aws

개발자 가이드

AMB 액세스 다각형

AMB 액세스 다각형: 개발자 가이드

Table of Contents

Amazon Managed Blockchain(AMB) Access Polygon은 평가판 릴리스이며 변경될 수 있습니다.

기계 번역으로 제공되는 번역입니다. 제공된 번역과 원본 영어의 내용이 상충하는 경우에는 영어 버전이 우선합니다.

Amazon Managed Blockchain(AMB) Access Polygon이란 무엇인가요?

Amazon Managed Blockchain(AMB) Access Polygon은 Polygon 블록체인에서 복원력이 뛰어난 Web3 애플리케이션을 구축하는 데 도움이 되는 완전 관리형 서비스입니다. AMB Access Polygon은 Polygon 블록체인에 대한 즉각적인 서버리스 액세스를 제공합니다.

Polygon은 Ethereum Virtual Machine(EVM)을 기반으로 사용하는 조정 솔루션입니다. Polygon 블록체인은 높은 트랜잭션 처리량과 낮은 트랜잭션 요금으로 알려져 있습니다. Polygon 블록체인은 proof-of-stake 합의 메커니즘을 사용합니다. 다각형은 특히 NFTs, Web3 게임 및 토큰화 사용 사례와 관련된 분산형 애플리케이션(dApps)을 구축하는 데 일반적으로 사용됩니다.

이 안내서에서는 Amazon Managed Blockchain(AMB) Access Polygon을 사용하여 Polygon 블록체인 리소스를 생성하고 관리하는 방법을 설명합니다.

처음 AMB Access Polygon 사용자를 위한 리소스

AMB Access Polygon을 처음 사용하는 경우 먼저 다음 섹션을 읽어보는 것이 좋습니다.

- 주요 개념: Amazon Managed Blockchain(AMB) Access Polygon
- Amazon Managed Blockchain(AMB) Access Polygon 시작하기
- AMB Access Polygon에서 지원되는 Managed Blockchain API 및 JSON-RPCs

주요 개념: Amazon Managed Blockchain(AMB) Access Polygon

> **Note**
>
> 이 안내서에서는 다각형에 필수적인 개념을 잘 알고 있다고 가정합니다. 이러한 개념에는 스테이킹, dApps, 트랜잭션, 지갑, 스마트 계약, 다각형(POL, 이전 MATIC) 등이 포함됩니다. Amazon Managed Blockchain(AMB) Access Polygon을 사용하기 전에 다각형 개발 설명서와 다각형 위키를 검토하는 것이 좋습니다.

Amazon Managed Blockchain(AMB) Access Polygon은 노드를 포함한 모든 Polygon 인프라를 프로비저닝하고 관리할 필요 없이 Polygon Mainnet 및 Polygon Mainnet 네트워크에 대한 서버리스 액세스를 제공합니다. 네트워크의 다각형 노드는 총체적으로 다각형 블록체인 상태를 저장하고, 트랜잭션을 확인하고, 합의에 참여하여 블록체인 상태를 변경합니다. 이 관리형 서비스를 사용하면 Polygon 네트워크에 온디맨드로 빠르게 액세스하여 전체 소유권 비용을 줄일 수 있습니다.

AMB Access Polygon을 사용하면 JSON 원격 프로시저(JSON-RPC) 호출에 액세스할 수 있습니다. 폴리곤 JSON-RPCs 호출하여 관리형 블록체인에서 관리하는 노드를 통해 폴리곤 블록체인과 통신할 수 있습니다. AMB Access Polygon 서비스를 사용하여 다각형 블록체인과 상호 작용하는 분산형 애플리케이션(dApps)을 개발하고 사용할 수 있습니다. dApps의 핵심 부분은 스마트 계약입니다. AMB Access Polygon을 사용하여 스마트 계약을 생성하고 Polygon 블록체인에 배포할 수 있습니다. 또한 Polygon 네트워크의 피어인 모든 노드에서 분산 방식으로 실행되는 AMB Access Polygon 엔드포인트에 대해 JSON-RPCs를 호출하여 지갑의 잔액, 트랜잭션 세부 정보, 추정 요금 등을 확인할 수 있습니다. Polygon 네트워크에 대한 모든 피어는 스마트 계약을 개발하고 배포할 수 있습니다.

> **Important**
>
> 폴리곤 주소를 생성, 유지 관리, 사용 및 관리하는 것은 사용자의 책임입니다. 또한 Polygon 주소의 내용에 대한 책임도 있습니다. AWS 는 Amazon Managed Blockchain에서 Polygon 노드를 사용하여 배포되거나 호출된 트랜잭션에 대해 책임을 지지 않습니다.

Amazon Managed Blockchain(AMB) Access Polygon 사용에 대한 고려 사항 및 제한 사항

Amazon Managed Blockchain(AMB) Access Polygon을 사용하는 경우 다음을 고려하세요.

- 지원되는 다각형 네트워크

  AMB Access Polygon은 다음과 같은 퍼블릭 네트워크를 지원합니다.

  - 메인넷 proof-of-stake 합의로 보호되고 폴리곤(POL) 토큰이 발급 및 처리되는 퍼블릭 폴리곤 블록체인입니다. Mainnet의 트랜잭션은 실제 값(즉, 실제 비용이 발생함)을 가지며 퍼블릭 블록체인에 기록됩니다.

- Polygon에서 더 이상 지원하지 않는 네트워크

  - Polygon Labs에서 전달한 대로 뭄바이 테스트넷 네트워크는 4월 중순에 일몰합니다. 이 뉴스에 따라 AMB Access Polygon은 2024년 4월 15일에 뭄바이 테스트넷 지원을 종료했습니다. 테스트 워크로드에는 Amoy Testnet을 사용하는 것이 좋습니다.
  - 프라이빗 네트워크는 지원되지 않습니다.
  - 또한 AMB Access Polygon에는 Polygon zkEVM 네트워크에 대한 지원이 포함되지 않습니다.

- 널리 사용되는 타사 프로그래밍 라이브러리와의 호환성

  AMB Access Polygon은 ethers.js와 같은 인기 있는 프로그래밍 라이브러리와 호환되므로 개발자는 익숙한 도구를 사용하여 기존 구현과 쉽게 통합하거나 새 애플리케이션을 빠르게 개발할 수 있습니다.

- 지원되는 리전:

  이 서비스는 미국 동부(버지니아 북부) 리전에서만 지원됩니다.

- Service endpoints

  다음은 AMB Access Polygon의 서비스 엔드포인트입니다. 서비스와 연결하려면 지원되는 리전 중 하나가 포함된 엔드포인트를 사용해야 합니다.

  - `mainnet.polygon.managedblockchain.us-east-1.amazonaws.com`

- 스테이킹이 지원되지 않음

  AMB Access Polygon은 proof-of-stake에 대해 Polygon(POL) 검사기 노드를 지원하지 않습니다.

- Polygon JSON-RPC 요청의 서명 버전 4 서명

Amazon Managed Blockchain에서 Polygon JSON-RPCs를 호출할 때 서명 버전 4 서명 프로세스를 사용하여 인증된 HTTPS 연결을 통해 호출할 수 있습니다. 즉, 계정의 AWS 승인된 IAM 보안 주체만 Polygon JSON-RPC를 호출할 수 있습니다. 이렇게 하려면 호출과 함께 AWS 자격 증명(액세스 키 ID 및 보안 액세스 키)을 제공해야 합니다.

> **Important**
> - 사용자 대면 애플리케이션에 클라이언트 자격 증명을 포함하지 마십시오.
> - IAM 정책을 사용하여 개별 Polygon JSON-RPCs에 대한 액세스를 제한할 수 없습니다.

- 토큰 기반 액세스 지원

  또한 서명 버전 4(SigV4) 서명 프로세스의 편리한 대안으로 Accessor 토큰을 사용하여 Polygon 네트워크 엔드포인트에 대한 JSON-RPC 호출을 수행할 수 있습니다. 생성하여 호출과 함께 파라미터로 추가하는 Accessor 토큰 중 하나`BILLING_TOKEN`에서를 제공해야 합니다.

  > **Important**
  > - 편의보다 보안 및 감사 가능성을 우선시하는 경우 SigV4 서명 프로세스를 대신 사용합니다.
  > - 서명 버전 4(SigV4) 및 토큰 기반 액세스를 사용하여 Polygon JSON-RPCs에 액세스할 수 있습니다. 그러나 두 프로토콜을 모두 사용하도록 선택하면 요청이 거부됩니다.
  > - 사용자 대면 애플리케이션에는 Accessor 토큰을 임베드해서는 안 됩니다.

- 원시 트랜잭션 제출만 지원됩니다.

  `eth_sendrawtransaction` JSON-RPC를 사용하여 다각형 블록체인 상태를 업데이트하는 트랜잭션을 제출합니다.

Amazon Managed Blockchain(AMB) 액세스 다각형 설정

Amazon Managed Blockchain(AMB) Access Polygon을 처음 사용하기 전에이 섹션의 단계에 따라를 생성합니다 AWS 계정. 다음 장에서는 AMB Access Polygon 사용을 시작하는 방법을 설명합니다.

AMB Access Polygon을 사용하기 위한 사전 조건

AWS 를 처음 사용하려면 먼저이 있어야 합니다 AWS 계정.

에 가입 AWS

에 가입하면 Amazon Managed Blockchain(AMB) Access Polygon을 AWS 서비스포함한 모든에 AWS 가 자동으로 등록 AWS 계정 됩니다. 사용한 서비스에 대해서만 청구됩니다.

AWS 계정 이 이미 있는 경우 다음 단계로 이동합니다. AWS 계정이 없는 경우에는 다음 절차에 따라 계정을 만드세요.

를 생성하려면 AWS 계정

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

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

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

적절한 권한을 가진 IAM 사용자 생성

AMB Access Polygon을 생성하고 사용하려면 필요한 관리형 블록체인 작업을 허용하는 권한이 있는 AWS Identity and Access Management (IAM) 보안 주체(사용자 또는 그룹)가 있어야 합니다.

Amazon Managed Blockchain에서 Polygon JSON-RPCs 호출할 때 서명 버전 4 서명 프로세스를 사용하여 인증된 HTTPS 연결을 통해 호출할 수 있습니다. 즉, 계정의 AWS 승인된 IAM 보안 주체만

Polygon JSON-RPC를 호출할 수 있습니다. 이렇게 하려면 호출과 함께 AWS 자격 증명(액세스 키 ID 및 보안 액세스 키)을 제공해야 합니다.

또한 서명 버전 4(SigV4) 서명 프로세스의 편리한 대안으로 Accessor 토큰을 사용하여 Polygon 네트워크 엔드포인트에 대한 JSON-RPC 호출을 수행할 수 있습니다. 생성하여 호출과 함께 파라미터로 추가하는 Accessor 토큰 중 하나`BILLING_TOKEN`에서를 제공해야 합니다. 그러나 AWS Management Console AWS CLI및 SDK를 사용하여 Accessor 토큰을 생성할 수 있는 권한을 얻으려면 여전히 IAM 액세스 권한이 필요합니다.

IAM 사용자를 생성하는 방법에 대한 자세한 내용은 계정에서 IAM 사용자 생성을 참조하세요 AWS. 사용자에게 권한 정책을 연결하는 방법에 대한 자세한 내용은 IAM 사용자의 권한 변경을 참조하세요. 사용자에게 AMB Access Polygon 작업 권한을 부여하는 데 사용할 수 있는 권한 정책의 예는 섹션을 참조하세요Amazon Managed Blockchain(AMB) Access Polygon에 대한 자격 증명 기반 정책 예제.

