aws

Panduan Developerr

Poligon Akses AMB

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

Poligon Akses AMB: Panduan Developerr

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

Merek dagang dan tampilan dagang Amazon tidak boleh digunakan sehubungan dengan produk atau layanan apa pun yang bukan milik Amazon, dengan cara apa pun yang dapat menyebabkan kebingungan di antara pelanggan, atau dengan cara apa pun yang merendahkan atau mendiskreditkan Amazon. Semua merek dagang lain yang tidak dimiliki oleh Amazon merupakan hak milik masing-masing pemiliknya, yang mungkin atau mungkin tidak terafiliasi, terkait dengan, atau disponsori oleh Amazon.

Table of Contents

Amazon Managed Blockchain (AMB) Access Polygon sedang dalam rilis pratinjau dan dapat berubah sewaktu-waktu.

Terjemahan disediakan oleh mesin penerjemah. Jika konten terjemahan yang diberikan bertentangan dengan versi bahasa Inggris aslinya, utamakan versi bahasa Inggris.

Apa itu Polygon Akses Amazon Managed Blockchain (AMB)?

Amazon Managed Blockchain (AMB) Access Polygon adalah layanan terkelola penuh yang membantu Anda membangun aplikasi Web3 yang tangguh di blockchain Polygon. AMB Access Polygon menyediakan akses instan dan tanpa server ke blockchain Polygon.

Polygon adalah solusi penskalaan yang menggunakan Ethereum Virtual Machine (EVM) sebagai fondasinya. Blockchain Polygon dikenal dengan throughput transaksi yang tinggi dan biaya transaksi yang rendah. Blockchain Polygon menggunakan mekanisme proof-of-stake konsensus. Polygon umumnya digunakan dalam membangun aplikasi terdesentralisasi (dApps) yang terkait dengan, game Web3 NFTs, dan kasus penggunaan tokenisasi, antara lain.

Panduan ini mencakup cara membuat dan mengelola sumber daya blockchain Polygon menggunakan Amazon Managed Blockchain (AMB) Access Polygon.

Sumber daya untuk pengguna AMB Access Polygon pertama kali

Jika ini adalah pertama kalinya Anda menggunakan AMB Access Polygon, kami sarankan Anda mulai dengan membaca bagian berikut:

- Konsep kunci: Polygon Akses Amazon Managed Blockchain (AMB)
- Memulai dengan Amazon Managed Blockchain (AMB) Access Polygon
- API Blockchain Terkelola dan JSON- RPCs didukung dengan AMB Access Polygon

Konsep kunci: Polygon Akses Amazon Managed Blockchain (AMB)

> **Note**
>
> Panduan ini mengasumsikan bahwa Anda terbiasa dengan konsep yang penting untuk Polygon. Konsep-konsep ini termasuk staking, dApps, transaksi, dompet, kontrak pintar, Polygon (POL, sebelumnya MATIC), dan lainnya. Sebelum menggunakan Amazon Managed Blockchain (AMB) Access Polygon, sebaiknya Anda meninjau Dokumentasi Pengembangan Polygon dan wiki Polygon.

Amazon Managed Blockchain (AMB) Access Polygon memberi Anda akses tanpa server ke jaringan Polygon Mainnet dan Polygon Mainnet, tanpa mengharuskan Anda menyediakan dan mengelola infrastruktur Polygon apa pun, termasuk node. Node poligon pada jaringan secara kolektif menyimpan status blockchain Polygon, memverifikasi transaksi, dan berpartisipasi dalam konsensus untuk mengubah status blockchain. Anda dapat menggunakan layanan terkelola ini untuk mengakses jaringan Polygon dengan cepat dan sesuai permintaan, sehingga mengurangi biaya kepemilikan Anda secara keseluruhan.

Dengan AMB Access Polygon, Anda memiliki akses ke panggilan JSON Remote Procedure (JSON-RPC). Anda dapat memanggil Polygon JSON- RPCs untuk berkomunikasi dengan blockchain Polygon melalui node yang dikelola oleh Blockchain Terkelola. Anda dapat menggunakan layanan AMB Access Polygon untuk mengembangkan dan menggunakan aplikasi terdesentralisasi (dApps) yang berinteraksi dengan blockchain Polygon. Bagian integral dari DApps adalah kontrak pintar. Anda dapat membuat dan menerapkan kontrak pintar ke dalam blockchain Polygon menggunakan AMB Access Polygon. Anda juga dapat memeriksa saldo untuk dompet, detail transaksi, perkiraan biaya, dan sebagainya, dengan memanggil JSON- RPCs terhadap titik akhir AMB Access Polygon yang berjalan dengan cara terdesentralisasi di semua node yang merupakan rekan ke jaringan Polygon. Setiap peer ke jaringan Polygon dapat mengembangkan dan menerapkan kontrak pintar.

> **Important**
>
> Anda bertanggung jawab untuk membuat, memelihara, menggunakan, dan mengelola alamat Polygon Anda. Anda juga bertanggung jawab atas isi alamat Polygon Anda. AWS tidak

bertanggung jawab atas transaksi apa pun yang digunakan atau dipanggil menggunakan node Polygon di Amazon Managed Blockchain.

Pertimbangan dan batasan untuk menggunakan Amazon Managed Blockchain (AMB) Access Polygon

Saat Anda menggunakan Amazon Managed Blockchain (AMB) Access Polygon, pertimbangkan hal berikut:

- Jaringan Polygon yang didukung

  AMB Access Polygon mendukung jaringan publik berikut:

  - Mainnet —Blockchain Polygon publik dijamin dengan proof-of-stake konsensus, dan di mana token Polygon (POL) dikeluarkan dan ditransaksikan. Transaksi di Mainnet memiliki nilai aktual (yaitu, mereka mengeluarkan biaya nyata) dan dicatat pada blockchain publik.

- Jaringan tidak lagi didukung oleh Polygon

  - Seperti yang dikomunikasikan oleh Polygon Labs, jaringan Testnet Mumbai akan terbenam pada pertengahan April. Sejalan dengan berita ini, AMB Access Polygon mengakhiri dukungan dari Mumbai Testnet pada 15 April 2024. Kami merekomendasikan menggunakan Amoy Testnet untuk beban kerja pengujian Anda.
  - Jaringan pribadi tidak didukung.
  - Selain itu, AMB Access Polygon tidak menyertakan dukungan untuk jaringan Polygon ZkEVM.

- Kompatibilitas dengan pustaka pemrograman pihak ketiga yang populer

  AMB Access Polygon kompatibel dengan pustaka pemrograman populer, seperti ethers.js, memungkinkan pengembang untuk berinteraksi dengan blockchain Polygon menggunakan alat yang sudah dikenal untuk berintegrasi dengan mudah dengan implementasi yang ada atau mengembangkan aplikasi baru dengan cepat.

- Wilayah yang didukung

  Layanan ini hanya didukung di Wilayah AS Timur (Virginia N.).

- Titik akhir layanan

Berikut ini adalah endpoint layanan untuk AMB Access Polygon. Untuk terhubung dengan layanan, Anda harus menggunakan titik akhir yang mencakup salah satu Wilayah yang didukung.

  - `mainnet.polygon.managedblockchain.us-east-1.amazonaws.com`
- Staking tidak didukung

  AMB Access Polygon tidak mendukung node validator Polygon (POL) untuk. proof-of-stake
- Tanda Tangan Versi 4 penandatanganan permintaan Polygon JSON-RPC

  Saat melakukan panggilan ke Polygon JSON- RPCs di Amazon Managed Blockchain, Anda dapat melakukannya melalui koneksi HTTPS yang diautentikasi menggunakan proses penandatanganan Signature Version 4. Ini berarti bahwa hanya prinsipal IAM resmi di AWS akun yang dapat melakukan panggilan Polygon JSON-RPC. Untuk melakukan ini, AWS kredensi (ID kunci akses dan kunci akses rahasia) harus diberikan dengan panggilan.

  > ⚠ Important
  >
  > - Jangan menyematkan kredensi klien dalam aplikasi yang menghadap pengguna.
  > - Anda tidak dapat menggunakan kebijakan IAM untuk membatasi akses ke Polygon JSON- individual. RPCs
- Support untuk Akses Berbasis Token

  Anda juga dapat menggunakan token Accessor untuk melakukan panggilan JSON-RPC ke titik akhir jaringan Polygon sebagai alternatif yang nyaman untuk proses penandatanganan Signature Version 4 (SigV4). Anda harus memberikan `BILLING_TOKEN` dari salah satu token Accessor yang Anda buat dan tambahkan sebagai parameter dengan panggilan Anda.

  > ⚠ Important
  >
  > - Jika Anda memprioritaskan keamanan dan auditabilitas daripada kenyamanan, gunakan proses penandatanganan SigV4 sebagai gantinya.
  > - Anda dapat mengakses Polygon JSON- RPCs menggunakan Signature Version 4 (SiGv4) dan akses berbasis token. Namun, jika Anda memilih untuk menggunakan kedua protokol, permintaan Anda ditolak.
  > - Anda tidak boleh menyematkan token Accessor di aplikasi yang dihadapi pengguna.
- Hanya pengiriman transaksi mentah yang didukung

Gunakan `eth_sendrawtransaction` JSON-RPC untuk mengirimkan transaksi yang memperbarui status blockchain Polygon.

Menyiapkan Polygon Akses Amazon Managed Blockchain (AMB)

Sebelum Anda menggunakan Amazon Managed Blockchain (AMB) Access Polygon untuk pertama kalinya, ikuti langkah-langkah di bagian ini untuk membuat file. Akun AWS Bab berikut membahas cara mulai menggunakan AMB Access Polygon.

Prasyarat untuk menggunakan AMB Access Polygon

Sebelum Anda menggunakan AWS untuk pertama kalinya, Anda harus memiliki Akun AWS.

Mendaftar untuk AWS

Ketika Anda mendaftar AWS, Anda Akun AWS secara otomatis mendaftar untuk semua Layanan AWS, termasuk Amazon Managed Blockchain (AMB) Access Polygon. Anda hanya akan dikenakan biaya untuk layanan yang digunakan.

Jika Anda Akun AWS sudah memiliki, lanjutkan ke langkah berikutnya. Jika Anda belum memiliki Akun AWS, gunakan prosedur berikut untuk membuatnya.

Untuk membuat Akun AWS

1. Buka https://portal.aws.amazon.com/billing/pendaftaran.
2. Ikuti petunjuk online.

   Bagian dari prosedur pendaftaran melibatkan menerima panggilan telepon atau pesan teks dan memasukkan kode verifikasi pada keypad telepon.

   Saat Anda mendaftar untuk sebuah Akun AWS, sebuah Pengguna root akun AWSdibuat. Pengguna root memiliki akses ke semua Layanan AWS dan sumber daya di akun. Sebagai praktik keamanan terbaik, tetapkan akses administratif ke pengguna, dan gunakan hanya pengguna root untuk melakukan tugas yang memerlukan akses pengguna root.

Buat pengguna IAM dengan izin yang sesuai

Untuk membuat dan bekerja dengan AMB Access Polygon, Anda harus memiliki prinsipal AWS Identity and Access Management (IAM) (pengguna atau grup) dengan izin yang memungkinkan tindakan Blockchain Terkelola yang diperlukan.

Saat melakukan panggilan ke Polygon JSON- RPCs di Amazon Managed Blockchain, Anda dapat melakukannya melalui koneksi HTTPS yang diautentikasi menggunakan proses penandatanganan Signature Version 4. Ini berarti bahwa hanya prinsipal IAM resmi di AWS akun yang dapat melakukan panggilan Polygon JSON-RPC. Untuk melakukan ini, AWS kredensi (ID kunci akses dan kunci akses rahasia) harus diberikan dengan panggilan.

