aws

Entwicklerhandbuch

AMB-Zugriffspolygon

AMB-Zugriffspolygon: Entwicklerhandbuch

Table of Contents

Amazon Managed Blockchain (AMB) Access Polygon befindet sich in der Vorschauversion und kann sich ändern.

Die vorliegende Übersetzung wurde maschinell erstellt. Im Falle eines Konflikts oder eines Widerspruchs zwischen dieser übersetzten Fassung und der englischen Fassung (einschließlich infolge von Verzögerungen bei der Übersetzung) ist die englische Fassung maßgeblich.

Was ist Amazon Managed Blockchain (AMB) Access Polygon?

Amazon Managed Blockchain (AMB) Access Polygon ist ein vollständig verwalteter Service, mit dem Sie robuste Web3-Anwendungen auf der Polygon-Blockchain erstellen können. AMB Access Polygon bietet sofortigen und serverlosen Zugriff auf die Polygon-Blockchain.

Polygon ist eine Skalierungslösung, die die Ethereum Virtual Machine (EVM) als Grundlage verwendet. Die Polygon-Blockchain ist bekannt für ihren hohen Transaktionsdurchsatz und niedrige Transaktionsgebühren. Die Polygon-Blockchain verwendet einen proof-of-stake Konsensmechanismus. Polygon wird häufig zum Erstellen dezentraler Anwendungen (DApps) verwendet NFTs, die sich unter anderem auf Web3-Spiele und Tokenisierung beziehen.

In diesem Handbuch wird beschrieben, wie Sie Polygon-Blockchain-Ressourcen mithilfe von Amazon Managed Blockchain (AMB) Access Polygon erstellen und verwalten.

Ressourcen für AMB Access Polygon-Erstbenutzer

Wenn Sie AMB Access Polygon zum ersten Mal verwenden, empfehlen wir Ihnen, zunächst die folgenden Abschnitte zu lesen:

- Schlüsselkonzepte: Amazon Managed Blockchain (AMB) Access Polygon
- Erste Schritte mit Amazon Managed Blockchain (AMB) Access Polygon
- Verwaltete Blockchain-API und RPCs JSON-Unterstützung durch AMB Access Polygon

Schlüsselkonzepte: Amazon Managed Blockchain (AMB) Access Polygon

> **Note**
>
> In diesem Handbuch wird davon ausgegangen, dass Sie mit den für Polygon wesentlichen Konzepten vertraut sind. Zu diesen Konzepten gehören Staking, DApps, Transaktionen, Wallets, Smart Contracts, Polygon (POL, ehemals MATIC) und andere. Bevor Sie Amazon Managed Blockchain (AMB) Access Polygon verwenden, empfehlen wir Ihnen, die Polygon-Entwicklungsdokumentation und das Polygon-Wiki zu lesen.

Amazon Managed Blockchain (AMB) Access Polygon bietet Ihnen serverlosen Zugriff auf die Polygon Mainnet- und Polygon Mainnet-Netzwerke, ohne dass Sie eine Polygon-Infrastruktur, einschließlich Knoten, bereitstellen und verwalten müssen. Polygon-Knoten in einem Netzwerk speichern gemeinsam einen Polygon-Blockchain-Status, verifizieren Transaktionen und stimmen einvernehmlich über die Änderung eines Blockchain-Status ab. Mit diesem verwalteten Service können Sie schnell und bei Bedarf auf die Polygon-Netzwerke zugreifen und so Ihre Gesamtbetriebskosten senken.

Mit AMB Access Polygon haben Sie Zugriff auf JSON Remote Procedure (JSON-RPC) -Aufrufe. Sie können Polygon JSON- aufrufen, um mit der Polygon-Blockchain über Knoten RPCs zu kommunizieren, die von Managed Blockchain verwaltet werden. Sie können den AMB Access Polygon-Dienst verwenden, um dezentrale Anwendungen (dApps) zu entwickeln und zu verwenden, die mit der Polygon-Blockchain interagieren. Ein wesentlicher Bestandteil von dApps sind Smart Contracts. Mit AMB Access Polygon können Sie intelligente Verträge erstellen und in der Polygon-Blockchain bereitstellen. Sie können auch die Salden Ihrer Wallets, Transaktionsdetails, geschätzte Gebühren usw. überprüfen, indem Sie JSON- RPCs für AMB Access Polygon-Endpunkte aufrufen, die dezentral auf allen Knoten laufen, die Peers des Polygon-Netzwerks sind. Jeder Peer im Polygon-Netzwerk kann einen intelligenten Vertrag entwickeln und einsetzen.

> **Important**
>
> Sie sind für die Erstellung, Pflege, Verwendung und Verwaltung Ihrer Polygon-Adressen verantwortlich. Sie sind auch für den Inhalt Ihrer Polygon-Adressen verantwortlich. AWS ist

nicht verantwortlich für Transaktionen, die mithilfe von Polygon-Knoten auf Amazon Managed Blockchain bereitgestellt oder aufgerufen werden.

Überlegungen und Einschränkungen bei der Verwendung von Amazon Managed Blockchain (AMB) Access Polygon

Wenn Sie Amazon Managed Blockchain (AMB) Access Polygon verwenden, sollten Sie Folgendes beachten:

- Unterstützte Polygon-Netzwerke

  AMB Access Polygon unterstützt die folgenden öffentlichen Netzwerke:

  - Mainnet — Die öffentliche Polygon-Blockchain, die durch proof-of-stake Konsens gesichert ist und auf der das Polygon-Token (POL) ausgestellt und abgewickelt wird. Transaktionen im Mainnet haben einen tatsächlichen Wert (das heißt, sie verursachen echte Kosten) und werden in der öffentlichen Blockchain aufgezeichnet.

- Netzwerke, die von Polygon nicht mehr unterstützt werden

  - Wie von Polygon Labs mitgeteilt, wird das Mumbai Testnet-Netzwerk Mitte April eingestellt. Aufgrund dieser Nachricht hat AMB Access Polygon die Unterstützung des Mumbai Testnet am 15. April 2024 eingestellt. Wir empfehlen, Amoy Testnet für Ihre Test-Workloads zu verwenden.
  - Private Netzwerke werden nicht unterstützt.
  - Darüber hinaus bietet AMB Access Polygon keine Unterstützung für das Polygon ZkEVM-Netzwerk.

- Kompatibilität mit gängigen Programmierbibliotheken von Drittanbietern

  AMB Access Polygon ist mit gängigen Programmierbibliotheken wie ethers.js kompatibel, sodass Entwickler mithilfe vertrauter Tools mit der Polygon-Blockchain interagieren können, um sie einfach in ihre bestehenden Implementierungen zu integrieren oder schnell neue Anwendungen zu entwickeln.

- Unterstützte Regionen

  Dieser Service wird nur in der Region USA Ost (Nord-Virginia) unterstützt.

- Service-Endpunkte

Im Folgenden sind die Dienstendpunkte für AMB Access Polygon aufgeführt. Um eine Verbindung mit dem Service herzustellen, müssen Sie einen Endpunkt verwenden, der eine der unterstützten Regionen enthält.

  - `mainnet.polygon.managedblockchain.us-east-1.amazonaws.com`
- Staking wird nicht unterstützt

  AMB Access Polygon unterstützt keine Polygon-Validierungsknoten (POL) für. proof-of-stake
- Signatur Version 4, Signierung von Polygon-JSON-RPC-Anfragen

  Wenn Sie Polygon JSON- RPCs auf Amazon Managed Blockchain aufrufen, können Sie dies über eine HTTPS-Verbindung tun, die mit dem Signature Version 4-Signaturprozess authentifiziert wurde. Das bedeutet, dass nur autorisierte IAM-Prinzipale im AWS Konto Polygon-JSON-RPC-Aufrufe tätigen können. Zu diesem Zweck müssen beim AWS Anruf Anmeldeinformationen (eine Zugriffsschlüssel-ID und ein geheimer Zugriffsschlüssel) bereitgestellt werden.

  > **Important**
  >
  > - Betten Sie keine Kundenanmeldedaten in benutzerseitige Anwendungen ein.
  > - Sie können IAM-Richtlinien nicht verwenden, um den Zugriff auf einzelne Polygon-JSON-Dateien einzuschränken. RPCs
- Support für Token-basierten Zugriff

  Sie können Accessor-Token auch verwenden, um JSON-RPC-Aufrufe an die Polygon-Netzwerkendpunkte als praktische Alternative zum Signaturprozess Signature Version 4 (Sigv4) zu tätigen. Sie müssen eines der `BILLING_TOKEN` von Ihnen erstellten Accessor-Token angeben und als Parameter bei Ihren Aufrufen hinzufügen.

  > **Important**
  >
  > - Wenn Sie Sicherheit und Überprüfbarkeit der Benutzerfreundlichkeit vorziehen, verwenden Sie stattdessen den SigV4-Signaturprozess.
  > - Sie können RPCs mithilfe von Signature Version 4 (Sigv4) und tokenbasiertem Zugriff auf das Polygon-JSON zugreifen. Wenn Sie sich jedoch dafür entscheiden, beide Protokolle zu verwenden, wird Ihre Anfrage abgelehnt.
  > - Sie dürfen Accessor-Token niemals in benutzerorientierte Anwendungen einbetten.