설치 및 구성 AWS Command Line Interface

아직 설치하지 않은 경우 터미널의 AWS 리소스로 작업할 최신 AWS Command Line Interface (AWS CLI)을 설치합니다. 자세한 내용은 최신 버전의 AWS CLI설치 또는 업데이트를 참조하세요.

> **Note**
>
> CLI 액세스를 위해서는 액세스 키 ID 및 비밀 액세스 키가 필요합니다. 가능하다면 장기 액세스 키 대신 임시 보안 인증 정보를 사용하세요. 임시 보안 인증도 액세스 키 ID와 비밀 액세스 키로 구성되지만 보안 인증이 만료되는 시간을 나타내는 보안 토큰이 포함되어 있습니다. 자세한 내용은 IAM 사용 설명서의 AWS 리소스에서 임시 자격 증명 사용을 참조하세요.

Amazon Managed Blockchain(AMB) Access Polygon 시작하기

이 섹션의 정보와 절차를 사용하여 Amazon Managed Blockchain(AMB) Access Polygon을 시작합니다.

주제

Polygon 블록체인 네트워크에 액세스하기 위한 IAM 정책 생성

Polygon Mainnet의 퍼블릭 엔드포인트에 액세스하여 JSON-RPC 호출을 수행하려면 Amazon Managed Blockchain(AWS_ACCESS_KEY_IDAMBAWS_SECRET_ACCESS_KEY) Access Polygon에 대한 적절한 IAM 권한이 있는 사용자 자격 증명(및)이 있어야 합니다. 이 AWS CLI 설치된 터미널에서 다음 명령을 실행하여 두 다각형 엔드포인트에 모두 액세스하는 IAM 정책을 생성합니다.

```
cat <<EOT > ~/amb-polygon-access-policy.json
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid" : "AMBPolygonAccessPolicy",
            "Effect": "Allow",
            "Action": [
                "managedblockchain:InvokeRpcPolygon*"
            ],
            "Resource": "*"
        }
    ]
}
EOT
aws iam create-policy --policy-name AmazonManagedBlockchainPolygonAccess --policy-
document file://$HOME/amb-polygon-access-policy.json
```

> **Note**
>
> 이전 예제에서는 사용 가능한 모든 Polygon 네트워크에 액세스할 수 있습니다. 특정 엔드포인트에 액세스하려면 다음 `Action` 명령을 사용합니다.
>
> - `"managedblockchain:InvokeRpcPolygonMainnet"`

정책을 생성한 후 해당 정책을 IAM 사용자의 역할에 연결하면 정책이 적용됩니다.
에서 IAM 서비스로 AWS Management Console이동하여 IAM 사용자에게 할당된 `AmazonManagedBlockchainPolygonAccess` 역할에 정책을 연결합니다.

를 사용하여 AMB Access RPC 편집기에서 Polygon 원격 프로시저 호출(RPC) 요청 수행 AWS Management Console

AMB Access Polygon을 AWS Management Console 사용하여에서 원격 프로RPCs)을 편집, 구성 및 제출할 수 있습니다. 이러한 RPCs 사용하면 데이터 검색 및 다각형 네트워크에 트랜잭션 제출을 포함하여 다각형 네트워크에서 데이터를 읽고 트랜잭션을 쓸 수 있습니다.

Example

다음 예제에서는 `eth_getBlockByNumber` RPC를 사용하여 최신 블록에 대한 정보를 가져오는 방법을 보여줍니다. 강조 표시된 변수를 자체 입력으로 변경하거나 나열된 RPC 메서드 중 하나를 선택하고 필요한 관련 입력에를 입력합니다.

1. https://console.aws.amazon.com/managedblockchain/ 관리형 블록체인 콘솔을 엽니다.
2. RPC 편집기를 선택합니다.
3. 요청 섹션에서 *#### ####POLYGON_MAINNET*로를 선택합니다.
4. RPC 메*eth_getBlockByNumber*서드로를 선택합니다.
5. *## ###* **latest**를 입력하고 전체 트랜잭션 플래그*False*로를 선택합니다.
6. 그런 다음 RPC 제출을 선택합니다.
7. 응답 섹션에서 `latest` 블록의 결과를 확인할 수 있습니다. 그런 다음 전체 원시 트랜잭션을 복사하여 추가 분석을 수행하거나 애플리케이션의 비즈니스 로직에 사용할 수 있습니다.

자세한 내용은 AMB Access Polygon에서 지원하는 RPCs.

를 awscurl 사용하여에서 AMB Access Polygon JSON-RPC 요청 수행 AWS CLI

Example

AMB Access Polygon 엔드포인트에 Polygon JSON-RPC 요청을 하려면 서명 버전 4(SigV4)를 사용하여 IAM 사용자 자격 증명으로 요청에 서명합니다. awscurl 명령줄 도구는 SigV4를 사용하여 AWS 서비스에 대한 요청에 서명하는 데 도움이 될 수 있습니다. 자세한 내용은 awscurl README.md 참조하십시오.

운영 체제에 적합한 방법을 awscurl 사용하여를 설치합니다. macOS에서는 HomeBrew가 권장되는 애플리케이션입니다.

```
brew install awscurl
```

를 이미 설치하고 구성한 경우 AWS CLI IAM 사용자 자격 증명과 기본값 AWS 리전 이 환경에 설정되고에 액세스할 수 있습니다awscurl. 를 사용하여 eth_getBlockByNumber RPC를 호출하여 Polygon Mainnet에 요청을 awscurl제출합니다. 이 호출은 정보를 검색하려는 블록 번호에 해당하는 문자열 파라미터를 수락합니다.

다음 명령은 params 배열의 블록 번호를 사용하여 헤더를 검색할 특정 블록을 선택하여 Polygon Mainnet에서 블록 데이터를 검색합니다.

```
awscurl -X POST -d '{ "jsonrpc": "2.0", "id": "eth_getBlockByNumber-curltest",
 "method":"eth_getBlockByNumber", "params":["latest", false] }' --service
 managedblockchain https://mainnet.polygon.managedblockchain.us-east-1.amazonaws.com -k
```

> **Tip**
>
> 토큰을 사용하여 curl 및 AMB 액세스 토큰 기반 액세스 기능을 사용하여 동일한 요청을 할 수도 Accessor 있습니다. 자세한 내용은 AMB Access Polygon 요청을 위한 토큰 기반 액세스를 위한 Accessor 토큰 생성 및 관리 단원을 참조하십시오.
>
> ```
> curl -X POST -d '{"jsonrpc":"2.0", "id": "eth_getBlockByNumber-curltest",
>  "method":"eth_getBlockByNumber", "params":["latest", false] }'
>  'https://mainnet.polygon.managedblockchain.us-east-1.amazonaws.com?
> billingtoken=your-billing-token'
> ```

두 명령의 응답은 최신 블록에 대한 정보를 반환합니다. 설명을 위해 다음 예제를 참조하세요.

```
{"error":null,"id":"eth_getBlockByNumber-curltest","jsonrpc":"1.0",
      "result":{"baseFeePerGas":"0x873bf591e","difficulty":"0x18",
      "extraData":"0xd78301000683626f7288676f312e32312e32856c696e757800000000000000000009a
\
      423a58511085d90eaf15201a612af21ccbf1e9f8350455adaba0d27eff0ecc4133e8cd255888304cc
\
      67176a33b451277c2c3c1a6a6482d2ec25ee1573e8ba000",
      "gasLimit":"0x1c9c380","gasUsed":"0x14ca04d",
      "hash":"0x1ee390533a3abc3c8e1306cc1690a1d28d913d27b437c74c761e1a49********;",
      "nonce":"0x0000000000000000","number":"0x2f0ec4d",

 "parentHash":"0x27d47bc2c47a6d329eb8aa62c1353f60e138fb0c596e3e8e9425de163afd6dec",

 "receiptsRoot":"0x394da96025e51cc69bbe3644bc4e1302942c2a6ca6bf0cf241a5724c74c063fd",

 "sha3Uncles":"0x1dcc4de8dec75d7aab85b567b6ccd41ad312451b948a7413f0a142fd40d49347",
      "size":"0xbd6b",
      "stateRoot":"0x7ca9363cfe9baf4d1c0dca3159461b2cca8604394e69b30af05d7d5c1beea6c3",
      "timestamp":"0x653ff542",
      "totalDifficulty":"0x33eb01dd","transactions":[...],

 "transactionsRoot":"0xda1602c66ffd746dd470e90a47488114a9d00f600ab598466ecc0f3340b24e0c",
      "uncles":[]}}
```

Node.js에서 Polygon JSON-RPC 요청

HTTPS를 사용하여 서명된 요청을 제출하여 Node.js의 기본 https 모듈을 사용하여 Polygon Mainnet 네트워크에 액세스하거나 AXIOS와 같은 타사 라이브러리를 사용하여 Polygon JSON-RPCs를 호출할 수 있습니다. https://nodejs.org/api/https.html 다음 Node.js 예제에서는 서명 버전 4(SigV4) 및 토큰 기반 액세스를 모두 사용하여 AMB Access Polygon 엔드포인트에 Polygon JSON-RPC 요청을 수행하는 방법을 보여줍니다. 첫 번째 예제에서는 한 주소에서 다른 주소로 트랜잭션을 전송하고 다음 예제에서는 블록체인에서 트랜잭션 세부 정보 및 밸런스 정보를 요청합니다.

Example

이 예제 Node.js 스크립트를 실행하려면 다음 사전 조건을 적용합니다.

1. 시스템에 노드 버전 관리자(nvm) 및 Node.js가 설치되어 있어야 합니다. OS에 대한 설치 지침은 여기에서 확인할 수 있습니다.
2. `node --version` 명령을 사용하여 노드 버전 18 이상을 사용하고 있는지 확인합니다. 필요한 경우 `nvm install v18.12.0` 명령을 사용한 다음 `nvm use v18.12.0` 명령을 사용하여 노드의 LTS 버전인 버전 18을 설치할 수 있습니다.
3. 환경 변수 `AWS_ACCESS_KEY_ID` 및 에는 계정과 연결된 자격 증명이 포함되어야 `AWS_SECRET_ACCESS_KEY` 합니다.

   다음 명령을 사용하여 이러한 변수를 클라이언트에서 문자열로 내보냅니다. 다음 문자열의 빨간색 값을 IAM 사용자 계정의 적절한 값으로 바꿉니다.

