



デベロッパーガイド

AMB アクセス Bitcoin



AMB アクセス Bitcoin: デベロッパーガイド

Copyright © 2025 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Amazon の商標およびトレードドレスはAmazon 以外の製品およびサービスに使用することはできません。また、お客様に誤解を与える可能性がある形式で、または Amazon の信用を損なう形式で使用することもできません。Amazon が所有していないその他のすべての商標は Amazon との提携、関連、支援関係の有無にかかわらず、それら該当する所有者の資産です。

Table of Contents

Amazon Managed Blockchain (AMB) とは何ですか？	1
AMB Access Bitcoin を初めてお使いになる方向けの情報	1
主要なコンセプト	3
考慮事項と制限事項	4
設定	6
前提条件と考慮事項	6
にサインアップする AWS	6
適切なアクセス許可を持つ IAM ユーザーを作成する	7
AWS Command Line Interfaceのインストールと設定	7
入門	8
IAM ポリシーを作成する	8
コンソール RPC の例	9
awscurl RPC の例	10
Node.js RPC の例	11
PrivateLink を介した AMB アクセス Bitcoin	15
Bitcoin のユースケース	16
BTC を送受信するための Bitcoin (BTC) ウォレットを構築する	16
Bitcoin ブロックチェーンのアクティビティを分析する	16
Bitcoin キーペアを使用して署名されたメッセージを検証する	17
Bitcoin メモリプールの検査	17
Bitcoin JSON-RPCs	18
サポートされている JSON-RPCs	19
セキュリティ	23
データ保護	24
データ暗号化	25
転送中の暗号化	25
Identity and Access Management	25
対象者	26
アイデンティティを使用した認証	26
ポリシーを使用したアクセスの管理	30
Amazon Managed Blockchain (AMB) Access Bitcoin と IAM の連携	33
アイデンティティベースのポリシーの例	40
トラブルシューティング	44
CloudTrail ログ	47

CloudTrail での AMB Access Bitcoin 情報	47
AMB Access Bitcoin ログファイルエントリについて	48
CloudTrail を使用して Bitcoin JSON-RPCs	48
.....	li

Amazon Managed Blockchain (AMB) とは何ですか？

Amazon Managed Blockchain (AMB) Access は、Ethereum と Bitcoin 用のパブリックブロックチェーンノードを提供し、Hyperledger Fabric フレームワークを使用してプライベートブロックチェーンネットワークを作成することもできます。パブリックブロックチェーンノードへのフルマネージド、シングルテナント (専用)、サーバーレスマルチテナント API オペレーションなど、パブリックブロックチェーンを操作するさまざまな方法から選択します。アクセスコントロールが重要なユースケースでは、フルマネージドプライベートブロックチェーンネットワークから選択できます。標準化された API オペレーションは、フルマネージド型の回復力のあるインフラストラクチャで即時のスケラビリティを提供するため、ブロックチェーンアプリケーションを構築できます。

AMB Access は、マルチテナントブロックチェーンネットワークアクセス API オペレーションと専用のブロックチェーンノードとネットワークという 2 つの異なるタイプのブロックチェーンインフラストラクチャサービスを提供します。専用のブロックチェーンインフラストラクチャを使用すると、パブリック Ethereum ブロックチェーンノードとプライベート Hyperledger Fabric ブロックチェーンネットワークを作成して独自の用途に使用できます。ただし、AMB Access Bitcoin などのマルチテナントの API ベースのサービスは、基盤となるブロックチェーンノードインフラストラクチャが顧客間で共有される API レイヤーの背後にある Bitcoin ノードのフリートで構成されます。

Bitcoin は、ネットワークのネイティブ暗号通貨である Bitcoin (BTC) に設定された安全な peer-to-peer の価値トランザクションを可能にする分散型ブロックチェーンネットワークです。Bitcoin ネットワークは、個人、金融機関、フィンテック企業、政府などによって使用されます。Bitcoin ネットワークは、交換の媒体、投資のための商品、または受信データのための公開検証可能でイミュータブルな台帳です。Amazon Managed Blockchain (AMB) Access Bitcoin を使用すると、リージョンエンドポイントを介して Bitcoin Mainnet および Testnet ネットワークのプールにアクセスできます。これにより、トランザクションの書き込み、台帳からのデータの読み取り、Bitcoin Core ノードクライアントで利用可能な JSON-RPC リクエストの呼び出しを行うことができます。サーバーレス Bitcoin エンドポイントを使用すると、Bitcoin ノードのプロビジョニング、保守、負荷分散などの差別化されていない作業に投資するのではなく、アプリケーションの構築に集中できます。Bitcoin ウォレットの構築、暗号交換の構築、Bitcoin ブロックチェーンデータの分析のいずれであっても、Bitcoin エンドポイントを介して行ったリクエストに対してのみ AMB Access Bitcoin を使用して支払います。

AMB Access Bitcoin を初めてお使いになる方向けの情報

AMB Access Bitcoin を初めて使用する場合は、まず以下のセクションを読むことをお勧めします。

- [主な概念: Amazon Managed Blockchain \(AMB\) Access Bitcoin](#)

- [Amazon Managed Blockchain \(AMB\) Access Bitcoin の開始方法](#)
- [Amazon Managed Blockchain \(AMB\) Access Bitcoin を使用したビットコインのユースケース](#)
- [Amazon Managed Blockchain \(AMB\) Access Bitcoin でサポートされている Bitcoin JSON-RPCs](#)

主な概念: Amazon Managed Blockchain (AMB) Access Bitcoin

Note

このガイドでは、Bitcoin に不可欠な概念を理解していることを前提としています。これらの概念には、分散、ノード、トランザクション、proof-of-work、ウォレット、パブリックキーとプライベートキー、半減などが含まれます。Amazon Managed Blockchain (AMB) Access Bitcoin を使用する前に、[Bitcoin 開発ドキュメント](#)と [Mastering Bitcoin](#)を確認することをお勧めします。

Amazon Managed Blockchain (AMB) Access Bitcoin は、ノードを含む Bitcoin インフラストラクチャをプロビジョニングおよび管理することなく、Bitcoin ブロックチェーンへのサーバーレスアクセスを提供します。このマネージドサービスを使用して、Bitcoin ネットワークに迅速かつオンデマンドでアクセスできるため、全体的な所有コストを削減できます。

AMB Access Bitcoin は、Bitcoin Core クライアントを実行しているフルノードを介して Bitcoin ネットワークへのアクセスを提供します。ウォレット機能は無効になっており、複数の JSON リモートプロシージャ (JSON-RPC) 呼び出しをサポートしています。Bitcoin JSON RPCs を呼び出して Managed Blockchain が管理する Bitcoin ノードと通信し、Bitcoin ネットワークとやり取りできます。Bitcoin JSON-RPCs を使用すると、Amazon Managed Blockchain サービスを使用して、データのクエリや Bitcoin ネットワークへのトランザクションの送信など、データの読み取りとトランザクションの書き込みを行うことができます。

Important

Bitcoin アドレスの作成、保守、使用、管理はお客様の責任となります。また、Bitcoin アドレスの内容についても責任を負います。AWS は、Amazon Managed Blockchain で Bitcoin ノードを使用してデプロイまたは呼び出されたトランザクションについては責任を負いません。

Amazon Managed Blockchain (AMB) Access Bitcoin を使用する際の考慮事項と制限事項

• サポートされている Bitcoin ネットワーク

AMB Access Bitcoin は、次のパブリックネットワークをサポートしています。

- Mainnet — proof-of-workコンセンサスで保護され、Bitcoin (BTC) 暗号通貨が発行および取引されるパブリック Bitcoin ブロックチェーン。Mainnet のトランザクションには実際の値 (つまり、実際のコストが発生します) があり、パブリックブロックチェーンに記録されます。
- Testnet — testnet は、テストに使用される代替 Bitcoin ブロックチェーンです。Testnet コインは、実際の Bitcoin (BTC) とは別個で区別され、通常は値はありません。

Note

プライベートネットワークはサポートされていません。

• サポートされるリージョン

このサービスでサポートされているリージョンは次のとおりです。

リージョン名	Code	リージョン
米国東部 (バージニア北部)	IAD	us-east-1
アジアパシフィック (東京)	NRT	ap-northeast-1
アジアパシフィック (ソウル)	ICN	ap-northeast-2
アジアパシフィック (シンガポール)	SIN	ap-southeast-1
欧州 (アイルランド)	DUB	eu-west-1
欧州 (ロンドン)	LHR	eu-west-2

• サービスエンドポイント

AMB Access Bitcoin のサービスエンドポイントを次に示します。サービスと接続するには、サポートされているリージョンのいずれかを含むエンドポイントを使用する必要があります。

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

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

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

- マイニングはサポートされていません

AMB Access Bitcoin は Bitcoin (BTC) マイニングをサポートしていません。

- Bitcoin JSON-RPC 呼び出しの署名バージョン 4 の署名

Amazon Managed Blockchain で Bitcoin JSON-RPCs を呼び出す場合、[署名バージョン 4 の署名プロセス](#)を使用して認証された HTTPS 接続を介して呼び出すことができます。つまり、アカウント内の AWS 承認された IAM プリンシパルのみが Bitcoin JSON-RPC 呼び出しを行うことができます。これを行うには、AWS 認証情報 (アクセスキー ID とシークレットアクセスキー) を呼び出しで指定する必要があります。

 Important

- ユーザー向けアプリケーションにクライアント認証情報を埋め込まないでください。
- IAM ポリシーを使用して、個々の Bitcoin JSON-RPCs へのアクセスを制限することはできません。

- 未加工トランザクションの送信のみがサポートされています

`sendrawtransaction` JSON-RPC を使用して、Bitcoin ブロックチェーンの状態を更新するトランザクションを送信します。

- AWS CloudTrail ログ記録のサポート

Bitcoin JSON-RPCsをログに記録するように CloudTrail を設定できます。詳細については、[を使用した Amazon Managed Blockchain \(AMB\) アクセス Bitcoin イベントのログ記録 AWS CloudTrail](#)を参照してください。

Amazon Managed Blockchain (AMB) Access Bitcoin のセットアップ

Amazon Managed Blockchain (AMB) を初めて使用する前に、このセクションの手順に従って AWS アカウントを作成します。次の章では、AMB Access Bitcoin の使用を開始する方法について説明します。

前提条件と考慮事項

AWS を初めて使用する場合は、事前に [が必要で AWS アカウント](#)。

にサインアップする AWS

にサインアップすると AWS、Amazon Managed Blockchain (AMB) Access Bitcoin を含む AWS のサービスすべての [に](#) が自動的にサインアップ AWS アカウント されます。サービスを実際に使用した分の料金のみが請求されます。

AWS アカウント をすでにお持ちの場合は、次のステップに進みます。AWS アカウントをお持ちでない場合は、次に説明する手順に従ってアカウントを作成してください。