- Nur das Einreichen von Rohtransaktionen wird unterstützt

  Verwenden Sie den `eth_sendrawtransaction` JSON-RPC, um Transaktionen einzureichen, die den Status der Polygon-Blockchain aktualisieren.

Einrichtung von Amazon Managed Blockchain (AMB) Access Polygon

Bevor Sie Amazon Managed Blockchain (AMB) Access Polygon zum ersten Mal verwenden, folgen Sie den Schritten in diesem Abschnitt, um ein. AWS-Konto Im folgenden Kapitel wird beschrieben, wie Sie mit der Nutzung von AMB Access Polygon beginnen.

Voraussetzungen für die Verwendung von AMB Access Polygon

Bevor Sie es AWS zum ersten Mal verwenden, benötigen Sie eine. AWS-Konto

Melde dich an für AWS

Wenn Sie sich für registrieren AWS, AWS-Konto ist Ihr Konto automatisch für alle registriert AWS-Services, einschließlich Amazon Managed Blockchain (AMB) Access Polygon. Berechnet werden Ihnen aber nur die Services, die Sie nutzen.

Wenn Sie AWS-Konto bereits eines haben, fahren Sie mit dem nächsten Schritt fort. Wenn Sie kein AWS-Konto haben, führen Sie die folgenden Schritte zum Erstellen eines Kontos aus.

Um ein zu erstellen AWS-Konto

1. Öffnen Sie https://portal.aws.amazon.com/billing/die Anmeldung.
2. Folgen Sie den Online-Anweisungen.

   Ein Teil des Anmeldevorgangs umfasst den Empfang eines Telefonanrufs oder einer Textnachricht und die Eingabe eines Bestätigungscodes auf der Telefontastatur.

   Wenn Sie sich für eine anmelden AWS-Konto, wird eine Root-Benutzer des AWS-Kontoserstellt. Der Root-Benutzer hat Zugriff auf alle AWS-Services und Ressourcen des Kontos. Als bewährte Sicherheitsmethode weisen Sie einem Administratorbenutzer Administratorzugriff zu und verwenden Sie nur den Root-Benutzer, um Aufgaben auszuführen, die Root-Benutzerzugriff erfordern.

Erstellen Sie einen IAM-Benutzer mit den entsprechenden Berechtigungen

Um AMB Access Polygon zu erstellen und damit zu arbeiten, benötigen Sie einen AWS Identity and Access Management (IAM-) Principal (Benutzer oder Gruppe) mit Berechtigungen, die die erforderlichen Managed Blockchain-Aktionen ermöglichen.

Wenn Sie Polygon JSON- RPCs auf Amazon Managed Blockchain aufrufen, können Sie dies über eine HTTPS-Verbindung tun, die mit dem Signature Version 4-Signaturprozess authentifiziert wurde. Das bedeutet, dass nur autorisierte IAM-Prinzipale im AWS Konto Polygon-JSON-RPC-Aufrufe tätigen können. Zu diesem Zweck müssen beim AWS Anruf Anmeldeinformationen (eine Zugriffsschlüssel-ID und ein geheimer Zugriffsschlüssel) bereitgestellt werden.

Als praktische Alternative zum Signaturprozess Signature Version 4 (Sigv4) können Sie auch Accessor-Token verwenden, um JSON-RPC-Aufrufe an die Polygon-Netzwerkendpunkte zu tätigen. Sie müssen eines der `BILLING_TOKEN` von Ihnen erstellten Accessor-Token angeben und als Parameter bei Ihren Aufrufen hinzufügen. Sie benötigen jedoch weiterhin IAM-Zugriff, um Berechtigungen zum Erstellen von Accessor-Token mithilfe des SDK AWS Management Console AWS CLI, und zu erhalten.

Informationen zum Erstellen eines IAM-Benutzers finden Sie unter Einen IAM-Benutzer in Ihrem Konto erstellen. AWS Weitere Informationen dazu, wie Sie einem Benutzer eine Berechtigungsrichtlinie zuordnen, finden Sie unter Berechtigungen für einen IAM-Benutzer ändern. Ein Beispiel für eine Berechtigungsrichtlinie, mit der Sie einem Benutzer die Erlaubnis erteilen können, mit AMB Access Polygon zu arbeiten, finden Sie unter. Beispiele für identitätsbasierte Richtlinien für Amazon Managed Blockchain (AMB) Access Polygon

Installieren und Konfigurieren der AWS Command Line Interface.

Falls Sie dies noch nicht getan haben, installieren Sie die neueste Version AWS Command Line Interface (AWS CLI), um mit AWS Ressourcen von einem Terminal aus zu arbeiten. Weitere Informationen finden Sie unter Installieren oder Aktualisierung auf die neueste Version von AWS CLI.

> **Note**
>
> Für CLI-Zugriff benötigen Sie eine Zugriffsschlüssel-ID und einen geheimen Zugriffsschlüssel. Verwenden Sie möglichst temporäre Anmeldeinformationen anstelle langfristiger Zugriffsschlüssel. Temporäre Anmeldeinformationen bestehen aus einer Zugriffsschlüssel-

ID, einem geheimen Zugriffsschlüssel und einem Sicherheits-Token, das angibt, wann die Anmeldeinformationen ablaufen. Weitere Informationen finden Sie unter Verwenden temporärer Anmeldeinformationen mit AWS Ressourcen im IAM-Benutzerhandbuch.

Erste Schritte mit Amazon Managed Blockchain (AMB) Access Polygon

Verwenden Sie die Informationen und Verfahren in diesem Abschnitt, um mit Amazon Managed Blockchain (AMB) Access Polygon zu beginnen.

Themen

Erstellen Sie eine IAM-Richtlinie für den Zugriff auf das Polygon-Blockchain-Netzwerk

Um auf den öffentlichen Endpunkt für das Polygon-Mainnet zuzugreifen, um JSON-RPC-Aufrufe zu tätigen, benötigen Sie Benutzeranmeldedaten (`AWS_ACCESS_KEY_ID`und`AWS_SECRET_ACCESS_KEY`), die über die entsprechenden IAM-Berechtigungen für Amazon Managed Blockchain (AMB) Access Polygon verfügen. Führen Sie in einem Terminal, auf dem das AWS CLI installiert ist, den folgenden Befehl aus, um eine IAM-Richtlinie für den Zugriff auf beide Polygon-Endpunkte zu erstellen:

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

```
}
EOT
aws iam create-policy --policy-name AmazonManagedBlockchainPolygonAccess --policy-
document file://$HOME/amb-polygon-access-policy.json
```

> **Note**
>
> Im vorherigen Beispiel erhalten Sie Zugriff auf alle verfügbaren Polygon-Netzwerke. Verwenden Sie den folgenden `Action` Befehl, um Zugriff auf einen bestimmten Endpunkt zu erhalten:
>
> - `"managedblockchain:InvokeRpcPolygonMainnet"`

Nachdem Sie die Richtlinie erstellt haben, fügen Sie diese Richtlinie der Rolle Ihres IAM-Benutzers hinzu, damit sie wirksam wird. Navigieren Sie im AWS Management Console zum IAM-Dienst und fügen Sie die Richtlinie der Rolle `AmazonManagedBlockchainPolygonAccess` hinzu, die Ihrem IAM-Benutzer zugewiesen ist.

Stellen Sie Polygon Remote Procedure Call (RPC) -Anfragen im AMB Access RPC-Editor mithilfe des AWS Management Console

Sie können Remote Procedure Calls (RPCs) auf dem AMB AWS Management Console Access Polygon bearbeiten, konfigurieren und einreichen. Mit diesen RPCs können Sie Daten lesen und Transaktionen im Polygon-Netzwerk schreiben, einschließlich des Abrufs von Daten und des Sendens von Transaktionen an das Polygon-Netzwerk.

Example

Das folgende Beispiel zeigt, wie Sie mithilfe von RPC Informationen über den neuesten Block abrufen können. `eth_getBlockByNumber` Ändern Sie die hervorgehobenen Variablen in Ihre eigenen Eingaben oder wählen Sie eine der aufgelisteten RPC-Methoden und geben Sie die entsprechenden erforderlichen Eingaben ein.

1. Öffnen Sie die Managed Blockchain-Konsole unter https://console.aws.amazon.com/managedblockchain/.
2. Wählen Sie den RPC-Editor.

3. Wählen Sie im Bereich Anfrage die ***Blockchain Network***Option*POLYGON_MAINNET*.
4. Wählen Sie *eth_getBlockByNumber* als RPC-Methode.
5. Geben Sie **latest** als das Kennzeichen „Vollständige Transaktion“ ein ***Block number***und wählen Sie *False* es aus.
6. Wählen Sie dann Submit RPC aus.
7. Die Ergebnisse des `latest` Blocks erhalten Sie im Abschnitt Antwort. Anschließend können Sie die vollständigen Rohtransaktionen zur weiteren Analyse oder zur Verwendung in der Geschäftslogik für Ihre Anwendungen kopieren.

