_win2022	<p>(NVIDIA, ultra) 3D 장면 복잡도가 높음 수준인 애플리케이션을 지원합니다. Microsoft Windows Server 2022 Base에서 애플리케이션을 실행하고 DirectX 12를 지원합니다. Unreal Engine 5.6 버전 이하, 32비트 및 64비트 애플리케이션, 치트 방지 기술과 호환됩니다. NVIDIA L4 Tensor Core GPU를 사용합니다.</p> <p>애플리케이션당 리소스: vCPU: 8 RAM: 32GB VRAM: 24GB</p> <p>테넌시: 최대 1개의 동시 스트림 세션을 지원합니다.</p>
gen6n_ultra	<p>(NVIDIA, ultra) 3D 장면 복잡도가 높음 수준인 애플리케이션을 지원합니다. NVIDIA L4 Tensor Core GPU를 사용합니다.</p> <p>애플리케이션당 리소스: vCPU: 8 RAM: 32GB VRAM: 24GB</p> <p>테넌시: 최대 1개의 동시 스트림 세션을 지원합니다.</p>
gen6n_high	<p>(NVIDIA, high) 3D 장면 복잡도가 보통-높음 수준인 애플리케이션을 지원합니다. NVIDIA L4 Tensor Core GPU를 사용합니다.</p> <p>애플리케이션당 리소스: vCPU: 4 RAM: 16GB VRAM: 12GB</p> <p>테넌시: 최대 2개의 동시 스트림 세션을 지원합니다.</p>

스트림 클래스	설명
gen6n_medium	<p>(NVIDIA, medium) 3D 장면 복잡도가 보통 수준인 애플리케이션을 지원합니다. NVIDIA L4 Tensor Core GPU를 사용합니다.</p> <p>애플리케이션당 리소스: vCPU: 2 RAM: 8GB VRAM: 6GB</p> <p>테넌시: 최대 4개의 동시 스트림 세션을 지원합니다.</p>
gen6n_small	<p>(NVIDIA, small) 3D 장면 복잡도가 경량 수준이며 CPU 사용량이 적은 애플리케이션을 지원합니다. NVIDIA L4 Tensor Core GPU를 사용합니다.</p> <p>애플리케이션당 리소스: vCPU: 1 RAM: 4GB VRAM: 2GB</p> <p>테넌시: 최대 12개의 동시 스트림 세션을 지원합니다.</p>
gen6n_medium_win2022	<p>(NVIDIA, 중간) 3D 장면 복잡성이 낮은 애플리케이션을 지원합니다. Microsoft Windows Server 2022 Base에서 애플리케이션을 실행합니다. NVIDIA L4 Tensor Core GPU를 사용합니다.</p> <p>애플리케이션당 리소스: vCPU: 8 RAM: 32GB VRAM: 6GB</p> <p>테넌시: 최대 1개의 동시 스트림 세션을 지원합니다.</p>
gen6n_small_win2022	<p>(NVIDIA, 스몰) 3D 장면 복잡성이 낮은 애플리케이션을 지원합니다. Microsoft Windows Server 2022 Base에서 애플리케이션을 실행합니다. NVIDIA L4 Tensor Core GPU를 사용합니다.</p> <p>애플리케이션당 리소스: vCPU: 2 RAM: 8GB VRAM: 3GB.</p> <p>테넌시: 최대 1개의 동시 스트림 세션을 지원합니다.</p>

스트림 클래스	설명
gen5n_win2022	<p>(NVIDIA, ultra) 3D 장면 복잡도가 매우 높음 수준인 애플리케이션을 지원합니다. Microsoft Windows Server 2022 Base에서 애플리케이션을 실행하고 DirectX 12 및 DirectX 11을 지원합니다. Unreal Engine 버전 5.6 이하, 32비트 및 64비트 애플리케이션, 치트 방지 기술을 지원합니다. NVIDIA A10G Tensor Core GPU를 사용합니다.</p> <p>애플리케이션당 리소스: vCPU: 8 RAM: 32GB VRAM: 24GB</p> <p>테넌시: 동시 스트림 세션 1개를 지원합니다.</p>
gen5n_high	<p>(NVIDIA, high) 3D 장면 복잡도가 보통-높음 수준인 애플리케이션을 지원합니다. NVIDIA A10G Tensor Core GPU를 사용합니다.</p> <p>애플리케이션당 리소스: vCPU: 4 RAM: 16GB VRAM: 12GB</p> <p>테넌시: 최대 2개의 동시 스트림 세션을 지원합니다.</p>
gen5n_ultra	<p>(NVIDIA, ultra) 3D 장면 복잡도가 매우 높음 수준인 애플리케이션을 지원합니다. NVIDIA A10G Tensor Core GPU를 사용합니다.</p> <p>애플리케이션당 리소스: vCPU: 8 RAM: 32GB VRAM: 24GB</p> <p>테넌시: 동시 스트림 세션 1개를 지원합니다.</p>
gen4n_win2022	<p>(NVIDIA, ultra) 3D 장면 복잡도가 높음 수준인 애플리케이션을 지원합니다. Microsoft Windows Server 2022 Base에서 애플리케이션을 실행하고 DirectX 12 및 DirectX 11을 지원합니다. Unreal Engine 버전 5.6 이하, 32비트 및 64비트 애플리케이션, 치트 방지 기술을 지원합니다. NVIDIA T4 Tensor Core GPU를 사용합니다.</p> <p>애플리케이션당 리소스: vCPU: 8 RAM: 32GB VRAM: 16GB</p> <p>테넌시: 동시 스트림 세션 1개를 지원합니다.</p>

스트림 클래스	설명
gen4n_high	(NVIDIA, high) 3D 장면 복잡도가 보통-높음 수준인 애플리케이션을 지원합니다. NVIDIA T4 Tensor Core GPU를 사용합니다. 애플리케이션당 리소스: vCPU: 4 RAM: 16GB VRAM: 8GB 테넌시: 최대 2개의 동시 스트림 세션을 지원합니다.
gen4n_ultra	(NVIDIA, ultra) 3D 장면 복잡도가 높음 수준인 애플리케이션을 지원합니다. NVIDIA T4 Tensor Core GPU를 사용합니다. 애플리케이션당 리소스: vCPU: 8 RAM: 32GB VRAM: 16GB 테넌시: 동시 스트림 세션 1개를 지원합니다.

계속하려면 다음을 선택합니다.

- 애플리케이션 연결에서 스트리밍할 애플리케이션을 선택하거나 "애플리케이션 없음"을 선택하여 나중에 하나를 선택합니다. 스트림 그룹을 생성한 후 편집하여 애플리케이션을 추가하거나 제거할 수 있습니다. Ready 상태이고 선택한 스트림 클래스와 호환되는 런타임이 있는 애플리케이션만 연결할 수 있습니다. 기본적으로 이러한 애플리케이션만 테이블에 표시됩니다. Ready 상태의 모든 애플리케이션을 보려면 드롭다운 목록에서 All runtimes를 선택합니다.

Note

애플리케이션이 나열되지 않으면 현재 AWS 리전 설정을 확인합니다. 애플리케이션을 동일한 리전에 있는 스트림 그룹에만 연결할 수 있습니다.

계속하려면 다음을 선택합니다.

- 스트림 설정 구성의 위치 및 용량에서 스트림 그룹에 애플리케이션을 스트리밍할 용량이 있는 위치를 하나 이상 선택합니다. 기본적으로 기본 위치라고 하는 스트림 그룹을 생성하는 리전은 스트림 그룹에 이미 추가되었으며 제거할 수 없습니다. 추가하려는 각 위치 옆의 확인란을 선택하여 위치를 추가할 수 있습니다. 지연 시간을 줄이고 스트리밍 품질을 높이려면 사용자에게 더 가까운 위치를 선택해야 합니다.

각 위치에 대해 스트리밍 용량을 지정할 수 있습니다. 스트림 용량은 한 번에 활성화할 수 있는 동시 스트림 수를 나타냅니다. 각 스트림 그룹의 위치별로 스트림 용량을 설정합니다.

- **상시 가동 용량:** 이 설정은 0이 아닌 경우 사용자에게 할당되고 서비스로 다시 릴리스되지 않는 최소 스트리밍 용량을 나타냅니다. 사용량이든 유휴 상태이든 관계없이 항상 이 기본 용량 수준에 대한 비용을 지불합니다.
- **최대 용량:** 서비스가 할당할 수 있는 최대 용량을 나타냅니다. 새로 생성된 스트림을 시작하는 데 몇 분 정도 걸릴 수 있습니다. 유휴 상태일 때 용량이 서비스로 다시 해제됩니다. 릴리스될 때까지 할당된 용량에 대해 비용을 지불합니다.
- **목표 유휴 용량:** 향후 활동을 예상하여 서비스가 미리 할당하고 보관하는 유휴 용량을 나타냅니다. 이렇게 하면 용량 할당 지연으로부터 사용자를 보호할 수 있습니다. 의도적인 유휴 상태로 유지되는 이 용량에 대해 비용을 지불합니다.

어느 용량이든 조정하여 위치에 대한 사용자 수요의 변화를 충족하기 위해 언제든지 총 스트림 용량을 늘리거나 줄일 수 있습니다. Amazon GameLift Streams는 사용 가능한 경우 상시 작동 용량 풀의 유휴 사전 할당된 리소스를 사용하여 스트리밍 요청을 이행합니다. 모든 상시 작동 용량이 사용 중인 경우 Amazon GameLift Streams는 온디맨드 용량에 지정된 최대 수까지 추가 컴퓨팅 리소스를 프로비저닝합니다. 할당된 용량이 조정되면 변경 사항이 스트림 그룹의 총 비용에 반영됩니다.

연결된 애플리케이션은 활성화된 각 위치에 자동으로 복제됩니다. 원격 위치에서 스트림을 호스팅하려면 애플리케이션이 원격 위치에서 복제를 완료해야 합니다. 복제 상태를 확인하려면 스트림 그룹을 생성한 후 열고 연결된 애플리케이션 테이블의 복제 상태 열을 참조하세요. 현재 상태를 클릭하여 추가된 각 위치의 복제 상태를 확인합니다.

Note

애플리케이션 데이터는 이 스트림 그룹의 기본 위치를 포함하여 활성화된 모든 위치에 저장됩니다. 스트림 세션 데이터는 기본 위치와 스트리밍이 발생한 위치 모두에 저장됩니다.

7. 스트림 그룹 검토 및 생성에서 스트림 그룹 구성을 확인하고 필요에 따라 변경합니다. 모든 것이 올바르게 스트림 그룹 생성을 선택합니다.

CLI

사전 조건

사용자 자격 증명과 선택한 AWS CLI 사용자를 구성해야 합니다. AWS 리전. 설정 지침은 단원을 참조하십시오. [다운로드 AWS CLI](#).

를 사용하여 스트림 그룹을 생성하려면 AWS CLI

에서 콘텐츠에 맞게 사용자 지정된 [CreateStreamGroup](#) 명령을 AWS CLI 사용합니다.

```
aws gameliftstreams create-stream-group \
  --description "Test_gen4_high" \
  --default-application-identifier arn:aws:gameliftstreams:us-west-2:111122223333:application/a-9ZY8X7Wv6 \
  --stream-class gen4n_high \
  --location-configurations '[{"LocationName": "us-east-1", "AlwaysOnCapacity": 2, "MaximumCapacity": 6, "TargetIdleCapacity": 1}]'
```

where

description:

사람이 읽을 수 있는 스트림 그룹 레이블입니다. 이 값은 고유하지 않아도 됩니다. 모범 사례를 따라 스트림 그룹에 의미 있는 설명, 이름 또는 레이블을 사용하세요. 이 필드는 언제든지 편집할 수 있습니다.

default-application-identifier

[Amazon GameLift Streams 애플리케이션 리소스에 할당된 Amazon 리소스 이름\(ARN\)](#) 값 또는 ID입니다. GameLift 애플리케이션은 READY 상태여야 합니다.

ARN 예제: arn:aws:gameliftstreams:us-west-2:111122223333:application/a-9ZY8X7Wv6

ID 예제: a-9ZY8X7Wv6

stream-class

스트림 클래스 옵션

애플리케이션을 실행하고 스트리밍할 컴퓨팅 리소스의 유형입니다. 이 선택은 스트리밍 경험의 품질과 비용에 영향을 미칩니다. 스트림 그룹당 하나의 스트림 클래스만 지정할 수 있습니다. 애플리케이션에 가장 적합한 클래스를 선택합니다.

스트림 클래스	설명
gen6n_pro_win2022	<p>(NVIDIA, pro) 최대 리소스가 필요하며 3D 장면 복잡성이 매우 높음 수준인 애플리케이션을 지원합니다. Microsoft Windows Server 2022 Base에서 애플리케이션을 실행하고 DirectX 12를 지원합니다. Unreal Engine 5.6 버전 이하, 32비트 및 64비트 애플리케이션, 치트 방지 기술과 호환됩니다. NVIDIA L4 Tensor Core GPU를 사용합니다.</p> <p>애플리케이션당 리소스: vCPU: 16 RAM: 64GB VRAM: 24GB</p> <p>테넌시: 최대 1개의 동시 스트림 세션을 지원합니다.</p>
gen6n_pro	<p>(NVIDIA, pro) 최대 리소스가 필요하며 3D 장면 복잡성이 매우 높음 수준인 애플리케이션을 지원합니다. NVIDIA L4 Tensor Core GPU를 사용합니다.</p> <p>애플리케이션당 리소스: vCPU: 16 RAM: 64GB VRAM: 24GB</p> <p>테넌시: 최대 1개의 동시 스트림 세션을 지원합니다.</p>
gen6n_ultra_win2022	<p>(NVIDIA, ultra) 3D 장면 복잡도가 높음 수준인 애플리케이션을 지원합니다. Microsoft Windows Server 2022 Base에서 애플리케이션을 실행하고 DirectX 12를 지원합니다. Unreal Engine 5.6 버전 이하, 32비트 및 64비트 애플리케이션, 치트 방지 기술과 호환됩니다. NVIDIA L4 Tensor Core GPU를 사용합니다.</p> <p>애플리케이션당 리소스: vCPU: 8 RAM: 32GB VRAM: 24GB</p> <p>테넌시: 최대 1개의 동시 스트림 세션을 지원합니다.</p>
gen6n_ultra	<p>(NVIDIA, ultra) 3D 장면 복잡도가 높음 수준인 애플리케이션을 지원합니다. NVIDIA L4 Tensor Core GPU를 사용합니다.</p> <p>애플리케이션당 리소스: vCPU: 8 RAM: 32GB VRAM: 24GB</p> <p>테넌시: 최대 1개의 동시 스트림 세션을 지원합니다.</p>
gen6n_high	<p>(NVIDIA, high) 3D 장면 복잡도가 보통-높음 수준인 애플리케이션을 지원합니다. NVIDIA L4 Tensor Core GPU를 사용합니다.</p>

스트림 클래스	설명
	<p>애플리케이션당 리소스: vCPU: 4 RAM: 16GB VRAM: 12GB</p> <p>테넌시: 최대 2개의 동시 스트림 세션을 지원합니다.</p>
gen6n_medium_win2022	<p>(NVIDIA, 중간) 3D 장면 복잡성이 낮은 애플리케이션을 지원합니다. Microsoft Windows Server 2022 Base에서 애플리케이션을 실행합니다. NVIDIA L4 Tensor Core GPU를 사용합니다.</p> <p>애플리케이션당 리소스: vCPU: 8 RAM: 32GB VRAM: 6GB</p> <p>테넌시: 최대 1개의 동시 스트림 세션을 지원합니다.</p>
gen6n_medium	<p>(NVIDIA, medium) 3D 장면 복잡도가 보통 수준인 애플리케이션을 지원합니다. NVIDIA L4 Tensor Core GPU를 사용합니다.</p> <p>애플리케이션당 리소스: vCPU: 2 RAM: 8GB VRAM: 6GB</p> <p>테넌시: 최대 4개의 동시 스트림 세션을 지원합니다.</p>
gen6n_small	<p>(NVIDIA, small) 3D 장면 복잡도가 경량 수준이며 CPU 사용량이 적은 애플리케이션을 지원합니다. NVIDIA L4 Tensor Core GPU를 사용합니다.</p> <p>애플리케이션당 리소스: vCPU: 1 RAM: 4GB VRAM: 2GB</p> <p>테넌시: 최대 12개의 동시 스트림 세션을 지원합니다.</p>
gen6n_small_win2022	<p>(NVIDIA, 스몰) 3D 장면 복잡성이 낮은 애플리케이션을 지원합니다. Microsoft Windows Server 2022 Base에서 애플리케이션을 실행합니다. NVIDIA L4 Tensor Core GPU를 사용합니다.</p> <p>애플리케이션당 리소스: vCPU: 2 RAM: 8GB VRAM: 3GB.</p> <p>테넌시: 최대 1개의 동시 스트림 세션을 지원합니다.</p>

스트림 클래스	설명
gen5n_win2022	<p>(NVIDIA, ultra) 3D 장면 복잡도가 매우 높음 수준인 애플리케이션을 지원합니다. Microsoft Windows Server 2022 Base에서 애플리케이션을 실행하고 DirectX 12 및 DirectX 11을 지원합니다. Unreal Engine 버전 5.6 이하, 32비트 및 64비트 애플리케이션, 치트 방지 기술을 지원합니다. NVIDIA A10G Tensor Core GPU를 사용합니다.</p> <p>애플리케이션당 리소스: vCPU: 8 RAM: 32GB VRAM: 24GB</p> <p>테넌시: 동시 스트림 세션 1개를 지원합니다.</p>
gen5n_ultra	<p>(NVIDIA, ultra) 3D 장면 복잡도가 매우 높음 수준인 애플리케이션을 지원합니다. NVIDIA A10G Tensor Core GPU를 사용합니다.</p> <p>애플리케이션당 리소스: vCPU: 8 RAM: 32GB VRAM: 24GB</p> <p>테넌시: 동시 스트림 세션 1개를 지원합니다.</p>
gen5n_high	<p>(NVIDIA, high) 3D 장면 복잡도가 보통-높음 수준인 애플리케이션을 지원합니다. NVIDIA A10G Tensor Core GPU를 사용합니다.</p> <p>애플리케이션당 리소스: vCPU: 4 RAM: 16GB VRAM: 12GB</p> <p>테넌시: 최대 2개의 동시 스트림 세션을 지원합니다.</p>
gen4n_win2022	<p>(NVIDIA, ultra) 3D 장면 복잡도가 높음 수준인 애플리케이션을 지원합니다. Microsoft Windows Server 2022 Base에서 애플리케이션을 실행하고 DirectX 12 및 DirectX 11을 지원합니다. Unreal Engine 버전 5.6 이하, 32비트 및 64비트 애플리케이션, 치트 방지 기술을 지원합니다. NVIDIA T4 Tensor Core GPU를 사용합니다.</p> <p>애플리케이션당 리소스: vCPU: 8 RAM: 32GB VRAM: 16GB</p> <p>테넌시: 동시 스트림 세션 1개를 지원합니다.</p>

스트림 클래스	설명
gen4n_ultra	(NVIDIA, ultra) 3D 장면 복잡도가 높음 수준인 애플리케이션을 지원합니다. NVIDIA T4 Tensor Core GPU를 사용합니다. 애플리케이션당 리소스: vCPU: 8 RAM: 32GB VRAM: 16GB 테넌시: 동시 스트림 세션 1개를 지원합니다.
gen4n_high	(NVIDIA, high) 3D 장면 복잡도가 보통-높음 수준인 애플리케이션을 지원합니다. NVIDIA T4 Tensor Core GPU를 사용합니다. 애플리케이션당 리소스: vCPU: 4 RAM: 16GB VRAM: 8GB 테넌시: 최대 2개의 동시 스트림 세션을 지원합니다.

location-configurations

이 스트림 그룹에 추가할 위치 집합 및 용량입니다. 기본적으로 용량이 지정되지 않은 경우 Amazon GameLift Streams는 스트림 그룹이 생성된 위치에서 하나의 스트림을 시작하기에 충분한 상시 가동 스트림 용량만 할당합니다. Amazon GameLift Streams가 지원하는 위치의 전체 목록은 [섹션을 참조하세요](#) [AWS 리전 Amazon GameLift Streams에서 지원하는 및 스트리밍 위치](#).

용량 값은 스트림 그룹 스트림 클래스의 테넌시 값의 정수 배수여야 합니다.

요청이 성공하면 Amazon GameLift Streams는 다음과 유사한 응답을 반환합니다.

```
{
  "Arn": "arn:aws:gameliftstreams:us-west-2:111122223333:streamgroup/
sg-1AB2C3De4",
  "Description": "Test_gen4_high",
  "DefaultApplication": {
    "Id": "a-9ZY8X7Wv6"
  },
  "StreamClass": "gen4n_high",
  "Id": "sg-1AB2C3De4",
  "Status": "ACTIVATING",
  "LastUpdatedAt": "2024-11-18T15:49:01.482000-08:00",
  "CreatedAt": "2024-11-18T15:49:01.482000-08:00"
}
```

Amazon GameLift Streams는 할당되지 않은 컴퓨팅 리소스를 검색하고 새 스트림 그룹에 프로비저닝하기 시작합니다. 이 작업은 몇 분 정도 걸릴 수 있습니다. 이 시간 동안 새 스트림 그룹은 활성화 중 상태입니다.

상태가 활성화일 때 스트림 그룹의 용량을 조정할 수 있습니다. 자세한 정보는 [용량 편집](#) 섹션을 참조하세요.

스트림 그룹이 활성화 상태이면 스트리밍을 위해 리소스를 배포할 준비가 된 것입니다. 스트리밍을 시작하려면 단원을 참조하십시오 [Amazon GameLift Streams로 스트림 세션 시작](#).

일반 설정 편집

Amazon GameLift Streams는 콘솔에서 상태, 스트림 그룹 ID, 설명, 스트림 그룹 ARN, 스트림 클래스 등의 설정을 그룹화합니다. 이 중에서 새 스트림 그룹을 생성하지 않고 업데이트할 수 있는 유일한 항목은 설명입니다.

Console

1. 에 로그인 AWS Management Console 하고 [Amazon GameLift Streams 콘솔](#)을 엽니다.
2. 탐색 모음에서 스트림 그룹을 선택하여 기존 스트림 그룹 목록을 봅니다. 편집할 스트림 그룹을 선택합니다.
3. 스트림 그룹 세부 정보 페이지에서 설정 편집을 선택합니다.
4. 설명을 업데이트하려면 새 값을 입력합니다.

CLI

사전 조건

사용자 자격 증명과 선택한 AWS CLI 사용자를 구성해야 합니다. AWS 리전. 설정 지침은 단원을 참조하십시오. [다운로드 AWS CLI](#).

를 사용하여 스트림 그룹의 설명을 편집하려면 AWS CLI

에서 콘텐츠에 맞게 사용자 지정된 [UpdateStreamGroup](#) 명령을 AWS CLI 사용합니다.

```
aws gameliftstreams update-stream-group \
  --identifier arn:aws:gameliftstreams:us-west-2:111122223333:streamgroup/
  sg-1AB2C3De4 \
  --description "MyGame - Ultra"
```

where

identifier

스트림 그룹 [리소스를 고유하게 식별하는 Amazon 리소스 이름\(ARN\)](#) 또는 ID입니다.

ARN 예제: arn:aws:gameliftstreams:us-west-2:111122223333:streamgroup/
sg-1AB2C3De4

ID 예제: sg-1AB2C3De4

description

사람이 읽을 수 있는 스트림 그룹 레이블입니다. 이 값은 고유하지 않아도 됩니다. 모범 사례를 따라 스트림 그룹에 의미 있는 설명, 이름 또는 레이블을 사용하세요. 이 필드는 언제든지 편집할 수 있습니다.

용량 편집

각 위치의 용량을 조정하여 스트림 그룹을 확장합니다.

AWS 계정위치별 스트림 그룹 용량 할당량과 이러한 할당량을 늘리는 방법에 대한 [Amazon GameLift Streams 서비스 할당량](#) 자세한 내용은 섹션을 참조하세요.

Console

1. 에 로그인 AWS Management Console 하고 [Amazon GameLift Streams 콘솔](#)을 엽니다.

2. 탐색 모음에서 스트림 그룹을 선택하여 기존 스트림 그룹 목록을 봅니다. 편집할 스트림 그룹을 선택합니다.
3. 스트림 그룹 세부 정보 페이지에서 구성 편집을 선택합니다.
4. 각 위치에 대해 테이블의 관련 셀에 새 상시 작동 용량, 최대 용량 및 대상 유휴 용량 값을 입력합니다. 용량 값은 스트림 그룹 스트림 클래스의 테넌시 값의 정수 배수여야 합니다.

상시 작동 용량 값을 0으로 설정하면 스트림 그룹이 스트림에 호스트를 할당하지 않습니다.

CLI

사전 조건

사용자 자격 증명과 선택한 AWS CLI 사용 여부를 구성해야 합니다 AWS 리전. 설정 지침은 단원을 참조하십시오 [다운로드 AWS CLI](#).

를 사용하여 스트림 용량을 편집하려면 AWS CLI

에서 콘텐츠에 맞게 사용자 지정된 [UpdateStreamGroup](#) 명령을 AWS CLI 사용합니다.

```
aws gameliftstreams update-stream-group \
  --identifier arn:aws:gameliftstreams:us-west-2:111122223333:streamgroup/sg-1AB2C3De4 \
  --location-configurations '[{"LocationName": "us-east-1", "AlwaysOnCapacity": 4, "MaximumCapacity": 8}, \
    {"LocationName": "ap-northeast-1", "AlwaysOnCapacity": 0, "MaximumCapacity": 2, "TargetIdleCapacity": 1}]'
```

where

identifier

스트림 그룹 [리소스를 고유하게 식별하는 Amazon 리소스 이름\(ARN\)](#) 또는 ID입니다.

ARN 예제: `arn:aws:gameliftstreams:us-west-2:111122223333:streamgroup/sg-1AB2C3De4`

ID 예제: `sg-1AB2C3De4`

location-configurations

이 스트림 그룹에서 새 용량으로 업데이트할 위치 집합입니다. 용량 값은 스트림 그룹 스트림 클래스의 테넌시 값의 정수 배수여야 합니다.

스트림 그룹 위치의 용량을 업데이트하면 Amazon GameLift Streams가 요청을 처리하기 시작하므로 시간이 걸릴 수 있습니다. 이 기간 동안 Amazon GameLift Streams는 설정한 원하는 상시 가동 스트림 용량을 충족하기 위해 필요에 따라 스트림 그룹에 리소스를 할당하거나 릴리스합니다. Amazon GameLift Streams 콘솔에서 스트림 그룹 세부 정보 페이지를 보거나 [GetStreamGroup](#) API를 호출하여 스트림 용량의 프로비저닝 상태를 볼 수 있습니다.

스트림 그룹이 활성 상태이고 사용 가능한 스트림 용량이 있으며 애플리케이션이 스트리밍하려는 위치로 복제를 완료한 경우 스트리밍을 시작할 수 있습니다. 자세한 정보는 [Amazon GameLift Streams로 스트림 세션 시작](#) 섹션을 참조하세요.

용량 축소 동작

용량을 축소하면 Amazon GameLift Streams는 호스트가 유휴 상태가 될 때까지 기다렸다가 릴리스합니다. 호스트는 1개 또는 2개의 세션을 지원할 수 있으므로 호스트의 모든 활성 세션이 종료될 때만 호스트가 유휴 상태입니다. 스트림 세션은 사용자가 세션을 종료하거나 세션 시간이 초과될 때 종료됩니다. 따라서 기존 세션이 가능한 최대 기간에 도달하도록 허용되는 극단적인 상황에서는 원하는 용량에 도달하는 데 최대 24시간이 걸릴 수 있습니다. 스트림 그룹의 모든 활성 스트림 세션을 강제로 종료하려면 콘솔에서 또는 [DeleteStreamGroup](#) API를 사용하여 스트림 그룹을 삭제하거나 [TerminateStreamSession](#) API를 사용하여 활성 세션을 한 번에 하나씩 종료할 수 있습니다.

스트림 그룹에 위치 추가

Console

Amazon GameLift Streams 콘솔을 사용하여 스트림 그룹에 위치를 추가하려면

1. 탐색 모음에서 스트림 그룹을 선택하여 기존 스트림 그룹 목록을 봅니다. 새 위치를 추가할 스트림 그룹을 선택합니다.
2. 스트림 그룹 세부 정보 페이지에서 구성 편집을 선택합니다.
3. 이 스트림 그룹에 추가할 위치(들) 옆의 확인란을 선택한 다음 용량을 설정합니다.
4. 스트림 용량 비용을 포함하여 선택한 위치의 요약을 검토합니다. 저장을 선택하여 선택을 확인합니다.

CLI

사전 조건

사용자 자격 증명과 선택한 AWS CLI 사용하여 구성해야 합니다 AWS 리전. 설정 지침은 단원을 참조하십시오 [다운로드 AWS CLI](#).

를 사용하여 스트림 그룹에 위치를 추가하려면 AWS CLI

에서 콘텐츠에 맞게 사용자 지정된 [AddStreamGroupLocations](#) 명령을 AWS CLI 사용합니다.

```
aws gameliftstreams add-stream-group-locations \
  --identifier arn:aws:gameliftstreams:us-west-2:111122223333:streamgroup/
sg-1AB2C3De4
  --location-configurations '[{"LocationName": "us-east-1", "AlwaysOnCapacity": 2,
"MaximumCapacity": 4, "TargetIdleCapacity": 1}]'
```

where

identifier

스트림 그룹 [리소스를 고유하게 식별하는 Amazon 리소스 이름\(ARN\)](#) 또는 ID입니다.

ARN 예제: arn:aws:gameliftstreams:us-west-2:111122223333:streamgroup/sg-1AB2C3De4

ID 예제: sg-1AB2C3De4

location-configurations

이 스트림 그룹에 추가할 위치 집합과 해당 용량입니다. Amazon GameLift Streams가 지원하는 위치의 전체 목록은 [섹션을 참조하세요 AWS 리전 Amazon GameLift Streams에서 지원하는 및 스트리밍 위치](#).

용량 값은 스트림 그룹 스트림 클래스의 테넌시 값의 정수 배수여야 합니다.

애플리케이션이 새 위치(들)로 복제를 완료하고 스트림 그룹에 사용 가능한 스트림 용량이 있으면 새 위치(들)에서 스트리밍을 시작할 수 있습니다. 스트리밍에 대한 자세한 내용은 [섹션을 참조하세요 Amazon GameLift Streams로 스트림 세션 시작](#). Amazon GameLift Streams가 요청 처리를 시작합니다. 이 기간 동안 Amazon GameLift Streams는 애플리케이션을 복제하고 새 위치에 컴퓨팅 리소스를 할당합니다. 복제 상태 열의 상태를 클릭하여 스트림 그룹 세부 정보 페이지의 연결된 애플리케이션 섹션에서 복제 상태를 볼 수 있습니다.

스트림 그룹의 위치 제거

특정 위치의 컴퓨팅 리소스 사용을 중지하려면 스트림 그룹에서 위치를 제거할 수 있습니다. 스트림 그룹의 기본 위치는 제거할 수 없습니다. 그러나 해당 위치에 컴퓨팅 리소스를 원하지 않는 경우 스트림 용량을 0으로 설정할 수 있습니다.

⚠ Warning

스트림 그룹의 위치를 제거하면 Amazon GameLift Streams는 해당 위치에서 활성 스트림의 연결을 해제하여 연결된 최종 사용자의 스트림을 중지합니다.

Console

Amazon GameLift Streams 콘솔을 사용하여 스트림 그룹에서 위치를 제거하려면

1. 탐색 창에서 스트림 그룹을 선택하여 기존 스트림 그룹 목록을 봅니다.
2. 위치를 제거할 스트림 그룹의 이름을 선택합니다.
3. 스트림 그룹 세부 정보 페이지에서 구성 편집을 선택합니다.
4. 제거하려는 위치 이름 옆의 확인란을 선택 취소합니다.
5. 저장을 선택합니다.

CLI**사전 조건**

사용자 자격 증명과 선택한 AWS CLI 사용자를 구성해야 합니다 AWS 리전. 설정 지침은 단원을 참조하십시오 [다운로드 AWS CLI](#).

를 사용하여 스트림 그룹에서 위치를 제거하려면 AWS CLI

에서 콘텐츠에 맞게 사용자 지정된 [RemoveStreamGroupLocations](#) 명령을 AWS CLI 사용합니다.

```
aws gameliftstreams remove-stream-group-locations \
  --identifier arn:aws:gameliftstreams:us-west-2:111122223333:streamgroup/sg-1AB2C3De4
  --locations us-east-1 eu-central-1
```

where

identifier

스트림 그룹 [리소스를 고유하게 식별하는 Amazon 리소스 이름\(ARN\)](#) 또는 ID입니다.

ARN 예제: `arn:aws:gameliftstreams:us-west-2:111122223333:streamgroup/sg-1AB2C3De4`

ID 예제: sg-1AB2C3De4

locations

이 스트림 그룹에서 제거할 위치 집합입니다. Amazon GameLift Streams가 지원하는 위치의 전체 목록은 섹션을 참조하세요 [AWS 리전 Amazon GameLift Streams에서 지원하는 및 스트리밍 위치](#).

스트림 그룹 삭제

모든 상태의 스트림 그룹을 삭제할 수 있습니다. 이 작업은 스트림 그룹을 영구적으로 삭제하고 해당 컴퓨팅 리소스를 해제합니다. 진행 중인 스트림이 있는 경우 이 작업은 스트림을 중지하고 최종 사용자는 더 이상 스트림을 볼 수 없습니다.

스트림 그룹을 삭제하기 전에 진행 중인 스트림을 확인하고 중지 조치를 취하는 것이 가장 좋습니다.

Console

Amazon GameLift Streams 콘솔을 사용하여 스트림 그룹을 삭제하려면

1. 에 로그인 AWS Management Console 하고 [Amazon GameLift Streams 콘솔](#)을 엽니다.
2. 기존 스트림 그룹 목록을 보려면 탐색 창에서 스트림 그룹을 선택합니다.
3. 삭제할 스트림 그룹의 이름을 선택합니다.
4. 스트림 그룹 세부 정보 페이지에서 삭제를 선택합니다.
5. 삭제 대화 상자에서 삭제 작업을 확인합니다.

CLI

사전 조건

사용자 자격 증명과 선택한 AWS CLI 사용자를 구성해야 합니다 AWS 리전. 설정 지침은 단원을 참조하십시오 [다운로드 AWS CLI](#).

를 사용하여 스트림 그룹을 삭제하려면 AWS CLI

에서 콘텐츠에 맞게 사용자 지정된 [DeleteStreamGroup](#) 명령을 AWS CLI 사용합니다.

```
aws gameliftstreams delete-stream-group \
  --identifier arn:aws:gameliftstreams:us-west-2:111122223333:streamgroup/
  sg-1AB2C3De4
```

where

identifier

스트림 그룹 [리소스를 고유하게 식별하는 Amazon 리소스 이름\(ARN\)](#) 또는 ID입니다.

ARN 예제: `arn:aws:gameliftstreams:us-west-2:111122223333:streamgroup/sg-1AB2C3De4`

ID 예제: `sg-1AB2C3De4`

Amazon GameLift Streams는 컴퓨팅 리소스를 해제하고 스트림 그룹을 삭제하기 시작합니다. 이 시간 동안 스트림 그룹은 삭제 중 상태입니다. Amazon GameLift Streams가 스트림 그룹을 삭제한 후에는 더 이상 검색할 수 없습니다.

연결된 애플리케이션

동일한 컴퓨팅 리소스 풀을 사용하여 여러 애플리케이션을 스트리밍하려는 경우 여러 애플리케이션을 동일한 스트림 그룹에 연결할 수 있습니다. 마찬가지로 다양한 컴퓨팅 리소스 세트를 사용하여 애플리케이션을 스트리밍하려는 경우 애플리케이션을 여러 스트림 그룹에 연결할 수 있습니다.

애플리케이션을 스트림 그룹에 연결하는 방법에 대한 자세한 내용은 [섹션을 참조하세요](#) [다중 애플리케이션 스트림 그룹 개요](#).

스트림 그룹 수명 주기

스트림 그룹의 최대 수명은 365일입니다. 중요한 서비스 업데이트 및 수정 사항을 수신하고 최적의 성능을 보장하려면 3~4주마다 스트림 그룹을 다시 생성하는 것이 좋습니다. 스트림 그룹을 다시 생성해도 업로드된 애플리케이션에는 영향을 주지 않습니다.

스트림 그룹의 수명에 따라 다음과 같은 제한이 적용됩니다.