AWS アカウントを作成するには

1. <https://portal.aws.amazon.com/billing/signup> を開きます。
2. オンラインの手順に従います。

サインアップ手順の一部では、電話またはテキストメッセージを受信し、電話のキーパッドに検証コードを入力します。

にサインアップすると AWS アカウント、AWS アカウントのルートユーザー が作成されます。ルートユーザーには、アカウントのすべての AWS のサービス とリソースへのアクセス権があります。セキュリティベストプラクティスとして、ユーザーに管理アクセス権を割り当て、[ルートユーザーアクセスが必要なタスク](#)の実行にはルートユーザーのみを使用するようにしてください。

適切なアクセス許可を持つ IAM ユーザーを作成する

AMB Access Bitcoin を作成して使用するには、必要な Managed Blockchain アクションを許可するアクセス許可を持つ AWS Identity and Access Management (IAM) プリンシパル (ユーザーまたはグループ) が必要です。

Bitcoin JSON-RPC 呼び出しを実行できるのは IAM プリンシパルのみです。Amazon Managed Blockchain で Bitcoin JSON-RPCs を呼び出す場合、[署名バージョン 4 の署名プロセス](#)を使用して認証された HTTPS 接続を介して呼び出すことができます。つまり、アカウント内の AWS 承認された IAM プリンシパルのみが Bitcoin JSON-RPC 呼び出しを行うことができます。これを行うには、コールで AWS 認証情報 (アクセスキー ID とシークレットアクセスキー) を指定する必要があります。

IAM ユーザーを作成する方法については、[「アカウントでの IAM ユーザーの作成 AWS」](#)を参照してください。アクセス許可ポリシーをユーザーにアタッチする方法の詳細については、[「IAM ユーザーのアクセス許可の変更」](#)を参照してください。AMB Access Bitcoin を操作するためのアクセス許可をユーザーに付与するために使用できるアクセス許可ポリシーの例については、「」を参照してください。[Amazon Managed Blockchain \(AMB\) Access Bitcoin のアイデンティティベースのポリシーの例](#)。

AWS Command Line Interfaceのインストールと設定

まだインストールしていない場合は、最新のコマンドラインインターフェイス (CLI) AWS をインストールして、ターミナルの AWS リソースを操作します。詳細については、[「Installing or updating the latest version of the AWS CLI」](#)を参照してください。

Note

CLI アクセスには、アクセスキー ID とシークレットアクセスキーが必要です。長期のアクセスキーの代わりに一時的な認証情報をできるだけ使用します。一時的な認証情報には、アクセスキー ID、シークレットアクセスキー、および認証情報の失効を示すセキュリティトークンが含まれています。詳細については、IAM [ユーザーガイドの「AWS リソースでの一時的な認証情報」](#)の使用」を参照してください。

Amazon Managed Blockchain (AMB) Access Bitcoin の開始方法

このセクションのstep-by-stepのチュートリアルを使用して、Amazon Managed Blockchain (AMB) Access Bitcoin を使用してタスクを実行する方法について説明します。これらの例では、いくつかの前提条件を満たす必要があります。AMB Access Bitcoin を初めて使用する場合は、このガイドの「セットアップ」セクションを参照して、これらの前提条件を満たしていることを確認してください。詳細については、「[Amazon Managed Blockchain \(AMB\) Access Bitcoin のセットアップ](#)」を参照してください。

トピック

- [Bitcoin JSON-RPCs にアクセスするための IAM ポリシーを作成する](#)
- [を使用して AMB Access RPC エディタで Bitcoin リモートプロシージャコール \(RPC\) リクエストを行う AWS Management Console](#)
- [を使用して awscli で AMB アクセス Bitcoin JSON-RPC リクエストを作成する AWS CLI](#)
- [Node.js で Bitcoin JSON-RPC リクエストを行う](#)
- [で AMB Access Bitcoin を使用する AWS PrivateLink](#)

Bitcoin JSON-RPCs にアクセスするための IAM ポリシーを作成する

Bitcoin Mainnet と Testnet のパブリックエンドポイントにアクセスして JSON-RPC 呼び出しを行うには、Amazon Managed Blockchain (AMB) アクセス Bitcoin に適切な IAM アクセス許可を持つユーザー認証情報 (AWS_ACCESS_KEY_ID と AWS_SECRET_ACCESS_KEY) が必要です。AWS CLI がインストールされているターミナルで、次のコマンドを実行して、両方の Bitcoin エンドポイントにアクセスするための IAM ポリシーを作成します。

```
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

前の例では、Bitcoin Mainnet と Testnet の両方にアクセスできます。特定のエンドポイントにアクセスするには、次のActionコマンドを使用します。

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

ポリシーを作成したら、そのポリシーを IAM ユーザーのロールにアタッチして有効にします。で AWS Management Console、IAM サービスに移動し、IAM ユーザーに割り当てられたロールにポリシー AmazonManagedBlockchainBitcoinAccess をアタッチします。詳細については、[「ロールの作成」](#) および [「IAM ユーザーへの割り当て」](#) を参照してください。

を使用して AMB Access RPC エディタで Bitcoin リモートプロシージャコール (RPC) リクエストを行う AWS Management Console

AMB Access AWS Management Console を使用して、リモートプロシージャコール (RPCs) を編集して送信できます。これらの RPCs を使用すると、Bitcoin ネットワークでデータの読み取り、書き込み、トランザクションの送信を行うことができます。

Example

次の例は、getBlockRPC blockhash を使用して 00000000c937983704a73af28acdec37b049d214adbda81d7e2a3dd146f6ed09 に関する情報を取得

する方法を示しています。強調表示された変数を独自の入力に置き換えるか、リストされている他の RPC メソッドのいずれかを選択して、必要な関連する入力を入力します。

1. 「<https://console.aws.amazon.com/managedblockchain/>.com で Managed Blockchain コンソールを開きます。
2. RPC エディタを選択します。
3. リクエストセクションで、ブロックチェーンネットワーク **BITCOIN_MAINNET** として を選択します。
4. RPC メソッド **getblock** として を選択します。
5. ブロック番号 **00000000c937983704a73af28acdec37b049d214adbda81d7e2a3dd146f6ed09** としてを入力し、詳細度 **0** として を選択します。
6. 次に、RPC の送信を選択します。
7. このページの「レスポンス」セクションに結果が表示されます。その後、詳細な分析やアプリケーションのビジネスロジックでの使用のために、完全な raw トランザクションをコピーできます。

詳細については、[RPCs](#)」を参照してください。

を使用して awscurl で AMB アクセス Bitcoin JSON-RPC リクエストを作成する AWS CLI

Example

[署名バージョン 4 \(SigV4\)](#) を使用して IAM ユーザー認証情報を使用してリクエストに署名し、AMB Access Bitcoin エンドポイントに Bitcoin JSON-RPC 呼び出しを行います。[awscurl](#) コマンドラインツールは、SigV4 を使用して AWS サービスへのリクエストに署名するのに役立ちます。詳細については、「[awscurl README.md://www.com](#)」を参照してください。

オペレーティングシステムに適した方法を使用して awscurl をインストールします。macOS では、HomeBrew が推奨アプリケーションです。

```
brew install awscurl
```

AWS CLI を既にインストールして設定している場合は、IAM ユーザー認証情報とデフォルトの AWS リージョンが環境に設定され、awscurl にアクセスできます。awscurl を使用して、getblockRPC

Example

このサンプル Node.js スクリプトを実行するには、次の前提条件を適用します。

1. マシンにはノードバージョンマネージャー (nvm) と Node.js がインストールされている必要があります。OS のインストール手順については、[こちらを参照してください](#)。
2. `node --version` コマンドを使用して、Node バージョン 14 以降を使用していることを確認します。必要に応じて、`nvm install 14` コマンドの後に `nvm use 14` コマンドを使用して、バージョン 14 をインストールできます。
3. 環境変数 `AWS_ACCESS_KEY_ID` と `AWS_SECRET_ACCESS_KEY` には、アカウントに関連付けられている認証情報が含まれています。環境変数には、AMB Access Bitcoin エンドポイントが含まれている `AMB_HTTP_ENDPOINT` が必要です。

次のコマンドを使用して、これらの変数をクライアントで文字列としてエクスポートします。次の文字列の強調表示された値を、IAM ユーザーアカウントの適切な値に置き換えます。

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

すべての前提条件を完了したら、エディタを使用して、次の `package.json` ファイルと `index.js` スクリプトをローカル環境にコピーします。

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();
```

前のサンプルコードでは、Axios を使用して Bitcoin エンドポイントに RPC リクエストを行い、公式の AWS SDK v3 ツールを使用して、適切な署名バージョン 4 (SigV4) ヘッダーでそれらのリクエストに署名しています。コードを実行するには、ファイルと同じディレクトリでターミナルを開き、以下を実行します。

```
npm i
node index.js
```

生成される結果は、次のようになります。

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

[illegible]

 Note

前のスクリプトのサンプルリクエストは、[を使用して awscli で AMB アクセス Bitcoin JSON-RPC リクエストを作成する AWS CLI](#)例と同じ入力パラメータブロックハッシュを使用してgetblock呼び出しを行います。他の呼び出しを行うには、スクリプト内の rpc オブジェクトを別の Bitcoin JSON-RPC で変更します。ホストプロパティオプションを Bitcoin に変更testnetして、そのエンドポイントで呼び出しを行うことができます。

で AMB Access Bitcoin を使用する AWS PrivateLink

AWS PrivateLink は、VPC 内にあるかのように、VPC を サービスにプライベートに接続するために使用できる、可用性が高くスケーラブルなテクノロジーです。プライベートサブネットからサービスと通信するために、インターネットゲートウェイ、NAT デバイス、パブリック IP アドレス、AWS Direct Connect 接続、または AWS Site-to-Site VPN 接続を使用する必要はありません。AWS PrivateLink または のセットアップの詳細については AWS PrivateLink、[「とは」を参照してください AWS PrivateLink。](#)

VPC エンドポイント AWS PrivateLink を使用して、経由で Bitcoin JSON-RPC リクエストを AMB Access Bitcoin に送信できます。このプライベートエンドポイントへのリクエストはオープンインターネット経由で渡されないため、同じ SigV4 認証を使用して Bitcoin エンドポイントに直接リクエストを送信できます。詳細については、「[を通じて AWS のサービスにアクセスする AWS PrivateLink](#)」を参照してください。

サービス名については、AWS サービス列で Amazon Managed Blockchain を探します。詳細については、「[AWS と統合する のサービス AWS PrivateLink](#)」を参照してください。エンドポイントのサービス名は、 の形式になります `com.amazonaws.AWS-REGION.managedblockchain.bitcoin.NETWORK-TYPE`。

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

Amazon Managed Blockchain (AMB) Access Bitcoin を使用したビットコインのユースケース

このトピックでは、AMB Access Bitcoin のユースケースを一覧表示します。

トピック

- [BTC を送受信するための Bitcoin \(BTC\) ウォレットを構築する](#)
- [Bitcoin ブロックチェーンのアクティビティを分析する](#)
- [Bitcoin キーペアを使用して署名されたメッセージを検証する](#)
- [Bitcoin メモリプールの検査](#)