Anda juga dapat menggunakan token Accessor untuk melakukan panggilan JSON-RPC ke titik akhir jaringan Polygon sebagai alternatif yang nyaman untuk proses penandatanganan Signature Version 4 (SigV4). Anda harus memberikan `BILLING_TOKEN` dari salah satu token Accessor yang Anda buat dan tambahkan sebagai parameter dengan panggilan Anda. Namun, Anda masih memerlukan akses IAM untuk mendapatkan izin untuk membuat token Accessor menggunakan Konsol Manajemen AWS, AWS CLI, dan SDK.

Untuk informasi tentang cara membuat pengguna IAM, lihat Membuat pengguna IAM di akun Anda AWS. Untuk informasi selengkapnya tentang cara melampirkan kebijakan izin ke pengguna, lihat Mengubah izin untuk pengguna IAM. Untuk contoh kebijakan izin yang dapat Anda gunakan untuk memberikan izin kepada pengguna agar bekerja dengan AMB Access Polygon, lihat. Contoh kebijakan berbasis identitas untuk Amazon Managed Blockchain (AMB) Access Polygon

Instal dan konfigurasikan AWS Command Line Interface

Jika Anda belum melakukannya, instal terbaru AWS Command Line Interface (AWS CLI) untuk bekerja dengan AWS sumber daya dari terminal. Untuk informasi selengkapnya, lihat Menginstal atau memperbarui versi terbaru AWS CLI.

> **Note**
>
> Untuk akses CLI, Anda memerlukan ID kunci akses dan kunci akses rahasia. Gunakan kredensi sementara alih-alih kunci akses jangka panjang jika memungkinkan. Kredensi sementara mencakup ID kunci akses, kunci akses rahasia, dan token keamanan yang menunjukkan kapan kredensialnya kedaluwarsa. Untuk informasi selengkapnya, lihat Menggunakan kredensial sementara dengan AWS sumber daya di Panduan Pengguna IAM.

Memulai dengan Amazon Managed Blockchain (AMB) Access Polygon

Mulailah dengan Amazon Managed Blockchain (AMB) Access Polygon dengan menggunakan informasi dan prosedur di bagian ini.

Topik

Buat kebijakan IAM untuk mengakses jaringan blockchain Polygon

Untuk mengakses titik akhir publik untuk Polygon Mainnet untuk melakukan panggilan JSON-RPC, Anda harus memiliki kredensi pengguna (`AWS_ACCESS_KEY_ID`dan`AWS_SECRET_ACCESS_KEY`) yang memiliki izin IAM yang sesuai untuk Amazon Managed Blockchain (AMB) Access Polygon. Di terminal dengan AWS CLI instalasi, jalankan perintah berikut untuk membuat kebijakan IAM untuk mengakses kedua titik akhir Polygon:

```
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**
>
> Contoh sebelumnya memberi Anda akses ke semua jaringan Polygon yang tersedia. Untuk mendapatkan akses ke titik akhir tertentu, gunakan `Action` perintah berikut:
>
> - `"managedblockchain:InvokeRpcPolygonMainnet"`

Setelah Anda membuat kebijakan, lampirkan kebijakan tersebut ke peran pengguna IAM agar kebijakan tersebut diterapkan. Di bagian Konsol Manajemen AWS, navigasikan ke layanan IAM, dan lampirkan kebijakan `AmazonManagedBlockchainPolygonAccess` ke peran yang ditetapkan ke pengguna IAM Anda.

Buat permintaan panggilan prosedur jarak jauh Polygon (RPC) pada editor AMB Access RPC menggunakan Konsol Manajemen AWS

Anda dapat mengedit, mengkonfigurasi, dan mengirimkan panggilan prosedur jarak jauh (RPCs) pada Konsol Manajemen AWS menggunakan AMB Access Polygon. Dengan ini RPCs, Anda dapat membaca data dan menulis transaksi di jaringan Polygon, termasuk mengambil data dan mengirimkan transaksi ke jaringan Polygon.

Example

Contoh berikut menunjukkan cara mendapatkan informasi tentang blok terbaru dengan menggunakan `eth_getBlockByNumber` RPC. Ubah variabel yang disorot ke input Anda sendiri atau pilih salah satu metode RPC yang terdaftar dan masukkan input yang relevan yang diperlukan.

1. Buka konsol Managed Blockchain di https://console.aws.amazon.com/managedblockchain/.
2. Pilih editor RPC.
3. Di bagian Permintaan, pilih *`POLYGON_MAINNET`* sebagai ***`Blockchain Network`***.
4. Pilih *`eth_getBlockByNumber`* sebagai metode RPC.

5. Masukkan **`latest`** sebagai ***`Block number`***dan pilih *`False`* sebagai bendera Transaksi penuh.
6. Kemudian, pilih Kirim RPC.
7. Anda mendapatkan hasil `latest` blok di bagian Respons. Anda kemudian dapat menyalin transaksi mentah lengkap untuk analisis lebih lanjut atau untuk digunakan dalam logika bisnis untuk aplikasi Anda.

Untuk informasi selengkapnya, lihat yang RPCs didukung oleh AMB Access Polygon

Buat permintaan AMB Access Polygon JSON-RPC dengan menggunakan **`awscurl`** AWS CLI

Example

Menandatangani permintaan dengan kredensi pengguna IAM Anda dengan menggunakan Signature Version 4 (SigV4) untuk membuat permintaan Polygon JSON-RPC ke titik akhir AMB Access Polygon. Alat baris `awscurl`perintah dapat membantu Anda menandatangani permintaan ke AWS layanan menggunakan SiGv4. Untuk informasi lebih lanjut, lihat awscurl README.md.

Instal `awscurl` dengan menggunakan metode yang sesuai dengan sistem operasi Anda. Di macOS, HomeBrew adalah aplikasi yang direkomendasikan:

```
brew install awscurl
```

Jika Anda telah menginstal dan mengonfigurasi AWS CLI, kredensi pengguna IAM Anda dan defaultnya AWS Region diatur di lingkungan Anda dan memiliki akses ke. `awscurl` Menggunakan`awscurl`, kirimkan permintaan ke Polygon Mainnet dengan menjalankan RPC. `eth_getBlockByNumber` Panggilan ini menerima parameter string yang sesuai dengan nomor blok yang ingin Anda ambil informasinya.

Perintah berikut mengambil data blok dari Polygon Mainnet dengan menggunakan nomor blok dalam `params` array untuk memilih blok tertentu untuk mengambil header.

```
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**
>
> Anda juga dapat membuat permintaan yang sama ini menggunakan `curl` dan fitur akses berbasis token AMB Access menggunakan `Accessor` token. Untuk informasi selengkapnya, lihat Membuat dan mengelola token Accessor untuk akses berbasis token untuk membuat permintaan AMB Access Polygon.
>
> ```
> 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'
> ```

Respons dari salah satu perintah mengembalikan informasi tentang blok terbaru. Lihat contoh berikut untuk tujuan ilustrasi:

```
{"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":[]}}
```

Buat permintaan Polygon JSON-RPC di Node.js

Anda dapat memanggil Polygon JSON- RPCs dengan mengirimkan permintaan yang ditandatangani menggunakan HTTPS untuk mengakses jaringan Polygon Mainnet menggunakan modul https asli di Node.js, atau Anda dapat menggunakan pustaka pihak ketiga seperti AXIOS. Contoh Node.js berikut menunjukkan cara membuat permintaan Polygon JSON-RPC ke titik akhir AMB Access Polygon menggunakan Signature Version 4 (SigV4) dan akses berbasis token. Contoh pertama mengirimkan transaksi dari satu alamat ke alamat lain dan contoh berikut meminta rincian transaksi dan informasi saldo dari blockchain.

Example

Untuk menjalankan contoh skrip Node.js ini, terapkan prasyarat berikut:

1. Anda harus memiliki node version manager (nvm) dan Node.js diinstal pada mesin Anda. Anda dapat menemukan petunjuk instalasi untuk OS Anda di sini.
2. Gunakan `node --version` perintah dan konfirmasikan bahwa Anda menggunakan Node versi 18 atau lebih tinggi. Jika diperlukan, Anda dapat menggunakan `nvm install v18.12.0` perintah, diikuti oleh `nvm use v18.12.0` perintah, untuk menginstal versi 18, versi LTS dari Node.
3. Variabel lingkungan `AWS_ACCESS_KEY_ID` dan `AWS_SECRET_ACCESS_KEY` harus berisi kredenal yang terkait dengan akun Anda.

   Ekspor variabel ini sebagai string pada klien Anda dengan menggunakan perintah berikut. Ganti nilai dengan warna merah pada string berikut dengan nilai yang sesuai dari akun pengguna IAM Anda.

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

Setelah Anda menyelesaikan semua prasyarat, salin file berikut ke direktori di lingkungan lokal Anda dengan menggunakan editor kode pilihan Anda:

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**
>
> Kode berikut menggunakan kunci pribadi hardcode untuk menghasilkan dompet Penandatangan menggunakan `Ethers.js` demi demonstrasi saja. Jangan gunakan kode ini di lingkungan produksi, karena memiliki dana nyata dan menimbulkan risiko keamanan. Jika perlu, hubungi tim akun Anda untuk memberi saran tentang praktik terbaik dompet dan Penandatangan.

```
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");
```

Setelah file-file ini disimpan ke direktori Anda, instal dependensi yang diperlukan untuk menjalankan kode menggunakan perintah berikut:

```
npm install
```

Kirim transaksi di Node.js

Contoh sebelumnya mengirimkan token Polygon Mainnet (POL) asli dari satu alamat ke alamat lain dengan menandatangani transaksi dan menyiarkannya ke Polygon Mainnet menggunakan AMB Access Polygon. Untuk melakukan ini, gunakan `sendTx.js` skrip, yang menggunakan`Ethers.js`, perpustakaan populer untuk berinteraksi dengan blockchain yang kompatibel dengan Ethereum dan Ethereum seperti Polygon. Anda perlu mengganti tiga variabel dalam kode yang disorot dengan warna merah, termasuk token `billingToken` untuk Accessor Anda untuk akses berbasis token, kunci pribadi yang Anda gunakan untuk menandatangani transaksi, dan alamat penerima yang menerima POL.

> **Tip**
>
> Kami menyarankan Anda membuat kunci pribadi baru (dompet) untuk tujuan ini daripada menggunakan kembali dompet yang ada untuk menghilangkan risiko kehilangan dana. Anda dapat menggunakan metode kelas Wallet pustaka Ethers createRandom () untuk menghasilkan dompet untuk diuji. Selain itu, jika Anda perlu meminta POL dari Polygon Mainnet, Anda dapat menggunakan faucet POL publik untuk meminta sejumlah kecil untuk digunakan untuk pengujian.

Setelah Anda `billingToken` memiliki kunci pribadi dompet yang didanai, dan alamat penerima ditambahkan ke kode, Anda menjalankan kode berikut untuk menandatangani transaksi untuk .0001 POL untuk dikirim dari alamat Anda ke alamat lain dan menyiarkannya ke Polygon Mainnet memanggil `eth_sendRawTransaction` JSON-RPC menggunakan AMB Access Polygon.

```
node sendTx.js
```

Tanggapan yang diterima kembali menyerupai yang berikut:

```
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: []
}
```

Tanggapan tersebut merupakan tanda terima transaksi. Simpan nilai propertihash. Ini adalah pengenal untuk transaksi yang baru saja Anda kirimkan ke blockchain. Anda menggunakan properti ini dalam contoh transaksi baca untuk mendapatkan detail tambahan tentang transaksi ini dari Polygon Mainnet.