- 180일: 새 애플리케이션 연결로 스트림 그룹을 더 이상 업데이트할 수 없습니다.
- 365일: 스트림 그룹이 만료되어 더 이상 세션을 스트리밍할 수 없음

스트림 그룹과 연결된 계정은에서 두 개의 알림 알림을 수신합니다. AWS Health 하나는 45일에, 두 번째 알림은 150일에 수신됩니다. 이러한 알림은 180일 이내에 애플리케이션 연결 기능이 손실됨을 알려 줍니다. 또한 스트림 그룹이 365일에 만료됨을 알리는 335일 최종 알림이 하나 있을 예정입니다. 유지

관리 경고는 AWS Health 대시보드와 Amazon GameLift Streams 콘솔의 스트림 그룹 페이지에도 표시됩니다.

스트림 그룹의 만료 날짜를 찾으려면 콘솔에서 스트림 그룹 세부 정보 페이지를 보거나 [GetStreamGroup](#) API 응답의 ExpiresAt 필드를 사용합니다.

만료된 스트림 그룹의 상태는 EXPIRED 이고 읽기 전용이 됩니다. 업데이트하거나 새 스트림 세션을 시작할 수 없습니다. 기능을 다시 얻으려면 스트림 그룹을 다시 생성합니다.

스트림 그룹 유지 관리

새 스트림 그룹을 사용해야 하는 기능이 릴리스될 때마다 스트림 그룹의 세부 정보 페이지 상단에 만료되었음을 알리는 "유지 관리 필요" 메시지가 표시됩니다. 스트림 그룹을 다시 생성하는 것은 수동 프로세스이지만, 이를 위해 메시지의 스트림 그룹 생성 버튼을 사용하여 프로세스를 시작합니다. 일부 필드는 자동으로 채워집니다.

스트림 그룹이 생성된 지 180일이 넘은 경우에도 스트림 그룹 유지 관리가 필요합니다. 기존 스트림 그룹을 다시 생성하기 전까지는 새 애플리케이션을 이러한 오래된 스트림 그룹에 더 이상 연결할 수 없습니다. 365일째가 되면 스트림 그룹에서 스트리밍할 수 없으며 스트림 그룹을 변경할 수 없습니다.

다중 애플리케이션 스트림 그룹 개요

다중 애플리케이션 스트림 그룹은 여러 애플리케이션에 연결된 스트림 그룹입니다. 이를 통해 단일 스트림 그룹에서 동일한 컴퓨팅 리소스 세트를 사용하여 여러 애플리케이션을 스트리밍할 수 있습니다.

다중 애플리케이션 스트림 그룹의 일반적인 사용 사례는 게임의 다양한 버전을 릴리스하는 것입니다. 예를 들어 스트림 그룹을 생성하고 기본 애플리케이션을 게임의 원래 버전으로 설정했다고 가정해 보겠습니다. 그런 다음 게임의 다른 버전이 포함된 추가 애플리케이션을 생성하여 스트림 그룹에 연결한다고 가정해 보겠습니다. 이러한 애플리케이션은 동일한 스트림 그룹과 연결되므로 이러한 모든 게임을 스트리밍하려면 단일 컴퓨팅 리소스 세트 또는 스트림 용량만 관리하면 됩니다. 즉, 최종 사용자 스트림의 애플리케이션에 관계없이 애플리케이션은 이 스트림 그룹이 할당한 것과 동일한 세트의 컴퓨팅 리소스에서 실행됩니다.

다음은 가능한 다른 실제 예입니다.

- 고객에게 다양한 스트리밍 티어를 제공하는 게임 스트리밍 플랫폼입니다.
- 게임의 여러 버전을 테스트하는 품질 보증 팀입니다.
- 여러 애플리케이션에 단일 스트림 그룹을 사용하여 스트림 용량 관리를 간소화합니다.

- 애플리케이션 세트가 동일한 스트림 용량 풀에서 스트리밍할 수 있도록 합니다.

제한 사항 및 요구 사항

애플리케이션을 호환되는 런타임 환경 및 스트림 클래스가 있는 스트림 그룹에만 연결할 수 있습니다. 자세한 정보는 [스트림 클래스](#) 섹션을 참조하세요.

애플리케이션 및 스트림 그룹에는 다음 연결 제한이 적용됩니다. 이러한 제한은 모든 고객에 대해 서비스 내에서 고정됩니다.

이름	기본값	조정 가능	설명
스트림 그룹의 애플리케이션	250	아니요	스트림 그룹에 연결할 수 있는 Amazon GameLift Streams 애플리케이션의 최대 수입니다.
애플리케이션당 스트림 그룹 연결	100	아니요	Amazon GameLift Streams 애플리케이션을 연결할 수 있는 최대 스트림 그룹 수입니다.

기본 애플리케이션 정보

각 스트림 그룹에는 기본 애플리케이션이 하나 있는데, 이 애플리케이션은 처음에 스트림 그룹에 추가하는 첫 번째 애플리케이션입니다. 기본 애플리케이션은 모든 상시 가동 컴퓨팅 리소스에 자동으로 사전 캐싱되므로 스트림 시작 시 애플리케이션 로드 시간을 줄일 수 있습니다. Amazon GameLift Streams 서비스는 최적화 프로세스 중에 연결된 다른 애플리케이션을 캐싱할 수도 있습니다.

기본 애플리케이션 및 기타 연결된 애플리케이션의 특성:

- 기본 애플리케이션은 스트림 시작 중 애플리케이션 로드 시간을 줄이는 데 도움이 되는 사전 캐시(항상 가동 용량과 같은 사전 할당된 컴퓨팅 리소스)입니다.
- 기본 애플리케이션을 변경할 수 있습니다. 스트림 그룹의 기본 애플리케이션을 전환할 때 모든 위치에서 새 기본 애플리케이션을 사전 캐시하는 데 최대 몇 시간이 걸릴 수 있습니다.
- 스트림 그룹에서 스트리밍을 시작하려면 연결된 애플리케이션이 하나 이상 필요합니다. 연결된 첫 번째 애플리케이션이 자동으로 기본 애플리케이션이 됩니다.

- 스트림 그룹의 기본 애플리케이션 연결을 해제하면 Amazon GameLift Streams는 연결된 나머지 애플리케이션이 있는 경우 자동으로 새 기본 애플리케이션을 선택합니다.
- 동일한 애플리케이션이 여러 스트림 그룹의 기본 애플리케이션일 수 있습니다.
- 연결된 애플리케이션 세트는 스트림 그룹이 180일이 될 때까지 변경할 수 있습니다. 즉, 스트림 그룹이 180일이 될 때까지 애플리케이션을 연결 및 연결 해제할 수 있습니다. 그런 다음 스트림 그룹의 나머지 수명 주기 동안에만 스트림 그룹에서 애플리케이션을 연결 해제할 수 있습니다.

기본 애플리케이션 변경

첫 번째 애플리케이션을 스트림 그룹에 연결하면 애플리케이션이 자동으로 기본 애플리케이션이 되고 캐싱 전 이점을 받습니다. 언제든지 기본 애플리케이션을 변경하여 이러한 이점을 다른 애플리케이션에 제공할 수 있습니다.

Note

스트림 그룹의 기본 애플리케이션을 전환하면 모든 위치에서 새 기본 애플리케이션이 사전 캐시되는 데 최대 몇 시간이 걸릴 수 있습니다.

Console

Amazon GameLift Streams 콘솔을 사용하여 기본 애플리케이션을 변경하려면

1. 에 로그인 AWS Management Console 하고 [Amazon GameLift Streams 콘솔](#)을 엽니다.
2. 탐색 모음에서 스트림 그룹을 선택하여 기존 스트림 그룹 목록을 봅니다.
3. 세부 정보를 볼 스트림 그룹을 선택합니다.
4. 연결된 애플리케이션에서 기본값으로 설정할 애플리케이션을 선택합니다.
5. 기본값 만들기를 선택합니다.

CLI

사전 조건

사용자 자격 증명과 선택한 AWS CLI 사용자를 구성해야 합니다 AWS 리전. 설정 지침은 단원을 참조하십시오 [다운로드 AWS CLI](#).

를 사용하여 기본 애플리케이션을 변경하려면 AWS CLI

에서 콘텐츠에 맞게 사용자 지정된 [UpdateStreamGroup](#) 명령을 AWS CLI 사용합니다. 기본값을 설정하려는 애플리케이션은 스트림 그룹에 이미 연결되어 있어야 합니다.

```
aws gameliftstreams update-stream-group \
  --identifier arn:aws:gameliftstreams:us-west-2:111122223333:streamgroup/sg-1AB2C3De4 \
  --default-application-identifier a-9ZY8X7Wv6
```

where

- **identifier:**

기본값을 설정하려는 애플리케이션이 있는 스트림 그룹입니다.

이 값은 스트림 그룹 [리소스를 고유하게 식별하는 Amazon 리소스 이름\(ARN\)](#) 또는 ID일 수 있습니다.

ARN 예제: `arn:aws:gameliftstreams:us-west-2:111122223333:streamgroup/sg-1AB2C3De4`

ID 예제: `sg-1AB2C3De4`

- **default-application-identifier:**

이 스트림 그룹에서 기본값으로 설정하려는 애플리케이션입니다.

이 값은 애플리케이션 [리소스를 고유하게 식별하는 Amazon 리소스 이름\(ARN\)](#) 또는 ID입니다.

ARN 예제: `arn:aws:gameliftstreams:us-west-2:111122223333:application/a-9ZY8X7Wv6`

ID 예제: `a-9ZY8X7Wv6`

애플리케이션 연결

애플리케이션을 스트림 그룹에 연결하거나 연결할 때 스트림 그룹은 애플리케이션을 스트리밍할 수 있습니다. 스트림 그룹의 첫 번째 애플리케이션인 경우 자동으로 기본 애플리케이션이 됩니다. 추가 애플리케이션을 180일이 경과할 때까지 스트림 그룹에 연결하고 연결을 해제할 수 있습니다. 그런 다음에는 그룹의 나머지 수명 주기 동안에만 스트림 그룹에서 애플리케이션을 연결 해제할 수 있습니다.

⚠ Important

180일이 지난 스트림 그룹에는 애플리케이션을 연결할 수 없습니다. 여러 애플리케이션을 스트림 그룹에 연결하려면 먼저 다시 생성해야 합니다. 스트림 그룹을 다시 생성하는 방법에 대한 지침은 섹션을 참조하세요 [스트림 그룹 유지 관리](#).

애플리케이션을 연결하기 전에 스트림 그룹이 활성 상태인지 확인합니다.

Console

Amazon GameLift Streams 콘솔을 사용하여 연결하려면

1. 에 로그인 AWS Management Console 하고 [Amazon GameLift Streams 콘솔](#)을 엽니다.
2. 탐색 모음에서 스트림 그룹을 선택하여 기존 스트림 그룹 목록을 봅니다.
3. 세부 정보를 볼 스트림 그룹을 선택합니다.
4. 연결된 애플리케이션에서 애플리케이션 연결을 선택합니다.
5. 연결할 애플리케이션을 선택합니다. 선택을 확인하고 애플리케이션 연결을 선택합니다.

CLI**사전 조건**

사용자 자격 증명과 선택한 AWS CLI 사용하여 구성해야 합니다 AWS 리전. 설정 지침은 단원을 참조하십시오 [다운로드 AWS CLI](#).

를 사용하여 애플리케이션(들)을 연결하려면 AWS CLI

에서 콘텐츠에 맞게 사용자 지정된 [AssociateApplications](#) 명령을 AWS CLI 사용합니다.

```
aws gameliftstreams associate-applications \
  --identifier arn:aws:gameliftstreams:us-west-2:111122223333:streamgroup/sg-1AB2C3De4 \
  --application-identifiers a-9ZY8X7Wv6 a-1Z78C7Wv6
```

where

- **identifier:**

이러한 애플리케이션을 연결할 스트림 그룹입니다.

이 값은 스트림 그룹 [리소스를 고유하게 식별하는 Amazon 리소스 이름\(ARN\)](#) 또는 ID일 수 있습니다.

ARN 예제: `arn:aws:gameliftstreams:us-west-2:111122223333:streamgroup/sg-1AB2C3De4`

ID 예제: `sg-1AB2C3De4`

- `application-identifiers:`

이 스트림 그룹과 연결하려는 애플리케이션 세트입니다.

이 값은 애플리케이션 [리소스를 고유하게 식별하는 Amazon 리소스 이름\(ARN\)](#) 또는 ID입니다.

ARN 예제: `arn:aws:gameliftstreams:us-west-2:111122223333:application/a-9ZY8X7Wv6`

ID 예제: `a-9ZY8X7Wv6`

애플리케이션 연결 해제

스트림 그룹에서 애플리케이션을 연결 해제하거나 연결 해제하면 해당 스트림 그룹의 할당된 컴퓨팅 리소스를 사용하여 더 이상 애플리케이션을 스트리밍할 수 없습니다. 처리 중인 모든 스트림은 종료 될 때까지 계속되므로 최종 사용자의 스트림이 중단되지 않습니다. Amazon GameLift Streams는 이 스트림 그룹을 사용하여 새 스트림을 시작하지 않습니다. 연결 해제 작업은 스트림 그룹의 스트림 용량에 영향을 주지 않습니다.

스트림 그룹의 기본 애플리케이션 연결을 해제하면 Amazon GameLift Streams는 연결된 나머지 애플리케이션이 있는 경우 자동으로 새 기본 애플리케이션을 선택합니다.

Console

Amazon GameLift Streams 콘솔을 사용하여 연결을 해제하려면

1. 에 로그인 AWS Management Console 하고 [Amazon GameLift Streams 콘솔](#)을 엽니다.
2. 탐색 모음에서 스트림 그룹을 선택하여 기존 스트림 그룹 목록을 봅니다.
3. 세부 정보를 볼 스트림 그룹을 선택합니다.
4. 연결된 애플리케이션에서 연결을 해제할 애플리케이션(들)을 선택합니다. 애플리케이션 연결 해제를 선택합니다.

5. 애플리케이션 연결 해제 대화 상자에서 연결 해제 작업을 확인하고 연결 해제를 선택합니다.

CLI

사전 조건

사용자 자격 증명과 선택한 AWS CLI 사용자를 구성해야 합니다. AWS 리전. 설정 지침은 단원을 참조하십시오. [다운로드 AWS CLI](#).

를 사용하여 애플리케이션(들)의 연결을 해제하려면 AWS CLI

에서 콘텐츠에 맞게 사용자 지정된 [DisassociateApplications](#) 명령을 AWS CLI 사용합니다.

```
aws gameliftstreams disassociate-applications \
  --identifier arn:aws:gameliftstreams:us-west-2:111122223333:streamgroup/sg-1AB2C3De4 \
  --application-identifiers a-9ZY8X7Wv6 a-1Z78C7Wv6
```

where

- **identifier:**

이러한 애플리케이션의 연결을 해제할 스트림 그룹입니다.

이 값은 스트림 그룹 [리소스를 고유하게 식별하는 Amazon 리소스 이름\(ARN\)](#) 또는 ID일 수 있습니다.

ARN 예제: `arn:aws:gameliftstreams:us-west-2:111122223333:streamgroup/sg-1AB2C3De4`

ID 예제: `sg-1AB2C3De4`

- **application-identifiers:**

이 스트림 그룹에서 연결을 해제하려는 애플리케이션 세트입니다.

이 값은 애플리케이션 [리소스를 고유하게 식별하는 Amazon 리소스 이름\(ARN\)](#) 또는 ID입니다.

ARN 예제: `arn:aws:gameliftstreams:us-west-2:111122223333:application/a-9ZY8X7Wv6`

ID 예제: `a-9ZY8X7Wv6`

Amazon GameLift Streams로 스트림 세션 시작

이 섹션에서는 스트림 세션, 즉 최종 사용자 또는 플레이어가 애플리케이션과 상호 작용하거나 게임을 플레이할 수 있는 스트림의 실제 인스턴스를 다룹니다. 자체 스트림 세션을 테스트하고 스트림 세션 수명 주기를 이해하는 방법에 대해 알아봅니다.

최종 사용자에게 스트림 세션을 시작하려면 Amazon GameLift Streams를 자체 서비스에 통합해야 합니다. 자세한 정보는 [Amazon GameLift Streams 백엔드 서비스 및 웹 클라이언트](#) 섹션을 참조하세요.

스트림 세션 정보

스트림 세션을 시작하기 위한 사전 조건은 준비 상태의 애플리케이션, 스트리밍하려는 위치에서 사용 가능한 용량이 있는 스트림 그룹, 스트리밍하려는 위치에 복제된 애플리케이션입니다. 스트림 세션은 스트림 그룹이 할당된 컴퓨팅 리소스 중 하나에서 실행됩니다. 스트림을 시작할 때 ARN 또는 ID 값을 사용하여 스트리밍할 스트림 그룹과 애플리케이션을 지정해야 합니다.

스트림 세션을 성공적으로 시작하면 해당 스트림 세션에 대한 고유 식별자가 수신됩니다. 그런 다음 해당 ID를 사용하여 스트림 세션을 최종 사용자에게 연결합니다. 자세한 내용은 Amazon GameLift Streams API 참조의 [StartStreamSession](#)을 참조하세요. GameLift

콘솔에서 스트림 테스트

Amazon GameLift Streams 콘솔을 통해 애플리케이션 스트림을 테스트하는 가장 직접적인 방법입니다. 스트림을 시작하면 Amazon GameLift Streams는 스트림 그룹이 할당하는 컴퓨팅 리소스 중 하나를 사용합니다. 따라서 스트림 그룹에 사용 가능한 용량이 있어야 합니다.

Amazon GameLift Streams 콘솔에서 스트림을 테스트하려면

1. 에 로그인 AWS Management Console 하고 [Amazon GameLift Streams 콘솔](#)을 엽니다.
2. 여러 가지 방법으로 스트림을 테스트할 수 있습니다. 스트림 그룹 페이지 또는 스트림 테스트 페이지에서 시작하여 다음 단계를 따릅니다.
 - a. 스트리밍에 사용할 스트림 그룹을 선택합니다.
 - b. 스트림 그룹 페이지에서 시작하는 경우 스트림 테스트를 선택합니다. 스트림 테스트 페이지에서 시작하는 경우 선택을 선택합니다. 그러면 선택한 스트림 그룹에 대한 스트림 구성 테스트 페이지가 열립니다.
 - c. 연결된 애플리케이션에서 애플리케이션을 선택합니다.
 - d. 위치에서 사용 가능한 용량이 있는 위치를 선택합니다.

- e. (선택 사항) 프로그램 구성에서 시작 시 애플리케이션에 전달할 명령줄 인수 또는 환경 변수를 입력합니다.
 - f. 선택을 확인하고 스트림 테스트를 선택합니다.
3. 스트림이 로드된 후 스트림에서 다음 작업을 수행할 수 있습니다.
 - a. 마우스, 키보드, 게임패드(테스트 스트림에서 지원되지 않는 마이크 제외)와 같은 입력을 연결하려면 입력 연결을 선택합니다. 커서를 스트림 창으로 이동할 때 마우스를 자동으로 연결합니다.
 - b. 스트리밍 세션 중에 생성된 파일을 세션 종료 시 Amazon S3 버킷으로 내보내려면 파일 내보내기를 선택하고 버킷 세부 정보를 지정합니다. 내보낸 파일은 세션 페이지에서 찾을 수 있습니다.
 - c. 전체 화면에서 스트림을 보려면 전체 화면을 선택합니다. 이 작업을 되돌리려면 이스케이프를 누릅니다.
 4. 스트림을 종료하려면 세션 종료를 선택합니다. 스트림 연결이 끊어지면 스트림 용량을 사용하여 다른 스트림을 시작할 수 있게 됩니다.

Note

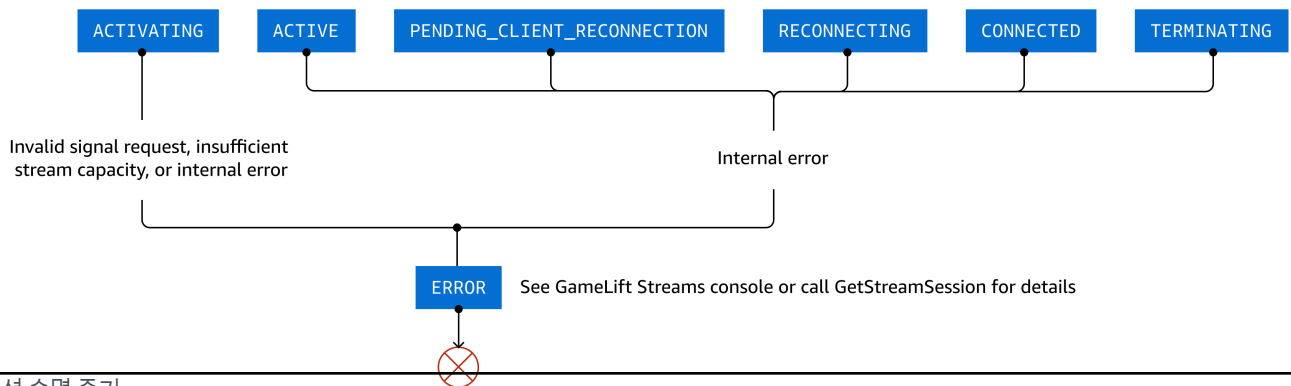
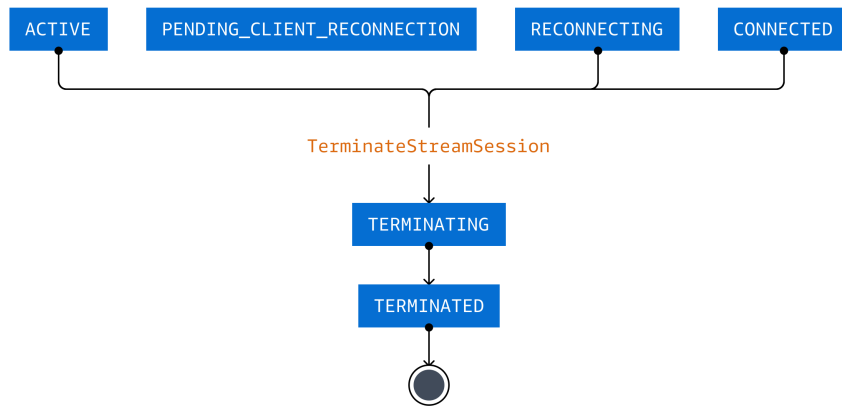
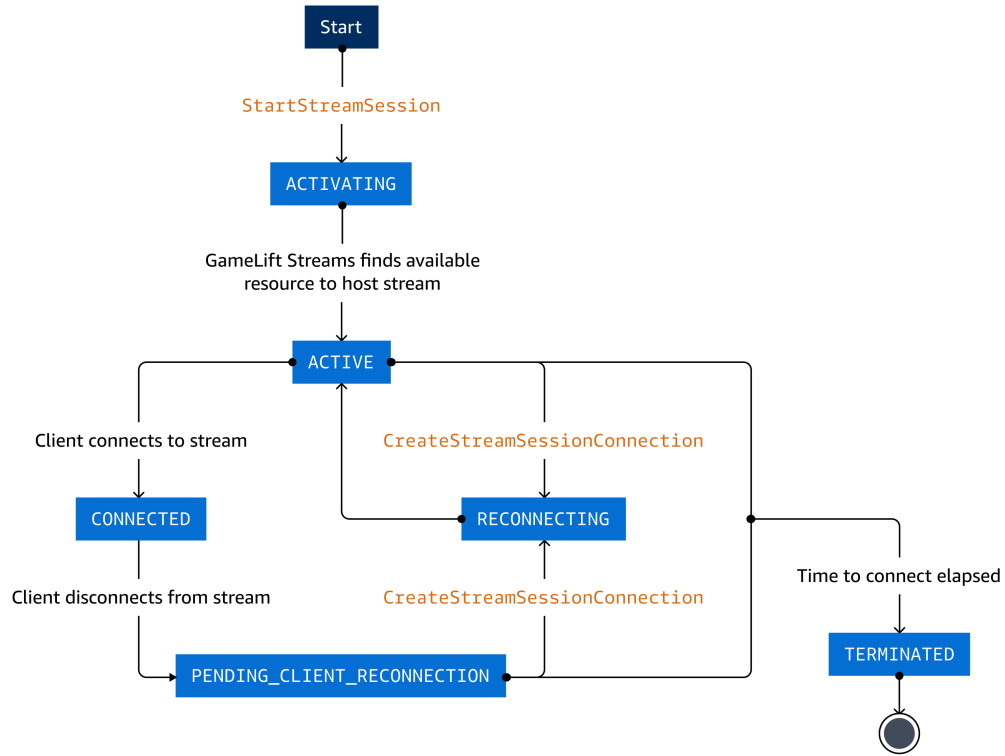
Amazon GameLift Streams 콘솔의 스트림 테스트 기능은 마이크를 지원하지 않습니다.

스트림 세션 수명 주기

Amazon GameLift Streams에서 스트림 세션을 사용할 때 다이어그램은 스트림 세션이 수명 주기 동안으로 전환되는 다양한 상태를 이해하는 데 도움이 될 수 있습니다.

- [StartStreamSession](#)은 ACTIVATING 상태에서 시작하는 새 스트림 세션을 생성합니다. Amazon GameLift Streams가 스트림을 호스팅할 수 있는 리소스를 찾으면 스트림 세션이로 전환됩니다ACTIVE. 클라이언트가 활성 스트림에 연결되면 스트림 세션이로 전환됩니다CONNECTED.
- 클라이언트가 스트림에서 연결을 끊으면 스트림 세션이 PENDING_CLIENT_RECONNECTION 상태로 전환됩니다. [CreateStreamSessionConnection](#)은 스트림 세션을로 전환RECONNECTING하고 클라이언트를 시작하여 스트림에 다시 연결하거나 새 스트림 세션을 생성합니다. 스트림 세션이 클라이언트를 다시 연결할 준비가 되면로 전환됩니다ACTIVE. 클라이언트가 다시 연결되면 다시로 전환됩니다CONNECTED. 클라이언트가 보다 오래 연결 해제되면 스트림 세션ConnectionTimeoutSeconds이 종료됩니다.

- 클라이언트가에 지정된 기간 내에 ACTIVE 또는 PENDING_CLIENT_RECONNECTION 상태의 스트림 세션에 연결하지 않으면 로 ConnectionTimeoutSeconds 전환됩니다TERMINATED.
- [TerminateStreamSession](#)은 스트림 종료를 시작하고 스트림 세션은 TERMINATING 상태로 전환됩니다. 스트림 세션이 성공적으로 종료되면 로 전환됩니다TERMINATED.
- 를 제외한 모든 상태의 스트림 세션은 로 전환할 TERMINATED수 있습니다ERROR. API 호출이 ERROR 상태 값으로 반환되면 StatusReason의 값에서 오류 원인에 대한 간단한 설명을 확인합니다. [GetStreamSession](#)을 호출하여 이러한 값을 확인할 수도 있습니다.



스트림 세션에 영향을 미치는 제한 시간 값

스트림 세션은 세션 수명 주기의 다양한 측면을 제어하는 여러 제한 시간 값에 의해 관리됩니다. 스트림 세션 수명 주기 동안 일반적으로 발생할 수 있는 시간순으로 다음을 포함합니다.

배치 제한 시간

Amazon GameLift Streams가 사용 가능한 용량을 사용하여 스트림 세션을 호스팅할 컴퓨팅 리소스를 찾는 데 걸리는 시간 제한입니다. 배치 제한 시간은 스트림 요청을 이행하는 데 사용되는 용량 유형에 따라 달라집니다.

- 상시 작동 용량: 75초
- 온디맨드 용량:
 - Linux/Proton 런타임: 90초
 - Windows 런타임: 10분
- 동작: Amazon GameLift Streams가 이 시간 내에 사용 가능한 리소스를 식별할 수 없는 경우 스트림 세션은 StatusReason의 ERROR를 사용하여 로 Status 변경됩니다placementTimeout.

연결 제한 시간

Amazon GameLift Streams가 클라이언트가 스트림 세션에 연결하거나 다시 연결할 때까지 기다리는 시간입니다.

- 파라미터: [StartStreamSession](#)ConnectionTimeoutSeconds의
- 범위: 1~3,600초(1시간)
- 기본값: 120초(2분)
- 동작: 스트림 세션이 ACTIVE 또는 PENDING_CLIENT_RECONNECTION 상태에 도달하면 타이머가 시작됩니다. 제한 시간 전에 클라이언트가 연결되지 않으면 세션이 로 Status 전환됩니다TERMINATED.

세션 길이 제한 시간

Amazon GameLift Streams의 최대 지속 시간은 스트림 세션을 열린 상태로 유지합니다.

- 파라미터: [StartStreamSession](#)SessionLengthSeconds의
- 범위: 1~86,400초(24시간)
- 기본값: 43200초(12시간)
- 동작: 시간 제한에 도달하면 기존 클라이언트 연결에 관계없이 스트림 세션을 종료합니다.

스트림 세션 종료

스트림 세션을 강제로 종료해야 하는 경우 다음과 같은 옵션이 있습니다.

- **TerminateStreamSession API 사용:** [TerminateStreamSession](#)을 사용하려면 스트림 그룹 ID와 스트림 세션 ID가 필요합니다. `--status CONNECTED` 파라미터와 함께 [ListStreamSessions](#) 또는 [ListStreamSessionsByAccount](#)를 사용하여 클라이언트가 연결된 스트림 세션 목록을 가져올 수 있습니다.
- **스트림 그룹에서 세션 위치 제거:** 세션이 스트리밍되는 스트림 그룹에서 위치를 제거하면 해당 위치의 모든 활성 스트림 세션이 종료됩니다. 콘솔에서 또는 [RemoveStreamGroupLocations](#) API를 사용하여 스트림 그룹의 위치를 제거할 수 있습니다.
- **세션의 스트림 그룹 삭제:** 스트림 그룹을 삭제하면 스트림 그룹의 모든 위치에서 모든 활성 스트림 세션이 종료됩니다. 콘솔에서 또는 [DeleteStreamGroup](#) API를 사용하여 스트림 그룹을 삭제할 수 있습니다. 클라이언트 연결이 갑자기 종료될 수 있으므로 주의해야 합니다.

스트림 세션에 다시 연결

클라이언트가 세션을 종료하지 않고 스트림 세션과의 연결이 끊어지면 스트림 세션이 시작된 `ConnectionTimeoutSeconds`에서 지정한 시간 내에 세션에 다시 연결할 수 있습니다. 세션에 다시 연결하려면 스트림 세션의 ID가 필요합니다. 자세한 내용은 Amazon GameLift Streams API 참조의 [CreateStreamSessionConnection](#)을 참조하세요. GameLift [React Starter 샘플](#)에서 스트림 세션에 다시 연결하는 예를 볼 수 있습니다.

스트림 세션 파일 내보내기

스트림 세션 중에 애플리케이션은 로그, 진단 정보, 충돌 덤프, 파일 저장, 사용자 데이터 및 스크린샷과 같이 애플리케이션을 디버깅하거나 확인하는 데 도움이 되는 출력 파일을 생성할 수 있습니다. 스트림 세션 파일 내보내기 기능은 세션 중에 생성되거나 수정된 파일을 수집하여 압축된 ZIP 파일로 제공된 Amazon S3 위치로 내보냅니다. 또한 이 기능은 내보내기 ZIP 파일에 포함된 세션의 성능 통계를 매초 수집합니다.

Warning

파일을 내보내기 전에 다음 사항에 유의하세요.

- 파일에는 자격 증명 정보를 포함하여 애플리케이션에서 작성한 민감한 정보가 포함될 수 있습니다.

- 파일 크기는 애플리케이션 크기에 따라 클 수 있으며, 이는 Amazon S3 스토리지 비용에 영향을 미칩니다.
- 스트림 그룹의 리전과 AWS 리전 다른에서 Amazon S3 버킷을 선택하면 내보낸 스트림 세션 파일이 리전 간에 이동합니다.

작동 방식

활성 스트림 세션에서 작업을 수동으로 호출하여 해당 세션 중에 생성된 파일을 내보내야 합니다. 스트림 세션은 특히, ACTIVE, CONNECTED PENDING_CLIENT_RECONNECTION 및 상태 중 하나에서 활성 상태여야 합니다 RECONNECTING. 세션이 끝나면 Amazon GameLift Streams는 Amazon Simple Storage Service(Amazon S3)의 버킷으로 파일을 내보냅니다. 따라서 내보낸 모든 데이터는 소유권 내에 있으며 Amazon S3 버킷의 권한 정책이 적용됩니다.

다음은 내보내기 파일이 활성화된 스트림 세션 수명 주기에 대한 안내입니다.

1. Amazon GameLift Streams는 사용자를 컴퓨팅 리소스에서 실행 중인 애플리케이션에 연결하여 세션을 시작합니다.
2. 애플리케이션이 스트리밍되는 동안 런타임 환경의 파일 시스템에 파일을 생성하거나 수정합니다.
3. 세션이 종료되면 Amazon GameLift Streams는 파일 시스템의 모든 새 파일 또는 수정된 파일의 사본을 가져와 Amazon S3 버킷으로 내보냅니다.

Amazon GameLift Streams는 다음과 같이 생성 및 수정된 파일을 수집합니다. .zip 아카이브의 해당 폴더에서 해당 폴더를 찾습니다.

- application/: 애플리케이션 또는 게임이 저장되는 폴더입니다.
- profile/: 사용자의 프로필 폴더에는 사용자의 개인 설정, 구성 및 데이터가 포함되어 있습니다.
- temp/: 시스템의 임시 폴더에는 애플리케이션과 시스템이 생성하는 임시 파일과 데이터가 포함되어 있습니다. 여기에는 캐시 파일, 로그 파일 또는 중간 처리 데이터가 포함될 수 있습니다.
- stats/: 초당 수집된 세션의 성능 통계를 보관perf_stats_v1.csv하는를 포함합니다. 여기에는 애플리케이션 수준 통계(CPU 및 메모리 사용률) 및 시스템 수준 통계(CPU, 메모리, GPU 및 VRAM 사용률)가 포함됩니다. CSV 파일에 포함된 각 통계에 대한 자세한 설명은 섹션을 참조하세요. [the section called “성능 통계 참조”](#)

파일을 삭제하려면 Amazon S3 버킷에서 객체를 삭제합니다.

비용 영향

Amazon S3에 파일을 저장하는 데 비용이 발생합니다. 스트림 세션은 애플리케이션에 따라 대량의 데이터를 생성할 수 있습니다. 이 기능이 활성화된 스트림 세션이 많으면 비용이 증가할 수 있습니다.

자세한 내용은 [Amazon S3 요금](#)을 참조하세요.

파일 내보내기(콘솔)

Amazon GameLift Streams 콘솔에서 스트림 세션 파일 내보내기를 활성화하려면

1. 에 로그인 AWS Management Console 하고 [Amazon GameLift Streams 콘솔](#)을 엽니다.
2. 탐색 모음에서 세션을 선택하여 지난 90일 동안의 활성 및 최근 스트림 세션 목록을 봅니다.
3. 활성 세션 탭에서 활성 스트림 세션을 선택합니다.
4. 파일 내보내기를 선택하여 해당 스트림 세션에 대해 파일 내보내기 기능을 활성화합니다.
5. 스트림 세션 파일 내보내기 대화 상자에서 새 S3 버킷 생성 또는 기존 S3 버킷 선택을 선택합니다. 콘솔의 단계에 따라 내보낸 데이터를 저장할 S3 객체를 생성하거나 선택합니다.

Warning

ZIP 파일 이름이 디렉터리의 기존 파일 이름과 일치하면 이전 파일을 덮어씁니다.

6. 확인을 선택합니다. 이제 내보낸 파일 탭에서 세션을 찾을 수 있습니다.
7. 세션이 종료되고 파일을 내보낼 때까지 기다립니다.

Amazon GameLift Streams는 세션이 종료됨 상태일 때 파일을 내보냅니다. 세션이 종료되면 활성 세션 탭에서 최근 세션 탭으로 이동합니다.

세션 내보내기 탭에서 내보내기 프로세스의 상태를 확인할 수 있습니다. 상태가 보류 중이면 스트림 세션이 여전히 활성 상태이므로 Amazon GameLift Streams가 아직 파일을 내보내지 않은 것입니다. 상태가 성공이면 제공된 링크를 사용하여 Amazon S3에서 파일을 다운로드할 수 있습니다. 상태가 실패인 경우 상태 위에 마우스를 올려 놓으면 실패 이유를 확인할 수 있습니다.

파일 내보내기(CLI)

사전 조건

사용자 자격 증명과 선택한 AWS CLI 사용하여 구성해야 합니다 AWS 리전. 설정 지침은 단원을 참조하십시오 [다운로드 AWS CLI](#).

에서 스트림 세션 파일을 내보내려면 AWS CLI

에서 콘텐츠에 맞게 사용자 지정된 [ExportStreamSessionFiles](#) 명령을 AWS CLI 사용합니다.

```
aws gameliftstreams export-stream-session-files \
  --identifier arn:aws:gameliftstreams:us-west-2:111122223333:streamgroup/sg-1AB2C3De4 \
  --stream-session-identifier arn:aws:gameliftstreams:us-west-2:111122223333:streamsession/sg-1AB2C3De4/ABC123def4567
  --output-uri s3://amzn-s3-demo-bucket/prefix
```

위치

`identifier`

스트림 그룹 [리소스를 고유하게 식별하는 Amazon 리소스 이름\(ARN\)](#) 또는 ID입니다.