   ```
   export AWS_ACCESS_KEY_ID="AKIAIOSFODNN7EXAMPLE"
   export AWS_SECRET_ACCESS_KEY="wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY"
   ```

모든 사전 조건을 완료한 후 원하는 코드 편집기를 사용하여 다음 파일을 로컬 환경의 디렉터리에 복사합니다.

package.json

```
{
  "name": "polygon-rpc",
  "version": "1.0.0",
  "description": "",
  "main": "index.js",
  "scripts": {
    "test": "echo \"Error: no test specified\" && exit 1"
  },
  "author": "",
  "license": "ISC",
  "dependencies": {
    "ethers": "^6.8.1",
    "@aws-crypto/sha256-js": "^5.2.0",
    "@aws-sdk/credential-provider-node": "^3.360.0",
    "@aws-sdk/protocol-http": "^3.357.0",
    "@aws-sdk/signature-v4": "^3.357.0",
    "axios": "^1.6.2"
  }
}
```

dispatch-evm-rpc.js

```
const axios = require("axios");
const SHA256 = require("@aws-crypto/sha256-js").Sha256;
const defaultProvider = require("@aws-sdk/credential-provider-node").defaultProvider;
const HttpRequest = require("@aws-sdk/protocol-http").HttpRequest;
const SignatureV4 = require("@aws-sdk/signature-v4").SignatureV4;

// define a signer object with AWS service name, credentials, and region
const signer = new SignatureV4({
  credentials: defaultProvider(),
  service: "managedblockchain",
  region: "us-east-1",
  sha256: SHA256,
});
const rpcRequest = async (rpcEndpoint, rpc) => {

  // parse the URL into its component parts (e.g. host, path)
  let url = new URL(rpcEndpoint);

  // create an HTTP Request object
  const req = new HttpRequest({
    hostname: url.hostname.toString(),
    path: url.pathname.toString(),
    body: JSON.stringify(rpc),
    method: "POST",
    headers: {
      "Content-Type": "application/json",
      "Accept-Encoding": "gzip",
      host: url.hostname,
    },
  });

  // use AWS SignatureV4 utility to sign the request, extract headers and body
  const signedRequest = await signer.sign(req, { signingDate: new Date() });

  try {
    //make the request using axios
    const response = await axios({
      ...signedRequest,
      url: url,
      data: req.body,
    });
    return response.data;
```

```
  } catch (error) {
    console.error("Something went wrong: ", error);
  }
};

module.exports = { rpcRequest: rpcRequest };
```

sendTx.js

> **Warning**
>
> 다음 코드는 하드 코딩된 프라이빗 키를 사용하여 데모Ethers.js용으로만를 사용하여 Wallet Signer를 생성합니다. 실제 자금이 있고 보안 위험이 있으므로 프로덕션 환경에서는이 코드를 사용하지 마십시오.
> 필요한 경우 계정 팀에 문의하여 wallet 및 Signer 모범 사례에 대해 조언하세요.

```
const ethers = require("ethers");

//set AMB Access Polygon endpoint using token based access (TBA)
let token = "your-billing-token"
let url = `https://mainnet.polygon.managedblockchain.us-east-1.amazonaws.com?
billingtoken=${token}`;

//prevent batch RPCs
let options = {
  batchMaxCount: 1,
};

//create JSON RPC provider with AMB Access endpoint and options
let provider = new ethers.JsonRpcProvider(url, null, options);

let sendTx = async (to) => {
  //create an instance of the Wallet class with a private key
  //DO NOT USE A WALLET YOU USE ON MAINNET, NEVER USE A RAW PRIVATE KEY IN PROD
  let pk = "wallet-private-key";
  let signer = new ethers.Wallet(pk, provider);

  //use this wallet to send a transaction of POL from one address to another
  const tx = await signer.sendTransaction({
```

```
    to: to,
    value: ethers.parseUnits("0.0001", "ether"),
  });

  console.log(tx);
};

sendTx("recipent-address");
```

readTx.js

```
let rpcRequest = require("./dispatch-evm-rpc").rpcRequest;
let ethers = require("ethers");

let getTxDetails = async (txHash) => {
  //set url to a Signature Version 4 endpoint for AMB Access
  let url = "https://mainnet.polygon.managedblockchain.us-east-1.amazonaws.com";

  //set RPC request body to get transaction details
  let getTransactionByHash = {
    id: "1",
    jsonrpc: "2.0",
    method: "eth_getTransactionByHash",
    params: [txHash],
  };

  //make RPC request for transaction details
  let txDetails = await rpcRequest(url, getTransactionByHash);

  //set RPC request body to get recipient user balance
  let getBalance = {
    id: "2",
    jsonrpc: "2.0",
    method: "eth_getBalance",
    params: [txDetails.result.to, "latest"],
  };

  //make RPC request for recipient user balance
  let recipientBalance = await rpcRequest(url, getBalance);

  console.log("TX DETAILS: ", txDetails.result, "BALANCE: ",
 ethers.formatEther(recipientBalance.result));
};
```

```
getTxDetails("your-transaction-id");
```

이러한 파일이 디렉터리에 저장되면 다음 명령을 사용하여 코드를 실행하는 데 필요한 종속성을 설치합니다.

```
npm install
```

Node.js로 트랜잭션 전송

이전 예제에서는 트랜잭션에 서명하고 AMB Access Polygon을 사용하여 Polygon Mainnet으로 브로드캐스트하여 한 주소에서 다른 주소로 네이티브 Polygon Mainnet 토큰(POL)을 보냅니다. 이렇게 `sendTx.js` 하려면 Ethereum 및 Polygon과 같은 Ethereum 호환 블록체인과 상호 작용하기 위해 널리 사용되는 라이브러리인 `Ethers.js`를 사용하는 스크립트를 사용합니다. 빨간색으로 강조 표시된 코드의 세 가지 변수를 바꿔야 합니다. 여기에는 토큰 기반 액세스를 `billingToken` 위한 액세스 도구 토큰의 , 트랜잭션에 서명하는 프라이빗 키, POL을 수신하는 수신자의 주소가 포함됩니다.

> **Tip**
>
> 기존 지갑을 재사용하여 자금 손실 위험을 없애는 대신이 목적으로 새 프라이빗 키(지갑)를 생성하는 것이 좋습니다. Ethers 라이브러리의 Wallet 클래스 메서드 createRandom()을 사용하여 테스트할 Wallet을 생성할 수 있습니다. 또한 Polygon Mainnet에서 POL을 요청해야 하는 경우 퍼블릭 POL 수도 꼭지를 사용하여 테스트에 사용할 소량을 요청할 수 있습니다.

`billingToken`, 조달된 Wallet의 프라이빗 키 및 수신자의 주소가 코드에 추가되면 다음 코드를 실행하여 .0001 POL이 주소에서 다른 주소로 전송되도록 트랜잭션에 서명하고 AMB Access Polygon을 사용하여 `eth_sendRawTransaction` JSON-RPC를 호출하는 Polygon Mainnet으로 브로드캐스트합니다.

```
node sendTx.js
```

수신된 응답은 다음과 유사합니다.

```
TransactionResponse {
provider: JsonRpcProvider {},
```

```
blockNumber: null,
blockHash: null,
index: undefined,
hash: '0x8d7538b4841261c5120c0a4dd66359e8ee189e7d1d34ac646a1d9923********',
type: 2,
to: '0xd2bb4f4f1BdC4CB54f715C249Fc5a991********',
from: '0xcf2C679AC6cb7de09Bf6BB6042ecCF05********',
nonce: 2,
gasLimit: 21000n,
gasPrice: undefined,
maxPriorityFeePerGas: 16569518669n,
maxFeePerGas: 16569518685n,
data: '0x',
value: 100000000000000n,
chainId: 80001n,
signature: Signature {
r: "0x1b90ad9e9e4e005904562d50e904f9db10430a18b45931c059960ede337238ee",
s: "0x7df3c930a964fd07fed4a59f60b4ee896ffc7df4ea41b0facfe82b470db448b7",
yParity: 0,
networkV: null
},
accessList: []
}
```

응답은 트랜잭션 수신을 구성합니다. 속성의 값을 저장합니다`hash`. 블록체인에 방금 제출한 트랜잭션의 식별자입니다. 읽기 트랜잭션 예제에서이 속성을 사용하여 Polygon Mainnet에서이 트랜잭션에 대한 추가 세부 정보를 가져옵니다.

및 `blockNumber`는 `blockHash`응답`null`에 있습니다. 이는 트랜잭션이 아직 Polygon 네트워크의 블록에 기록되지 않았기 때문입니다. 이러한 값은 나중에 정의되며 다음 섹션에서 트랜잭션 세부 정보를 요청할 때 표시될 수 있습니다.

Node.js에서 트랜잭션 읽기

이 섹션에서는 이전에 제출한 트랜잭션에 대한 트랜잭션 세부 정보를 요청하고 AMB Access Polygon을 사용하여 Polygon Mainnet에 대한 읽기 요청을 사용하여 수신자 주소의 POL 밸런스를 검색합니다. `readTx.js` 파일에서 레이블이 지정된 변수를 이전 섹션의 코드를 실행한 응답에서 저장`hash`한 변수*`your-transaction-id`*로 바꿉니다.

이 코드는 AWS SDK의 필수 서명 버전 4(SigV4) 모듈을 사용하여 AMB Access Polygon에 대한 HTTPS 요청에 `dispatch-evm-rpc.js`서명하고 널리 사용되는 HTTP 클라이언트인 AXIOS를 사용하여 요청을 보내는 유틸리티를 사용합니다.

수신된 응답은 다음과 유사합니다.

```
TX DETAILS: {
blockHash: '0x59433e0096c783acab0659175460bb3c919545ac14e737d7465b3ddc********',
blockNumber: '0x28b4059',
from: '0xcf2c679ac6cb7de09bf6bb6042eccf05b7fa1394',
gas: '0x5208',
gasPrice: '0x3db9eca5d',
maxPriorityFeePerGas: '0x3db9eca4d',
maxFeePerGas: '0x3db9eca5d',
hash: '0x8d7538b4841261c5120c0a4dd66359e8ee189e7d1d34ac646a1d9923********',
input: '0x',
nonce: '0x2',
to: '0xd2bb4f4f1bdc4cb54f715c249fc5a991********',
transactionIndex: '0x0',
value: '0x5af3107a4000',
type: '0x2',
accessList: [],
chainId: '0x13881',
v: '0x0',
r: '0x1b90ad9e9e4e005904562d50e904f9db10430a18b45931c059960ede337238ee',
s: '0x7df3c930a964fd07fed4a59f60b4ee896ffc7df4ea41b0facfe82b470db448b7'
} BALANCE: 0.0003
```

응답은 트랜잭션 세부 정보를 나타냅니다. 이제 `blockHash` 및가 정의`blockNumber`되었을 수 있습니다. 이는 트랜잭션이 블록에 기록되었음을 나타냅니다. 이러한 값이 여전히 인 경우 몇 분 기`null`다린 다음 코드를 다시 실행하여 트랜잭션이 블록에 포함되었는지 확인합니다. 마지막으로 수신자 주소 밸런스(0x110d9316ec000)의 16진수 표현은 Ethers의 `formatEther()` 메서드를 사용하여 10진수로 변환됩니다.이 메서드는 16진수를 10진수로 변환하고 소수 자릿수를 18(10^18)만큼 전환하여 POL에서 실제 밸런스를 제공합니다.

> **Tip**
>
> 위의 코드 예제에서는 Node.js, Ethers 및 Axios를 사용하여 AMB Access Polygon에서 지원되는 몇 가지 JSON-RPCs를 활용하는 방법을 보여 주지만,이 서비스를 사용하여 예제를 수정하고 다른 코드를 작성하여 Polygon에서 애플리케이션을 빌드할 수 있습니다. AMB Access Polygon에서 지원되는 JSON-RPCsAMB Access Polygon에서 지원되는 Managed Blockchain API 및 JSON-RPCs.

AMB Access Polygon 요청을 위한 토큰 기반 액세스를 위한 Accessor 토큰 생성 및 관리

또한 서명 버전 4(SigV4) 서명 프로세스의 편리한 대안으로 Accessor 토큰을 사용하여 Polygon 네트워크 엔드포인트에 대한 JSON-RPC 호출을 수행할 수 있습니다. 생성하여 호출과 함께 파라미터로 추가하는 Accessor 토큰 중 하나`BILLING_TOKEN`에서를 제공해야 합니다.