Perhatikan bahwa `blockNumber` dan `blockHash` berada `null` dalam tanggapan. Ini karena transaksi belum tercatat dalam satu blok di jaringan Polygon. Perhatikan bahwa nilai-nilai ini ditentukan nanti dan Anda mungkin melihatnya saat meminta detail transaksi di bagian berikut.

Membaca transaksi di Node.js

Pada bagian ini, Anda meminta rincian transaksi untuk transaksi yang dikirimkan sebelumnya dan mengambil saldo POL untuk alamat penerima menggunakan permintaan baca ke Polygon Mainnet menggunakan AMB Access Polygon. Dalam `readTx.js` file, ganti variabel berlabel *`your-transaction-id`* dengan yang `hash` Anda simpan dari respons dari menjalankan kode di bagian sebelumnya.

Kode ini menggunakan utilitas`dispatch-evm-rpc.js`, yang menandatangani permintaan HTTPS ke AMB Access Polygon dengan modul Signature Version 4 (SigV4) yang diperlukan dari AWS SDK dan mengirimkan permintaan menggunakan klien HTTP yang banyak digunakan, AXIOS.

Tanggapan yang diterima kembali menyerupai yang berikut:

```
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
```

Respons mewakili detail transaksi. Perhatikan bahwa `blockHash` dan `blockNumber` sekarang mungkin didefinisikan. Ini menunjukkan bahwa transaksi telah dicatat dalam satu blok. Jika nilai-nilai ini masih`null`, tunggu beberapa menit, lalu jalankan kode lagi untuk memeriksa apakah transaksi Anda telah dimasukkan dalam blok. Terakhir, representasi heksadesimal dari saldo alamat penerima (0x110d9316ec000) dikonversi menjadi desimal menggunakan *`formatEther()`* metode Ether, yang mengubah hex menjadi desimal dan menggeser tempat desimal sebesar 18 (10^18) untuk memberikan keseimbangan sebenarnya dalam POL.

> **Tip**
>
> Sementara contoh kode sebelumnya menggambarkan cara menggunakan Node.js, Ether, dan Axios untuk memanfaatkan beberapa JSON- RPCs di AMB Access Polygon yang didukung, Anda dapat memodifikasi contoh dan menulis kode lain untuk membangun aplikasi Anda di Polygon menggunakan layanan ini. Untuk daftar lengkap RPCs JSON-on AMB Access Polygon yang didukung, lihat. API Blockchain Terkelola dan JSON- RPCs didukung dengan AMB Access Polygon

Membuat dan mengelola token Accessor untuk akses berbasis token untuk membuat permintaan AMB Access Polygon

Anda juga dapat menggunakan token Accessor untuk melakukan panggilan JSON-RPC ke titik akhir jaringan Polygon sebagai alternatif yang nyaman untuk proses penandatanganan Signature Version 4 (SigV4). Anda harus memberikan `BILLING_TOKEN` dari salah satu token Accessor yang Anda buat dan tambahkan sebagai parameter dengan panggilan Anda.

> **Important**
>
> - Jika Anda memprioritaskan keamanan dan auditabilitas daripada kenyamanan, gunakan proses penandatanganan SigV4 sebagai gantinya.
> - Anda dapat mengakses Polygon JSON- RPCs menggunakan Signature Version 4 (SiGv4) dan akses berbasis token. Namun, jika Anda memilih untuk menggunakan kedua protokol, permintaan Anda ditolak.
> - Anda tidak boleh menyematkan token Accessor di aplikasi yang dihadapi pengguna.

Di konsol, halaman Token Accessors menampilkan daftar semua token Accessor yang dapat Anda gunakan untuk melakukan panggilan AMB Access Polygon JSON-RPC dari kode from Anda pada klien. Akun AWS

Untuk informasi selengkapnya tentang permintaan AMB Access Polygon JSON-RPC, lihat. API Blockchain Terkelola dan JSON- RPCs didukung dengan AMB Access Polygon

Anda dapat membuat dan mengelola token Accessor menggunakan. Konsol Manajemen AWS Anda juga dapat membuat dan mengelola token Accessor menggunakan operasi API berikut:`CreateAccessor`,, `GetAccessorListAccessors`, dan `DeleteAccessor`. A `BILLING_TOKEN` adalah properti dari Accessor. `BILLING_TOKEN`Properti ini digunakan untuk melacak Accessor Anda dan untuk penagihan permintaan AMB Access Polygon JSON-RPC yang dibuat dari Anda. Akun AWS

Semua tindakan API yang terkait dengan pembuatan dan pengelolaan token Accessor juga tersedia melalui Konsol Manajemen AWS, AWS CLI, dan SDKs.

Membuat token Accessor untuk akses berbasis token

Anda dapat membuat token Accessor dan menggunakannya untuk melakukan panggilan AMB Access Polygon API pada node AMB Access Polygon di node Anda. Akun AWS

Buat token Accessor untuk membuat permintaan AMB Access Polygon JSON-RPC menggunakan Konsol Manajemen AWS

1. Buka konsol Managed Blockchain di https://console.aws.amazon.com/managedblockchain/.
2. Pilih Pengakses Token.
3. Pilih Buat Pengakses.
4. Pilih Jaringan blockchain Polygon yang valid.
5. Opsional, tambahkan Tag untuk Accessor Anda.
6. Pilih Buat Pengakses untuk membuat token Accessor baru.

Buat token Accessor untuk membuat permintaan AMB Access Polygon JSON-RPC menggunakan AWS CLI

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

Perintah sebelumnya mengembalikan `AccessorId` bersama dengan`BillingToken`, seperti yang ditunjukkan pada contoh berikut.

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

Elemen kunci dalam respons Anda adalah`BillingToken`. Anda dapat menggunakan properti ini untuk melakukan panggilan AMB Access Polygon JSON-RPC. Beberapa nilai dalam contoh telah dikaburkan karena alasan keamanan tetapi akan muncul sepenuhnya dalam tanggapan yang sebenarnya.

> **Note**
>
> Setelah operasi dijalankan, Blockchain Terkelola menyediakan dan mengonfigurasi token untuk Anda. Lamanya proses ini tergantung pada banyak variabel.

Melihat detail token Accessor

Anda dapat melihat properti untuk setiap token Accessor yang Anda Akun AWS miliki. Misalnya, Anda dapat melihat ID Accessor atau Amazon Resource Name (ARN) dari Accessor. Anda juga dapat melihat status, jenis, tanggal pembuatan, dan`BillingToken`.

Untuk melihat informasi token Accessor menggunakan Konsol Manajemen AWS

1. Buka konsol Managed Blockchain di https://console.aws.amazon.com/managedblockchain/.
2. Di panel navigasi, pilih Token Accessors.
3. Pilih ID Pengakses token dari daftar.

Halaman detail token muncul. Dari halaman ini, Anda dapat melihat properti token.

Untuk melihat informasi token Accessor menggunakan AWS CLI

Jalankan perintah berikut untuk melihat detail token Accessor. Ganti nilai `--accessor-id` dengan ID Accessor Anda.

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

Properti kunci `BillingToken` dan lainnya dikembalikan seperti yang ditunjukkan pada contoh berikut. Beberapa nilai dalam contoh telah dikaburkan karena alasan keamanan tetapi muncul sepenuhnya dalam respons yang sebenarnya.

```
{
  "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********"
  }
}
```

Menghapus token Accessor

Saat Anda menghapus token Accessor, token berubah dari `PENDING_DELETION` status `AVAILABLE` ke status. Anda tidak dapat menggunakan token Accessor dengan `PENDING_DELETION` status.

Untuk menghapus token Accessor menggunakan Konsol Manajemen AWS

1. Buka konsol Managed Blockchain di https://console.aws.amazon.com/managedblockchain/.
2. Di panel navigasi, pilih Token Accessors.
3. Pilih token Accessor yang Anda inginkan dari daftar.
4. Pilih Hapus.
5. Konfirmasikan pilihan Anda.

Anda dikembalikan ke halaman pengakses Token dengan token Accessor yang dihapus. Halaman menampilkan `PENDING_DELETION` status.

Untuk menghapus token Accessor menggunakan AWS CLI

Contoh berikut menunjukkan cara menghapus token. Gunakan `delete-accessor` perintah untuk menghapus token. Tetapkan nilai `--accessor-id` dengan ID Accessor Anda.

Menghapus token Accessor menggunakan CLI AWS

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

Jika perintah ini berhasil berjalan, tidak ada pesan yang dikembalikan.

API Blockchain Terkelola dan JSON- RPCs didukung dengan AMB Access Polygon

Amazon Managed Blockchain menyediakan operasi API untuk membuat dan mengelola pengakses token untuk AMB Access Polygon. Untuk informasi selengkapnya, lihat Panduan Referensi API Blockchain Terkelola.

Topik berikut menyediakan daftar dan referensi Polygon JSON- RPCs yang didukung AMB Access Polygon. Setiap JSON-RPC yang didukung memiliki deskripsi singkat tentang penggunaannya. Anda menggunakan Polygon JSON- RPCs untuk menanyakan dan mendapatkan data kontrak cerdas, mendapatkan detail transaksi, mengirimkan transaksi, dan utilitas lain seperti menjalankan jejak transaksi, dan memperkirakan biaya.

AMB Access Polygon mendukung metode JSON-RPC berikut. Setiap JSON-RPC yang didukung memiliki kategori dan deskripsi singkat tentang utilitas dan kuota permintaan defaultnya. Pertimbangan unik untuk menggunakan metode JSON-RPC dengan Amazon Managed Blockchain ditunjukkan jika berlaku.

Note

- Metode apa pun yang tidak terdaftar tidak didukung.
- Saat melakukan panggilan ke Polygon JSON- RPCs di Amazon Managed Blockchain, Anda dapat melakukannya melalui koneksi HTTPS yang diautentikasi menggunakan proses penandatanganan Signature Version 4. Ini berarti bahwa hanya prinsipal IAM resmi di AWS akun yang dapat melakukan panggilan Polygon JSON-RPC. Untuk melakukan ini, AWS kredensil (ID kunci akses dan kunci akses rahasia) harus diberikan dengan panggilan.
- Anda juga dapat menggunakan akses berbasis token sebagai alternatif yang nyaman untuk proses penandatanganan Signature Version 4 (SigV4). Jika Anda memprioritaskan keamanan dan auditabilitas daripada kenyamanan, gunakan proses penandatanganan SiGv4 sebagai gantinya. Namun, jika Anda menggunakan SiGv4 dan akses berbasis token, permintaan Anda tidak akan berfungsi.
- Permintaan batch JSON-RPC tidak didukung di Amazon Managed Blockchain (AMB) Access Polygon untuk pratinjau ini.

- Kolom Kuota pada tabel berikut mencantumkan kuota untuk setiap JSON-RPC. Kuota ditetapkan dalam permintaan per detik (RPS) per Wilayah per jaringan Polygon (Mainnet) untuk setiap JSON-RPC.

  Untuk meningkatkan kuota Anda, Anda harus menghubungi Dukungan. Untuk menghubungi Dukungan, masuk ke AWS Support Center Console. Pilih Buat kasus. Pilih Teknis. Pilih Blockchain Terkelola sebagai layanan Anda. Pilih Access:Polygon sebagai Kategori Anda dan panduan Umum sebagai Keparahan Anda. Masukkan Kuota RPC sebagai Subjek dan di kotak teks Deskripsi daftar JSON-RPC dan batas kuota yang berlaku untuk kebutuhan Anda di RPS per jaringan Polygon per Wilayah. Kirimkan kasus Anda.