ARN 예제: `arn:aws:gameliftstreams:us-west-2:111122223333:streamgroup/sg-1AB2C3De4`

ID 예제: `sg-1AB2C3De4`

`stream-session-identifier`

스트림 세션 [리소스를 고유하게 식별하는 Amazon 리소스 이름\(ARN\)](#) 또는 ID입니다.

ARN 예제: `arn:aws:gameliftstreams:us-west-2:111122223333:streamsession/sg-1AB2C3De4/ABC123def4567`

ID 예제: `ABC123def4567`

`output-uri`


Amazon GameLift Streams가 이 스트림 세션에 대해 압축된 내보낸 파일 세트를 업로드하는 Amazon S3 버킷 URI입니다.

제공할 수 있는 유효한 형식은 두 가지입니다. URI에 `.zip` 또는 `.ZIP` 파일 확장명이 있는 경우 Amazon GameLift Streams는 내보낸 파일을 제공된 URI에 저장합니다. 그렇지 않으면 Amazon GameLift Streams는 압축된 폴더의 이름을 생성하여 URI에 저장합니다. 생성된 이름은 패턴을 따릅니다 `date-time-applicationId-streamGroupId-streamSessionId`. 예제:

- 라는 URI를 제공하면 `s3://amzn-s3-demo-bucket/MyGame_Session1.zip` Amazon GameLift Streams가 정확한 ZIP 폴더에 파일을 저장합니다.

- 라는 URI를 제공하면 `s3://amzn-s3-demo-bucket/MyGame_Session1/Amazon GameLift Streams`가 파일을 저장합니다 `s3://amzn-s3-demo-bucket/MyGame_Session1/YYYYMMDD-HHMMSS-applicationId-streamGroupId-sessionId.zip`.

ZIP 파일 이름이 Amazon Simple Storage Service 사용 설명서의 [객체 키 이름 지정 지침](#)을 준수하는지 확인합니다.

 Warning

ZIP 파일 이름이 디렉터리의 기존 파일 이름과 일치하면 이전 파일을 덮어씁니다.

[GetStreamSession](#) API를 호출하여 활성 세션의 상태를 확인할 수 있습니다. 스트림 세션 요약에서 내보낸 파일의 상태에 대한 세부 정보를 얻을 수 있습니다. 상태가 보류 중이면 스트림 세션이 여전히 활성 상태이므로 Amazon GameLift Streams가 아직 파일을 내보내지 않은 것입니다. 상태가 성공이면 출력 URI로 이동하여 Amazon S3의 파일을 확인합니다. 상태가 실패인 경우 `StatusReason`에서를 확인합니다 `ExportFilesMetaData`.

Amazon GameLift Streams에 Amazon VPC의 리소스에 대한 액세스 권한 부여

기본적으로 Amazon GameLift Streams는 퍼블릭 인터넷에 액세스할 수 있지만 프라이빗 Amazon VPCs의 리소스에는 액세스할 수 없는 컴퓨팅 리소스에서 스트리밍 애플리케이션을 실행합니다. 스트리밍 애플리케이션에 데이터베이스, 캐시 서버 또는 내부 APIs와 같은 프라이빗 리소스에 대한 액세스 권한을 부여하려면 스트림 그룹을 생성할 때 VPC 연결을 구성할 수 있습니다.

Amazon GameLift Streams는 AWS Transit Gateway를 사용하여 스트림이 실행되는 서비스 관리형 VPC와 자체 Amazon VPC 간에 프라이빗 네트워크 연결을 설정합니다. 이를 통해 스트리밍 애플리케이션은 트래픽을 퍼블릭 인터넷에 노출하지 않고도 프라이빗 IP 주소를 통해 Amazon VPC의 리소스와 통신할 수 있습니다.

VPC 연결 작동 방식

AWS Transit Gateway는 Virtual Private Cloud(VPCs)와 온프레미스 네트워크를 상호 연결하는 데 사용할 수 있는 네트워크 전송 허브입니다. 전송 게이트웨이는 VPCs와 기타 연결된 네트워크 간에 흐르는 트래픽을 위한 리전 가상 라우터 역할을 합니다. 전송 게이트웨이에 대한 자세한 내용은 Amazon VPC Transit Gateway 안내서의 [전송 게이트웨이란 무엇입니까?](#)를 참조하세요.

VPC 연결이 활성화된 스트림 그룹 위치를 생성하면 Amazon GameLift Streams가 다음 작업을 수행합니다.

1. 스트리밍 위치에 전송 게이트웨이를 생성합니다(또는 동일한 VPC에 연결된 다른 스트림 그룹이 있는 경우 기존 게이트웨이를 재사용합니다).
2. AWS Resource Access Manager(RAM)를 AWS 계정 사용하여 전송 게이트웨이를와 공유합니다.
3. 스트림 그룹의 Amazon GameLift Streams 서비스 관리형 VPC를 전송 게이트웨이에 연결합니다.
4. 전송 게이트웨이를 통해 CIDR 블록으로 향하는 트래픽을 전달하도록 Amazon GameLift Streams 서비스 관리형 VPC의 라우팅을 구성합니다.

스트림 그룹 위치가 활성화되면 VPC 연결이 구성된 각 스트림 그룹 위치에 대해 다음 단계를 수행하여 설정을 완료할 수 있습니다. 자세한 지침은 [the section called “VPC 연결 구성”](#) 섹션을 참조하세요.

1. RAM 리소스 공유 초대 수락 - 계정에 전송 게이트웨이에 대한 액세스 권한을 부여합니다. 동일한 VPC를 사용하는 다른 스트림 그룹에 대한 리소스 공유 초대를 이미 수락한 경우 다시 수락할 필요가 없습니다.

Note

리소스 공유 초대는 7일 후에 만료됩니다. 초대를 수락하기 전에 초대가 만료되면 스트림 그룹 또는 스트림 그룹 위치를 삭제하고 다시 생성하여 새 초대를 생성해야 합니다.

2. VPC 연결 생성 - VPC를 공유 전송 게이트웨이에 연결합니다. `CreateStreamGroup` 요청과 일치하는 VPC만 전송 게이트웨이에 연결할 수 있습니다.
3. VPC 라우팅 테이블에 경로 추가 - 전송 게이트웨이를 통해 Amazon GameLift Streams 서비스 관리형 VPC로 향하는 트래픽을 지시합니다.
4. (선택 사항) 보안 그룹 업데이트 - Amazon GameLift Streams 서비스 관리형 VPC CIDR 블록의 인바운드 트래픽이 프라이빗 리소스에 도달하도록 허용합니다.

요구 사항 및 고려 사항

요구 사항

VPC 연결에는 다음과 같은 요구 사항이 있습니다.

- 겹치는 CIDR 블록 없음: VPC CIDR 블록이 서비스 VPC CIDR 블록과 겹칠 수 없습니다. `Ipv4CidrBlocks` 파라미터에 VPC CIDR 블록을 지정하면 Amazon GameLift Streams는 사용자가 제공한 CIDR 블록과 겹치지 않는 서비스 VPC CIDR 블록을 자동으로 선택합니다. 를 호출하면 서비스 VPC CIDR 블록이 `InternalVpcIpv4CidrBlock` 필드에 반환됩니다 `GetStreamGroup`. VPC에서 경로를 구성할 때 이 값을 사용해야 합니다.
- 동일한 계정: VPC는 스트림 그룹을 생성한 AWS 계정 것과 동일한에 있어야 합니다.
- VPC ID는 기본 위치에 대해 변경할 수 없음: 스트림 그룹이 생성된 후에는 스트림 그룹의 기본 위치에 대한 VPC ID를 변경할 수 없습니다. 그러나 다른 스트리밍 위치의 경우 스트림 그룹 위치를 삭제하고 다른 VPC ID로 다시 생성하여 VPC를 변경할 수 있습니다. [UpdateStreamGroup](#)을 호출하여 모든 위치의 CIDR 블록을 업데이트할 수 있습니다.
- VPC 리전은 스트리밍 위치와 일치해야 합니다. VPC는 스트리밍 위치와 동일한 리전에 있어야 합니다. 예를 들어에 스트리밍 위치를 추가하는 경우에 있는 VPC를 지정eu-west-1해야 합니다eu-west-1.
- IPv4 전용: 이중 스택 IPv6를 지원하는 스트림 그룹의 경우 현재 IPv4 VPC 트래픽만 지원됩니다.

필수 IAM 권한

VPC 연결을 구성하려면 GameLift Streams 권한 외에도 IAM 자격 증명에 다음 권한이 있어야 합니다.

- `ec2:DescribeVpcs` - Amazon GameLift Streams가 VPC 구성을 검증하는 데 필요합니다.
- `ec2:CreateTransitGatewayVpcAttachment` - VPC를 전송 게이트웨이에 연결하는 데 필요합니다.
- `ec2:CreateRoute` - VPC 라우팅 테이블에 경로를 추가하는 데 필요합니다.
- `ram:AcceptResourceShareInvitation` - 전송 게이트웨이 리소스 공유를 수락하는 데 필요합니다.

추가 고려 사항

VPC 연결을 구성하기 전에 다음 사항을 고려하세요.

- 추가 지연 시간: 전송 게이트웨이를 통해 라우팅되는 트래픽은 직접 연결에 비해 지연 시간이 약간 더 길어질 수 있습니다.
- 비용: 전송 게이트웨이 연결에는 추가 요금이 발생합니다. 자세한 내용은 [AWS Transit Gateway 요금](#)을 참조하세요.
- 할당량: 리전별 계정당 VPC 전송 구성의 기본 제한은 5개입니다.

VPC 연결 구성

이 섹션에서는를 사용하여 Amazon GameLift Streams 스트림 그룹에 대한 VPC 연결을 구성하는 방법을 안내합니다 AWS CLI.

1단계: VPC 구성을 사용하여 스트림 그룹 생성

스트림 그룹을 생성할 때 위치 구성에 `VpcTransitConfiguration` 파라미터를 포함합니다. 스트리밍 애플리케이션이 액세스해야 하는 VPC ID와 CIDR 블록을 지정합니다.

```
aws gameliftstreams create-stream-group \
  --description "Stream group with VPC connectivity" \
  --stream-class gen5n_high \
  --default-application-identifier arn:aws:gameliftstreams:us-
west-2:123456789012:application/a-ABC123def \
  --location-configurations '[{
    "LocationName": "us-west-2",
    "AlwaysOnCapacity": 1,
    "VpcTransitConfiguration": {
      "VpcId": "vpc-0123456789abcdef0",
      "Ipv4CidrBlocks": ["10.0.0.0/16"]
    }
  }]'
```

스트림 그룹이 활성화될 때까지 기다립니다.

```
aws gameliftstreams wait stream-group-active \
  --identifier sg-1AB2C3De4
```

스트림 그룹 상태가 이면 스트림 그룹 세부 정보를 ACTIVE 가져오고 응답에서 다음 값을 기록해 둡니다.

```
aws gameliftstreams get-stream-group \
  --identifier sg-1AB2C3De4
```

- `TransitGatewayId` - Amazon GameLift Streams에서 생성한 전송 게이트웨이의 ID입니다.
- `TransitGatewayResourceShareArn` - RAM 리소스 공유의 ARN입니다.
- `InternalVpcIpv4CidrBlock` - 라우팅 테이블에 추가해야 하는 서비스 VPC의 CIDR 블록입니다.

2단계: RAM 리소스 공유 수락

리소스 공유 초대를 수락하여 전송 게이트웨이에 대한 액세스 권한을 얻습니다.

```
# Get the resource share invitation
aws ram get-resource-share-invitations \
  --resource-share-arns arn:aws:ram:us-west-2:123456789012:resource-share/
abc12345-1234-1234-1234-abc123456789

# Accept the invitation
aws ram accept-resource-share-invitation \
  --resource-share-invitation-arn arn:aws:ram:us-west-2:123456789012:resource-share-
invitation/abc12345-1234-1234-1234-abc123456789
```

3단계: VPC 연결 생성

VPC를 전송 게이트웨이에 연결합니다. VPC에서 서브넷을 하나 이상 지정해야 합니다.

```
# Get your subnet IDs
aws ec2 describe-subnets \
  --filters "Name=vpc-id,Values=vpc-0123456789abcdef0" \
  --query "Subnets[*].SubnetId"

# Create the VPC attachment
aws ec2 create-transit-gateway-vpc-attachment \
  --transit-gateway-id tgw-0123456789abcdef0 \
  --vpc-id vpc-0123456789abcdef0 \
  --subnet-ids subnet-0123456789abcdef0 subnet-0123456789abcdef1
```

첨부 파일을 사용할 수 있을 때까지 기다립니다.

```
aws ec2 describe-transit-gateway-vpc-attachments \
  --transit-gateway-attachment-ids tgw-attach-0123456789abcdef0 \
  --query "TransitGatewayVpcAttachments[0].State"
```

4단계: 라우팅 구성

VPC 라우팅 테이블에 경로를 추가하여 전송 게이트웨이를 통해 서비스 VPC로 향하는 트래픽을 전달합니다. 스트림 그룹 응답의 `InternalVpcIpv4CidrBlock` 값을 사용합니다.

```
# Get your route table ID
aws ec2 describe-route-tables \
  --filters "Name=vpc-id,Values=vpc-0123456789abcdef0" \
  --query "RouteTables[*].RouteTableId"

# Add the route
aws ec2 create-route \
  --route-table-id rtb-0123456789abcdef0 \
  --destination-cidr-block 10.1.0.0/16 \
  --transit-gateway-id tgw-0123456789abcdef0
```

Note

10.1.0.0/16를 스트림 그룹의 실제 `InternalVpcIpv4CidrBlock` 값으로 바꿉니다.

(선택 사항) 5단계: 보안 그룹 업데이트

VPC의 EC2 인스턴스에 연결할 때 애플리케이션이 EC2 인스턴스로 트래픽을 전송할 수 있도록 서비스 VPC CIDR 블록의 인바운드 트래픽을 허용하도록 EC2 인스턴스의 보안 그룹을 업데이트합니다.

```
aws ec2 authorize-security-group-ingress \
  --group-id sg-0123456789abcdef0 \
  --protocol tcp \
  --port 443 \
  --cidr 10.1.0.0/16
```

Note

다음 값을 실제 구성으로 바꿉니다.

- `sg-0123456789abcdef0` - 프라이빗 리소스의 보안 그룹 ID입니다.
- `tcp` - 애플리케이션이 사용하는 프로토콜(tcp 또는 udp).
- `443` - 애플리케이션이 수신 대기하는 포트 번호입니다.
- `10.1.0.0/16` - 스트림 그룹의 `InternalVpcIpv4CidrBlock` 값입니다.

(선택 사항) 6단계: CIDR 블록 업데이트

스트림 그룹을 다시 생성하지 않고도 스트림 그룹 위치의 VPC 연결 구성에 대한 CIDR 블록을 업데이트할 수 있습니다. 이는 스트리밍 애플리케이션이 VPC에서 액세스할 수 있는 IP 주소 범위를 확장하거나 수정해야 하는 경우에 유용합니다.

CIDR 블록을 업데이트하려면 UpdateStreamGroup API를 사용합니다.

```
aws gameliftstreams update-stream-group \
  --identifier sg-1AB2C3De4 \
  --location-configurations '[{
    "LocationName": "us-west-2",
    "VpcTransitConfiguration": {
      "VpcId": "vpc-0123456789abcdef0",
      "Ipv4CidrBlocks": ["10.0.0.0/16", "10.2.0.0/16"]
    }
  }]'
```

CIDR 블록을 업데이트한 후 Amazon GameLift Streams는 서비스 관리형 VPC에서 라우팅 구성을 자동으로 업데이트합니다.

Note

CIDR 블록을 업데이트할 때는 VPC ID를 변경할 수 없습니다. 다른 VPC에 연결하려면 스트림 그룹 위치(기본이 아닌 스트리밍 위치의 경우)를 삭제하고 다시 생성하거나 새 스트림 그룹(기본 위치의 경우)을 생성해야 합니다.

연결 확인

VPC 연결이 올바르게 작동하는지 확인하려면:

1. 스트림 그룹을 사용하여 스트림 세션을 시작합니다.
2. 스트리밍 애플리케이션 내에서 프라이빗 IP 주소를 사용하여 VPC의 리소스에 연결합니다.
3. 연결이 성공하고 데이터를 교환할 수 있는지 확인합니다.

연결에 실패하면 다음을 확인합니다.

- 전송 게이트웨이 연결이 available 상태입니다.

- VPC 라우팅 테이블과 전송 게이트웨이 라우팅 테이블 모두에서 경로가 올바르게 구성되어 있습니다.
- 보안 그룹은 서비스 VPC CIDR 블록의 인바운드 트래픽을 허용합니다.
- 네트워크 ACLs(사용된 경우)은 필요한 트래픽을 허용합니다.

Amazon GameLift Streams 백엔드 서비스 및 웹 클라이언트

Amazon GameLift Streams를 사용하면 웹 브라우저를 통해 애플리케이션을 스트리밍할 수 있습니다. Amazon GameLift Streams 웹 SDK를 사용하여 백엔드 스트리밍 서비스를 설정할 수 있습니다. 그런 다음 최종 사용자는 웹 클라이언트를 통해 스트림에 연결합니다. 클라우드를 통해 게임을 플레이하거나 애플리케이션과 상호 작용할 수 있습니다.

Amazon GameLift Streams 웹 SDK에는 백엔드 서비스 생성을 시작하는 데 사용할 수 있는 샘플 백엔드 서버와 샘플 웹 클라이언트가 포함되어 있습니다. 또한 이러한 샘플을 사용하여 추가 개발 없이 Amazon GameLift Streams가 스트리밍되는 방식을 테스트할 수 있습니다. 시작하려면 단원을 참조하십시오 [Amazon GameLift Streams를 사용하여 웹 서버 및 클라이언트 설정](#).

주제

- [지원되는 브라우저 및 입력](#)
- [필수 포트](#)
- [Amazon GameLift Streams를 사용하여 웹 서버 및 클라이언트 설정](#)
- [스트림 모양 사용자 지정](#)
- [로컬 기본 설정](#)
- [마우스 이동 처리](#)
- [애플리케이션과 웹 클라이언트 간의 데이터 채널 통신](#)

지원되는 브라우저 및 입력

다음은 Amazon GameLift Streams 스트림 및 호환되는 입력 주변 장치를 보는 데 지원되는 플랫폼 및 브라우저입니다. 브라우저는 H.264라고도 하는 고급 비디오 코딩(AVC)과도 호환되어야 합니다.

전반적으로 최상의 최종 사용자 경험과 특히 게임 컨트롤러와의 호환성을 극대화하려면 Google Chrome, Microsoft Edge 또는 사용자 지정 Chromium 기반 데스크톱 애플리케이션을 사용하는 것이 좋습니다.

어떤 컨트롤러가 어떤 브라우저와 호환되는지에 대한 자세한 내용은 [Web Gamepad API](#)를 참조하세요. Amazon GameLift Streams에는 일부 지침이 적용되지 않을 수 있지만 대부분의 게임 컨트롤러가 Bluetooth를 통해 성공적으로 연결될 것으로 예상됩니다.

운영 체제	브라우저	Input
Windows	Chrome, Edge	키보드, 마우스, 마이크, 게임 컨트롤러(햅틱 피드백 포함)
	Firefox	키보드, 마우스, 마이크, 게임 컨트롤러
Mac	Chrome, Edge, Safari	키보드, 마우스, 마이크, 게임 컨트롤러(블루투스 모드)(햅틱 피드백 포함)
	Firefox	키보드, 마우스, 마이크
Linux	Chrome, Edge, Firefox	키보드, 마우스
Android	Chrome, Edge	간단한 touch-to-mouse 에뮬레이션, 마이크, 외부 물리적 마우스, 키보드 및 게임 컨트롤러(블루투스 모드)
iOS	Chrome, Edge, Firefox, Safari	간단한 touch-to-mouse 에뮬레이션, 마이크, 외부 물리적 마우스, 키보드 및 게임 컨트롤러(블루투스 모드)

알려진 문제

다음은 브라우저 및 입력과 관련하여 알려진 문제입니다.

- Safari를 누를 때마다 즉시 전체 화면을 종료Esc합니다. 재정의할 수 없습니다.

- LinkedIn, Yelp, Instagram 등과 같은 모바일 앱 내부와 같은 “임베디드” 또는 “앱 내” 브라우저 뷰는 iOS에서 지원되지 않습니다. 이는 실시간 대화형 스트리밍에 필요한 브라우저 WebRTC 지원을 비활성화하는 경향이 있습니다. 비표준 브라우저 문자열을 감지하고 사용자에게 Safari에서 열라는 메시지를 표시하는 것이 좋습니다.
- 애플리케이션의 화면 해상도가 1080p로 설정되지 않은 경우 마우스 추적이 영향을 받을 수 있습니다. 가능하면 다른 해상도 선택을 비활성화하는 것이 좋습니다. 윈도우 모드는 비활성화하고 전체 화면에서만 실행하는 것이 좋습니다.
- 기본 Linux 애플리케이션에서 게임 컨트롤러를 지원하지 않음에도 불구하고 Proton에서 게임 컨트롤러의 플러그 앤 플레이를 지원하기 위해 Proton 런타임 환경에서 실행되는 게임은 클라이언트에 연결된 게임 컨트롤러가 없는 경우에도 항상 연결된 상태로 표시됩니다. 이는 컨트롤러가 유휴 상태이고 사용되지 않은 경우에도 컨트롤러 입력을 묻는 게임의 경우 문제가 될 수 있습니다. 게임은 마지막 입력 방법을 기반으로 입력 UI를 표시하는 것이 좋습니다.

제한 사항

- Ubuntu 22.04 LTS를 제외한 대부분의 런타임 환경은 게임 컨트롤러를 지원합니다. 게임 컨트롤러 지원이 필요한 경우 다른 런타임 환경을 사용하여 게임을 생성하는 것이 좋습니다. 다른 런타임 환경 목록은 [섹션을 참조하세요](#) [런타임 환경](#).
- PlayStation 5 및 Luna 게임 컨트롤러는 Firefox에서 지원되지 않습니다.
- 햅틱 피드백 지원:
 - PlayStation 4 및 Xbox 시리즈 S/X 컨트롤러에 대한 햅틱 피드백은 Chrome, Edge 및 Safari에서 지원됩니다.
 - PlayStation 5 DualSense 컨트롤러의 햅틱은 Safari 브라우저에서만 지원됩니다.
 - Firefox는 컨트롤러에 대한 햅틱 피드백을 지원하지 않습니다.
 - Android 및 iOS 디바이스는 컨트롤러에 대한 햅틱 피드백을 지원하지 않습니다.
- Amazon GameLift Streams 콘솔의 스트림 테스트 기능은 마이크를 지원하지 않습니다.

IPv6 지원

IPv6-only 클라이언트로의 스트리밍은 Windows 런타임 애플리케이션에서만 지원됩니다.

런타임	IPv4를 통한 스트리밍	IPv6를 통한 스트리밍
Microsoft Windows Server 2022 Base	예	예
Ubuntu 22.04 LTS	예	아니요
Proton 런타임	예	아니요

필수 포트

Amazon GameLift Streams를 통합하려면 네트워크 인프라에 필요한 포트가 열려 있고 액세스할 수 있는지 확인합니다. 다음은 Amazon GameLift Streams와 통신하기 위해 네트워크에서 열려야 하는 포트 목록입니다.

포트	프로토콜	용도
443	(HTTPS) TCP	Amazon GameLift Streams를 포함한 AWS APIs
33435-33465	UDP	웹 RTC

Amazon GameLift Streams를 사용하여 웹 서버 및 클라이언트 설정

이 자습서에서는 Amazon GameLift Streams의 스트리밍 서비스를 통합하는 웹 클라이언트 애플리케이션을 설정합니다. 그런 다음 Amazon GameLift Streams 웹 SDK, JavaScript 라이브러리 및 시작할 수 있는 샘플 코드를 사용합니다. 샘플 코드에는 간단한 Amazon GameLift Streams 백엔드 웹 서버와 간단한 웹 클라이언트가 포함되어 있습니다. 이 자습서를 마치면 샘플 코드를 사용하여 스트림을 시작할 수 있습니다.

Amazon GameLift Streams를 처음 사용하는 경우 [Amazon GameLift Streams에서 첫 번째 스트림 시작](#) 자습서부터 시작하는 것이 좋습니다. 자습서에서는 Amazon S3에 게임을 업로드하고 브라우저의 Amazon GameLift Streams 콘솔 내에서 스트리밍을 테스트하는 방법을 안내합니다.

사전 조건

- 프로그래밍 방식 액세스를 위한 적절한 자격 증명이 있는 AWS 계정입니다. 자세한 내용은 [Amazon GameLift Streams를 개발자로 설정](#) 단원을 참조하십시오.
- AWS SDK.
- Amazon GameLift Streams 지원 웹 브라우저 - 섹션을 참조하세요 [지원되는 브라우저 및 입력](#).
- Node.js - [Node.js 다운로드 페이지를 참조하세요](#).

웹 SDK 다운로드

이 자습서에서는 [제품 시작하기 페이지의 리소스 섹션에서](#) 다음 자료를 다운로드해야 합니다.

- Amazon GameLift Streams 웹 SDK 번들: 여기에는 간단한 백엔드 서비스 및 웹 클라이언트에 대한 샘플 코드가 포함됩니다.
- Amazon GameLift Streams Web SDK API 참조: 이 API 참조는 JavaScript용 Amazon GameLift Streams API 래퍼를 문서화합니다.

스트리밍 리소스 설정

스트림을 시작하려면 애플리케이션과 스트림 그룹인 스트림 리소스가 있어야 합니다. 특히 다음이 있어야 합니다.

- 준비 상태인 애플리케이션입니다.
- 사용 가능한 스트림 용량이 있는 활성 상태의 스트림 그룹입니다.
- 기본 위치가 아닌 다른 위치에서 스트리밍하려면 애플리케이션이 해당 위치로 복제를 완료했어야 합니다.

Amazon GameLift Streams 콘솔 또는 Amazon GameLift Streams CLI를 사용하여 애플리케이션과 스트림 그룹을 설정하려면 [Amazon GameLift Streams 스트림 그룹을 사용하여 스트리밍 관리](#) 각각 [Amazon GameLift Streams에서 애플리케이션 준비](#) 및 섹션을 참조하세요. 또는 Amazon GameLift Streams 콘솔의 end-to-end 연습은 섹션을 참조하세요 [Amazon GameLift Streams에서 첫 번째 스트림 시작](#).

백엔드 서버 설정

백엔드 서버는 최종 사용자를 대신하여 사용자 인증, 스트림 파라미터 구성, Amazon GameLift Streams 서비스 API 호출 수행과 같은 작업을 처리합니다. 이 설정에 대해 자세히 알아보려면 샘플 코드와 Amazon GameLift Streams Web SDK API 참조를 검토하세요. 특히 Amazon GameLift Streams Web SDK 패키지의 `server.js` 파일을 참조하세요.

Important

이 코드는 테스트 및 평가 목적으로만 사용되는 예제 코드이며 프로덕션 용량에 사용해서는 안 됩니다.

샘플 백엔드 서비스를 실행하려면

1. 터미널 또는 명령 프롬프트를 열고 폴더로 이동합니다 `AmazonGameLiftStreamsWebSDK\GameLiftStreamsSampleGamePublisherService\`.
2. 다음 명령을 실행합니다.

```
npm install
node server.js
```

샘플 백엔드 서비스를 실행하면 최종 사용자가 웹 클라이언트를 통해 스트림에 연결할 수 있습니다. 다음 단계에서 웹 클라이언트를 테스트합니다.

웹 클라이언트 시작

웹 클라이언트 애플리케이션은 Amazon GameLift Streams 스트림을 수신 및 디코딩하고, 최종 사용자에게 스트리밍하고, 최종 사용자가 애플리케이션에 참여할 수 있도록 웹 브라우저 UI를 제공하는 역할을 합니다. JavaScript Amazon GameLift Streams Web SDK를 자체 웹 클라이언트 애플리케이션에 통합하는 방법에 대해 자세히 알아보려면 샘플 코드와 Amazon GameLift Streams Web SDK API 참조를 검토하세요. 특히 Amazon GameLift Streams 웹 SDK 패키지 `public/index.html`의 섹션을 참조하세요. 브라우저에서 웹 클라이언트를 시작할 때 웹 페이지 소스를 볼 수도 있습니다.

Note

Amazon GameLift Streams의 Windows 런타임은 IPv4 또는 IPv6를 통한 스트림 세션을 지원합니다. 그러나 Linux 및 Proton 런타임 환경은 IPv4를 통한 스트리밍만 지원합니다.

웹 클라이언트 애플리케이션을 시작하려면

1. 웹 브라우저를 열고 로 이동합니다 `http://localhost:port/`. 포트 번호는 백엔드 서버에서 설정합니다. 기본적으로 HTTP 포트 8000입니다.
2. 게임을 플레이하거나 소프트웨어를 사용합니다.
 - a. 마우스와 같은 입력을 연결하려면 입력 연결을 선택합니다.
 - b. 게임을 종료하려면 Esc 키를 선택합니다.
 - c. 서버 프로세스를 중지하려면 Ctrl+C 키를 선택합니다.

스트리밍 리소스 정리

Warning

스트림 그룹에 스트리밍 용량이 할당되면 해당 용량이 사용되지 않더라도 비용이 발생합니다. 불필요한 비용을 방지하려면 스트림 그룹을 필요한 크기로 조정하세요. 개발 중에는 사용하지 않을 때 스트림 그룹의 상시 가동 용량과 대상 유휴 용량을 0으로 조정하는 것이 좋습니다. 자세한 정보는 [스트림 그룹을 0 용량으로 확장](#) 섹션을 참조하세요.

자습서를 완료하고 더 이상 애플리케이션을 스트리밍할 필요가 없는 경우 다음 단계에 따라 Amazon GameLift Streams 리소스를 정리합니다.

스트림 그룹 삭제

스트림 그룹을 삭제하면 Amazon GameLift Streams가 모든 스트림 용량을 해제합니다.

Amazon GameLift Streams 콘솔을 사용하여 스트림 그룹을 삭제하려면

1. 에 로그인 AWS Management Console 하고 [Amazon GameLift Streams 콘솔](#)을 엽니다.
2. 기존 스트림 그룹 목록을 보려면 탐색 창에서 스트림 그룹을 선택합니다.

3. 삭제할 스트림 그룹의 이름을 선택합니다.
4. 스트림 그룹 세부 정보 페이지에서 삭제를 선택합니다.
5. 삭제 대화 상자에서 삭제 작업을 확인합니다.

Amazon GameLift Streams는 컴퓨팅 리소스를 해제하고 스트림 그룹을 삭제하기 시작합니다. 이 시간 동안 스트림 그룹은 삭제 중 상태입니다. Amazon GameLift Streams가 스트림 그룹을 삭제한 후에는 더 이상 검색할 수 없습니다.

애플리케이션 삭제

다음 조건을 충족하는 애플리케이션만 삭제할 수 있습니다.

- 애플리케이션이 준비 완료 또는 오류 상태입니다.
- 진행 중인 스트림 세션에서는 애플리케이션이 스트리밍되지 않습니다. 클라이언트가 스트림 세션을 종료할 때까지 기다리거나 Amazon GameLift Streams API에서 [TerminateStreamSession](#)을 직접적으로 호출하여 스트림을 종료해야 합니다.

애플리케이션이 스트림 그룹에 연결된 경우 애플리케이션을 삭제하기 전에 모든 관련 스트림 그룹에서 연결을 해제해야 합니다. 콘솔에서 이 프로세스를 안내하는 대화 상자가 나타납니다.

Amazon GameLift Streams 콘솔을 사용하여 애플리케이션을 삭제하려면

1. 에 로그인 AWS Management Console 하고 [Amazon GameLift Streams 콘솔](#)을 엽니다.
2. 탐색 모음에서 애플리케이션을 선택하여 기존 애플리케이션 목록을 봅니다. 삭제할 애플리케이션을 선택합니다.
3. 애플리케이션 세부 정보 페이지에서 삭제를 선택합니다.
4. 삭제 대화 상자에서 삭제 작업을 확인합니다.

Amazon GameLift Streams가 애플리케이션 삭제를 시작합니다. 이 시간 동안 애플리케이션은 Deleting 상태입니다. Amazon GameLift Streams가 애플리케이션을 삭제한 후에는 더 이상 검색할 수 없습니다.

스트림 모양 사용자 지정

화면 로드

고객이 스트림을 보기 위해 웹 브라우저를 열면 웹 클라이언트가 Amazon GameLift Streams 스트림 세션에 대한 연결을 설정하기 시작합니다. 스트림 세션이 로드되는 동안 사용자 지정 배경과 로고를 고객 화면에 표시할 수 있습니다.

GameLiftStreamsSampleGamePublisherService/public/LoadingScreen/loadingscreen.js 파일의 Amazon GameLift Streams Web SDK 샘플 클라이언트는 프런트 엔드 웹 클라이언트에서 애니메이션 로고를 구현하는 방법을 보여줍니다. 기본 로드 화면은 배경과 전경이라는 2개의 이미지로 구성됩니다. 포그라운드 이미지는 중간에 위치하며 펄스 애니메이션이 있습니다. 애니메이션은 스트림 세션이 연결되어 있는 동안에만 재생됩니다.

로딩 화면을 활성화하려면

1. Amazon GameLift Streams Web SDK 샘플 클라이언트에서 GameLiftStreamsSampleGamePublisherService/public/LoadingScreen/ 폴더로 이동합니다.
2. 기본 이름인 `Background.png` 및 전경 이미지를 추가합니다. `LoadingLogo.png`. 이름을 바꾸거나 다른 이미지 형식을 사용하려면에서 코드를 업데이트해야 합니다. GameLiftStreamsSampleGamePublisherService/public/loadingscreen.js.
3. (선택 사항)에서 JavaScript 코드를 GameLiftStreamsSampleGamePublisherService/public/loadingscreen.js 업데이트하여 다양한 애니메이션을 구현합니다.

로캘 기본 설정

Amazon GameLift Streams에서 스트림당 로캘 기본 설정을 지정할 수 있습니다. 이는 애플리케이션이 시간 또는 통화와 같은 최종 사용자의 운영 체제에서 위치별 정보를 검색하는 경우에 유용합니다.

Amazon GameLift Streams는 다음 언어를 지원합니다.

값	설명
en_US	미국 영어(기본값)

값	설명
ja_jp.UTF-8	일본어

로캘 설정을 변경하려면

Amazon GameLift Streams API를 사용하여 [StartStreamSession](#)을 호출할 때 `LANG=<Language>`에 추가합니다 `AdditionalEnvironmentVariables`. 로캘 기본 설정은 사용자 별로 고유하므로 스트림 세션 수준에서 설정합니다. 이를 설정하지 않으면 스트림은 기본적으로 미국 영어를 사용합니다.

Example예제

```
aws gameliftstreams start-stream-session \
  --identifier arn:aws:gameliftstreams:us-west-2:123456789012:streamgroup/1AB2C3De4 \
  --protocol WebRTC \
  --signal-request "[webrtc-ice-offer json string]" \
  --user-id xnshijwh \
  --additional-environment-variables '{"LANG": "ja_JP.UTF-8"}'
```

마우스 이동 처리

마우스 이동 처리는 스트리밍된 애플리케이션에서 응답성이 뛰어나고 직관적인 사용자 경험을 제공하는 데 매우 중요합니다. Amazon GameLift Streams는 애플리케이션의 커서 동작에 따라 마우스 입력 전송을 자동으로 최적화하여 커서가 숨겨져 있든 보이지 않든 마우스 움직임이 자연스럽게 느껴지도록 합니다. Amazon GameLift Streams가 마우스 이벤트를 처리하는 방법을 이해하면 스트리밍 서비스와 원활하게 작동하고 최상의 사용자 경험을 제공하는 애플리케이션을 설계하는 데 도움이 됩니다.

마우스 입력 모드

Amazon GameLift Streams는 마우스 이벤트를 애플리케이션에 전송하기 위해 두 가지 고유한 모드를 사용하며, 커서 가시성에 따라 적절한 모드를 자동으로 선택합니다.

상대 모드

상대 모드에서 마우스 업데이트는 이전 위치와의 작은 증분 차이로 전송됩니다. 이 모드는 3D 방향을 사용하는 1인칭 슈팅 게임(FPS) 또는 인터페이스와 같이 정확하고 지속적인 마우스 움직임 추적이 필요한 애플리케이션에 적합합니다. Amazon GameLift Streams는 운영 체제 커서가 숨겨지거나 완전히 투명한 경우 상대 모드를 사용합니다.

