



Panduan Developer

AMBAkses Bitcoin



AMBAkses Bitcoin: Panduan Developer

Copyright © 2024 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 tidak terafiliasi, terkait dengan, atau disponsori oleh Amazon.

Table of Contents

Apa itu Amazon Managed Blockchain (AMB) Akses Bitcoin?	1
Apakah Anda pengguna AMB Access Bitcoin pertama kali?	1
Konsep utama	3
Pertimbangan dan batasan	4
Pengaturan	6
Prasyarat dan pertimbangan	6
Daftar untuk AWS	6
Buat IAM pengguna dengan izin yang sesuai	7
Instal dan konfigurasi AWS Command Line Interface	7
Memulai	8
Buat IAM kebijakan	8
RPCContoh konsol	9
contoh awscurl RPC	10
RPCContoh Node.js	11
AMBAkses Bitcoin di atas PrivateLink	15
Kasus penggunaan Bitcoin	16
Buat dompet Bitcoin (BTC) untuk mengirim dan menerima BTC	16
Analisis aktivitas pada blockchain Bitcoin	16
Verifikasi pesan yang ditandatangani menggunakan key pair Bitcoin	17
Periksa mempool Bitcoin	17
Bitcoin JSON-RPC	19
Didukung JSON-RPC	20
Keamanan	24
Perlindungan data	25
Enkripsi data	26
Enkripsi bergerak	26
Pengelolaan identitas dan akses	26
Audiens	27
Mengautentikasi dengan identitas	27
Mengelola akses menggunakan kebijakan	31
Bagaimana Amazon Managed Blockchain (AMB) Access Bitcoin bekerja dengan IAM	34
Contoh kebijakan berbasis identitas	41
Pemecahan Masalah	45
CloudTrail log	48

AMB Akses informasi Bitcoin di CloudTrail	48
Memahami entri file log AMB Access Bitcoin	49
Menggunakan CloudTrail untuk melacak Bitcoin JSON-RPC	50
.....	lii

Apa itu Amazon Managed Blockchain (AMB) Akses Bitcoin?

Amazon Managed Blockchain (AMB) Access memberi Anda node blockchain publik untuk Ethereum dan Bitcoin, dan Anda juga dapat membuat jaringan blockchain pribadi dengan kerangka Hyperledger Fabric. Pilih dari berbagai metode untuk terlibat dengan blockchain publik, termasuk operasi API multi-tenant yang dikelola sepenuhnya, penyewa tunggal (khusus), dan tanpa server ke node blockchain publik. Untuk kasus penggunaan di mana kontrol akses penting, Anda dapat memilih dari jaringan blockchain pribadi yang dikelola sepenuhnya. Operasi API standar memberi Anda skalabilitas instan pada infrastruktur yang dikelola sepenuhnya dan tangguh, sehingga Anda dapat membangun aplikasi blockchain.

AMB Access memberi Anda dua jenis layanan infrastruktur blockchain yang berbeda: operasi API akses jaringan blockchain multi-tenant dan node dan jaringan blockchain khusus. Dengan infrastruktur blockchain khusus, Anda dapat membuat dan menggunakan node blockchain Ethereum publik dan jaringan blockchain Hyperledger Fabric pribadi untuk penggunaan Anda sendiri. Penawaran multi-penyewa berbasis API, bagaimanapun, seperti AMB Access Bitcoin, terdiri dari armada node Bitcoin di belakang lapisan API di mana infrastruktur node blockchain yang mendasarinya dibagi di antara pelanggan.

Bitcoin adalah jaringan blockchain terdesentralisasi yang memungkinkan peer-to-peer transaksi aman dengan nilai dalam mata uang kripto asli jaringan, Bitcoin (BTC). Jaringan Bitcoin digunakan oleh individu, lembaga keuangan, perusahaan fintech, pemerintah, dan banyak lagi. Jaringan Bitcoin adalah media pertukaran, komoditas untuk investasi, atau buku besar yang dapat diverifikasi secara publik dan tidak dapat diubah untuk data tertulis. Dengan Amazon Managed Blockchain (AMB) Access Bitcoin, Anda dapat mengakses kumpulan jaringan Bitcoin Mainnet dan Testnet melalui titik akhir Regional, di mana Anda dapat menulis transaksi, membaca data dari buku besar, dan memanggil permintaan JSON-RPC yang tersedia di klien node Bitcoin Core. Dengan titik akhir Bitcoin tanpa server, Anda dapat fokus membangun aplikasi Anda alih-alih berinvestasi dalam pekerjaan yang tidak terdiferensiasi seperti penyediaan, pemeliharaan, dan penyeimbangan beban node Bitcoin. Baik Anda sedang membangun dompet Bitcoin, membangun pertukaran kripto, atau menganalisis data blockchain Bitcoin, Anda hanya membayar permintaan yang Anda buat melalui titik akhir Bitcoin dengan menggunakan AMB Access Bitcoin.

Apakah Anda pengguna AMB Access Bitcoin pertama kali?

Jika Anda adalah pengguna pertama kali AMB Access Bitcoin, kami sarankan Anda mulai dengan membaca bagian berikut:

- [Konsep kunci: Amazon Managed Blockchain \(AMB\) Mengakses Bitcoin](#)
- [Memulai dengan Amazon Managed Blockchain \(AMB\) Akses Bitcoin](#)
- [Kasus penggunaan Bitcoin dengan Amazon Managed Blockchain \(AMB\) Akses Bitcoin](#)
- [Bitcoin JSON-RPC yang Didukung dengan Amazon Managed Blockchain \(AMB\) Akses Bitcoin](#)

Konsep kunci: Amazon Managed Blockchain (AMB) Mengakses Bitcoin

Note

Panduan ini mengasumsikan bahwa Anda terbiasa dengan konsep-konsep yang penting untuk Bitcoin. Konsep-konsep ini termasuk desentralisasi, node, transaksi, dompet proof-of-work, kunci publik dan pribadi, halvings, dan lain-lain. Sebelum menggunakan Amazon Managed Blockchain (AMB) Akses Bitcoin, kami sarankan Anda meninjau [Dokumentasi Pengembangan Bitcoin](#) dan [Menguasai Bitcoin](#).

Amazon Managed Blockchain (AMB) Access Bitcoin memberi Anda akses tanpa server ke blockchain Bitcoin, tanpa mengharuskan Anda menyediakan dan mengelola infrastruktur Bitcoin apa pun, termasuk node. Anda dapat menggunakan layanan terkelola ini untuk mengakses jaringan Bitcoin dengan cepat dan sesuai permintaan, mengurangi biaya kepemilikan Anda secara keseluruhan.

AMB Access Bitcoin memberi Anda akses ke jaringan Bitcoin melalui node penuh yang menjalankan klien Bitcoin Core, dengan fungsionalitas dompet dinonaktifkan, dan mendukung beberapa panggilan JSON Remote Procedure (JSON-RPC). Anda dapat memanggil Bitcoin JSON RPC untuk berkomunikasi dengan node Bitcoin yang dikelola oleh Managed Blockchain untuk berinteraksi dengan jaringan Bitcoin. Dengan Bitcoin JSON-RPC, Anda dapat membaca data dan menulis transaksi, termasuk menanyakan data dan mengirimkan transaksi ke jaringan Bitcoin dengan menggunakan layanan Amazon Managed Blockchain.

Important


Anda bertanggung jawab untuk membuat, memelihara, menggunakan, dan mengelola alamat Bitcoin Anda. Anda juga bertanggung jawab atas isi alamat Bitcoin Anda. AWS tidak bertanggung jawab atas transaksi apa pun yang digunakan atau dipanggil menggunakan node Bitcoin di Amazon Managed Blockchain.

Pertimbangan dan batasan untuk menggunakan Amazon Managed Blockchain (AMB) Akses Bitcoin

- Jaringan Bitcoin yang didukung

AMB Access Bitcoin mendukung jaringan publik berikut:

- Mainnet — Blockchain Bitcoin publik dijamin dengan proof-of-work konsensus, dan di mana cryptocurrency Bitcoin (BTC) dikeluarkan dan ditransaksikan. Transaksi di Mainnet memiliki nilai aktual (yaitu, mereka mengeluarkan biaya riil) dan dicatat pada blockchain publik.
- Testnet — Testnet adalah blockchain Bitcoin alternatif yang digunakan untuk pengujian. Koin Testnet terpisah dan berbeda dari Bitcoin aktual (BTC) dan biasanya tidak memiliki nilai apa pun.

 Note

Jaringan pribadi tidak didukung.

- Wilayah yang didukung

Berikut ini adalah Wilayah yang didukung untuk layanan ini:

Nama Wilayah	Kode	Wilayah
AS Timur (Virginia Utara)	IAD	us-east-1
Asia Pasifik (Tokyo)	NRT	ap-northeast-1
Asia Pasifik (Seoul)	ICN	ap-northeast-2
Asia Pasifik (Singapura)	DOSA	ap-southeast-1
Eropa (Irlandia)	SULIH SUARA	eu-west-1
Eropa (London)	LHR	eu-west-2

- Titik akhir layanan

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

- `mainnet.bitcoin.managedblockchain.Region.amazonaws.com`

- `testnet.bitcoin.managedblockchain.Region.amazonaws.com`

Sebagai contoh: `mainnet.bitcoin.managedblockchain.eu-west-2.amazonaws.com`

- Penambangan tidak didukung

AMB Access Bitcoin tidak mendukung penambangan Bitcoin (BTC).

- Tanda Tangan Versi 4 penandatanganan panggilan Bitcoin JSON-RPC

Saat melakukan panggilan ke Bitcoin JSON-RPC 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 Bitcoin JSON-RPC. Untuk melakukan ini, AWS kredensi (ID kunci akses dan kunci akses rahasia) harus diberikan dengan panggilan.

Important

- Jangan menyimpan kredensi klien dalam aplikasi yang menghadap pengguna.
- Anda tidak dapat menggunakan kebijakan IAM untuk membatasi akses ke masing-masing Bitcoin JSON-RPC.

- Hanya pengiriman transaksi mentah yang didukung

Gunakan `sendrawtransaction` JSON-RPC untuk mengirimkan transaksi yang memperbarui status blockchain Bitcoin.

- AWS CloudTrail dukungan logging

Anda dapat mengonfigurasi CloudTrail untuk mencatat Bitcoin JSON-RPC Anda. Untuk informasi selengkapnya, lihat [Logging Amazon Managed Blockchain \(AMB\) Mengakses peristiwa Bitcoin dengan menggunakan AWS CloudTrail](#)

Menyiapkan Amazon Managed Blockchain (AMB) Akses Bitcoin

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

Prasyarat dan pertimbangan

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

Daftar untuk AWS

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

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

Untuk membuat AWS akun

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

Bagian dari prosedur pendaftaran melibatkan tindakan menerima panggilan telepon dan memasukkan kode verifikasi di keypad telepon.