Kategori	JSON-RPC	Deskripsi	Pertimbangan
Ethereum	ETH_BlockNumber	Mengembalikan jumlah blok terbaru.	
	eth_call	Segera jalankan panggilan pesan baru tanpa membuat transaksi di blockchain.	`eth_call`mengkonsumsi 0 gas, tetapi memiliki parameter gas untuk pesan yang membutuhkannya.
	Eth_ChainID	Mengembalikan nilai integer untuk nilai saat ini dikonfigurasi `Chain Id` yang diperkenalkan di EIP-155. Pengembalian `None` jika `Chain Id` tidak tersedia.	

Kategori	JSON-RPC	Deskripsi	Pertimbangan
	Eth_estimategas	Memperkirakan dan mengembalikan gas yang diperlukan untuk transaksi tanpa menambahkan transaksi ke blockchain.	
	ETH_feeHistory	Mengembalikan koleksi informasi gas historis.	
	ETH_gasHarga	Mengembalikan harga saat ini per gas di Wei.	
	Eth_getBalance	Mengembalikan saldo akun untuk alamat akun yang ditentukan dan pengidentifikasi blok.	
	eth_ Hash getBlockBy	Mengembalikan informasi tentang blok yang ditentukan menggunakan hash blok.	
	Nomor eth_ getBlockBy	Mengembalikan informasi tentang blok yang ditentukan menggunakan nomor blok.	

Kategori	JSON-RPC	Deskripsi	Pertimbangan
	et_ getBlockReceipts	Mengembalikan tanda terima tentang blok yang ditentuka n menggunakan nomor blok.	
	et_ getBlockTransaction CountByHash	Mengembalikan jumlah transaksi di blok yang ditentuka n menggunakan hash blok.	
	et_ getBlockTransaction CountByNumber	Mengembalikan jumlah transaksi di blok yang ditentuka n menggunakan nomor blok.	
	Eth_getCode	Mengembal ikan kode di alamat akun yang ditentukan dan pengidentifikasi blok.	

Kategori	JSON-RPC	Deskripsi	Pertimbangan
	Eth_getLog	Mengembalikan array dari semua log untuk objek filter tertentu.	Anda dapat membuat `eth_getloqs` permintaan pada rentang blok apa pun dengan rentang blok 1K secara default saat alamat kontrak diberikan. Kontrak dengan aktivitas tinggi mungkin terbatas pada rentang blok yang lebih kecil. Jika tidak ada alamat kontrak yang diberikan, rentang blok akan menjadi 8.
	et_ getRawTransaction ByHash	Mengembalikan bentuk mentah dari transaksi yang ditentukan oleh`transacti on_hash` .	

Kategori	JSON-RPC	Deskripsi	Pertimbangan
	et_ getStorageAt	Mengembalikan nilai posisi penyimpanan yang ditentukan untuk alamat akun tertentu dan pengidentifikasi blok.	
	et_ getTransactionBy BlockHashAndIndex	Mengembalikan informasi tentang transaksi menggunakan hash blok tertentu dan posisi indeks transaksi.	
	et_ getTransactionBy BlockNumberAndIndex	Mengembalikan informasi tentang transaksi menggunakan nomor blok yang ditentukan dan posisi indeks transaksi.	
	eth_ Hash getTransactionBy	Mengembalikan informasi tentang transaksi dengan hash transaksi yang ditentukan.	

Kategori	JSON-RPC	Deskripsi	Pertimbangan
	et_ getTransactionCount	Mengembalikan jumlah transaksi yang dikirim dari alamat yang ditentukan dan pengidentifikasi blok.	
	et_ getTransactionReceipt	Mengembalikan tanda terima transaksi menggunakan hash transaksi yang ditentukan.	
	et_ getUncleBy BlockHashAndIndex	Mengembalikan informasi tentang blok paman ditentukan menggunakan hash blok dan posisi indeks paman.	
	et_ getUncleBy BlockNumberAndIndex	Mengembalikan informasi tentang blok paman yang ditentukan menggunakan nomor blok dan posisi indeks paman.	

Kategori	JSON-RPC	Deskripsi	Pertimbangan
	et_ getUncleCount ByBlockHash	Mengembalikan jumlah hitungan dalam paman yang ditentuka n menggunakan hash paman.	
	et_ getUncleCount ByBlockNumber	Mengembalikan jumlah hitungan dalam paman yang ditentuka n menggunakan nomor paman.	
	et_ maxPriorityFee PerGas	Mengembalikan biaya per gas yang merupakan perkiraan berapa banyak yang dapat Anda bayar sebagai biaya prioritas, atau “tip,” untuk mendapatk an transaksi yang termasuk dalam blok saat ini.	Umumnya Anda menggunak an nilai yang dikembalikan dari metode ini untuk mengatur transaksi berikutnya yang Anda kirimkan. `maxFeePerGas`
	ETH_ProtocolVersion	Mengembalikan versi protokol Ethereum saat ini.	

Kategori	JSON-RPC	Deskripsi	Pertimbangan
	et_ sendRawTransaction	Membuat transaksi panggilan pesan baru atau pembuatan kontrak untuk transaksi yang ditandata ngani.	Blockchain Terkelola hanya mendukung transaksi mentah. Anda harus membuat dan menandatangani transaksi sebelum mengirimnya.
Debug	debug_ Hash traceBlockBy	Mengembal ikan kemungkin an nomor hasil penelusur an dengan mengeksekusi semua transaksi di blok yang ditentuka n oleh hash blok dengan pelacak (Mode Jejak diperlukan).	
	debug_ Nomor traceBlockBy	Mengembalikan hasil penelusur an dengan mengeksekusi semua transaksi di blok yang ditentukan oleh nomor dengan pelacak (Mode Jejak diperlukan).	

Kategori	JSON-RPC	Deskripsi	Pertimbangan
	Debug_TraceCall	Mengembalikan jumlah kemungkinan hasil penelusuran dengan mengeksekusi panggilan eth dalam konteks eksekusi blok yang diberikan (Mode Pelacakan diperlukan).	
	Debug_TraceTransaction	Mengembalikan semua jejak transaksi tertentu (Trace Mode diperlukan).	
Bersih	net_version	Mengembalikan id jaringan saat ini.	
Jejak	trace_block	Mengembalikan jejak tumpukan penuh dari semua opcode yang dipanggil dari semua transaksi yang disertakan dalam blok.	

Kategori	JSON-RPC	Deskripsi	Pertimbangan
	trace_call	Mengembalikan jumlah kemungkinan hasil penelusuran dengan mengeksekusi panggilan eth dalam konteks eksekusi blok yang diberikan (Mode Pelacakan diperlukan).	
	trace_transaction	Mengembalikan semua jejak transaksi tertentu (Trace Mode diperlukan).	
Kolam Tx	txpool_content	Mengembalikan semua transaksi yang tertunda dan antri.	
	txpool_status	Memberikan hitungan semua transaksi yang saat ini tertunda inklusi di blok berikutnya, dan yang antri (dijadwalkan untuk eksekusi di masa depan saja).	
Web	Web3_ClientVersion	Mengembalikan versi klien saat ini.	

Kasus penggunaan poligon dengan Amazon Managed Blockchain (AMB) Access Polygon

Blockchain Polygon umumnya digunakan dalam membangun aplikasi terdesentralisasi (DApps) yang terkait dengan, game Web3 NFTs, dan kasus penggunaan tokenisasi, antara lain. Topik ini menyediakan daftar beberapa kasus penggunaan yang dapat Anda terapkan menggunakan Amazon Managed Blockchain (AMB) Access Polygon.

Topik

Analisis data NFT Poligon

Anda dapat mengumpulkan data tentang Polygon NFTs, termasuk informasi seperti peristiwa transfer dan metadata NFT untuk periode tertentu. Anda kemudian dapat menganalisis data ini untuk menarik wawasan seperti mana yang NFTs sedang tren atau pengguna mana yang paling sering berinteraksi dengan koleksi tertentu.

Untuk informasi selengkapnya, lihat API Blockchain Terkelola dan JSON- RPCs didukung dengan AMB Access Polygon.

Support pembelian NFT

Anda dapat menggunakan AMB Access Polygon untuk mengirimkan transaksi pembelian NFT menggunakan mint awal, allowlist, atau di pasar sekunder. Dengan menggunakan kombinasi AWS layanan lain, Anda kemudian dapat mengizinkan pembelian menggunakan kartu kredit, menerima Fiat atau cryptocurrency, dengan penyelesaian cepat untuk semua pemangku kepentingan yang terlibat.

Untuk informasi selengkapnya, lihat API Blockchain Terkelola dan JSON- RPCs didukung dengan AMB Access Polygon.

Buat dompet Polygon

Anda dapat menggunakan AMB Access Polygon untuk melayani fungsi penting dompet aset digital, seperti membaca saldo token pengguna dari kontrak pintar di blockchain atau menyiarkan transaksi yang ditandatangani ke blockchain.

Untuk informasi selengkapnya, lihat API Blockchain Terkelola dan JSON- RPCs didukung dengan AMB Access Polygon.

Dompet sebagai layanan

Anda dapat menggunakan AMB Access Polygon untuk mengembangkan operasi yang wallet-as-a-service diperlukan untuk mendukung transaksi dompet umum seperti memeriksa saldo, transfer aset, pengiriman aset, dan estimasi biaya, menggunakan Polygon JSON- yang didukung. RPCs

Untuk informasi selengkapnya, lihat API Blockchain Terkelola dan JSON- RPCs didukung dengan AMB Access Polygon.

Pengalaman berpagar token

Anda dapat menggunakan AMB Access Polygon untuk membangun pengalaman berpagar token bagi pengguna Anda. Misalnya, Anda dapat memberikan akses ke konten secara kondisional hanya kepada pemilik NFT tertentu. Untuk mencapai ini, Anda harus membaca blockchain untuk menentukan kepemilikan NFT atas alamat pengguna.

Untuk informasi selengkapnya, lihat API Blockchain Terkelola dan JSON- RPCs didukung dengan AMB Access Polygon.

Tutorial untuk Amazon Managed Blockchain (AMB) Akses Polygon

Tutorial berikut yang disorot di bagian ini adalah Artikel Komunitas AWS re:Post yang menyediakan panduan untuk membantu Anda mempelajari cara melakukan beberapa tugas umum di blockchain Polygon menggunakan AMB Access Polygon.

- Mengirim transaksi menggunakan AMB Access Polygon dan web3.js
- Menerapkan kontrak pintar menggunakan AMB Access Polygon dan Hardhat Ignition
- Berinteraksi dengan Smart Contract
- Ambil data harga saat ini di luar rantai menggunakan umpan data AMB Access Polygon dan Chainlink
- Analisis data token ERC-20 di Polygon Mainnet dengan AMB Access

Keamanan di Amazon Managed Blockchain (AMB) Access Polygon

Keamanan cloud di AWS adalah prioritas tertinggi. Sebagai AWS pelanggan, Anda mendapat manfaat dari pusat data dan arsitektur jaringan yang dibangun untuk memenuhi persyaratan organisasi yang paling sensitif terhadap keamanan.