BTC を送受信するための Bitcoin (BTC) ウォレットを構築する

Bitcoin ネットワーク上のネイティブ暗号通貨である BTC は、ネットワークのセキュリティモデルに不可欠なコンポーネントとして機能します。また、機関、企業、個人によって広く使用されている商品および交換手段としても機能します。したがって、多くのウォレットアプリケーションは Bitcoin ブロックチェーンとやり取りするために Bitcoin ノードに依存しています。これらのアプリケーションは、特定のアドレスセットの未処理の出力 (UTXOs) のバランスを計算し、トランザクションに署名して Bitcoin ネットワークに送信し、履歴トランザクションに関するデータを取得します。

以下は、Amazon Managed Blockchain (AMB) Access Bitcoin が BTC ウォレットトランザクションでサポートする Bitcoin JSON-RPCs のサンプルです。

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

詳細については、「[サポートされている JSON-RPCs](#)」を参照してください。

Bitcoin ブロックチェーンのアクティビティを分析する

JSON-RPC メソッドを使用して、Bitcoin `getchaintxstats` ブロックチェーン上のトランザクションアクティビティのボリュームを分析できます。この JSON-RPC を使用すると、1 秒あたりの平均

トランザクション率、合計トランザクション数、ブロック数などのメトリクスにアクセスできます。必要に応じて、ブロック番号またはブロックハッシュのウィンドウを区切り文字として定義して、ネットワーク内の特定のブロックセットの統計を計算することもできます。

詳細については、「[サポートされている JSON-RPCs](#)」を参照してください。

Bitcoin キーペアを使用して署名されたメッセージを検証する

ビットコインウォレットには、キーペアを構成するプライベートキーとパブリックキーがあります。これらのキーは、トランザクションに署名し、ブロックチェーン上のユーザーの ID として機能するために使用されます。パブリックキーは、標準化された英数字識別子 (27 ~ 34 文字) であるアドレスを作成するために使用されます。これらのアドレスは、BTC 出力を受信し、トランザクションまたはメッセージを処理するために使用します。

Bitcoin ウォレットを使用すると、ユーザーはメッセージを暗号的に署名して検証することもできます。このプロセスは、特定のウォレットアドレスとそれに関連付けられた BTC の所有権を証明するためによく使用されます。verifymessage Bitcoin JSON-RPC を使用すると、別のウォレットによって署名されたメッセージの信頼性と有効性を確認できます。具体的には、Bitcoin ノードを使用して、署名されたメッセージ自体内の指定されたパブリックキー派生アドレスに対応するプライベートキーを使用してメッセージが署名されているかどうかを確認できます。

詳細については、「[サポートされている JSON-RPCs](#)」を参照してください。

Bitcoin メモリプールの検査

多くのアプリケーションは、保留中のトランザクションを追跡したり、保留中のすべてのトランザクションのリストを取得したり、トランザクションの発信元を確認したりするために、mempool にアクセスする必要があります。これを行うには、このアクティビティgetrawmempoolをサポートするgetmempoolancestors、getmempoolentry、などの Bitcoin JSON-RPCs があります。これらの Bitcoin JSON-RPCs アプリケーションが mempool から必要な情報を取得するのに役立ちます。

Amazon Managed Blockchain (AMB) Access Bitcoin は testmempoolaccept Bitcoin JSON-RPCs もサポートしています。これにより、トランザクションがプロトコルルールを満たし、送信前にノードによって受け入れられるかどうかを確認できます。Bitcoin ブロックチェーンにトランザクションを直接送信するウォレット、エクスチェンジ、およびその他のエンティティは、これらの Bitcoin JSON-RPCs を使用します。

詳細については、「[サポートされている JSON-RPCs](#)」を参照してください。

Amazon Managed Blockchain (AMB) Access Bitcoin でサポートされている Bitcoin JSON-RPCs

このトピックでは、Managed Blockchain がサポートする Bitcoin JSON-RPCs のリストとリファレンスについて説明します。サポートされている各 JSON-RPC には、その使用に関する簡単な説明があります。

Note

- [署名バージョン 4 \(SigV4\) 署名プロセス](#)を使用して、マネージドブロックチェーンで Bitcoin JSON-RPCs を認証できます。つまり、AWS Bitcoin JSON-RPCs を使用してアカウント内の承認された IAM プリンシパルのみが操作できます。呼び出しで AWS 認証情報 (アクセスキー ID とシークレットアクセスキー) を指定します。
- HTTP レスポンスが 10 MB を超える場合は、エラーが発生します。これを修正するには、圧縮ヘッダーを に設定する必要があります `Accept-Encoding: gzip`。次に、クライアントが受け取る圧縮レスポンスには、ヘッダー `Content-Type: application/json` とが含まれます `Content-Encoding: gzip`。
- Amazon Managed Blockchain (AMB) Access Bitcoin は、不正な形式の JSON-RPC リクエストに対して 400 エラーを生成します。
- `sendrawtransaction` JSON-RPC を使用して、Bitcoin ブロックチェーンの状態を更新するトランザクションを送信します。
- AMB Access Bitcoin のデフォルトのリクエスト制限は、リージョンごとに 1 秒あたり 100 リクエスト (RPS) NETWORK_TYPE です AWS 。

クォータを増やすには、AWS サポートに連絡する必要があります。AWS サポートに連絡するには、[AWS サポートセンターコンソール](#)にサインインします。[ケースを作成] を選択します。[技術] を選択します。サービスとして Managed Blockchain を選択します。カテゴリとして Access:Bitcoin を選択し、重要度として一般的なガイダンスを選択します。RPC クォータをサブジェクトとして入力し、説明テキストボックスに、リージョンごとのビットコインネットワークあたりの RPS のニーズに適用されるクォータ制限を一覧表示します。ケースを送信します。

サポートされている JSON-RPCs

AMB Access Bitcoin は、次の Bitcoin JSON-RPCsをサポートしています。サポートされている各呼び出しには、その使用に関する簡単な説明があります。

カテゴリ	JSON-RPC	説明
ブロックチェーンRPCs	getbestblockhash	最も機能し、完全に検証されたチェーン内の最適な (ヒント) ブロックのハッシュを返します。
	getblock	詳細度が 0 の場合、 はブロック「ハッシュ」のシリアル化された 16 進エンコードされたデータの文字列を返します。詳細度が 1 の場合、 はブロック「ハッシュ」に関する情報を含む オブジェクトを返します。詳細度が 2 の場合、 はブロック「ハッシュ」に関する情報と各トランザクションに関する情報を含む オブジェクトを返します。詳細度が 3 の場合、 はブロック「ハッシュ」に関する情報と、入力の情報を含む各トランザクションprevoutに関する情報を含む オブジェクトを返します。
	getblockchaininfo	ブロックチェーン処理に関するさまざまな状態情報を含むオブジェクトを返します。
	getblockcount	最も機能し、完全に検証されたチェーンの高さを返します。生成ブロックの高さは 0 です。
	getblockfilter	ブロックハッシュを使用して、特定のブロックの BIP 157 コンテンツフィルターを取得します。
	getblockhash	指定された高さでbest-block-chainのハッシュを返します。
	getblockheader	詳細が false の場合、 はブロックヘッダーの「ハッシュ」に対してシリアル化された 16 進

カテゴリ	JSON-RPC	説明
		エンコードされたデータの文字列を返します。詳細が true の場合、 はブロックヘッダーの「ハッシュ」に関する情報を含む オブジェクトを返します。
	getblockstats	特定のウィンドウのブロックごとの統計を計算します。すべての金額はサトシです。プルーニングでは一部の高さでは機能しません。
	getchaintips	メインチェーンや孤立したブランチなど、ブロックツリー内のすべての既知のヒントに関する情報を返します。
	getchaintxstats	チェーン内のトランザクションの合計数とレートに関する統計を計算します。
	getdifficulty	proof-of-work難易度を最小難易度の倍数として返します。
	getmempoolancestors	txid が mempool にある場合、 はすべてのインメモリプールの祖先を返します。
	getmempool の子孫	txid が mempool にある場合、 はすべてのインメモリプールの子孫を返します。
	getmempoolentry	特定のトランザクションの mempool データを返します。
	getmempoolinfo	TX メモリプールのアクティブ状態に関する詳細を返します。

カテゴリ	JSON-RPC	説明
rawtransactions RPCs	getrawmempool	メモリプール内のすべてのトランザクション IDs 文字列 トランザクション IDs。  Note verbose = true はサポートされていません。
	gettxout	未使用のトランザクション出力に関する詳細を返します。
	gettxoutproof	「txid」がブロックに含まれていたことを示す 16 進エンコードされた証明を返します。
	createrawtransaction	指定された入力を使用するトランザクションを作成し、新しい出力を作成します。
	デコードraw トランザクション	シリアル化された 16 進エンコードされたトランザクションを表す JSON オブジェクトを返します。
	デコードスクリプト	16 進エンコードされたスクリプトをデコードします。
	getrawtransaction	raw トランザクションデータを返します。
	sendrawtransaction	raw トランザクション (シリアル化、16 進エンコード) をローカルノードとネットワークに送信します。
	testmempoolaccept	未加工トランザクション (シリアル化、16 進エンコード) が mempool によって受け入れられるかどうかを示す mempool 受け入れテストの結果を返します。これにより、トランザクションがコンセンサスルールまたはポリシールールに違反しているかどうかを確認します。

カテゴリ	JSON-RPC	説明
RPCs の使用	マルチシグの作成	必要な m キーの n 個の署名を持つマルチ署名アドレスを作成します。
	見積り マートフィー	トランザクションが conf_target ブロック内で確認を開始するために必要な概算料金を推定し、見積りが有効であるブロック数を返します。BIP 141 で定義されている仮想トランザクションサイズを使用します (立会人データは割引されます)。
	検証アドレス	指定されたビットコインアドレスに関する情報を返します。
	検証メッセージ	署名されたメッセージを検証します。

Amazon Managed Blockchain (AMB) のセキュリティが Bitcoin にアクセスする

でのクラウドセキュリティ AWS は最優先事項です。お客様は AWS、最もセキュリティの影響を受けやすい組織の要件を満たすように構築されたデータセンターとネットワークアーキテクチャを活用できます。

セキュリティは、AWS とユーザーの間で共有される責任です。責任[共有モデル](#)では、これをクラウドのセキュリティとクラウド内のセキュリティの両方として説明しています。

- クラウドのセキュリティ – AWS は、で AWS サービスを実行するインフラストラクチャを保護する責任を担います AWS クラウド。AWS また、では、安全に使用できるサービスも提供しています。サードパーティーの監査人は、[AWS コンプライアンスプログラム](#)の一環として、セキュリティの有効性を定期的にテストおよび検証します。Amazon Managed Blockchain (AMB) Access Bitcoin に適用されるコンプライアンスプログラムについては、[AWS 「コンプライアンスプログラムによる対象範囲内のサービス」](#)を参照してください。
- クラウド内のセキュリティ – お客様の責任は、使用する AWS サービスによって決まります。また、お客様は、お客様のデータの機密性、企業の要件、および適用可能な法律や規制といった他の要因 についても責任を担います。