Ketika Anda mendaftar untuk Akun AWS, sebuah Pengguna root akun AWS diciptakan. 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 IAM pengguna dengan izin yang sesuai

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

Hanya IAM kepala sekolah yang dapat melakukan panggilan BitcoinJSON. RPC Saat melakukan panggilan ke Bitcoin JSON - RPCs di Amazon Managed Blockchain, Anda dapat melakukannya melalui HTTPS koneksi yang diautentikasi menggunakan [proses penandatanganan Signature Version 4](#). Ini berarti bahwa hanya IAM kepala sekolah yang berwenang di AWS akun dapat membuat Bitcoin JSON - RPC panggilan. Untuk melakukan ini, AWS kredensi (ID kunci akses dan kunci akses rahasia) harus disediakan dengan panggilan.

Untuk informasi tentang cara membuat IAM pengguna, lihat [Membuat IAM pengguna di AWS akun](#). 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 untuk bekerja dengan AMB Access Bitcoin, lihat [Contoh kebijakan berbasis identitas untuk Amazon Managed Blockchain \(\) Akses Bitcoin AMB](#).

Instal dan konfigurasi AWS Command Line Interface

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

Note

Untuk CLI akses, 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 kredensi sementara dengan AWS sumber daya](#) dalam Panduan IAM Pengguna.

Memulai dengan Amazon Managed Blockchain (AMB) Akses Bitcoin

Gunakan step-by-step tutorial di bagian ini untuk mempelajari cara melakukan tugas dengan menggunakan Amazon Managed Blockchain (AMB) Akses Bitcoin. Contoh-contoh ini mengharuskan Anda untuk menyelesaikan beberapa prasyarat. Jika Anda baru mengenal AMB Akses Bitcoin, tinjau bagian Pengaturan dari panduan ini untuk memastikan Anda telah menyelesaikan prasyarat tersebut. Untuk informasi selengkapnya, lihat [Menyiapkan Amazon Managed Blockchain \(AMB\) Akses Bitcoin](#).

Topik

- [Buat IAM kebijakan untuk mengakses Bitcoin JSON - RPCs](#)
- [Buat permintaan panggilan \(RPC\) prosedur jarak jauh Bitcoin pada RPC editor AMB Access menggunakan AWS Management Console](#)
- [Buat AMB Akses Bitcoin JSON - RPC permintaan di awscurl dengan menggunakan AWS CLI](#)
- [Buat Bitcoin JSON - RPC permintaan di Node.js](#)
- [Gunakan AMB Akses Bitcoin di atas AWS PrivateLink](#)

Buat IAM kebijakan untuk mengakses Bitcoin JSON - RPCs

Untuk mengakses titik akhir publik untuk Mainnet Bitcoin dan Testnet untuk melakukan JSON RPC panggilan, Anda harus memiliki kredensial pengguna (AWS_ACCESS_KEY_ID dan AWS_SECRET_ _) yang memiliki izin yang sesuai untuk IAM Amazon Managed Blockchain (KEY) Akses Bitcoin. ACCESS AMB Di terminal dengan AWS CLI diinstal, jalankan perintah berikut untuk membuat IAM Kebijakan untuk mengakses kedua titik akhir Bitcoin:

```
cat <<EOT > ~/amb-btc-access-policy.json
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid" : "AMBBitcoinAccessPolicy",
      "Effect": "Allow",
      "Action": [
        "managedblockchain:InvokeRpcBitcoin*"
      ],
      "Resource": "*"
    }
  ]
}
```

```
    }  
  ]  
}  
EOT  
aws iam create-policy --policy-name AmazonManagedBlockchainBitcoinAccess --policy-  
document file://$HOME/amb-btc-access-policy.json
```

Note

Contoh sebelumnya memberi Anda akses ke Bitcoin Mainnet dan Testnet. Untuk mendapatkan akses ke titik akhir tertentu, gunakan Action perintah berikut:

- "managedblockchain:InvokeRpcBitcoinMainnet"
- "managedblockchain:InvokeRpcBitcoinTestnet"

Setelah membuat kebijakan, lampirkan kebijakan tersebut ke Peran IAM pengguna agar kebijakan tersebut diterapkan. Dalam AWS Management Console, navigasikan ke IAM layanan, dan lampirkan kebijakan AmazonManagedBlockchainBitcoinAccess ke Peran yang ditetapkan untuk IAM pengguna Anda. Untuk informasi selengkapnya, lihat [Membuat Peran dan menetapkan ke IAM pengguna](#).

Buat permintaan panggilan (RPC) prosedur jarak jauh Bitcoin pada RPC editor AMB Access menggunakan AWS Management Console

Anda dapat mengedit dan mengirimkan panggilan prosedur jarak jauh (RPCs) pada AWS Management Console menggunakan AMB Access. Dengan ini RPCs, Anda dapat membaca data, menulis, dan mengirimkan transaksi di jaringan Bitcoin.

Example

Contoh berikut menunjukkan bagaimana untuk mendapatkan informasi tentang 00000000c937983704a73af28acdec37b049d214adbda81d7e2a3dd146f6ed09 b1ockhash dengan menggunakan. getBlock RPC Ganti variabel yang disorot dengan input Anda sendiri atau pilih salah satu RPC metode lain yang terdaftar dan masukkan input yang relevan yang diperlukan.

1. Buka konsol Managed Blockchain di <https://console.aws.amazon.com/managedblockchain/>.
2. Pilih RPCeditor.
3. Di bagian Permintaan, pilih **BITCOIN_MAINNET** sebagai Jaringan Blockchain.
4. Pilih **getblock** sebagai RPCmetode.
5. Masukkan **00000000c937983704a73af28acdec37b049d214adbda81d7e2a3dd146f6ed09** sebagai nomor Blokir dan pilih **0** sebagai verbositas.
6. Lalu, pilih KirimRPC.
7. Anda akan mendapatkan hasil di bagian Respons di halaman ini. 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 Bitcoin](#)

Buat AMB Akses Bitcoin JSON - RPC permintaan di awscurl dengan menggunakan AWS CLI

Example

Menandatangani permintaan dengan kredensial IAM pengguna Anda dengan menggunakan [Signature Version 4 \(SigV4\)](#) untuk melakukan RPC panggilan Bitcoin ke titik JSON akhir Access Bitcoin. AMB Alat baris perintah [awscurl](#) dapat membantu Anda menandatangani permintaan AWS layanan menggunakan SiGv4. Untuk informasi lebih lanjut, lihat [READMEawscurl](#) .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 mengkonfigurasi AWS CLI, kredensial IAM pengguna Anda dan AWS Wilayah default disetel di lingkungan Anda dan memiliki akses ke awscurl. Menggunakan awscurl, kirimkan permintaan ke Bitcoin Mainnet dan Testnet dengan memanggil. getblock RPC Panggilan ini menerima parameter string yang sesuai dengan hash blok yang ingin Anda ambil informasinya.

Perintah berikut mengambil data header blok dari Bitcoin Mainnet dengan menggunakan hash blok dalam params array untuk memilih blok tertentu untuk mengambil header. Contoh ini menggunakan us-east-1 endpoint. Anda dapat mengganti ini dengan Bitcoin pilihan Anda JSON - RPC dan AWS Wilayah yang didukung oleh Amazon Managed Blockchain (AMB) Akses Bitcoin. Selanjutnya, Anda dapat membuat permintaan terhadap jaringan Testnet, bukan Mainnet, dengan mengganti mainnet dengan testnet perintah.

```
aws curl -X POST -d '{ "jsonrpc": "1.0", "id": "getblockheader-curltest", "method":  
  "getblockheader", "params":  
  ["0000000000000000000000000000105bebab2f9dd16234a30950d38ec6ddc24d466e750a0"] }' --service  
managedblockchain https://mainnet.bitcoin.managedblockchain.us-east-1.amazonaws.com  
--region us-east-1 -k
```

Hasilnya mencakup rincian dari header blok dan daftar hash transaksi yang termasuk dalam blok yang diminta. Lihat contoh berikut ini:

```
{"result":{"hash":"0000000000000000000000000000105bebab2f9dd16234a30950d38ec6ddc24d466e750a0",  
  "confirmations":2,"height":799243,"version":664485888,"versionHex":"279b4000",  
  "merkleroot":"568e79752e1921ecf40c961435abb41bc5700fe2833ecadc4abfc2f615ddc1b8",  
  "time":1689684290,"mediantime":1689681317,"nonce":2091174943,"bits":"17053894",  
  "difficulty":53911173001054.59,  
  "chainwork":"0000000000000000000000000000000000000000000000000000004f375cf72ff64e2404c1589c",  
  "nTx":2135,  
  
  "previousblockhash":"00000000000000000000002ffe4efe07ae74ec8b92c7696f5e12b5da506f015ba6b",  
  
  "nextblockhash":"00000000000000000000000038f05ddcf3f483fdb74f4be606c022bcb673424fa4ca"},  
  "error":null,"id":"curltest"}
```

Buat Bitcoin JSON - RPC permintaan di Node.js

Anda dapat mengirimkan permintaan yang ditandatangani dengan menggunakan HTTPS untuk mengakses titik akhir Bitcoin Mainnet dan Testnet dan melakukan JSON RPC API panggilan dengan menggunakan [modul https asli di Node.js](#), atau Anda dapat menggunakan pustaka pihak ketiga seperti [AXIOS](#) Contoh berikut menunjukkan kepada Anda cara membuat RPC permintaan Bitcoin JSON ke titik akhir AMB Access Bitcoin.

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 14 atau lebih tinggi. Jika diperlukan, Anda dapat menggunakan `nvm install 14` perintah, diikuti oleh `nvm use 14` perintah, untuk menginstal versi 14.
3. Variabel lingkungan `AWS_ACCESS_KEY_ID` dan `AWS_SECRET_ACCESS_KEY` harus berisi kredensial yang terkait dengan akun Anda. Variabel lingkungan `AMB_HTTP_ENDPOINT` harus berisi titik akhir AMB Access Bitcoin Anda.