절대 모드

절대 모드에서 마우스 커서 위치는 정확한 화면 좌표로 전송됩니다. 이 모드는 point-and-click 게임 또는 클릭 가능한 요소가 있는 UI와 같이 정확한 커서 위치 지정에 의존하는 애플리케이션에 적합합니다. Amazon GameLift Streams는 애플리케이션에 사용자 지정 커서 이미지가 표시되더라도 운영 체제 커서가 표시될 때 절대 모드를 사용합니다.

이 자동 선택은 수동 구성 없이도 다양한 애플리케이션 유형에 최적의 성능을 보장합니다.

포인터 잠금

포인터 잠금은 특정 요소 내에서 마우스 커서를 캡처하여 커서를 숨기고 지정된 영역을 벗어나지 못하게 하는 웹 API 기능입니다. 이 기능은 눈에 보이는 커서의 방해나 창 가장자리 도달 제한 없이 카메라 제어 또는 조준을 위해 무제한 마우스 이동이 필요한 게임에 특히 유용합니다.

Amazon GameLift Streams는 웹 SDK `InputConfiguration` 인터페이스의 `autoPointerLock` 속성을 통해 자동 포인터 잠금 기능을 제공합니다. 이 기능은 [requestPointerLock API](#)와 통합되어 직관적이고 컨텍스트를 인식하는 마우스 캡처를 제공합니다.

자동 포인터 잠금 동작

Amazon GameLift Streams는 애플리케이션이 전체 화면이고 스트림 호스트에서 원격 커서가 보이지 않을 때 포인터 잠금을 자동으로 활성화합니다. 이 동작은 일반적인 게임 개발 패턴과 잘 일치합니다.

- FPS/TPS 게임 및 3D 방향 제어 - 포인터가 자동으로 잠기고 커서가 숨겨져 FPS 게임 플레이에 필수적인 무제한 카메라 제어를 제공합니다.
- Point-and-click 게임 및 UI 제어 - 게임에서 메뉴 상호 작용 또는 전략 게임 플레이에 커서를 표시하면 포인터가 계속 표시되고 잠금 해제되어 의도한 사용자 경험을 유지합니다.

구성 옵션

`autoPointerLock` 속성은 다음 값을 허용합니다.

`true`

마우스는 원격 커서가 보이지 않을 때 항상 캡처됩니다.

`false`

마우스는 커서 가시성에 관계없이 캡처되지 않습니다.

'fullscreen'(기본값)

마우스는 비디오 요소가 전체 화면 모드이고 원격 커서가 보이지 않는 경우에만 캡처됩니다.

Important

autoPointerLock는 플랫폼 제한으로 인해 Safari 브라우저 또는 iOS 플랫폼에 영향을 주지 않습니다.

모범 사례

스트리밍된 애플리케이션에서 최적의 마우스 처리를 보장하려면:

- 항상 전체 화면 스트리밍 - 서비스에서 제대로 작동하려면 애플리케이션이 이미 전체 화면 모드로 실행 중이어야 합니다. 또한 브라우저 지원을 사용하여 스트림을 최상의 최종 사용자 경험을 위한 전체 화면 요소로 만드는 것이 좋습니다. 이렇게 하면 시스템 커서와 소프트웨어 커서 간의 정렬 문제와 같은 문제를 방지하는 데 도움이 됩니다.
- 상대 모션에 대한 커서 숨기기 - 애플리케이션에서 상대 마우스 모션(예: FPS 스타일 카메라 제어 또는 드래그 기반 상호 작용)이 필요한 경우 이러한 상호 작용 중에 운영 체제 커서를 숨깁니다. 일부 시나리오에서는 마우스 아래로 커서를 숨기고 마우스 위로 커서를 다시 표시해야 할 수 있습니다.
- 절대 위치 지정을 위한 커서 표시 - 애플리케이션에 UI 상호 작용을 위한 정확한 커서 위치가 필요한 경우 절대 좌표 모드를 활성화하기 위해 운영 체제 커서가 계속 표시되는지 확인합니다.
- 다양한 입력 시나리오 테스트 - Amazon GameLift Streams가 커서 가시성 변경에 따라 모드 간에 전환할 수 있으므로 애플리케이션이 상대적 마우스 모드와 절대 마우스 모드를 모두 올바르게 처리하는지 확인합니다.
- 다양한 창 모드 테스트 - 해당하는 경우 창 모드와 전체 화면 모드 모두에서 애플리케이션의 마우스 처리를 테스트합니다. 입력 구성에 가장 적합한 autoPointerLock 설정을 결정합니다.

애플리케이션과 웹 클라이언트 간의 데이터 채널 통신

데이터 채널을 사용하면 Amazon GameLift Streams 애플리케이션과 웹 클라이언트(최종 사용자의 웹 브라우저에서 실행되는 JavaScript 코드) 간에 임의의 메시지를 안전하게 전달할 수 있습니다. 이를 통해 최종 사용자는 스트림을 보고 있는 웹 브라우저를 통해 Amazon GameLift Streams가 스트리밍하는 애플리케이션과 상호 작용할 수 있습니다.

다음은 Amazon GameLift Streams의 데이터 채널 사용 사례의 예입니다.

- 사용자는 로컬 브라우저에서 애플리케이션에서 URLs 열 수 있습니다.
- 사용자는 클립보드의 콘텐츠를 애플리케이션에 앞뒤로 전달할 수 있습니다.
- 사용자는 로컬 시스템에서 애플리케이션으로 콘텐츠를 업로드할 수 있습니다.
- 개발자는 애플리케이션에 명령을 보내는 브라우저에서 UI를 구현할 수 있습니다.
- 사용자는 스키마를 전달하여 시각화 계층의 표시를 제어할 수 있습니다.

Features

메시지 크기 제한

Amazon GameLift Streams Web SDK는 메시지당 최대 64KB(65536바이트)의 크기 제한을 적용합니다. 이렇게 하면 메시지 크기 제한이 대부분의 브라우저와 호환되고 통신이 스트림의 총 대역폭에 미치는 영향이 낮아집니다.

지표

스트림 세션이 종료되면 데이터 채널 사용량에 대한 지표가 AWS 계정으로 전송됩니다. 자세한 내용은 Amazon GameLift 스트림 모니터링 섹션 [데이터 채널](#)의 섹션을 참조하세요. GameLift

데이터 채널 사용

Amazon GameLift Streams Web SDK는 메시지를 애플리케이션에 바이트 배열로 전송하는 `sendApplicationMessage` 함수를 제공합니다. 메시지는 `clientConnection.applicationMessage` 사용자가 정의한 콜백 함수에 의해 처리됩니다.

애플리케이션이 데이터 채널 포트에 연결되기 전에 클라이언트가 메시지를 보내면 메시지가 대기열에 추가됩니다. 그런 다음 애플리케이션이 연결되면 메시지를 수신합니다. 그러나 클라이언트가 데이터 채널 포트에 연결하기 전에 애플리케이션이 메시지를 보내면 메시지가 손실됩니다. 애플리케이션은 메시지를 보내기 전에 클라이언트의 연결 상태를 확인해야 합니다.

클라이언트 측에서

웹 클라이언트 애플리케이션에 다음 코드를 작성합니다.

1. 애플리케이션에서 수신 메시지를 수신할 콜백 함수를 정의합니다.

```
function streamApplicationMessageCallback(message) {
    console.log('Received ' + message.length + ' bytes of message from
    Application');
}
```

- 콜백 함수 `clientConnection.applicationMessage`로 설정합니다.

```
clientConnection: {
    connectionState: streamConnectionStateCallback,
    channelError: streamChannelErrorCallback,
    serverDisconnect: streamServerDisconnectCallback,
    applicationMessage: streamApplicationMessageCallback,
}
```

- `GameLiftStreams.sendApplicationMessage` 함수를 호출하여 애플리케이션에 메시지를 보냅니다. 스트림 세션이 활성 상태이고 입력이 연결되어 있는 한 언제든지 이를 호출할 수 있습니다.

예를 들어 클라이언트 측에서 간단한 데이터 채널을 설정하는 방법을 보여주는 Amazon GameLift Streams Web SDK 샘플 클라이언트를 참조하세요.

애플리케이션 측에서

애플리케이션에 다음 로직을 작성합니다.

1단계. 데이터 채널 포트에 연결

애플리케이션이 시작되면 40712의 포트에 연결합니다localhost. 애플리케이션은 전체 실행 기간 동안이 연결을 유지해야 합니다. 애플리케이션이 연결을 닫으면 다시 열 수 없습니다.

2단계. 이벤트 수신 대기

이벤트는 고정 크기 헤더로 시작한 다음 가변 길이 관련 데이터로 시작합니다. 애플리케이션이 이벤트를 수신하면 이벤트를 구문 분석하여 정보를 검색합니다.

이벤트 형식

- 헤더: 양식의 4바이트 헤더 abcc
 - a : 클라이언트 ID 바이트입니다. 이는 여러 연결의 경우(연결 해제 및 재연결로 인해) 특정 클라이언트 연결을 식별합니다.

- **b** : 이벤트 유형 바이트. 0 - 클라이언트 연결됨, 1 - 클라이언트 연결 해제됨, 2 - 클라이언트에서 메시지가 전송됩니다. 향후 Amazon GameLift Streams 서비스 업데이트 시 다른 이벤트 유형이 수신될 수 있으므로 무시해야 합니다.
- **cc** : 연결된 이벤트 데이터의 길이입니다. 이는 빅 엔디안 순서가 있는 2바이트로 표시됩니다(첫 번째 바이트가 가장 중요함). 이벤트 유형이 2인 경우 이벤트 데이터는 클라이언트의 메시지 콘텐츠를 나타냅니다.
- **데이터**: 나머지 바이트에는 클라이언트 메시지와 같은 이벤트 데이터가 포함됩니다. 데이터 길이는 헤더 **cc**에 로 표시됩니다.

이벤트를 수신하려면

1. 네 개의 헤더 바이트를 읽고 이벤트 데이터의 클라이언트 ID, 이벤트 유형 및 길이를 검색합니다.
2. 헤더에 설명된 길이에 따라 클라이언트 ID 및 이벤트 유형에 관계없이 가변 길이 이벤트 데이터를 읽습니다. 이벤트 데이터가 버퍼에 남아 다음 이벤트 헤더와 혼동되지 않도록 데이터를 조건 없이 읽는 것이 중요합니다. 이벤트 유형에 따라 데이터 길이를 가정하지 마십시오.
3. 애플리케이션에서 인식하는 경우 이벤트 유형에 따라 적절한 조치를 취합니다. 이 작업에는 수신 연결 또는 연결 해제 로깅 또는 클라이언트 메시지 구문 분석과 애플리케이션 로직 트리거가 포함될 수 있습니다.

3단계. 클라이언트로 메시지 전송

애플리케이션은 수신 이벤트에서 사용하는 것과 동일한 4바이트 헤더 형식의 메시지를 전송해야 합니다.

클라이언트에 메시지를 전송하려면

1. 다음 속성을 사용하여 헤더를 작성합니다.
 - a : 클라이언트 ID 바이트입니다. 메시지가 클라이언트 메시지에 대한 응답인 경우 이전 클라이언트 연결에서 새로 다시 연결된 클라이언트로 응답을 전달하는 것과 같은 경쟁 조건을 방지하기 위해 수신 클라이언트 메시지와 동일한 클라이언트 ID를 재사용해야 합니다. 애플리케이션이 클라이언트에 원치 않는 메시지를 보내는 경우 가장 최근의 "클라이언트 연결" 이벤트(이벤트 유형 0)와 일치하도록 클라이언트 ID를 설정해야 합니다.
 - b : 발신 메시지의 이벤트 유형은 항상 2여야 합니다. 클라이언트는 다른 이벤트 유형의 메시지를 무시합니다.
 - c : **cc** : 메시지 길이, 바이트 단위.

2. 메시지 바이트를 작성합니다.

클라이언트가 연결 해제되지 않는 한 메시지는 지정된 클라이언트로 전송됩니다. 연결이 해제된 클라이언트가 다시 연결되면 클라이언트 연결 이벤트를 통해 새 클라이언트 ID가 할당됩니다. 이전 클라이언트 ID에 대한 전송되지 않은 메시지는 모두 삭제됩니다.

Example

다음 의사 코드는 애플리케이션 측에서 메시지를 통신하는 로직을 보여줍니다. Winsock을 사용하는 전체 예제는 Windows 소켓 2 설명서의 [Winsock 클라이언트 코드 완료](#)를 참조하세요.

```
connection = connect_to_tcp_socket("localhost:40712")
loop:
    while has_pending_bytes(connection):
        client_id = read_unsigned_byte(connection)
        event_type = read_unsigned_byte(connection)
        event_length = 256 * read_unsigned_byte(connection)
        event_length = event_length + read_unsigned_byte(connection)
        event_data = read_raw_bytes(connection, event_length)
        if message_type == 0:
            app_process_client_connected(client_id)
        else if message_type == 1:
            app_process_client_disconnected(client_id)
        else if message_type == 2:
            app_process_client_message(client_id, event_data)
        else:
            log("ignoring unrecognized event type")
    while app_has_outgoing_messages():
        target_client_id, message_bytes = app_next_outgoing_message()
        message_length = length(message_bytes)
        write_unsigned_byte(connection, target_client_id)
        write_unsigned_byte(connection, 2)
        write_unsigned_byte(connection, message_length / 256)
        write_unsigned_byte(connection, message_length mod 256)
        write_raw_bytes(connection, message_bytes)
```

Amazon GameLift Streams 시작 체크리스트

Amazon GameLift Streams에서 성공적인 출시를 준비하려면 계획과 조정이 필요합니다. 이 세부 체크리스트를 따라 이벤트까지 몇 주 동안 원활한 경험을 보장합니다.

Amazon GameLift Streams 팀에 알림

조치: 최소 8주 전에 기술 계정 관리자, 계정 팀 또는 계정 솔루션 아키텍트에게 시작 타임라인과 예상되는 피크 동시 스트림에 대해 알립니다.

이유: 프로덕션 워크로드의 규모를 이해하면 서비스 제한이 적절한지 확인하고 필요한 경우 조정할 수 있습니다. 또한 시작을 위한 용량 가용성 및 권장 사항에 대한 지침도 제공합니다.

호환성 및 성능 테스트

작업: 용량이 있는 모든 위치에서 애플리케이션을 대규모로 테스트하여 긍정적인 고객 경험을 확인합니다. Amazon GameLift Streams는 다양한 수준의 성능과 런타임이 지원되는 NVIDIA 기반 스트림 클래스를 제공합니다.

이유: 철저한 테스트를 통해 출시 전에 잠재적 호환성 및 성능 문제를 식별하고 해결할 수 있습니다. 스트림 클래스에 대해 다음 사항에 유의하세요.

- "높음" 스트림 클래스는 다중 테넌시를 지원하므로 단일 인스턴스에서 두 애플리케이션을 동시에 실행할 수 있습니다. "높음" 스트림 클래스를 사용하는 경우 최소 2개의 동시 스트림으로 테스트하여 CPU, GPU 및 메모리와 같은 공유 리소스에서 애플리케이션이 어떻게 작동하는지 확인합니다.

용량 예약

조치: 시작 최소 8주 전에 계정 팀에 문의하여 용량을 예약하세요. 특히 중요한 대규모 요구가 예상되는 경우 더욱 그렇습니다. 호환성 테스트, 성능 요구 사항 및 예산을 기반으로 스트림 클래스 및 스트리밍 위치를 결정합니다. 시작/종료 시간과 필요한 용량을 제공합니다. AWS에서는 모든 용량 예약을 예약 기한 6~8주 전에 완료해야 합니다.

이유: Amazon GameLift Streams는 온디맨드 용량을 사용하여 선착순으로 작동합니다. 예약은 필요한 용량을 보장하는 데 필수적입니다.

대규모 성능 테스트

작업: APIs 및 Amazon GameLift Streams 구성에 대한 철저한 로드 테스트를 수행하여 로드 시 성능 (지연 시간, 해상도 및 프레임 속도)을 관찰합니다. 를 확인하여 시작 및 그 이후에 사용할 수 [Amazon GameLift Streams API 속도 제한](#) 있는 충분한 여유 공간이 있는지 확인합니다. 한도 증가가 필요하다 고 생각되면 계정 관리자에게 문의하거나 지원 티켓을 제출해야 합니다.

이유: 로드 테스트는 애플리케이션 및 Amazon GameLift Streams 구성이 시작 전에 스트레스 상태에서 서 어떻게 수행되는지 보여줍니다. 이는 대규모로 원활한 성능을 보장하는 데 매우 중요합니다.

사전 시작 설정

작업: 시작 최소 2~3일 전에 최종 애플리케이션 리소스 및 스트림 그룹을 생성합니다. 스트리밍 성능을 검증하고 필요에 따라 용량을 확장합니다.

이유: 이렇게 하면 모든 구성 요소가 예상대로 작동하여 예상치 못한 문제의 위험을 최소화하고 이벤트 중에 더 쉽게 진단하고 복구할 수 있습니다.

추가 팁

- 일관성이 핵심: 시작 이벤트 전체에서 동일한 기존 스트림 그룹을 사용하면 Amazon GameLift Streams 백엔드의 일관성을 유지하여 문제 해결을 간소화합니다.
- 면밀히 모니터링: 성능 및 사용자 피드백을 면밀히 모니터링하여 문제를 신속하게 해결합니다. 운영 대시보드를 구축합니다. Amazon CloudWatch를 사용하여 스트림 용량, 사용량 및 성능을 모니터링 합니다([CloudWatch를 사용하여 모니터링](#) 자세한 내용은 참조). 추가 지침은 [Well-Architected 프레임 워크](#)를 참조하세요.

추가 지원이 필요하십니까?

질문이 있거나 추가 지원이 필요한 경우 언제든지 [Amazon GameLift Streams 지원](#) 팀에 문의하세요. 를 성공적으로 원활하게 시작할 수 있도록 지원합니다.

Amazon GameLift Streams의 보안

의 클라우드 보안 AWS 이 최우선 순위입니다. AWS 고객은 보안에 가장 민감한 조직의 요구 사항을 충족하도록 구축된 데이터 센터 및 네트워크 아키텍처의 이점을 누릴 수 있습니다.

보안은 AWS 와 사용자 간의 공동 책임입니다. [공동 책임 모델](#)은 이 사항을 클라우드의 보안 및 클라우드 내 보안으로 설명합니다.

- 클라우드 보안 - AWS 는에서 AWS 서비스를 실행하는 인프라를 보호할 책임이 있습니다 AWS 클라우드. AWS 또한는 안전하게 사용할 수 있는 서비스를 제공합니다. 타사 감사자는 [AWS 규정 준수 프로그램](#) 일환으로 보안의 효과를 정기적으로 테스트하고 확인합니다. Amazon GameLift Streams에 적용되는 규정 준수 프로그램에 대한 자세한 내용은 규정 준수 [AWS 프로그램 제공 범위 내 서비스 규정 준수](#) .
- 클라우드의 보안 - 사용자의 책임은 사용하는 AWS 서비스에 따라 결정됩니다. Amazon GameLift Streams는 사용자가 제공하는 프로그램을 실행하도록 설계되었으며 해당 프로그램의 콘텐츠 및 보안에 대한 책임은 전적으로 사용자에게 있습니다. 또한 귀하는 귀사의 데이터 민감도, 귀사의 요구 사항, 관련 법률 및 규정을 비롯한 기타 요소에 대해서도 책임이 있습니다.

이 설명서는 Amazon GameLift Streams를 사용할 때 공동 책임 모델을 적용하는 방법을 이해하는 데 도움이 됩니다. 다음 주제에서는 보안 및 규정 준수 목표에 맞게 Amazon GameLift Streams를 구성하는 방법을 보여줍니다. 또한 Amazon GameLift Streams 리소스를 모니터링하고 보호하는 데 도움이 되는 다른 AWS 서비스를 사용하는 방법을 알아봅니다.

주제

- [Amazon GameLift Streams의 데이터 보호](#)
- [Amazon GameLift Streams의 ID 및 액세스 관리](#)
- [Amazon GameLift Streams에 대한 규정 준수 검증](#)
- [Amazon GameLift 스트림의 복원력](#)
- [Amazon GameLift Streams의 인프라 보안](#)
- [Amazon GameLift Streams의 구성 및 취약성 분석](#)
- [Amazon GameLift Streams의 보안 모범 사례](#)

Amazon GameLift Streams의 데이터 보호

AWS [공동 책임 모델](#) Amazon GameLift Streams의 데이터 보호에 적용됩니다. 이 모델에 설명된 대로 AWS 는 모든를 실행하는 글로벌 인프라를 보호할 책임이 있습니다 AWS 클라우드. 사용자는 이 인프라에 호스팅되는 콘텐츠에 대한 통제 권한을 유지할 책임이 있습니다. 사용하는 AWS 서비스 의 보안 구성과 관리 태스크에 대한 책임도 사용자에게 있습니다. 데이터 프라이버시에 관한 자세한 내용은 [데이터 프라이버시 FAQ](#)를 참조하세요. 유럽의 데이터 보호에 대한 자세한 내용은 AWS 보안 블로그의 [AWS 공동 책임 모델 및 GDPR](#) 블로그 게시물을 참조하세요.

데이터 보호를 위해 자격 증명을 보호하고 AWS 계정 AWS IAM Identity Center 또는 AWS Identity and Access Management (IAM)를 사용하여 개별 사용자를 설정하는 것이 좋습니다. 이렇게 하면 개별 사용자에게 자신의 직무를 충실히 이행하는 데 필요한 권한만 부여됩니다. 또한 다음과 같은 방법으로 데이터를 보호하는 것이 좋습니다.

- 각 계정에 다중 인증(MFA)을 사용합니다.
- SSL/TLS를 사용하여 AWS 리소스와 통신합니다. TLS 1.2는 필수이며 TLS 1.3을 권장합니다.
- 를 사용하여 API 및 사용자 활동 로깅을 설정합니다 AWS CloudTrail. CloudTrail 추적을 사용하여 AWS 활동을 캡처하는 방법에 대한 자세한 내용은 AWS CloudTrail 사용 설명서의 [CloudTrail 추적 작업을](#) 참조하세요.
- 내의 모든 기본 보안 제어와 함께 AWS 암호화 솔루션을 사용합니다 AWS 서비스.
- Amazon S3에 저장된 민감한 데이터를 검색하고 보호하는 데 도움이 되는 Amazon Macie와 같은 고급 관리형 보안 서비스를 사용합니다.
- 명령줄 인터페이스 또는 API를 AWS 통해 액세스할 때 FIPS 140-3 검증 암호화 모듈이 필요한 경우 FIPS 엔드포인트를 사용합니다. 사용 가능한 FIPS 엔드포인트에 대한 자세한 내용은 [연방 정보 처리 표준\(FIPS\) 140-3](#)을 참조하세요.

고객의 이메일 주소와 같은 기밀 정보나 중요한 정보는 태그나 이름 필드와 같은 자유 형식 텍스트 필드에 입력하지 않는 것이 좋습니다. 여기에는 Amazon GameLift Streams 또는 기타 AWS 서비스 에서 콘솔 AWS CLI, API 또는 AWS SDKs를 사용하여 작업하는 경우가 포함됩니다. 이름에 사용되는 태그 또는 자유 형식 텍스트 필드에 입력하는 모든 데이터는 청구 또는 진단 로그에 사용될 수 있습니다. 외부 서버에 URL을 제공할 때 해당 서버에 대한 요청을 검증하기 위해 자격 증명을 URL에 포함해서는 안 됩니다.

Amazon GameLift Streams는 다음과 같이 서비스별 데이터를 처리합니다.

- 고객 제공 애플리케이션 - Amazon GameLift Streams는 제공된 경우 고객 데이터를 내부 서비스 관리형 Amazon S3 버킷 및 Amazon EC2 인스턴스에 연결된 NVME 스토리지 드라이브에 저장합니다.

모든 데이터는 저장 시 서비스 관리형 암호화로 저장됩니다. 이 데이터 사본에는 고객이 직접 액세스할 수 없습니다. 애플리케이션을 삭제하려면 Amazon GameLift Streams 콘솔 또는 서비스 API를 사용합니다.

- 고객 제공 메타데이터 - 고객은 설명, 연결 정보, 고객 ID와 같은 불투명 식별자를 포함하여 Amazon GameLift Streams APIs에 메타데이터를 제공할 수 있습니다. IDs 이 메타데이터는 항상 특정 고객 리소스와 연결됩니다.
- 고객 생성 데이터 - 애플리케이션이 정상 작업의 일부로 새 데이터를 쓰는 경우 이 고객 생성 데이터는 사용자 세션이 끝날 때까지 유지됩니다. 세션이 끝나면 생성된 데이터를 고객이 선택한 Amazon S3 버킷 대상으로 선택적으로 내보낼 수 있습니다. 그렇지 않으면 고객이 생성한 데이터는 생성된 Amazon EC2 인스턴스를 벗어나지 않습니다. 데이터 처리에 대한 자세한 내용은 [세션 격리](#)에 대한 주제를 참조하세요.
- 지표 및 이벤트 데이터 - Amazon GameLift Streams 콘솔을 통해 또는 서비스 API를 호출하여 액세스할 수 있는 Amazon GameLift Streams 지표 및 이벤트 데이터입니다. 데이터는 애플리케이션, 스트림 그룹 및 스트림 세션에서 사용할 수 있습니다. 권한 있는 사용자는 Amazon CloudWatch 및 CloudWatch Events를 통해 이 데이터에 액세스할 수도 있습니다.

Important

Amazon GameLift Streams에 고객 IDs 또는 기타 식별자를 제공하는 경우 이러한 값은 익명화된 참조이며 민감한 정보나 개인 정보를 포함하지 않을 것으로 예상됩니다. Amazon GameLift Streams는 메타데이터 필드를 수정하지 않습니다.

데이터 보호에 대한 자세한 내용은 AWS 보안 블로그의 [AWS 공동 책임 모델 및 GDPR](#) 블로그 게시물을 참조하세요.

저장 시 암호화

Amazon GameLift Streams 관련 데이터의 저장 데이터 암호화는 다음과 같이 처리됩니다.

- 애플리케이션 콘텐츠는 서비스 관리형 암호화된 Amazon S3 버킷과 서비스 관리형 Amazon EC2 인스턴스에 연결된 하드웨어 암호화 NVME 드라이브에 추가로 저장됩니다.

전송 중 암호화

Amazon GameLift Streams APIs에 대한 호출은 보안(SSL) 연결을 통해 이루어지며 [AWS 서명 버전 4](#)를 사용하여 인증됩니다(AWS CLI 또는 AWS SDK를 통해 연결할 경우 서명이 자동으로 처리됨). 호출 엔터티는 Amazon GameLift Streams 리소스에 대해 정의된 IAM 액세스 정책을 적용하여 인증되는 보안 자격 증명을 사용합니다.

다중 위치 스트림 그룹의 컨텍스트에서 스트리밍 용량이 할당된 스트림 그룹의 모든 위치에서 애플리케이션을 스트리밍하기 위해 Amazon GameLift Streams는 애플리케이션을 해당 위치에 안전하게 복제합니다.

마찬가지로 Amazon GameLift Streams는 요청 시 세션 종료 시 고객 이름의 Amazon S3 버킷에 로그 데이터 및 세션 파일을 저장합니다. 버킷이 세션과 동일한 위치에 있지 않은 경우 Amazon GameLift Streams는 버킷 AWS 리전 이 위치한 로 파일을 안전하게 전송합니다.

최종 사용자 스트림 보호

개별 최종 사용자 스트림은 최종 사용자의 웹 브라우저와 Amazon GameLift Streams 백엔드 호스트 간의 직접 연결입니다. 이러한 스트림은 업계 표준 WebRTC 암호화로 보호되며 스트림의 두 엔드포인트 모두 스트림 세션 API를 통해 협상된 SignalRequest 및 SignalResponse 값의 일부인 암호화 식별자로 명확하게 식별됩니다. APIs

데이터 채널 메시지는 스트림에 사용되는 WebRTC 암호화에도 포함됩니다. 이러한 메시지는 Amazon GameLift Streams에서 복호화되며 암호화되지 않은 API를 통해 on-the-host 고객의 애플리케이션으로 로컬로 전달됩니다. Amazon GameLift Streams 서비스에서도 복호화할 수 없는 end-to-end 암호화가 필요한 경우가 추가 암호화 계층은 애플리케이션 개발자의 책임입니다.

Linux 스트림 클래스의 세션 격리

Linux 스트림 클래스(Ubuntu 및 Proton 런타임)에서 Amazon GameLift Streams는 컨테이너 격리를 사용합니다. 모든 세션은 사용 후 폐기되는 새 Linux 컨테이너에서 실행됩니다. 즉, 각 새 세션은 컴퓨팅 리소스를 공유하는 다른 사용자와 격리된 새로운 환경에서 실행됩니다(공유 리소스 스트림 클래스에서 실행되는 경우). 새 세션이 시작될 때 이전 세션의 데이터가 존재하지 않습니다.

Windows 스트림 클래스의 세션 격리

Windows 스트림 클래스(Microsoft Windows Server 런타임)에서 Amazon GameLift Streams는 소프트웨어 격리를 사용합니다. 서비스는 소프트웨어 에이전트를 사용하여 세션 간에 중요한 시스템 상태를

재설정합니다. 일부 폴더는 온호스트 디스크 캐싱과 같은 성능 최적화를 위해 여러 세션에 보존됩니다. 소프트웨어 에이전트는 이전 스트림 세션 중에 사용자의 프로파일 디렉터리에 생성된 모든 파일을 자동으로 제거합니다. 그러나 에이전트는 애플리케이션이 실행되기 전에 존재했고 애플리케이션이 실행되는 동안 수정된 파일을 제거하지 않습니다. 애플리케이션이 추가한 Windows 레지스트리 키도 제거하지 않습니다. 고객은 전체 운영 체제의 무결성이 손상되지 않도록 하는 것이 고객의 책임임을 알고 있어야 합니다. 애플리케이션은 관리자 사용자로 실행되므로 여러 세션에서 지속되는 변경 사항을 포함하여 중요한 시스템 수준 파일을 수정할 수 있습니다. 애플리케이션을 보호하고 안전하지 않거나 불안정한 운영 체제 수정이 발생하지 않도록 보호하는 것은 고객의 책임입니다.

고객은 애플리케이션이 시작될 때 수정된 파일과 이전 세션에서 추가된 레지스트리 키를 정리할 책임이 있습니다. 이는 애플리케이션이 사용자의 프로파일 디렉터리에 작성하는 기밀 또는 민감한 정보를 보호하는 중요한 단계입니다. 이를 위해 고객은 다음 작업을 수행하는 자체 사용자 지정 스크립트를 작성할 수 있습니다.

- 애플리케이션에서 수정한 %USERPROFILE% 디렉터리 외부의 모든 파일을 복원합니다.
- 애플리케이션이 추가한 모든 민감한 레지스트리 키 또는 사용자별 레지스트리 키를 정리합니다.

암호화 키 관리

서비스는 AWS관리형 암호화 키를 사용합니다. 각 리전은 별도의 KMS 키를 사용합니다. 고객 관리형 키(CMKs)는 지원되지 않습니다.

Amazon GameLift Streams에 제공된 애플리케이션 파일은 서비스에서 다시 게시하거나 내보낼 수 없습니다. 고객은 서비스 콘솔 또는 APIs를 사용하여 애플리케이션을 삭제할 수 있습니다. 이전에 이러한 애플리케이션 파일을 보관한 드라이브는 연결된 스트림 그룹을 삭제하여 완전히 제거할 수 있습니다.

인터넷워크 트래픽 개인 정보 보호

Amazon GameLift Streams는 퍼블릭 네트워크를 사용하여 스트림 세션을 호스팅합니다. 각 스트림 그룹은 다른 스트림 그룹 및 다른 고객과 격리된 하나 이상의 서비스 관리형 VPC 네트워크로 구성됩니다. 인바운드 네트워크 연결은 인증된 서비스 중개 WebRTC 스트림 연결을 제외하고 거부됩니다. 고객 애플리케이션은 이러한 VPCs에서 다른 퍼블릭 주소로 제한 없이 연결할 수 있습니다.

또한 고객이 서비스 API 호출 또는 설정만 사용하여 스트림 또는 애플리케이션 데이터를 공개적으로 액세스할 수 있는 방법은 없습니다. 모든 서비스 상호 작용은 AWS인증된 API 직접 호출로 게이트됩니다. 고객이 스트림을 퍼블릭에 액세스하도록 하려면 인증된 호출을 통해 스트림을 시작하고 표시하는 자체 클라이언트 웹 애플리케이션을 생성해야 합니다.

Amazon GameLift Streams의 ID 및 액세스 관리

AWS Identity and Access Management (IAM)는 관리자가 AWS 리소스에 대한 액세스를 안전하게 제어하는 데 도움이 되는 AWS 서비스입니다. IAM 관리자는 Amazon GameLift Streams 리소스를 사용할 수 있는 인증(로그인) 및 권한(권한 있음)을 받을 수 있는 사용자를 제어합니다. IAM은 추가 비용 없이 사용할 수 있는 AWS 서비스입니다.

주제

- [대상](#)
- [ID를 통한 인증](#)
- [정책을 사용하여 액세스 관리](#)
- [Amazon GameLift Streams가 IAM과 작동하는 방식](#)
- [Amazon GameLift Streams의 자격 증명 기반 정책 예제](#)
- [Amazon GameLift Streams 자격 증명 및 액세스 문제 해결](#)

대상

AWS Identity and Access Management (IAM)를 사용하는 방법은 역할에 따라 다릅니다.

- 서비스 사용자 - 기능에 액세스할 수 없는 경우 관리자에게 권한 요청([참조 Amazon GameLift Streams 자격 증명 및 액세스 문제 해결](#))
- 서비스 관리자 - 사용자 액세스 결정 및 권한 요청 제출([Amazon GameLift Streams가 IAM과 작동하는 방식 참조](#))
- IAM 관리자 - 액세스를 관리하기 위한 정책 작성([Amazon GameLift Streams의 자격 증명 기반 정책 예제 참조](#))

ID를 통한 인증

인증은 자격 증명 자격 증명을 AWS 사용하여 로그인하는 방법입니다. AWS 계정 루트 사용자, IAM 사용자 또는 IAM 역할을 수임하여 인증해야 합니다.

AWS IAM Identity Center (IAM Identity Center), Single Sign-On 인증 또는 Google/Facebook 자격 증명과 같은 자격 증명 소스의 자격 증명을 사용하여 페더레이션 자격 증명으로 로그인할 수 있습니다. 로그인하는 방법에 대한 자세한 내용은 AWS 로그인 사용 설명서의 [AWS 계정에 로그인하는 방법](#) 섹션을 참조하세요.

프로그래밍 방식 액세스를 위해서는 요청에 암호화 방식으로 서명할 수 있는 SDK 및 CLI를 AWS 제공합니다. 자세한 내용은 IAM 사용 설명서의 [API 요청용 AWS Signature Version 4](#) 섹션을 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 사용자 자격 증명이 필요한 작업은 IAM 사용 설명서의 [루트 사용자 자격 증명에 필요한 작업](#) 섹션을 참조하세요.

페더레이션 ID

가장 좋은 방법은 인간 사용자에게 자격 증명 공급자와의 페더레이션을 사용하여 임시 자격 증명을 AWS 서비스 사용하여 액세스하도록 요구하는 것입니다.

페더레이션 자격 증명은 엔터프라이즈 디렉터리, 웹 자격 증명 공급자 또는 자격 증명 소스의 자격 증명을 AWS 서비스 사용하여 Directory Service 에 액세스하는 사용자입니다. 페더레이션 ID는 임시 자격 증명을 제공하는 역할을 수임합니다.

중앙 집중식 액세스 관리를 위해 AWS IAM Identity Center를 추천합니다. 자세한 정보는 AWS IAM Identity Center 사용 설명서의 [What is IAM Identity Center?](#)를 참조하세요.

IAM 사용자 및 그룹

[IAM 사용자](#)는 단일 개인 또는 애플리케이션에 대한 특정 권한을 가진 ID입니다. 장기 자격 증명에 있는 IAM 사용자 대신 임시 자격 증명을 사용하는 것이 좋습니다. 자세한 내용은 IAM 사용 설명서의 [자격 증명 공급자와의 페더레이션을 사용하여 임시 자격 증명을 AWS 사용하여 액세스하도록 인간 사용자에게 요구하기](#)를 참조하세요.

[IAM 그룹](#)은 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 역할](#)은 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 방법](#)을 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 액세스](#)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. 보안 주체가 요청할 때 이러한 정책을 AWS 평가합니다. 대부분의 정책은 JSON 문서로 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.

ID 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 정의](#)를 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.

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 지정](#)해야 합니다.

리소스 기반 정책은 해당 서비스에 있는 인라인 정책입니다. 리소스 기반 정책에서는 IAM의 AWS 관리형 정책을 사용할 수 없습니다.

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 범위](#)를 참조하세요.
- 서비스 제어 정책(SCP) - AWS Organizations내 조직 또는 조직 단위에 대한 최대 권한을 지정합니다. 자세한 내용은 AWS Organizations 사용 설명서의 [서비스 제어 정책](#)을 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 사용자에게 대해 임시 세션을 프로그래밍 방식으로 생성할 때 파라미터로 전달하는 고급 정책입니다. 자세한 내용은 IAM 사용 설명서의 [세션 정책](#)을 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 로직](#)을 참조하세요.