データ保護、認証、アクセス制御を提供するために、Amazon Managed Blockchain は AWS Managed Blockchain で実行されているオープンソースフレームワークの機能を使用します。

このドキュメントは、AMB Access Bitcoin を使用する際の責任共有モデルの適用方法を理解するのに役立ちます。以下のトピックでは、セキュリティとコンプライアンスの目標を達成するために AMB Access Bitcoin を設定する方法について説明します。また、AMB Access Bitcoin リソースのモニタリングや保護に役立つ他の AWS サービスの使用方法についても説明します。

トピック

- [Amazon Managed Blockchain \(AMB\) Access Bitcoin でのデータ保護](#)
- [Amazon Managed Blockchain \(AMB\) Access Bitcoin の Identity and Access Management](#)

Amazon Managed Blockchain (AMB) Access Bitcoin でのデータ保護

責任 AWS [共有モデル](#)、Amazon Managed Blockchain (AMB) Access Bitcoin でのデータ保護に適用されます。このモデルで説明されているように、AWS はすべての を実行するグローバルインフラストラクチャを保護する責任があります AWS クラウド。ユーザーは、このインフラストラクチャでホストされるコンテンツに対する管理を維持する責任があります。また、使用する「AWS のサービス」のセキュリティ設定と管理タスクもユーザーの責任となります。データプライバシーの詳細については、[データプライバシーに関するよくある質問](#)を参照してください。欧州でのデータ保護の詳細については、AWS セキュリティブログに投稿された [AWS 責任共有モデルおよび GDPR](#) のブログ記事を参照してください。

データ保護の目的で、認証情報を保護し AWS アカウント、AWS IAM Identity Center または AWS Identity and Access Management (IAM) を使用して個々のユーザーを設定することをお勧めします。この方法により、それぞれのジョブを遂行するために必要な権限のみが各ユーザーに付与されます。また、次の方法でデータを保護することもお勧めします：

- 各アカウントで多要素認証 (MFA) を使用します。
- SSL/TLS を使用して AWS リソースと通信します。TLS 1.2 が必須で、TLS 1.3 をお勧めします。
- で API とユーザーアクティビティのログ記録を設定します AWS CloudTrail。CloudTrail 証跡を使用して AWS アクティビティをキャプチャする方法については、「AWS CloudTrail ユーザーガイド」の [CloudTrail 証跡の使用](#) を参照してください。
- AWS 暗号化ソリューションと、その中のすべてのデフォルトのセキュリティコントロールを使用します AWS のサービス。
- Amazon Macie などの高度な管理されたセキュリティサービスを使用します。これらは、Amazon S3 に保存されている機密データの検出と保護を支援します。
- コマンドラインインターフェイスまたは API AWS を介して にアクセスするときに FIPS 140-3 検証済み暗号化モジュールが必要な場合は、FIPS エンドポイントを使用します。利用可能な FIPS エンドポイントの詳細については、「[連邦情報処理規格 \(FIPS\) 140-3](#)」を参照してください。

お客様の E メールアドレスなどの極秘または機密情報を、タグ、または [名前] フィールドなどの自由形式のテキストフィールドに含めないことを強くお勧めします。これは、コンソール、API、または SDK を使用して AMB Access Bitcoin AWS CLI または他の AWS のサービス を使用する場合も同様です。AWS SDKs タグ、または名前に使用される自由記述のテキストフィールドに入力したデータは、請求または診断ログに使用される場合があります。外部サーバーに URL を提供する場合、そ

のサーバーへのリクエストを検証できるように、認証情報を URL に含めないことを強くお勧めします。

データ暗号化

データ暗号化は、権限のないユーザーがブロックチェーンネットワークおよび関連するデータストレージシステムからデータを読み取るのを防ぐのに役立ちます。これには、ネットワークを移動するときに傍受される可能性のあるデータが含まれます。これは転送中のデータと呼ばれます。

転送中の暗号化

デフォルトでは、Managed Blockchain は HTTPS/TLS 接続を使用して、を実行するクライアントコンピュータから AWS サービスエンドポイントに送信されるすべてのデータを暗号化します AWS CLI。

HTTPS/TLS の使用を有効にするために必要な操作はありません。コマンドを使用して個々の AWS CLI コマンドに対して明示的に無効にしない限り、常に有効になります `--no-verify-ssl`。

Amazon Managed Blockchain (AMB) Access Bitcoin の Identity and Access Management

AWS Identity and Access Management (IAM) は、管理者が AWS リソースへのアクセスを安全に制御 AWS のサービス するのに役立つ です。IAM 管理者は、誰を認証 (サインイン) し、誰に AMB Access Bitcoin リソースの使用を許可する (アクセス許可を付与する) かを制御します。IAM は、追加料金なしで AWS のサービス 使用できる です。

トピック

- [対象者](#)
- [アイデンティティを使用した認証](#)
- [ポリシーを使用したアクセスの管理](#)
- [Amazon Managed Blockchain \(AMB\) Access Bitcoin と IAM の連携](#)
- [Amazon Managed Blockchain \(AMB\) Access Bitcoin のアイデンティティベースのポリシーの例](#)
- [Amazon Managed Blockchain \(AMB\) Access Bitcoin のアイデンティティとアクセスのトラブルシューティング](#)

対象者

AWS Identity and Access Management (IAM) の使用方法は、AMB Access Bitcoin で行う作業によって異なります。

サービスユーザー – ジョブを実行するために AMB Access Bitcoin サービスを使用する場合、管理者から必要な認証情報とアクセス許可が与えられます。さらに多くの AMB Access Bitcoin 機能を使用して作業を行う場合は、追加のアクセス許可が必要になる場合があります。アクセスの管理方法を理解すると、管理者に適切なアクセス許可をリクエストするのに役に立ちます。AMB Access Bitcoin の機能にアクセスできない場合は、「」を参照してください[Amazon Managed Blockchain \(AMB\) Access Bitcoin のアイデンティティとアクセスのトラブルシューティング](#)。

サービス管理者 – 社内の AMB Access Bitcoin リソースを担当している場合は、通常、AMB Access Bitcoin へのフルアクセスがあります。サービスユーザーがどの AMB Access Bitcoin 機能とリソースにアクセスするかを決めるのは管理者の仕事です。その後、IAM 管理者にリクエストを送信して、サービスユーザーの権限を変更する必要があります。このページの情報を点検して、IAM の基本概念を理解してください。会社で AMB Access Bitcoin で IAM を使用方法の詳細については、「」を参照してください[Amazon Managed Blockchain \(AMB\) Access Bitcoin と IAM の連携](#)。

IAM 管理者 - IAM 管理者は、AMB Access Bitcoin へのアクセスを管理するポリシーの作成方法の詳細について確認する場合があります。IAM で使用できる AMB Access Bitcoin アイデンティティベースのポリシーの例を表示するには、「」を参照してください[Amazon Managed Blockchain \(AMB\) Access Bitcoin のアイデンティティベースのポリシーの例](#)。

アイデンティティを使用した認証

認証は、ID 認証情報 AWS を使用して にサインインする方法です。として、IAM ユーザーとして AWS アカウントのルートユーザー、または IAM ロールを引き受けることによって、認証 (にサインイン AWS) される必要があります。

ID ソースを介して提供された認証情報を使用して、フェデレーティッド ID AWS として にサインインできます。AWS IAM Identity Center (IAM Identity Center) ユーザー、会社のシングルサインオン認証、Google または Facebook 認証情報は、フェデレーティッド ID の例です。フェデレーティッド ID としてサインインする場合、IAM ロールを使用して、前もって管理者により ID フェデレーションが設定されています。フェデレーション AWS を使用して にアクセスすると、間接的にロールを引き受けることになります。

ユーザーの種類に応じて、AWS Management Console または AWS アクセスポータルにサインインできます。へのサインインの詳細については AWS、AWS サインイン ユーザーガイドの[「へのサインイン方法 AWS アカウント」](#)を参照してください。

AWS プログラムで にアクセスする場合、 はソフトウェア開発キット (SDK) とコマンドラインインターフェイス (CLI) AWS を提供し、認証情報を使用してリクエストに暗号で署名します。AWS ツールを使用しない場合は、自分でリクエストに署名する必要があります。リクエストに自分で署名する推奨方法の使用については、「IAM ユーザーガイド」の「[API リクエストに対するAWS Signature Version 4](#)」を参照してください。

使用する認証方法を問わず、追加セキュリティ情報の提供をリクエストされる場合もあります。例えば、では、多要素認証 (MFA) を使用してアカウントのセキュリティを向上させる AWS ことをお勧めします。詳細については、「AWS IAM Identity Center ユーザーガイド」の「[多要素認証](#)」および「IAM ユーザーガイド」の「[IAM のAWS 多要素認証](#)」を参照してください。

AWS アカウント ルートユーザー

を作成するときは AWS アカウント、アカウント内のすべての およびリソースへの AWS のサービス 完全なアクセス権を持つ 1 つのサインインアイデンティティから始めます。この ID は AWS アカウント ルートユーザーと呼ばれ、アカウントの作成に使用した E メールアドレスとパスワードでサインインすることでアクセスできます。日常的なタスクには、ルートユーザーを使用しないことを強くお勧めします。ルートユーザーの認証情報は保護し、ルートユーザーでしか実行できないタスクを実行するときに使用します。ルートユーザーとしてサインインする必要があるタスクの完全なリストについては、「IAM ユーザーガイド」の「[ルートユーザー認証情報が必要なタスク](#)」を参照してください。

フェデレーティッドアイデンティティ

ベストプラクティスとして、管理者アクセスを必要とするユーザーを含む人間のユーザーに、一時的な認証情報を使用して にアクセスするために ID プロバイダーとのフェデレーション AWS のサービスを使用することを要求します。

フェデレーティッド ID は、エンタープライズユーザーディレクトリ、ウェブ ID プロバイダー、AWS Directory Serviceアイデンティティセンターディレクトリ、または ID ソースを介して提供された認証情報 AWS のサービスを使用して にアクセスするすべてのユーザーです。フェデレーティッド ID がアクセスすると AWS アカウント、ロールを引き受け、ロールは一時的な認証情報を提供します。

アクセスを一元管理する場合は、AWS IAM Identity Centerを使用することをお勧めします。IAM Identity Center でユーザーとグループを作成するか、独自の ID ソースのユーザーとグループのセットに接続して同期し、すべての AWS アカウント とアプリケーションで使用できます。IAM Identity Center の詳細については、「AWS IAM Identity Center ユーザーガイド」の「[What is IAM Identity Center?](#)」(IAM Identity Center とは) を参照してください。