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

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

Setelah Anda menyelesaikan semua prasyarat, salin `package.json` file dan `index.js` skrip berikut ke lingkungan lokal Anda dengan menggunakan editor Anda:

`package.json`

```
{  
  "name": "bitcoin-rpc",  
  "version": "1.0.0",  
  "description": "",  
  "main": "index.js",  
  "scripts": {  
    "test": "echo \"Error: no test specified\" && exit 1"  
  },  
  "author": "",  
  "license": "ISC",  
  "dependencies": {  
    "@aws-crypto/sha256-js": "^4.0.0",  
    "@aws-sdk/credential-provider-node": "^3.360.0",  
    "@aws-sdk/protocol-http": "^3.357.0",  
    "@aws-sdk/signature-v4": "^3.357.0",  
    "axios": "^1.4.0"  
  }  
}
```


index.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 () => {

  // create a remote procedure call (RPC) request object definig the method, input
  params
  let rpc = {
    jsonrpc: "1.0",
    id: "1001",
    method: 'getblock',
    params: ["00000000c937983704a73af28acdec37b049d214adbda81d7e2a3dd146f6ed09"]
  }

  //bitcoin endpoint
  let bitcoinURL = 'https://mainnet.bitcoin.managedblockchain.us-
east-1.amazonaws.com/';

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

  // 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: bitcoinURL, data: req.body})

  console.log(response.data)
} catch (error) {
  console.error('Something went wrong: ', error)
  throw error
}

}

rpcRequest();

```

Kode sampel sebelumnya menggunakan Axios untuk membuat RPC permintaan ke titik akhir Bitcoin, dan menandatangani permintaan tersebut dengan header Signature Version 4 (SigV4) yang sesuai dengan menggunakan kode resmi AWS SDK v3. Untuk menjalankan kode, buka terminal di direktori yang sama dengan file Anda dan jalankan yang berikut:

```

npm i
node index.js

```

Hasil yang dihasilkan akan menyerupai yang berikut:

```

{"hash":"00000000c937983704a73af28acdec37b049d214adbda81d7e2a3dd146f6ed09",
  "confirmations":784126,"height":1000, "version":1,"versionHex":"00000001",
  "merkleroot":"fe28050b93faea61fa88c4c630f0e1f0a1c24d0082dd0e10d369e13212128f33",
  "time":1232346882,
  "mediantime":1232344831,"nonce":2595206198,"bits":"1d00ffff","difficulty":1,
  "chainwork":"000000000000000000000000000000000000000000000000000000000000003e903e903e9",
  "nTx":1,

  "previousblockhash":"0000000008e647742775a230787d66fdf92c46a48c896bfbc85cdc8acc67e87d",

```