> **Important**
>
> - 편의보다 보안 및 감사 가능성을 우선시하는 경우 SigV4 서명 프로세스를 대신 사용합니다.
> - 서명 버전 4(SigV4) 및 토큰 기반 액세스를 사용하여 Polygon JSON-RPCs에 액세스할 수 있습니다. 그러나 두 프로토콜을 모두 사용하도록 선택하면 요청이 거부됩니다.
> - 사용자 대면 애플리케이션에는 Accessor 토큰을 임베드해서는 안 됩니다.

콘솔의 토큰 액세스 관리자 페이지에는 클라이언트의 AWS 계정 코드에서에서 AMB Access Polygon JSON-RPC를 호출하는 데 사용할 수 있는 모든 액세스 관리자 토큰 목록이 표시됩니다.

AMB Access Polygon JSON-RPC 요청에 대한 자세한 내용은 섹션을 참조하세요AMB Access Polygon에서 지원되는 Managed Blockchain API 및 JSON-RPCs.

를 사용하여 Accessor 토큰을 생성하고 관리할 수 있습니다 AWS Management Console. `CreateAccessor`, , 및 API 작업을 사용하여 Accessor 토큰을 생성하고 관리할 수도 있습니다 `GetAccessorListAccessors DeleteAccessor`. `BILLING_TOKEN`는 Accessor의 속성입니다. 이 `BILLING_TOKEN` 속성은 Accessor를 추적하고에서 이루어진 AMB Access Polygon JSON-RPC 요청을 결제하는 데 사용됩니다 AWS 계정.

Accessor 토큰 생성 및 관리와 관련된 모든 API 작업은 AWS Management Console AWS CLI및 SDKs.

토큰 기반 액세스를 위한 액세스 토큰 생성

Accessor 토큰을 생성하고 이를 사용하여의 모든 AMB Access Polygon 노드에서 AMB Access Polygon API를 호출할 수 있습니다 AWS 계정.

를 사용하여 AMB Access Polygon JSON-RPC 요청을 하는 액세스 토큰 생성 AWS Management Console

1. https://console.aws.amazon.com/managedblockchain/ 관리형 블록체인 콘솔을 엽니다.
2. 토큰 액세서를 선택합니다.
3. Create Accessor를 선택합니다.
4. 유효한 다각형 블록체인 네트워크를 선택합니다.
5. 선택 사항으로, Accessor에 대한 태그를 추가합니다.
6. 새 Accessor 토큰을 생성하려면 Create Accessor를 선택합니다.

를 사용하여 AMB Access Polygon JSON-RPC 요청을 하는 액세스 토큰 생성 AWS CLI

```
aws managedblockchain create-accessor --accessor-type BILLING_TOKEN --network-type
 POLYGON_MAINNET
```

다음 예제와 `BillingToken`같이 이전 명령은와 `AccessorId` 함께를 반환합니다.

```
{
"AccessorId": "ac-NGQ6QNKXLNEBXD3UI6********",
"NetworkType": "POLYGON_MAINNET",
"BillingToken": "jZlP8OUI-PcQSKINyX9euJJDC5-IcW9e-n********"
}
```

응답의 키 요소는 입니다`BillingToken`. 이 속성을 사용하여 AMB Access Polygon JSON-RPC 호출을 수행할 수 있습니다. 이 예제의 일부 값은 보안상의 이유로 난독화되었지만 실제 응답에 완전히 표시됩니다.

> **Note**
>
> 작업이 실행된 후 관리형 블록체인은 토큰을 프로비저닝하고 구성합니다. 이 프로세스의 길이는 많은 변수에 따라 달라집니다.

Accessor 토큰 세부 정보 보기

AWS 계정 소유한 각 Accessor 토큰의 속성을 볼 수 있습니다. 예를 들어, Accessor ID 또는 Accessor의 Amazon 리소스 이름(ARN)을 볼 수 있습니다. 상태, 유형, 생성 날짜 및 도 볼 수 있습니다`BillingToken`.

를 사용하여 액세스 토큰의 정보를 보려면 AWS Management Console

1. https://console.aws.amazon.com/managedblockchain/ 관리형 블록체인 콘솔을 엽니다.
2. 탐색 창에서 토큰 액세서를 선택합니다.
3. 목록에서 토큰의 Accessor ID를 선택합니다.

토큰 세부 정보 페이지에 팝업이 표시됩니다. 이 페이지에서 토큰의 속성을 볼 수 있습니다.

를 사용하여 Accessor 토큰의 정보를 보려면 AWS CLI

다음 명령을 실행하여 Accessor 토큰의 세부 정보를 봅니다. 의 값을 사용자의 액세스 ID`--accessor-id`로 바꿉니다.

```
aws managedblockchain get-accessor --accessor-id ac-NGQ6QNKXLNEBXD3UI6********
```

`BillingToken` 및 기타 키 속성은 다음 예제와 같이 반환됩니다. 이 예제의 일부 값은 보안상의 이유로 난독화되었지만 실제 응답에 완전히 표시됩니다.

```
{
  "Accessor": {
  "Id": "ac-NGQ6QNKXLNEBXD3UI6********",
  "Type": "BILLING_TOKEN",
  "BillingToken": "jZlP8OUI-PcQSKINyX9euJJDC5-IcW9e-n********",
  "Status": "AVAILABLE",
  "NetworkType": "POLYGON_MAINNET"
  "CreationDate": "2022-01-04T23:09:47.750Z",
  "Arn": "arn:aws:managedblockchain:us-east-1:666666666666:accessors/ac-
NGQ6QNKXLNEBXD3UI6********"
  }
}
```

Accessor 토큰 삭제

Accessor 토큰을 삭제하면 토큰이에서 AVAILABLE PENDING_DELETION 상태로 변경됩니다. PENDING_DELETION 상태에는 Accessor 토큰을 사용할 수 없습니다.

를 사용하여 Accessor 토큰을 삭제하려면 AWS Management Console

1. https://console.aws.amazon.com/managedblockchain/ 관리형 블록체인 콘솔을 엽니다.
2. 탐색 창에서 토큰 액세스 관리자를 선택합니다.
3. 목록에서 원하는 Accessor 토큰을 선택합니다.
4. 삭제를 선택합니다.
5. 선택 내용을 확인합니다.

삭제된 액세스 도구 토큰과 함께 토큰 액세스 도구 페이지로 돌아갑니다. 페이지에 PENDING_DELETION 상태가 표시됩니다.

를 사용하여 Accessor 토큰을 삭제하려면 AWS CLI

다음 예제에서는 토큰을 삭제하는 방법을 보여줍니다. delete-accessor 명령을 사용하여 토큰을 삭제합니다. 의 값을 액세스 ID--accessor-id로 설정합니다.

AWS CLI를 사용하여 Accessor 토큰 삭제

```
aws managedblockchain delete-accessor --accessor-id ac-NGQ6QNKXLNEBXD3UI6********
```

이 명령이 성공적으로 실행되면 메시지가 반환되지 않습니다.

AMB Access Polygon에서 지원되는 Managed Blockchain API 및 JSON-RPCs

Amazon Managed Blockchain은 AMB Access Polygon용 토큰 액세스 관리자를 생성하고 관리하기 위한 API 작업을 제공합니다. 자세한 내용은 관리형 블록체인 API 참조 안내서를 참조하세요.

다음 주제에서는 AMB Access Polygon이 지원하는 Polygon JSON-RPCs의 목록과 참조를 제공합니다. 지원되는 각 JSON-RPC에는 사용에 대한 간략한 설명이 있습니다. Polygon JSON-RPCs합니다.

AMB Access Polygon은 다음 JSON-RPC 메서드를 지원합니다. 지원되는 각 JSON-RPC에는 유틸리티와 기본 요청 할당량에 대한 범주와 간략한 설명이 있습니다. 해당하는 경우 Amazon Managed Blockchain에서 JSON-RPC 메서드를 사용하기 위한 고유한 고려 사항이 표시됩니다.

> **Note**
>
> - 목록에 없는 메서드는 지원되지 않습니다.
> - Amazon Managed Blockchain에서 Polygon JSON-RPCs 호출할 때 서명 버전 4 서명 프로세스를 사용하여 인증된 HTTPS 연결을 통해 호출할 수 있습니다. 즉, 계정의 AWS 승인된 IAM 보안 주체만 Polygon JSON-RPC를 호출할 수 있습니다. 이렇게 하려면 호출과 함께 AWS 자격 증명(액세스 키 ID 및 보안 액세스 키)을 제공해야 합니다.
> - 서명 버전 4(SigV4) 서명 프로세스의 편리한 대안으로 토큰 기반 액세스를 사용할 수도 있습니다. 편의를 위해 보안 및 감사 가능성을 우선시하는 경우 SigV4 서명 프로세스를 대신 사용합니다. 그러나 SigV4 및 토큰 기반 액세스를 모두 사용하는 경우 요청이 작동하지 않습니다.
> - JSON-RPC 배치 요청은이 미리 보기를 위해 Amazon Managed Blockchain(AMB) Access Polygon에서 지원되지 않습니다.
> - 다음 표의 할당량 열에는 각 JSON-RPC의 할당량이 나열되어 있습니다. 할당량은 각 JSON-RPC에 대해 폴리곤 네트워크(메인넷)당 리전당 초당 요청 수(RPS)로 설정됩니다.
>
>   할당량을 늘리려면에 문의해야 합니다 지원. 에 연락하려면에 지원로그인합니다AWS Support Center Console. 사례 생성을 선택합니다. 기술을 선택합니다. 서비스로 관리형 블록체인을 선택합니다. 액세스:Polygon을 범주로 선택하고 일반 지침을 심각도로 선택합니다. RPC 할당량을 제목으로 입력하고 설명 텍스트 상자에 리전별 다각형 네트워크당 RPS의 요구 사항에 적용되는 JSON-RPC 및 할당량 제한을 나열합니다. 사례를 제출합니다.

범주	JSON-RPC	설명	고려 사항
이더리움	eth_blockNumber	가장 최근 블록의 수를 반환합니다.	
	eth_call	는 블록체인에서 트랜잭션을 생성하지 않고 즉시 새 메시지 호출을 실행합니다.	`eth_call`는 0개의 가스를 사용하지만 필요한 메시지에 대한 가스 파라미터가 있습니다.
	eth_chainId	EIP-155에 도입된 현재 구성된 값의 정수 `Chain Id` 값을 반환합니다. 사용할 수 없는 `None` 경우 `Chain Id`를 반환합니다.	
	eth_estimateGas	트랜잭션을 블록체인에 추가하지 않고 트랜잭션에 필요한 가스를 추정하고 반환합니다.	
	eth_feeHistory	과거 가스 정보 모음을 반환합니다.	
	eth_gasPrice	Wei의 가스당 현재 가격을 반환합니다.	
	eth_getBalance	지정된 계정 주소 및 블록 식별자에 대한 계정의 잔액을 반환합니다.	