Weitere Informationen finden Sie im von AMB Access RPCs unterstützten Polygon

Richten Sie AMB Access Polygon JSON-RPC-Anfragen ein, indem Sie **awscurl**AWS CLI

Example

Signieren Sie Anfragen mit Ihren IAM-Benutzeranmeldedaten, indem Sie Signature Version 4 (Sigv4) verwenden, um Polygon-JSON-RPC-Anfragen an die AMB Access Polygon-Endpunkte zu stellen. Das `awscurl`Befehlszeilentool kann Ihnen helfen, Anfragen an Dienste zu signieren, die Sigv4 verwenden. AWS Weitere Informationen finden Sie in der Datei awscurl README.md.

Verwenden Sie für `awscurl` die Installation die für Ihr Betriebssystem geeignete Methode. Unter macOS HomeBrew ist die empfohlene Anwendung:

```
brew install awscurl
```

Wenn Sie das bereits installiert und konfiguriert haben AWS CLI, AWS-Region sind Ihre IAM-Benutzeranmeldedaten und der Standard in Ihrer Umgebung festgelegt und Sie haben Zugriff `awscurl` auf. Verwenden Sie`awscurl`, um eine Anfrage an das Polygon-Mainnet zu senden, indem Sie den RPC aufrufen. `eth_getBlockByNumber` Dieser Aufruf akzeptiert einen Zeichenkettenparameter, der der Blocknummer entspricht, für die Sie Informationen abrufen möchten.

Der folgende Befehl ruft die Blockdaten aus dem Polygon-Mainnet ab, indem er anhand der Blocknummer im `params` Array den spezifischen Block auswählt, für den die Header abgerufen werden sollen.

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

> **Tip**
>
> Sie können dieselbe Anfrage auch mithilfe `curl` der Token-basierten Zugriffsfunktion von AMB Access unter Verwendung von Tokens stellen. `Accessor` Weitere Informationen finden Sie unter Accessor-Token für tokenbasierten Zugriff erstellen und verwalten, um AMB Access Polygon-Anfragen zu stellen.
>
> ```
> curl -X POST -d '{"jsonrpc":"2.0", "id": "eth_getBlockByNumber-curltest",
>  "method":"eth_getBlockByNumber", "params":["latest", false] }'
>  'https://mainnet.polygon.managedblockchain.us-east-1.amazonaws.com?
> billingtoken=your-billing-token'
> ```

Die Antwort eines der Befehle gibt Informationen über den letzten Block zurück. Zur Veranschaulichung sehen Sie sich das folgende Beispiel an:

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

 "parentHash":"0x27d47bc2c47a6d329eb8aa62c1353f60e138fb0c596e3e8e9425de163afd6dec",

 "receiptsRoot":"0x394da96025e51cc69bbe3644bc4e1302942c2a6ca6bf0cf241a5724c74c063fd",

 "sha3Uncles":"0x1dcc4de8dec75d7aab85b567b6ccd41ad312451b948a7413f0a142fd40d49347",
      "size":"0xbd6b",
      "stateRoot":"0x7ca9363cfe9baf4d1c0dca3159461b2cca8604394e69b30af05d7d5c1beea6c3",
      "timestamp":"0x653ff542",
```

```
      "totalDifficulty":"0x33eb01dd","transactions":[...],

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

Stellen Sie Polygon-JSON-RPC-Anfragen in Node.js

Sie können Polygon JSON- aufrufen, RPCs indem Sie signierte Anfragen über HTTPS einreichen, um mithilfe des nativen https-Moduls in Node.js auf das Polygon-Mainnet-Netzwerk zuzugreifen, oder Sie können eine Drittanbieterbibliothek wie AXIOS verwenden. Die folgenden Beispiele für Node.js zeigen Ihnen, wie Sie Polygon-JSON-RPC-Anfragen an den AMB Access Polygon-Endpunkt sowohl mit Signature Version 4 (Sigv4) als auch mit tokenbasiertem Zugriff stellen. Im ersten Beispiel wird eine Transaktion von einer Adresse an eine andere gesendet, und im folgenden Beispiel werden Transaktionsdetails und Saldoinformationen aus der Blockchain angefordert.

Example

Um dieses Beispielskript Node.js auszuführen, müssen die folgenden Voraussetzungen erfüllt sein:

1. Sie müssen Node Version Manager (nvm) und Node.js auf Ihrem Computer installiert haben. Installationsanweisungen für Ihr Betriebssystem finden Sie hier.
2. Verwenden Sie den `node --version` Befehl und bestätigen Sie, dass Sie Node Version 18 oder höher verwenden. Bei Bedarf können Sie den `nvm install v18.12.0` Befehl, gefolgt vom Befehl, verwenden, um Version 18, die LTS-Version von Node, zu installieren. `nvm use v18.12.0`
3. Die Umgebungsvariablen `AWS_ACCESS_KEY_ID` und `AWS_SECRET_ACCESS_KEY` müssen die Anmeldeinformationen enthalten, die mit Ihrem Konto verknüpft sind.

   Exportieren Sie diese Variablen mithilfe der folgenden Befehle als Zeichenfolgen auf Ihrem Client. Ersetzen Sie die rot markierten Werte in den folgenden Zeichenketten durch entsprechende Werte aus Ihrem IAM-Benutzerkonto.

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

Nachdem Sie alle Voraussetzungen erfüllt haben, kopieren Sie die folgenden Dateien mit Ihrem bevorzugten Code-Editor in ein Verzeichnis in Ihrer lokalen Umgebung:

package.json

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

dispatch-evm-rpc.js

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

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

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

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

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

  try {
    //make the request using axios
    const response = await axios({
      ...signedRequest,
      url: url,
      data: req.body,
    });
    return response.data;
  } catch (error) {
    console.error("Something went wrong: ", error);
  }
};

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

sendTx.js

> **Warning**
>
> Der folgende Code verwendet einen fest codierten privaten Schlüssel, um ein Wallet zu generieren, das Signer nur zu `Ethers.js` Demonstrationszwecken verwendet. Verwenden Sie diesen Code nicht in Produktionsumgebungen, da er über echtes Geld verfügt und ein Sicherheitsrisiko darstellt.
>
> Wenden Sie sich bei Bedarf an Ihr Account-Team, um Sie über bewährte Methoden für Wallet und Signer zu informieren.

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

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

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

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

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

  //use this wallet to send a transaction of POL from one address to another
  const tx = await signer.sendTransaction({
    to: to,
    value: ethers.parseUnits("0.0001", "ether"),
  });

  console.log(tx);
};

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

readTx.js

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

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

  //set RPC request body to get transaction details
```

```
  let getTransactionByHash = {
    id: "1",
    jsonrpc: "2.0",
    method: "eth_getTransactionByHash",
    params: [txHash],
  };

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

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

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

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

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

Sobald diese Dateien in Ihrem Verzeichnis gespeichert sind, installieren Sie die Abhängigkeiten, die für die Ausführung des Codes erforderlich sind, mit dem folgenden Befehl:

```
npm install
```

Senden Sie eine Transaktion in Node.js

Im vorherigen Beispiel wird das native Polygon-Mainnet-Token (POL) von einer Adresse an eine andere gesendet, indem eine Transaktion signiert und mithilfe von AMB Access Polygon an das Polygon-Mainnet gesendet wird. Verwenden Sie dazu das `sendTx.js` Skript, das eine beliebte Bibliothek für die Interaktion mit Ethereum `Ethers.js` und Ethereum-kompatiblen Blockchains wie Polygon verwendet. Sie müssen drei Variablen im Code ersetzen, sofern sie rot markiert sind, darunter das `billingToken` für Ihren Accessor-Token für den tokenbasierten Zugriff, den privaten

Schlüssel, mit dem Sie die Transaktion signieren, und die Adresse des Empfängers, der die POL erhält.

> **Tip**
>
> Wir empfehlen Ihnen, zu diesem Zweck einen neuen privaten Schlüssel (Wallet) zu erstellen, anstatt ein vorhandenes Wallet wiederzuverwenden, um das Risiko eines Geldverlusts auszuschließen. Sie können die Wallet-Klassenmethode createRandom () der Ethers-Bibliothek verwenden, um ein Wallet zu generieren, mit dem Sie testen können. Wenn Sie POL vom Polygon-Mainnet anfordern müssen, können Sie außerdem den öffentlichen POL-Faucet verwenden, um eine kleine Menge für Tests anzufordern.

Sobald Sie den privaten Schlüssel einer finanzierten Wallet und die Adresse des Empfängers zum Code hinzugefügt haben, führen Sie den folgenden Code aus, um eine Transaktion für .0001 POL zu signieren, die von Ihrer Adresse an eine andere gesendet und an Polygon Mainnet gesendet wird, wobei der `eth_sendRawTransaction` JSON-RPC mithilfe des AMB Access Polygon aufgerufen wird. `billingToken`

```
node sendTx.js
```

Die Antwort, die wir zurückerhalten haben, sieht wie folgt aus:

```
TransactionResponse {
provider: JsonRpcProvider {},
blockNumber: null,
blockHash: null,
index: undefined,
hash: '0x8d7538b4841261c5120c0a4dd66359e8ee189e7d1d34ac646a1d9923********',
type: 2,
to: '0xd2bb4f4f1BdC4CB54f715C249Fc5a991********',
from: '0xcf2C679AC6cb7de09Bf6BB6042ecCF05********',
nonce: 2,
gasLimit: 21000n,
gasPrice: undefined,
maxPriorityFeePerGas: 16569518669n,
maxFeePerGas: 16569518685n,
data: '0x',
value: 100000000000000n,
chainId: 80001n,
```

```
signature: Signature {
r: "0x1b90ad9e9e4e005904562d50e904f9db10430a18b45931c059960ede337238ee",
s: "0x7df3c930a964fd07fed4a59f60b4ee896ffc7df4ea41b0facfe82b470db448b7",
yParity: 0,
networkV: null
},
accessList: []
}
```

Die Antwort stellt den Transaktionsbeleg dar. Speichern Sie den Wert der Immobilie`hash`. Dies ist die Kennung für die Transaktion, die Sie gerade an die Blockchain übermittelt haben. Sie verwenden diese Eigenschaft im Beispiel für eine Lesetransaktion, um zusätzliche Details zu dieser Transaktion aus dem Polygon-Mainnet zu erhalten.