```
"nextblockhash":"00000000a2887344f8db859e372e7e4bc26b23b9de340f725afbf2edb265b4c6",  
"strippedsize":216,"size":216,"weight":864,  
"tx":["fe28050b93faea61fa88c4c630f0e1f0a1c24d0082dd0e10d369e13212128f33"]},  
"error":null,"id":"1001"}
```

Note

Permintaan sampel dalam skrip sebelumnya membuat `getBlock` panggilan dengan hash blok parameter input yang sama seperti [Buat AMB Akses Bitcoin JSON - RPC permintaan di awscurl dengan menggunakan AWS CLI](#) contoh. Untuk melakukan panggilan lain, memodifikasi `rpc` objek dalam skrip dengan Bitcoin yang berbeda JSON -RPC. Anda dapat mengubah opsi properti host ke Bitcoin testnet untuk melakukan panggilan pada titik akhir tersebut.

Gunakan AMB Akses Bitcoin di atas AWS PrivateLink

AWS PrivateLink adalah teknologi yang sangat tersedia dan dapat diskalakan yang dapat Anda gunakan untuk menghubungkan Anda VPC ke layanan secara pribadi seolah-olah mereka ada di layanan Anda. VPC Anda tidak harus menggunakan gateway internet, NAT perangkat, alamat IP publik, AWS Koneksi Direct Connect, atau AWS VPN Koneksi site-to-site untuk berkomunikasi dengan layanan dari subnet pribadi Anda. Untuk informasi lebih lanjut tentang AWS PrivateLink atau untuk mengatur AWS PrivateLink, lihat [Apa itu AWS PrivateLink?](#)

Anda dapat mengirim Bitcoin JSON - RPC permintaan untuk AMB Mengakses Bitcoin AWS PrivateLink dengan menggunakan VPC endpoint. Permintaan ke titik akhir pribadi ini tidak diteruskan melalui internet terbuka, sehingga Anda dapat mengirim permintaan langsung ke titik akhir Bitcoin dengan menggunakan otentikasi SiGv4 yang sama. Untuk informasi selengkapnya, lihat [Akses AWS layanan melalui AWS PrivateLink](#).

Untuk nama Layanan, cari Amazon Managed Blockchain di AWS kolom layanan. Untuk informasi selengkapnya, silakan lihat [AWS Layanan yang terintegrasi dengan AWS PrivateLink](#). Nama layanan untuk titik akhir akan dalam format berikut: `com.amazonaws.AWS-REGION.managedblockchain.bitcoin.NETWORK-TYPE`.

Misalnya: `com.amazonaws.us-east-1.managedblockchain.bitcoin.testnet`.

Kasus penggunaan Bitcoin dengan Amazon Managed Blockchain (AMB) Akses Bitcoin

Topik ini menyediakan daftar kasus penggunaan AMB Access Bitcoin

Topik

- [Buat dompet Bitcoin \(BTC\) untuk mengirim dan menerima BTC](#)
- [Analisis aktivitas pada blockchain Bitcoin](#)
- [Verifikasi pesan yang ditandatangani menggunakan key pair Bitcoin](#)
- [Periksa mempool Bitcoin](#)

Buat dompet Bitcoin (BTC) untuk mengirim dan menerima BTC

BTC, cryptocurrency asli di jaringan Bitcoin, berfungsi sebagai komponen penting dari model keamanan jaringan. Ini juga bertindak sebagai komoditas dan media pertukaran, banyak digunakan oleh lembaga, bisnis, dan individu. Akibatnya, banyak aplikasi dompet mengandalkan node Bitcoin untuk berinteraksi dengan blockchain Bitcoin. Aplikasi ini menghitung saldo output yang tidak terpakai (UTXO) untuk satu set alamat tertentu, menandatangani dan mengirim transaksi ke jaringan Bitcoin, dan mengambil data tentang transaksi historis.

Berikut ini adalah contoh dari beberapa Bitcoin JSON-RPC yang Amazon Managed Blockchain (AMB) Access Bitcoin mendukung untuk transaksi dompet BTC:

- `estimatesmartfee`
- `createmultisig`
- `createrawtransaction`
- `sendrawtransaction`

Untuk informasi selengkapnya, lihat [Didukung JSON-RPC](#).

Analisis aktivitas pada blockchain Bitcoin

Anda dapat menganalisis volume aktivitas transaksi pada blockchain Bitcoin dengan menggunakan metode `getchaintxstats` JSON-RPC. JSON-RPC ini memungkinkan Anda untuk mengakses

metrik seperti tarif transaksi rata-rata per detik, jumlah total transaksi, jumlah blok, dan banyak lagi. Anda juga dapat menentukan jendela nomor blok atau hash blok sebagai pembatas untuk menghitung statistik ini untuk sekumpulan blok tertentu dalam jaringan, jika diinginkan.

Untuk informasi selengkapnya, lihat [Didukung JSON-RPC](#).

Verifikasi pesan yang ditandatangani menggunakan key pair Bitcoin

Dompet Bitcoin memiliki kunci pribadi dan kunci publik yang membentuk key pair. Kunci ini digunakan untuk menandatangani transaksi dan berfungsi sebagai identitas pengguna di blockchain. Kunci publik digunakan untuk membuat alamat, yang merupakan pengidentifikasi alfanumerik standar (panjang 27 hingga 34 karakter). Alamat ini digunakan untuk menerima output BTC dan menangani transaksi atau pesan.

Dengan dompet Bitcoin, pengguna juga dapat menandatangani dan memverifikasi pesan secara kriptografi. Proses ini sering digunakan untuk membuktikan kepemilikan alamat dompet tertentu dan BTC yang terkait dengannya. Dengan menggunakan `verifymessage` Bitcoin JSON-RPC, Anda dapat memeriksa keaslian dan validitas pesan yang ditandatangani oleh dompet lain. Secara khusus, node Bitcoin dapat digunakan untuk memverifikasi apakah pesan telah ditandatangani menggunakan kunci pribadi yang sesuai dengan alamat turunan kunci publik yang disediakan dalam pesan yang ditandatangani itu sendiri.

Untuk informasi selengkapnya, lihat [Didukung JSON-RPC](#).

Periksa mempool Bitcoin

Banyak aplikasi perlu mengakses mempool untuk melacak transaksi yang tertunda, mendapatkan daftar semua transaksi yang tertunda, atau mencari tahu dari mana transaksi berasal. Untuk melakukan ini, ada Bitcoin JSON-RPC seperti `getmempoolancestors`, `getmempoolentry`, dan `getrawmempool` yang mendukung aktivitas ini. Bitcoin JSON-RPC ini membantu aplikasi mendapatkan informasi yang mereka butuhkan dari mempool.

Amazon Managed Blockchain (AMB) Access `testmempoolaccept` Bitcoin juga mendukung Bitcoin JSON-RPC, yang memungkinkan Anda memverifikasi apakah transaksi memenuhi aturan protokol dan akan diterima oleh node sebelum mengirimkan. Dompet, pertukaran, dan entitas lain yang secara langsung mengirimkan transaksi ke blockchain Bitcoin menggunakan Bitcoin JSON-RPC ini.

Untuk informasi selengkapnya, lihat [Didukung JSON-RPC](#).

Bitcoin JSON-RPC yang Didukung dengan Amazon Managed Blockchain (AMB) Akses Bitcoin

Topik ini memberikan daftar dan referensi ke Bitcoin JSON-RPC yang didukung oleh Managed Blockchain. Setiap JSON-RPC yang didukung memiliki deskripsi singkat tentang penggunaannya.

Note

- Anda dapat mengautentikasi Bitcoin JSON-RPC di Blockchain Terkelola dengan menggunakan proses [penandatanganan Signature Version 4 \(SigV4\)](#). Ini berarti bahwa hanya prinsipal IAM resmi di AWS akun yang dapat berinteraksi dengannya dengan menggunakan Bitcoin JSON-RPC. Berikan AWS kredensi (ID kunci akses dan kunci akses rahasia) dengan panggilan.
- Jika respons HTTP Anda lebih besar dari 10 MB, Anda akan mendapatkan kesalahan. Untuk memperbaikinya, Anda harus mengatur header kompresi ke `Accept-Encoding: gzip`. Respons terkompresi yang diterima klien Anda berisi header berikut: `Content-Type: application/json` dan `Content-Encoding: gzip`
- Amazon Managed Blockchain (AMB) Akses Bitcoin menghasilkan kesalahan 400 untuk permintaan JSON-RPC yang salah bentuk.
- Gunakan `sendrawtransaction` JSON-RPC untuk mengirimkan transaksi yang memperbarui status blockchain Bitcoin.
- AMB Access Bitcoin memiliki batas permintaan default 100 permintaan per detik (RPS), per `NETWORK_TYPE`, per AWS Wilayah.


Untuk meningkatkan kuota Anda, Anda harus menghubungi AWS dukungan. Untuk menghubungi AWS dukungan, masuk ke [AWS Support Center Console](#). Pilih Buat kasus. Pilih Teknis. Pilih Blockchain Terkelola sebagai layanan Anda. Pilih Access:Bitcoin sebagai Kategori Anda dan panduan Umum sebagai Keparahan Anda. Masukkan Kuota RPC sebagai Subjek dan di kotak teks Deskripsi dan cantumkan batas kuota yang berlaku untuk kebutuhan Anda di RPS per jaringan Bitcoin per Wilayah. Kirimkan kasus Anda.

Didukung JSON-RPC

AMB Access Bitcoin mendukung Bitcoin JSON-RPC berikut. Setiap panggilan yang didukung memiliki deskripsi singkat tentang penggunaannya.

Kategori	JSON-RPC	Deskripsi
RPC Blockchain	getbestblockhash	Mengembalikan hash dari blok (tip) terbaik di rantai yang paling banyak divalidasi dan sepenuhnya divalidasi.
	getblock	Jika verbositas adalah 0, mengembalikan string yang diserialisasi, hex-encode data untuk blok 'hash'. Jika verbositas adalah 1, mengembalikan Object dengan informasi tentang blok 'hash'. Jika verbositas adalah 2, mengembalikan Object dengan informasi tentang blok 'hash' dan informasi tentang setiap transaksi. Jika verbositas adalah 3, mengembalikan Object dengan informasi tentang blok 'hash' dan informasi tentang setiap transaksi, termasuk <code>prevout</code> informasi untuk input.
	getblockchaininfo	Mengembalikan objek yang berisi berbagai info negara mengenai pemrosesan blockchain.
	getblockcount	Mengembalikan ketinggian rantai yang paling banyak bekerja dan sepenuhnya divalidasi. Blok genesis memiliki tinggi 0.
	getblockfilter	Mengambil filter konten BIP 157 untuk blok tertentu menggunakan hash blok.
	getblockhash	Mengembalikan hash blok best-block-chain pada ketinggian yang disediakan.
	getblockheader	Jika verbose adalah false, mengembalikan string yang diserialisasi, hex-encode data

Kategori	JSON-RPC	Deskripsi
		untuk blockheader 'hash'. Jika verbose adalah true, mengembalikan Object dengan informasi tentang blockheader 'hash'.
	getblockstats	Menghitung statistik per blok untuk jendela tertentu. Semua jumlah dalam satoshi. Ini tidak akan berhasil untuk beberapa ketinggian dengan pemangkasan.
	getchaintips	Mengembalikan informasi tentang semua tip yang diketahui di pohon blok, termasuk rantai utama dan cabang yatim piatu.
	getchaintxstats	Menghitung statistik tentang jumlah total dan tingkat transaksi dalam rantai.
	mendapatkan kesulitan	Mengembalikan proof-of-work kesulitan sebagai kelipatan dari kesulitan minimum.
	getmempoolancestors	Jika txid ada di mempool, mengembalikan semua leluhur dalam mempool.
	getmempooldescendants	Jika txid ada di mempool, mengembalikan semua turunan in-mempool.
	getmempoolentry	Mengembalikan data mempool untuk transaksi yang diberikan.
	getmempoolinfo	Mengembalikan rincian tentang keadaan aktif kolam memori TX.

Kategori	JSON-RPC	Deskripsi
	<u>getrawmempool</u>	Mengembalikan semua ID transaksi di kolom memori sebagai array JSON dari ID transaksi string. <div style="border: 1px solid #add8e6; border-radius: 10px; padding: 10px; background-color: #e6f2ff;"> <p> Note verbose = true tidak didukung.</p> </div>
	<u>gettxout</u>	Mengembalikan rincian tentang output transaksi yang tidak terpakai.
	<u>gettxoutproof</u>	Mengembalikan bukti hex-encoded bahwa “txid” disertakan dalam blok.
<u>RPC Transaksi Mentah</u>	<u>createrawtransaksi</u>	Membuat transaksi menghabiskan input yang diberikan dan menciptakan output baru.
	<u>decoderawtransaksi</u>	Mengembalikan objek JSON yang mewakili transaksi serial, hex-encode.
	<u>decodescript</u>	Mendekode skrip yang dikodekan hex-.
	<u>getrawtransaksi</u>	Mengembalikan data transaksi mentah.
	<u>sendrawtransaksi</u>	Mengirimkan transaksi mentah (serial, hex-encoded) ke node lokal dan jaringan.
	<u>testmempoolaccept</u>	Mengembalikan hasil tes penerimaan mempool yang menunjukkan apakah transaksi mentah (serial, hex-encoded) akan diterima oleh mempool. Ini memeriksa apakah transaksi melanggar konsensus atau aturan kebijakan.
<u>Util RPC</u>	<u>createmultisig</u>	Membuat alamat multi-tanda tangan dengan n tanda tangan dari kunci m diperlukan.

Kategori	JSON-RPC	Deskripsi
	<u>estimasi martfee</u>	Memperkirakan perkiraan biaya per kilobyte yang diperlukan untuk transaksi untuk memulai konfirmasi dalam blok <code>conf_target</code> , jika memungkinkan, dan mengembalikan jumlah blok yang estimasi valid. Menggunakan ukuran transaksi virtual, sebagaimana didefinisikan dalam BIP 141 (data saksi didiskon).
	<u>validatealamat</u>	Mengembalikan informasi tentang alamat bitcoin yang diberikan.
	<u>verifymessage</u>	Memverifikasi pesan yang ditandatangani.

Keamanan di Amazon Managed Blockchain (AMB) Akses Bitcoin