범주	JSON-RPC	설명	고려 사항
	eth_getBlockByHash	블록 해시를 사용하여 지정된 블록에 대한 정보를 반환합니다.	
	eth_getBlockByNumber	블록 번호를 사용하여 지정된 블록에 대한 정보를 반환합니다.	
	eth_getBlockReceipts	블록 번호를 사용하여 지정된 블록에 대한 수신을 반환합니다.	
	eth_getBlockTransactionCountByHash	블록 해시를 사용하여 지정된 블록의 트랜잭션 수를 반환합니다.	
	eth_getBlockTransactionCountByNumber	블록 번호를 사용하여 지정된 블록의 트랜잭션 수를 반환합니다.	
	eth_getCode	지정된 계정 주소 및 블록 식별자에서 코드를 반환합니다.	

범주	JSON-RPC	설명	고려 사항
	eth_getLogs	지정된 필터 객체에 대한 모든 로그의 배열을 반환합니다.	계약 주소가 제공될 때 기본적으로 1K 블록 범위를 사용하여 모든 블록 범위에 대해 `eth_getloqs` 요청할 수 있습니다. 활동이 높은 계약은 더 작은 블록 범위로 제한될 수 있습니다. 계약 주소가 제공되지 않으면 블록 범위는 8이 됩니다.
	eth_getRawTransactionByHash	에서 지정한 트랜잭션의 원시 형식을 반환합니다`transaction_hash` .	
	eth_getStorageAt	지정된 계정 주소 및 블록 식별자에 대해 지정된 스토리지 위치의 값을 반환합니다.	
	eth_getTransactionByBlockHashAndIndex	지정된 블록 해시 및 트랜잭션 인덱스 위치를 사용하여 트랜잭션에 대한 정보를 반환합니다.	

범주	JSON-RPC	설명	고려 사항
	eth_getTransactionByBlockNumberAndIndex	지정된 블록 번호 및 트랜잭션 인덱스 위치를 사용하여 트랜잭션에 대한 정보를 반환합니다.	
	eth_getTransactionByHash	지정된 트랜잭션 해시가 있는 트랜잭션에 대한 정보를 반환합니다.	
	eth_getTransactionCount	지정된 주소 및 블록 식별자에서 전송된 트랜잭션 수를 반환합니다.	
	eth_getTransactionReceipt	지정된 트랜잭션 해시를 사용하여 트랜잭션의 수신을 반환합니다.	
	eth_getUncleByBlockHashAndIndex	블록 해시 및 삼촌 인덱스 위치를 사용하여 지정된 삼촌 블록에 대한 정보를 반환합니다.	
	eth_getUncleByBlockNumberAndIndex	블록 번호 및 서클 인덱스 위치를 사용하여 지정된 서클 블록에 대한 정보를 반환합니다.	

범주	JSON-RPC	설명	고려 사항
	eth_getUncleCountByBlockHash	서클 해시를 사용하여 지정된 서클의 개수를 반환합니다.	
	eth_getUncleCountByBlockNumber	서클 번호를 사용하여 지정된 서클의 개수를 반환합니다.	
	eth_maxPriorityFeePerGas	현재 블록에 포함된 트랜잭션을 가져오기 위해 우선 순위 요금으로 지불할 수 있는 금액 또는 "tip"의 추정치인 가스당 요금을 반환합니다.	일반적으로이 메서드에서 반환된 값을 사용하여 제출하려는 후속 트랜잭션maxFeePerGas 에서를 설정합니다.
	eth_protocolVersion	현재 Ethereum 프로토콜 버전을 반환합니다.	
	eth_sendRawTransaction	서명된 트랜잭션에 대한 새 메시지 호출 트랜잭션 또는 계약 생성을 생성합니다.	관리형 블록체인은 원시 트랜잭션만 지원합니다. 트랜잭션을 전송하기 전에 트랜잭션을 생성하고 서명해야 합니다.

범주	JSON-RPC	설명	고려 사항
디버그	debug_traceBlockByHash	추적기를 사용하여 블록 해시에서 지정한 블록에서 모든 트랜잭션을 실행하여 가능한 추적 결과 번호를 반환합니다(추적 모드 필요).	
	debug_traceBlockByNumber	추적기를 사용하여 번호로 지정된 블록에서 모든 트랜잭션을 실행하여 추적 결과를 반환합니다(추적 모드 필요).	
	debug_traceCall	지정된 블록 실행의 컨텍스트 내에서 eth 직접 호출을 실행하여 가능한 추적 결과 수를 반환합니다(추적 모드 필요).	
	debug_traceTransaction	지정된 트랜잭션의 모든 트레이스를 반환합니다(트레이스 모드 필요).	
Net	net_version	현재 네트워크 ID를 반환합니다.	

범주	JSON-RPC	설명	고려 사항
추적	trace_block	블록에 포함된 모든 트랜잭션의 호출된 모든 옵코드의 전체 스택 트레이스를 반환합니다.	
	trace_call	지정된 블록 실행의 컨텍스트 내에서 eth 직접 호출을 실행하여 가능한 추적 결과 수를 반환합니다(추적 모드 필요).	
	trace_transaction	지정된 트랜잭션의 모든 트레이스를 반환합니다(트레이스 모드 필요).	
Tx 풀	txpool_content	대기 중이거나 대기 중인 모든 트랜잭션을 반환합니다.	
	txpool_status	현재 다음 블록에 포함 대기 중인 모든 트랜잭션 수와 대기열에 있는 트랜잭션 수를 제공합니다(향후 실행에만 예약됨).	
웹	web3_clientVersion	현재 클라이언트 버전을 반환합니다.	

Amazon Managed Blockchain(AMB) Access Polygon을 사용한 다각형 사용 사례

Polygon 블록체인은 특히 NFTs, Web3 게임 및 토큰화 사용 사례와 관련된 분산형 애플리케이션(dApps)을 구축하는 데 일반적으로 사용됩니다. 이 주제에서는 Amazon Managed Blockchain(AMB) Access Polygon을 사용하여 구현할 수 있는 몇 가지 사용 사례 목록을 제공합니다.

주제

다각형 NFT 데이터 분석

지정된 기간 동안 전송 이벤트 및 NFTs 메타데이터와 같은 정보를 포함하여 Polygon NFT에 대한 데이터를 수집할 수 있습니다. 그런 다음이 데이터를 분석하여 어떤 NFTs 유행하고 있는지 또는 어떤 사용자가 특정 컬렉션과 가장 자주 상호 작용하는지와 같은 인사이트를 얻을 수 있습니다.

자세한 내용은 AMB Access Polygon에서 지원되는 Managed Blockchain API 및 JSON-RPCs 단원을 참조하십시오.

NFT 구매 지원

AMB Access Polygon을 사용하여 초기 민트, 허용 목록 또는 보조 시장에서 NFT 구매에 대한 트랜잭션을 제출할 수 있습니다. 그런 다음 다른 AWS 서비스의 조합을 사용하여 신용 카드를 사용한 구매를 허용하고 Fiat 또는 암호화폐를 수락할 수 있으며 관련된 모든 이해 관계자를 신속하게 해결할 수 있습니다.

자세한 내용은 AMB Access Polygon에서 지원되는 Managed Blockchain API 및 JSON-RPCs 단원을 참조하십시오.

Polygon Wallet 생성

AMB Access Polygon을 사용하여 블록체인의 스마트 계약에서 사용자 토큰 밸런스를 읽거나 서명된 트랜잭션을 블록체인으로 브로드캐스팅하는 등 디지털 자산 지갑의 중요한 기능을 제공할 수 있습니다.

자세한 내용은 AMB Access Polygon에서 지원되는 Managed Blockchain API 및 JSON-RPCs 단원을 참조하십시오.

Wallet as a Service

AMB Access Polygon을 사용하여 지원되는 Polygon JSON-RPC를 사용하여 잔액, 자산 이전, 자산 전송 및 요금 추정 확인과 같은 일반적인 Wallet 트랜잭션을 지원하는 데 필요한 운영 Walletwallet-as-a-service를 개발할 수 있습니다.RPCs

자세한 내용은 AMB Access Polygon에서 지원되는 Managed Blockchain API 및 JSON-RPCs 단원을 참조하십시오.

토큰 게이팅 경험

AMB Access Polygon을 사용하여 사용자를 위한 토큰 연결 환경을 구축할 수 있습니다. 예를 들어 특정 NFT의 소유자에게만 콘텐츠에 대한 액세스 권한을 조건부로 제공할 수 있습니다. 이를 위해서는 블록체인을 읽고 사용자 주소의 NFT 소유권을 결정해야 합니다.

자세한 내용은 AMB Access Polygon에서 지원되는 Managed Blockchain API 및 JSON-RPCs 단원을 참조하십시오.

Amazon Managed Blockchain(AMB) Access Polygon용 자습서

이 섹션에서 강조 표시된 다음 자습서는 AMB Access Polygon을 사용하여 Polygon 블록체인에서 몇 가지 일반적인 작업을 수행하는 방법을 배우는 데 도움이 되는 연습을 제공하는의 커뮤니티 문서 AWS re:Post 입니다.

- AMB Access Polygon 및 web3.js를 사용하여 트랜잭션 전송
- AMB Access Polygon 및 Hardhat Ignition을 사용하여 스마트 계약 배포
- 스마트 계약과의 상호 작용
- AMB Access Polygon 및 Chainlink 데이터 피드를 사용하여 체인 외부에서 현재 가격 데이터 검색
- AMB Access를 사용하여 Polygon Mainnet에서 ERC-20 토큰 데이터 분석

Amazon Managed Blockchain(AMB) Access Polygon의 보안

의 클라우드 보안 AWS 이 가장 우선합니다. 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 모델은 이를 클라우드의 보안과 클라우드의 보안으로 설명합니다.

- 클라우드 보안 - AWS 는에서 AWS 서비스를 실행하는 인프라를 보호할 책임이 있습니다 AWS 클라우드. AWS 또한는 안전하게 사용할 수 있는 서비스를 제공합니다. 서드 파티 감사자는 정기적으로 AWS 규정 준수 프로그램의 일환으로 보안 효과를 테스트하고 검증합니다. Amazon Managed Blockchain(AMB) Access Polygon에 적용되는 규정 준수 프로그램에 대한 자세한 내용은 규정 AWS 준수 프로그램 제공 범위 내 서비스를 참조하세요.
- 클라우드의 보안 - 사용자의 책임은 사용하는 AWS 서비스에 따라 결정됩니다. 또한 사용자는 데이터의 민감도, 회사 요구 사항, 관련 법률 및 규정을 비롯한 기타 요소에 대해서도 책임이 있습니다.

Amazon Managed Blockchain은 데이터 보호, 인증 및 액세스 제어를 제공하기 위해 Managed Blockchain에서 실행되는 오픈 소스 프레임워크의 기능과 기능을 사용합니다 AWS .

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

주제

- Amazon Managed Blockchain(AMB) Access Polygon의 데이터 보호
- Amazon Managed Blockchain(AMB) Access Polygon의 ID 및 액세스 관리

Amazon Managed Blockchain(AMB) Access Polygon의 데이터 보호

AWS 공동 책임 모델 Amazon Managed Blockchain(AMB) Access Polygon의 데이터 보호에 적용됩니다. 이 모델에 설명된 대로 AWS 는 모든를 실행하는 글로벌 인프라를 보호할 책임이 있습니다 AWS 클라우드. 사용자는 인프라에서 호스팅되는 콘텐츠를 관리해야 합니다. 사용하는 AWS 서비스 의 보안 구성과 관리 태스크에 대한 책임도 사용자에게 있습니다. 데이터 프라이버시에 대한 자세한 내용은 데이터 프라이버시 FAQ를 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 사용하세요.
- 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.
- AWS 암호화 솔루션과 내부의 모든 기본 보안 제어를 사용합니다 AWS 서비스.
- Amazon S3에 저장된 민감한 데이터를 검색하고 보호하는 데 도움이 되는 Amazon Macie와 같은 고급 관리형 보안 서비스를 사용하세요.
- 명령줄 인터페이스 또는 API를 AWS 통해에 액세스할 때 FIPS 140-3 검증 암호화 모듈이 필요한 경우 FIPS 엔드포인트를 사용합니다. 사용 가능한 FIPS 엔드포인트에 대한 자세한 내용은 Federal Information Processing Standard(FIPS) 140-3을 참조하세요.