Beachten Sie, dass die `blockNumber` und `null` in der Antwort `blockHash` stehen. Dies liegt daran, dass die Transaktion noch nicht in einem Block im Polygon-Netzwerk aufgezeichnet wurde. Beachten Sie, dass diese Werte später definiert werden und Sie sie möglicherweise sehen, wenn Sie die Transaktionsdetails im folgenden Abschnitt anfordern.

Lesen Sie eine Transaktion in Node.js

In diesem Abschnitt fordern Sie die Transaktionsdetails für die zuvor übermittelte Transaktion an und rufen den POL-Saldo für die Empfängeradresse mithilfe von Leseanfragen an das Polygon-Mainnet mithilfe von AMB Access Polygon ab. Ersetzen Sie in der `readTx.js` Datei die Variable *`your-transaction-id`* mit der Bezeichnung, die Sie aus der Antwort gespeichert haben, die `hash` Sie beim Ausführen des Codes im vorherigen Abschnitt gespeichert haben.

Dieser Code verwendet ein Hilfsprogramm`dispatch-evm-rpc.js`, das HTTPS-Anfragen an AMB Access Polygon mit den erforderlichen Signature Version 4 (Sigv4) -Modulen aus dem AWS SDK signiert und Anfragen über den weit verbreiteten HTTP-Client AXIOS sendet.

Die zurückgesendete Antwort sieht wie folgt aus:

```
TX DETAILS: {
blockHash: '0x59433e0096c783acab0659175460bb3c919545ac14e737d7465b3ddc********',
blockNumber: '0x28b4059',
from: '0xcf2c679ac6cb7de09bf6bb6042eccf05b7fa1394',
gas: '0x5208',
gasPrice: '0x3db9eca5d',
maxPriorityFeePerGas: '0x3db9eca4d',
maxFeePerGas: '0x3db9eca5d',
```

```
hash: '0x8d7538b4841261c5120c0a4dd66359e8ee189e7d1d34ac646a1d9923********',
input: '0x',
nonce: '0x2',
to: '0xd2bb4f4f1bdc4cb54f715c249fc5a991********',
transactionIndex: '0x0',
value: '0x5af3107a4000',
type: '0x2',
accessList: [],
chainId: '0x13881',
v: '0x0',
r: '0x1b90ad9e9e4e005904562d50e904f9db10430a18b45931c059960ede337238ee',
s: '0x7df3c930a964fd07fed4a59f60b4ee896ffc7df4ea41b0facfe82b470db448b7'
} BALANCE: 0.0003
```

Die Antwort stellt die Transaktionsdetails dar. Beachten Sie, dass die `blockHash` und jetzt wahrscheinlich definiert `blockNumber` sind. Dies weist darauf hin, dass die Transaktion in einem Block aufgezeichnet wurde. Wenn diese Werte immer noch vorhanden sind`null`, warten Sie einige Minuten und führen Sie dann den Code erneut aus, um zu überprüfen, ob Ihre Transaktion in einem Block enthalten ist. Schließlich wird die hexadezimale Darstellung des Saldos der Empfängeradresse (0x110d9316ec000) mithilfe der `formatEther()` Ethers-Methode in eine Dezimalzahl umgewandelt. Dabei wird der Hexadezimalwert in eine Dezimalzahl umgewandelt und die Dezimalstellen um 18 (10^18) verschoben, um den wahren Saldo in POL zu erhalten.

> **Tip**
>
> In den vorangegangenen Codebeispielen wird zwar veranschaulicht, wie Sie Node.js, Ethers und Axios verwenden können, um einige der unterstützten RPCs JSON-on-AMB-Zugriffspolygone zu verwenden. Sie können die Beispiele jedoch ändern und anderen Code schreiben, um Ihre Anwendungen auf Polygon mithilfe dieses Dienstes zu erstellen. Eine vollständige Liste der unterstützten JSON-Daten RPCs auf AMB Access Polygon finden Sie unter. Verwaltete Blockchain-API und RPCs JSON-Unterstützung durch AMB Access Polygon

Accessor-Token für tokenbasierten Zugriff erstellen und verwalten, um AMB Access Polygon-Anfragen zu stellen

Sie können Accessor-Token auch verwenden, um JSON-RPC-Aufrufe an die Polygon-Netzwerkendpunkte als praktische Alternative zum Signaturprozess Signature Version 4 (Sigv4) zu tätigen. Sie müssen eines der `BILLING_TOKEN` von Ihnen erstellten Accessor-Token angeben und als Parameter bei Ihren Aufrufen hinzufügen.

> **Important**
>
> - Wenn Sie Sicherheit und Überprüfbarkeit der Benutzerfreundlichkeit vorziehen, verwenden Sie stattdessen den SigV4-Signaturprozess.
> - Sie können RPCs mithilfe von Signature Version 4 (Sigv4) und tokenbasiertem Zugriff auf das Polygon-JSON zugreifen. Wenn Sie sich jedoch dafür entscheiden, beide Protokolle zu verwenden, wird Ihre Anfrage abgelehnt.
> - Sie dürfen Accessor-Token niemals in benutzerorientierte Anwendungen einbetten.

In der Konsole wird auf der Seite Token-Accessors eine Liste aller Accessor-Token angezeigt, die Sie verwenden können, um AMB Access Polygon JSON-RPC-Aufrufe von Ihrem Absendercode auf einem Client aus durchzuführen. AWS-Konto

Weitere Informationen zu AMB Access Polygon JSON-RPC-Anfragen finden Sie unter. Verwaltete Blockchain-API und RPCs JSON-Unterstützung durch AMB Access Polygon

Sie können Accessor-Token mit dem erstellen und verwalten. AWS Management Console
Sie können Accessor-Token auch mithilfe der folgenden API-Operationen erstellen und verwalten:`CreateAccessor`, `GetAccessorListAccessors`, und. `DeleteAccessor` A `BILLING_TOKEN` ist eine Eigenschaft des Accessors. Diese `BILLING_TOKEN` Eigenschaft wird verwendet, um Ihren Accessor zu verfolgen und AMB Access Polygon JSON-RPC-Anfragen von Ihnen abzurechnen. AWS-Konto

Alle API-Aktionen im Zusammenhang mit der Erstellung und Verwaltung von Accessor-Token sind auch über, und verfügbar. AWS Management Console AWS CLI SDKs

Erstellen eines Accessor-Tokens für den tokenbasierten Zugriff

Sie können ein Accessor-Token erstellen und es verwenden, um AMB Access Polygon API-Aufrufe auf jedem AMB Access Polygon-Knoten in Ihrem durchzuführen. AWS-Konto

Erstellen Sie ein Accessor-Token, um AMB Access Polygon JSON-RPC-Anfragen zu stellen, indem Sie AWS Management Console

1. Öffnen Sie die Managed Blockchain-Konsole unter. https://console.aws.amazon.com/managedblockchain/
2. Wählen Sie Token Accessors.
3. Wählen Sie Create Accessor.
4. Wählen Sie ein gültiges Polygon-Blockchain-Netzwerk.
5. Fügen Sie optional Tags für Ihren Accessor hinzu.
6. Wählen Sie Create Accessor, um ein neues Accessor-Token zu erstellen.

Erstellen Sie ein Accessor-Token, um AMB Access Polygon JSON-RPC-Anfragen zu stellen. Verwenden Sie dazu AWS CLI

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

Der vorherige Befehl gibt das `AccessorId` zusammen mit dem zurück`BillingToken`, wie im folgenden Beispiel gezeigt.

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

Das Schlüsselelement in Ihrer Antwort ist der`BillingToken`. Sie können diese Eigenschaft verwenden, um AMB Access Polygon JSON-RPC-Aufrufe zu tätigen. Einige Werte im Beispiel wurden aus Sicherheitsgründen verschleiert, werden aber in den tatsächlichen Antworten vollständig angezeigt.

> **Note**
>
> Nachdem der Vorgang ausgeführt wurde, stellt Managed Blockchain das Token für Sie bereit und konfiguriert es. Die Dauer dieses Prozesses hängt von vielen Variablen ab.