Amazon GameLift Streams가 IAM과 작동하는 방식

IAM을 사용하여 Amazon GameLift Streams에 대한 액세스를 관리하기 전에 Amazon GameLift Streams에서 사용할 수 있는 IAM 기능에 대해 알아봅니다.

Amazon GameLift Streams와 함께 사용할 수 있는 IAM 기능

IAM 특성	Amazon GameLift Streams 지원
자격 증명 기반 정책	예
리소스 기반 정책	아니요
정책 작업	예
정책 리소스	예
정책 조건 키(서비스별)	예
ACL	아니요
ABAC(정책 내 태그)	부분. ABAC는 애플리케이션 및 스트림 그룹에만 지원됩니다.

IAM 특성	Amazon GameLift Streams 지원
임시 자격 증명	예
엔터티 권한	예
서비스 역할	아니요
서비스 연결 역할	아니요

Amazon GameLift Streams 및 기타 AWS 서비스에서 대부분의 IAM 기능을 사용하는 방법을 전체적으로 알아보려면 IAM 사용 설명서의 [AWS IAM으로 작업하는 서비스를](#) 참조하세요.

Amazon GameLift Streams에 대한 자격 증명 기반 정책

ID 기반 정책 지원: 예

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 정의](#)를 참조하세요.

IAM ID 기반 정책을 사용하면 허용되거나 거부되는 작업과 리소스뿐 아니라 작업이 허용되거나 거부되는 조건을 지정할 수 있습니다. JSON 정책에서 사용할 수 있는 모든 요소에 대해 알아보려면 IAM 사용 설명서의 [IAM JSON 정책 요소 참조](#)를 참조하세요.

Amazon GameLift Streams의 자격 증명 기반 정책 예제

Amazon GameLift Streams 자격 증명 기반 정책의 예를 보려면 섹션을 참조하세요 [Amazon GameLift Streams의 자격 증명 기반 정책 예제](#).

Amazon GameLift Streams 내의 리소스 기반 정책

리소스 기반 정책 지원: 아니요

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 정의합

니다. 리소스 기반 정책에서 [보안 주체를 지정](#)해야 합니다. 보안 주체에는 계정, 사용자, 역할, 페더레이션 사용자 또는가 포함될 수 있습니다 AWS 서비스.

교차 계정 액세스를 활성화하려는 경우, 전체 계정이나 다른 계정의 IAM 개체를 리소스 기반 정책의 보안 주체로 지정할 수 있습니다. 자세한 내용은 IAM 사용 설명서의 [IAM에서 교차 계정 리소스 액세스](#)를 참조하세요.

Amazon GameLift Streams에 대한 정책 작업

정책 작업 지원: 예

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 Action요소는 정책에서 액세스를 허용하거나 거부하는 데 사용할 수 있는 작업을 설명합니다. 연결된 작업을 수행할 수 있는 권한을 부여하기 위한 정책에 작업을 포함하세요.

Amazon GameLift Streams의 정책 작업은 작업 앞에 다음 접두사를 사용합니다.

```
gameliftstreams
```

단일 문에서 여러 작업을 지정하려면 쉼표로 구분합니다.

Example

```
"Action": [
    "gameliftstreams:action1",
    "gameliftstreams:action2"
]
```

Amazon GameLift Streams 자격 증명 기반 정책의 예를 보려면 섹션을 참조하세요 [Amazon GameLift Streams의 자격 증명 기반 정책 예제](#).

Amazon GameLift Streams에 대한 정책 리소스

정책 리소스 지원: 예

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": "*"
```

Amazon GameLift Streams 자격 증명 기반 정책의 예를 보려면 섹션을 참조하세요 [Amazon GameLift Streams의 자격 증명 기반 정책 예제](#).

Amazon GameLift Streams에 사용되는 정책 조건 키

서비스별 정책 조건 키 지원: 예

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 연산자](#)를 사용하여 정책의 조건을 요청의 값과 일치시키는 조건식을 생성할 수 있습니다. 모든 AWS 전역 조건 키를 보려면 IAM 사용 설명서의 [AWS 전역 조건 컨텍스트 키](#)를 참조하세요.

Amazon GameLift Streams 자격 증명 기반 정책의 예를 보려면 섹션을 참조하세요 [Amazon GameLift Streams의 자격 증명 기반 정책 예제](#).

Amazon GameLift Streams ACLs

ACL 지원: 아니요

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.

Amazon GameLift Streams를 사용한 ABAC

ABAC 지원(정책의 태그): 부분적

속성 기반 액세스 제어(ABAC)는 태그라고 불리는 속성을 기반으로 권한을 정의하는 권한 부여 전략입니다. IAM 엔터티 및 AWS 리소스에 태그를 연결한 다음 보안 주체의 태그가 리소스의 태그와 일치할 때 작업을 허용하는 ABAC 정책을 설계할 수 있습니다.

태그에 근거하여 액세스를 제어하려면 `aws:ResourceTag/key-name`, `aws:RequestTag/key-name` 또는 `aws:TagKeys` 조건 키를 사용하여 정책의 [조건 요소](#)에 태그 정보를 제공합니다.

서비스가 모든 리소스 유형에 대해 세 가지 조건 키를 모두 지원하는 경우, 값은 서비스에 대해 예입니다. 서비스가 일부 리소스 유형에 대해서만 세 가지 조건 키를 모두 지원하는 경우, 값은 부분적입니다.

ABAC에 대한 자세한 내용은 IAM 사용 설명서의 [ABAC 권한 부여를 통한 권한 정의](#)를 참조하세요. ABAC 설정 단계가 포함된 자습서를 보려면 IAM 사용 설명서의 [속성 기반 액세스 제어\(ABAC\) 사용](#)을 참조하세요.

Amazon GameLift Streams에서 임시 자격 증명 사용

임시 자격 증명 지원: 예

임시 자격 증명은 AWS 리소스에 대한 단기 액세스를 제공하며 페더레이션을 사용하거나 역할을 전환할 때 자동으로 생성됩니다. 장기 액세스 키를 사용하는 대신 임시 자격 증명을 동적으로 생성하는 것이 AWS 좋습니다. 자세한 내용은 IAM 사용 설명서의 [IAM의 임시 보안 자격 증명 및 IAM으로 작업하는 AWS 서비스](#) 섹션을 참조하세요.

Amazon GameLift Streams에 대한 교차 서비스 보안 주체 권한

전달 액세스 세션(FAS) 지원: 예

전달 액세스 세션(FAS)은 호출하는 보안 주체의 권한을 다운스트림 서비스에 AWS 서비스 대한 요청과 AWS 서비스 함께 사용합니다. FAS 요청 시 정책 세부 정보는 [전달 액세스 세션](#)을 참조하세요.

새 애플리케이션 리소스를 생성할 때 Amazon GameLift Streams는 호출 보안 주체의 권한을 사용하여 고객의 애플리케이션 파일이 포함된 Amazon S3 버킷에 액세스합니다. 또한 Amazon GameLift Streams는 호출 보안 주체를 검사하여 다중 위치 스트림 그룹과 같은 특정 리전 간 기능에 대한 옵트인 자격을 확인합니다.

Amazon GameLift Streams의 서비스 역할

서비스 역할 지원: 아니요

서비스 역할은 서비스가 사용자를 대신하여 작업을 수행하는 것으로 가정하는 [IAM 역할](#)입니다. IAM 관리자는 IAM 내에서 서비스 역할을 생성, 수정 및 삭제할 수 있습니다. 자세한 내용은 IAM 사용 설명서의 [AWS 서비스 AWS에 권한을 위임할 역할 생성](#)을 참조하세요.

⚠ Warning

서비스 역할에 대한 권한을 변경하면 Amazon GameLift Streams 기능이 중단될 수 있습니다. Amazon GameLift Streams가 관련 지침을 제공하는 경우에만 서비스 역할을 편집합니다.

Amazon GameLift Streams의 서비스 연결 역할

서비스 연결 역할 지원: 아니요

서비스 연결 역할은 연결된 서비스 역할의 한 유형입니다. AWS 서비스. 서비스는 사용자를 대신하여 작업을 수행하기 위해 역할을 수임할 수 있습니다. 서비스 연결 역할은 표시 AWS 계정 되며 서비스가 소유합니다. IAM 관리자는 서비스 연결 역할의 권한을 볼 수 있지만 편집은 할 수 없습니다.

서비스 연결 역할 생성 또는 관리에 대한 자세한 내용은 [IAM으로 작업하는 AWS 서비스](#)를 참조하세요. 서비스 연결 역할 열에서 Yes가 포함된 서비스를 테이블에서 찾습니다. 해당 서비스에 대한 서비스 연결 역할 설명서를 보려면 예(Yes) 링크를 선택합니다.

Amazon GameLift Streams의 자격 증명 기반 정책 예제

기본적으로 사용자 및 역할은 Amazon GameLift Streams 리소스를 생성하거나 수정할 수 있는 권한이 없습니다. 사용자에게 사용자가 필요한 리소스에서 작업을 수행할 권한을 부여하려면 IAM 관리자가 IAM 정책을 생성하면 됩니다.

이러한 예제 JSON 정책 문서를 사용하여 IAM ID 기반 정책을 생성하는 방법을 알아보려면 IAM 사용 설명서의 [IAM 정책 생성\(콘솔\)](#)을 참조하세요.

각 리소스 유형에 대한 ARNs 형식을 포함하여 Amazon GameLift Streams에서 정의한 작업 및 리소스 유형에 대한 자세한 내용은 서비스 승인 참조의 [Amazon GameLift Streams에 사용되는 작업, 리소스 및 조건 키](#)를 참조하세요.

주제

- [정책 모범 사례](#)
- [Amazon GameLift Streams 콘솔 사용](#)
- [사용자가 자신의 고유한 권한을 볼 수 있도록 허용](#)

정책 모범 사례

자격 증명 기반 정책에 따라 계정에서 사용자가 Amazon GameLift Streams 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 정책](#)을 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 권한](#)을 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 특정을 통해 사용되는 경우 조건을 사용하여 서비스 작업에 대한 액세스 권한을 부여할 수도 있습니다 CloudFormation. 자세한 내용은 IAM 사용 설명서의 [IAM JSON 정책 요소: 조건](#)을 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 검증](#)을 참조하세요.
- 다중 인증(MFA) 필요 -에서 IAM 사용자 또는 루트 사용자가 필요한 시나리오가 있는 경우 추가 보안을 위해 MFA를 AWS 계정합니다. API 작업을 직접적으로 호출할 때 MFA가 필요하다면 정책에 MFA 조건을 추가합니다. 자세한 내용은 IAM 사용 설명서의 [MFA를 통한 보안 API 액세스](#)를 참조하세요.

IAM의 모범 사례에 대한 자세한 내용은 IAM 사용 설명서의 [IAM의 보안 모범 사례](#)를 참조하세요.

Amazon GameLift Streams 콘솔 사용

Amazon GameLift Streams 콘솔에 액세스하려면 최소 권한 집합이 있어야 합니다. 이러한 권한은에서 Amazon GameLift Streams 리소스에 대한 세부 정보를 나열하고 볼 수 있도록 허용해야 합니다 AWS 계정. 최소 필수 권한보다 더 제한적인 ID 기반 정책을 생성하는 경우, 콘솔이 해당 정책에 연결된 엔티티(사용자 또는 역할)에 대해 의도대로 작동하지 않습니다.

AWS CLI 또는 AWS API만 호출하는 사용자에게는 최소 콘솔 권한을 허용할 필요가 없습니다. 대신, 수행하려는 API 작업과 일치하는 작업에만 액세스할 수 있도록 합니다.

사용자가 자신의 고유한 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 방식으로 이 작업을 완료할 수 있는 권한이 포함됩니다.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "ViewOwnUserInfo",
      "Effect": "Allow",
      "Action": [
        "iam:GetUserPolicy",
        "iam:ListGroupForUser",
        "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": "*"
    }
  ]
}
```

Amazon GameLift Streams 자격 증명 및 액세스 문제 해결

다음 정보를 사용하여 Amazon GameLift Streams 및 IAM 작업 시 발생할 수 있는 일반적인 문제를 진단하고 수정할 수 있습니다.

주제

- [Amazon GameLift Streams에서 작업을 수행할 권한이 없음](#)
- [내 외부의 사람이 내 Amazon GameLift Streams 리소스에 액세스 AWS 계정 하도록 허용하고 싶습니다.](#)

Amazon GameLift Streams에서 작업을 수행할 권한이 없음

작업을 수행할 권한이 없다는 오류가 표시되면 작업을 수행할 수 있도록 정책을 업데이트해야 합니다.

다음의 예제 오류는 mateojackson IAM 사용자가 콘솔을 사용하여 가상 *my-example-widget* 리소스에 대한 세부 정보를 보려고 하지만 가상 `gameliftstreams:GetWidget` 권한이 없을 때 발생합니다.

```
User: arn:aws:iam::123456789012:user/mateojackson is not authorized to perform:
gameliftstreams:GetWidget on resource: my-example-widget
```

이 경우, `gameliftstreams:GetWidget` 작업을 사용하여 *my-example-widget* 리소스에 액세스할 수 있도록 mateojackson 사용자 정책을 업데이트해야 합니다.

도움이 필요한 경우 AWS 관리자에게 문의하세요. 관리자는 로그인 자격 증명을 제공한 사람입니다.

[내 외부의 사람이 내 Amazon GameLift Streams 리소스에 액세스 AWS 계정 하도록 허용하고 싶습니다.](#)

Amazon GameLift Streams에서는 이것이 불가능합니다. 모든 API 액세스는 리소스를 소유한 계정으로 제한됩니다. 대신 외부에서 콘텐츠를 공유하려는 고객은 자신의 계정을 사용하여 Amazon GameLift Streams APIs를 사용하여 다른 사용자를 대신하여 새 스트림 세션을 시작하고 해당 외부 사용자의 웹 브라우저에 적절한 연결 정보를 전달할 책임이 있습니다.

Amazon GameLift Streams에 대한 규정 준수 검증

AWS 서비스 가 특정 규정 준수 프로그램의 범위 내에 있는지 알아보려면 [AWS 서비스 규정 준수 프로그램 범위 내](#)를 참조하고 관심 있는 규정 준수 프로그램을 선택합니다. 일반 정보는 [AWS 규정 준수 프로그램](#).

를 사용하여 타사 감사 보고서를 다운로드할 수 있습니다 AWS Artifact. 자세한 내용은 [Downloading Reports in Downloading AWS Artifact](#)을 참조하세요.

사용 시 규정 준수 책임은 데이터의 민감도, 회사의 규정 준수 목표, 관련 법률 및 규정에 따라 AWS 서비스 결정됩니다. 사용 시 규정 준수 책임에 대한 자세한 내용은 [AWS 보안 설명서](#)를 AWS 서비스 참조하세요.

Amazon GameLift 스트림의 복원력

AWS 글로벌 인프라는 AWS 리전 및 가용 영역을 중심으로 구축됩니다.는 물리적으로 분리되고 격리된 여러 가용 영역을 AWS 리전 제공하며,이 가용 영역은 지연 시간이 짧고 처리량이 높으며 중복성이 높은 네트워킹과 연결됩니다. 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 내용은 [AWS 글로벌 인프라](#)를 참조하세요.

AWS 글로벌 인프라에서 제공하는 데이터 중복 외에도 Amazon GameLift Streams는 복원력이 뛰어난 다중 가용 영역 인프라를 기반으로 구축되었습니다. 가용 영역이 중단된 경우 개별 기존 세션이 영향을 받을 수 있지만 서비스는 정상 가용 영역 전체에서 새 세션을 계속 로드 밸런싱합니다.

Amazon GameLift Streams의 인프라 보안

관리형 서비스인 Amazon GameLift Streams는 AWS 글로벌 네트워크 보안으로 보호됩니다. AWS 보안 서비스 및가 인프라를 AWS 보호하는 방법에 대한 자세한 내용은 [AWS 클라우드 보안을](#) 참조하세요. 인프라 보안 모범 사례를 사용하여 환경을 설계하려면 보안 원칙 AWS Well-Architected Framework의 [인프라 보호](#)를 참조하세요 AWS .

AWS 에서 게시한 API 호출을 사용하여 네트워크를 통해 Amazon GameLift Streams에 액세스합니다. 클라이언트는 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.

Amazon GameLift Streams에서 재사용 및 다중 테넌시

Amazon GameLift Streams는 스트림 그룹 간에 또는 다른 AWS 고객과 컴퓨팅 리소스를 공유하지 않습니다. 일부 Amazon GameLift Streams 스트림 그룹은 내부 리소스 공유에 의존합니다.

컴퓨팅 리소스 재사용

스트림 그룹 내에서 리소스는 시간이 지남에 따라 재사용되어 가동 중지 시간을 최소화하면서 여러 세션을 제공합니다. 재사용에 대한 구체적인 세부 정보는 Windows 스트림 그룹과 Windows가 아닌 스트림 그룹 간에 다릅니다.

와 같은 스트림 클래스가 있는 Windows가 아닌 스트림 그룹은 세션별 전용 컨테이너 내에서 애플리케이션을 `gen6n_ultra` 실행합니다. 각 스트림 세션은 애플리케이션 파일 사본과 빈 사용자 프로필 폴더로 시작됩니다. 세션이 종료되면 모든 파일 시스템 수정 사항이 삭제되고 애플리케이션에서 시작한 모든 프로세스가 컨테이너 정리의 일부로 종료됩니다.

와 같은 스트림 클래스가 있는 Windows 기반 스트림 그룹은 호스트 운영 체제에서 직접 애플리케이션을 `gen6n_ultra_win2022` 실행합니다. 각 스트림 세션은 애플리케이션 파일 사본과 빈 사용자 프로필 폴더로 시작됩니다. 세션이 종료되면 사용자 프로필 폴더와 애플리케이션 폴더가 완전히 재설정됩니다. 애플리케이션에서 시작한 하위 프로세스가 종료됩니다. 애플리케이션이 사용자 프로필 폴더 및 애플리케이션 폴더 외부의 파일을 수정하거나 시스템 레지스트리를 수정하는 경우 이러한 변경 사항은 여러 세션에서 지속될 수 있습니다.

모든 스트림 그룹 구성의 경우 기본 컴퓨팅 리소스와 운영 체제 환경은 시간이 지남에 따라 재사용되어 새 스트림 세션을 시작합니다. [공동 책임 모델에](#) 따라 애플리케이션의 보안을 유지하고 신뢰할 수 없는 코드를 실행하거나 중요한 운영 체제 파일을 수정하지 않는 것은 사용자의 책임입니다.

다중 테넌트 스트림 그룹

스트림 그룹은 스트림 클래스 선택에 따라 단일 테넌트 또는 다중 테넌트입니다. 다중 테넌트 스트림 클래스는 여러 동시 세션에서 하나의 GPU를 공유합니다. 이 컨텍스트에서 다중 테넌시는 기본 하드웨어에서 한 번에 두 개 이상의 세션을 실행하는 것을 의미합니다. 하드웨어는 여전히 스트림 그룹 전용이며 스트림 그룹 간에 또는 다른 AWS 고객과 공유되지 않습니다.

이 멀티 테넌트 스트림 그룹 모델은 Amazon GameLift Streams에 고유하며 중요한 보안 및 성능 영향을 미칩니다. 다중 테넌트 스트림 그룹의 보안 태세는 단일 물리적 서버에서 여러 애플리케이션 컨테이

너를 호스팅하는 것과 같습니다. 이 태세는 본질적으로 안전하지 않지만 애플리케이션에서 기존 보안 취약성의 영향을 증폭할 수 있습니다. [공동 책임 모델에](#) 따라 애플리케이션의 보안을 유지하는 것은 사용자의 책임입니다.

Amazon GameLift Streams는 다중 테넌트 세션이 서로 방해하지 않도록 노력합니다. 그러나 애플리케이션이 스트림 클래스의 정의된 제한에 관계없이 CPU 또는 GPU 리소스를 사용하는 경우 동일한 공유 리소스를 사용하려는 다른 스트림에 영향을 미칠 수 있습니다. 예를 들어 GPU당 테넌트가 2개인 "높음" 스트림 그룹에서 복잡한 애플리케이션은 최대 1개의 다른 스트림에 부정적인 영향을 미칠 수 있습니다. 애플리케이션은 자체 리소스 소비를 규제해야 합니다. 애플리케이션이 자체적으로 규제할 수 없고 사용 사례에 잠재적인 "시끄러운 이웃" 성능 변화에 대한 허용 오차가 없는 경우, gen5n_win2022, gen6n_pro_win2022 gen5n_ultra 또는와 같은 단일 테넌트 스트림 클래스 gen6n_ultra가 권장됩니다.

Amazon GameLift Streams의 인터페이스 VPC 엔드포인트

인터페이스 VPC 엔드포인트를 사용하도록 Amazon GameLift Streams를 구성하여 VPC의 보안 태세를 개선할 수 있습니다. 인터페이스 엔드포인트는 프라이빗 IP 주소를 사용하여 Amazon GameLift Streams APIs에 비공개로 액세스할 수 있는 AWS PrivateLink 기술로 구동됩니다. AWS PrivateLink는 VPC와 Amazon GameLift Streams 간의 모든 네트워크 트래픽을 Amazon 네트워크로 제한합니다. 인터넷 게이트웨이, NAT 디바이스 또는 가상 프라이빗 게이트웨이가 필요 없습니다.

AWS PrivateLink 및 VPC 엔드포인트에 대한 자세한 내용은 Amazon [VPC 사용 설명서의 VPC 엔드포인트](#)를 참조하세요.

Note

AWS PrivateLink는 API 엔드포인트에만 적용됩니다. Amazon GameLift Streams 관리형 스트림 세션은 항상 퍼블릭 네트워크 주소를 사용합니다.

Amazon GameLift Streams용 VPC 엔드포인트 생성

Amazon GameLift Streams 서비스에 대한 VPC 엔드포인트를 생성하려면 Amazon [VPC 사용 설명서의 인터페이스 VPC 엔드포인트를 사용하여 AWS 서비스 액세스](#) 절차를 사용하여 다음 엔드포인트를 생성합니다.

- `com.amazonaws.region.gameliftstreams`

Note

*region*은 미국 동부(오하이오) 리전과 같이 Amazon GameLift Streams에서 AWS 리전 지원 하는 us-east-2의 리전 식별자를 나타냅니다.

Amazon GameLift Streams에 대한 VPC 엔드포인트 정책 생성

Amazon GameLift Streams에 대한 액세스를 제어하는 VPC 엔드포인트에 엔드포인트 정책을 연결할 수 있습니다. 이 정책은 다음 정보를 지정합니다.

- 작업을 수행할 수 있는 위탁자.
- 수행할 수 있는 작업.
- 작업을 수행할 수 있는 리소스

자세한 내용은 Amazon VPC 사용자 가이드의 [엔드포인트 정책을 사용하여 VPC 엔드포인트로 액세스 제어](#)를 참조하세요.

Example예: Amazon GameLift Streams에 대한 VPC 엔드포인트 정책

다음은 Amazon GameLift Streams에 대한 엔드포인트 정책의 예입니다. 엔드포인트에 연결되면 이 정책은 스트림 그룹을 생성하고 나열할 수 있는 권한을 부여합니다.

```
{
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": "*",
      "Action": [
        "gameliftstreams:CreateStreamGroup",
        "gameliftstreams:ListStreamGroups"
      ],
      "Resource": [
        "*"
      ]
    }
  ]
}
```

Amazon GameLift Streams의 구성 및 취약성 분석

구성 및 IT 제어는 AWS 와 고객 간의 공동 책임입니다. 자세한 내용은 AWS [공동 책임 모델을](#) 참조하세요. AWS 는 게스트 운영 체제(OS) 및 데이터베이스 패치, 방화벽 구성, 재해 복구와 같은 기본 보안 작업을 처리합니다. 적합한 제3자가 이 절차를 검토하고 인증하였습니다. 자세한 내용은 다음 리소스를 참조하세요. [Amazon Web Services: 보안 프로세스 개요](#)(백서).

다음 보안 모범 사례에서는 Amazon GameLift Streams의 구성 및 취약성 분석도 다룹니다.

- 고객은 스트림 호스팅을 위해 Amazon GameLift Streams 스트림 그룹에 배포된 소프트웨어를 관리할 책임이 있습니다. 구체적으로 설명하면 다음과 같습니다.
 - 업데이트 및 보안 패치를 포함하여 고객이 제공한 애플리케이션 콘텐츠 및 소프트웨어를 유지해야 합니다. 업데이트하려면 새 Amazon GameLift Streams 애플리케이션을 생성하고 새 스트림 그룹에 배포합니다.
 - 현재 스트림 그룹의 운영 체제 및 런타임 환경은 새 스트림 그룹을 생성할 때만 업데이트됩니다. 런타임 환경의 일부인 운영 체제 및 기타 애플리케이션을 패치, 업데이트 및 보호하려면 애플리케이션 업데이트와 관계없이 2~4주마다 스트림 그룹을 재활용하는 것이 좋습니다.
- 고객은 SDK 및 Amazon GameLift Streams 웹 클라이언트 SDK를 포함한 최신 AWS SDK 버전으로 게임을 정기적으로 업데이트하는 것을 고려해야 합니다.

Amazon GameLift Streams의 보안 모범 사례

Amazon GameLift Streams는 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 환경에 적절하지 않거나 충분하지 않을 수 있으므로 참고용으로만 사용하세요.

- 현재 스트림 그룹의 운영 체제 및 런타임 환경은 새 스트림 그룹을 생성할 때만 업데이트됩니다. 런타임 환경의 일부인 운영 체제 및 기타 애플리케이션을 패치, 업데이트 및 보호하려면 애플리케이션 업데이트와 관계없이 2~4주마다 스트림 그룹을 재활용하는 것이 좋습니다.
- [보안, 자격 증명 및 규정 준수 모범 사례](#)

Amazon GameLift 스트림 모니터링

모니터링은 Amazon GameLift Streams 및 기타 AWS 솔루션의 안정성, 가용성 및 성능을 유지하는 데 중요한 부분입니다. Amazon GameLift Streams를 모니터링하고, 이상이 있을 때 이를 보고하고, 적절한 경우 자동 조치를 취할 수 있는 다음과 같은 모니터링 도구를 AWS 제공합니다.

- Amazon CloudWatch는 AWS 리소스와 AWS에서 실행하는 애플리케이션을 실시간으로 모니터링합니다. 지표를 수집 및 추적하고, 사용자 지정 대시보드를 생성할 수 있으며, 지정된 지표가 지정한 임계값에 도달하면 사용자에게 알리거나 조치를 취하도록 경보를 설정할 수 있습니다. 자세한 내용은 [Amazon CloudWatch 사용 설명서](#)를 참조하십시오.
- Amazon CloudWatch Logs를 사용하면 Amazon Elastic Compute Cloud와 같은 서비스 및 기타 소스에서 로그 파일을 모니터링 AWS CloudTrail, 저장 및 액세스할 수 있습니다. CloudWatch Logs는 로그 파일의 정보를 모니터링하고 서비스가 특정 임계값을 충족할 때 사용자에게 알릴 수 있습니다. 또한 매우 내구성이 뛰어난 스토리지에 로그 데이터를 저장할 수 있습니다. 자세한 내용은 [Amazon CloudWatch Logs 사용자 안내서](#)를 참조하세요.
- AWS CloudTrail은 AWS 계정에서 또는 계정을 대신하여 수행한 API 호출 및 관련 이벤트를 캡처하고 사용자가 지정한 Amazon Simple Storage Service 버킷으로 로그 파일을 전송합니다. 호출한 사용자 및 계정 AWS, 호출이 수행된 소스 IP 주소, 호출이 발생한 시기를 식별할 수 있습니다. 자세한 내용은 [AWS CloudTrail 사용 설명서](#)를 참조하십시오.
- 실시간 성능 통계는 스트림 세션 중에 애플리케이션 수준 및 공유 시스템 수준 성능 통계를 수집합니다. 이러한 통계는 클라이언트에서 실시간으로 수신하거나 세션 후 내보낸 세션 파일에서 CSV 파일로 수신할 수 있습니다. 이 기능을 사용하면 스트림의 CPU, 메모리, GPU 및 VRAM 사용률을 모니터링할 수 있습니다. 자세한 내용은 [the section called “실시간 성능 통계”](#) 단원을 참조하십시오.

Amazon CloudWatch를 사용하여 Amazon GameLift 스트림 모니터링

원시 데이터를 수집하여 읽기 가능하며 실시간에 가까운 지표로 처리하는 CloudWatch를 사용하여 Amazon GameLift Streams를 모니터링할 수 있습니다. 이러한 통계는 15개월간 보관되므로 기록 정보에 액세스하고 웹 애플리케이션 또는 서비스가 어떻게 실행되고 있는지 전체적으로 더 잘 파악할 수 있습니다. 특정 임계값을 주시하다가 해당 임계값이 충족될 때 알림을 전송하거나 조치를 취하도록 경보를 설정할 수도 있습니다. 자세한 내용은 [Amazon CloudWatch 사용 설명서](#)를 참조하세요.

Amazon GameLift Streams는 고객이 다음을 모니터링하는 데 도움이 되는 지표를 제공합니다.

- 그룹 용량 및 사용량을 스트리밍합니다.
- 성능 및 리소스 사용량을 스트리밍합니다.
- 상태를 스트리밍하여 문제를 해결하고 사용자를 지원합니다.
- 콘텐츠 상품 전반의 고객 참여.
- 데이터 채널 사용량.

다음 표에는 Amazon GameLift Streams의 차원 및 지표가 나열되어 있습니다.

스트림 그룹 용량 및 사용량

이러한 지표를 사용하여 수요에 맞게 리소스를 확장할 수 있습니다. 이러한 지표는 1분마다 게시됩니다.

Important

2025년 9월 5일 이전에 생성된 스트림 그룹의 경우

CloudWatch의 데이터 보존 정책 문제로 인해 정확한 용량 지표는 지난 15일 동안만 사용할 수 있습니다. 15일이 지난 용량 지표의 경우 기간이 1분이면 데이터가 표시되지 않고 기간이 5분 이상이면 표시된 데이터가 부정확해집니다.

해결 방법으로 CloudWatch 그래프의 합계 유형 통계에 $SUM(METRICS())/5$ 수식(예: 5분 기간 사용)을 해결 방법으로 추가하여 15일, 1분 지표 보존 제한을 초과하는 정확한 용량 수를 확인할 수 있습니다.

이 문제에 대한 수정 사항을 적용하려면 스트림 그룹을 다시 생성합니다.

지표	설명	측정기준	단위
ActiveCapacity	프로비저닝되어 스트리밍할 준비가 된 컴퓨팅 리소스의 수입니다. 여기에는 현재 스트리밍 중인 리소스와 유휴 상태이고 새 스트림 요청에 응답할 준비가 된 리소스가 포함됩니다.	(StreamGroupID, 위치)	개수
IdleCapacity	현재 스트리밍되지 않는 활성 용량의 숫자 부분입니다. 새 스	(StreamGroupID, 위치)	개수

지표	설명	측정기준	단위
	트림 요청에 응답할 컴퓨팅 리소스의 가용성을 나타냅니다.		

스트림 그룹 성능 및 리소스 사용률

이러한 지표는 1분마다 게시됩니다.

지표	설명	측정기준	단위
MemoryUtilization	스트림에서 사용하는 사용 가능한 메모리의 %입니다.	(StreamGroupId, Location), (ApplicationId, StreamClass)	백분율
CPUUtilization	스트림에서 사용하는 사용 가능한 CPU의 %입니다.	(StreamGroupId, Location), (ApplicationId, StreamClass)	백분율
FrameCaptureRate	애플리케이션에서 프레임이 캡처되는 속도입니다.	(StreamGroupId, Location), (ApplicationId, StreamClass)	없음
AudioCaptureRate	애플리케이션에서 오디오 샘플을 캡처하는 속도입니다.	(StreamGroupId, Location), (ApplicationId, StreamClass)	없음

지표	설명	측정기준	단위
		ionId, StreamClass)	
RoundTripTime	클라이언트와 서버 간의 왕복 시간입니다.	(StreamGroupId, Location) , (ApplicationId, StreamClass)	ms

스트림 상태

이러한 지표는 스트림 세션이 끝날 때 게시됩니다.

지표	설명	측정기준	단위
TerminatedStreamSessions	상태로 종료된 세션 수 TERMINATED	(StreamGroupId, Location) , (ApplicationId, StreamClass)	개수
ErroredStreamSessions	상태로 종료된 세션 수 ERROR	(StreamGroupId, Location) , (ApplicationId, StreamClass)	개수

고객 참여

이러한 지표는 스트림 세션이 끝날 때 게시됩니다.

지표	설명	측정기준	단위
세션 길이	스트림 세션 기간	(StreamGroupId, Location), (ApplicationId, StreamClass)	초

데이터 채널

이러한 지표는 스트림 세션이 끝날 때 게시됩니다.

지표	설명	측정기준	단위
DataChannel-ApplicationConnected	애플리케이션이 데이터 채널 포트에 연결되는 횟수입니다. 이 수는 스트림 세션당 최대 1개입니다.	(StreamGroupId, Location), (ApplicationId, StreamClass)	개수
DataChannel-ApplicationMessage	애플리케이션이 클라이언트에 전송한 메시지 수입니다.	(StreamGroupId, Location), (ApplicationId, StreamClass)	개수
DataChannel-ApplicationMessageBytes	애플리케이션이 클라이언트에 전송한 메시지의 총 바이트입니다.	(StreamGroupId, Location), (ApplicationId, StreamClass)	바이트

지표	설명	측정기준	단위
DataChannel-ClientMessage	클라이언트가 애플리케이션에 전송한 메시지 수입니다.	(StreamGroupId, Location), (ApplicationId, StreamClass)	개수
DataChannel-ClientMessageBytes	클라이언트가 애플리케이션에 전송한 메시지의 총 바이트입니다.	(StreamGroupId, Location), (ApplicationId, StreamClass)	바이트

를 사용하여 Amazon GameLift Streams API 호출 로깅 AWS CloudTrail

Amazon GameLift Streams는 사용자 [AWS CloudTrail](#), 역할 또는가 수행한 작업에 대한 레코드를 제공하는 서비스인과 통합됩니다 AWS 서비스. CloudTrail은 Amazon GameLift Streams에 대한 모든 API 호출을 이벤트로 캡처합니다. 캡처되는 호출에는 Amazon GameLift Streams 콘솔의 호출과 Amazon GameLift Streams API 작업에 대한 코드 호출이 포함됩니다. CloudTrail에서 수집한 정보를 사용하여 Amazon GameLift Streams에 수행된 요청, 요청이 수행된 IP 주소, 요청이 수행된 시간 및 추가 세부 정보를 확인할 수 있습니다.

모든 이벤트 또는 로그 항목에는 요청을 생성했던 사용자에 대한 정보가 포함됩니다. 자격 증명을 이용하면 다음을 쉽게 판단할 수 있습니다.

- 요청을 루트 사용자로 했는지 사용자 보안 인증으로 했는지 여부.
- IAM Identity Center 사용자를 대신하여 요청이 이루어졌는지 여부입니다.
- 역할 또는 페더레이션 사용자에 대한 임시 자격 증명을 사용하여 요청이 생성되었는지 여부.
- 다른 AWS 서비스에서 요청했는지 여부.

CloudTrail은 계정을 생성할 AWS 계정 때에서 활성화되며 CloudTrail 이벤트 기록에 자동으로 액세스할 수 있습니다. CloudTrail 이벤트 기록은 지난 90일 간 AWS 리전의 관리 이벤트에 대해 보기, 검색 및 다운로드가 가능하고, 수정이 불가능한 레코드를 제공합니다. 자세한 설명은 AWS CloudTrail 사용 설명서의 [CloudTrail 이벤트 기록 작업](#)을 참조하세요. Event history(이벤트 기록) 보기는 CloudTrail 요금이 부과되지 않습니다.

AWS 계정 지난 90일 동안의 이벤트를 지속적으로 기록하려면 추적 또는 [CloudTrail Lake](#) 이벤트 데이터 스토어를 생성합니다.

CloudTrail 추적

CloudTrail은 추적을 사용하여 Amazon S3 버킷으로 로그 파일을 전송할 수 있습니다. 를 사용하여 생성된 모든 추적 AWS Management Console 은 다중 리전입니다. AWS CLI를 사용하여 단일 리전 또는 다중 리전 추적을 생성할 수 있습니다. 계정의 모든 AWS 리전 에서 활동을 캡처하므로 다중 리전 추적을 생성하는 것이 좋습니다. 단일 리전 추적을 생성하는 경우 추적의 AWS 리전에 로깅된 이벤트만 볼 수 있습니다. 추적에 대한 자세한 내용은 AWS CloudTrail 사용 설명서의 [Creating a trail for your AWS 계정](#) 및 [Creating a trail for an organization](#)을 참조하세요.