고객의 이메일 주소와 같은 기밀 정보나 중요한 정보는 태그나 이름 필드와 같은 자유 형식 텍스트 필드에 입력하지 않는 것이 좋습니다. 여기에는 AMB Access Polygon 또는 기타 AWS 서비스 에서 콘솔, API AWS CLI또는 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.

데이터 암호화

데이터 암호화는 권한이 없는 사용자가 블록체인 네트워크 및 관련 데이터 스토리지 시스템에서 데이터를 읽지 못하도록 하는 데 도움이 됩니다. 여기에는 전송 중인 데이터라고 하는 네트워크를 이동할 때 가로챌 수 있는 데이터가 포함됩니다.

전송 중 암호화

기본적으로 관리형 블록체인은 HTTPS/TLS 연결을 사용하여를 실행하는 AWS CLI 클라이언트 컴퓨터에서 AWS 서비스 엔드포인트로 전송되는 모든 데이터를 암호화합니다.

HTTPS/TLS 사용을 활성화하기 위해 어떤 조치도 필요하지 않습니다. `--no-verify-ssl` 명령을 사용하여 개별 AWS CLI 명령에 대해 명시적으로 비활성화하지 않는 한 항상 활성화됩니다.

Amazon Managed Blockchain(AMB) Access Polygon의 ID 및 액세스 관리

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

주제

대상

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

- 서비스 사용자 - 기능에 액세스할 수 없는 경우 관리자에게 권한 요청(참조Amazon Managed Blockchain(AMB) 액세스 다각형 자격 증명 및 액세스 문제 해결)
- 서비스 관리자 - 사용자 액세스 결정 및 권한 요청 제출(Amazon Managed Blockchain(AMB) Access Polygon이 IAM과 작동하는 방식 참조)
- IAM 관리자 - 액세스를 관리하기 위한 정책 작성(Amazon Managed Blockchain(AMB) Access Polygon에 대한 자격 증명 기반 정책 예제 참조)

ID를 통한 인증

인증은 자격 증명 자격 증명을 AWS 사용하여에 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 위해는 요청에 암호화 방식으로 서명할 수 있는 SDK 및 CLI를 AWS 제공합니다. 자세한 내용은 IAM 사용 설명서의 API 요청용AWS 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 증명이 필요한 작업 섹션을 참조하세요.

페더레이션 ID

가장 좋은 방법은 인간 사용자에게 자격 증명 공급자와의 페더레이션을 사용하여 임시 자격 증명을 AWS 서비스 사용하여에 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 사용자의 자격 증명입니다. 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 사용자는 단일 개인 또는 애플리케이션에 대한 특정 권한을 가진 ID입니다. 장기 자격 증명이 있는 IAM 사용자 대신 임시 자격 증명을 사용하는 것이 좋습니다. 자세한 내용은 IAM 사용 설명서의 자격 증명 공급자와의 페더레이션을 사용하여 임시 자격 증명을 AWS 사용하여에 액세스하도록 인간 사용자에게 요구하기를 참조하세요.

IAM 그룹은 IAM 사용자 모음을 지정하고 대규모 사용자 집합에 대한 관리 권한을 더 쉽게 만듭니다. 자세한 내용은 IAM 사용 설명서의 IAM 사용자 사용 사례 섹션을 참조하세요.

IAM 역할

IAM 역할은 임시 자격 증명을 제공하는 특정 권한이 있는 자격 증명입니다. 사용자에서 IAM 역할(콘솔)로 전환하거나 또는 API 작업을 호출하여 역할을 수임할 수 있습니다. AWS CLI AWS 자세한 내용은 IAM 사용 설명서의 역할 수임 방법을 참조하세요.

IAM 역할은 페더레이션 사용자 액세스, 임시 IAM 사용자 권한, 교차 계정 액세스, 교차 서비스 액세스 및 Amazon EC2에서 실행되는 애플리케이션에 유용합니다. 자세한 내용은 IAM 사용 설명서의 교차 계정 리소스 액세스를 참조하세요.

정책을 사용하여 액세스 관리

정책을 AWS 생성하고 자격 증명 또는 리소스에 연결하여 AWS 에서 액세스를 제어합니다. 정책은 자격 증명 또는 리소스와 연결될 때 권한을 정의합니다.는 보안 주체가 요청할 때 이러한 정책을 AWS 평가합니다. 대부분의 정책은에 JSON 문서 AWS 로 저장됩니다. JSON 정책 문서에 대한 자세한 내용은 IAM 사용 설명서의 JSON 정책 개요 섹션을 참조하세요.

정책을 사용하여 관리자는 어떤 보안 주체가 어떤 리소스에 대해 어떤 조건에서 작업을 수행할 수 있는지 정의하여 누가 무엇을 액세스할 수 있는지 지정합니다.

기본적으로 사용자 및 역할에는 어떠한 권한도 없습니다. IAM 관리자는 IAM 정책을 생성하고 사용자가 수임할 수 있는 역할에 추가합니다. IAM 정책은 작업을 수행하기 위해 사용하는 방법과 관계없이 작업에 대한 권한을 정의합니다.

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 정의를 참조하세요.

ID 기반 정책은 인라인 정책(단일 ID에 직접 포함) 또는 관리형 정책(여러 ID에 연결된 독립 실행형 정책)일 수 있습니다. 관리형 정책 또는 인라인 정책을 선택하는 방법을 알아보려면 IAM 사용 설명서의 관리형 정책 및 인라인 정책 중에서 선택 섹션을 참조하세요.

리소스 기반 정책

리소스 기반 정책은 리소스에 연결하는 JSON 정책 설명서입니다. 예를 들어 IAM 역할 신뢰 정책 및 Amazon S3 버킷 정책이 있습니다. 리소스 기반 정책을 지원하는 서비스에서 서비스 관리자는 이러한 정책을 사용하여 특정 리소스에 대한 액세스를 통제할 수 있습니다. 리소스 기반 정책에서 보안 주체를 지정해야 합니다.

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

기타 정책 유형

AWS 는 보다 일반적인 정책 유형에서 부여한 최대 권한을 설정할 수 있는 추가 정책 유형을 지원합니다.

- 권한 경계 - ID 기반 정책에서 IAM 엔터티에 부여할 수 있는 최대 권한을 설정합니다. 자세한 정보는 IAM 사용 설명서의 IAM 엔터티의 권한 범위를 참조하세요.
- 서비스 제어 정책(SCP) - AWS Organizations내 조직 또는 조직 단위에 대한 최대 권한을 지정합니다. 자세한 내용은AWS Organizations 사용 설명서의 서비스 제어 정책을 참조하세요.
- 리소스 제어 정책(RCP) – 계정의 리소스에 사용할 수 있는 최대 권한을 설정합니다. 자세한 내용은 AWS Organizations 사용 설명서의 리소스 제어 정책(RCP)을 참조하세요.
- 세션 정책 – 역할 또는 페더레이션 사용자에 대해 임시 세션을 프로그래밍 방식으로 생성할 때 파라미터로 전달하는 고급 정책입니다. 자세한 내용은 IAM 사용 설명서의 세션 정책을 참조하세요.

여러 정책 유형

여러 정책 유형이 요청에 적용되는 경우, 결과 권한은 이해하기가 더 복잡합니다. 에서 여러 정책 유형이 관련될 때 요청을 허용할지 여부를 AWS 결정하는 방법을 알아보려면 IAM 사용 설명서의 정책 평가 로직을 참조하세요.

Amazon Managed Blockchain(AMB) Access Polygon이 IAM과 작동하는 방식

IAM을 사용하여 AMB Access Polygon에 대한 액세스를 관리하기 전에 AMB Access Polygon에서 사용할 수 있는 IAM 기능을 알아봅니다.

Amazon Managed Blockchain(AMB) Access Polygon과 함께 사용할 수 있는 IAM 기능

IAM 특성	AMB Access 다각형 지원
자격 증명 기반 정책	예
리소스 기반 정책	아니요
정책 작업	예
정책 리소스	아니요

IAM 특성	AMB Access 다각형 지원
정책 조건 키	아니요
ACL	아니요
ABAC(정책 내 태그)	아니요
임시 보안 인증	아니요
위탁자 권한	아니요
서비스 역할	아니요
서비스 연결 역할	아니요

AMB Access Polygon 및 기타에서 대부분의 IAM 기능을 AWS 서비스 사용하는 방법을 전체적으로 알아보려면 IAM 사용 설명서의 AWS IAM으로 작업하는 서비스를 참조하세요.

AMB Access Polygon에 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 정의를 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 참조를 참조하세요.

AMB Access Polygon에 대한 자격 증명 기반 정책 예제

AMB Access Polygon 자격 증명 기반 정책의 예를 보려면 섹션을 참조하세요Amazon Managed Blockchain(AMB) Access Polygon에 대한 자격 증명 기반 정책 예제.

AMB Access Polygon 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 정의합니다. 리소스 기반 정책에서 보안 주체를 지정해야 합니다. 보안 주체에는 계정, 사용자, 역할, 페더레이션 사용자 또는이 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 액세스를 참조하세요.

AMB Access Polygon에 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 `Action`요소는 정책에서 액세스를 허용하거나 거부하는 데 사용할 수 있는 작업을 설명합니다. 연결된 작업을 수행할 수 있는 권한을 부여하기 위한 정책에 작업을 포함하세요.

AMB Access Polygon 작업 목록을 보려면 서비스 승인 참조의 Amazon Managed Blockchain(AMB) Access Polygon에서 정의한 작업을 참조하세요.

AMB Access Polygon의 정책 작업은 작업 앞에 다음 접두사를 사용합니다.

```
managedblockchain:
```

단일 문에서 여러 작업을 지정하려면 쉼표로 구분합니다.

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

와일드카드(*)를 사용하여 여러 작업을 지정할 수 있습니다. 예를 들어, `InvokeRpcPolygon`라는 단어로 시작하는 모든 작업을 지정하려면 다음 작업을 포함합니다.

```
"Action": "managedblockchain::InvokeRpcPolygon*"
```

AMB Access Polygon 자격 증명 기반 정책의 예를 보려면 섹션을 참조하세요Amazon Managed Blockchain(AMB) Access Polygon에 대한 자격 증명 기반 정책 예제.