IAM ユーザーとグループ

[IAM ユーザー](#)は、1 人のユーザーまたはアプリケーションに対して特定のアクセス許可 AWS アカウントを持つ 内の ID です。可能であれば、パスワードやアクセスキーなどの長期的な認証情報を保有する IAM ユーザーを作成する代わりに、一時的な認証情報を使用することをお勧めします。ただし、IAM ユーザーでの長期的な認証情報が必要な特定のユースケースがある場合は、アクセスキーをローテーションすることをお勧めします。詳細については、「IAM ユーザーガイド」の「[長期的な認証情報を必要とするユースケースのためにアクセスキーを定期的にローテーションする](#)」を参照してください。

[IAM グループ](#)は、IAM ユーザーの集団を指定するアイデンティティです。グループとしてサインインすることはできません。グループを使用して、複数のユーザーに対して一度に権限を指定できます。多数のユーザーグループがある場合、グループを使用することで権限の管理が容易になります。例えば、IAMAdmins という名前のグループを設定して、そのグループに IAM リソースを管理する許可を与えることができます。

ユーザーは、ロールとは異なります。ユーザーは 1 人の人または 1 つのアプリケーションに一意に関連付けられますが、ロールはそれを必要とする任意の人が引き受けるようになっています。ユーザーには永続的な長期の認証情報がありますが、ロールでは一時認証情報が提供されます。詳細については、「IAM ユーザーガイド」の「[IAM ユーザーに関するユースケース](#)」を参照してください。

IAM ロール

[IAM ロール](#)は、特定のアクセス許可 AWS アカウントを持つ 内の ID です。これは IAM ユーザーに似ていますが、特定のユーザーには関連付けられていません。で IAM ロールを一時的に引き受けるには AWS Management Console、[ユーザーから IAM ロール \(コンソール\) に切り替える](#)ことができます。ロールを引き受けるには、または AWS API オペレーションを AWS CLI 呼び出すか、カスタム URL を使用します。ロールを使用する方法の詳細については、「IAM ユーザーガイド」の「[ロールを引き受けるための各種方法](#)」を参照してください。

IAM ロールと一時的な認証情報は、次の状況で役立ちます:

- フェデレーションユーザーアクセス – フェデレーティッド ID に許可を割り当てるには、ロールを作成してそのロールの許可を定義します。フェデレーティッド ID が認証されると、その ID はロールに関連付けられ、ロールで定義されている許可が付与されます。フェデレーションのロールについては、「IAM ユーザーガイド」の「[サードパーティー ID プロバイダー \(フェデレーション\) 用のロールを作成する](#)」を参照してください。IAM Identity Center を使用する場合は、許可セットを設定します。アイデンティティが認証後にアクセスできるものを制御するため、IAM Identity

Center は、権限セットを IAM のロールに関連付けます。アクセス許可セットの詳細については、「AWS IAM Identity Center User Guide」の「[Permission sets](#)」を参照してください。

- 一時的な IAM ユーザー権限 - IAM ユーザーまたはロールは、特定のタスクに対して複数の異なる権限を一時的に IAM ロールで引き受けることができます。
- クロスアカウントアクセス - IAM ロールを使用して、自分のアカウントのリソースにアクセスすることを、別のアカウントの人物 (信頼済みプリンシパル) に許可できます。クロスアカウントアクセス権を付与する主な方法は、ロールを使用することです。ただし、一部の では AWS のサービス、(プロキシとしてロールを使用する代わりに) リソースに直接ポリシーをアタッチできます。クロスアカウントアクセスにおけるロールとリソースベースのポリシーの違いについては、「IAM ユーザーガイド」の「[IAM でのクロスアカウントのリソースへのアクセス](#)」を参照してください。
- クロスサービスアクセス — 一部の は他の の機能 AWS のサービス を使用します AWS のサービス。例えば、あるサービスで呼び出しを行うと、通常そのサービスによって Amazon EC2 でアプリケーションが実行されたり、Amazon S3 にオブジェクトが保存されたりします。サービスでは、呼び出し元プリンシパルの許可、サービスロール、またはサービスリンクロールを使用してこれを行う場合があります。
- 転送アクセスセッション (FAS) – IAM ユーザーまたはロールを使用してアクションを実行すると AWS、プリンシパルと見なされます。一部のサービスを使用する際に、アクションを実行することで、別のサービスの別のアクションがトリガーされることがあります。FAS は、 を呼び出すプリンシパルのアクセス許可と AWS のサービス、ダウンストリームサービス AWS のサービスへのリクエストのリクエストをリクエストする を使用します。FAS リクエストは、サービスが他の AWS のサービス またはリソースとのやり取りを完了する必要があるリクエストを受け取った場合にのみ行われます。この場合、両方のアクションを実行するためのアクセス許可が必要です。FAS リクエストを行う際のポリシーの詳細については、「[転送アクセスセッション](#)」を参照してください。
- サービスロール - サービスがユーザーに代わってアクションを実行するために引き受ける [IAM ロール](#)です。IAM 管理者は、IAM 内からサービスロールを作成、変更、削除することができます。詳細については、「IAM ユーザーガイド」の「[AWS のサービスに許可を委任するロールを作成する](#)」を参照してください。
- サービスにリンクされたロール – サービスにリンクされたロールは、 にリンクされたサービスロールの一種です AWS のサービス。サービスは、ユーザーに代わってアクションを実行するロールを引き受けることができます。サービスにリンクされたロールは に表示され AWS アカウント、 サービスによって所有されます。IAM 管理者は、サービスリンクロールのアクセス許可を表示できますが、編集することはできません。

- Amazon EC2 で実行されているアプリケーション – IAM ロールを使用して、EC2 インスタンスで実行され、AWS CLI または AWS API リクエストを行うアプリケーションの一時的な認証情報を管理できます。これは、EC2 インスタンス内でのアクセスキーの保存に推奨されます。AWS ロールを EC2 インスタンスに割り当て、そのすべてのアプリケーションで使えるようにするには、インスタンスにアタッチされたインスタンスプロファイルを作成します。インスタンスプロファイルにはロールが含まれ、EC2 インスタンスで実行されるプログラムは一時的な認証情報を取得できます。詳細については、「IAM ユーザーガイド」の「[Amazon EC2 インスタンスで実行されるアプリケーションに IAM ロールを使用して許可を付与する](#)」を参照してください。

ポリシーを使用したアクセスの管理

でアクセスを制御する AWS には、ポリシーを作成し、ID AWS またはリソースにアタッチします。ポリシーは AWS、アイデンティティまたはリソースに関連付けられているときにアクセス許可を定義するオブジェクトです。は、プリンシパル(ユーザー、ルートユーザー、またはロールセッション) がリクエストを行うときに、これらのポリシー AWS を評価します。ポリシーでの権限により、リクエストが許可されるか拒否されるかが決まります。ほとんどのポリシーは JSON ドキュメント AWS として保存されます。JSON ポリシードキュメントの構造と内容の詳細については、IAM ユーザーガイドの [JSON ポリシー概要](#)を参照してください。

管理者は JSON AWS ポリシーを使用して、誰が何にアクセスできるかを指定できます。つまり、どのプリンシパルがどのリソースに対してどのような条件下でアクションを実行できるかということです。

デフォルトでは、ユーザーやロールに権限はありません。IAM 管理者は、リソースに必要なアクションを実行するための権限をユーザーに付与する IAM ポリシーを作成できます。その後、管理者はロールに IAM ポリシーを追加し、ユーザーはロールを引き受けることができます。

IAM ポリシーは、オペレーションの実行方法を問わず、アクションの許可を定義します。例えば、iam:GetRole アクションを許可するポリシーがあるとします。そのポリシーを持つユーザーは、AWS Management Console、AWS CLI または AWS API からロール情報を取得できます。

アイデンティティベースのポリシー

アイデンティティベースポリシーは、IAM ユーザーグループ、ユーザーのグループ、ロールなど、アイデンティティにアタッチできる JSON 許可ポリシードキュメントです。これらのポリシーは、ユーザーとロールが実行できるアクション、リソース、および条件をコントロールします。アイデンティティベースポリシーの作成方法については、「IAM ユーザーガイド」の「[カスタム IAM アクセス許可を定義する](#)」を参照してください。

アイデンティティベースのポリシーは、さらにインラインポリシーまたはマネージドポリシーに分類できます。インラインポリシーは、単一のユーザー、グループ、またはロールに直接埋め込まれています。管理ポリシーは、内の複数のユーザー、グループ、ロールにアタッチできるスタンドアロンポリシーです AWS アカウント。管理ポリシーには、AWS 管理ポリシーとカスタマー管理ポリシーが含まれます。マネージドポリシーまたはインラインポリシーのいずれかを選択する方法については、「IAM ユーザーガイド」の「[管理ポリシーとインラインポリシーのいずれかを選択する](#)」を参照してください。

リソースベースのポリシー

リソースベースのポリシーは、リソースに添付する JSON ポリシードキュメントです。リソースベースのポリシーには例として、IAM ロールの信頼ポリシーや Amazon S3 バケットポリシーがあげられます。リソースベースのポリシーをサポートするサービスでは、サービス管理者はポリシーを使用して特定のリソースへのアクセスを制御できます。ポリシーがアタッチされているリソースの場合、指定されたプリンシパルがそのリソースに対して実行できるアクションと条件は、ポリシーによって定義されます。リソースベースのポリシーでは、[プリンシパルを指定する](#)必要があります。プリンシパルには、アカウント、ユーザー、ロール、フェデレーティッドユーザー、またはを含めることができます AWS のサービス。

リソースベースのポリシーは、そのサービス内にあるインラインポリシーです。リソースベースのポリシーでは、IAM の AWS マネージドポリシーを使用できません。

アクセスコントロールリスト (ACL)

アクセスコントロールリスト (ACL) は、どのプリンシパル (アカウントメンバー、ユーザー、またはロール) がリソースにアクセスするための許可を持つかを制御します。ACL はリソースベースのポリシーに似ていますが、JSON ポリシードキュメント形式は使用しません。

Amazon S3、AWS WAF、および Amazon VPC は、ACLs。ACL の詳細については、「Amazon Simple Storage Service デベロッパーガイド」の「[アクセスコントロールリスト \(ACL\) の概要](#)」を参照してください。

その他のポリシータイプ

AWS は、一般的でない追加のポリシータイプをサポートします。これらのポリシータイプでは、より一般的なポリシータイプで付与された最大の権限を設定できます。

- アクセス許可の境界 - アクセス許可の境界は、アイデンティティベースポリシーによって IAM エンティティ (IAM ユーザーまたはロール) に付与できる権限の上限を設定する高度な機能です。エ

エンティティにアクセス許可の境界を設定できます。結果として得られる権限は、エンティティのアイデンティティベースポリシーとそのアクセス許可の境界の共通部分になります。Principal フィールドでユーザーまたはロールを指定するリソースベースのポリシーでは、アクセス許可の境界は制限されません。これらのポリシーのいずれかを明示的に拒否した場合、権限は無効になります。アクセス許可の境界の詳細については、「IAM ユーザーガイド」の「[IAM エンティティのアクセス許可の境界](#)」を参照してください。