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 Bitcoin, 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 Bitcoin. Topik berikut menunjukkan cara mengonfigurasi AMB Access Bitcoin 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 Bitcoin Anda.

Topik

- [Perlindungan data di Amazon Managed Blockchain \(AMB\) Akses Bitcoin](#)
- [Manajemen identitas dan akses untuk Amazon Managed Blockchain \(AMB\) Akses Bitcoin](#)

Perlindungan data di Amazon Managed Blockchain (AMB) Akses Bitcoin

Bagian AWS [model tanggung jawab bersama model](#) berlaku untuk perlindungan data di Amazon Managed Blockchain (AMB) Akses Bitcoin. Seperti yang dijelaskan dalam model ini, AWS bertanggung jawab untuk melindungi infrastruktur global yang menjalankan semua AWS Cloud. Anda bertanggung jawab untuk menjaga kontrol atas konten Anda yang di-host di infrastruktur ini. Anda juga bertanggung jawab atas konfigurasi keamanan dan tugas manajemen untuk Layanan AWS yang Anda gunakan. Untuk informasi selengkapnya tentang privasi data, lihat [Privasi Data FAQ](#). Untuk informasi tentang perlindungan data di Eropa, lihat [AWS Model Tanggung Jawab Bersama dan posting GDPR](#) blog di AWS Blog Keamanan.

Untuk tujuan perlindungan data, kami menyarankan Anda untuk melindungi Akun AWS kredensi 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 otentikasi multi-faktor (MFA) dengan setiap akun.
- Gunakan SSL/TLS untuk berkomunikasi dengan AWS sumber daya. Kami membutuhkan TLS 1.2 dan merekomendasikan TLS 1.3.
- Siapkan API dan pencatatan aktivitas pengguna dengan AWS CloudTrail. Untuk informasi tentang menggunakan CloudTrail jalur untuk menangkap AWS kegiatan, lihat [Bekerja dengan CloudTrail jalan setapak](#) di AWS CloudTrail Panduan Pengguna.
- Gunakan AWS solusi enkripsi, bersama dengan semua kontrol keamanan default di dalamnya Layanan AWS.
- Gunakan layanan keamanan terkelola lanjut seperti Amazon Macie, yang membantu menemukan dan mengamankan data sensitif yang disimpan di Amazon S3.
- Jika Anda memerlukan FIPS 140-3 modul kriptografi yang divalidasi saat mengakses AWS melalui antarmuka baris perintah atau API, gunakan FIPS titik akhir. Untuk informasi selengkapnya tentang FIPS titik akhir yang tersedia, lihat [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 ketika Anda bekerja dengan AMB Access Bitcoin 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. Jika Anda memberikan URL ke server eksternal, kami sangat menyarankan agar Anda tidak menyertakan informasi kredensial dalam URL untuk memvalidasi permintaan Anda ke server tersebut.

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 HTTPS TLS /koneksi untuk mengenkripsi semua data yang dikirimkan dari komputer klien yang menjalankan AWS CLI kepada AWS titik akhir layanan.

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

Manajemen identitas dan akses untuk Amazon Managed Blockchain (AMB) Akses Bitcoin

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

Topik

- [Audiens](#)
- [Mengautentikasi dengan identitas](#)
- [Mengelola akses menggunakan kebijakan](#)
- [Bagaimana Amazon Managed Blockchain \(AMB\) Access Bitcoin bekerja dengan IAM](#)
- [Contoh kebijakan berbasis identitas untuk Amazon Managed Blockchain \(\) Akses Bitcoin AMB](#)

- [Pemecahan Masalah Amazon Managed Blockchain \(AMB\) Akses identitas dan akses Bitcoin](#)

Audiens

Cara Anda menggunakan AWS Identity and Access Management (IAM) berbeda, tergantung pada pekerjaan yang Anda lakukan di AMB Access Bitcoin.

Pengguna layanan — Jika Anda menggunakan layanan AMB Access Bitcoin untuk melakukan pekerjaan Anda, administrator Anda memberi Anda kredensi dan izin yang Anda butuhkan. Saat Anda menggunakan lebih banyak fitur AMB Access Bitcoin untuk melakukan pekerjaan Anda, Anda mungkin memerlukan izin tambahan. Memahami cara akses dikelola dapat membantu Anda meminta izin yang tepat dari administrator Anda. Jika Anda tidak dapat mengakses fitur di AMB Access Bitcoin, lihat [Pemecahan Masalah Amazon Managed Blockchain \(AMB\) Akses identitas dan akses Bitcoin](#).

Administrator layanan — Jika Anda bertanggung jawab atas sumber daya AMB Akses Bitcoin di perusahaan Anda, Anda mungkin memiliki akses penuh ke AMB Akses Bitcoin. Tugas Anda adalah menentukan fitur dan sumber daya AMB Access Bitcoin mana yang harus diakses pengguna layanan Anda. Anda kemudian harus mengirimkan permintaan ke IAM administrator Anda untuk mengubah izin pengguna layanan Anda. Tinjau informasi di halaman ini untuk memahami konsep dasar IAM. Untuk mempelajari lebih lanjut tentang bagaimana perusahaan Anda dapat menggunakan IAM AMB Access Bitcoin, lihat [Bagaimana Amazon Managed Blockchain \(AMB\) Access Bitcoin bekerja dengan IAM](#).

IAM administrator — Jika Anda seorang IAM administrator, Anda mungkin ingin mempelajari detail tentang cara menulis kebijakan untuk mengelola akses ke AMB Akses Bitcoin. Untuk melihat contoh AMB Akses kebijakan berbasis identitas Bitcoin yang dapat Anda gunakan, lihat. IAM [Contoh kebijakan berbasis identitas untuk Amazon Managed Blockchain \(\) Akses Bitcoin AMB](#)

Mengautentikasi dengan identitas

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

Anda dapat masuk AWS sebagai identitas federasi dengan menggunakan kredensi yang disediakan melalui sumber identitas. AWS IAM Identity Center Pengguna (Pusat IAM Identitas), autentikasi masuk tunggal perusahaan Anda, dan kredensi Google atau Facebook Anda adalah contoh identitas federasi. Saat Anda masuk sebagai identitas federasi, administrator Anda sebelumnya menyiapkan

federasi identitas menggunakan IAM peran. Ketika Anda mengakses AWS dengan menggunakan federasi, Anda secara tidak langsung mengambil peran.

Bergantung pada jenis pengguna Anda, Anda dapat masuk ke AWS Management Console atau portal AWS akses. Untuk informasi selengkapnya tentang masuk AWS, lihat [Cara masuk ke Panduan AWS Sign-In Pengguna Anda Akun AWS](#).

Jika Anda mengakses AWS secara terprogram, AWS sediakan kit pengembangan perangkat lunak (SDK) dan antarmuka baris perintah (CLI) untuk menandatangani permintaan Anda secara kriptografis dengan menggunakan kredensial Anda. Jika Anda tidak menggunakan AWS alat, Anda harus menandatangani permintaan sendiri. Untuk informasi selengkapnya tentang menggunakan metode yang disarankan untuk menandatangani permintaan sendiri, lihat [Versi AWS Tanda Tangan 4 untuk API permintaan](#) di Panduan IAM Pengguna.

Apa pun metode autentikasi yang digunakan, Anda mungkin diminta untuk menyediakan informasi keamanan tambahan. Misalnya, AWS merekomendasikan agar Anda menggunakan otentikasi multi-faktor (MFA) untuk meningkatkan keamanan akun Anda. Untuk mempelajari selengkapnya, lihat [Autentikasi multi-faktor](#) di Panduan AWS IAM Identity Center Pengguna dan [Autentikasi AWS multi-faktor IAM di](#) Panduan Pengguna. IAM

Akun AWS pengguna root

Saat Anda membuat Akun AWS, Anda mulai dengan satu identitas masuk yang memiliki akses lengkap ke semua Layanan AWS dan sumber daya di akun. Identitas ini disebut pengguna Akun AWS root dan diakses dengan masuk dengan alamat email dan kata sandi yang Anda gunakan untuk membuat akun. Kami sangat menyarankan agar Anda tidak menggunakan pengguna root untuk tugas sehari-hari. Lindungi kredensial pengguna root Anda dan gunakan kredensial tersebut untuk melakukan tugas yang hanya dapat dilakukan pengguna root. Untuk daftar lengkap tugas yang mengharuskan Anda masuk sebagai pengguna root, lihat [Tugas yang memerlukan kredensi pengguna root](#) di IAMPanduan Pengguna.

Identitas gabungan

Sebagai praktik terbaik, mewajibkan pengguna manusia, termasuk pengguna yang memerlukan akses administrator, untuk menggunakan federasi dengan penyedia identitas untuk mengakses Layanan AWS dengan menggunakan kredensi sementara.

Identitas federasi adalah pengguna dari direktori pengguna perusahaan Anda, penyedia identitas web, direktori Pusat Identitas AWS Directory Service, atau pengguna mana pun yang mengakses Layanan AWS dengan menggunakan kredensi yang disediakan melalui sumber identitas. Ketika

identitas federasi mengakses Akun AWS, mereka mengambil peran, dan peran memberikan kredensi sementara.

Untuk manajemen akses terpusat, kami sarankan Anda menggunakan AWS IAM Identity Center. Anda dapat membuat pengguna dan grup di Pusat IAM Identitas, atau Anda dapat menghubungkan dan menyinkronkan ke sekumpulan pengguna dan grup di sumber identitas Anda sendiri untuk digunakan di semua aplikasi Akun AWS dan aplikasi Anda. Untuk informasi tentang Pusat IAM Identitas, lihat [Apa itu Pusat IAM Identitas?](#) dalam AWS IAM Identity Center User Guide.

Pengguna dan grup IAM

[IAMPengguna](#) adalah identitas dalam diri Anda Akun AWS yang memiliki izin khusus untuk satu orang atau aplikasi. Jika memungkinkan, sebaiknya mengandalkan kredensi sementara daripada membuat IAM pengguna yang memiliki kredensi jangka panjang seperti kata sandi dan kunci akses. Namun, jika Anda memiliki kasus penggunaan khusus yang memerlukan kredensi jangka panjang dengan IAM pengguna, kami sarankan Anda memutar kunci akses. Untuk informasi selengkapnya, lihat [Memutar kunci akses secara teratur untuk kasus penggunaan yang memerlukan kredensi jangka panjang](#) di IAMPanduan Pengguna.

[IAMGrup](#) adalah identitas yang menentukan kumpulan IAM pengguna. Anda tidak dapat masuk sebagai grup. Anda dapat menggunakan grup untuk menentukan izin bagi beberapa pengguna sekaligus. Grup mempermudah manajemen izin untuk sejumlah besar pengguna sekaligus. Misalnya, Anda dapat memiliki grup bernama IAMAdmins dan memberikan izin grup tersebut untuk mengelola sumber daya IAM.

Pengguna berbeda dari peran. Pengguna secara unik terkait dengan satu orang atau aplikasi, tetapi peran dimaksudkan untuk dapat digunakan oleh siapa pun yang membutuhkannya. Pengguna memiliki kredensial jangka panjang permanen, tetapi peran memberikan kredensial sementara. Untuk mempelajari selengkapnya, lihat [Kasus penggunaan untuk IAM pengguna](#) di Panduan IAM Pengguna.

IAMperan

[IAMPeran](#) adalah identitas dalam diri Anda Akun AWS yang memiliki izin khusus. Ini mirip dengan IAM pengguna, tetapi tidak terkait dengan orang tertentu. Untuk mengambil IAM peran sementara di dalam AWS Management Console, Anda dapat [beralih dari pengguna ke IAM peran \(konsol\)](#). Anda dapat mengambil peran dengan memanggil AWS CLI atau AWS API operasi atau dengan menggunakan kustomURL. Untuk informasi selengkapnya tentang metode penggunaan peran, lihat [Metode untuk mengambil peran](#) dalam Panduan IAM Pengguna.

IAMperan dengan kredensi sementara berguna dalam situasi berikut:

- Akses pengguna terfederasi – Untuk menetapkan izin ke identitas terfederasi, Anda membuat peran dan menentukan izin untuk peran tersebut. Ketika identitas terfederasi mengautentikasi, identitas tersebut terhubung dengan peran dan diberi izin yang ditentukan oleh peran. Untuk informasi tentang peran untuk federasi, lihat [Membuat peran untuk penyedia identitas pihak ketiga \(federasi\)](#) di Panduan IAM Pengguna. Jika Anda menggunakan Pusat IAM Identitas, Anda mengonfigurasi set izin. Untuk mengontrol apa yang dapat diakses identitas Anda setelah diautentikasi, Pusat IAM Identitas menghubungkan izin yang disetel ke peran. IAM Untuk informasi tentang set izin, lihat [Set izin](#) dalam Panduan Pengguna AWS IAM Identity Center .
- Izin IAM pengguna sementara — IAM Pengguna atau peran dapat mengambil IAM peran untuk sementara mengambil izin yang berbeda untuk tugas tertentu.
- Akses lintas akun — Anda dapat menggunakan IAM peran untuk memungkinkan seseorang (prinsipal tepercaya) di akun lain mengakses sumber daya di akun Anda. Peran adalah cara utama untuk memberikan akses lintas akun. Namun, dengan beberapa Layanan AWS, Anda dapat melampirkan kebijakan secara langsung ke sumber daya (alih-alih menggunakan peran sebagai proxy). Untuk mempelajari perbedaan antara peran dan kebijakan berbasis sumber daya untuk akses lintas akun, lihat [Akses sumber daya lintas akun di IAM](#) Panduan Pengguna. IAM