AMB Access Polygon에 대한 정책 리소스

정책 리소스 지원: 아니요

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": "*"
```

AMB Access Polygon 리소스 유형 및 해당 ARNs 목록을 보려면 서비스 승인 참조의 Amazon Managed Blockchain(AMB) Access Polygon에서 정의한 리소스를 참조하세요. 각 리소스의 ARN을 지정할 수 있는 작업을 알아보려면 Amazon Managed Blockchain(AMB) Access Polygon에서 정의한 작업을 참조하세요.

AMB Access Polygon 자격 증명 기반 정책의 예를 보려면 섹션을 참조하세요Amazon Managed Blockchain(AMB) Access Polygon에 대한 자격 증명 기반 정책 예제.

AMB Access Polygon에 사용되는 정책 조건 키

서비스별 정책 조건 키 지원: 아니요

관리자는 AWS JSON 정책을 사용하여 누가 무엇에 액세스할 수 있는지 지정할 수 있습니다. 즉, 어떤 보안 주체가 어떤 리소스와 어떤 조건에서 작업을 수행할 수 있는지를 지정할 수 있습니다.

`Condition` 요소는 정의된 기준에 따라 문이 실행되는 시기를 지정합니다. 같음(equals) 또는 미만(less than)과 같은 조건 연산자를 사용하여 정책의 조건을 요청의 값과 일치시키는 조건식을 생성할 수 있습니다. 모든 AWS 전역 조건 키를 보려면 IAM 사용 설명서의 AWS 전역 조건 컨텍스트 키를 참조하세요.

AMB 액세스 다각형 조건 키 목록을 보려면 서비스 승인 참조의 Amazon Managed Blockchain(AMB) 액세스 다각형에 대한 조건 키를 참조하세요. 조건 키를 사용할 수 있는 작업과 리소스를 알아보려면 Amazon Managed Blockchain(AMB) Access Polygon에서 정의한 작업을 참조하세요.

AMB Access Polygon 자격 증명 기반 정책의 예를 보려면 섹션을 참조하세요Amazon Managed Blockchain(AMB) Access Polygon에 대한 자격 증명 기반 정책 예제.

AMB Access 다각형의 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.

AMB 액세스 다각형이 있는 ABAC

ABAC 지원(정책의 태그): 아니요

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 요소에 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 정의를 참조하세요. ABAC 설정 단계가 포함된 자습서를 보려면 IAM 사용 설명서의 속성 기반 액세스 제어(ABAC) 사용을 참조하세요.

AMB Access Polygon에서 임시 자격 증명 사용

임시 자격 증명 지원: 아니요

임시 자격 증명은 AWS 리소스에 대한 단기 액세스를 제공하며 페더레이션 또는 전환 역할을 사용할 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.

AMB Access Polygon에 대한 교차 서비스 보안 주체 권한

전달 액세스 세션(FAS) 지원: 아니요

전달 액세스 세션(FAS)은를 호출하는 보안 주체의 권한을 다운스트림 서비스에 AWS 서비스 대한 요청과 AWS 서비스함께 사용합니다. FAS 요청 시 정책 세부 정보는 전달 액세스 세션을 참조하세요.

AMB Access Polygon의 서비스 역할

서비스 역할 지원: 아니요

서비스 역할은 서비스가 사용자를 대신하여 작업을 수행하는 것으로 가정하는 IAM 역할입니다. IAM 관리자는 IAM 내에서 서비스 역할을 생성, 수정 및 삭제할 수 있습니다. 자세한 내용은 IAM 사용 설명서의 AWS 서비스 AWS에 권한을 위임할 역할 생성을 참조하세요.

> **Warning**
>
> 서비스 역할에 대한 권한을 변경하면 AMB Access Polygon 기능이 중단될 수 있습니다. AMB Access Polygon이 관련 지침을 제공하는 경우에만 서비스 역할을 편집합니다.

AMB Access Polygon에 대한 서비스 연결 역할

서비스 연결 역할 지원: 아니요

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

서비스 연결 역할 생성 또는 관리에 대한 자세한 내용은 IAM으로 작업하는AWS 서비스를 참조하세요. 서비스 연결 역할 열에서 `Yes`가 포함된 서비스를 테이블에서 찾습니다. 해당 서비스에 대한 서비스 연결 역할 설명서를 보려면 예(Yes) 링크를 선택합니다.

Amazon Managed Blockchain(AMB) Access Polygon에 대한 자격 증명 기반 정책 예제

기본적으로 사용자 및 역할에는 AMB Access Polygon 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 AMB Access Polygon에서 정의한 작업 및 리소스 유형에 대한 자세한 내용은 서비스 권한 부여 참조의 Amazon Managed Blockchain(AMB) Access Polygon에 사용되는 작업, 리소스 및 조건 키를 참조하세요.

주제

- 정책 모범 사례
- AMB Access Polygon 콘솔 사용
- 사용자가 자신의 고유한 권한을 볼 수 있도록 허용
- 다각형 네트워크 액세스

정책 모범 사례

자격 증명 기반 정책에 따라 계정에서 AMB Access Polygon 리소스를 생성, 액세스 또는 삭제할 수 있는지 여부가 결정됩니다. 이 작업으로 인해 AWS 계정에 비용이 발생할 수 있습니다. ID 기반 정책을 생성하거나 편집할 때는 다음 지침과 권장 사항을 따르세요.

- AWS 관리형 정책을 시작하고 최소 권한으로 전환 - 사용자 및 워크로드에 권한 부여를 시작하려면 많은 일반적인 사용 사례에 대한 권한을 부여하는 AWS 관리형 정책을 사용합니다. 에서 사용할 수 있습니다 AWS 계정. 사용 사례에 맞는 AWS 고객 관리형 정책을 정의하여 권한을 추가로 줄이는 것이 좋습니다. 자세한 내용은 IAM 사용 설명서의 AWS 관리형 정책 또는 AWS 직무에 대한 관리형 정책을 참조하세요.
- 최소 권한 적용 – IAM 정책을 사용하여 권한을 설정하는 경우, 작업을 수행하는 데 필요한 권한만 부여합니다. 이렇게 하려면 최소 권한으로 알려진 특정 조건에서 특정 리소스에 대해 수행할 수 있는 작업을 정의합니다. IAM을 사용하여 권한을 적용하는 방법에 대한 자세한 정보는 IAM 사용 설명서에 있는 IAM의 정책 및 권한을 참조하세요.
- IAM 정책의 조건을 사용하여 액세스 추가 제한 – 정책에 조건을 추가하여 작업 및 리소스에 대한 액세스를 제한할 수 있습니다. 예를 들어, SSL을 사용하여 모든 요청을 전송해야 한다고 지정하는 정책 조건을 작성할 수 있습니다. AWS 서비스와 같은 특정를 통해 사용되는 경우 조건을 사용하여 서비스 작업에 대한 액세스 권한을 부여할 수도 있습니다 CloudFormation. 자세한 내용은 IAM 사용 설명서의 IAM JSON 정책 요소: 조건을 참조하세요.
- IAM Access Analyzer를 통해 IAM 정책을 확인하여 안전하고 기능적인 권한 보장 - IAM Access Analyzer에서는 IAM 정책 언어(JSON)와 모범 사례가 정책에서 준수되도록 새로운 및 기존 정책을 확인합니다. IAM Access Analyzer는 100개 이상의 정책 확인 항목과 실행 가능한 추천을 제공하

여 안전하고 기능적인 정책을 작성하도록 돕습니다. 자세한 내용은 IAM 사용 설명서의 IAM Access Analyzer에서 정책 검증을 참조하세요.

- 다중 인증(MFA) 필요 -에서 IAM 사용자 또는 루트 사용자가 필요한 시나리오가 있는 경우 추가 보안을 위해 MFA를 AWS 계정켭니다. API 작업을 직접적으로 호출할 때 MFA가 필요하면 정책에 MFA 조건을 추가합니다. 자세한 내용은 IAM 사용 설명서의 MFA를 통한 보안 API 액세스를 참조하세요.

IAM의 모범 사례에 대한 자세한 내용은 IAM 사용 설명서의 IAM의 보안 모범 사례를 참조하세요.

AMB Access Polygon 콘솔 사용

Amazon Managed Blockchain(AMB) Access Polygon 콘솔에 액세스하려면 최소 권한 집합이 있어야 합니다. 이러한 권한은에서 AMB Access Polygon 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.

사용자와 역할이 여전히 AMB Access Polygon 콘솔을 사용할 수 있도록 하려면 AMB Access Polygon *ConsoleAccess* 또는 *ReadOnly* AWS 관리형 정책도 엔터티에 연결합니다. 자세한 내용은 IAM 사용 설명서의 사용자에게 권한 추가를 참조하세요.

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 방식으로이 작업을 완료할 수 있는 권한이 포함됩니다.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "ViewOwnUserInfo",
            "Effect": "Allow",
            "Action": [
                "iam:GetUserPolicy",
                "iam:ListGroupsForUser",
                "iam:ListAttachedUserPolicies",
                "iam:ListUserPolicies",
                "iam:GetUser"
            ],
```

```
            "Resource": ["arn:aws:iam::*:user/${aws:username}"]
        },
        {
            "Sid": "NavigateInConsole",
            "Effect": "Allow",
            "Action": [
                "iam:GetGroupPolicy",
                "iam:GetPolicyVersion",
                "iam:GetPolicy",
                "iam:ListAttachedGroupPolicies",
                "iam:ListGroupPolicies",
                "iam:ListPolicyVersions",
                "iam:ListPolicies",
                "iam:ListUsers"
            ],
            "Resource": "*"
        }
    ]
}
```

다각형 네트워크 액세스

> **Note**
>
> Polygon의 퍼블릭 엔드포인트에 액세스`mainnet`하고 JSON-RPC 호출`mainnet`을 수행하려면 AMB Access Polygon에 대한 적절한 IAM 권한이 있는 사용자 자격 증명(`AWS_ACCESS_KEY_ID` 및 `AWS_SECRET_ACCESS_KEY`)이 필요합니다.

Example모든 Polygon 네트워크에 액세스하기 위한 IAM 정책

이 예제에서는의 IAM 사용자에게 모든 Polygon 네트워크에 대한 AWS 계정 액세스 권한을 부여합니다.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "AccessAllPolygonNetworks",
```

```
            "Effect": "Allow",
            "Action": [
                "managedblockchain:InvokeRpcPolygon*"
            ],
            "Resource": "*"
        }
    ]
}
```

Example Polygon Mainnet 네트워크에 액세스하기 위한 IAM 정책

이 예제에서는의 IAM 사용자에게 Polygon Mainnet 네트워크에 대한 AWS 계정 액세스 권한을 부여합니다.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "AccessPolygonTestnet",
            "Effect": "Allow",
            "Action": [
                "managedblockchain:InvokeRpcPolygonMainnet"
            ],
            "Resource": "*"
        }
    ]
}
```

Amazon Managed Blockchain(AMB) 액세스 다각형 자격 증명 및 액세스 문제 해결

다음 정보를 사용하여 AMB Access Polygon 및 IAM으로 작업할 때 발생할 수 있는 일반적인 문제를 진단하고 수정할 수 있습니다.

주제

AMB Access Polygon에서 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 managedblockchain::*GetWidget* 권한이 없을 때 발생합니다.

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

이 경우, managedblockchain::*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.

iam:PassRole을 수행하도록 인증되지 않음

iam:PassRole 작업을 수행할 권한이 없다는 오류가 수신되면 AMB Access Polygon에 역할을 전달할 수 있도록 정책을 업데이트해야 합니다.

일부 AWS 서비스 에서는 새 서비스 역할 또는 서비스 연결 역할을 생성하는 대신 기존 역할을 해당 서비스에 전달할 수 있습니다. 이렇게 하려면 역할을 서비스에 전달할 권한이 있어야 합니다.