CloudTrail에서 추적을 생성하여 진행 중인 관리 이벤트의 사본 하나를 Amazon S3 버킷으로 무료로 전송할 수는 있지만, Amazon S3 스토리지 요금이 부과됩니다. CloudTrail 요금에 대한 자세한 내용은 [AWS CloudTrail 요금](#)을 참조하세요. Amazon S3 요금에 대한 자세한 내용은 [Amazon S3 요금](#)을 참조하세요.

CloudTrail Lake 이벤트 데이터 스토어

CloudTrail Lake를 사용하면 이벤트에 대해 SQL 기반 쿼리를 실행할 수 있습니다. CloudTrail Lake는 행 기반 JSON 형식의 기존 이벤트를 [Apache ORC](#) 형식으로 변환합니다. ORC는 빠른 데이터 검색에 최적화된 열 기반 스토리지 형식입니다. 이벤트는 이벤트 데이터 스토어로 집계되며, 이벤트 데이터 스토어는 [고급 이벤트 선택기](#)를 적용하여 선택한 기준을 기반으로 하는 변경 불가능한 이벤트 컬렉션입니다. 이벤트 데이터 스토어에 적용하는 선택기는 어떤 이벤트가 지속되고 쿼리할 수 있는지 제어합니다. CloudTrail Lake에 대한 자세한 내용은 AWS CloudTrail 사용 설명서의 [AWS CloudTrail Lake 작업](#)을 참조하세요.

CloudTrail Lake 이벤트 데이터 스토어 및 쿼리에는 비용이 발생합니다. 이벤트 데이터 스토어를 생성할 때 이벤트 데이터 스토어에 사용할 [요금 옵션](#)을 선택합니다. 요금 옵션에 따라 이벤트 모으기 및 저장 비용과 이벤트 데이터 스토어의 기본 및 최대 보존 기간이 결정됩니다. CloudTrail 요금에 대한 자세한 내용은 [AWS CloudTrail 요금](#)을 참조하세요.

CloudTrail에서 Amazon GameLift Streams 데이터 이벤트

[데이터 이벤트](#)는 리소스에서 또는 리소스에서 수행된 리소스 작업에 대한 정보를 제공합니다(예: 스트림 그룹에서 스트림 세션 시작). 이를 데이터 영역 작업이라고도 합니다. 데이터 이벤트가 대량 활동인 경우도 있습니다. 기본적으로 CloudTrail은 데이터 이벤트를 로깅하지 않습니다. CloudTrail 이벤트 기록은 데이터 이벤트를 기록하지 않습니다.

데이터 이벤트에는 추가 요금이 적용됩니다. CloudTrail 요금에 대한 자세한 내용은 [AWS CloudTrail 요금](#)을 참조하세요.

CloudTrail 콘솔 AWS CLI 또는 CloudTrail API 작업을 사용하여 Amazon GameLift Streams 리소스 유형에 대한 데이터 이벤트를 로깅할 수 있습니다. 데이터 이벤트를 로깅하는 방법에 대한 자세한 내용은 AWS CloudTrail 사용 설명서의 [Logging data events with the AWS Management Console](#) 및 [Logging data events with the AWS Command Line Interface](#)를 참조하세요.

다음 표에는 데이터 이벤트를 로깅할 수 있는 Amazon GameLift Streams 리소스 유형이 나열되어 있습니다. 리소스 유형(콘솔) 열에는 CloudTrail 콘솔의 리소스 유형 목록에서 선택할 값이 표시됩니다. resources.type 값 열에는 AWS CLI 또는 CloudTrail APIs를 사용하여 고급 이벤트 선택기를 구성할 때 지정하는 resources.type 값이 표시됩니다. CloudTrail에 로깅되는 데이터 API 열에는 리소스 유형에 대해 CloudTrail에 로깅된 API 호출이 표시됩니다.

리소스 유형(콘솔)	resources.type 값	CloudTrail에 로깅되는 데이터 API
GameLift Streams 애플리케이션	AWS::GameLiftStreams::Application	<ul style="list-style-type: none"> • StartStreamSession
GameLift Streams 스트림 그룹	AWS::GameLiftStreams::StreamGroup	<ul style="list-style-type: none"> • CreateStreamSessionConnection • ExportStreamSessionFiles • GetStreamSession • ListStreamSessions • ListStreamSessionsByAccount • StartStreamSession • TerminateStreamSession

eventName, readOnly 및 resources.ARN 필드를 필터링하여 중요한 이벤트만 로깅하도록 고급 이벤트 선택기를 구성할 수 있습니다. 이러한 필드에 대한 자세한 내용은 AWS CloudTrail API 참조의 [AdvancedFieldSelector](#) 섹션을 참조하세요.

CloudTrail의 Amazon GameLift Streams 관리 이벤트

[관리 이벤트](#)는 리소스에서 수행되는 관리 작업에 대한 정보를 제공합니다. 이 리소스를 플레인 작업이라고도 합니다. 기본적으로 CloudTrail은 관리 이벤트를 로깅합니다.

Amazon GameLift Streams는 다음과 같은 Amazon GameLift Streams 컨트롤 플레인 작업을 관리 이벤트로 CloudTrail에 로깅합니다.

- [AddStreamGroupLocations](#)
- [AssociateApplications](#)
- [CreateApplication](#)
- [CreateStreamGroup](#)
- [DeleteApplication](#)
- [DeleteStreamGroup](#)
- [DisassociateApplications](#)
- [GetApplication](#)
- [GetStreamGroup](#)
- [ListApplications](#)
- [ListStreamGroups](#)
- [ListTagsForResource](#)
- [RemoveStreamGroupLocations](#)
- [TagResource](#)
- [UntagResource](#)
- [UpdateApplication](#)
- [UpdateStreamGroup](#)

Amazon GameLift Streams 이벤트 예제

이벤트는 모든 소스로부터의 단일 요청을 나타내며 요청된 API 작업, 작업 날짜와 시간, 요청 파라미터 등에 대한 정보가 들어 있습니다. CloudTrail 로그 파일은 퍼블릭 API 직접 호출의 주문 스택 추적이 아니므로 이벤트가 특정 순서로 표시되지 않습니다.

다음 예제에서는 [CreateApplication](#) 작업을 보여주는 CloudTrail 관리 이벤트를 보여줍니다.

```
{
  "eventVersion": "1.09",
  "userIdentity": {
    "type": "AssumedRole",
    "principalId": "AROAI23456789EXAMPLE:assume-temporary-gameliftstreams-access-role",
    "arn": "arn:aws:sts::111122223333:assumed-role/GameLiftStreamsTestRole/assume-temporary-gameliftstreams-access-role",
    "accountId": "111122223333",
    "accessKeyId": "ASIAIOSFODNN7EXAMPLE",
    "sessionContext": {
      "sessionIssuer": {
        "type": "Role",
        "principalId": "AROAI23456789EXAMPLE",
        "arn": "arn:aws:iam::111122223333:role/GameLiftStreamsTestRole",
        "accountId": "111122223333",
        "userName": "GameLiftStreamsTestRole"
      },
      "webIdFederationData": {},
      "attributes": {
        "creationDate": "2025-07-23T21:18:19Z",
        "mfaAuthenticated": "false"
      }
    }
  },
  "eventTime": "2025-07-23T21:58:54Z",
  "eventSource": "gameliftstreams.amazonaws.com",
  "eventName": "CreateApplication",
  "awsRegion": "us-west-2",
  "sourceIPAddress": "203.0.113.0",
  "userAgent": "aws-sdk-javascript/2.0.0 Linux/4.14.291-218.527.amzn2.x86_64 OpenJDK_64-Bit_Server_VM/11.0.17+9-LTS Java/11.0.17 vendor/Amazon.com_Inc. exec-env/AWS_ECS_FARGATE io/sync http/Apache cfg/retry-mode/legacy",
  "requestParameters": {
    "ApplicationSourceUri": "s3://amzn-s3-demo-bucket/MyGame",
  }
}
```

```

    "Description": "MyGame canary - Proton 8",
    "RuntimeEnvironment": {
      "Type": "PROTON",
      "Version": "20230704"
    },
    "ClientToken": "a1b2c3d4-5678-90ab-cdef-EXAMPLE33333",
    "ExecutablePath": "MyGame100.exe"
  },
  "responseElements": {
    "Status": "INITIALIZED",
    "ApplicationSourceUri": "s3://amzn-s3-demo-bucket/MyGame",
    "Description": "MyGame canary - Proton 8",
    "RuntimeEnvironment": {
      "Type": "PROTON",
      "Version": "20230704"
    },
    "LastUpdatedAt": 1753307934.293,
    "CreatedAt": 1753307934.293,
    "Id": "a-9ZY8X7Wv6",
    "Arn": "arn:aws:gameliftstreams:us-west-2:111122223333:application/
a-9ZY8X7Wv6",
    "ExecutablePath": "MyGame100.exe"
  },
  "requestID": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
  "eventID": "a1b2c3d4-5678-90ab-cdef-EXAMPLEbbbbbb",
  "readOnly": false,
  "eventType": "AwsApiCall",
  "managementEvent": true,
  "recipientAccountId": "111122223333",
  "eventCategory": "Management"
}

```

다음 예제는 [StartStreamSession](#) 작업을 보여주는 추적 로그의 CloudTrail 데이터 이벤트를 보여줍니다.

```

{
  "Records": [
    {
      "eventVersion": "1.09",
      "userIdentity": {
        "type": "AssumedRole",
        "principalId": "AR0A123456789EXAMPLE:assume-temporary-gameliftstreams-
access-role",

```

```

    "arn": "arn:aws:sts::111122223333:assumed-role/GameLiftStreamsTestRole/
assume-temporary-gameliftstreams-access-role",
    "accountId": "111122223333",
    "accessKeyId": "ASIAIOSFODNN7EXAMPLE",
    "sessionContext": {
      "sessionIssuer": {
        "type": "Role",
        "principalId": "AROAI123456789EXAMPLE",
        "arn": "arn:aws:iam::111122223333:role/
GameLiftStreamsTestRole",
        "accountId": "111122223333",
        "userName": "GameLiftStreamsTestRole"
      },
      "attributes": {
        "creationDate": "2025-07-23T21:18:19Z",
        "mfaAuthenticated": "false"
      }
    }
  },
  "eventTime": "2025-07-23T23:43:46Z",
  "eventSource": "gameliftstreams.amazonaws.com",
  "eventName": "StartStreamSession",
  "awsRegion": "us-east-2",
  "sourceIPAddress": "203.0.113.0",
  "userAgent": "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36
(KHTML, like Gecko) Chrome/138.0.0.0 Safari/537.36",
  "requestParameters": {
    "Identifier": "sg-1AB2C3De4",
    "Description": "StreamGroup sg-1AB2C3De4 Application a-9ZY8X7Wv6
Console stream",
    "AdditionalLaunchArgs": [],
    "UserId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "Locations": [
      "us-east-2"
    ],
    "SignalRequest": "****",
    "Protocol": "WebRTC",
    "ApplicationIdentifier": "a-9ZY8X7Wv6",
    "ClientToken": "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
    "ConnectionTimeoutSeconds": 100,
    "AdditionalEnvironmentVariables": {}
  },
  "responseElements": {
    "Status": "ACTIVATING",

```

```

        "ApplicationArn": "arn:aws:gameliftstreams:us-
west-2:111122223333:application/a-9ZY8X7Wv6",
        "Description": "StreamGroup sg-1AB2C3De4 Application a-9ZY8X7Wv6
Console stream",
        "LastUpdatedAt": 1.753314225925E9,
        "CreatedAt": 1.753314225925E9,
        "AdditionalEnvironmentVariables": {},
        "ConnectionTimeoutSeconds": 100,
        "AdditionalLaunchArgs": [],
        "StreamGroupId": "sg-1AB2C3De4",
        "UserId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
        "SessionLengthSeconds": 43200,
        "SignalRequest": "****",
        "Arn": "arn:aws:gameliftstreams:us-west-2:111122223333:streamsession/
sg-1AB2C3De4/ABC123def4567",
        "Protocol": "WebRTC",
        "WebSdkProtocolUrl": "https://123456789012.cloudfront.net/
e3b0c44298fc1c149afb4c8996fb92427ae41e4649b934ca495991b7852b855.js"
    },
    "requestID": "a1b2c3d4-5678-90ab-cdef-EXAMPLEaaaaa",
    "eventID": "a1b2c3d4-5678-90ab-cdef-EXAMPLEbbbbbb",
    "readOnly": false,
    "resources": [
        {
            "accountId": "111122223333",
            "type": "AWS::GameLiftStreams::StreamGroup",
            "ARN": "arn:aws:gameliftstreams:us-west-2:111122223333:streamgroup/
sg-1AB2C3De4"
        },
        {
            "accountId": "111122223333",
            "type": "AWS::GameLiftStreams::Application",
            "ARN": "arn:aws:gameliftstreams:us-west-2:111122223333:application/
a-9ZY8X7Wv6"
        }
    ],
    "eventType": "AwsApiCall",
    "managementEvent": false,
    "recipientAccountId": "111122223333",
    "eventCategory": "Data"
}
]
}

```

CloudTrail 레코드 콘텐츠에 대한 자세한 내용은 AWS CloudTrail 사용 설명서의 [CloudTrail record contents](#)를 참조하세요.

실시간 성능 통계

Amazon GameLift Streams는 활성 스트림 세션 중에 성능 통계를 수집하여 초당 리소스 사용률을 측정합니다. 이러한 통계를 사용하여 애플리케이션의 성능을 모니터링하고, 리소스 병목 현상을 식별하고, 스트리밍 환경을 최적화합니다.

성능 통계에는 애플리케이션 수준 통계(특정 애플리케이션의 CPU 및 메모리 사용률)와 시스템 수준 통계(공유 컴퓨팅 인프라의 CPU, 메모리, GPU 및 VRAM 사용률)가 모두 포함됩니다.

다음 두 가지 방법으로 성능 통계를 수신할 수 있습니다.

- 세션 중 실시간: Amazon GameLift Streams Web SDK를 사용하여 수집되는 통계를 수신합니다. 이를 통해 애플리케이션과 상호 작용할 때 성능 오버레이를 구축하고 리소스 사용률을 모니터링할 수 있습니다.
- 세션 후를 CSV 파일로: 세션 파일을 내보내면 통계가 `stats/perf_stats_v1.csv`로 포함됩니다. 세션 후 분석 및 디버깅에 대한 전체 레코드를 제공합니다.

성능 통계 수신

실시간 통계 수신

활성 세션 중에 클라이언트 애플리케이션에서 성능 통계를 수신하려면 `StartStreamSession` API를 호출할 때 `SharedWithClient` 파라미터를 `true`로 설정합니다. Amazon GameLift Streams 웹 SDK는 스트리밍 세션에서 새 통계가 도착할 때마다 트리거되는 `performanceStats` 콜백을 제공합니다.

Warning

최종 사용자와 함께 프로덕션 세션 `SharedWithClient`에 대해 활성화하지 마십시오. 내부 디버깅 및 테스트와 같이 클라이언트가 신뢰할 수 있는 경우에만 활성화합니다.

Amazon GameLift Streams Web SDK를 초기화할 때 성능 통계를 수신 `clientConnection.performanceStats` 콜백 함수로 설정합니다.

```

const gls = new gameliftstreams.GameLiftStreams({
  videoElement: document.getElementById('streamVideoElement'),
  audioElement: document.getElementById('streamAudioElement'),
  inputConfiguration: {
    ...
  },
  clientConnection: {
    ...
    performanceStats: (perfStats) => {
      // Your callback logic here
      console.log('CPU: ' + perfStats.application.cpuNormalized);
      console.log('Memory: ' + perfStats.application.memoryMB + ' MB');
      console.log('GPU: ' + perfStats.system.gpuPercent + '%');
    },
  },
});

```

콜백은 애플리케이션 수준 및 시스템 수준 통계가 모두 포함된 PerformanceStats 객체를 수신합니다. 인터페이스 구조에 대한 자세한 내용은 [시작하기 제품 페이지의](#) Amazon GameLift Streams Web SDK 설명서를 참조하세요.

Amazon GameLift Streams 콘솔에는 테스트 스트림 기능을 사용할 때 내장된 성능 오버레이도 포함되어 있으므로 구현 작업 없이 실시간으로 통계를 모니터링할 수 있습니다.

성능 통계를 Amazon GameLift Streams Web SDK의 `getVideoRTCStats()` 및 `getAudioRTCStats()` 함수에서 제공하는 WebRTC 통계와 결합할 수 있습니다. 이 조합은 네트워크 통계, 클라이언트 프레임 속도 및 리소스 사용률을 포함하여 스트리밍 성능에 대한 전체 그림을 제공합니다.

세션 후 통계 수신

Amazon GameLift Streams는 모든 스트림 세션 중에 성능 통계를 자동으로 수집합니다. 세션 파일을 내보낼 때 통계는 `stats/perf_stats_v1.csv` 내보낸 ZIP 파일에 포함됩니다. 세션 후 분석 및 디버깅을 위해 세션 중에 수집된 모든 통계에 대한 전체 레코드를 제공합니다.

세션 파일 내보내기에 대한 자세한 내용은 [섹션을 참조하세요](#) [the section called “스트림 세션 파일 내보내기”](#).

성능 통계 참조

다음 표에는 Amazon GameLift Streams에서 수집한 모든 성능 통계가 나열되어 있습니다. 애플리케이션 통계는 현재 세션에 고유하지만 공유 시스템 통계는 다중 테넌트 스트림 클래스의 세션별 공유 컴퓨팅의 총 사용률을 반영합니다.

다중 테넌트 스트림 클래스의 정규화된 통계

Amazon GameLift Streams는 여러 세션이 동일한 컴퓨팅 인스턴스를 공유할 수 있는 다중 테넌트 스트림 클래스를 지원합니다. 정규화된 통계(애플리케이션 CPU 및 메모리 사용률)는 할당된 공정 공유를 기준으로 애플리케이션의 리소스 사용량을 측정합니다. 공정 공유는 스트림 클래스 테넌시를 기반으로 컴퓨팅 인스턴스에서 사용 가능한 총 CPU와 메모리를 균등하게 나누어 계산됩니다.

값이 1.0이면 애플리케이션이 공정 공유 할당을 정확히 사용하고 있음을 의미합니다. 1.0 미만의 값은 할당보다 적게 사용하고 있음을 나타냅니다. 값이 1.0을 초과하면 과도하게 사용되어 세션 성능이 저하될 수 있습니다. 다중 테넌트 스트림 클래스(테넌시가 1보다 큼)에서 과도하게 사용하면 동일한 컴퓨팅 인스턴스를 공유하는 다른 세션에도 영향을 미칠 수 있습니다.

다음 표에 나열된 통계 이름은 내보낸 파일의 CSV 열 헤더로 사용됩니다. Amazon GameLift Streams Web SDK를 통해 실시간으로 통계를 수신하는 경우 이러한 통계는 카멜 대소문자의 속성 이름이 있는 PerformanceStats 인터페이스를 통해 사용할 수 있습니다. 정확한 인터페이스 구조 및 속성 이름은 [시작하기 제품 페이지](#)의 Amazon GameLift Streams Web SDK API 참조 가이드를 참조하세요.

통계 이름(CSV 열)	설명	Scope
timestamp	ISO 8601 형식의 측정을 수행한 시간입니다.	모두
app_cpu_normalized	공정 공유 할당에 대해 정규화된 애플리케이션의 CPU 사용량입니다. 여기서 1.0은 목표 공정 공유 한도를 나타냅니다. 사용량이 1.0을 초과하면 과도하게 사용되어 성능 문제가 발생할 수 있습니다.	애플리케이션
app_mem_mb	애플리케이션에서 사용하는 총 메모리(RAM)(MiB로 측정)	애플리케이션

통계 이름(CSV 열)	설명	Scope
app_mem_normalized	공정 공유 할당에 대해 정규화된 애플리케이션의 메모리 사용량입니다. 여기서 1.0은 목표 공정 공유 한도를 나타냅니다. 사용량이 1.0을 초과하면 과도하게 사용되어 성능 문제가 발생할 수 있습니다.	애플리케이션
shared_sys_cpu_pct	공유 컴퓨팅에서 총 CPU 사용량의 백분율입니다.	공유 시스템
shared_sys_mem_mb	인스턴스에 사용된 총 메모리(MiB로 측정).	공유 시스템
shared_sys_mem_pct	공유 컴퓨팅에서 사용 중인 총 메모리의 백분율입니다.	공유 시스템
shared_sys_gpu_pct	공유 컴퓨팅 전반의 총 GPU 사용률 백분율입니다.	공유 시스템
shared_sys_vram_mb	공유 컴퓨팅에 사용된 총 VRAM(GPU 메모리)(MiB로 측정).	공유 시스템
shared_sys_vram_pct	공유 컴퓨팅에서 사용 중인 총 VRAM(GPU 메모리)의 백분율입니다.	공유 시스템

Amazon GameLift 스트림 문제 해결

주제

- [Amazon GameLift Streams 서비스에 요청할 때 액세스가 거부됨](#)
- [애플리케이션 문제](#)
- [성능 문제](#)
- [스트림 연결 및 네트워크 성능 문제](#)
- [스트림 입력 문제](#)
- [스트림 세션 문제](#)
- [Proton for Amazon GameLift Streams와의 호환성 테스트 및 문제 해결](#)
- [Unreal Engine 성능 프로파일링](#)

Amazon GameLift Streams 서비스에 요청할 때 액세스가 거부됨

Amazon GameLift Streams APIs하거나 콘솔에서 리소스를 사용할 때 AccessDenied 예외가 발생하면 AWS Identity and Access Management (IAM) 역할에 Amazon GameLift Streams에 대한 권한이 충분하지 않을 수 있습니다. 다음을 확인하세요.

- IAM 역할에 명시적 "모두 거부" 정책이 있는 경우 "gameliftstreams:*" [NotAction](#) 요소에를 추가하여 Amazon GameLift Streams를 해당 정책에 대한 예외로 명시적으로 나열해야 합니다. 예제:

```
{
  "Sid": "DenyAllExceptListedIfNoMFA",
  "Effect": "Deny",
  "NotAction": [
    "iam:CreateVirtualMFADevice",
    "iam:EnableMFADevice",
    "iam:GetUser",
    "iam:ListMFADevices",
    "iam:ListVirtualMFADevices",
    "iam:ResyncMFADevice",
    "sts:GetSessionToken",
    "gameliftstreams:*" // Add this
  ],
  "Resource": "*",
  "Condition": {
    "BoolIfExists": {"aws:MultiFactorAuthPresent": "false"}
  }
}
```

- 자세한 내용은 보안 [Amazon GameLift Streams의 ID 및 액세스 관리](#) 장의 섹션을 참조하고 IAM 사용 설명서의 [액세스 거부 오류 메시지 문제 해결을 참조하세요](#).

애플리케이션 문제

이 섹션에서는 애플리케이션이 실행되지 않거나 Amazon GameLift Streams에서 다르게 나타나는 문제의 잠재적 원인을 식별합니다.

예비 검사

- 다른 시스템에서 애플리케이션을 실행하여 올바르게 패키징되었는지 확인합니다. 이렇게 하면 애플리케이션 콘텐츠에 하드코딩된 경로, 누락된 자산, 라이브러리 또는 다른 디바이스에서 작동하지 않을 수 있는 바이너리가 포함되지 않습니다.

양성자 문제

- 애플리케이션이 Proton과 호환되는지 확인합니다. Amazon GameLift Streams 서버가 없는 로컬 환경에서 애플리케이션을 테스트하여 Proton과 호환되는지 확인합니다. 지침은 [Proton for Amazon GameLift Streams와의 호환성 테스트 및 문제 해결](#) 섹션을 참조하세요.

화면 해상도로 인한 애플리케이션 문제

1920x1080이 아닌 전체 화면 해상도를 사용하려고 하면 애플리케이션이 동결되거나 충돌하거나 잘못 렌더링될 수 있습니다. 경계 없는 전체 화면 창을 사용하여 애플리케이션을 실행하고 해상도 변경을 시도하지 않는 것이 좋습니다.

스트림 세션 시작 시 애플리케이션이 종료됩니다.

스트림 세션이 시작될 때 애플리케이션이 즉시 종료되는 경우 다음에서 잠재적 원인과 해결 방법을 검토하세요.

- 런타임을 확인합니다. Amazon GameLift Streams 애플리케이션 구성에서 실행 파일 시작 경로에 지정한 파일이 실행 파일 또는 스크립트이고 선택한 런타임 환경에 맞는지 확인합니다. Windows 애플리케이션의 파일 형식은 ".exe", ".bat" 또는 ".cmd"여야 하며 Windows 또는 Proton 런타임을 대상으로 해야 합니다. 네이티브 Linux 애플리케이션은 Ubuntu 22.04 LTS 런타임을 대상으로 하는 실행 파일이어야 합니다.
- 필요한 DLLs 확인합니다. Windows 애플리케이션에 필수 DLLs. 예를 들어 애플리케이션이 디버그 빌드인 경우 Microsoft C 및 C++(MSVC) 런타임 라이브러리의 디버그 버전이 필요합니다. 이 문제를 해결하려면 빌드와 DLLs side-by-side 패키징하는 것이 좋습니다. 지침은 [Microsoft에서 디버그 실행 파일을 실행하기 위한 테스트 머신 준비를](#) 참조하세요.

일반적으로 Amazon GameLift Streams를 사용하기 전에 먼저 클린 머신에서 빌드를 테스트하는 것이 좋습니다. Amazon EC2 인스턴스 테스트에 대한 지침은 [섹션을 참조하세요](#) [원격 시스템 설정](#).

Unreal Engine 애플리케이션이 충돌하거나 추가 종속성이 필요함

Unreal Engine 애플리케이션이 시작되지 않거나 충돌하거나 Microsoft C 및 C++(MSVC) 런타임과 같은 추가 종속성을 설치해야 하는 경우 다음을 시도하세요.

- 올바른 실행 파일을 사용합니다. 애플리케이션이 Amazon GameLift Streams에서 올바르게 작동하려면 애플리케이션 경로를 Binaries/Win64/ (또는 유사한) 하위 폴더에 있는 전체 실행 파일로 설정합니다. Unreal Engine은 폴더의 루트에 있는 작은 부트스트랩 실행 파일과 Binaries/Win64/ 하위 폴더에 있는 플랫폼 대상 실행 파일이라는 두 가지 실행 파일을 생성합니다. 사전 조건을 검증하려는 루트의 부트스트랩 실행 파일이 올바르고 Amazon GameLift Streams에서 애플리케이션 시작을 방해하는 거짓 긍정을 생성할 수 있습니다. 플랫폼 대상 실행 파일이 누락된 경우 애플리케이션이 올바르게 빌드되지 않았을 수 있습니다. 예를 들어 샘플 Unreal 애플리케이션의 다음 폴더 구조를 참조하세요.

```
BuildApp
|-> MyUnrealApp.exe
|-> MyUnrealApp
    |-> Binaries
        |-> Win64
            |-> MyUnrealApp.exe
```

- Unreal Engine Asserts를 끕니다. check, verify 및 ensure 매크로를 비활성화합니다. 애플리케이션이 충돌 덤프를 생성하지 못하게 할 수 있습니다. 자세한 내용은 [Unreal Engine 설명서의 어설트를 참조하세요](#).
- 빌드 `USE_CHECKS_IN_SHIPPING=0`에서를 정의하여 check 및 verify 매크로를 비활성화합니다.
- `-handleensurepercent=0` 명령줄 인수를 사용하여 ensure 매크로를 비활성화합니다.

성능 문제

이 섹션에서는 Amazon GameLift Streams에서 실행할 때 게임 성능 문제의 잠재적 원인을 식별하고 서비스에서 스트림을 최적화하기 위한 제안을 제공합니다.

Amazon GameLift Streams에서 스트리밍할 때 게임 성능이 저하됩니다.

게임이 자체 머신에서 잘 실행되지만 Amazon GameLift Streams에서 스트리밍할 때 성능 문제가 발생하는 경우 다음을 고려하세요.

- 시스템에 Amazon GameLift Streams보다 더 강력한 하드웨어가 있을 수 있습니다. Amazon GameLift Streams가 사용하는 하드웨어와 비슷한 성능의 시스템에서 애플리케이션을 테스트해야 합니다.
 - gen4n: NVIDIA RTX 2060 GPU와 유사
 - gen5n: NVIDIA RTX 3080 GPU와 유사
 - gen6n: NVIDIA RTX 4060 GPU와 유사

이렇게 하면 애플리케이션의 렌더링 설정이 GPU와 호환되고 성능이 사용자의 기대치를 충족하는지 확인할 수 있습니다.

- 문제는 네트워크 연결 또는 Amazon GameLift Streams의 설정 때문일 수 있습니다. [스트림 연결 문제](#) 섹션의 문제 해결 팁을 시도해 보세요.

로컬에서 실행할 때도 게임 속도가 느리면 성능을 최적화해야 합니다. 최상의 최적화 방법은 사용 중인 특정 엔진 또는 프레임워크에 따라 달라집니다.

- Unreal Engine 게임은 단원을 참조하십시오 [Unreal Engine 성능 프로파일링](#).

Windows 애플리케이션에서 로드 시간이 느리거나 문제가 발생함

게임의 로드 시간이 길거나 동작이 끊기는 경우 다음과 같은 조치를 취하는 것이 좋습니다.

1. 콘텐츠 및 셰이더 성능 최적화에 대한 엔진 공급업체의 지침을 사용하여 애플리케이션이 로드 성능에 맞게 패키징되고 최적화되었는지 확인합니다.
2. 애플리케이션이 스트림 그룹의 [기본 애플리케이션](#)으로 설정되어 있는지 확인합니다.
3. 애플리케이션 패키징의 일부로 셰이더를 캐싱하여 서비스에서 애플리케이션 최초 시작을 최적화합니다.

셰이더 캐싱을 활성화하는 방법에는 두 가지가 있습니다.

- 드라이버 기반 캐싱 -이 접근 방식은 런타임 환경 GPU 및 드라이버 버전에 따라 다릅니다. 이 옵션은 모든 애플리케이션에 적용할 수 있으므로 기본 권장 접근 방식입니다. 이 접근 방식의 단계는 모든 GPU/드라이버 조합에 대해 복제해야 합니다.
- 엔진 기반 캐싱 -이 접근 방식은 가능한 경우 게임 엔진을 통한 셰이더 캐싱을 활성화합니다. 이는 개발자에게 사전 베이킹된 파이프라인 상태 객체(PSO) 캐시를 생성하는 부담을 줍니다. 또한 엔진이 동일한 GPU 하드웨어에서 서로 다른 드라이버에 대한 캐시 지원을 처리할 수 있다고 가정합니다.

주어진 엔진에 대해 PSO 캐싱이 구현되는 방식을 깊이 이해할 필요가 없으므로 드라이버 기반 캐싱을 먼저 구현하는 것이 좋습니다.

이러한 구현을 통해 셰이더 파일을 애플리케이션과 함께 내보내고 패키징할 수 있으므로 새 스트림이 시작될 때마다 파일을 생성할 필요가 없습니다.

Windows 런타임 애플리케이션에 대한 드라이버 기반 캐싱 수정 사항을 구현하려면

1. 기본 애플리케이션 스트리밍을 시작하고 광범위하게 재생하여 애플리케이션용 셰이더를 생성합니다.

⚠ Important

환경의 모든 영역 또는 수준을 방문하여 가능한 한 많은 셰이더를 생성해야 합니다.

2. 스트림을 닫기 전에 활성 스트림 세션에서 내보내기 기능을 활성화합니다. 자세한 내용은 [스트림 세션 파일 내보내기](#)를 참조하세요.
3. 애플리케이션 메뉴에서 종료하거나 애플리케이션의 종료 명령을 사용하여 애플리케이션을 정상적으로 닫습니다. 이렇게 하면 셰이더 캐시를 내보낼 준비가 됩니다.
4. 내보내기 기능을 활성화할 때 지정한 Amazon S3 버킷에서 스트림 세션 내보내기 .zip 파일을 다운로드합니다. 세션 페이지의 Amazon GameLift Streams 콘솔에서 다운로드 링크를 찾을 수 있습니다.
5. 스트림 세션 내보내기 내에서 셰이더 폴더를 찾습니다. 일반적으로이 위치인에 저장됩니다AppData\Local\NVIDIA\DXCache. 생성된 셰이더 파일(*.nvph)을 애플리케이션의 Amazon S3 버킷에 업로드합니다.
6. 런타임 시 셰이더 .bat 파일을 NVIDIA 캐싱 폴더에 복사할 파일을 생성합니다. 이 폴더는 일반적으로 있습니다C:\Users\Administrator\AppData\Local\NVIDIA\DXCache. Amazon S3 애플리케이션 버킷에 .bat 파일을 업로드합니다.
7. .bat 파일을 실행 가능한 경로로 사용하여 새 Amazon GameLift Streams 애플리케이션을 생성합니다.

애플리케이션이 스트리밍을 시작하면 .bat 파일은 애플리케이션을 시작하기 전에 미리 생성된 셰이더를 셰이더 캐시에 복사하여 스트림 로드 성능을 개선합니다.

i Note

애플리케이션을 업데이트하거나 Amazon GameLift Streams 애플리케이션을 새 스트림 그룹에 연결할 때마다이 단계를 반복해야 할 수 있습니다. 최신 스트림 그룹에는 서비스의 업데이트된 GPU 드라이버가 포함될 수 있습니다.

다음 예제 .bat 파일은 셰이더 파일이 Amazon S3 버킷 접두사에 저장되어 있다고 가정합니다Shaders\. 다른 폴더 구조를 사용할 수 있습니다.

```
@echo off
set CURRENT_PATH=%cd%
set DXCACHE_DIR=%CURRENT_PATH%\Shaders
```

```
set NVIDIA_DXCACHE_DIR=C:\Users\Administrator\AppData\Local\NVIDIA\DXCache

if not exist "%NVIDIA_DXCACHE_DIR%" (
    mkdir "%NVIDIA_DXCACHE_DIR%"
)

xcopy /s /f "%DXCACHE_DIR%" "%NVIDIA_DXCACHE_DIR%"

start %CURRENT_PATH%\app.exe
```

Proton 런타임 애플리케이션에 대한 드라이버 기반 캐싱 수정 사항을 구현하려면

1. 다음 환경 변수 재정의로 기본 애플리케이션 스트리밍을 시작합니다.

```
"__GL_SHADER_DISK_CACHE_PATH" : "/home/unpriv/games"
```

2. 애플리케이션을 광범위하게 재생하여 셰이더를 생성합니다.

Important

환경의 모든 영역 또는 수준을 방문하여 가능한 한 많은 셰이더를 생성해야 합니다.

3. 스트림을 닫기 전에 활성 스트림 세션에서 내보내기 기능을 활성화합니다. 자세한 내용은 [스트림 세션 파일 내보내기](#)를 참조하세요.
4. 애플리케이션 메뉴에서 종료하거나 애플리케이션의 종료 명령을 사용하여 애플리케이션을 정상적으로 닫습니다. 이렇게 하면 셰이더 캐시를 내보낼 준비가 됩니다.
5. 내보내기 기능을 활성화할 때 지정한 Amazon S3 버킷에서 스트림 세션 내보내기 .zip 파일을 다운로드합니다. 세션 페이지의 Amazon GameLift Streams 콘솔에서 다운로드 링크를 찾을 수 있습니다.
6. 스트림 세션 내보내기 내에서 셰이더 폴더 및 파일을 찾습니다.
 - a. application\GLCache 폴더
 - b. 애플리케이션에서 DX11을 사용하는 경우: application*path-to-exe*\exe-*name*.dxvk-cache 파일
 - c. 애플리케이션에서 DX12를 사용하는 경우: application*path-to-exe*\vkd3d-proton.cache.write 파일
7. 생성된 셰이더 파일을 애플리케이션의 Amazon S3 버킷에 업로드합니다.
 - a. GLCache 폴더를 애플리케이션의 루트 디렉터리에 복사합니다.

- b. 사용 가능한 경우 `.dxvk-cache` 또는 `vkd3d-proton.cache.write` 캐시 파일을 애플리케이션 실행 파일이 포함된 폴더에 복사합니다.
8. 동일한 Proton 구성으로 새 Amazon GameLift Streams 애플리케이션을 생성합니다.
9. 동일한 환경 변수 재정의로 애플리케이션을 실행합니다.

```
"__GL_SHADER_DISK_CACHE_PATH" : "/home/unpriv/games"
```

애플리케이션이 스트리밍을 시작하면 미리 생성된 셰이더를 사용하여 스트림 로드 성능을 개선합니다.

Note

애플리케이션을 업데이트하거나 Amazon GameLift Streams 애플리케이션을 새 스트림 그룹에 연결할 때마다 단계를 반복해야 할 수 있습니다. 최신 스트림 그룹에는 서비스의 업데이트된 GPU 드라이버가 포함될 수 있습니다.