Details eines Accessor-Tokens anzeigen

Sie können die Eigenschaften für jedes Accessor-Token anzeigen, das Ihnen AWS-Konto gehört. Sie können beispielsweise die Accessor-ID oder den Amazon-Ressourcennamen (ARN) des Accessors anzeigen. Sie können auch den Status, den Typ, das Erstellungsdatum und die anzeigen. `BillingToken`

Um die Informationen eines Accessor-Tokens mit dem AWS Management Console

1. Öffnen Sie die Managed Blockchain-Konsole unter https://console.aws.amazon.com/managedblockchain/.
2. Wählen Sie im Navigationsbereich Token Accessors aus.
3. Wählen Sie die Accessor-ID des Tokens aus der Liste aus.

Die Seite mit den Token-Details wird angezeigt. Auf dieser Seite können Sie die Eigenschaften des Tokens einsehen.

Um die Informationen eines Accessor-Tokens mit dem AWS CLI

Führen Sie den folgenden Befehl aus, um die Details eines Accessor-Tokens anzuzeigen. Ersetzen Sie die Werte von `--accessor-id` durch Ihre Accessor-ID.

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

Die `BillingToken` und andere Schlüsseleigenschaften werden wie im folgenden Beispiel gezeigt zurückgegeben. Einige Werte im Beispiel wurden aus Sicherheitsgründen verschleiert, tauchen aber vollständig in den tatsächlichen Antworten auf.

```
{
  "Accessor": {
  "Id": "ac-NGQ6QNKXLNEBXD3UI6********",
  "Type": "BILLING_TOKEN",
```

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

Löschen eines Accessor-Tokens

Wenn Sie ein Accessor-Token löschen, ändert sich der Status des Tokens von `AVAILABLE` in den `PENDING_DELETION` Status. Sie können kein Accessor-Token mit dem `PENDING_DELETION` Status verwenden.

Um ein Accessor-Token mit dem zu löschen AWS Management Console

1. Öffnen Sie die Managed Blockchain-Konsole unter https://console.aws.amazon.com/managedblockchain/.
2. Wählen Sie im Navigationsbereich Token Accessors aus.
3. Wählen Sie das gewünschte Accessor-Token aus der Liste aus.
4. Wählen Sie Löschen.
5. Bestätigen Sie Ihre Auswahl.

Sie kehren mit Ihrem gelöschten Accessor-Token zur Token-Accessor-Seite zurück. Auf der Seite wird der Status angezeigt. `PENDING_DELETION`

Um ein Accessor-Token mit dem zu löschen AWS CLI

Das folgende Beispiel zeigt, wie ein Token gelöscht wird. Verwenden Sie den `delete-accessor` Befehl, um ein Token zu löschen. Stellen Sie den Wert von `--accessor-id` mit Ihrer Accessor-ID ein.

Löschen eines Accessor-Tokens mit der CLI AWS

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

Wenn dieser Befehl erfolgreich ausgeführt wird, werden keine Nachrichten zurückgegeben.

Verwaltete Blockchain-API und RPCs JSON-Unterstützung durch AMB Access Polygon

Amazon Managed Blockchain bietet API-Operationen für die Erstellung und Verwaltung von Token-Accessors für AMB Access Polygon. Weitere Informationen finden Sie im Referenzhandbuch zur Managed Blockchain API.

Das folgende Thema enthält eine Liste und eine Referenz der Polygon-JSON-DateienRPCs , die AMB Access Polygon unterstützt. Zu jedem unterstützten JSON-RPC gibt es eine kurze Beschreibung seiner Verwendung. Sie verwenden Polygon JSON-, RPCs um Smart-Contract-Daten abzufragen und abzurufen, Transaktionsdetails abzurufen, Transaktionen einzureichen und andere Hilfsprogramme wie die Ablaufverfolgung von Transaktionen und die Schätzung von Gebühren durchzuführen.

AMB Access Polygon unterstützt die folgenden JSON-RPC-Methoden. Jedes unterstützte JSON-RPC hat eine Kategorie und eine kurze Beschreibung seines Dienstprogramms und seiner Standardanforderungsquoten. Besondere Überlegungen zur Verwendung der JSON-RPC-Methode mit Amazon Managed Blockchain werden gegebenenfalls angegeben.

> **Note**
>
> - Alle Methoden, die nicht aufgeführt sind, werden nicht unterstützt.
> - Wenn Sie Polygon JSON- RPCs auf Amazon Managed Blockchain aufrufen, können Sie dies über eine HTTPS-Verbindung tun, die mit dem Signature Version 4-Signaturprozess authentifiziert wurde. Das bedeutet, dass nur autorisierte IAM-Prinzipale im AWS Konto Polygon-JSON-RPC-Aufrufe tätigen können. Zu diesem Zweck müssen beim AWS Anruf Anmeldeinformationen (eine Zugriffsschlüssel-ID und ein geheimer Zugriffsschlüssel) bereitgestellt werden.
> - Sie können den tokenbasierten Zugriff auch als bequeme Alternative zum Signaturprozess mit Signature Version 4 (Sigv4) verwenden. Wenn Sie Sicherheit und Überprüfbarkeit der Benutzerfreundlichkeit vorziehen, verwenden Sie stattdessen den SigV4-Signaturprozess. Wenn Sie jedoch sowohl SigV4 als auch den tokenbasierten Zugriff verwenden, funktionieren Ihre Anfragen nicht.
> - JSON-RPC-Batchanfragen werden auf Amazon Managed Blockchain (AMB) Access Polygon für diese Vorschau nicht unterstützt.

- In der Spalte Kontingente in der folgenden Tabelle sind die Kontingente für jeden JSON-RPC aufgeführt. Kontingente werden in Anfragen pro Sekunde (RPS) pro Region pro Polygon-Netzwerk (Mainnet) für jeden JSON-RPC festgelegt.

  Um Ihr Kontingent zu erhöhen, wenden Sie sich bitte an. Support Um Kontakt aufzunehmen Support, melden Sie sich bei der an AWS Support Center Console. Wählen Sie Create case (Fall erstellen) aus. Wählen Sie Technisch. Wählen Sie Managed Blockchain als Ihren Service. Wähle Access:Polygon als Kategorie und General Guidance als Schweregrad. Geben Sie RPC Quota als Betreff ein und listen Sie im Textfeld Beschreibung das JSON-RPC und die für Ihre Bedürfnisse geltenden Kontingentgrenzen in RPS pro Polygon-Netzwerk pro Region auf. Reichen Sie Ihren Fall ein.

Kategorie	JSON-RPC	Beschreibung	Überlegungen
Äther	ETH_Blocknummer	Gibt die Nummer des letzten Blocks zurück.	
	eth_call	Führt sofort einen neuen Nachricht enaufruf aus, ohne eine Transaktion in der Blockchain zu erstellen.	`eth_call`verbrauch t 0 Gas, hat aber einen Gasparame ter für Nachrichten, die dies erfordern.
	ETH_ChainID	Gibt einen Integer-Wert für den aktuell konfigurierten `Chain Id` Wert zurück, der in EIP-155 eingeführ t wurde. Gibt zurück`None`, wenn kein verfügbar `Chain Id` ist.	

Kategorie	JSON-RPC	Beschreibung	Überlegungen
	ETH_EstimateGas	Schätzt das Gas, das für eine Transaktion benötigt wird, und gibt es zurück, ohne die Transaktion zur Blockchain hinzuzufügen.	
	Verlauf von ETH_FEE	Gibt eine Sammlung historischer Gasinformationen zurück.	
	ETH_GasPrice	Gibt den aktuellen Preis pro Gas in Wei zurück.	
	ETH_GetBalance	Gibt den Saldo eines Kontos für die angegebene Kontoadresse und Block-ID zurück.	
	eth_ Hash getBlockBy	Gibt Informationen über den Block zurück, der mit dem Block-Hash angegeben wurde.	

Kategorie	JSON-RPC	Beschreibung	Überlegungen
	eth_ Zahl getBlockBy	Gibt Informationen über den Block zurück, der anhand der Blocknummer angegeben wurde.	
	eth_ getBlockReceipts	Gibt unter Verwendung der Blocknummer Quittungen über den angegebenen Block zurück.	
	eth_ getBlockTransaction CountByHash	Gibt die Anzahl der Transaktionen in dem Block zurück, der mit dem Block-Hash angegeben wurde.	
	eth_ getBlockTransaction CountByNu mber	Gibt die Anzahl der Transaktionen in dem Block zurück, der anhand der Blocknummer angegeben wurde.	
	ETH_GetCode	Gibt den Code an der angegebenen Kontoadresse und Block-ID zurück.	

Kategorie	JSON-RPC	Beschreibung	Überlegungen
	ETH_GetLogs	Gibt ein Array aller Logs für ein angegebenes Filterobjekt zurück.	Sie können `eth_getloqs` Anfragen für jeden Blockbere ich mit einem Blockbereich von standardmäßig 1 KB stellen, wenn eine Vertragsa dresse angegeben wird. Verträge mit hoher Aktivität können auf kleinere Blockbere iche beschränk t werden. Wenn keine Vertragsa dresse angegeben wird, beträgt der Blockbereich 8.
	eth_ getRawTransaction ByHash	Gibt die Rohform der von der angegeben en Transakti on zurück. `transacti on_hash`	

Kategorie	JSON-RPC	Beschreibung	Überlegungen
	eth_ getStorageAt	Gibt den Wert der angegebenen Speicherposition für die angegebene Kontoadresse und Block-ID zurück.	
	eth_ getTransactionBy BlockHash AndIndex	Gibt Informationen über eine Transakti on zurück, die den angegebenen Blockhash und die Transaktionsindexp osition verwendet.	
	eth_ getTransactionBy BlockNumb erAndIndex	Gibt Informationen über eine Transakti on zurück, wobei die angegeben e Blocknummer und die Transakti onsindexposition verwendet werden.	
	eth_ Hash getTransactionBy	Gibt Informati onen über die Transaktion mit dem angegebenen Transaktions-Hash zurück.	