- Akses lintas layanan — Beberapa Layanan AWS menggunakan fitur lain Layanan AWS. Misalnya, saat Anda melakukan panggilan dalam suatu layanan, biasanya layanan tersebut menjalankan aplikasi di Amazon EC2 atau menyimpan objek di Amazon S3. Sebuah layanan mungkin melakukannya menggunakan izin prinsipal yang memanggil, menggunakan peran layanan, atau peran terkait layanan.
- Sesi akses teruskan (FAS) — Saat Anda menggunakan IAM pengguna atau peran untuk melakukan tindakan AWS, Anda dianggap sebagai prinsipal. Ketika Anda menggunakan beberapa layanan, Anda mungkin melakukan sebuah tindakan yang kemudian menginisiasi tindakan lain di layanan yang berbeda. FAS menggunakan izin dari pemanggilan utama Layanan AWS, dikombinasikan dengan permintaan Layanan AWS untuk membuat permintaan ke layanan hilir. FAS permintaan hanya dibuat ketika layanan menerima permintaan yang memerlukan interaksi dengan orang lain Layanan AWS atau sumber daya untuk menyelesaikannya. Dalam hal ini, Anda harus memiliki izin untuk melakukan kedua tindakan tersebut. Untuk detail kebijakan saat membuat FAS permintaan, lihat [Meneruskan sesi akses](#).
- Peran layanan — Peran layanan adalah [IAMperan](#) yang diasumsikan layanan untuk melakukan tindakan atas nama Anda. IAM Administrator dapat membuat, memodifikasi, dan menghapus peran layanan dari dalam IAM. Untuk informasi selengkapnya, lihat [Membuat peran untuk mendelegasikan izin ke Layanan AWS](#) dalam IAMPanduan Pengguna.

- Peran terkait layanan — Peran terkait layanan adalah jenis peran layanan yang ditautkan ke peran layanan. 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. IAMAdministrator dapat melihat, tetapi tidak mengedit izin untuk peran terkait layanan.
- Aplikasi yang berjalan di Amazon EC2 — Anda dapat menggunakan IAM peran untuk mengelola kredensial sementara untuk aplikasi yang berjalan pada EC2 instance dan membuat AWS CLI atau AWS API meminta. Ini lebih baik untuk menyimpan kunci akses dalam EC2 instance. Untuk menetapkan AWS peran ke EC2 instance dan membuatnya tersedia untuk semua aplikasinya, Anda membuat profil instance yang dilampirkan ke instance. Profil instance berisi peran dan memungkinkan program yang berjalan pada EC2 instance untuk mendapatkan kredensial sementara. Untuk informasi selengkapnya, lihat [Menggunakan IAM peran untuk memberikan izin ke aplikasi yang berjalan di EC2 instans Amazon](#) di IAMPanduan Pengguna.

Mengelola akses menggunakan kebijakan

Anda mengontrol akses AWS dengan membuat kebijakan dan melampirkannya ke AWS identitas atau sumber daya. Kebijakan adalah objek AWS yang, ketika dikaitkan dengan identitas atau sumber daya, menentukan izinnya. AWS mengevaluasi kebijakan ini ketika prinsipal (pengguna, pengguna root, atau sesi peran) membuat permintaan. Izin dalam kebijakan menentukan apakah permintaan diizinkan atau ditolak. Sebagian besar kebijakan disimpan AWS sebagai JSON dokumen. Untuk informasi selengkapnya tentang struktur dan isi dokumen JSON kebijakan, lihat [Ringkasan JSON kebijakan](#) di Panduan IAM Pengguna.

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

Secara default, pengguna dan peran tidak memiliki izin. Untuk memberikan izin kepada pengguna untuk melakukan tindakan pada sumber daya yang mereka butuhkan, IAM administrator dapat membuat IAM kebijakan. Administrator kemudian dapat menambahkan IAM kebijakan ke peran, dan pengguna dapat mengambil peran.

IAMkebijakan menentukan izin untuk tindakan terlepas dari metode yang Anda gunakan untuk melakukan operasi. Misalnya, anggaplah Anda memiliki kebijakan yang mengizinkan tindakan `iam:GetRole`. Pengguna dengan kebijakan itu bisa mendapatkan informasi peran dari AWS Management Console, AWS CLI, atau AWS API.

Kebijakan berbasis identitas

Kebijakan berbasis identitas adalah dokumen kebijakan JSON izin yang dapat dilampirkan ke identitas, seperti pengguna, grup IAM pengguna, atau peran. 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 [Menentukan IAM izin khusus dengan kebijakan yang dikelola pelanggan di Panduan Pengguna](#). IAM

Kebijakan berbasis identitas dapat dikategorikan lebih lanjut sebagai kebijakan inline atau kebijakan yang dikelola. Kebijakan inline disematkan langsung ke satu pengguna, grup, atau peran. Kebijakan terkelola adalah kebijakan mandiri yang dapat Anda lampirkan ke beberapa pengguna, grup, dan peran dalam. Akun AWS Kebijakan AWS terkelola mencakup kebijakan terkelola dan kebijakan yang dikelola pelanggan. Untuk mempelajari cara memilih antara kebijakan terkelola atau kebijakan sebaris, lihat [Memilih antara kebijakan terkelola dan kebijakan sebaris](#) di IAMPanduan Pengguna.

Kebijakan berbasis sumber daya

Kebijakan berbasis sumber daya adalah dokumen JSON kebijakan yang Anda lampirkan ke sumber daya. Contoh kebijakan berbasis sumber daya adalah kebijakan kepercayaan IAM peran 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 prinsipal tertentu pada sumber daya tersebut dan dalam kondisi apa. Anda harus [menentukan prinsipal](#) dalam kebijakan berbasis sumber daya. Prinsipal dapat mencakup akun, pengguna, peran, pengguna federasi, atau. Layanan AWS

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

Daftar kontrol akses (ACLs)

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

Amazon S3, AWS WAF, dan Amazon VPC adalah contoh layanan yang mendukung. ACLs Untuk mempelajari selengkapnya ACLs, lihat [Ikhtisar daftar kontrol akses \(ACL\)](#) di Panduan Pengembang Layanan Penyimpanan Sederhana Amazon.

Jenis-jenis kebijakan lain

AWS mendukung jenis kebijakan tambahan yang kurang umum. Jenis-jenis kebijakan ini dapat mengatur izin maksimum yang diberikan kepada Anda oleh jenis kebijakan yang lebih umum.

- **Batas izin** — Batas izin adalah fitur lanjutan tempat Anda menetapkan izin maksimum yang dapat diberikan oleh kebijakan berbasis identitas kepada entitas (pengguna atau peran). IAM IAM Anda dapat menetapkan batasan izin untuk suatu entitas. Izin yang dihasilkan adalah perpotongan antara kebijakan berbasis identitas milik entitas dan batasan izinnya. Kebijakan berbasis sumber daya yang menentukan pengguna atau peran dalam bidang `Principal` tidak dibatasi oleh batasan izin. Penolakan eksplisit dalam salah satu kebijakan ini akan menggantikan pemberian izin. Untuk informasi selengkapnya tentang batas izin, lihat [Batas izin untuk IAM entitas](#) di IAMPanduan Pengguna.
- **Kebijakan kontrol layanan (SCPs)** — SCPs adalah JSON kebijakan yang menentukan izin maksimum untuk organisasi atau unit organisasi (OU) di AWS Organizations. AWS Organizations adalah layanan untuk mengelompokkan dan mengelola secara terpusat beberapa Akun AWS yang dimiliki bisnis Anda. Jika Anda mengaktifkan semua fitur dalam suatu organisasi, maka Anda dapat menerapkan kebijakan kontrol layanan (SCPs) ke salah satu atau semua akun Anda. SCPMembatasi izin untuk entitas di akun anggota, termasuk masing-masing Pengguna root akun AWS. Untuk informasi selengkapnya tentang Organizations danSCPs, lihat [Kebijakan kontrol layanan](#) di Panduan AWS Organizations Pengguna.
- **Kebijakan sesi** – Kebijakan sesi adalah kebijakan lanjutan yang Anda berikan sebagai parameter ketika Anda membuat sesi sementara secara programatis untuk peran atau pengguna terfederasi. Izin sesi yang dihasilkan adalah perpotongan antara kebijakan berbasis identitas pengguna atau peran dan kebijakan sesi. Izin juga bisa datang dari kebijakan berbasis sumber daya. Penolakan secara tegas dalam salah satu kebijakan ini membatalkan izin. Untuk informasi selengkapnya, lihat [Kebijakan sesi](#) di Panduan IAM Pengguna.

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 IAM Pengguna.

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

Sebelum Anda menggunakannya IAM untuk mengelola akses ke AMB Access Bitcoin, pelajari IAM fitur apa saja yang tersedia untuk digunakan dengan AMB Access Bitcoin.

IAM fitur yang dapat Anda gunakan dengan Amazon Managed Blockchain (AMB) Akses Bitcoin

IAM fitur	AMBAkses dukungan Bitcoin
Kebijakan berbasis identitas	Ya
Kebijakan berbasis sumber daya	Tidak
Tindakan kebijakan	Ya
Sumber daya kebijakan	Tidak
Kunci kondisi kebijakan	Tidak
ACLs	Tidak
ABAC(tag dalam kebijakan)	Tidak
Kredensial sementara	Tidak
Izin prinsipal	Tidak
Peran layanan	Tidak
Peran terkait layanan	Tidak

Untuk mendapatkan tampilan tingkat tinggi tentang cara AMB Access Bitcoin dan AWS layanan lainnya bekerja dengan sebagian besar IAM fitur, lihat [AWS layanan yang berfungsi IAM](#) di Panduan IAM Pengguna.

Kebijakan berbasis identitas untuk Access Bitcoin AMB

Mendukung kebijakan berbasis identitas: Ya

Kebijakan berbasis identitas adalah dokumen kebijakan JSON izin yang dapat dilampirkan ke identitas, seperti pengguna, grup IAM pengguna, atau peran. 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 [Menentukan IAM izin khusus dengan kebijakan yang dikelola pelanggan di Panduan Pengguna](#). IAM

Dengan kebijakan IAM berbasis identitas, Anda dapat menentukan tindakan dan sumber daya yang diizinkan atau ditolak serta kondisi di mana tindakan diizinkan atau ditolak. Anda tidak dapat menentukan secara spesifik prinsipal dalam sebuah kebijakan berbasis identitas karena prinsipal berlaku bagi pengguna atau peran yang melekat kepadanya. Untuk mempelajari semua elemen yang dapat Anda gunakan dalam JSON kebijakan, lihat [referensi elemen IAM JSON kebijakan](#) di Panduan IAM Pengguna.

Contoh kebijakan berbasis identitas untuk Access Bitcoin AMB

Untuk melihat contoh kebijakan berbasis identitas AMB Access Bitcoin, lihat. [Contoh kebijakan berbasis identitas untuk Amazon Managed Blockchain \(\) Akses Bitcoin AMB](#)

Kebijakan berbasis sumber daya dalam Access Bitcoin AMB

Mendukung kebijakan berbasis sumber daya: Tidak

Kebijakan berbasis sumber daya adalah dokumen JSON kebijakan yang Anda lampirkan ke sumber daya. Contoh kebijakan berbasis sumber daya adalah kebijakan kepercayaan IAM peran 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 prinsipal tertentu pada sumber daya tersebut dan dalam kondisi apa. Anda harus [menentukan prinsipal](#) dalam kebijakan berbasis sumber daya. Prinsipal dapat mencakup akun, pengguna, peran, pengguna federasi, atau. Layanan AWS

Untuk mengaktifkan akses lintas akun, Anda dapat menentukan seluruh akun atau IAM entitas di akun lain sebagai prinsipal dalam kebijakan berbasis sumber daya. Menambahkan prinsipal akun silang ke kebijakan berbasis sumber daya hanya setengah dari membangun hubungan kepercayaan. Ketika prinsipal dan sumber daya berbeda Akun AWS, IAM administrator di akun tepercaya juga harus memberikan izin entitas utama (pengguna atau peran) untuk mengakses sumber daya. Mereka memberikan izin dengan melampirkan kebijakan berbasis identitas kepada entitas. Namun, jika kebijakan berbasis sumber daya memberikan akses ke prinsipal dalam akun yang sama, tidak

diperlukan kebijakan berbasis identitas tambahan. Untuk informasi selengkapnya, lihat [Akses sumber daya lintas akun IAM di Panduan IAM Pengguna](#).

Tindakan kebijakan untuk AMB Akses Bitcoin

Mendukung tindakan kebijakan: Ya

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

ActionElemen JSON kebijakan menjelaskan tindakan yang dapat Anda gunakan untuk mengizinkan atau menolak akses dalam kebijakan. Tindakan kebijakan biasanya memiliki nama yang sama dengan AWS API operasi terkait. Ada beberapa pengecualian, seperti tindakan khusus izin yang tidak memiliki operasi yang cocok. API Ada juga beberapa operasi yang memerlukan beberapa tindakan dalam suatu kebijakan. Tindakan tambahan ini disebut tindakan dependen.

Menyertakan tindakan dalam kebijakan untuk memberikan izin untuk melakukan operasi terkait.

Untuk melihat daftar tindakan AMB Akses Bitcoin, lihat [Tindakan yang Ditentukan oleh Amazon Managed Blockchain \(AMB\) Mengakses Bitcoin](#) di Referensi Otorisasi Layanan.

Tindakan kebijakan di AMB Access Bitcoin 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 `InvokeRpcBitcoin`, sertakan tindakan berikut:

```
"Action": "managedblockchain::InvokeRpcBitcoin*"
```


Untuk melihat contoh kebijakan berbasis identitas AMB Access Bitcoin, lihat. [Contoh kebijakan berbasis identitas untuk Amazon Managed Blockchain \(\) Akses Bitcoin AMB](#)

Sumber daya kebijakan untuk AMB Akses Bitcoin

Mendukung sumber daya kebijakan: Tidak

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

Elemen `Resource` JSON kebijakan menentukan objek atau objek yang tindakan tersebut berlaku. Pernyataan harus menyertakan elemen `Resource` atau `NotResource`. Sebagai praktik terbaik, tentukan sumber daya menggunakan [Amazon Resource Name \(ARN\)](#). Anda dapat melakukan ini untuk tindakan yang mendukung jenis sumber daya tertentu, yang dikenal sebagai izin tingkat sumber daya.

Untuk tindakan yang tidak mendukung izin di tingkat sumber daya, misalnya operasi pencantuman, gunakan wildcard (*) untuk menunjukkan bahwa pernyataan tersebut berlaku untuk semua sumber daya.

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

Untuk melihat daftar jenis sumber daya AMB Akses Bitcoin dan jenisnya ARNs, lihat Sumber Daya yang [Ditentukan oleh Amazon Managed Blockchain \(AMB\) Akses Bitcoin](#) di Referensi Otorisasi Layanan. Untuk mempelajari tindakan mana yang dapat Anda tentukan ARN dari setiap sumber daya, lihat [Tindakan yang Ditentukan oleh Amazon Managed Blockchain \(AMB\) Akses Bitcoin](#).

Untuk melihat contoh kebijakan berbasis identitas AMB Access Bitcoin, lihat. [Contoh kebijakan berbasis identitas untuk Amazon Managed Blockchain \(\) Akses Bitcoin AMB](#)

Kunci kondisi kebijakan untuk AMB Akses Bitcoin

Mendukung kunci kondisi kebijakan khusus layanan: Tidak

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

Elemen Condition (atau blok Condition) akan memungkinkan Anda menentukan kondisi yang menjadi dasar suatu pernyataan berlaku. Elemen Condition bersifat opsional. 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.

Jika Anda menentukan beberapa elemen Condition dalam sebuah pernyataan, atau beberapa kunci dalam elemen Condition tunggal, maka AWS akan mengevaluasinya menggunakan operasi AND logis. Jika Anda menentukan beberapa nilai untuk satu kunci kondisi, AWS mengevaluasi kondisi menggunakan OR operasi logis. Semua kondisi harus dipenuhi sebelum izin pernyataan diberikan.

Anda juga dapat menggunakan variabel placeholder saat menentukan kondisi. Misalnya, Anda dapat memberikan izin IAM pengguna untuk mengakses sumber daya hanya jika ditandai dengan nama IAM pengguna mereka. Untuk informasi selengkapnya, lihat [elemen IAM kebijakan: variabel dan tag](#) di Panduan IAM Pengguna.

AWS mendukung kunci kondisi global dan kunci kondisi khusus layanan. Untuk melihat semua kunci kondisi AWS global, lihat [kunci konteks kondisi AWS global](#) di Panduan IAM Pengguna.

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

Untuk melihat contoh kebijakan berbasis identitas AMB Access Bitcoin, lihat. [Contoh kebijakan berbasis identitas untuk Amazon Managed Blockchain \(\) Akses Bitcoin AMB](#)

ACLs di AMB Akses Bitcoin

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 AMB Akses Bitcoin

Mendukung ABAC (tag dalam kebijakan): Tidak

Attribute-based access control (ABAC) adalah strategi otorisasi yang mendefinisikan izin berdasarkan atribut. Dalam AWS, atribut ini disebut tag. Anda dapat melampirkan tag ke IAM entitas (pengguna

atau peran) dan ke banyak AWS sumber daya. Menandai entitas dan sumber daya adalah langkah pertama dari ABAC. Kemudian Anda merancang ABAC kebijakan untuk mengizinkan operasi ketika tag prinsipal cocok dengan tag pada sumber daya yang mereka coba akses.

ABAC membantu dalam lingkungan yang berkembang pesat dan membantu dengan situasi di mana manajemen kebijakan menjadi rumit.

Untuk mengendalikan akses berdasarkan tag, berikan informasi tentang tag 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 ABAC, lihat [Menentukan izin dengan ABAC otorisasi](#) di IAM Panduan Pengguna. Untuk melihat tutorial dengan langkah-langkah persiapan ABAC, lihat [Menggunakan kontrol akses berbasis atribut \(ABAC\)](#) di IAM Panduan Pengguna.

Menggunakan kredensi sementara dengan Access Bitcoin AMB

Mendukung kredensi sementara: Tidak

Beberapa Layanan AWS tidak berfungsi saat Anda masuk menggunakan kredensi sementara. Untuk informasi tambahan, termasuk yang Layanan AWS bekerja dengan kredensi sementara, lihat [Layanan AWS yang berfungsi IAM](#) di IAM Panduan Pengguna.

Anda menggunakan kredensi sementara jika Anda masuk AWS Management Console menggunakan metode apa pun kecuali nama pengguna dan kata sandi. Misalnya, ketika Anda mengakses AWS menggunakan link sign-on (SSO) tunggal perusahaan Anda, proses tersebut secara otomatis membuat kredensi sementara. Anda juga akan secara otomatis membuat kredensial sementara ketika Anda masuk ke konsol sebagai seorang pengguna lalu beralih peran. Untuk informasi selengkapnya tentang beralih peran, lihat [Beralih dari pengguna ke IAM peran \(konsol\)](#) di Panduan IAM Pengguna.

Anda dapat secara manual membuat kredensi sementara menggunakan atau. AWS CLI AWS API Anda kemudian dapat menggunakan kredensi sementara tersebut untuk mengakses. AWS AWS merekomendasikan agar Anda secara dinamis menghasilkan kredensi sementara alih-alih menggunakan kunci akses jangka panjang. Untuk informasi selengkapnya, lihat [Kredensi keamanan sementara](#) di IAM

Izin utama lintas layanan untuk Akses Bitcoin AMB

Mendukung sesi akses maju (FAS): Tidak

Saat Anda menggunakan IAM pengguna atau peran untuk melakukan tindakan AWS, Anda dianggap sebagai prinsipal. Ketika Anda menggunakan beberapa layanan, Anda mungkin melakukan sebuah tindakan yang kemudian menginisiasi tindakan lain di layanan yang berbeda. FAS menggunakan izin dari pemanggilan utama Layanan AWS, dikombinasikan dengan permintaan Layanan AWS untuk membuat permintaan ke layanan hilir. FAS permintaan hanya dibuat ketika layanan menerima permintaan yang memerlukan interaksi dengan orang lain Layanan AWS atau sumber daya untuk menyelesaikannya. Dalam hal ini, Anda harus memiliki izin untuk melakukan kedua tindakan tersebut. Untuk detail kebijakan saat membuat FAS permintaan, lihat [Meneruskan sesi akses](#).

Peran layanan untuk AMB Akses Bitcoin

Mendukung peran layanan: Tidak

Peran layanan adalah [IAM peran](#) yang diasumsikan layanan untuk melakukan tindakan atas nama Anda. IAM Administrator dapat membuat, memodifikasi, dan menghapus peran layanan dari dalam IAM. Untuk informasi selengkapnya, lihat [Membuat peran untuk mendelegasikan izin ke Layanan AWS](#) dalam [IAM Panduan Pengguna](#).

Warning

Mengubah izin untuk peran layanan dapat merusak fungsionalitas AMB Access Bitcoin. Edit peran layanan hanya jika AMB Access Bitcoin memberikan panduan untuk melakukannya.

Peran terkait layanan untuk Access Bitcoin AMB

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. IAM Administrator dapat melihat, tetapi tidak mengedit izin untuk peran terkait layanan.

Untuk detail tentang membuat atau mengelola peran terkait layanan, lihat [AWS layanan yang berfungsi](#) dengannya. 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 () Akses Bitcoin AMB

Secara default, pengguna dan peran tidak memiliki izin untuk membuat atau memodifikasi sumber daya AMB Access Bitcoin. Mereka juga tidak dapat melakukan tugas dengan menggunakan AWS Management Console, AWS Command Line Interface (AWS CLI), atau AWS API. Untuk memberikan izin kepada pengguna untuk melakukan tindakan pada sumber daya yang mereka butuhkan, IAM administrator dapat membuat IAM kebijakan. Administrator kemudian dapat menambahkan IAM kebijakan ke peran, dan pengguna dapat mengambil peran.

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

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

Topik

- [Praktik terbaik kebijakan](#)
- [Menggunakan konsol AMB Access Bitcoin](#)
- [Mengizinkan pengguna melihat izin mereka sendiri](#)
- [Mengakses jaringan Bitcoin](#)

Praktik terbaik kebijakan

Kebijakan berbasis identitas menentukan apakah seseorang dapat membuat, mengakses, atau menghapus sumber daya AMB Access Bitcoin 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 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 AWS terkelola](#) atau [kebijakan terkelola untuk fungsi pekerjaan](#) di Panduan IAM Pengguna.

- Menerapkan izin hak istimewa paling sedikit — Saat Anda menetapkan izin dengan IAM kebijakan, berikan hanya 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 penggunaan IAM untuk menerapkan izin, lihat [Kebijakan dan izin IAM di IAM](#) Panduan Pengguna.
- Gunakan ketentuan dalam IAM kebijakan untuk membatasi akses lebih lanjut — Anda dapat menambahkan kondisi ke kebijakan Anda untuk membatasi akses ke tindakan dan sumber daya. Misalnya, 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 AWS CloudFormation. Untuk informasi selengkapnya, lihat [elemen IAM JSON kebijakan: Kondisi](#) dalam Panduan IAM Pengguna.
- Gunakan IAM Access Analyzer untuk memvalidasi IAM kebijakan Anda guna memastikan izin yang aman dan fungsional — IAM Access Analyzer memvalidasi kebijakan baru dan yang sudah ada sehingga kebijakan tersebut mematuhi bahasa IAM kebijakan () 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 [Memvalidasi kebijakan dengan IAM Access Analyzer](#) di IAM Panduan Pengguna.
- Memerlukan otentikasi multi-faktor (MFA) - Jika Anda memiliki skenario yang mengharuskan IAM pengguna atau pengguna root di Anda Akun AWS, aktifkan MFA untuk keamanan tambahan. Untuk meminta MFA kapan API operasi dipanggil, tambahkan MFA kondisi ke kebijakan Anda. Untuk informasi selengkapnya, lihat [API Akses aman dengan MFA](#) di Panduan IAM Pengguna.

Untuk informasi selengkapnya tentang praktik terbaik di IAM, lihat [Praktik terbaik keamanan IAM di](#) Panduan IAM Pengguna.

Menggunakan konsol AMB Access Bitcoin

Untuk mengakses konsol Amazon Managed Blockchain (AMB) Access Bitcoin, Anda harus memiliki seperangkat izin minimum. Izin ini harus memungkinkan Anda untuk membuat daftar dan melihat detail tentang sumber daya AMB Akses Bitcoin di sumber daya 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 cocok dengan API operasi yang mereka coba lakukan.

Untuk memastikan bahwa pengguna dan peran masih dapat menggunakan konsol AMB Access Bitcoin, lampirkan juga AMB Access Bitcoin *ConsoleAccess* atau kebijakan *ReadOnly* AWS terkelola ke entitas. Untuk informasi selengkapnya, lihat [Menambahkan izin ke pengguna](#) di Panduan IAM Pengguna.

Mengizinkan pengguna melihat izin mereka sendiri

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

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "ViewOwnUserInfo",
      "Effect": "Allow",
      "Action": [
        "iam:GetUserPolicy",
        "iam:ListGroupsWithUser",
        "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 Bitcoin

Note

Untuk mengakses titik akhir publik untuk Bitcoin mainnet dan testnet melakukan JSON RPC panggilan, Anda memerlukan kredensi pengguna (AWS_ACCESS_KEY_ID dan AWS_SECRET_ACCESS_KEY) yang memiliki IAM izin yang sesuai untuk Akses Bitcoin. AMB

Example IAM Kebijakan untuk mengakses semua Jaringan Bitcoin

Contoh ini memberi IAM pengguna Akun AWS akses Anda ke semua jaringan Bitcoin.