Keamanan adalah tanggung jawab bersama antara Anda AWS dan Anda. Model tanggung jawab bersama menggambarkan ini sebagai keamanan cloud dan keamanan di cloud:

- Keamanan cloud — AWS bertanggung jawab untuk melindungi infrastruktur yang menjalankan AWS layanan di AWS Cloud. AWS juga memberi Anda layanan yang dapat Anda gunakan dengan aman. Auditor pihak ketiga secara teratur menguji dan memverifikasi keefektifan keamanan kami sebagai bagian dari program kepatuhan AWS. Untuk mempelajari tentang program kepatuhan yang berlaku untuk Amazon Managed Blockchain (AMB) Access Polygon, lihat AWS Layanan dalam Lingkup berdasarkan Program Kepatuhan.
- Keamanan di cloud — Tanggung jawab Anda ditentukan oleh AWS layanan yang Anda gunakan. Anda juga bertanggung jawab atas faktor-faktor lain, termasuk sensitivitas data Anda, persyaratan perusahaan Anda, dan hukum dan peraturan yang berlaku.

Untuk memberikan perlindungan data, otentikasi, dan kontrol akses, Amazon Managed Blockchain menggunakan AWS fitur dan fitur kerangka kerja sumber terbuka yang berjalan di Blockchain Terkelola.

Dokumentasi ini membantu Anda memahami cara menerapkan model tanggung jawab bersama saat menggunakan AMB Access Polygon. Topik berikut menunjukkan cara mengonfigurasi AMB Access Polygon untuk memenuhi tujuan keamanan dan kepatuhan Anda. Anda juga mempelajari cara menggunakan AWS layanan lain yang membantu Anda memantau dan mengamankan sumber daya AMB Access Polygon Anda.

Topik

Perlindungan data di Amazon Managed Blockchain (AMB) Access Polygon

Model tanggung jawab AWS bersama model tanggung berlaku untuk perlindungan data di Amazon Managed Blockchain (AMB) Access Polygon. Seperti yang dijelaskan dalam model AWS ini, bertanggung jawab untuk melindungi infrastruktur global yang menjalankan semua AWS Cloud. Anda bertanggung jawab untuk mempertahankan kendali atas konten yang di-host pada infrastruktur ini. Anda juga bertanggung jawab atas tugas-tugas konfigurasi dan manajemen keamanan untuk Layanan AWS yang Anda gunakan. Lihat informasi yang lebih lengkap tentang privasi data dalam Pertanyaan Umum Privasi Data. Lihat informasi tentang perlindungan data di Eropa di pos blog Model Tanggung Jawab Bersama dan GDPR AWS di Blog Keamanan AWS .

Untuk tujuan perlindungan data, kami menyarankan Anda melindungi Akun AWS kredensyal dan mengatur pengguna individu dengan AWS IAM Identity Center atau AWS Identity and Access Management (IAM). Dengan cara itu, setiap pengguna hanya diberi izin yang diperlukan untuk memenuhi tanggung jawab tugasnya. Kami juga menyarankan supaya Anda mengamankan data dengan cara-cara berikut:

- Gunakan autentikasi multi-faktor (MFA) pada setiap akun.
- Gunakan SSL/TLS untuk berkomunikasi dengan sumber daya. AWS Kami mensyaratkan TLS 1.2 dan menganjurkan TLS 1.3.
- Siapkan API dan pencatatan aktivitas pengguna dengan AWS CloudTrail. Untuk informasi tentang penggunaan CloudTrail jejak untuk menangkap AWS aktivitas, lihat Bekerja dengan CloudTrail jejak di AWS CloudTrail Panduan Pengguna.
- Gunakan solusi AWS enkripsi, bersama dengan semua kontrol keamanan default di dalamnya Layanan AWS.
- Gunakan layanan keamanan terkelola tingkat lanjut seperti Amazon Macie, yang membantu menemukan dan mengamankan data sensitif yang disimpan di Amazon S3.
- Jika Anda memerlukan modul kriptografi tervalidasi FIPS 140-3 saat mengakses AWS melalui antarmuka baris perintah atau API, gunakan titik akhir FIPS. Lihat informasi selengkapnya tentang titik akhir FIPS yang tersedia di Standar Pemrosesan Informasi Federal (FIPS) 140-3.

Kami sangat merekomendasikan agar Anda tidak pernah memasukkan informasi identifikasi yang sensitif, seperti nomor rekening pelanggan Anda, ke dalam tanda atau bidang isian bebas seperti bidang Nama. Ini termasuk saat Anda bekerja dengan AMB Access Polygon atau lainnya Layanan

AWS menggunakan konsol, API AWS CLI, atau. AWS SDKs Data apa pun yang Anda masukkan ke dalam tanda atau bidang isian bebas yang digunakan untuk nama dapat digunakan untuk log penagihan atau log diagnostik. Saat Anda memberikan URL ke server eksternal, kami sangat menganjurkan supaya Anda tidak menyertakan informasi kredensial di dalam URL untuk memvalidasi permintaan Anda ke server itu.

Enkripsi data

Enkripsi data membantu mencegah pengguna yang tidak sah membaca data dari jaringan blockchain dan sistem penyimpanan data terkait. Ini termasuk data yang mungkin dicegat saat melakukan perjalanan jaringan, yang dikenal sebagai data dalam perjalanan.

Enkripsi bergerak

Secara default, Managed Blockchain menggunakan koneksi HTTPS/TLS untuk mengenkripsi semua data yang dikirimkan dari komputer klien yang menjalankan titik akhir layanan to. AWS CLI AWS

Anda tidak perlu melakukan apapun untuk mengaktifkan penggunaan HTTPS/TLS. Itu selalu diaktifkan kecuali Anda secara eksplisit menonaktifkannya untuk AWS CLI perintah individual dengan menggunakan perintah. `--no-verify-ssl`

Manajemen identitas dan akses untuk Amazon Managed Blockchain (AMB) Access Polygon

AWS Identity and Access Management (IAM) adalah Layanan AWS yang membantu administrator mengontrol akses ke AWS sumber daya dengan aman. Administrator IAM mengontrol siapa yang dapat diautentikasi (masuk) dan diberi wewenang (memiliki izin) untuk menggunakan sumber daya AMB Access Polygon. IAM adalah Layanan AWS yang dapat Anda gunakan tanpa biaya tambahan.

Topik

Audiens

Cara Anda menggunakan AWS Identity and Access Management (IAM) berbeda berdasarkan peran Anda:

- Pengguna layanan - minta izin dari administrator Anda jika Anda tidak dapat mengakses fitur (lihat Pemecahan Masalah Amazon Managed Blockchain (AMB) Akses identitas dan akses Polygon)
- Administrator layanan - tentukan akses pengguna dan mengirimkan permintaan izin (lihat Bagaimana Amazon Managed Blockchain (AMB) Access Polygon bekerja dengan IAM)
- Administrator IAM - tulis kebijakan untuk mengelola akses (lihat Contoh kebijakan berbasis identitas untuk Amazon Managed Blockchain (AMB) Access Polygon)

Mengautentikasi dengan identitas

Otentikasi adalah cara Anda masuk AWS menggunakan kredensi identitas Anda. Anda harus diautentikasi sebagai Pengguna root akun AWS, pengguna IAM, atau dengan mengambil peran IAM.

Anda dapat masuk sebagai identitas federasi menggunakan kredensional dari sumber identitas seperti AWS IAM Identity Center (Pusat Identitas IAM), autentikasi masuk tunggal, atau kredensional. Google/Facebook Untuk informasi selengkapnya tentang cara masuk, lihat Cara masuk ke Akun AWS Anda dalam Panduan Pengguna AWS Sign-In .

Untuk akses terprogram, AWS sediakan SDK dan CLI untuk menandatangani permintaan secara kriptografis. Untuk informasi selengkapnya, lihat AWS Signature Version 4 untuk permintaan API dalam Panduan Pengguna IAM.

Akun AWS pengguna root

Saat Anda membuat Akun AWS, Anda mulai dengan satu identitas masuk yang disebut pengguna Akun AWS root yang memiliki akses lengkap ke semua Layanan AWS dan sumber daya. Kami sangat menyarankan agar Anda tidak menggunakan pengguna root untuk tugas sehari-hari. Untuk tugas yang memerlukan kredensial pengguna root, lihat Tugas yang memerlukan kredensial pengguna root dalam Panduan Pengguna IAM.

Identitas terfederasi

Sebagai praktik terbaik, mewajibkan pengguna manusia untuk menggunakan federasi dengan penyedia identitas untuk mengakses Layanan AWS menggunakan kredensi sementara.

Identitas federasi adalah pengguna dari direktori perusahaan Anda, penyedia identitas web, atau Directory Service yang mengakses Layanan AWS menggunakan kredensi dari sumber identitas. Identitas terfederasi mengambil peran yang memberikan kredensial sementara.

Untuk manajemen akses terpusat, kami menyarankan AWS IAM Identity Center. Untuk informasi selengkapnya, lihat Apa itu Pusat Identitas IAM? dalam Panduan Pengguna AWS IAM Identity Center .

Pengguna dan grup IAM

Pengguna IAM adalah identitas dengan izin khusus untuk satu orang atau aplikasi. Sebaiknya gunakan kredensial sementara alih-alih pengguna IAM dengan kredensial jangka panjang. Untuk informasi selengkapnya, lihat Mewajibkan pengguna manusia untuk menggunakan federasi dengan penyedia identitas untuk mengakses AWS menggunakan kredensi sementara di Panduan Pengguna IAM.

Grup IAM menentukan kumpulan pengguna IAM dan mempermudah pengelolaan izin untuk pengguna dalam jumlah besar. Untuk mempelajari selengkapnya, lihat Kasus penggunaan untuk pengguna IAM dalam Panduan Pengguna IAM.

Peran IAM

Peran IAM adalah identitas dengan izin khusus yang menyediakan kredensial sementara. Anda dapat mengambil peran dengan beralih dari pengguna ke peran IAM (konsol) atau dengan memanggil operasi AWS CLI atau AWS API. Untuk informasi selengkapnya, lihat Metode untuk mengambil peran dalam Panduan Pengguna IAM.

Peran IAM berguna untuk akses pengguna terfederasi, izin pengguna IAM sementara, akses lintas akun, akses lintas layanan, dan aplikasi yang berjalan di Amazon EC2. Untuk informasi selengkapnya, lihat Akses sumber daya lintas akun di IAM dalam Panduan Pengguna IAM.

Mengelola akses menggunakan kebijakan

Anda mengontrol akses AWS dengan membuat kebijakan dan melampirkannya ke AWS identitas atau sumber daya. Kebijakan menentukan izin saat dikaitkan dengan identitas atau sumber daya. AWS mengevaluasi kebijakan ini ketika kepala sekolah membuat permintaan. Sebagian besar kebijakan disimpan AWS sebagai dokumen JSON. Untuk informasi selengkapnya tentang dokumen kebijakan JSON, lihat Gambaran umum kebijakan JSON dalam Panduan Pengguna IAM.

Menggunakan kebijakan, administrator menentukan siapa yang memiliki akses ke apa dengan mendefinisikan principal mana yang dapat melakukan tindakan pada sumber daya apa, dan dalam kondisi apa.

Secara default, pengguna dan peran tidak memiliki izin. Administrator IAM membuat kebijakan IAM dan menambahkannya ke peran, yang kemudian dapat diambil oleh pengguna. Kebijakan IAM mendefinisikan izin terlepas dari metode yang Anda gunakan untuk melakukan operasinya.