- サービスコントロールポリシー (SCPs) – SCPsは、 の組織または組織単位 (OU) の最大アクセス許可を指定する JSON ポリシーです AWS Organizations。AWS Organizations は、ビジネスが所有する複数の をグループ化して一元管理するためのサービス AWS アカウント です。組織内のすべての機能を有効にすると、サービスコントロールポリシー (SCP) を一部またはすべてのアカウントに適用できます。SCP は、各 を含むメンバーアカウントのエンティティのアクセス許可を制限します AWS アカウントのルートユーザー。Organizations と SCP の詳細については、「AWS Organizations ユーザーガイド」の「[サービスコントロールポリシー \(SCP\)](#)」を参照してください。
- リソースコントロールポリシー (RCP) – RCP は、所有する各リソースにアタッチされた IAM ポリシーを更新することなく、アカウント内のリソースに利用可能な最大数のアクセス許可を設定するために使用できる JSON ポリシーです。RCP は、メンバーアカウントのリソースに対するアクセス許可を制限し、組織に属しているかどうかにかかわらず AWS アカウントのルートユーザー、 を含む ID に対する有効なアクセス許可に影響を与える可能性があります。RCP AWS のサービスをサポートする のリストを含む Organizations と RCPs「[リソースコントロールポリシー \(RCPs\)](#)」を参照してください。AWS Organizations
- セッションポリシー - セッションポリシーは、ロールまたはフェデレーションユーザーの一時的なセッションをプログラムで作成する際にパラメータとして渡す高度なポリシーです。結果としてセッションの権限は、ユーザーまたはロールのアイデンティティベースポリシーとセッションポリシーの共通部分になります。また、リソースベースのポリシーから権限が派生する場合もあります。これらのポリシーのいずれかを明示的に拒否した場合、権限は無効になります。詳細については、「IAM ユーザーガイド」の「[セッションポリシー](#)」を参照してください。

複数のポリシータイプ

1 つのリクエストに複数のタイプのポリシーが適用されると、結果として作成される権限を理解するのがさらに難しくなります。複数のポリシータイプが関係する場合にリクエストを許可するかどうか AWS を決定する方法については、IAM ユーザーガイドの「[ポリシー評価ロジック](#)」を参照してください。

Amazon Managed Blockchain (AMB) Access Bitcoin と IAM の連携

IAM を使用して AMB Access Bitcoin へのアクセスを管理する前に、AMB Access Bitcoin で使用できる IAM 機能について説明します。

Amazon Managed Blockchain (AMB) Access Bitcoin で使用できる IAM 機能

IAM 機能	AMB Access Bitcoin のサポート
アイデンティティベースポリシー	はい
リソースベースのポリシー	いいえ
ポリシーアクション	はい
ポリシーリソース	いいえ
ポリシー条件キー	いいえ
ACL	いいえ
ABAC (ポリシー内のタグ)	いいえ
一時的な認証情報	いいえ
プリンシパル権限	いいえ
サービスロール	いいえ
サービスリンクロール	いいえ

AMB Access Bitcoin およびその他の AWS のサービスがほとんどの IAM 機能と連携する方法の概要については、IAM ユーザーガイドの[AWS 「IAM と連携する のサービス」](#)を参照してください。

AMB Access Bitcoin のアイデンティティベースのポリシー

アイデンティティベースのポリシーのサポート: あり

アイデンティティベースポリシーは、IAM ユーザーグループ、ユーザーのグループ、ロールなど、アイデンティティにアタッチできる JSON 許可ポリシードキュメントです。これらのポリシーは、

ユーザーとロールが実行できるアクション、リソース、および条件をコントロールします。ID ベースのポリシーの作成方法については、「IAM ユーザーガイド」の「[カスタマー管理ポリシーでカスタム IAM アクセス許可を定義する](#)」を参照してください。

IAM アイデンティティベースのポリシーでは、許可または拒否するアクションとリソース、およびアクションを許可または拒否する条件を指定できます。プリンシパルは、それが添付されているユーザーまたはロールに適用されるため、アイデンティティベースのポリシーでは指定できません。JSON ポリシーで使用するすべての要素について学ぶには、「IAM ユーザーガイド」の「[IAM JSON ポリシーの要素のリファレンス](#)」を参照してください。

AMB Access Bitcoin のアイデンティティベースのポリシーの例

AMB Access Bitcoin アイデンティティベースのポリシーの例を表示するには、「」を参照してください [Amazon Managed Blockchain \(AMB\) Access Bitcoin のアイデンティティベースのポリシーの例](#)。

AMB Access Bitcoin 内のリソースベースのポリシー

リソースベースのポリシーのサポート: なし

リソースベースのポリシーは、リソースに添付する JSON ポリシードキュメントです。リソースベースのポリシーには例として、IAM ロールの信頼ポリシーや Amazon S3 バケットポリシーがあげられます。リソースベースのポリシーをサポートするサービスでは、サービス管理者はポリシーを使用して特定のリソースへのアクセスを制御できます。ポリシーがアタッチされているリソースの場合、指定されたプリンシパルがそのリソースに対して実行できるアクションと条件は、ポリシーによって定義されます。リソースベースのポリシーでは、[プリンシパルを指定する](#)必要があります。プリンシパルには、アカウント、ユーザー、ロール、フェデレーティッドユーザー、またはを含めることができます AWS のサービス。

クロスアカウントアクセスを有効にするには、アカウント全体、または別のアカウントの IAM エンティティをリソースベースのポリシーのプリンシパルとして指定します。リソースベースのポリシーにクロスアカウントのプリンシパルを追加しても、信頼関係は半分しか確立されない点に注意してください。プリンシパルとリソースが異なる場合 AWS アカウント、信頼されたアカウントの IAM 管理者は、プリンシパルエンティティ (ユーザーまたはロール) にリソースへのアクセス許可も付与する必要があります。IAM 管理者は、アイデンティティベースのポリシーをエンティティにアタッチすることで権限を付与します。ただし、リソースベースのポリシーで、同じアカウントのプリンシパルへのアクセス権が付与されている場合は、アイデンティティベースのポリシーをさらに付与する必要はありません。詳細については、「IAM ユーザーガイド」の「[IAM でのクロスアカウントリソースアクセス](#)」を参照してください。

AMB Access Bitcoin のポリシーアクション

ポリシーアクションのサポート:あり

管理者は JSON AWS ポリシーを使用して、誰が何にアクセスできるかを指定できます。つまり、どのプリンシパルがどのリソースに対してどのような条件下でアクションを実行できるかということです。

JSON ポリシーの Action 要素にはポリシー内のアクセスを許可または拒否するために使用できるアクションが記述されます。ポリシーアクションの名前は通常、関連付けられた AWS API オペレーションと同じです。一致する API オペレーションのない許可のみのアクションなど、いくつかの例外があります。また、ポリシーに複数のアクションが必要なオペレーションもあります。これらの追加アクションは依存アクションと呼ばれます。

このアクションは関連付けられたオペレーションを実行するためのアクセス許可を付与するポリシーで使用されます。

AMB Access Bitcoin アクションのリストを確認するには、「サービス認可リファレンス」の「[Amazon Managed Blockchain \(AMB\) Access Bitcoin で定義されるアクション](#)」を参照してください。

AMB Access Bitcoin のポリシーアクションは、アクションの前に次のプレフィックスを使用します。

```
managedblockchain:
```

単一のステートメントで複数のアクションを指定するには、アクションをカンマで区切ります。

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

ワイルドカード (*) を使用して複数アクションを指定できます。例えば、InvokeRpcBitcoin という単語で始まるすべてのアクションを指定するには次のアクションを含めます。

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

AMB Access Bitcoin アイデンティティベースのポリシーの例を表示するには、「」を参照してください[Amazon Managed Blockchain \(AMB\) Access Bitcoin のアイデンティティベースのポリシーの例](#)。

AMB Access Bitcoin のポリシーリソース

ポリシーリソースのサポート: なし

管理者は JSON AWS ポリシーを使用して、誰が何にアクセスできるかを指定できます。つまり、どのプリンシパルがどのリソースに対してどのような条件下でアクションを実行できるかということです。

Resource JSON ポリシー要素はアクションが適用されるオブジェクトを指定します。ステートメントには Resource または NotResource 要素を含める必要があります。ベストプラクティスとして、[Amazon リソースネーム \(ARN\)](#) を使用してリソースを指定します。これは、リソースレベルの許可と呼ばれる特定のリソースタイプをサポートするアクションに対して実行できます。

オペレーションのリスト化など、リソースレベルの権限をサポートしないアクションの場合は、ステートメントがすべてのリソースに適用されることを示すために、ワイルドカード (*) を使用します。

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

AMB Access Bitcoin リソースタイプとその ARNs [「Amazon Managed Blockchain \(AMB\) Access Bitcoin で定義されるリソース」](#)を参照してください。各リソースの ARN を指定できるアクションについては、[「Amazon Managed Blockchain \(AMB\) Access Bitcoin で定義されるアクション」](#)を参照してください。

AMB Access Bitcoin アイデンティティベースのポリシーの例を表示するには、「」を参照してください[Amazon Managed Blockchain \(AMB\) Access Bitcoin のアイデンティティベースのポリシーの例](#)。

AMB Access Bitcoin のポリシー条件キー

サービス固有のポリシー条件キーへのサポート: なし

管理者は JSON AWS ポリシーを使用して、誰が何にアクセスできるかを指定できます。つまり、どのプリンシパルがどのリソースに対してどのような条件下でアクションを実行できるかということです。

Condition 要素 (または Condition ブロック) を使用すると、ステートメントが有効な条件を指定できます。Condition 要素はオプションです。イコールや未満などの [条件演算子](#) を使用して条件式を作成して、ポリシーの条件とリクエスト内の値を一致させることができます。

1 つのステートメントに複数の Condition 要素を指定する場合、または 1 つの Condition 要素に複数のキーを指定する場合、AWS では AND 論理演算子を使用してそれらを評価します。1 つの条件キーに複数の値を指定すると、は論理ORオペレーションを使用して条件 AWS を評価します。ステートメントの権限が付与される前にすべての条件が満たされる必要があります。

条件を指定する際にプレースホルダー変数も使用できます。例えば IAM ユーザーに、IAM ユーザー名がタグ付けされている場合のみリソースにアクセスできる権限を付与することができます。詳細については、「IAM ユーザーガイド」の「[IAM ポリシーの要素: 変数およびタグ](#)」を参照してください。

AWS は、グローバル条件キーとサービス固有の条件キーをサポートしています。すべての AWS グローバル条件キーを確認するには、「IAM ユーザーガイド」の[AWS 「グローバル条件コンテキストキー」](#)を参照してください。