Unreal Engine을 사용하여 애플리케이션에 대한 엔진 기반 캐싱 수정을 구현하려면

이 접근 방식에서는 Unreal Engine 기능을 사용하여 Amazon GameLift Streams 애플리케이션에 대한 파이프라인 상태 객체(PSO) 캐시를 생성할 수 있습니다. PSO 캐시를 사용하면 런타임 컴파일 시간이 줄어 사전 컴파일된 그래픽 파이프라인 상태를 제공할 수 있으므로 로드 및 렌더링 중에 히치를 줄일 수 있습니다. 이를 위해서는 Unreal Engine에 대한 고급 지식이 필요하므로 여기에서 모든 엔진별 세부 정보를 다루지는 않습니다. 추가 지침은 [번들링된 PSO 캐시 생성](#), "컬렉션 흐름" 섹션의 Unreal Engine의 지침을 참조하세요.

1. PSO 로깅이 활성화된 애플리케이션의 셰이더를 생성합니다.
 - a. PSO 지원 애플리케이션과 함께 패키지 빌드를 사용하여 새 Amazon GameLift Streams 애플리케이션을 생성합니다.
 - b. PSO 로깅 앱에서 `-logPSO` 명령으로 스트림을 시작합니다. Amazon GameLift Streams 콘솔의 스트림 구성 테스트 페이지에서 명령줄 인수 옵션을 사용할 수 있습니다.

Important

환경의 모든 영역 또는 수준을 방문하여 가능한 한 많은 셰이더를 생성해야 합니다.

- c. 스트림을 닫기 전에 활성 스트림 세션에서 내보내기 기능을 활성화합니다. 자세한 내용은 [스트림 세션 파일 내보내기](#)를 참조하세요.
 - d. 메뉴에서 또는 Unreal 종료 명령을 사용하여 애플리케이션을 종료합니다. 스트림을 직접 닫으면 Unreal 셰이더 파이프라인 파일이 생성되지 않습니다.
 - e. 내보내기 단계에서 지정한 Amazon S3 버킷에서 스트림 세션 내보내기 .zip 파일을 다운로드합니다. 세션 페이지의 Amazon GameLift Streams 콘솔에서 다운로드 링크를 찾을 수 있습니다.
2. Unreal 셰이더 파이프라인 파일을 Amazon GameLift Streams 애플리케이션에 패키징합니다.
 - a. 의 스트림 세션 내보내기에서 기록된 PSO 파일(rec.pipelinecache)을 찾습니다 Saved/CollectedPSOs. Unreal 명령을 사용하여 PSO 파일의 압축을 풉니다.
 - b. 압축 해제에서 생성된 출력으로 새 Unreal 빌드를 패키징합니다. Unreal 지침의 [PSO 캐시 변환 및 애플리케이션에 PSO 캐시 포함](#) 섹션을 따릅니다.

⚠ Important

"PSO 캐시 변환" 섹션에서 Unreal 명령을 실행할 때는 동일한 드라이버의 버전 입력 파일을 사용해야 합니다. 예: DX12의 경우 SM6 파일만 입력으로 사용합니다. 그렇지 않으면 새 애플리케이션을 패키징할 때 오류가 발생합니다.

- c. PSO 파일을 사용하여 새 패키지 빌드에 대한 새 Amazon GameLift Streams 애플리케이션을 생성합니다.
- d. 스트림을 시작하고 테스트할 때 PSO 캐시가 로드되고 있는지 확인합니다. 게임 로그에서 다음 줄을 확인합니다.

```
Opened FPipelineCacheFile: ../../../../...
```

i Note

애플리케이션을 업데이트하거나 Amazon GameLift Streams 애플리케이션을 새 스트림 그룹에 연결할 때마다 단계를 반복해야 할 수 있습니다. 최신 스트림 그룹에는 서비스의 업데이트된 GPU 드라이버가 포함될 수 있습니다.

스트림 연결 및 네트워크 성능 문제

[Amazon GameLift Streams 백엔드 서비스를 설정할 때](#) 다음을 확인하세요.

- 최종 사용자에게 AWS 리전 가장 가까운를 선택합니다. 클라이언트에서 스트림을 호스팅하는 리전까지의 지연 시간이 길면 스트림 품질에 영향을 미칠 수 있습니다. 스트리밍할 수 있는 위치 [AWS 리전 Amazon GameLift Streams에서 지원하는 및 스트리밍 위치](#) 목록은 섹션을 참조하세요. 리전에서 AWS 콘솔 엔드포인트를 ping하여 대략적인 지연 시간 측정값을 얻을 수 있습니다.
- 스트림 그룹에 새 스트림에 대한 용량이 있는지 확인합니다.
- 최종 사용자가 웹 클라이언트 시간이 초과되기 전에 연결할 수 있도록 ConnectionTimeoutSeconds가 충분히 설정되어 있는지 확인합니다.

최종 사용자에게 다음을 확인하도록 권고합니다.

- 방화벽이 UDP 포트 범위 33435-33465에 대한 액세스를 허용하여 Amazon GameLift Streams에서 스트리밍을 허용하는지 확인합니다. Amazon GameLift Streams가 이러한 포트에 연결할 수 없는 경우 검은색 또는 회색 화면과 같은 스트리밍 문제가 발생할 수 있습니다.
- 인터넷 연결이 1080p 스트림에 대해 10Mbps 이상의 연결 속도를 유지할 수 있는지 확인합니다. Amazon GameLift Streams에서 재생하는 동안 네트워크 문제를 감지하면 인터넷 속도가 변동하고 최소 10Mbps를 일관되게 유지하지 못할 수 있습니다. 인터넷 속도 테스트를 실행하고 문제 해결 단계를 계속 진행합니다.
- 가능하면 유선 네트워크를 사용합니다. Wi-Fi를 사용할 때는 신호 강도를 높이기 위해 디바이스를 라우터 가까이로 이동합니다.
- 2.4GHz 및 5GHz 대역 모두에서 Wi-Fi 라우터를 사용하는 경우 다른 대역에 연결해 보세요. 라우터를 다른 밴드로 전환하는 방법을 잘 모르는 경우 Wi-Fi 라우터의 제조업체 또는 공급자의 지원 페이지를 방문하세요. 고객 서비스에 문의할 수도 있습니다.
- 동일한 네트워크(특히 홈 Wi-Fi)의 다른 사용자가 비디오 스트리밍, 다운로드, 온라인 게임 또는 백업과 같은 고대역폭 애플리케이션을 실행하고 있는지 확인합니다.
- 디바이스에서 대역폭을 차지하는 다른 애플리케이션을 닫습니다.
- 스트리밍 중에는 VPN 또는 프록시를 사용하지 마십시오. 지연 시간이 길어지고 게임 플레이에 영향을 미칠 수 있습니다.
- iPad 또는 iPhone에서 재생할 때 셀룰러 네트워크 대신 Wi-Fi를 사용하고 있는지 확인합니다. 셀룰러 네트워크를 사용하면 연결 문제가 발생할 수 있습니다.
- MacOS 사용자는 위치 서비스를 비활성화해야 합니다. 그러면 Wi-Fi가 때때로 일시 중지되어 스트리밍 환경이 저하됩니다.

스트림 입력 문제

이 섹션에서는 스트림 세션의 사용자 입력과 관련된 문제의 잠재적 원인과 해결 방법을 설명합니다.

일반 입력 문제 해결

- 테스트하여 문제가 브라우저 지정인지 확인합니다. 전반적으로 최상의 최종 사용자 경험과 특히 게임 컨트롤러와의 호환성을 위해 Google Chrome, Microsoft Edge 또는 사용자 지정 Chromium 기반 데스크톱 애플리케이션을 사용하는 것이 좋습니다.
- 클라이언트에서 전송되고 애플리케이션에서 수신한 입력 이벤트를 로깅하여 프런트 엔드 코드에 입력 불일치가 있는 위치를 식별합니다.
- 알려진 문제 및 제한을 포함하여 지원되는 브라우저 및 입력 디바이스에 대한 [지원되는 브라우저 및 입력](#) 추가 정보를 확인해야 합니다.

게임패드 및 마이크 입력은 네이티브 Linux 애플리케이션에서 작동하지 않습니다.

게임패드 및 마이크 입력은 네이티브 Linux 애플리케이션에서 지원되지 않습니다. 알려진 문제 및 제한 사항을 포함하여 지원되는 입력 디바이스에 [지원되는 브라우저 및 입력](#) 대한 자세한 내용은 섹션을 참조하세요.

MacOS 클라이언트에서 키 입력이 멈춘 것처럼 보임

MacOS 클라이언트에서 명령 한정자 키와 다른 키를 동시에 누르면 키가 갑자기 멈춰서 키 이벤트를 반복하는 것처럼 보일 수 있습니다. 예를 들어 명령 키도 누르면 화살표 키가 멈출 수 있습니다. 게임에서 화살표 키를 사용하여 카메라를 회전하면 카메라가 무한히 회전합니다.

- 문제: MacOS의 명령 키는 Microsoft Windows의 Windows 키에 매핑되는 Meta 키 이벤트에 매핑됩니다. 이 문제는 명령과 다른 키를 동시에 눌렀을 때 MacOS 브라우저에 영향을 미치는 [버그](#)입니다. 여기서 Meta 키는 릴리스 시 재설정되지만 브라우저가 화살표 키에 대한 키업 이벤트를 캡처하지 않았기 때문에 화살표 키는 재설정되지 않으므로 Web SDK 클라이언트가 서버에 키업 이벤트를 보내지 않고 스트리밍 애플리케이션은 여전히 키가 눌러지고 있다고 생각할 것입니다.
- 해결 방법: 명령 키를 사용하지 않는 경우 웹 SDK의 InputConfiguration 인터페이스에 있는 웹 SDK 키보드 필터 메커니즘(keyboardFilter)을 사용하여 필터링할 수 있습니다.

OS UI 요소를 열 때 입력 고정

데스크톱 및 모바일 브라우저 클라이언트에서는 특정 OS 수준 UI 요소에 우선 순위가 있는 경우 키 릴리스와 같은 입력 이벤트가 처리되지 않습니다. 이렇게 하면 키를 해제했다라도 키가 여전히 정지되어 있는 것처럼 캐릭터가 움직이거나 작업이 반복될 수 있습니다.

- 문제: 특정 OS 수준 UI 요소(예: 데스크톱의 브라우저 메뉴 모음 또는 iOS의 Control Center 및 Notification Center)를 열면 브라우저가 블러 또는 포커스 이벤트를 트리거하지 않고 입력 이벤트 실행을 중지합니다. 이렇게 하면 서버가 마지막 입력 상태를 계속 수신합니다. 이는 안정적으로 감지할 수 없는 브라우저 수준 제한입니다.
- 해결 방법: 데스크톱 브라우저에서 전체 화면 모드를 사용하여 브라우저 메뉴 표시줄에 대한 액세스를 방지합니다. 키보드가 연결된 iOS 사용자의 경우 네이티브 앱이 포커스 손실을 더 잘 감지하고 처리할 수 있는 웹 보기로 네이티브 앱 래퍼를 생성하여 브라우저 창 포커스 및 블러 이벤트를 명시적으로 트리거하는 것이 좋습니다. 또는 프론트 엔드 HTML 또는 게임 내 UI 요소를 사용하여 사용자에게 키가 여전히 눌러져 있음을 알리고 이 iOS 제한에 대한 정보를 제공합니다.

마우스 움직임은 Amazon GameLift Streams에서 다르게 동작합니다.

Amazon GameLift Streams로 스트리밍할 때 예상보다 빠르게 이동하는 등 마우스 움직임이 다르게 동작하는 경우 애플리케이션에서 마우스 처리 및 커서 관리 로직을 조정해야 할 수 있습니다.

- 문제: Amazon GameLift Streams는 휴리스틱을 사용하여 마우스 이벤트를 “상대적” 또는 “절대적” 모드로 전송할지 여부를 선택합니다. 상대 모드에서 새 마우스 업데이트는 이전 업데이트와 작은 증분 차이로 제공됩니다. 절대 모드에서 마우스 커서는 클라이언트와 동기화된 화면 위치로 계속 강제됩니다. 스트리밍된 콘텐츠 위에 운영 체제 커서가 표시되면 휴리스틱은 항상 절대 좌표를 선택합니다. 이로 인해 애플리케이션이 작고 상대적인 업데이트를 기대하는 경우 예기치 않게 큰 이동 델타가 발생할 수 있습니다.
- 해결 방법: 애플리케이션에서 상대적 마우스 모션(예: FPS 스타일 카메라 제어 또는 드래그 기반 상호 작용)이 필요한 경우 마우스 상호 작용 중에 운영 체제 커서를 숨깁니다. 예를 들어 마우스 아래로 커서를 숨기고 마우스 위로 다시 표시합니다. 이렇게 하면 끌기 모션이 상대 좌표를 사용하고 절대 위치는 버튼을 놓을 때만 동기화됩니다.

Amazon GameLift Streams의 마우스 움직임에 대한 자세한 내용은 섹션을 참조하세요 [마우스 이동 처리](#).

스트림 세션 문제

이 섹션에서는 예기치 않게 시작되거나 종료되는 스트림 세션과 관련된 문제의 잠재적 원인과 해결 방법을 식별합니다.

스트림 세션이 시작되지 않음

잠재적 원인:

- 애플리케이션이 중단되거나 충돌합니다. 문제 해결 지침은 [애플리케이션 문제](#) 섹션을 참조하세요.
- 스트림 그룹 상태가 `Active`가 아닙니다. 스트림 그룹의 상태를 확인합니다.
- 온디맨드 용량이 [StartStreamSession](#) API에서 `ConnectionTimeoutSeconds`에 지정된 제한 시간보다 오래 걸립니다. Windows 런타임에서 온디맨드 스핀 업 시간은 5분 이상 걸릴 수 있습니다.
- 스트리밍 위치에 사용 가능한 용량이 없습니다. 할당된 용량이 사용 중인 용량보다 크거나 사용 중이 아닌 온디맨드 용량이 있는지 확인합니다(할당된 용량이 항상 켜져 있는 용량 + 온디맨드 용량보다 작음). 콘솔의 스트림 그룹 목록 또는 스트림 그룹 세부 정보 페이지에서 이러한 값을 찾을 수 있습니다. 서비스 API를 사용하면 [GetStreamGroup](#)을 사용하여 이러한 값을 찾을 수 있습니다. 사용 가능한 용량이 일시적으로 0인 몇 가지 시나리오는 다음과 같습니다.
- 스트리밍 위치에서 상시 작동 용량을 방금 늘린 경우 용량이 할당될 때까지 몇 분 정도 기다립니다.
- 스트리밍 위치에 사용 가능한 용량이 하나뿐이고 클라이언트가 예기치 않게 연결 해제된 경우에도 이전 세션은 여전히 연결 해제 상태일 수 있습니다. 세션이 시간 초과될 때까지 몇 분 정도 기다렸다가 다시 시도하세요.
- 최근에 스트림 그룹에 위치를 추가했지만 해당 위치에 애플리케이션이 없는 경우 애플리케이션이 해당 위치 복제를 완료하지 않았을 수 있습니다. 콘솔의 스트림 그룹 세부 정보 페이지에서 복제 상태를 확인합니다. 또는 [GetApplication](#) API를 사용하여 `ReplicationStatuses` 값을 확인하여 원하는 스트리밍 위치 `Status`의가 인지 확인할 수 있습니다 `COMPLETED`.
- 네트워크 조건이 너무 좋지 않아 프레임, 특히 첫 번째 프레임이 전송되지 않습니다. 클라이언트와 스트리밍 위치 간의 네트워크 상태를 확인하고 다른 위치를 조정하거나 시도합니다.

스트림 세션 종료됨

애플리케이션이 충돌하거나 종료되거나 클라이언트 연결이 끊어지면 스트림 세션이 자동으로 종료됩니다. 세션은 다음과 같은 제한 시간 값으로 인해 종료될 수도 있습니다.

- 배치 제한 시간: Amazon GameLift Streams가 스트림 세션을 호스팅할 컴퓨팅 리소스를 찾는 제한 시간 값입니다.
- 연결 제한 시간: 클라이언트가 스트림 세션에 연결하거나 다시 연결하는 제한 시간 값입니다.
- 유휴 제한 시간: 사용자 입력 없이 스트림 세션을 유휴 상태로 유지할 수 있는 최대 시간입니다.
- 세션 길이 제한 시간: 스트림 세션의 최대 시간입니다.

각 제한 시간 및 가능한 값에 대한 자세한 설명은 섹션을 참조하세요 [스트림 세션에 영향을 미치는 제한 시간 값](#).

Proton for Amazon GameLift Streams와의 호환성 테스트 및 문제 해결

Amazon GameLift Streams 애플리케이션이 Proton 런타임 환경에서 실행되는 경우가 이 섹션에서는 애플리케이션과 Proton 계층 간의 호환성 문제를 해결하는 데 도움이 될 수 있습니다. 이 지침에는 Amazon GameLift Streams가 사용할 환경을 시뮬레이션하여 자체 시스템에 Proton을 설치하는 스크립트 세트가 포함되어 있습니다. Amazon GameLift Streams 서비스 없이 문제를 해결하면 애플리케이션 및 런타임 환경과 관련된 문제를 해결하는 데 집중할 수 있습니다.

테스트 및 문제 해결을 위한 상위 수준 단계

1. Ubuntu 22.04 LTS 시스템을 획득합니다. 로컬 시스템 또는 Amazon EC2 클라우드 기반 데스크톱을 사용할 수 있습니다. 지침을 보려면 다음 주제 중에서 선택합니다.
 - [로컬 시스템 설정](#)
 - [원격 시스템 설정](#)
2. Proton 런타임 환경을 설치하여 애플리케이션을 테스트하고 디버깅합니다. 지침은 단원 [Proton 문제 해결](#)을 참조하십시오.

Proton과 관련된 알려진 문제

최신 호환성 및 문제 해결 리소스는 [Proton GitHub wiki](#)를 참조하세요. Proton GitHub 문제 [트래커에서 문제를](#) 검색할 수도 있습니다. 다음은 고객이 Proton에서 Windows 애플리케이션을 실행할 때 겪은 몇 가지 특정 문제입니다.

Proton의 Godot 애플리케이션

- Amazon Vulkan 캡처 계층이 활성화된 경우 Proton에서 실행되는 Godot 기반 애플리케이션에 검은 색 화면이 표시될 수 있습니다. 이 문제를 완화하려면 환경 변수를 설정하여 스트리밍할 때 공유 텍스처를 비활성화합니다. `VK_LAYER_AMZN_BLITSURFACE_SHARED_TEXTURES=0`.

Proton의 Unreal Engine 애플리케이션

- Proton 8.x에서 Electra Media Player(Unreal Engine 플러그인)와 관련된 문제가 발생하는 경우 <https://github.com/ValveSoftware/wine/pull/257> 있는 수정 사항을 사용하는 것이 좋습니다.

Proton 문제를 해결하도록 로컬 시스템 설정

Proton은 Linux에서 Windows 애플리케이션을 실행할 수 있는 호환성 계층입니다. 따라서 테스트하고 문제를 해결할 수 있는 Ubuntu 머신이 있어야 합니다. 로컬 Ubuntu 머신이 없는 경우 Amazon EC2를 사용하여 원격 머신을 설정할 수 있습니다. 이렇게 하려면 [원격 시스템 설정](#) 대신의 단계를 따르세요.

사전 조건

- [Ubuntu 22.04 LTS](#). 설치 지침의 경우 Ubuntu의 [Ubuntu 데스크톱 설치](#) 자습서를 사용할 수 있습니다.
- NVIDIA GPU

GPU 드라이버 설치

최신 GPU 드라이버를 설치하면 애플리케이션의 성능 저하 및 충돌을 방지할 수 있습니다.

시스템에서 사용하는 GPU 드라이버를 확인하려면

1. 터미널에서 다음 명령을 실행합니다.

```
lshw -C display | grep driver
```

2. 올바른 드라이버가 설치된 경우 다음과 같은 출력 또는 이와 유사한 출력이 표시되어야 합니다. 여기서 `<gpu>`는 NVIDIA Nvidia용입니다. `configuration: driver=<gpu> latency=0`

최신 NVIDIA GPU 드라이버를 설치하려면

[NVIDIA 드라이버 설치](#)의 지침을 따릅니다.

GPU 드라이버 확인

GPU 드라이버가 설치되어 올바르게 작동하는지 확인합니다. 이를 확인하는 한 가지 방법은 터미널에서 [vkcube](#) 애플리케이션을 실행하는 것입니다.

1. 다음 명령을 사용하여 `vulkan-tools` apt 패키지를 설치합니다.

```
sudo apt install -y vulkan-tools
```

2. `vkcube`을(를) 실행합니다.

3. 출력 결과를 검토합니다.

- 시스템이 올바른 GPU를 올바르게 사용하고 있는 경우 GPU 이름과 함께 다음과 비슷한 출력이 표시됩니다. `Selected GPU 0: AMD Radeon Pro V520 (RADV NAVI12), type: 2`
- 애플리케이션이 GPU를 올바르게 사용할 수 없는 경우 다음과 비슷한 다른 출력이 표시될 수 있습니다. `Selected GPU 0: llvmpipe (LLVM 15.0.7, 256 bits), type: 4`

이 경우 GPU 드라이버를 확인하고 필요한 경우 다시 설치합니다.

다음 단계

로컬 Ubuntu 시스템을 준비한 상태에서 다음 단계는 Proton을 설정하는 것입니다. 지침은 단원을 참조하십시오 [Proton 문제 해결](#).

Proton 문제를 해결하기 위한 원격 Amazon EC2 시스템 설정

로컬 Ubuntu 머신이 없는 경우 다음 지침에 따라 대신 원격 머신을 설정합니다.

이 단계에서는 Amazon GameLift Streams용 Proton과 애플리케이션의 호환성 문제를 해결하는 데 사용할 Amazon Elastic Compute Cloud(Amazon EC2)를 사용하여 원격 Ubuntu 시스템을 설정합니다. 이 주제에서는 Ubuntu 22.04 LTS, 필요한 GPU 드라이버 및 시각적 원격 데스크톱용 Amazon DCV 서버를 사용하여 Amazon EC2 인스턴스를 설정하는 방법을 설명합니다.

Ubuntu 22.04 LTS AMI를 사용하여 Amazon EC2 인스턴스 시작

1. AWS Management Console에서 Amazon EC2로 이동합니다.
2. 인스턴스 시작을 선택합니다.
3. 이름에 "Amazon GameLift 스트림 테스트"를 입력합니다.

4. 애플리케이션 및 OS 이미지(Amazon Machine Image)에 대해 Ubuntu Server 22.04 LTS(HVM)를 선택합니다.
5. 인스턴스 유형에서 g4dn.2xlarge를 선택합니다.
6. SSH를 사용하여 인스턴스에 액세스하려면 키 페어(로그인)에서 키 페어를 선택합니다. 를 사용하여 인스턴스에 연결하려면 AmazonSSManagedInstanceCore 정책과 함께 인스턴스 프로파일을 사용하는 것이 좋습니다 AWS Systems Manager Session Manager. 자세한 내용은 [기존 IAM 역할에 세션 관리자 권한 추가를 참조하세요](#).
7. 네트워크 설정에서 새 보안 그룹을 생성합니다.
8. 보안 그룹 이름에 DCV를 입력합니다.
9. Amazon DCV를 사용하여 액세스를 허용 Anywhere하려면 유형 Custom TCP8443, 포트 범위 및 소스 유형이 있는 인바운드 보안 그룹 규칙을 추가합니다.
10. 스토리지를 256GiB 이상으로 늘리고 스토리지 유형으로 gp3를 선택합니다.
11. 인스턴스 시작을 선택합니다.

이제 인스턴스가 시작됩니다.

[Linux 인스턴스에 연결](#)의 지침에 따라 SSH 또는 를 사용하여 인스턴스에 연결합니다 AWS Systems Manager Session Manager.

GPU 드라이버 설치

G4dn - NVIDIA GPU

다음 명령을 실행하여 추가 모듈과 Linux 펌웨어를 설치합니다.

```
sudo apt install linux-modules-extra-aws linux-firmware

# Install the AWS CLI required for NVIDIA driver installation
curl "https://awscli.amazonaws.com/awscli-exe-linux-x86_64.zip" -o "awscliv2.zip"
sudo apt install unzip
unzip awscliv2.zip
sudo ./aws/install
```

[Linux에 NVIDIA 드라이버 설치의 Ubuntu 및 Debian용 NVIDIA GRID 드라이버에 대한 지침을 따릅니다](#).

사용자 환경 설정

다음 명령을 실행하여 GPU를 사용할 수 있도록 사용자 환경을 설정합니다. 이렇게 하면 다음과 같은 작업이 수행됩니다.

- video 그룹에 추가하여 비디오 디바이스에 대한 액세스 권한을 부여하고 render 그룹에 추가하여 렌더링 디바이스에 대한 액세스 권한을 부여합니다.
- NVIDIA 드라이버와 Amazon S3에서 애플리케이션 또는 게임을 다운로드하는 데 AWS CLI 필요함을 설치합니다.

```
sudo adduser user

# Add the current user to the video and render group
sudo usermod -a -G video user
sudo usermod -a -G render user
sudo adduser user sudo

# Install the AWS CLI
curl "https://awscli.amazonaws.com/awscli-exe-linux-x86_64.zip" -o "awscliv2.zip"
sudo apt install unzip
unzip awscliv2.zip
sudo ./aws/install

sudo reboot
```

Amazon DCV 설치 및 구성

SSH 또는를 사용하여 인스턴스에 다시 연결하고 [Ubuntu용 Linux에 Amazon DCV 서버 설치](#)의 지침을 AWS Systems Manager Session Manager 따릅니다.

- 설명서에 설명된 대로 서버가 올바르게 구성되어 있는지 확인합니다.
- NVIDIA GPU용 [NVIDIA 드라이버 설치 및 구성](#)의 단계를 따릅니다.
- [서버 설치 가이드\(Ubuntu 탭으로 이동\)의 7단계](#)에 설명된 대로 비디오 그룹에 Amazon DCV 사용자를 추가합니다.

Amazon DCV 서버의 선택적 부분을 설치할 필요가 없습니다.

완료되면 다음 명령을 실행하여 Amazon DCV 서버를 시작합니다.

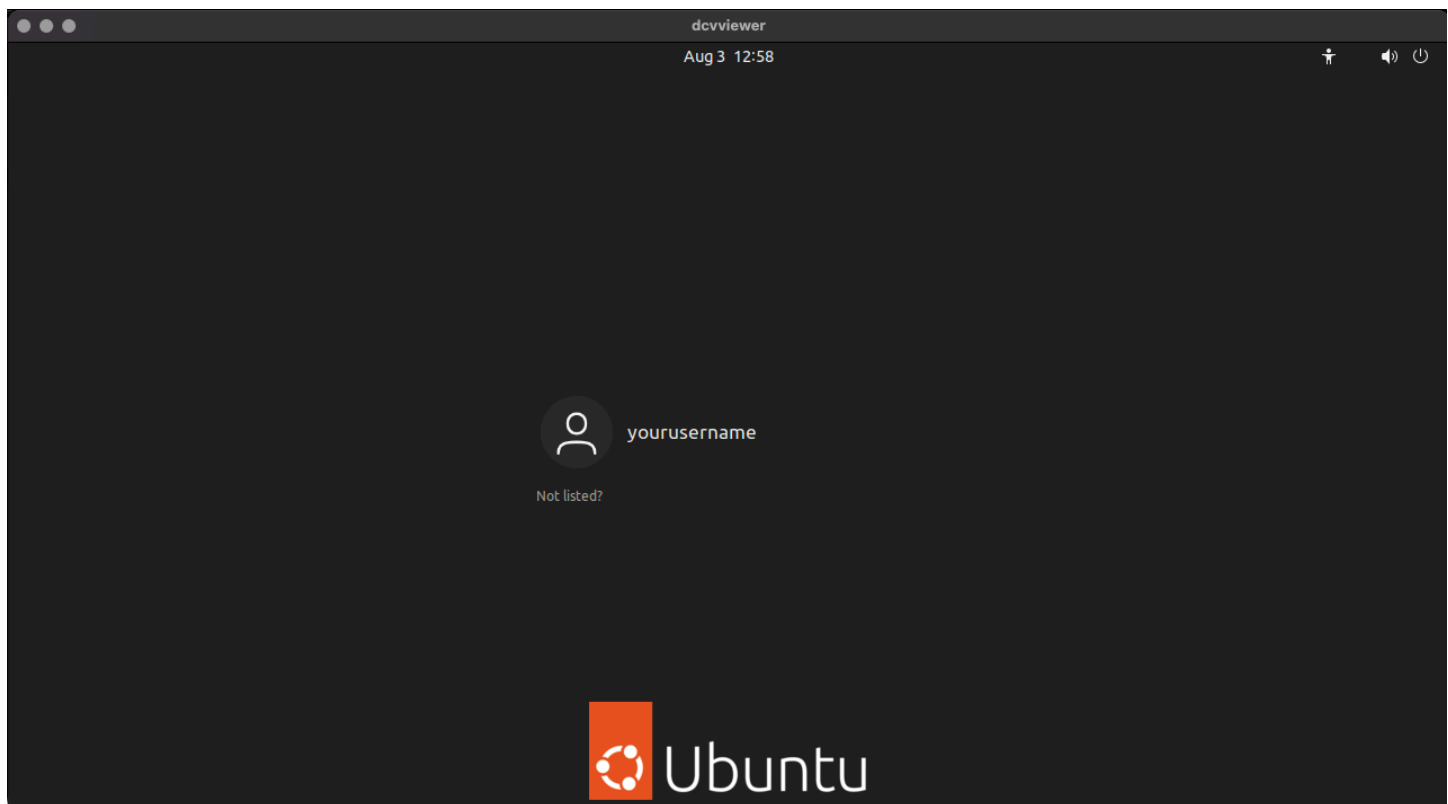
```
sudo systemctl start dcvserver
sudo systemctl enable dcvserver
```

Amazon DCV 클라이언트를 사용하여 Ubuntu Server에 연결

Ubuntu 인스턴스에 다시 연결하고 다음을 실행하여 사용자를 위한 세션을 생성합니다.

```
sudo dcv create-session --owner user --user user my-session --type console
```

이제 Amazon DCV 클라이언트를 사용하여 퍼블릭 IP 주소를 사용하여 Ubuntu 인스턴스에 액세스할 수 있습니다. Amazon DCV 클라이언트를 시작하면 시각적 표시를 통해 Ubuntu 인스턴스에 액세스할 수 있는 창이 나타납니다.



GPU 드라이버 확인

GPU 드라이버가 설치되어 올바르게 작동하는지 확인합니다. 이를 확인하는 한 가지 방법은 터미널에서 [vkcube](#) 애플리케이션을 실행하는 것입니다.

1. 다음 명령을 사용하여 `vulkan-tools` apt 패키지를 설치합니다.

```
sudo apt install -y vulkan-tools
```

2. vkcube를 실행합니다.

3. 출력 결과를 검토합니다.

- 시스템이 올바른 GPU를 올바르게 사용하고 있는 경우 GPU 이름과 함께 다음과 비슷한 출력이 표시됩니다. Selected GPU 0: AMD Radeon Pro V520 (RADV NAVI12), type: 2
- 애플리케이션이 GPU를 올바르게 사용할 수 없는 경우 다음과 비슷한 다른 출력이 표시될 수 있습니다. Selected GPU 0: llvmpipe (LLVM 15.0.7, 256 bits), type: 4

이 경우 GPU 드라이버를 확인하고 필요한 경우 다시 설치합니다.

Podman 설정(Proton만 해당)

Proton 런타임을 사용하는 경우 Proton의 빌드 프로세스에서 사용하는 컨테이너인 [Podman](#)을 설치해야 합니다. 터미널에서 다음 단계를 완료합니다.

1. Proton의 빌드 프로세스에서 사용하는 컨테이너인 Podman을 설치합니다.

```
sudo apt install podman
```

2. 및 파일에서 /etc/subgid /etc/subgid

- 파일에 Linux 시스템 사용자 이름과 ID가 나열되어 있는지 확인합니다. 파일을 열거나 cat 명령을 사용하여 파일의 내용을 확인할 수 있습니다. 형식 예제: test:165536:65536, 여기서 test는 사용자 이름에 해당합니다.
- 목록에 없는 경우 추가합니다. 형식 예제: test:165536:65536, 여기서 test는 사용자 이름에 해당합니다.

```
$ cat /etc/subuid
ceadmin:100000:65536
test:165536:65536

$ cat /etc/subgid
ceadmin:100000:65536
test:165536:65536
```

자세한 내용은 [Podman 설명서의 Basic Setup and Use of Podman in a Rootless environment](#)를 참조하세요.

다음 단계

이제 Amazon GameLift Streams와의 호환성 문제를 해결하기 위한 Amazon EC2 인스턴스 및 환경 설정이 완료되었습니다. GameLift 다음 단계는 Proton을 설정하는 것입니다. 지침은 단원을 참조하십시오 [Proton 문제 해결](#).

Proton의 호환성 문제 해결

이 단계에서는 Amazon GameLift Streams 애플리케이션과 Proton 간의 호환성 문제를 해결할 수 있도록 자체 시스템에 Proton을 설정합니다. Amazon GameLift Streams 서버 없이 시뮬레이션된 환경에서 애플리케이션을 실행하면 애플리케이션 및 런타임 환경과 관련된 문제를 식별하는 데 도움이 될 수 있습니다.

사전 조건

- GPU 드라이버가 설치된 Ubuntu 22.04 LTS. 지침은 [로컬 시스템 설정](#) 또는 단원을 참조하십시오 [원격 시스템 설정](#).

Proton 설치

Ubuntu 22.04 LTS 시스템에 Proton을 설치하려면 다음 스크립트를 사용하여 Proton [GitHub 리포지토리](#)에서 테스트하려는 Proton 버전을 복제, 빌드 및 구성합니다.

- 다음 코드를 복사하여 Ubuntu 22.04 LTS 시스템의 라는 파일에 붙여 proton-setup.sh 넣습니다.

```
#!/bin/bash
# This is a script to build Proton. The default build is a tag from the
# experimental_9.0 branch of Proton, but can be changed as a parameter to this
# script.
#
# Usage: ./proton-setup.sh [optional proton_branch_name {default:
# experimental-9.0-20241121b}]
set -e

sudo apt install -y podman make git

# clone proton from github, recurse submodules
# if no proton git link is supplied, use a default tag from the experimental_8.0
# branch
```

```
PROTON_BRANCH=${1:-"experimental-9.0-20241121b"}
PROTON_BUILD_DIR=protonBuild
PROTON_DIR=$(pwd)/proton
if git clone https://github.com/ValveSoftware/Proton.git --recurse-submodules --
branch $PROTON_BRANCH proton;
then
  echo "Successfully cloned Proton and its submodules."
else
  echo "Warning: a proton directory/repository already exists. It is recommended to
delete this folder and re-run this script unless it is a valid repository with
initialized submodules."
fi

if [ -d $PROTON_BUILD_DIR ];
then
  echo "Error: protonBuild directory already exists. Delete this folder first to
create a fresh build of Proton before re-running this script."
  exit 1
fi
mkdir $PROTON_BUILD_DIR
cd $PROTON_BUILD_DIR
$PROTON_DIR/configure.sh --enable-ccache --container-engine=podman

# build proton
echo "Building Proton"
make
echo "Done building Proton!"

# prepare proton for execution
cd dist
mkdir compatdata
if [ -e ./dist ]; then
  PROTON_FILES=dist
elif [ -e ./files ]; then
  PROTON_FILES=files
fi
cp version $PROTON_FILES/
echo "Finished installing proton. Proton binary location: $(pwd)/proton"
echo "STEAM_COMPAT_DATA_PATH: $(pwd)/compatdata"
echo "STEAM_COMPAT_CLIENT_INSTALL_PATH: anything"
```

- 이 단계에서는 Proton 설정 스크립트를 실행하여 Proton 및 추가 종속성을 복제하고 설치합니다. 스크립트는 설치하려는 Proton 버전의 태그 또는 브랜치 이름을 인수로 허용합니다. Amazon

GameLift Streams가 제공하는 Proton의 사용자 지정 빌드 중 하나를 시뮬레이션하려면 아래 해당 버전에 대한 지침을 사용합니다.

Note

GitHub에서 복제하는 데 시간이 걸릴 것으로 예상합니다. 다운로드할 수 있는 하위 모듈은 여러 기가바이트입니다.

터미널에서 스크립트를 실행하고 Proton 버전 `proton-setup.sh` 브랜치를 지정합니다.

- 기본 제공 Proton 버전

- Proton 9.0-2(PROTON-20250516)의 경우 [experimental-9.0-20241121b](#)를 사용합니다.

```
proton-setup.sh experimental-9.0-20241121b
```

- Proton 8.0-5(PROTON-20241007)의 경우 [experimental-8.0-20240205](#)를 사용합니다.

```
proton-setup.sh experimental-8.0-20240205
```

일반적으로 추가 소스 코드는 필요하지 않습니다. 그러나 Electra Media Player(Unreal Engine 플러그인)에 문제가 발생하는 경우 <https://github.com/ValveSoftware/wine/pull/257> 있는 수정 사항을 사용하는 것이 좋습니다.