Kebijakan berbasis identitas

Kebijakan berbasis identitas adalah dokumen kebijakan izin JSON yang Anda lampirkan ke identitas (pengguna, grup, atau peran). Kebijakan ini mengontrol tindakan apa yang bisa dilakukan oleh identitas tersebut, terhadap sumber daya yang mana, dan dalam kondisi apa. Untuk mempelajari cara membuat kebijakan berbasis identitas, lihat Tentukan izin IAM kustom dengan kebijakan yang dikelola pelanggan dalam Panduan Pengguna IAM.

Kebijakan berbasis identitas dapat berupa kebijakan inline (disematkan langsung ke dalam satu identitas) atau kebijakan terkelola (kebijakan mandiri yang dilampirkan pada banyak identitas). Untuk mempelajari cara memilih antara kebijakan terkelola dan kebijakan inline, lihat Pilih antara kebijakan terkelola dan kebijakan inline dalam Panduan Pengguna IAM.

Kebijakan berbasis sumber daya

Kebijakan berbasis sumber daya adalah dokumen kebijakan JSON yang Anda lampirkan ke sumber daya. Contohnya termasuk kebijakan kepercayaan peran IAM dan kebijakan bucket Amazon S3. Dalam layanan yang mendukung kebijakan berbasis sumber daya, administrator layanan dapat menggunakannya untuk mengontrol akses ke sumber daya tertentu. Anda harus menentukan principal dalam kebijakan berbasis sumber daya.

Kebijakan berbasis sumber daya merupakan kebijakan inline yang terletak di layanan tersebut. Anda tidak dapat menggunakan kebijakan AWS terkelola dari IAM dalam kebijakan berbasis sumber daya.

Jenis-jenis kebijakan lain

AWS mendukung jenis kebijakan tambahan yang dapat menetapkan izin maksimum yang diberikan oleh jenis kebijakan yang lebih umum:

- Batasan izin – Menetapkan izin maksimum yang dapat diberikan oleh kebijakan berbasis identitas kepada entitas IAM. Untuk informasi selengkapnya, lihat Batasan izin untuk entitas IAM dalam Panduan Pengguna IAM.

- Kebijakan kontrol layanan (SCPs) — Tentukan izin maksimum untuk organisasi atau unit organisasi di AWS Organizations. Untuk informasi selengkapnya, lihat Kebijakan kontrol layanan dalam Panduan Pengguna AWS Organizations .
- Kebijakan kontrol sumber daya (RCPs) — Tetapkan izin maksimum yang tersedia untuk sumber daya di akun Anda. Untuk informasi selengkapnya, lihat Kebijakan kontrol sumber daya (RCPs) di Panduan AWS Organizations Pengguna.
- Kebijakan sesi – Kebijakan lanjutan yang diteruskan sebagai parameter saat membuat sesi sementara untuk peran atau pengguna terfederasi. Untuk informasi selengkapnya, lihat Kebijakan sesi dalam Panduan Pengguna IAM.

Berbagai jenis kebijakan

Ketika beberapa jenis kebijakan berlaku pada suatu permintaan, izin yang dihasilkan lebih rumit untuk dipahami. Untuk mempelajari cara AWS menentukan apakah akan mengizinkan permintaan saat beberapa jenis kebijakan terlibat, lihat Logika evaluasi kebijakan di Panduan Pengguna IAM.

Bagaimana Amazon Managed Blockchain (AMB) Access Polygon bekerja dengan IAM

Sebelum Anda menggunakan IAM untuk mengelola akses ke AMB Access Polygon, pelajari fitur IAM apa yang tersedia untuk digunakan dengan AMB Access Polygon.

Fitur IAM yang dapat Anda gunakan dengan Amazon Managed Blockchain (AMB) Access Polygon

Fitur IAM	Dukungan AMB Access Polygon
Kebijakan berbasis identitas	Ya
Kebijakan berbasis sumber daya	Tidak
Tindakan kebijakan	Ya
Sumber daya kebijakan	Tidak
Kunci kondisi kebijakan	Tidak
ACLs	Tidak

Fitur IAM	Dukungan AMB Access Polygon
ABAC (tanda dalam kebijakan)	Tidak
Kredensial sementara	Tidak
Izin principal	Tidak
Peran layanan	Tidak
Peran terkait layanan	Tidak

Untuk mendapatkan tampilan tingkat tinggi tentang cara Layanan AWS kerja AMB Access Polygon dan lainnya dengan sebagian besar fitur IAM, lihat AWS layanan yang bekerja dengan IAM di Panduan Pengguna IAM.

Kebijakan berbasis identitas untuk AMB Access Polygon

Mendukung kebijakan berbasis identitas: Ya

Kebijakan berbasis identitas adalah dokumen kebijakan izin JSON yang dapat Anda lampirkan ke sebuah identitas, seperti pengguna IAM, grup pengguna IAM, atau peran IAM. Kebijakan ini mengontrol jenis tindakan yang dapat dilakukan oleh pengguna dan peran, di sumber daya mana, dan berdasarkan kondisi seperti apa. Untuk mempelajari cara membuat kebijakan berbasis identitas, lihat Tentukan izin IAM kustom dengan kebijakan terkelola pelanggan dalam Panduan Pengguna IAM.

Dengan kebijakan berbasis identitas IAM, Anda dapat menentukan secara spesifik apakah tindakan dan sumber daya diizinkan atau ditolak, serta kondisi yang menjadi dasar dikabulkan atau ditolaknya tindakan tersebut. Untuk mempelajari semua elemen yang dapat Anda gunakan dalam kebijakan JSON, lihat Referensi elemen kebijakan JSON IAM dalam Panduan Pengguna IAM.

Contoh kebijakan berbasis identitas untuk AMB Access Polygon

Untuk melihat contoh kebijakan berbasis identitas AMB Access Polygon, lihat. Contoh kebijakan berbasis identitas untuk Amazon Managed Blockchain (AMB) Access Polygon

Kebijakan berbasis sumber daya dalam AMB Access Polygon

Mendukung kebijakan berbasis sumber daya: Tidak

Kebijakan berbasis sumber daya adalah dokumen kebijakan JSON yang Anda lampirkan ke sumber daya. Contoh kebijakan berbasis sumber daya adalah kebijakan kepercayaan peran IAM dan kebijakan bucket Amazon S3. Dalam layanan yang mendukung kebijakan berbasis sumber daya, administrator layanan dapat menggunakannya untuk mengontrol akses ke sumber daya tertentu. Untuk sumber daya tempat kebijakan dilampirkan, kebijakan menentukan tindakan apa yang dapat dilakukan oleh principal tertentu pada sumber daya tersebut dan dalam kondisi apa. Anda harus menentukan principal dalam kebijakan berbasis sumber daya. Prinsipal dapat mencakup akun, pengguna, peran, pengguna federasi, atau. Layanan AWS

Untuk mengaktifkan akses lintas akun, Anda dapat menentukan secara spesifik seluruh akun atau entitas IAM di akun lain sebagai principal dalam kebijakan berbasis sumber daya. Untuk informasi selengkapnya, lihat Akses sumber daya lintas akun di IAM dalam Panduan Pengguna IAM.

Tindakan kebijakan untuk AMB Access Polygon

Mendukung tindakan kebijakan: Ya

Administrator dapat menggunakan kebijakan AWS JSON untuk menentukan siapa yang memiliki akses ke apa. Yaitu, di mana utama dapat melakukan tindakan pada sumber daya, dan dalam kondisi apa.

Elemen `Action` dari kebijakan JSON menjelaskan tindakan yang dapat Anda gunakan untuk mengizinkan atau menolak akses dalam sebuah kebijakan. Sertakan tindakan dalam kebijakan untuk memberikan izin untuk melakukan operasi terkait.

Untuk melihat daftar tindakan AMB Access Polygon, lihat Tindakan yang Ditentukan oleh Amazon Managed Blockchain (AMB) Access Polygon di Referensi Otorisasi Layanan.

Tindakan kebijakan di AMB Access Polygon menggunakan awalan berikut sebelum tindakan:

```
managedblockchain:
```

Untuk menetapkan secara spesifik beberapa tindakan dalam satu pernyataan, pisahkan tindakan tersebut dengan koma.

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

Anda juga dapat menentukan beberapa tindakan menggunakan wildcard (*). Sebagai contoh, untuk menentukan semua tindakan yang dimulai dengan kata `InvokeRpcPolygon`, sertakan tindakan berikut:

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

Untuk melihat contoh kebijakan berbasis identitas AMB Access Polygon, lihat. Contoh kebijakan berbasis identitas untuk Amazon Managed Blockchain (AMB) Access Polygon

Sumber daya kebijakan untuk AMB Access Polygon

Mendukung sumber daya kebijakan: Tidak

Administrator dapat menggunakan kebijakan AWS JSON untuk menentukan siapa yang memiliki akses ke apa. Yaitu, di mana utama dapat melakukan tindakan pada sumber daya, dan dalam kondisi apa.

Elemen kebijakan JSON `Resource` menentukan objek yang menjadi target penerapan tindakan. Praktik terbaiknya, tentukan sumber daya menggunakan Amazon Resource Name (ARN). Untuk tindakan yang tidak mendukung izin di tingkat sumber daya, gunakan wildcard (*) untuk menunjukkan bahwa pernyataan tersebut berlaku untuk semua sumber daya.

```
"Resource": "*"
```

Untuk melihat daftar jenis sumber daya AMB Access Polygon beserta jenisnya ARNs, lihat Sumber Daya yang Ditentukan oleh Amazon Managed Blockchain (AMB) Access Polygon dalam Referensi Otorisasi Layanan. Untuk mempelajari tindakan mana yang dapat Anda tentukan ARN dari setiap sumber daya, lihat Tindakan yang Ditentukan oleh Amazon Managed Blockchain (AMB) Access Polygon.

Untuk melihat contoh kebijakan berbasis identitas AMB Access Polygon, lihat. Contoh kebijakan berbasis identitas untuk Amazon Managed Blockchain (AMB) Access Polygon

Kunci kondisi kebijakan untuk AMB Access Polygon

Mendukung kunci kondisi kebijakan khusus layanan: Tidak

Administrator dapat menggunakan kebijakan AWS JSON untuk menentukan siapa yang memiliki akses ke apa. Yaitu, principal dapat melakukan tindakan pada suatu sumber daya, dan dalam suatu syarat.

Elemen `Condition` menentukan ketika pernyataan dieksekusi berdasarkan kriteria yang ditetapkan. Anda dapat membuat ekspresi bersyarat yang menggunakan operator kondisi, misalnya sama dengan atau kurang dari, untuk mencocokkan kondisi dalam kebijakan dengan nilai-nilai yang diminta. Untuk melihat semua kunci kondisi AWS global, lihat kunci konteks kondisi AWS global di Panduan Pengguna IAM.

Untuk melihat daftar kunci kondisi AMB Access Polygon, lihat Kunci Kondisi untuk Polygon Akses Amazon Managed Blockchain (AMB) di Referensi Otorisasi Layanan. Untuk mempelajari tindakan dan sumber daya yang dapat Anda gunakan kunci kondisi, lihat Tindakan yang Ditentukan oleh Amazon Managed Blockchain (AMB) Access Polygon.

Untuk melihat contoh kebijakan berbasis identitas AMB Access Polygon, lihat. Contoh kebijakan berbasis identitas untuk Amazon Managed Blockchain (AMB) Access Polygon

ACLs di AMB Access Polygon

Mendukung ACLs: Tidak

Access control lists (ACLs) mengontrol prinsipal mana (anggota akun, pengguna, atau peran) yang memiliki izin untuk mengakses sumber daya. ACLs mirip dengan kebijakan berbasis sumber daya, meskipun mereka tidak menggunakan format dokumen kebijakan JSON.