AMB Access Bitcoin 条件キーのリストを確認するには、「サービス認可リファレンス」の「[Amazon Managed Blockchain \(AMB\) Access Bitcoin の条件キー](#)」を参照してください。条件キーを使用できるアクションとリソースについては、「[Amazon Managed Blockchain \(AMB\) Access Bitcoin で定義されるアクション](#)」を参照してください。

AMB Access Bitcoin アイデンティティベースのポリシーの例を表示するには、「」を参照してください[Amazon Managed Blockchain \(AMB\) Access Bitcoin のアイデンティティベースのポリシーの例](#)。

AMB Access Bitcoin ACLs

ACL のサポート: なし

アクセスコントロールリスト (ACL) は、どのプリンシパル (アカウントメンバー、ユーザー、またはロール) がリソースにアクセスするための許可を持つかを制御します。ACL はリソースベースのポリシーに似ていますが、JSON ポリシードキュメント形式は使用しません。

AMB Access Bitcoin を使用した ABAC

ABAC (ポリシー内のタグ) のサポート: なし

属性ベースのアクセス制御 (ABAC) は、属性に基づいてアクセス許可を定義する認可戦略です。では AWS、これらの属性はタグと呼ばれます。タグは、IAM エンティティ (ユーザーまたはロール) およ

び多くの AWS リソースにアタッチできます。エンティティとリソースのタグ付けは、ABAC の最初の手順です。その後、プリンシパルのタグがアクセスしようとしているリソースのタグと一致した場合にオペレーションを許可するように ABAC ポリシーをします。

ABAC は、急成長する環境やポリシー管理が煩雑になる状況で役立ちます。

タグに基づいてアクセスを管理するには、`aws:ResourceTag/key-name`、`aws:RequestTag/key-name`、または `aws:TagKeys` の条件キーを使用して、ポリシーの [条件要素](#) でタグ情報を提供します。

サービスがすべてのリソースタイプに対して 3 つの条件キーすべてをサポートする場合、そのサービスの値はありです。サービスが一部のリソースタイプに対してのみ 3 つの条件キーのすべてをサポートする場合、値は「部分的」になります。

ABAC の詳細については、「IAM ユーザーガイド」の「[ABAC 認可でアクセス許可を定義する](#)」を参照してください。ABAC をセットアップする手順を説明するチュートリアルについては、「IAM ユーザーガイド」の「[属性ベースのアクセスコントロール \(ABAC\) を使用する](#)」を参照してください。

AMB Access Bitcoin での一時的な認証情報の使用

一時的な認証情報のサポート: なし

一部の AWS のサービスは、一時的な認証情報を使用してサインインすると機能しません。一時的な認証情報 AWS のサービスを使用するなどの詳細については、IAM ユーザーガイド [AWS のサービスの「IAM と連携する」](#) を参照してください。

ユーザー名とパスワード以外の AWS Management Console 方法でサインインする場合、一時的な認証情報を使用します。たとえば、会社のシングルサインオン (SSO) リンク AWS を使用してアクセスすると、そのプロセスによって一時的な認証情報が自動的に作成されます。また、ユーザーとしてコンソールにサインインしてからロールを切り替える場合も、一時的な認証情報が自動的に作成されます。ロールの切り替えに関する詳細については、「IAM ユーザーガイド」の「[ユーザーから IAM ロールに切り替える \(コンソール\)](#)」を参照してください。

一時的な認証情報は、AWS CLI または AWS API を使用して手動で作成できます。その後、これらの一時的な認証情報を使用してアクセスすることができます AWS。長期的なアクセスキーを使用する代わりに、一時的な認証情報を動的に生成 AWS することをお勧めします。詳細については、「[IAM の一時的セキュリティ認証情報](#)」を参照してください。

AMB Access Bitcoin のクロスサービスプリンシパル許可

転送アクセスセッション (FAS) のサポート: なし

IAM ユーザーまたはロールを使用してアクションを実行すると AWS、プリンシパルと見なされます。一部のサービスを使用する際に、アクションを実行することで、別のサービスの別のアクションがトリガーされることがあります。FAS は、 を呼び出すプリンシパルのアクセス許可と AWS のサービス、ダウストリームサービス AWS のサービス へのリクエストのリクエストをリクエストする を使用します。FAS リクエストは、サービスが他の AWS のサービス またはリソースとのやり取りを完了する必要があるリクエストを受け取った場合にのみ行われます。この場合、両方のアクションを実行するためのアクセス許可が必要です。FAS リクエストを行う際のポリシーの詳細については、「[転送アクセスセッション](#)」を参照してください。

AMB Access Bitcoin のサービスロール

サービスロールのサポート: なし

サービスロールとは、サービスがユーザーに代わってアクションを実行するために引き受ける [IAM ロール](#) です。IAM 管理者は、IAM 内からサービスロールを作成、変更、削除できます。詳細については、「IAM ユーザーガイド」の「[AWS のサービスに許可を委任するロールを作成する](#)」を参照してください。

Warning

サービスロールのアクセス許可を変更すると、AMB Access Bitcoin の機能が破損する可能性があります。AMB Access Bitcoin が指示する場合にのみ、サービスロールを編集します。

AMB Access Bitcoin のサービスにリンクされたロール

サービスにリンクされたロールのサポート: なし

サービスにリンクされたロールは、 にリンクされたサービスロールの一種です AWS のサービス。サービスは、ユーザーに代わってアクションを実行するロールを引き受けることができます。サービスにリンクされたロールは に表示され AWS アカウント、 サービスによって所有されます。IAM 管理者は、サービスにリンクされたロールのアクセス許可を表示できますが、編集することはできません。

サービスにリンクされたロールの作成または管理の詳細については、「[IAM と提携するAWS のサービス](#)」を参照してください。表の「サービスリンクロール」列に Yes と記載されたサービスを見つ

けます。サービスにリンクされたロールに関するドキュメントをサービスで表示するには、[はい] リンクを選択します。

Amazon Managed Blockchain (AMB) Access Bitcoin のアイデンティティベースのポリシーの例

デフォルトでは、ユーザーとロールには AMB Access Bitcoin リソースを作成または変更するアクセス許可はありません。また、AWS Command Line Interface (AWS CLI)、AWS Management Console、または AWS API を使用してタスクを実行することはできません。IAM 管理者は、リソースに必要なアクションを実行するための権限をユーザーに付与する IAM ポリシーを作成できます。その後、管理者はロールに IAM ポリシーを追加し、ユーザーはロールを引き継ぐことができます。

これらサンプルの JSON ポリシードキュメントを使用して、IAM アイデンティティベースのポリシーを作成する方法については、「IAM ユーザーガイド」の「[IAM ポリシーを作成する \(コンソール\)](#)」を参照してください。

各リソースタイプの ARNs「サービス認可リファレンス」の「[Amazon Managed Blockchain \(AMB\) Access Bitcoin のアクション、リソース、および条件キー](#)」を参照してください。

トピック

- [ポリシーに関するベストプラクティス](#)
- [AMB Access Bitcoin コンソールの使用](#)
- [自分の権限の表示をユーザーに許可する](#)
- [Bitcoin ネットワークへのアクセス](#)

ポリシーに関するベストプラクティス

ID ベースのポリシーは、アカウント内の AMB Access Bitcoin リソースを作成、アクセス、または削除できるかどうかを決定します。これらのアクションを実行すると、AWS アカウントに料金が発生する可能性があります。アイデンティティベースポリシーを作成したり編集したりする際には、以下のガイドラインと推奨事項に従ってください:

- AWS 管理ポリシーを開始し、最小特権のアクセス許可に移行 – ユーザーとワークロードにアクセス許可の付与を開始するには、多くの一般的なユースケースにアクセス許可を付与する AWS 管理ポリシーを使用します。これらは使用できます AWS アカウント。ユースケースに固有の AWS カスタマー管理ポリシーを定義することで、アクセス許可をさらに減らすことをお勧めします。詳細については、「IAM ユーザーガイド」の「[AWS マネージドポリシー](#)」または「[ジョブ機能の AWS マネージドポリシー](#)」を参照してください。

- 最小特権を適用する – IAM ポリシーで許可を設定する場合は、タスクの実行に必要な許可のみを付与します。これを行うには、特定の条件下で特定のリソースに対して実行できるアクションを定義します。これは、最小特権アクセス許可とも呼ばれています。IAM を使用して許可を適用する方法の詳細については、「IAM ユーザーガイド」の「[IAM でのポリシーとアクセス許可](#)」を参照してください。
- IAM ポリシーで条件を使用してアクセスをさらに制限する – ポリシーに条件を追加して、アクションやリソースへのアクセスを制限できます。例えば、ポリシー条件を記述して、すべてのリクエストを SSL を使用して送信するように指定できます。条件を使用して、などの特定の を通じてサービスアクションが使用される場合に AWS のサービス、サービスアクションへのアクセスを許可することもできます AWS CloudFormation。詳細については、「IAM ユーザーガイド」の「[IAM JSON ポリシー要素:条件](#)」を参照してください。
- IAM Access Analyzer を使用して IAM ポリシーを検証し、安全で機能的な権限を確保する – IAM Access Analyzer は、新規および既存のポリシーを検証して、ポリシーが IAM ポリシー言語 (JSON) および IAM のベストプラクティスに準拠するようにします。IAM アクセスアナライザーは 100 を超えるポリシーチェックと実用的な推奨事項を提供し、安全で機能的なポリシーの作成をサポートします。詳細については、「IAM ユーザーガイド」の「[IAM Access Analyzer でポリシーを検証する](#)」を参照してください。
- 多要素認証 (MFA) を要求する – IAM ユーザーまたはルートユーザーを必要とするシナリオがある場合は AWS アカウント、セキュリティを強化するために MFA を有効にします。API オペレーションが呼び出されるときに MFA を必須にするには、ポリシーに MFA 条件を追加します。詳細については、「IAM ユーザーガイド」の「[MFA を使用した安全な API アクセス](#)」を参照してください。

IAM でのベストプラクティスの詳細については、「IAM ユーザーガイド」の「[IAM でのセキュリティのベストプラクティス](#)」を参照してください。

AMB Access Bitcoin コンソールの使用

Amazon Managed Blockchain (AMB) Access Bitcoin コンソールにアクセスするには、最小限のアクセス許可のセットが必要です。これらのアクセス許可により、内の AMB Access Bitcoin リソースの詳細を一覧表示および表示できます AWS アカウント。最小限必要な許可よりも制限が厳しいアイデンティティベースのポリシーを作成すると、そのポリシーを持つエンティティ (ユーザーまたはロール) に対してコンソールが意図したとおりに機能しません。

AWS CLI または AWS API のみを呼び出すユーザーには、最小限のコンソールアクセス許可を付与する必要はありません。代わりに、実行しようとしている API オペレーションに一致するアクションのみへのアクセスが許可されます。