Kategorie	JSON-RPC	Beschreibung	Überlegungen
	eth_ getTransactionCount	Gibt die Anzahl der Transaktionen zurück, die von der angegeben en Adresse und Block-ID gesendet wurden.	
	eth_ getTransactionReceipt	Gibt den Empfang der Transaktion unter Verwendung des angegebenen Transaktions-Hash zurück.	
	eth_ getUncleBy BlockHashAndIndex	Gibt Informati onen über den angegebenen Uncle-Block zurück, der mit dem Block-Has h und der Uncle-Index-Position angegeben wurde.	
	eth_ getUncleBy BlockNumberAndIndex	Gibt Informati onen über den angegebenen Uncle-Block zurück, der anhand der Blocknumm er und der Uncle-Indexposition angegeben wurde.	

Kategorie	JSON-RPC	Beschreibung	Überlegungen
	eth_ getUncleCount ByBlockHash	Gibt die Anzahl der Zählungen in dem Onkel zurück, der mit dem Uncle-Hash angegeben wurde.	
	eth_ getUncleCount ByBlockNumber	Gibt die Anzahl der Zählungen im Onkel zurück, der anhand der Onkelnummer angegeben wurde.	
	eth_ maxPriorityFee PerGas	Gibt die Gebühr pro Gas zurück. Dabei handelt es sich um eine Schätzung, wie viel Sie als Vorzugsgebühr oder „Trinkgeld“ zahlen können, damit eine Transaktion in den aktuellen Block aufgenommen wird.	Im Allgemeinen verwenden Sie den Wert, der von dieser Methode zurückgegeben wird, um den Wert `maxFeePerGas` in der nachfolgenden Transaktion festzulegen, die Sie einreichen.
	ETH_ProtocolVersion	Gibt die aktuelle Version des Ethereum-Protokolls zurück.	

Kategorie	JSON-RPC	Beschreibung	Überlegungen
	eth_ sendRawTransaction	Erstellt eine neue Nachrichtenaufruftransaktion oder eine Vertragserstellung für signierte Transaktionen.	Managed Blockchain unterstützt nur Rohtransaktionen. Sie müssen Transaktionen erstellen und signieren, bevor Sie sie senden können.
Debugger	debug_ Hash traceBlockBy	Gibt die mögliche Nummer des Ablaufverfolgungsergebnisses zurück, indem alle Transaktionen in dem durch den Block-Hash angegebenen Block mit einem Tracer ausgeführt werden (Trace-Modus erforderlich).	

Kategorie	JSON-RPC	Beschreibung	Überlegungen
	debug_ Nummer traceBlockBy	Gibt das Tracing-Ergebnis zurück, indem alle Transakti onen in dem durch Nummer angegebenen Block mit einem Tracer ausgeführt werden (Trace-Mo dus erforderlich).	
	Debug_TraceCall	Gibt die Anzahl der möglichen Ablaufverfolgungse rgebnisse zurück, wenn ein eth-Aufru f im Kontext der angegebenen Blockausführung ausgeführt wird (Trace-Modus erforderlich).	
	debug_traceTransaction	Gibt alle Traces einer bestimmten Transaktion zurück (Trace-Modus erforderlich).	
Netz	net_version	Gibt die aktuelle Netzwerk-ID zurück.	

Kategorie	JSON-RPC	Beschreibung	Überlegungen
Trace	trace_block	Gibt einen vollständigen Stack-Trace aller aufgerufenen Opcodes aller Transaktionen zurück, die in einem Block enthalten waren.	
	trace_call	Gibt die Anzahl der möglichen Ablaufverfolgungsergebnisse zurück, wenn ein eth-Aufruf im Kontext der angegebenen Blockausführung ausgeführt wird (Trace-Modus erforderlich).	
	trace_transaction	Gibt alle Traces einer bestimmten Transaktion zurück (Trace-Modus erforderlich).	
Tx-Pool	txpool_content	Gibt alle ausstehenden Transaktionen und Transaktionen in der Warteschlange zurück.	

Kategorie	JSON-RPC	Beschreibung	Überlegungen
	txpool_status	Liefert die Anzahl aller Transakti onen, die derzeit noch nicht in die nächsten Blöcke aufgenommen werden müssen, sowie aller Transaktionen, die sich in der Warteschlange befinden (die nur für die future Ausführung geplant sind).	
Web	Web3_ClientVersion	Gibt die aktuelle Client-Version zurück.	

Polygon-Anwendungsfälle mit Amazon Managed Blockchain (AMB) Access Polygon

Die Polygon-Blockchain wird häufig zum Erstellen dezentraler Anwendungen (DApps) verwendet NFTs, die sich unter anderem auf Web3-Spiele und Tokenisierung beziehen. Dieses Thema enthält eine Liste einiger Anwendungsfälle, die Sie mit Amazon Managed Blockchain (AMB) Access Polygon implementieren können.

Themen

Analysieren Sie Polygon-NFT-Daten

Sie können Daten über Polygon sammeln NFTs, einschließlich Informationen wie Übertragungsereignisse und NFT-Metadaten für einen bestimmten Zeitraum. Sie können diese Daten dann analysieren, um Erkenntnisse darüber zu gewinnen, welche im Trend NFTs liegen oder welche Benutzer am häufigsten mit einer bestimmten Sammlung interagieren.

Weitere Informationen finden Sie unter Verwaltete Blockchain-API und RPCs JSON-Unterstützung durch AMB Access Polygon.

Support Sie NFT-Käufe

Sie können AMB Access Polygon verwenden, um Transaktionen für NFT-Käufe mithilfe von Initial Mint, Allowlists oder auf dem Sekundärmarkt einzureichen. Mithilfe einer Kombination anderer AWS Dienste können Sie dann Käufe mit Kreditkarten zulassen und Fiat- oder Kryptowährungen akzeptieren, was eine schnelle Abwicklung für alle Beteiligten ermöglicht.

Weitere Informationen finden Sie unter Verwaltete Blockchain-API und RPCs JSON-Unterstützung durch AMB Access Polygon.

Erstellen Sie eine Polygon-Wallet

Sie können AMB Access Polygon verwenden, um wichtige Funktionen von Wallets für digitale Vermögenswerte zu erfüllen, z. B. das Lesen von Benutzer-Token-Guthaben aus intelligenten Verträgen in der Blockchain oder das Senden signierter Transaktionen an die Blockchain.

Weitere Informationen finden Sie unter Verwaltete Blockchain-API und RPCs JSON-Unterstützung durch AMB Access Polygon.

Wallet als Service

Sie können AMB Access Polygon verwenden, um mithilfe des unterstützten Polygon-JSON-Tools ein Betriebssystem zu entwickeln, das für die Unterstützung gängiger Wallet-Transaktionen wie die Überprüfung eines Saldos, die Übertragung von Vermögenswerten, das Senden von Vermögenswerten und Gebührenschätzungen wallet-as-a-service erforderlich ist. RPCs

Weitere Informationen finden Sie unter Verwaltete Blockchain-API und RPCs JSON-Unterstützung durch AMB Access Polygon.

Erlebnisse, die auf Tokens basieren

Sie können AMB Access Polygon verwenden, um tokengesteuerte Erlebnisse für Ihre Benutzer zu erstellen. Sie können beispielsweise nur den Besitzern einer bestimmten NFT den Zugriff auf einen Inhalt unter bestimmten Bedingungen gewähren. Um dies zu erreichen, müssen Sie die Blockchain lesen, um festzustellen, ob der NFT-Eigentümer der Adresse eines Benutzers ist.

Weitere Informationen finden Sie unter Verwaltete Blockchain-API und RPCs JSON-Unterstützung durch AMB Access Polygon.

Anleitungen für Amazon Managed Blockchain (AMB) Access Polygon

Bei den folgenden Tutorials, die in diesem Abschnitt hervorgehoben werden, handelt es sich um Community-Artikel, in AWS re:Post denen Sie anhand von exemplarischen Vorgehensweisen lernen, wie Sie einige allgemeine Aufgaben in der Polygon-Blockchain mithilfe von AMB Access Polygon ausführen können.

- Senden von Transaktionen mit AMB Access Polygon und web3.js
- Stellen Sie einen intelligenten Vertrag mit AMB Access Polygon und Hardhat Ignition bereit
- Interaktion mit einem intelligenten Vertrag
- Rufen Sie mithilfe von AMB Access Polygon- und Chainlink-Datenfeeds aktuelle Preisdaten außerhalb der Kette ab
- Analysieren Sie ERC-20-Token-Daten im Polygon Mainnet mit AMB Access

Sicherheit in Amazon Managed Blockchain (AMB) Access Polygon

Cloud-Sicherheit hat höchste AWS Priorität. Als AWS Kunde profitieren Sie von Rechenzentren und Netzwerkarchitekturen, die darauf ausgelegt sind, die Anforderungen der sicherheitssensibelsten Unternehmen zu erfüllen.