Note

Proton 8.0-2c(PROTON-20230704)의 경우 Amazon GameLift Streams는 로컬에서 확장할 수 없는 독점 빌드를 사용합니다.

- 권장 사용자 지정 Proton 버전

사용자 지정 Proton 버전의 경우 Proton `experimental_8.0` 브랜치를 사용하는 것이 좋습니다.

```
proton-setup.sh experimental_8.0
```

- 기타 사용자 지정 Proton 버전

다른 Proton 버전의 경우 [Proton 릴리스](#)에 나열된 정확한 브랜치 또는 태그 이름을 사용합니다.

```
proton-setup.sh branch-or-tag-name
```

설치에 성공하면 터미널의 출력은 다음과 비슷해야 합니다.

```
...
Done building Proton!
Finished preparing proton. Proton binary location: /home/test/protonBuild/dist/
proton
STEAM_COMPAT_DATA_PATH: /home/test/protonBuild/dist/compatdata
STEAM_COMPAT_CLIENT_INSTALL_PATH: anything
```

다음 단계에서 Proton을 실행해야 하므로 출력에서 다음 변수를 기록해 둡니다.

- 양성자 바이너리 위치
- STEAM_COMPAT_DATA_PATH
- STEAM_COMPAT_CLIENT_INSTALL_PATH

Proton에서 애플리케이션 실행

다음 단계에서는 애플리케이션 실행 파일이 있다고 가정합니다 `path/myapplication/bin/application.exe`. 이를 애플리케이션의 경로 및 파일 이름으로 바꿉니다.

- 터미널에서 애플리케이션 실행 파일이 있는 폴더로 이동합니다.

```
cd path/myapplication/bin/application.exe
```

- Proton에서 애플리케이션을 실행합니다. 이전 단계에서 얻은 Proton 바이너리 위치와 환경 변수를 사용합니다.

```
STEAM_COMPAT_DATA_PATH=/home/test/protonBuild/dist/compatdata
STEAM_COMPAT_CLIENT_INSTALL_PATH=anything /home/test/protonBuild/dist/proton run
application.exe
```

이제 애플리케이션이 시작을 시도해야 합니다. 애플리케이션이 로컬에서 시작되지만 Amazon GameLift Streams에서는 시작되지 않는 경우 Amazon GameLift Streams APIs. API 호출 파라미터가 올바른지 확인합니다. 그렇지 않으면 디버깅을 위한 다음 단계를 계속합니다.

로그 파일을 통해 애플리케이션 디버깅

애플리케이션에 로컬 Proton 환경에서 실행되는 문제가 있는 경우 출력 로그를 확인합니다. 로그에는 애플리케이션 및 런타임 환경의 출력이 포함됩니다. 애플리케이션이 애플리케이션 측에서 문제를 발견하지 못하는 위치를 추적합니다.

로그 출력을와 같은 텍스트 파일로 덤프하려면 다음 명령을 `proton.log` 사용합니다.

```
STEAM_COMPAT_DATA_PATH=/home/test/protonBuild/dist/compatdata
STEAM_COMPAT_CLIENT_INSTALL_PATH=anything /home/test/protonBuild/dist/proton run
application.exe &>proton.log
```

또한 Proton은 Wine 플러그인, 구현되지 않은 함수, 누락된 dll 등으로 인한 문제인지 여부를 나타냅니다. 자세한 내용은 [Wine HQ의 Debugging Wine](#) 가이드를 참조하세요. 애플리케이션 측에서 수정할 수 없는 Proton 또는 Wine 오류가 로그에서 발견되면 AWS 계정 관리자에게 문의하거나 [AWS re:Post](#)에 질문을 게시하여 추가 디버깅에 대한 도움을 받으세요.

Unreal Engine 성능 프로파일링

이 섹션에서는 Unreal Engine 게임 또는 애플리케이션 성능을 분석하는 방법을 알아봅니다. 이렇게 하면 최적화할 영역의를 식별하여 Amazon GameLift Streams에서 스트리밍을 더 원활하게 만들 수 있습니다.

Unreal Engine의 콘솔과 내장 통계 명령을 사용하여 게임의 성능을 자세히 살펴볼 수 있습니다. 배송할 수 없는 빌드 또는 편집기에서 콘솔에 액세스할 수 있습니다. 배송할 수 없는 빌드는 디버그 또는 개발 구성을 사용하여 빌드된 프로젝트를 나타냅니다.

콘솔에 액세스하려면

배송할 수 없는 빌드와 [편집기에서 재생](#) 모드에서 물결표(~) 키를 눌러 콘솔을 엽니다. 물결표 키를 두 번 눌러 콘솔을 확장합니다.

다음은 콘솔 사용에 대한 몇 가지 팁입니다.

- 키워드를 입력하여 해당 키워드가 포함된 가능한 모든 명령을 나열합니다. 화살표 키를 사용하여 목록을 스크롤합니다.
- 화살표 키 또는 페이지 위로 및 페이지 아래로 키를 사용하여 기록을 스크롤합니다.
- 로그는 프로젝트 Saved/Logs 디렉터리의 .txt 파일에 저장됩니다.

게임 성능을 프로파일링하려면

1. 먼저 `stat fps` 및 `stat unit` 명령을 실행합니다. 이렇게 하면 게임이 성능에 어려움을 겪는 부분에 대한 개요를 볼 수 있습니다.
 - `stat fps`: 현재 초당 프레임을 표시합니다.
 - `stat unit`: 프레임을 여러 하위 섹션으로 나눕니다.
 - 프레임: 프레임 시뮬레이션이 시작되는 시점부터 프레임 프레젠테이션이 화면에 표시되는 시점까지의 총 벽시계 시간입니다.
 - 게임: 프레임당 게임 시뮬레이션 스레드에 걸린 총 CPU 시간입니다.
 - 그리기: 렌더링 스레드가 장면을 GPU의 명령으로 변환하고 GPU에 제출하는 데 걸리는 총 CPU 시간입니다.
 - GPU: GPU가 모든 명령을 처리하는 데 걸리는 총 시간입니다.
 - 드로우: 프레임에 대해 제출된 총 드로우 수입니다.
 - Prims: 그려진 총 삼각형 수입니다.
2. 게임을 플레이하고 게임, 그리기 또는 GPU에서 FPS 감소와 시간 증가로 나타나는 성능이 낮은 영역을 식별합니다.
3. 를 실행 `stat game` 하여 다양한 게임 플레이 그룹에 소요되는 시간을 확인합니다.
4. AI, 애니메이션, 물리, 게임 플레이, 스크립팅 등과 같은 특정 게임 플레이 요소에 대한 통계를 구체화합니다. 다음은 몇 가지 예입니다.
 - `stat ai`: AI 동작을 계산하는 데 걸리는 시간입니다.
 - `stat anim`: 스키닝된 메시를 계산하는 시간입니다.
 - `stat physics`: 물리 시뮬레이션을 계산하는 데 걸리는 시간입니다.
5. 를 실행 `stat drawcount` 하여 가장 많은 드로우를 생성하는 렌더링 영역을 확인합니다. 목록에는 드로우를 내보내는 렌더 패스와 각 프레임에서 내보내는 드로우 수가 표시됩니다. 다음 단계에서 GPU 통계를 분석하여 자세한 정보를 얻을 수 있습니다.
6. 를 실행 `stat gpu` 하여 GPU 시간을 가장 많이 소비하는 렌더링 유형을 확인합니다.
7. 렌더링 유형을 조명, 그림자, 내강(조명), 머리카락, 후처리 등과 같은 광범위한 그룹으로 구체화합니다. 다음은 몇 가지 일반적인 예입니다.
 - `stat lightrendering`: 조명과 그림자를 렌더링하는 GPU 시간입니다.
 - `stat shadowrendering`: 다양한 새도우를 업데이트하는 GPU 시간입니다.
 - `stat scenerendering`: 장면을 렌더링하는 GPU 시간입니다.

이 섹션에서는 사용 가능한 명령의 하위 집합만 다룹니다. 게임의 기능에 따라 자산 스트리밍, 가상 텍스처링, CPU 작업 워크로드 배포, 스테딩, 사운드, 파티클 등과 같은 영역의 통계를 살펴봅니다. 자세한 내용은 [통계 명령을](#) 참조하세요.

리전, 할당량 및 제한 사항

Amazon GameLift Streams는 IPv4 및 IPv6 연결을 모두 지원하는 듀얼 스택 서비스 엔드포인트를 AWS 리전제공하는 여러에서 사용할 수 있습니다. 이 서비스는 미국 동부(오하이오), 미국 서부(오레곤), 아시아 태평양(도쿄), 유럽(프랑크푸르트)을 포함한 기본 위치에서 운영되며, 최적화된 지연 시간과 스트림 품질을 위해 원격 위치라고 하는 추가 AWS 리전 및 위치를 관리할 수 있습니다.

서비스 인프라는 세 가지 주요 제약 범주로 관리됩니다.

- Service Quotas
- API 속도 제한
- 서비스 제한 수정

여기에는 애플리케이션 크기, 리전당 애플리케이션 수, 파일 관리 용량, 다양한 스트림 클래스 및 리전에서 GPU 할당에 대한 제한이 포함됩니다. 이 서비스는 초당 1~20개의 요청으로 다양한 작업에 대해 특정 API 속도 제한을 구현하여 안정적인 서비스 성능을 보장합니다. 또한 모든 고객에게 균일하게 적용되는 스트림 그룹 구성, GPU 배포 및 애플리케이션 연결과 관련된 고정 서비스 제한 사항이 있습니다.

AWS 리전 Amazon GameLift Streams에서 지원하는 및 스트리밍 위치

AWS 리전은 지리적 영역의 AWS 리소스 모음입니다. 각 AWS 리전은 격리되어 있으며 다른 리전에 독립적입니다. 에 대한 일반적인 내용은 [관리를 AWS 리전](#) AWS 리전참조하세요AWS 일반 참조.

다음 표에는 Amazon GameLift Streams 서비스를 사용할 수 있는 AWS 리전 있는과 각 리전의 엔드포인트가 나와 있습니다. Amazon GameLift Streams 콘솔에서 작업하든, (AWS CLI)를 AWS Command Line Interface 사용하든, 프로그래밍 방식으로 호출하든 관계없이 지정된 리전에서 모든 Amazon GameLift Streams 애플리케이션 및 스트림 그룹 리소스를 생성합니다. 이러한 리소스를 생성하는 리전을 기본 위치라고 합니다. 기본 위치의 엔드포인트를 사용하여 프로그래밍 방식으로 Amazon GameLift Streams 서비스에 연결합니다.

서비스 엔드포인트

Amazon GameLift Streams는 듀얼 스택 서비스 엔드포인트를 지원하므로 클라이언트와 리소스가 IPv6 또는 IPv4를 사용하여 서비스와 상호 작용할 수 있습니다.

리전 이름	리전	엔드포인트	프로토콜
미국 동부 (오하이오)	us-east-2	gameliftstreams.us-east-2.api.aws	HTTPS
미국 서부 (오리곤)	us-west-2	gameliftstreams.us-west-2.api.aws	HTTPS
아시아 태평양(도쿄)	ap-northeast-1	gameliftstreams.ap-northeast-1.api.aws	HTTPS
유럽(프랑크푸르트)	eu-central-1	gameliftstreams.eu-central-1.api.aws	HTTPS

스트리밍 위치

Amazon GameLift Streams는 서비스 엔드포인트의 다음 모든 위치에서 스트리밍을 지원합니다. 지연 시간과 스트림 품질을 최적화하려면 사용자와 지리적으로 가까운 스트리밍 위치를 선택하는 것이 좋습니다.

리전 이름	AWS 리전
미국 동부(버지니아 북부)	us-east-1
미국 동부(오하이오)	us-east-2
미국 서부(오리건)	us-west-2
Asia Pacific (Mumbai)	ap-south-1
아시아 태평양(서울)	ap-northeast-2
아시아 태평양(시드니)	ap-southeast-2
아시아 태평양(도쿄)	ap-northeast-1

리전 이름	AWS 리전
유럽(프랑크푸르트)	eu-central-1
유럽(아일랜드)	eu-west-1
유럽(런던)	eu-west-2
유럽(스톡홀름)	eu-north-1
남아메리카(상파울루)	sa-east-1

Amazon GameLift Streams에서 스트림 클래스별로 지원되는 위치

다음 표에는 지원되는 모든 AWS 리전 및 스트리밍 위치에서 각 스트림 클래스 패밀리의 가용성이 나와 있습니다.

리전 이름	리전	gen6*	gen5*	gen4*
미국 동부(버지니아 북부)	us-east-1	✓ 예	✓ 예	✓ 예
미국 동부(오하이오)	us-east-2	✓ 예	✓ 예	✓ 예
미국 서부(오리건)	us-west-2	✓ 예	✓ 예	✓ 예
아시아 태평양(뭄바이)	ap-south-1	✓ 예	✓ 예	✓ 예
아시아 태평양(서울)	ap-northeast-2	✓ 예	✓ 예	✓ 예
아시아 태평양(시드니)	ap-southeast-2	✓ 예	✓ 예	✓ 예
아시아 태평양(도쿄)	ap-northeast-1	✓ 예	✓ 예	✓ 예
유럽(프랑크푸르트)	eu-central-1	✓ 예	✓ 예	✓ 예
유럽(아일랜드)	eu-west-1	x 아 니요	✓ 예	✓ 예
Europe (London)	eu-west-2	✓ 예	✓ 예	✓ 예

리전 이름	리전	gen6*	gen5*	gen4*
유럽(스톡홀름)	eu-north-1	✓ 예	✓ 예	✓ 예
남아메리카(상파울루)	sa-east-1	✓ 예	✓ 예	✓ 예

Amazon GameLift Streams 서비스 할당량

한도라고도 하는 서비스 할당량은 AWS 계정에 대한 서비스 리소스 또는 작업의 최대 수입니다.

Amazon GameLift Streams의 많은 서비스 할당량은 계정에서 스트리밍을 위해 구성할 수 있는 총 GPU(컴퓨팅 리소스) 수를 제한합니다. 보다 구체적으로, 이러한 GPU 서비스 할당량은 계정의 모든 스트림 그룹에 걸쳐 위치당 요청할 수 있는 특정 스트림 클래스 패밀리의 최대 GPU 수를 지정합니다. 예를 들어 계정에서의 gen5n GPU 한도가 5개인 경우 모든 스트림 그룹에 us-west-2 대의 총 스트림 용량을 제공하는 데 필요한 gen5n GPU us-west-2합계는 5보다 작거나 같아야 합니다. 여기에는 상시 가동 용량과 온디맨드 용량 모두에 대한 GPU 포함됩니다.

할당량이 스트림 용량과 상호 작용하는 방법에 대한 자세한 내용은 [섹션을 참조하세요](#) [용량 및 서비스 할당량](#). 또한 Amazon GameLift Streams에서 알아야 할 추가 제한 사항이 [기타 제한 사항](#) 있는지 [API 속도 제한](#) 및를 확인해야 합니다.

GameLift 스트림을 AWS 서비스로 선택하여 Service Quotas 콘솔에서 기본 또는 적용된 계정 수준 할당량과 사용률을 확인합니다.

서비스 할당량에 대한 일반적인 내용은 [AWS의 서비스 할당량을 참조하세요](#) AWS 일반 참조.

Service Quotas

다음 표에서 GPU 할당량은 기본적으로 모두 0입니다. 그러나 계정에 적용된 할당량은 다를 수 있습니다. 확인하려면 로그인 AWS Management Console 하고 [Amazon GameLift Streams](#)에 대한 Service Quotas 콘솔을 엽니다. 여기서 적용된 계정 수준 할당량 값 열의 현재 할당량과 사용률 열의 이러한 할당량 사용률을 검토하고 이러한 값을 늘리기 위한 요청을 제출할 수 있습니다.

이름	기본값	조정 가능	설명
애플리케이션 크기(GiB)	지원되는 각 리전: 100	예	이 계정에서 애플리케이션의 최대 총 크기(GiB)입니다. 참고로 기비바이트(GiB)는 1024*1024*1024 바이트입니다.
애플리케이션	지원되는 각 리전: 20개	예	AWS 리전별로 이 계정에서 생성할 수 있는 최대 애플리케이션 수입니다.
애플리케이션당 파일	지원되는 각 리전: 30,000	예	이 계정에서 애플리케이션별로 사용할 수 있는 최대 파일 수입니다.
Gen4n GPU, ap-northeast-1	지원되는 각 리전: 0	예	이 계정의 모든 스트림 그룹에 걸쳐 ap-northeast-1 위치에서 스트리밍하도록 구성할 수 있는 최대 Gen4n GPUs. "Gen4n_high"와 같은 다중 테넌트 스트림 클래스는 GPU당 둘 이상의 세션 스트리밍을 지원합니다.
Gen4n GPUs, ap-northeast-2	지원되는 각 리전: 0	예	이 계정의 모든 스트림 그룹에 걸쳐 ap-northeast-2 위치에서 스트리밍하도록 구성할 수 있는 최대 Gen4n GPUs. "Gen4n_high"와 같은 다중 테넌트 스트림 클래스는 GPU당

이름	기본값	조정 가능	설명
			둘 이상의 세션 스트리밍을 지원합니다.
Gen4n GPUs, ap-south-1	지원되는 각 리전: 0	예	이 계정의 모든 스트림 그룹에 걸쳐 ap-south-1 위치에서 스트리밍하도록 구성할 수 있는 최대 Gen4n GPUs. "Gen4n_high"와 같은 다중 테넌트 스트림 클래스는 GPU당 둘 이상의 세션 스트리밍을 지원합니다.
Gen4n GPUs, ap-southeast-2	지원되는 각 리전: 0	예	이 계정의 모든 스트림 그룹에 걸쳐 ap-southeast-2 위치에서 스트리밍하도록 구성할 수 있는 최대 Gen4n GPUs. "Gen4n_high"와 같은 다중 테넌트 스트림 클래스는 GPU당 둘 이상의 세션 스트리밍을 지원합니다.
Gen4n GPU, eu-central-1	지원되는 각 리전: 0	예	이 계정의 모든 스트림 그룹에 걸쳐 eu-central-1 위치에서 스트리밍하도록 구성할 수 있는 최대 Gen4n GPUs. "Gen4n_high"와 같은 다중 테넌트 스트림 클래스는 GPU당 둘 이상의 세션 스트리밍을 지원합니다.

이름	기본값	조정 가능	설명
Gen4n GPUs, eu-north-1	지원되는 각 리전: 0	예	이 계정의 모든 스트림 그룹에 걸쳐 eu-north-1 위치에서 스트리밍하도록 구성할 수 있는 최대 Gen4n GPUs. "Gen4n_high"와 같은 다중 테넌트 스트림 클래스는 GPU당 둘 이상의 세션 스트리밍을 지원합니다.
Gen4n GPU, eu-west-1	지원되는 각 리전: 0	예	이 계정의 모든 스트림 그룹에 걸쳐 eu-west-1 위치에서 스트리밍하도록 구성할 수 있는 최대 Gen4n GPUs. "Gen4n_high"와 같은 다중 테넌트 스트림 클래스는 GPU당 둘 이상의 세션 스트리밍을 지원합니다.
Gen4n GPUs, eu-west-2	지원되는 각 리전: 0	예	이 계정의 모든 스트림 그룹에 걸쳐 eu-west-2 위치에서 스트리밍하도록 구성할 수 있는 최대 Gen4n GPUs. "Gen4n_high"와 같은 다중 테넌트 스트림 클래스는 GPU당 둘 이상의 세션 스트리밍을 지원합니다.

이름	기본값	조정 가능	설명
Gen4n GPUs, sa-east-1	지원되는 각 리전: 0	예	이 계정의 모든 스트림 그룹에 걸쳐 sa-east-1 위치에서 스트리밍하도록 구성할 수 있는 최대 Gen4n GPUs. "Gen4n_high"와 같은 다중 테넌트 스트림 클래스는 GPU당 둘 이상의 세션 스트리밍을 지원합니다.
Gen4n GPU, us-east-1	지원되는 각 리전: 0	예	이 계정의 모든 스트림 그룹에서 us-east-1 위치에서 스트리밍하도록 구성할 수 있는 최대 Gen4n GPUs. "Gen4n_high"와 같은 다중 테넌트 스트림 클래스는 GPU당 둘 이상의 세션 스트리밍을 지원합니다.
Gen4n GPU, us-east-2	지원되는 각 리전: 0	예	이 계정의 모든 스트림 그룹에서 us-east-2 위치에서 스트리밍하도록 구성할 수 있는 최대 Gen4n GPUs. "Gen4n_high"와 같은 다중 테넌트 스트림 클래스는 GPU당 둘 이상의 세션 스트리밍을 지원합니다.

이름	기본값	조정 가능	설명
Gen4n GPU, us-west-2	지원되는 각 리전: 0	예	이 계정의 모든 스트림 그룹에서 us-west-2 위치에서 스트리밍하도록 구성할 수 있는 최대 Gen4n GPUs. "Gen4n_high"와 같은 다중 테넌트 스트림 클래스는 GPU당 둘 이상의 세션 스트리밍을 지원합니다.
Gen5n GPU, ap-northeast-1	지원되는 각 리전: 0	예	이 계정의 모든 스트림 그룹에 걸쳐 ap-northeast-1 위치에서 스트리밍하도록 구성할 수 있는 최대 Gen5n GPUs. "Gen5n_high"와 같은 다중 테넌트 스트림 클래스는 GPU당 둘 이상의 세션 스트리밍을 지원합니다.
Gen5n GPUs, ap-northeast-2	지원되는 각 리전: 0	예	이 계정의 모든 스트림 그룹에 걸쳐 ap-northeast-2 위치에서 스트리밍하도록 구성할 수 있는 최대 Gen5n GPUs. "Gen5n_high"와 같은 다중 테넌트 스트림 클래스는 GPU당 둘 이상의 세션 스트리밍을 지원합니다.

이름	기본값	조정 가능	설명
Gen5n GPUs, ap-south-1	지원되는 각 리전: 0	예	이 계정의 모든 스트림 그룹에 걸쳐 ap-south-1 위치에서 스트리밍하도록 구성할 수 있는 최대 Gen5n GPUs. "Gen5n_high"와 같은 다중 테넌트 스트림 클래스는 GPU당 둘 이상의 세션 스트리밍을 지원합니다.
Gen5n GPUs, ap-southeast-2	지원되는 각 리전: 0	예	이 계정의 모든 스트림 그룹에 걸쳐 ap-southeast-2 위치에서 스트리밍하도록 구성할 수 있는 최대 Gen5n GPUs. "Gen5n_high"와 같은 다중 테넌트 스트림 클래스는 GPU당 둘 이상의 세션 스트리밍을 지원합니다.
Gen5n GPU, eu-central-1	지원되는 각 리전: 0	예	이 계정의 모든 스트림 그룹에 걸쳐 eu-central-1 위치에서 스트리밍하도록 구성할 수 있는 최대 Gen5n GPUs. "Gen5n_high"와 같은 다중 테넌트 스트림 클래스는 GPU당 둘 이상의 세션 스트리밍을 지원합니다.

이름	기본값	조정 가능	설명
Gen5n GPUs, eu-north-1	지원되는 각 리전: 0	예	이 계정의 모든 스트림 그룹에 걸쳐 eu-north-1 위치에서 스트리밍하도록 구성할 수 있는 최대 Gen5n GPUs. "Gen5n_high"와 같은 다중 테넌트 스트림 클래스는 GPU당 둘 이상의 세션 스트리밍을 지원합니다.
Gen5n GPU, eu-west-1	지원되는 각 리전: 0	예	이 계정의 모든 스트림 그룹에 걸쳐 eu-west-1 위치에서 스트리밍하도록 구성할 수 있는 최대 Gen5n GPUs. "Gen5n_high"와 같은 다중 테넌트 스트림 클래스는 GPU당 둘 이상의 세션 스트리밍을 지원합니다.
Gen5n GPUs, eu-west-2	지원되는 각 리전: 0	예	이 계정의 모든 스트림 그룹에 걸쳐 eu-west-2 위치에서 스트리밍하도록 구성할 수 있는 최대 Gen5n GPUs. "Gen5n_high"와 같은 다중 테넌트 스트림 클래스는 GPU당 둘 이상의 세션 스트리밍을 지원합니다.

이름	기본값	조정 가능	설명
Gen5n GPUs, sa-east-1	지원되는 각 리전: 0	예	이 계정의 모든 스트림 그룹에 걸쳐 sa-east-1 위치에서 스트리밍하도록 구성할 수 있는 최대 Gen5n GPUs. "Gen5n_high"와 같은 다중 테넌트 스트림 클래스는 GPU당 둘 이상의 세션 스트리밍을 지원합니다.
Gen5n GPU, us-east-1	지원되는 각 리전: 0	예	이 계정의 모든 스트림 그룹에서 us-east-1 위치에서 스트리밍하도록 구성할 수 있는 최대 Gen5n GPUs. "Gen5n_high"와 같은 다중 테넌트 스트림 클래스는 GPU당 둘 이상의 세션 스트리밍을 지원합니다.
Gen5n GPU, us-east-2	지원되는 각 리전: 0	예	이 계정의 모든 스트림 그룹에서 us-east-2 위치에서 스트리밍하도록 구성할 수 있는 최대 Gen5n GPUs. "Gen5n_high"와 같은 다중 테넌트 스트림 클래스는 GPU당 둘 이상의 세션 스트리밍을 지원합니다.

이름	기본값	조정 가능	설명
Gen5n GPU, us-west-2	지원되는 각 리전: 0	예	이 계정의 모든 스트림 그룹에서 us-west-2 위치에서 스트리밍하도록 구성할 수 있는 최대 Gen5n GPUs. "Gen5n_high"와 같은 다중 테넌트 스트림 클래스는 GPU당 둘 이상의 세션 스트리밍을 지원합니다.
Gen6n GPUs, ap-northeast-1	지원되는 각 리전: 0	예	이 계정의 모든 스트림 그룹에 걸쳐 ap-northeast-1 위치에서 스트리밍하도록 구성할 수 있는 최대 Gen6n GPUs. "Gen6n_high"와 같은 다중 테넌트 스트림 클래스는 GPU당 둘 이상의 세션 스트리밍을 지원합니다.
Gen6n GPUs, ap-northeast-2	지원되는 각 리전: 0	예	이 계정의 모든 스트림 그룹에 걸쳐 ap-northeast-2 위치에서 스트리밍하도록 구성할 수 있는 최대 Gen6n GPUs. "Gen6n_high"와 같은 다중 테넌트 스트림 클래스는 GPU당 둘 이상의 세션 스트리밍을 지원합니다.

이름	기본값	조정 가능	설명
Gen6n GPUs, ap-south-1	지원되는 각 리전: 0	예	이 계정의 모든 스트림 그룹에 걸쳐 ap-south-1 위치에서 스트리밍하도록 구성할 수 있는 최대 Gen6n GPUs. "Gen6n_high"와 같은 다중 테넌트 스트림 클래스는 GPU당 둘 이상의 세션 스트리밍을 지원합니다.
Gen6n GPUs, ap-southeast-2	지원되는 각 리전: 0	예	이 계정의 모든 스트림 그룹에 걸쳐 ap-southeast-2 위치에서 스트리밍하도록 구성할 수 있는 최대 Gen6n GPUs. "Gen6n_high"와 같은 다중 테넌트 스트림 클래스는 GPU당 둘 이상의 세션 스트리밍을 지원합니다.
Gen6n GPUs, eu-central-1	지원되는 각 리전: 0	예	이 계정의 모든 스트림 그룹에 걸쳐 eu-central-1 위치에서 스트리밍하도록 구성할 수 있는 최대 Gen6n GPUs. "Gen6n_high"와 같은 다중 테넌트 스트림 클래스는 GPU당 둘 이상의 세션 스트리밍을 지원합니다.

이름	기본값	조정 가능	설명
Gen6n GPUs, eu-north-1	지원되는 각 리전: 0	예	이 계정의 모든 스트림 그룹에 걸쳐 eu-north-1 위치에서 스트리밍하도록 구성할 수 있는 최대 Gen6n GPUs. "Gen6n_high"와 같은 다중 테넌트 스트림 클래스는 GPU당 둘 이상의 세션 스트리밍을 지원합니다.
Gen6n GPUs, eu-west-2	지원되는 각 리전: 0	예	이 계정의 모든 스트림 그룹에 걸쳐 eu-west-2 위치에서 스트리밍하도록 구성할 수 있는 최대 Gen6n GPUs. "Gen6n_high"와 같은 다중 테넌트 스트림 클래스는 GPU당 둘 이상의 세션 스트리밍을 지원합니다.
Gen6n GPUs, sa-east-1	지원되는 각 리전: 0	예	이 계정의 모든 스트림 그룹에 걸쳐 sa-east-1 위치에서 스트리밍하도록 구성할 수 있는 최대 Gen6n GPUs. "Gen6n_high"와 같은 다중 테넌트 스트림 클래스는 GPU당 둘 이상의 세션 스트리밍을 지원합니다.

이름	기본값	조정 가능	설명
Gen6n GPUs, us-east-1	지원되는 각 리전: 0	예	이 계정의 모든 스트림 그룹에 걸쳐 us-east-1 위치에서 스트리밍하도록 구성할 수 있는 최대 Gen6n GPUs. "Gen6n_high"와 같은 다중 테넌트 스트림 클래스는 GPU당 둘 이상의 세션 스트리밍을 지원합니다.
Gen6n GPUs, us-east-2	지원되는 각 리전: 0	예	이 계정의 모든 스트림 그룹에서 us-east-2 위치에서 스트리밍하도록 구성할 수 있는 최대 Gen6n GPUs. "Gen6n_high"와 같은 다중 테넌트 스트림 클래스는 GPU당 둘 이상의 세션 스트리밍을 지원합니다.
Gen6n GPUs, us-west-2	지원되는 각 리전: 0	예	이 계정의 모든 스트림 그룹에서 us-west-2 위치에서 스트리밍하도록 구성할 수 있는 최대 Gen6n GPUs. "Gen6n_high"와 같은 다중 테넌트 스트림 클래스는 GPU당 둘 이상의 세션 스트리밍을 지원합니다.

이름	기본값	조정 가능	설명
스트림 그룹	지원되는 각 리전: 5개	예	AWS 리전별로이 계정에서 생성할 수 있는 최대 스트림 그룹 수입니다. 스트림 그룹은 애플리케이션을 최종 사용자에게 스트리밍하는 컴퓨팅 리소스 모음입니다.

Amazon GameLift Streams API 속도 제한

이러한 제한은에서 AWS 계정 Amazon GameLift Streams 서비스로의 초당 최대 요청 속도를 반영합니다 AWS 리전.

API 작업	초당 요청
AddStreamGroupLocations	5
AssociateApplications	5
CreateApplication	5
CreateStreamGroup	1
CreateStreamSessionConnection	20
DeleteApplication	5
DeleteStreamGroup	5
DisassociateApplications	5
ExportStreamSessionFiles	20

API 작업	초당 요청
GetApplication	10
GetStreamGroup	10
GetStreamSession	20
ListApplications	10
ListStreamGroups	10
ListStreamSessions	20
ListStreamSessionsByAccount	20
ListTagsForResource	10
RemoveStreamGroupLocations	5
StartStreamSession	20
TagResource	10
TerminateStreamSession	20
UntagResource	10
UpdateApplication	5
UpdateStreamGroup	5

기타 Amazon GameLift Streams 제한 사항

이 페이지에는 스트리밍 솔루션을 생성할 때 알아야 할 다른 제한 사항이 나열되어 있습니다. 이러한 제한은 모든 고객에 대해 서비스 내에서 고정됩니다.

이름	제한 사항	설명
스트림 그룹의 애플리케이션	100	스트림 그룹에 연결할 수 있는 Amazon GameLift Streams 애플리케이션의 최대 수입니다.
스트림 그룹의 GPUs	2500	모든 리전 및 원격 위치에서 스트림 그룹의 최대 GPUs 수입니다.
단일 파일 크기(GiB)	80GiB	애플리케이션에서 단일 파일의 최대 크기(GiB)입니다. 참고로 기비바이트(GiB)는 1024*1024*1024바이트입니다.
애플리케이션당 스트림 그룹 연결	100	Amazon GameLift Streams 애플리케이션을 연결할 수 있는 최대 스트림 그룹 수입니다.
VPC 전송 구성	5	리전 AWS 계정 당 당 최대 VPC 전송 구성 수입니다.

Amazon GameLift Streams의 사용량 및 청구서 관리

이 주제에서는 스트리밍 비용을 최적화하기 위해 Amazon GameLift Streams 사용량, 비용 및 결제를 모니터링하고 관리하는 방법을 다룹니다.

다음 정보는 Amazon GameLift 스트림 [요금 페이지](#)를 참조하세요.

- 비용 분석: Amazon GameLift Streams를 사용할 때 AWS 청구되는 요금을 이해합니다.
- Amazon GameLift Streams 요금: Amazon GameLift Streams 비용을 확인하고 다양한 옵션을 비교합니다.
- 스트림 용량 예약: 미리 계획을 세우고 고객의 요구에 맞는 충분한 스트림 용량을 확보하세요.

Amazon GameLift Streams 청구서 및 사용량 검토

AWS 콘솔 또는의 AWS 결제 및 비용 관리 도구를 사용하여 Amazon GameLift Streams 청구서 및 사용량을 검토할 수 있습니다 AWS CLI.

AWS 콘솔을 통해 청구서를 보려면 AWS Billing 사용 설명서의 [청구서 보기를 참조하세요](#).

를 통해 청구서를 보려면 Billing and Cost Management API를 [GetCostAndUsage](#) 사용하여 AWS CLI호출합니다. 예를 들어 다음 명령을 사용하여 Amazon GameLift Streams에 대한 월별 청구서를 검색하고 날짜를 해당 날짜로 바꿉니다.

Example: GetCostAndUsage API를 사용하여 청구서 보기

```
aws ce get-cost-and-usage /
  --time-period Start=2023-01-01,End=2023-01-31 /
  --granularity MONTHLY /
  --metrics BlendedCost /
  --filter Amazon GameLift Streams-bill-filter.json
```

여기서와 같은 필터는 다음과 같이 Amazon GameLift Streams 서비스를 Amazon GameLift Streams-bill-filter.json지정합니다.

```
{
  "Dimensions": {
    "Key": "SERVICE",
    "Values": ["Amazon Amazon GameLift Streams"]
  }
}
```

Amazon GameLift Streams 비용 관리 모범 사례

예기치 않은 비용을 방지하려면 다음 도구와 기법을 사용하여 Amazon GameLift Streams 비용을 관리하는 것이 좋습니다.

결제 알림을 생성하여 사용량 모니터링

AWS Budgets를 사용하여 결제 알림을 설정하면 비용과 사용량을 추적하고 알림에 신속하게 응답하여 예상치 못한 비용을 방지할 수 있습니다. 예산을 유지하는 데 도움이 되는 작업을 트리거하도록 결제 알림을 구성할 수도 있습니다. 기본적으로 예산에는 모든 AWS 서비스가 포함됩니다. Amazon GameLift Streams에 대해서만 예산을 지정하려면 [예산 필터](#)를 추가합니다.

자세한 정보는 다음의 주제를 참조하세요.

- [예산 생성](#)
- [AWS Budgets 모범 사례](#)

스트림 그룹을 0 용량으로 확장

할당된 스트림 용량은 현재 스트림 세션을 호스팅하지 않는 경우에도 비용이 계속 발생합니다. 불필요한 비용을 방지하기 위해 사용하지 않을 때 스트림 그룹을 0의 용량으로 확장합니다. 이렇게 하면 스트림 그룹이 리소스를 할당할 수 없습니다. 상시 가동 및 온디맨드 스트림 용량을 0으로 설정하면 연결된 모든 스트림이 종료됩니다. 준비가 되면 용량을 다시 조정하여 스트림 그룹을 재사용할 수 있습니다.

지침은 단원을 참조하십시오 [용량 편집](#).

Warning

스트림 그룹을 다시 사용할 계획이 없다면 스트림 그룹을 삭제하지 마세요. 스트림 그룹을 삭제하면 원래 스트림 그룹을 복원할 수 없으며 새 스트림 그룹을 생성해야 합니다.

원본 애플리케이션 파일 삭제

스토리지 비용을 최적화하기 위해 Amazon S3 버킷에 업로드한 원래 애플리케이션 파일을 삭제할 수 있습니다. 애플리케이션이 준비 상태인 경우 파일을 삭제해도 안전합니다. 이때 Amazon GameLift Streams에는 애플리케이션 파일의 스냅샷이 있으며 더 이상 원본 파일에 액세스하지 않습니다.

기계 번역으로 제공되는 번역입니다. 제공된 번역과 원본 영어의 내용이 상충하는 경우에는 영어 버전이 우선합니다.