다음 예제 오류는 라는 IAM 사용자가 콘솔을 사용하여 AMB Access Polygon에서 작업을 수행하려고 marymajor 할 때 발생합니다. 하지만 작업을 수행하려면 서비스 역할이 부여한 권한이 서비스에 있어야 합니다. Mary는 서비스에 역할을 전달할 권한이 없습니다.

```
User: arn:aws:iam::123456789012:user/marymajor is not authorized to perform:
 iam:PassRole
```

이 경우, Mary가 iam:PassRole작업을 수행할 수 있도록 Mary의 정책을 업데이트해야 합니다.

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

내 외부의 사람이 내 AMB Access Polygon 리소스 AWS 계정 에 액세스하도록 허용하고 싶습니다.

다른 계정의 사용자 또는 조직 외부의 사람이 리소스에 액세스할 때 사용할 수 있는 역할을 생성할 수 있습니다. 역할을 수임할 신뢰할 수 있는 사람을 지정할 수 있습니다. 리소스 기반 정책 또는 액세스 제어 목록(ACL)을 지원하는 서비스의 경우, 이러한 정책을 사용하여 다른 사람에게 리소스에 대한 액세스 권한을 부여할 수 있습니다.

자세한 내용은 다음을 참조하세요.

- AMB Access Polygon이 이러한 기능을 지원하는지 여부를 알아보려면 섹션을 참조하세요Amazon Managed Blockchain(AMB) Access Polygon이 IAM과 작동하는 방식.
- 소유 AWS 계정 한의 리소스에 대한 액세스 권한을 제공하는 방법을 알아보려면 IAM 사용 설명서의 소유한 다른의 IAM 사용자에게 액세스 권한 제공을 참조 AWS 계정 하세요.
- 타사에 리소스에 대한 액세스 권한을 제공하는 방법을 알아보려면 IAM 사용 설명서의 타사가 AWS 계정 소유한에 대한 액세스 권한 제공을 AWS 계정참조하세요.
- ID 페더레이션을 통해 액세스 권한을 제공하는 방법을 알아보려면 IAM 사용 설명서의 외부에서 인증된 사용자에게 액세스 권한 제공(ID 페더레이션)을 참조하세요.
- 크로스 계정 액세스에 대한 역할과 리소스 기반 정책 사용의 차이점을 알아보려면 IAM 사용 설명서의 IAM의 크로스 계정 리소스 액세스를 참조하세요.

를 사용하여 Amazon Managed Blockchain(AMB) 액세스 다각형 이벤트 로깅 AWS CloudTrail

> **Note**
>
> Amazon Managed Blockchain(AMB) Access Polygon은 관리 이벤트를 지원하지 않습니다.

Amazon Managed Blockchain은 관리 AWS CloudTrail형 블록체인에서 사용자, 역할 또는 서비스가 수행한 작업에 대한 레코드를 제공하는 AWS 서비스인에서 실행됩니다. CloudTrail은 Managed Blockchain에 대한 AMB Access Polygon 엔드포인트를 데이터 영역 이벤트로 호출한 사람을 캡처합니다.

원하는 데이터 영역 이벤트를 수신하도록 구독하는 적절하게 구성된 추적을 생성하는 경우 AMB Access Polygon 관련 CloudTrail 이벤트를 S3 버킷으로 지속적으로 전송할 수 있습니다. CloudTrail에서 수집한 정보를 사용하여 AMB Access Polygon 엔드포인트 중 하나에 요청이 이루어졌는지, 요청이 수행된 IP 주소, 요청을 수행한 사람, 요청이 수행된 시간 및 기타 추가 세부 정보를 확인할 수 있습니다.

CloudTrail에 대한 자세한 내용은 AWS CloudTrail 사용 설명서를 참조하세요.

CloudTrail의 AMB Access Polygon 정보

CloudTrail은 생성할 AWS 계정 때에서 활성화됩니다. 그러나 AMB Access Polygon 엔드포인트를 호출한 사용자를 볼 수 있도록 데이터 영역 이벤트를 구성해야 합니다.

AMB Access Polygon에 대한 이벤트를 AWS 계정포함하여에 이벤트를 지속적으로 기록하려면 추적을 생성합니다. 추적은 CloudTrail이 S3 버킷으로 로그 파일을 전송할 수 있도록 합니다. 콘솔에서 트레일을 생성하면 기본적으로 모든 AWS 리전에 트레일이 적용됩니다. 추적은 AWS 파티션에서 지원되는 모든 리전의 이벤트를 로깅하고 지정한 S3 버킷으로 로그 파일을 전송합니다. 또한 CloudTrail 로그에서 수집된 이벤트 데이터를 추가로 분석하고 이에 대해 조치를 취 AWS 서비스 하도록 다른를 구성할 수 있습니다. 자세한 내용은 다음 자료를 참조하세요.

- CloudTrail을 사용하여 다각형 JSON-RPCs 추적
- 추적 생성 개요
- CloudTrail 지원 서비스 및 통합

- CloudTrail에 대한 Amazon SNS 알림 구성
- 여러 리전에서 CloudTrail 로그 파일 수신 및 여러 계정에서 CloudTrail 로그 파일 수신

CloudTrail 데이터 이벤트를 분석하여 AMB Access Polygon 엔드포인트를 호출한 사용자를 모니터링할 수 있습니다.

모든 이벤트 또는 로그 항목에는 요청을 생성했던 사용자에 관한 정보가 포함됩니다. ID 정보를 이용하면 다음을 쉽게 판단할 수 있습니다.

- 요청이 루트 또는 AWS Identity and Access Management (IAM) 사용자 자격 증명으로 이루어졌는지 여부
- 역할 또는 연합된 사용자에 대한 임시 보안 자격 증명을 사용하여 요청이 생성되었는지 여부
- 요청이 다른에 의해 이루어졌는지 여부 AWS 서비스

자세한 내용은 CloudTrail userIdentity 요소를 참조하세요.

AMB Access Polygon 로그 파일 항목 이해

데이터 영역 이벤트의 경우 추적은 이벤트를 지정된 S3 버킷에 로그 파일로 전송할 수 있는 구성입니다. 각 CloudTrail 로그 파일에는 모든 소스의 단일 요청을 나타내는 하나 이상의 로그 항목이 포함되어 있습니다. 이러한 항목은 작업 날짜 및 시간, 연결된 요청 파라미터를 포함하여 요청된 작업에 대한 세부 정보를 제공합니다.

> **Note**
>
> 로그 파일의 CloudTrail 데이터 이벤트는 AMB Access Polygon API 호출의 정렬된 스택 추적이 아니므로 특정 순서로 표시되지 않습니다.

CloudTrail을 사용하여 다각형 JSON-RPCs 추적

CloudTrail을 사용하여 계정에서 AMB Access Polygon 엔드포인트를 호출한 사람과 데이터 이벤트로 호출된 JSON-RPC를 추적할 수 있습니다. 기본적으로 추적을 생성하면 데이터 이벤트가 로깅되지 않습니다. AMB Access Polygon 엔드포인트를 CloudTrail 데이터 이벤트로 호출한 사용자를 기록하려면 활동을 수집하려는 지원되는 리소스 또는 리소스 유형을 추적에 명시적으로 추가해야 합니다. AMB Access Polygon은 AWS Management Console AWS CLI, 및 SDK를 사용하여 데이터 이벤트 추가를

지원합니다. 자세한 내용은 AWS CloudTrail 사용 설명서의 고급 선택기를 사용하여 이벤트 로깅을 참조하세요.

추적에 데이터 이벤트를 로깅하려면 추적을 생성한 후 put-event-selectors 작업을 사용합니다. `--advanced-event-selectors` 옵션을 사용하여 데이터 이벤트 로깅을 시작하여 AMB Access Polygon 엔드포인트를 호출한 사용자를 결정하기 위한 `AWS::ManagedBlockchain::Network` 리소스 유형을 지정합니다.

Example계정의 모든 AMB Access Polygon 엔드포인트 요청의 데이터 이벤트 로그 항목

다음 예제에서는 `put-event-selectors` 작업을 사용하여 `us-east-1` 리전의 추적`my-polygon-trail`에 대한 계정의 모든 AMB Access Polygon 엔드포인트 요청을 로깅하는 방법을 보여줍니다.

```
aws cloudtrail put-event-selectors \

--region us-east-1 \
--trail-name my-polygon-trail  \
--advanced-event-selectors '[{
    "Name": "Test",
    "FieldSelectors": [
      { "Field": "eventCategory", "Equals": ["Data"] },
      { "Field": "resources.type", "Equals": ["AWS::ManagedBlockchain::Network"] } ]}]'
```

구독한 후 이전 예제에 지정된 추적에 연결된 S3 버킷의 사용량을 추적할 수 있습니다.

다음 결과는 CloudTrail에서 수집한 정보의 CloudTrail 데이터 이벤트 로그 항목을 보여줍니다. AMB Access Polygon 엔드포인트 중 하나에 대해 Polygon JSON-RPC 요청이 이루어졌는지, 요청이 수행된 IP 주소, 요청을 수행한 사람, 요청이 수행된 시간 및 기타 추가 세부 정보를 확인할 수 있습니다. 다음 예제의 일부 값은 보안상의 이유로 난독화되었지만 실제 로그 항목에 완전히 표시됩니다.

```
{
        "eventVersion": "1.09",
        "userIdentity": {
            "type": "AssumedRole",
            "principalId": "AROA554UO62RJ7KSB7FAX:777777777777",
            "arn": "arn:aws:sts::111122223333:assumed-role/Admin/777777777777",
            "accountId": "111122223333"
        },
        "eventTime": "2023-04-12T19:00:22Z",
        "eventSource": "managedblockchain.amazonaws.com",
        "eventName": "gettxout",
        "awsRegion": "us-east-1",
```

```
        "sourceIPAddress": "111.222.333.444",
        "userAgent": "python-requests/2.28.1",
        "errorCode": "-",
        "errorMessage": "-",
        "requestParameters": {
            "jsonrpc": "2.0",
            "method": "gettxout",
            "params": [],
            "id": 1
        },
        "responseElements": null,
        "requestID": "DRznHHEj********",
        "eventID": "baeb232d-2c6b-46cd-992c-0e40********",
        "readOnly": true,
        "resources": [{
            "type": "AWS::ManagedBlockchain::Network",
            "ARN": "arn:aws:managedblockchain:::networks/n-polygon-mainnet"
        }],
        "eventType": "AwsApiCall",
        "managementEvent": false,
        "recipientAccountId": "111122223333",
        "eventCategory": "Data"
}
```

AMB Access Polygon 사용 설명서의 문서 기록

다음 표에서는 AMB Access Polygon에 대한 설명서 릴리스를 설명합니다.

변경 사항	설명	날짜
JSON-RPC 할당량 업데이트	지원되는 각 JSON-RPC에 대해 AMB Access Polygon이 지원하는 할당량이 업데이트됩니다.	2024년 4월 12일
뭄바이 테스트넷 네트워크에 대한 지원 종료	AMB Access Polygon은 2024년 4월 15일에 뭄바이 테스트넷 지원을 종료했습니다.	2024년 4월 10일
자습서 주제 추가	AWS re:Post의 커뮤니티 문서 섹션에서 AMB Access Polygon 자습서를 참조하세요.	2024년 4월 9일
퍼블릭 미리 보기	Amazon Managed Blockchain(AMB) Access Polygon 서비스의 공개 미리 보기 릴리스입니다.	2023년 11월 24일