Sicherheit ist eine gemeinsame AWS Verantwortung von Ihnen und Ihnen. Das Modell der gemeinsamen Verantwortung beschreibt dies sowohl als Sicherheit in der Cloud als auch als Sicherheit in der Cloud:

- Sicherheit der Cloud — AWS ist verantwortlich für den Schutz der Infrastruktur, auf der AWS Dienste in der ausgeführt AWS Cloud werden. AWS bietet Ihnen auch Dienste, die Sie sicher nutzen können. Auditoren von Drittanbietern testen und überprüfen die Effektivität unserer Sicherheitsmaßnahmen im Rahmen der AWS -Compliance-Programme regelmäßig. Weitere Informationen zu den Compliance-Programmen, die für Amazon Managed Blockchain (AMB) Access Polygon gelten, finden Sie unter AWS Services in Scope by Compliance Program.
- Sicherheit in der Cloud — Ihre Verantwortung richtet sich nach dem AWS Service, den Sie nutzen. Sie sind auch für andere Faktoren verantwortlich, etwa für die Vertraulichkeit Ihrer Daten, die Anforderungen Ihres Unternehmens und die geltenden Gesetze und Vorschriften.

Um Datenschutz, Authentifizierung und Zugriffskontrolle zu gewährleisten, verwendet Amazon Managed Blockchain AWS Funktionen und Funktionen des Open-Source-Frameworks, das in Managed Blockchain ausgeführt wird.

Diese Dokumentation hilft Ihnen zu verstehen, wie Sie das Modell der gemeinsamen Verantwortung bei der Verwendung von AMB Access Polygon anwenden können. Die folgenden Themen zeigen Ihnen, wie Sie AMB Access Polygon konfigurieren, um Ihre Sicherheits- und Compliance-Ziele zu erreichen. Sie erfahren auch, wie Sie andere AWS Dienste nutzen können, die Sie bei der Überwachung und Sicherung Ihrer AMB Access Polygon-Ressourcen unterstützen.

Themen

Datenschutz in Amazon Managed Blockchain (AMB) Access Polygon

Das Modell der AWS gemeinsamen Verantwortung gilt für den Datenschutz in Amazon Managed Blockchain (AMB) Access Polygon. Wie in diesem Modell beschrieben, AWS ist es für den Schutz der globalen Infrastruktur verantwortlich, auf der alle Systeme laufen. AWS Cloud Sie sind dafür verantwortlich, die Kontrolle über Ihre in dieser Infrastruktur gehosteten Inhalte zu behalten. Sie sind auch für die Sicherheitskonfiguration und die Verwaltungsaufgaben für die von Ihnen verwendeten AWS-Services verantwortlich. Weitere Informationen zum Datenschutz finden Sie unter Häufig gestellte Fragen zum Datenschutz. Informationen zum Datenschutz in Europa finden Sie im Blog-Beitrag AWS -Modell der geteilten Verantwortung und in der DSGVO im AWS -Sicherheitsblog.

Aus Datenschutzgründen empfehlen wir, dass Sie AWS-Konto Anmeldeinformationen schützen und einzelne Benutzer mit AWS IAM Identity Center oder AWS Identity and Access Management (IAM) einrichten. So erhält jeder Benutzer nur die Berechtigungen, die zum Durchführen seiner Aufgaben erforderlich sind. Außerdem empfehlen wir, die Daten mit folgenden Methoden schützen:

- Verwenden Sie für jedes Konto die Multi-Faktor-Authentifizierung (MFA).
- Verwenden Sie SSL/TLS, um mit Ressourcen zu kommunizieren. AWS Wir benötigen TLS 1.2 und empfehlen TLS 1.3.
- Richten Sie die API und die Protokollierung von Benutzeraktivitäten mit ein. AWS CloudTrail Informationen zur Verwendung von CloudTrail Pfaden zur Erfassung von AWS Aktivitäten finden Sie unter Arbeiten mit CloudTrail Pfaden im AWS CloudTrail Benutzerhandbuch.
- Verwenden Sie AWS Verschlüsselungslösungen zusammen mit allen darin enthaltenen Standardsicherheitskontrollen AWS-Services.
- Verwenden Sie erweiterte verwaltete Sicherheitsservices wie Amazon Macie, die dabei helfen, in Amazon S3 gespeicherte persönliche Daten zu erkennen und zu schützen.
- Wenn Sie für den Zugriff AWS über eine Befehlszeilenschnittstelle oder eine API FIPS 140-3-validierte kryptografische Module benötigen, verwenden Sie einen FIPS-Endpunkt. Weitere Informationen über verfügbare FIPS-Endpunkte finden Sie unter Federal Information Processing Standard (FIPS) 140-3.

Wir empfehlen dringend, in Freitextfeldern, z. B. im Feld Name, keine vertraulichen oder sensiblen Informationen wie die E-Mail-Adressen Ihrer Kunden einzugeben. Dies gilt auch, wenn Sie mit AMB Access Polygon oder anderen Geräten AWS-Services über die Konsole, API oder arbeiten. AWS

CLI AWS SDKs Alle Daten, die Sie in Tags oder Freitextfelder eingeben, die für Namen verwendet werden, können für Abrechnungs- oder Diagnoseprotokolle verwendet werden. Wenn Sie eine URL für einen externen Server bereitstellen, empfehlen wir dringend, keine Anmeldeinformationen zur Validierung Ihrer Anforderung an den betreffenden Server in die URL einzuschließen.

Datenverschlüsselung

Datenverschlüsselung verhindert, dass unbefugte Benutzer Daten aus einem Blockchain-Netzwerk und den zugehörigen Datenspeichersystemen lesen. Dazu gehören Daten, die bei der Übertragung durch das Netzwerk möglicherweise abgefangen werden, sogenannte Daten bei der Übertragung.

Verschlüsselung während der Übertragung

Standardmäßig verwendet Managed Blockchain eine HTTPS/TLS-Verbindung, um alle Daten zu verschlüsseln, die von einem Client-Computer übertragen werden, auf dem die beiden Dienstendpunkte ausgeführt werden. AWS CLI AWS

Sie müssen nichts tun, um die Verwendung von HTTPS/TLS zu aktivieren. Sie ist immer aktiviert, es sei denn, Sie deaktivieren sie explizit für einen einzelnen AWS CLI Befehl, indem Sie den Befehl verwenden. `--no-verify-ssl`

Identitäts- und Zugriffsmanagement für Amazon Managed Blockchain (AMB) Access Polygon

AWS Identity and Access Management (IAM) hilft einem Administrator AWS-Service , den Zugriff auf AWS Ressourcen sicher zu kontrollieren. IAM-Administratoren kontrollieren, wer authentifiziert (angemeldet) und autorisiert werden kann (über Berechtigungen verfügt), um AMB Access Polygon-Ressourcen zu verwenden. IAM ist ein Programm AWS-Service , das Sie ohne zusätzliche Kosten nutzen können.

Themen

- Beispiele für identitätsbasierte Richtlinien für Amazon Managed Blockchain (AMB) Access Polygon
- Fehlerbehebung bei Identität und Zugriff auf Amazon Managed Blockchain (AMB) Access Polygon

Zielgruppe

Wie Sie AWS Identity and Access Management (IAM) verwenden, hängt von der Arbeit ab, die Sie in AMB Access Polygon ausführen.

Dienstbenutzer — Wenn Sie den AMB Access Polygon-Service für Ihre Arbeit verwenden, stellt Ihnen Ihr Administrator die Anmeldeinformationen und Berechtigungen zur Verfügung, die Sie benötigen. Da Sie für Ihre Arbeit mehr Funktionen von AMB Access Polygon verwenden, benötigen Sie möglicherweise zusätzliche Berechtigungen. Wenn Sie die Funktionsweise der Zugriffskontrolle nachvollziehen, wissen Sie bereits, welche Berechtigungen Sie von Ihrem Administrator anfordern müssen. Wenn Sie in AMB Access Polygon nicht auf ein Feature zugreifen können, finden Sie weitere Informationen unter. Fehlerbehebung bei Identität und Zugriff auf Amazon Managed Blockchain (AMB) Access Polygon

Serviceadministrator — Wenn Sie in Ihrem Unternehmen für die AMB Access Polygon-Ressourcen verantwortlich sind, haben Sie wahrscheinlich vollen Zugriff auf AMB Access Polygon. Es ist Ihre Aufgabe, zu bestimmen, auf welche Funktionen und Ressourcen von AMB Access Polygon Ihre Servicebenutzer zugreifen sollen. Anschließend müssen Sie Anforderungen an Ihren IAM-Administrator senden, um die Berechtigungen der Servicebenutzer zu ändern. Lesen Sie die Informationen auf dieser Seite, um die Grundkonzepte von IAM nachzuvollziehen. Weitere Informationen darüber, wie Ihr Unternehmen IAM mit AMB Access Polygon verwenden kann, finden Sie unter. So funktioniert Amazon Managed Blockchain (AMB) Access Polygon mit IAM