ABAC dengan Poligon Akses AMB

Mendukung ABAC (tag dalam kebijakan): Tidak

Kontrol akses berbasis atribut (ABAC) adalah strategi otorisasi yang menentukan izin berdasarkan atribut tanda. Anda dapat melampirkan tag ke entitas dan AWS sumber daya IAM, lalu merancang kebijakan ABAC untuk mengizinkan operasi saat tag prinsipal cocok dengan tag pada sumber daya.

Untuk mengendalikan akses berdasarkan tanda, berikan informasi tentang tanda di elemen kondisi dari kebijakan menggunakan kunci kondisi `aws:ResourceTag/`*`key-name`*, `aws:RequestTag/`*`key-name`*, atau `aws:TagKeys`.

Jika sebuah layanan mendukung ketiga kunci kondisi untuk setiap jenis sumber daya, nilainya adalah Ya untuk layanan tersebut. Jika suatu layanan mendukung ketiga kunci kondisi untuk hanya beberapa jenis sumber daya, nilainya adalah Parsial.

Untuk informasi selengkapnya tentang ABAC, lihat Tentukan izin dengan otorisasi ABAC dalam Panduan Pengguna IAM. Untuk melihat tutorial yang menguraikan langkah-langkah pengaturan ABAC, lihat Menggunakan kontrol akses berbasis atribut (ABAC) dalam Panduan Pengguna IAM.

Menggunakan kredensil sementara dengan AMB Access Polygon

Mendukung kredensi sementara: Tidak

Kredensi sementara menyediakan akses jangka pendek ke AWS sumber daya dan secara otomatis dibuat saat Anda menggunakan federasi atau beralih peran. AWS merekomendasikan agar Anda secara dinamis menghasilkan kredensi sementara alih-alih menggunakan kunci akses jangka panjang. Untuk informasi selengkapnya, lihat Kredensial keamanan sementara di IAM dan Layanan AWS yang berfungsi dengan IAM dalam Panduan Pengguna IAM.

Izin utama lintas layanan untuk AMB Access Polygon

Mendukung sesi akses maju (FAS): Tidak

Sesi akses teruskan (FAS) menggunakan izin dari pemanggilan utama Layanan AWS, dikombinasikan dengan permintaan Layanan AWS untuk membuat permintaan ke layanan hilir. Untuk detail kebijakan ketika mengajukan permintaan FAS, lihat Sesi akses terusan.

Peran layanan untuk AMB Access Polygon

Mendukung peran layanan: Tidak

Peran layanan adalah peran IAM yang diambil oleh sebuah layanan untuk melakukan tindakan atas nama Anda. Administrator IAM dapat membuat, mengubah, dan menghapus peran layanan dari dalam IAM. Untuk informasi selengkapnya, lihat Buat sebuah peran untuk mendelegasikan izin ke Layanan AWS dalam Panduan pengguna IAM.

> **Warning**
>
> Mengubah izin untuk peran layanan dapat merusak fungsionalitas AMB Access Polygon. Edit peran layanan hanya jika AMB Access Polygon memberikan panduan untuk melakukannya.

Peran terkait layanan untuk AMB Access Polygon

Mendukung peran terkait layanan: Tidak

Peran terkait layanan adalah jenis peran layanan yang ditautkan ke. Layanan AWS Layanan tersebut dapat menjalankan peran untuk melakukan tindakan atas nama Anda. Peran terkait layanan muncul di Anda Akun AWS dan dimiliki oleh layanan. Administrator IAM dapat melihat, tetapi tidak dapat mengedit izin untuk peran terkait layanan.

Untuk detail tentang pembuatan atau manajemen peran terkait layanan, lihat Layanan AWS yang berfungsi dengan IAM. Cari layanan dalam tabel yang memiliki `Yes` di kolom Peran terkait layanan. Pilih tautan Ya untuk melihat dokumentasi peran terkait layanan untuk layanan tersebut.

Contoh kebijakan berbasis identitas untuk Amazon Managed Blockchain (AMB) Access Polygon

Secara default, pengguna dan peran tidak memiliki izin untuk membuat atau memodifikasi sumber daya AMB Access Polygon. Untuk memberikan izin kepada pengguna untuk melakukan tindakan di sumber daya yang mereka perlukan, administrator IAM dapat membuat kebijakan IAM.

Untuk mempelajari cara membuat kebijakan berbasis identitas IAM dengan menggunakan contoh dokumen kebijakan JSON ini, lihat Membuat kebijakan IAM (konsol) di Panduan Pengguna IAM.

Untuk detail tentang tindakan dan jenis sumber daya yang ditentukan oleh AMB Access Polygon, termasuk format ARNs untuk setiap jenis sumber daya, lihat Tindakan, Sumber Daya, dan Kunci Kondisi untuk Polygon Akses Amazon Managed Blockchain (AMB) dalam Referensi Otorisasi Layanan.

Topik

- Praktik terbaik kebijakan
- Menggunakan konsol AMB Access Polygon
- Mengizinkan pengguna melihat izin mereka sendiri
- Mengakses jaringan Polygon

Praktik terbaik kebijakan

Kebijakan berbasis identitas menentukan apakah seseorang dapat membuat, mengakses, atau menghapus sumber daya AMB Access Polygon di akun Anda. Tindakan ini membuat Akun AWS Anda dikenai biaya. Ketika Anda membuat atau mengedit kebijakan berbasis identitas, ikuti panduan dan rekomendasi ini:

- Mulailah dengan kebijakan AWS terkelola dan beralih ke izin hak istimewa paling sedikit — Untuk mulai memberikan izin kepada pengguna dan beban kerja Anda, gunakan kebijakan AWS terkelola yang memberikan izin untuk banyak kasus penggunaan umum. Mereka tersedia di Anda Akun AWS. Kami menyarankan Anda mengurangi izin lebih lanjut dengan menentukan kebijakan yang dikelola AWS pelanggan yang khusus untuk kasus penggunaan Anda. Untuk informasi selengkapnya, lihat Kebijakan yang dikelola AWS atau Kebijakan yang dikelola AWS untuk fungsi tugas dalam Panduan Pengguna IAM.
- Menerapkan izin dengan hak akses paling rendah – Ketika Anda menetapkan izin dengan kebijakan IAM, hanya berikan izin yang diperlukan untuk melakukan tugas. Anda melakukannya dengan mendefinisikan tindakan yang dapat diambil pada sumber daya tertentu dalam kondisi tertentu, yang juga dikenal sebagai izin dengan hak akses paling rendah. Untuk informasi selengkapnya tentang cara menggunakan IAM untuk mengajukan izin, lihat Kebijakan dan izin dalam IAM dalam Panduan Pengguna IAM.
- Gunakan kondisi dalam kebijakan IAM untuk membatasi akses lebih lanjut – Anda dapat menambahkan suatu kondisi ke kebijakan Anda untuk membatasi akses ke tindakan dan sumber daya. Sebagai contoh, Anda dapat menulis kondisi kebijakan untuk menentukan bahwa semua permintaan harus dikirim menggunakan SSL. Anda juga dapat menggunakan ketentuan untuk memberikan akses ke tindakan layanan jika digunakan melalui yang spesifik Layanan AWS, seperti CloudFormation. Untuk informasi selengkapnya, lihat Elemen kebijakan JSON IAM: Kondisi dalam Panduan Pengguna IAM.
- Gunakan IAM Access Analyzer untuk memvalidasi kebijakan IAM Anda untuk memastikan izin yang aman dan fungsional – IAM Access Analyzer memvalidasi kebijakan baru dan yang sudah ada sehingga kebijakan tersebut mematuhi bahasa kebijakan IAM (JSON) dan praktik terbaik IAM. IAM Access Analyzer menyediakan lebih dari 100 pemeriksaan kebijakan dan rekomendasi yang dapat ditindaklanjuti untuk membantu Anda membuat kebijakan yang aman dan fungsional. Untuk informasi selengkapnya, lihat Validasi kebijakan dengan IAM Access Analyzer dalam Panduan Pengguna IAM.
- Memerlukan otentikasi multi-faktor (MFA) - Jika Anda memiliki skenario yang mengharuskan pengguna IAM atau pengguna root di Anda, Akun AWS aktifkan MFA untuk keamanan tambahan. Untuk meminta MFA ketika operasi API dipanggil, tambahkan kondisi MFA pada kebijakan Anda. Untuk informasi selengkapnya, lihat Amankan akses API dengan MFA dalam Panduan Pengguna IAM.

Untuk informasi selengkapnya tentang praktik terbaik dalam IAM, lihat Praktik terbaik keamanan di IAM dalam Panduan Pengguna IAM.

Menggunakan konsol AMB Access Polygon

Untuk mengakses konsol Amazon Managed Blockchain (AMB) Access Polygon, Anda harus memiliki set izin minimum. Izin ini harus memungkinkan Anda untuk membuat daftar dan melihat detail tentang sumber daya AMB Access Polygon di situs Anda. Akun AWS Jika Anda membuat kebijakan berbasis identitas yang lebih ketat daripada izin minimum yang diperlukan, konsol tidak akan berfungsi sebagaimana mestinya untuk entitas (pengguna atau peran) dengan kebijakan tersebut.

Anda tidak perlu mengizinkan izin konsol minimum untuk pengguna yang melakukan panggilan hanya ke AWS CLI atau AWS API. Sebagai gantinya, izinkan akses hanya ke tindakan yang sesuai dengan operasi API yang coba mereka lakukan.

Untuk memastikan bahwa pengguna dan peran masih dapat menggunakan konsol AMB Access Polygon, lampirkan juga Polygon Akses AMB *ConsoleAccess* atau kebijakan *ReadOnly* AWS terkelola ke entitas. Untuk informasi selengkapnya, lihat Menambah izin untuk pengguna dalam Panduan Pengguna IAM.

Mengizinkan pengguna melihat izin mereka sendiri

Contoh ini menunjukkan cara membuat kebijakan yang mengizinkan pengguna IAM melihat kebijakan inline dan terkelola yang dilampirkan ke identitas pengguna mereka. Kebijakan ini mencakup izin untuk menyelesaikan tindakan ini di konsol atau menggunakan API atau secara terprogram. AWS CLI AWS

```
{
    "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": "*"
        }
    ]
}
```

Mengakses jaringan Polygon

> **Note**
>
> Untuk mengakses titik akhir publik untuk Polygon `mainnet` dan `mainnet` melakukan panggilan JSON-RPC, Anda memerlukan kredensi pengguna (`AWS_ACCESS_KEY_ID`dan`AWS_SECRET_ACCESS_KEY`) yang memiliki izin IAM yang sesuai untuk AMB Access Polygon.