```

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

```

Example IAM Kebijakan untuk mengakses jaringan Bitcoin Testnet

Contoh ini memberi IAM pengguna Akun AWS akses Anda ke testnet jaringan Bitcoin.

```

{
  "Version": "2012-10-17",

```



```
"Statement": [  
  {  
    "Sid": "AccessBitcoinTestnet",  
    "Effect": "Allow",  
    "Action": [  
      "managedblockchain:InvokeRpcBitcoinTestnet"  
    ],  
    "Resource": "*"   
  }  
]
```

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

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

Topik

- [Saya tidak berwenang untuk melakukan tindakan di AMB Access Bitcoin](#)
- [Saya tidak berwenang untuk melakukan iam: PassRole](#)
- [Saya ingin mengizinkan orang-orang di luar saya Akun AWS untuk mengakses sumber daya AMB Akses Bitcoin saya](#)

Saya tidak berwenang untuk melakukan tindakan di AMB Access Bitcoin

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 `mateojackson` IAM pengguna mencoba menggunakan konsol untuk melihat detail tentang `my-example-widget` sumber daya fiksi tetapi tidak memiliki izin `managedblockchain::GetWidget` fiksi.

```
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 berwenang untuk melakukan `iam:PassRole` tindakan, kebijakan Anda harus diperbarui agar Anda dapat meneruskan peran ke AMB Akses Bitcoin.

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 IAM pengguna bernama `marymajor` mencoba menggunakan konsol untuk melakukan tindakan di AMB Access Bitcoin. 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-orang di luar saya Akun AWS untuk mengakses sumber daya AMB Akses Bitcoin 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 Bitcoin mendukung fitur-fitur ini, lihat [Bagaimana Amazon Managed Blockchain \(AMB\) Access Bitcoin 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 IAM pengguna lain Akun AWS yang Anda miliki](#) di Panduan IAM Pengguna.
- 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 IAM Pengguna.
- Untuk mempelajari cara menyediakan akses melalui federasi identitas, lihat [Menyediakan akses ke pengguna yang diautentikasi secara eksternal \(federasi identitas\) di 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](#) Panduan Pengguna. IAM

Logging Amazon Managed Blockchain (AMB) Mengakses peristiwa Bitcoin dengan menggunakan AWS CloudTrail

Note

Amazon Managed Blockchain (AMB) Akses Bitcoin tidak mendukung acara manajemen.

Amazon Managed Blockchain terintegrasi dengan 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 Bitcoin untuk Blockchain Terkelola sebagai peristiwa pesawat data.

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

Untuk mempelajari selengkapnya CloudTrail, lihat [Panduan AWS CloudTrail Pengguna](#).

AMB Akses informasi Bitcoin di CloudTrail

AWS CloudTrail diaktifkan secara default saat Anda membuat Akun AWS. Namun, untuk melihat siapa yang memanggil titik akhir AMB Access Bitcoin, Anda harus mengonfigurasi CloudTrail untuk mencatat peristiwa bidang data.

Untuk menyimpan catatan peristiwa yang sedang berlangsung di Anda Akun AWS, termasuk peristiwa pesawat data untuk AMB Access Bitcoin, Anda harus membuat jejak. Jejak membuat CloudTrail pengiriman file log ke bucket Amazon S3. Secara default, saat Anda membuat jejak di AWS Management Console, jejak berlaku untuk semua Wilayah AWS. Jejak mencatat peristiwa dari semua Wilayah yang didukung di AWS partisi dan mengirimkan file log ke bucket Amazon S3 yang Anda tentukan. Selain itu, Anda dapat mengonfigurasi AWS layanan lain untuk menganalisis data ini lebih lanjut dan bertindak atas data peristiwa yang dikumpulkan di CloudTrail log. Untuk informasi selengkapnya, lihat berikut:

- [Menggunakan CloudTrail untuk melacak Bitcoin JSON-RPC](#)
- [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 Bitcoin.

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

- Apakah permintaan itu dibuat dengan kredensial pengguna root atau AWS Identity and Access Management (IAM).
- Apakah permintaan tersebut dibuat dengan kredensial keamanan sementara atau tidak untuk peran atau pengguna gabungan.
- Apakah permintaan itu dibuat oleh AWS layanan lain.

Untuk informasi selengkapnya, lihat elemen [CloudTrail UserIdentity](#).

Memahami entri file log AMB Access Bitcoin

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 yang diurutkan dari panggilan AMB Access Bitcoin API, sehingga tidak muncul dalam urutan tertentu.

Menggunakan CloudTrail untuk melacak Bitcoin JSON-RPC

Anda dapat menggunakan CloudTrail untuk melacak siapa di akun Anda yang memanggil titik akhir AMB Access Bitcoin dan apa JSON-RPC dipanggil sebagai peristiwa data. Secara default, saat Anda membuat jejak, peristiwa data tidak dicatat. Untuk merekam siapa yang memanggil titik akhir AMB Access Bitcoin sebagai peristiwa CloudTrail data, Anda harus secara eksplisit menambahkan sumber daya atau jenis sumber daya yang didukung yang ingin Anda kumpulkan aktivitasnya ke jejak. Amazon Managed Blockchain mendukung penambahan peristiwa data dengan menggunakan AWS Management Console, AWS SDK, dan AWS CLI. Untuk informasi selengkapnya, lihat [Log peristiwa menggunakan pemilih lanjutan](#) di Panduan AWS CloudTrail Pengguna.

Untuk mencatat peristiwa data dalam jejak, gunakan [put-event-selectors](#) operasi setelah Anda membuat jejak. Gunakan `--advanced-event-selectors` opsi untuk menentukan jenis `AWS::ManagedBlockchain::Network` sumber daya untuk mulai mencatat peristiwa data untuk menentukan siapa yang memanggil titik akhir AMB Access Bitcoin.

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

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

```
aws cloudtrail put-event-selectors \  
  
--region us-east-1 \  
--trail-name my-bitcoin-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 Bitcoin JSON-RPC dibuat ke salah satu titik akhir AMB Access Bitcoin, alamat IP tempat permintaan itu berasal, siapa yang membuat permintaan, kapan dibuat, dan detail tambahan lainnya.

```
{
  "eventVersion": "1.08",
  "userIdentity": {
    "type": "AssumedRole",
    "principalId": "ARO554U062RJ7KSB7FAX:777777777777",
    "arn": "arn:aws:sts::111122223333:assumed-role/Admin/777777777777",
    "accountId": "111122223333"
  },
  "eventTime": "2023-04-12T19:00:22Z",
  "eventSource": "managedblockchain.amazonaws.com",
  "eventName": "getblock",
  "awsRegion": "us-east-1",
  "sourceIPAddress": "111.222.333.444",
  "userAgent": "python-requests/2.28.1",
  "errorCode": "-",
  "errorMessage": "-",
  "requestParameters": {
    "jsonrpc": "2.0",
    "method": "getblock",
    "params": [],
    "id": 1
  },
  "responseElements": null,
  "requestID": "DRznHHEjIAMFSzA=",
  "eventID": "baeb232d-2c6b-46cd-992c-0e4033aace86",
  "readOnly": true,
  "resources": [{
    "type": "AWS::ManagedBlockchain::Network",
    "ARN": "arn:aws:managedblockchain::networks/n-bitcoin-mainnet"
  }],
  "eventType": "AwsApiCall",
  "managementEvent": false,
  "recipientAccountId": "111122223333",
  "eventCategory": "Data"
}
```

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