ユーザーとロールが引き続き AMB Access Bitcoin コンソールを使用できるようにするには、エンティティに AMB Access Bitcoin *ConsoleAccess* または *ReadOnly* AWS マネージドポリシーもアタッチします。詳細については、「IAM ユーザーガイド」の「[ユーザーへのアクセス許可の追加](#)」を参照してください。

自分の権限の表示をユーザーに許可する

この例では、ユーザーアイデンティティにアタッチされたインラインおよびマネージドポリシーの表示を IAM ユーザーに許可するポリシーの作成方法を示します。このポリシーには、コンソールで、または AWS CLI または AWS API を使用してプログラムでこのアクションを実行するアクセス許可が含まれています。

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

```
]
}
```

Bitcoin ネットワークへのアクセス

Note

Bitcoin のパブリックエンドポイントにアクセスし mainnet、JSON-RPC 呼び出し testnet を行うには、AMB Access Bitcoin に適切な IAM アクセス許可を持つユーザー認証情報 (AWS_ACCESS_KEY_ID および AWS_SECRET_ACCESS_KEY) が必要です。

Example すべての Bitcoin Networks にアクセスするための IAM ポリシー

この例では、すべての Bitcoin ネットワーク AWS アカウント へのアクセス権を IAM ユーザーに付与します。

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

Example Bitcoin Testnet ネットワークにアクセスするための IAM ポリシー

この例では、Bitcoin testnet ネットワーク AWS アカウント へのアクセスを IAM ユーザーに許可します。

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

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

Amazon Managed Blockchain (AMB) Access Bitcoin のアイデンティティとアクセスのトラブルシューティング

以下の情報は、AMB Access Bitcoin と IAM の使用時に発生する可能性がある一般的な問題の診断と修正に役立ちます。

トピック

- [AMB Access Bitcoin でアクションを実行する権限がありません](#)
- [iam:PassRole を実行する権限がありません](#)
- [自分の 以外のユーザーに AMB Access Bitcoin リソース AWS アカウント へのアクセスを許可したい](#)

AMB Access Bitcoin でアクションを実行する権限がありません

アクションを実行する権限がないというエラーが表示された場合は、そのアクションを実行できるようにポリシーを更新する必要があります。

次のエラー例は、mateojackson IAM ユーザーがコンソールを使用して、ある *my-example-widget* リソースに関する詳細情報を表示しようとしたことを想定して、その際に必要なmanagedblockchain::*GetWidget* アクセス許可を持っていない場合に発生するものです。

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

この場合、managedblockchain::*GetWidget* アクションを使用して *my-example-widget* リソースへのアクセスを許可するように、mateojackson ユーザーのポリシーを更新する必要があります。

サポートが必要な場合は、AWS 管理者にお問い合わせください。サインイン認証情報を提供した担当者が管理者です。

iam:PassRole を実行する権限がありません

iam:PassRole アクションを実行する権限がないというエラーが表示された場合は、ポリシーを更新して AMB Access Bitcoin にロールを渡すことができるようにする必要があります。

一部の AWS のサービスでは、新しいサービスロールまたはサービスにリンクされたロールを作成する代わりに、そのサービスに既存のロールを渡すことができます。そのためには、サービスにロールを渡す権限が必要です。

次の例のエラーは、という名前の IAM marymajor ユーザーがコンソールを使用して AMB Access Bitcoin でアクションを実行しようとするると発生します。ただし、このアクションをサービスが実行するには、サービスロールから付与された権限が必要です。メアリーには、ロールをサービスに渡す許可がありません。

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

この場合、Mary のポリシーを更新してメアリーに iam:PassRole アクションの実行を許可する必要があります。

サポートが必要な場合は、AWS 管理者にお問い合わせください。サインイン資格情報を提供した担当者が管理者です。

自分の 以外のユーザーに AMB Access Bitcoin リソース AWS アカウント へのアクセスを許可したい

他のアカウントのユーザーや組織外の人、リソースにアクセスするために使用できるロールを作成できます。ロールの引き受けを委託するユーザーを指定できます。リソースベースのポリシーまたはアクセスコントロールリスト (ACL) をサポートするサービスの場合、それらのポリシーを使用して、リソースへのアクセスを付与できます。

詳細については、以下を参照してください:

- AMB Access Bitcoin がこれらの機能をサポートしているかどうかを確認するには、「」を参照してください [Amazon Managed Blockchain \(AMB\) Access Bitcoin と IAM の連携](#)。

- 所有 AWS アカウント している のリソースへのアクセスを提供する方法については、IAM ユーザーガイドの「[所有 AWS アカウント している別の の IAM ユーザーへのアクセスを提供する](#)」を参照してください。
- リソースへのアクセスをサードパーティーに提供する方法については AWS アカウント、IAM ユーザーガイドの「[サードパーティー AWS アカウント が所有する へのアクセスを提供する](#)」を参照してください。
- ID フェデレーションを介してアクセスを提供する方法については、「IAM ユーザーガイド」の「[外部で認証されたユーザー \(ID フェデレーション\) へのアクセスの許可](#)」を参照してください。
- クロスアカウントアクセスにおけるロールとリソースベースのポリシーの使用法の違いについては、「IAM ユーザーガイド」の「[IAM でのクロスアカウントのリソースへのアクセス](#)」を参照してください。

を使用した Amazon Managed Blockchain (AMB) アクセス Bitcoin イベントのログ記録 AWS CloudTrail

Note

Amazon Managed Blockchain (AMB) Access Bitcoin は管理イベントをサポートしていません。

Amazon Managed Blockchain は AWS CloudTrail、Managed Blockchain のユーザー、ロール、または のサービスによって実行されたアクションを記録する AWS サービスであると統合されています。CloudTrail は、マネージドブロックチェーンの AMB Access Bitcoin エンドポイントと呼び出すユーザーをデータプレーンイベントとしてキャプチャします。

目的のデータプレーンイベントを受信するようにサブスクライブされている適切に設定された証跡を作成すると、AMB Access Bitcoin 関連の CloudTrail イベントを Amazon S3 バケットに継続的に配信できます。CloudTrail によって収集された情報を使用して、AMB Access Bitcoin エンドポイントの 1 つ、リクエスト元の IP アドレス、リクエスト者、リクエスト日時などの詳細に対してリクエストが行われたかどうかを判断できます。

CloudTrail の詳細については、「[AWS CloudTrail ユーザーガイド](#)」を参照してください。

CloudTrail での AMB Access Bitcoin 情報

AWS CloudTrail を作成すると、はデフォルトで有効になります AWS アカウント。ただし、誰が AMB Access Bitcoin エンドポイントと呼び出したかを確認するには、データプレーンイベントをログに記録するように CloudTrail を設定する必要があります。

AMB Access Bitcoin のデータプレーンイベントなど AWS アカウント、 のイベントを継続的に記録するには、証跡を作成する必要があります。証跡により、CloudTrail はログファイルを Amazon S3 バケットに配信します。デフォルトでは、 で証跡を作成すると AWS Management Console、証跡はすべてに適用されます AWS リージョン。証跡は、AWS パーティションでサポートされているすべてのリージョンからのイベントをログに記録し、指定した Amazon S3 バケットにログファイルを配信します。さらに、このデータをさらに分析し、CloudTrail ログで収集されたイベントデータを処理するように他の AWS サービスを設定できます。詳細については、次を参照してください:

- [CloudTrail を使用して Bitcoin JSON-RPCs](#)

- [追跡を作成するための概要](#)
- 「[CloudTrail がサポートされているサービスと統合](#)」
- 「[CloudTrail の Amazon SNS 通知の設定](#)」
- [複数のリージョンから CloudTrail ログファイルを受け取る](#) および [複数のアカウントから CloudTrail ログファイルを受け取る](#)

CloudTrail データイベントを分析することで、AMB Access Bitcoin エンドポイントを呼び出したユーザーをモニタリングできます。

各イベントまたはログエントリには、誰がリクエストを生成したかという情報が含まれます。アイデンティティ情報は、以下を判別するのに役立ちます。

- リクエストが root または AWS Identity and Access Management (IAM) ユーザー認証情報を使用して行われたかどうか。
- リクエストが、ロールとフェデレーティッドユーザーのどちらかの一時的なセキュリティ認証情報を使用して送信されたかどうか。
- リクエストが別の AWS サービスによって行われたかどうか。

詳細については、「[CloudTrail userIdentity エlement](#)」を参照してください。

AMB Access Bitcoin ログファイルエントリについて

データプレーンイベントの場合、証跡は、指定された S3 バケットへのログファイルとしてのイベントの配信を可能にする設定です。各 CloudTrail ログファイルには、任意のソースからの 1 つのリクエストを表す 1 つ以上のログエントリが含まれています。これらのエントリは、アクションの日時、関連するリクエストパラメータなど、リクエストされたアクションに関する詳細を提供します。

Note

ログファイルの CloudTrail データイベントは、AMB Access Bitcoin API コールの順序付けられたスタックトレースではないため、特定の順序では表示されません。

CloudTrail を使用して Bitcoin JSON-RPCs

CloudTrail を使用して、アカウント内の誰が AMB Access Bitcoin エンドポイントを呼び出し、どの JSON-RPC がデータイベントとして呼び出されたかを追跡できます。デフォルトでは、証跡を作

成すると、データイベントはログに記録されません。AMB Access Bitcoin エンドポイント呼び出したユーザーを CloudTrail データイベントとして記録するには、アクティビティを収集するサポートされているリソースまたはリソースタイプを証跡に明示的に追加する必要があります。Amazon Managed Blockchain は AWS Management Console、AWS SDK、および `awscli` を使用してデータイベントを追加することをサポートしています AWS CLI。詳細については、「[AWS CloudTrail ユーザーガイド](#)」の「[高度なセレクトアを使用してイベントをログに記録する](#)」を参照してください。

証跡のデータイベントをログに記録するには、証跡を作成した後、[put-event-selectors](#) オペレーションを使用します。--advanced-event-selectors オプションを使用して `AWS::ManagedBlockchain::Network` リソースタイプを指定し、データイベントのログ記録を開始して、誰が AMB Access Bitcoin エンドポイント呼び出したかを判断します。

Example アカountのすべての AMB Access Bitcoin エンドポイントリクエストのデータイベントログエントリ

次の例は、`put-event-selectors` オペレーションを使用して、`us-east-1` リージョン `my-bitcoin-trail` の証跡に対するアカウントの AMB Access Bitcoin エンドポイントリクエストをすべてログに記録する方法を示しています。

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

サブスクライブすると、前の例で指定した証跡に接続されている S3 バケットの使用状況を追跡できます。

次の結果は、CloudTrail によって収集された情報の CloudTrail データイベントログエントリを示しています。Bitcoin JSON-RPC リクエストが AMB Access Bitcoin エンドポイントの 1 つ、リクエスト元の IP アドレス、リクエスト者、リクエスト日時などの詳細に対して行われたかどうかを判断できます。

```
{
  "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"
}
```

翻訳は機械翻訳により提供されています。提供された翻訳内容と英語版の間で齟齬、不一致または矛盾がある場合、英語版が優先します。