Example Kebijakan IAM untuk mengakses semua Jaringan Polygon

Contoh ini memberikan pengguna IAM dalam Akun AWS akses Anda ke semua jaringan Polygon.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "AccessAllPolygonNetworks",
            "Effect": "Allow",
            "Action": [
                "managedblockchain:InvokeRpcPolygon*"
            ],
```

```
            "Resource": "*"
        }
    ]
}
```

Example Kebijakan IAM untuk mengakses jaringan Polygon Mainnet

Contoh ini memberikan pengguna IAM dalam Akun AWS akses Anda ke jaringan Polygon Mainnet.

JSON

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

Pemecahan Masalah Amazon Managed Blockchain (AMB) Akses identitas dan akses Polygon

Gunakan informasi berikut untuk membantu Anda mendiagnosis dan memperbaiki masalah umum yang mungkin Anda temui saat bekerja dengan AMB Access Polygon dan IAM.

Topik

- Saya tidak berwenang untuk melakukan tindakan di AMB Access Polygon
- Saya tidak berwenang untuk melakukan iam: PassRole
- Saya ingin mengizinkan orang di luar saya mengakses sumber daya Akun AWS AMB Access Polygon saya

Saya tidak berwenang untuk melakukan tindakan di AMB Access Polygon

Jika Anda menerima pesan kesalahan bahwa Anda tidak memiliki otorisasi untuk melakukan tindakan, kebijakan Anda harus diperbarui agar Anda dapat melakukan tindakan tersebut.

Contoh kesalahan berikut terjadi ketika pengguna IAM `mateojackson` mencoba menggunakan konsol untuk melihat detail tentang suatu sumber daya *`my-example-widget`* rekaan, tetapi tidak memiliki izin `managedblockchain::`*`GetWidget`* rekaan.

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

Dalam hal ini, kebijakan untuk pengguna `mateojackson` harus diperbarui untuk mengizinkan akses ke sumber daya *`my-example-widget`* dengan menggunakan tindakan `managedblockchain::`*`GetWidget`*.

Jika Anda memerlukan bantuan, hubungi AWS administrator Anda. Administrator Anda adalah orang yang memberi Anda kredensial masuk.

Saya tidak berwenang untuk melakukan iam: PassRole

Jika Anda menerima kesalahan bahwa Anda tidak diizinkan untuk melakukan `iam:PassRole` tindakan, kebijakan Anda harus diperbarui agar Anda dapat meneruskan peran ke AMB Access Polygon.

Beberapa Layanan AWS memungkinkan Anda untuk meneruskan peran yang ada ke layanan tersebut alih-alih membuat peran layanan baru atau peran terkait layanan. Untuk melakukannya, Anda harus memiliki izin untuk meneruskan peran ke layanan.

Contoh kesalahan berikut terjadi ketika pengguna IAM bernama `marymajor` mencoba menggunakan konsol untuk melakukan tindakan di AMB Access Polygon. Namun, tindakan tersebut memerlukan layanan untuk mendapatkan izin yang diberikan oleh peran layanan. Mary tidak memiliki izin untuk meneruskan peran tersebut pada layanan.

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

Dalam kasus ini, kebijakan Mary harus diperbarui agar dia mendapatkan izin untuk melakukan tindakan `iam:PassRole` tersebut.

Jika Anda memerlukan bantuan, hubungi AWS administrator Anda. Administrator Anda adalah orang yang memberi Anda kredensial masuk.

Saya ingin mengizinkan orang di luar saya mengakses sumber daya Akun AWS AMB Access Polygon saya

Anda dapat membuat peran yang dapat digunakan pengguna di akun lain atau orang-orang di luar organisasi Anda untuk mengakses sumber daya Anda. Anda dapat menentukan siapa saja yang dipercaya untuk mengambil peran tersebut. Untuk layanan yang mendukung kebijakan berbasis sumber daya atau daftar kontrol akses (ACLs), Anda dapat menggunakan kebijakan tersebut untuk memberi orang akses ke sumber daya Anda.

Untuk mempelajari selengkapnya, periksa referensi berikut:

- Untuk mengetahui apakah AMB Access Polygon mendukung fitur ini, lihat. Bagaimana Amazon Managed Blockchain (AMB) Access Polygon bekerja dengan IAM
- Untuk mempelajari cara menyediakan akses ke sumber daya Anda di seluruh sumber daya Akun AWS yang Anda miliki, lihat Menyediakan akses ke pengguna IAM di pengguna lain Akun AWS yang Anda miliki di Panduan Pengguna IAM.
- Untuk mempelajari cara menyediakan akses ke sumber daya Anda kepada pihak ketiga Akun AWS, lihat Menyediakan akses yang Akun AWS dimiliki oleh pihak ketiga dalam Panduan Pengguna IAM.
- Untuk mempelajari cara memberikan akses melalui federasi identitas, lihat Menyediakan akses ke pengguna terautentikasi eksternal (federasi identitas) dalam Panduan Pengguna IAM.
- Untuk mempelajari perbedaan antara menggunakan peran dan kebijakan berbasis sumber daya untuk akses lintas akun, lihat Akses sumber daya lintas akun di IAM di Panduan Pengguna IAM.

Mencatat peristiwa Amazon Managed Blockchain (AMB) Akses Polygon dengan menggunakan AWS CloudTrail

> **Note**
>
> Amazon Managed Blockchain (AMB) Access Polygon tidak mendukung peristiwa manajemen.

Amazon Managed Blockchain berjalan pada AWS CloudTrail, layanan yang menyediakan catatan tindakan yang diambil oleh pengguna, peran, atau AWS layanan di Blockchain Terkelola. CloudTrail menangkap siapa yang memanggil titik akhir AMB Access Polygon untuk Blockchain Terkelola sebagai peristiwa pesawat data.

Jika Anda membuat jejak yang dikonfigurasi dengan benar yang berlangganan untuk menerima peristiwa bidang data yang diinginkan, Anda dapat menerima pengiriman berkelanjutan CloudTrail peristiwa terkait AMB Access Polygon ke bucket S3. Dengan menggunakan informasi yang dikumpulkan CloudTrail, Anda dapat menentukan bahwa permintaan dibuat ke salah satu titik akhir AMB Access Polygon, alamat IP tempat permintaan berasal, siapa yang mengajukan permintaan, kapan permintaan dibuat, dan detail tambahan lainnya.

Untuk mempelajari selengkapnya CloudTrail, lihat Panduan AWS CloudTrail Pengguna.

Informasi AMB Access Polygon di CloudTrail

CloudTrail diaktifkan pada Anda Akun AWS saat Anda membuatnya. Namun, Anda harus mengonfigurasi peristiwa bidang data untuk melihat siapa yang memanggil titik akhir AMB Access Polygon.

Untuk catatan acara yang sedang berlangsung di Anda Akun AWS, termasuk acara untuk AMB Access Polygon, buat jejak. Jejak memungkinkan CloudTrail untuk mengirimkan file log ke bucket S3. Secara default, saat Anda membuat jejak di konsol, jejak tersebut berlaku untuk semua Wilayah AWS. Jejak mencatat peristiwa dari semua Wilayah yang didukung di AWS partisi dan mengirimkan file log ke bucket S3 yang Anda tentukan. Selain itu, Anda dapat mengonfigurasi Layanan AWS orang lain untuk menganalisis lebih lanjut dan bertindak atas data peristiwa yang dikumpulkan dalam CloudTrail log. Untuk informasi selengkapnya, lihat berikut:

- Menggunakan CloudTrail untuk melacak Polygon JSON- RPCs
- Gambaran umum untuk membuat jejak
- CloudTrail layanan dan integrasi yang didukung
- Mengonfigurasi notifikasi Amazon SNS untuk CloudTrail
- Menerima file CloudTrail log dari beberapa Wilayah dan Menerima file CloudTrail log dari beberapa akun

Dengan menganalisis peristiwa CloudTrail data, Anda dapat memantau siapa yang memanggil titik akhir AMB Access Polygon.

Setiap entri peristiwa atau log berisi informasi tentang entitas yang membuat permintaan tersebut. Informasi identitas membantu Anda menentukan hal berikut ini:

- Apakah permintaan dibuat dengan root atau AWS Identity and Access Management (IAM) kredensional pengguna
- Apakah permintaan dibuat dengan kredensi keamanan sementara untuk peran atau pengguna federasi
- Apakah permintaan itu dibuat oleh orang lain Layanan AWS

Untuk informasi selengkapnya, lihat Elemen userIdentity CloudTrail .

Memahami entri file log AMB Access Polygon

Untuk peristiwa bidang data, jejak adalah konfigurasi yang memungkinkan pengiriman peristiwa sebagai file log ke bucket S3 tertentu. Setiap file CloudTrail log berisi satu atau lebih entri log yang mewakili satu permintaan dari sumber apa pun. Entri ini memberikan rincian tentang tindakan yang diminta, termasuk tanggal dan waktu tindakan, dan parameter permintaan terkait.

> **Note**
>
> CloudTrail peristiwa data dalam file log bukanlah jejak tumpukan terurut dari panggilan AMB Access Polygon API, sehingga tidak muncul dalam urutan tertentu.

Menggunakan CloudTrail untuk melacak Polygon JSON- RPCs

Anda dapat menggunakan CloudTrail untuk melacak siapa di akun Anda yang memanggil titik akhir AMB Access Polygon dan JSON-RPC mana yang dipanggil sebagai peristiwa data. Secara default, saat Anda membuat jejak, peristiwa data tidak dicatat. Untuk merekam siapa yang memanggil titik akhir AMB Access Polygon sebagai peristiwa CloudTrail data, Anda harus secara eksplisit menambahkan sumber daya atau jenis sumber daya yang didukung yang ingin Anda kumpulkan aktivitasnya ke jejak. AMB Access Polygon mendukung penambahan peristiwa data dengan menggunakan Konsol Manajemen AWS, AWS CLI, dan SDK. Untuk informasi selengkapnya, lihat Log peristiwa menggunakan pemilih lanjutan di Panduan AWS CloudTrail Pengguna.

Untuk mencatat peristiwa data dalam jejak, gunakan put-event-selectorsoperasi setelah Anda membuat jejak. Gunakan `--advanced-event-selectors` opsi untuk menentukan jenis `AWS::ManagedBlockchain::Network` sumber daya untuk memulai pencatatan peristiwa data untuk menentukan siapa yang memanggil titik akhir AMB Access Polygon.

Example Entri log peristiwa data dari semua permintaan titik akhir AMB Access Polygon akun Anda

Contoh berikut menunjukkan cara menggunakan `put-event-selectors` operasi untuk mencatat semua permintaan titik akhir AMB Access Polygon akun Anda untuk jejak `my-polygon-trail` di Wilayah. `us-east-1`

```
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"] } ]}]'
```

Setelah berlangganan, Anda dapat melacak penggunaan di bucket S3 yang terhubung ke jejak yang ditentukan dalam contoh sebelumnya.

Hasil berikut menunjukkan entri log peristiwa CloudTrail data dari informasi yang dikumpulkan oleh CloudTrail. Anda dapat menentukan bahwa permintaan Polygon JSON-RPC dibuat ke salah satu titik akhir AMB Access Polygon, alamat IP tempat permintaan berasal, siapa yang membuat permintaan, kapan dibuat, dan detail tambahan lainnya. Beberapa nilai dalam contoh berikut telah dikaburkan karena alasan keamanan tetapi muncul sepenuhnya dalam entri log yang sebenarnya.

```
{
        "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"
}
```

Riwayat dokumen untuk Panduan Pengguna AMB Access Polygon

Tabel berikut menjelaskan rilis dokumentasi untuk AMB Access Polygon.

Perubahan	Deskripsi	Tanggal
Kuota yang diperbarui untuk JSON-RPC	Kuota yang didukung AMB Access Polygon untuk setiap JSON-RPC yang didukung diperbarui.	April 12, 2024
Akhir dukungan untuk jaringan testnet Mumbai	AMB Access Polygon mengakhiri dukungan testnet Mumbai pada 15 April 2024.	April 10, 2024
Penambahan topik Tutorial	Tutorial AMB Access Polygon dari bagian Artikel Komunitas AWS re:Post.	April 9, 2024
Pratinjau publik	Rilis pratinjau publik dari layanan Polygon Akses Amazon Managed Blockchain (AMB).	November 24, 2023