IAM-Administrator — Wenn Sie ein IAM-Administrator sind, möchten Sie vielleicht mehr darüber erfahren, wie Sie Richtlinien zur Verwaltung des Zugriffs auf AMB Access Polygon schreiben können. Beispiele für identitätsbasierte AMB Access Polygon-Richtlinien, die Sie in IAM verwenden können, finden Sie unter. Beispiele für identitätsbasierte Richtlinien für Amazon Managed Blockchain (AMB) Access Polygon

Authentifizierung mit Identitäten

Authentifizierung ist die Art und Weise, wie Sie sich mit Ihren Identitätsdaten anmelden. AWS Sie müssen als IAM-Benutzer authentifiziert (angemeldet AWS) sein oder eine IAM-Rolle annehmen. Root-Benutzer des AWS-Kontos

Sie können sich AWS als föderierte Identität anmelden, indem Sie Anmeldeinformationen verwenden, die über eine Identitätsquelle bereitgestellt wurden. AWS IAM Identity Center (IAM Identity Center) -Benutzer, die Single Sign-On-Authentifizierung Ihres Unternehmens und Ihre Google- oder Facebook-Anmeldeinformationen sind Beispiele für föderierte Identitäten. Wenn Sie sich als Verbundidentität anmelden, hat der Administrator vorher mithilfe von IAM-Rollen einen Identitätsverbund eingerichtet. Wenn Sie über den Verbund darauf zugreifen AWS , übernehmen Sie indirekt eine Rolle.

Je nachdem, welcher Benutzertyp Sie sind, können Sie sich beim AWS Management Console oder beim AWS Zugangsportal anmelden. Weitere Informationen zur Anmeldung finden Sie AWS unter So melden Sie sich bei Ihrem an AWS-Konto im AWS-Anmeldung Benutzerhandbuch.

Wenn Sie AWS programmgesteuert darauf zugreifen, AWS stellt es ein Software Development Kit (SDK) und eine Befehlszeilenschnittstelle (CLI) bereit, um Ihre Anfragen mithilfe Ihrer Anmeldeinformationen kryptografisch zu signieren. Wenn Sie keine AWS Tools verwenden, müssen Sie Anfragen selbst signieren. Weitere Informationen zur Verwendung der empfohlenen Methode für die Selbstsignierung von Anforderungen finden Sie unter AWS Signature Version 4 für API-Anforderungen im IAM-Benutzerhandbuch.

Unabhängig von der verwendeten Authentifizierungsmethode müssen Sie möglicherweise zusätzliche Sicherheitsinformationen bereitstellen. AWS Empfiehlt beispielsweise, die Multi-Faktor-Authentifizierung (MFA) zu verwenden, um die Sicherheit Ihres Kontos zu erhöhen. Weitere Informationen finden Sie unter Multi-Faktor-Authentifizierung im AWS IAM Identity Center - Benutzerhandbuch und AWS Multi-Faktor-Authentifizierung (MFA) in IAM im IAM-Benutzerhandbuch.

AWS-Konto Root-Benutzer

Wenn Sie einen erstellen AWS-Konto, beginnen Sie mit einer Anmeldeidentität, die vollständigen Zugriff auf alle AWS-Services Ressourcen im Konto hat. Diese Identität wird als AWS-Konto Root-Benutzer bezeichnet. Sie können darauf zugreifen, indem Sie sich mit der E-Mail-Adresse und dem Passwort anmelden, mit denen Sie das Konto erstellt haben. Wir raten ausdrücklich davon ab, den Root-Benutzer für Alltagsaufgaben zu verwenden. Schützen Sie Ihre Root-Benutzer-Anmeldeinformationen. Verwenden Sie diese nur, um die Aufgaben auszuführen, die nur der Root-Benutzer ausführen kann. Eine vollständige Liste der Aufgaben, für die Sie sich als Root-Benutzer anmelden müssen, finden Sie unter Aufgaben, die Root-Benutzer-Anmeldeinformationen erfordern im IAM-Benutzerhandbuch.

Verbundidentität

Als bewährte Methode sollten menschliche Benutzer, einschließlich Benutzer, die Administratorzugriff benötigen, für den Zugriff AWS-Services mithilfe temporärer Anmeldeinformationen den Verbund mit einem Identitätsanbieter verwenden.

Eine föderierte Identität ist ein Benutzer aus Ihrem Unternehmensbenutzerverzeichnis, einem Web-Identitätsanbieter AWS Directory Service, dem Identity Center-Verzeichnis oder einem beliebigen Benutzer, der mithilfe AWS-Services von Anmeldeinformationen zugreift, die über eine Identitätsquelle bereitgestellt wurden. Wenn föderierte Identitäten darauf zugreifen AWS-Konten, übernehmen sie Rollen, und die Rollen stellen temporäre Anmeldeinformationen bereit.

Für die zentrale Zugriffsverwaltung empfehlen wir Ihnen, AWS IAM Identity Center zu verwenden. Sie können Benutzer und Gruppen in IAM Identity Center erstellen, oder Sie können eine Verbindung zu einer Gruppe von Benutzern und Gruppen in Ihrer eigenen Identitätsquelle herstellen und diese synchronisieren, um sie in all Ihren AWS-Konten Anwendungen zu verwenden. Informationen zu IAM Identity Center finden Sie unter Was ist IAM Identity Center? im AWS IAM Identity Center - Benutzerhandbuch.

IAM-Benutzer und -Gruppen

Ein IAM-Benutzer ist eine Identität innerhalb Ihres Unternehmens AWS-Konto , die über spezifische Berechtigungen für eine einzelne Person oder Anwendung verfügt. Wenn möglich, empfehlen wir, temporäre Anmeldeinformationen zu verwenden, anstatt IAM-Benutzer zu erstellen, die langfristige Anmeldeinformationen wie Passwörter und Zugriffsschlüssel haben. Bei speziellen Anwendungsfällen, die langfristige Anmeldeinformationen mit IAM-Benutzern erfordern, empfehlen wir jedoch, die Zugriffsschlüssel zu rotieren. Weitere Informationen finden Sie unter Regelmäßiges Rotieren von Zugriffsschlüsseln für Anwendungsfälle, die langfristige Anmeldeinformationen erfordern im IAM-Benutzerhandbuch.

Eine IAM-Gruppe ist eine Identität, die eine Sammlung von IAM-Benutzern angibt. Sie können sich nicht als Gruppe anmelden. Mithilfe von Gruppen können Sie Berechtigungen für mehrere Benutzer gleichzeitig angeben. Gruppen vereinfachen die Verwaltung von Berechtigungen, wenn es zahlreiche Benutzer gibt. Sie könnten beispielsweise eine Gruppe benennen IAMAdminsund dieser Gruppe Berechtigungen zur Verwaltung von IAM-Ressourcen erteilen.

Benutzer unterscheiden sich von Rollen. Ein Benutzer ist einer einzigen Person oder Anwendung eindeutig zugeordnet. Eine Rolle kann von allen Personen angenommen werden, die sie benötigen. Benutzer besitzen dauerhafte Anmeldeinformationen. Rollen stellen temporäre Anmeldeinformationen

bereit. Weitere Informationen finden Sie unter Anwendungsfälle für IAM-Benutzer im IAM-Benutzerhandbuch.

IAM-Rollen

Eine IAM-Rolle ist eine Identität innerhalb von Ihnen AWS-Konto , die über bestimmte Berechtigungen verfügt. Sie ist einem IAM-Benutzer vergleichbar, jedoch nicht mit einer bestimmten Person verknüpft. Um vorübergehend eine IAM-Rolle in der zu übernehmen AWS Management Console, können Sie von einer Benutzer- zu einer IAM-Rolle (Konsole) wechseln. Sie können eine Rolle übernehmen, indem Sie eine AWS CLI oder AWS API-Operation aufrufen oder eine benutzerdefinierte URL verwenden. Weitere Informationen zu Methoden für die Verwendung von Rollen finden Sie unter Methoden für die Übernahme einer Rolle im IAM-Benutzerhandbuch.

IAM-Rollen mit temporären Anmeldeinformationen sind in folgenden Situationen hilfreich:

- Verbundbenutzerzugriff – Um einer Verbundidentität Berechtigungen zuzuweisen, erstellen Sie eine Rolle und definieren Berechtigungen für die Rolle. Wird eine Verbundidentität authentifiziert, so wird die Identität der Rolle zugeordnet und erhält die von der Rolle definierten Berechtigungen. Informationen zu Rollen für den Verbund finden Sie unter Erstellen von Rollen für externe Identitätsanbieter (Verbund) im IAM-Benutzerhandbuch. Wenn Sie IAM Identity Center verwenden, konfigurieren Sie einen Berechtigungssatz. Wenn Sie steuern möchten, worauf Ihre Identitäten nach der Authentifizierung zugreifen können, korreliert IAM Identity Center den Berechtigungssatz mit einer Rolle in IAM. Informationen zu Berechtigungssätzen finden Sie unter Berechtigungssätze im AWS IAM Identity Center -Benutzerhandbuch.
- Temporäre IAM-Benutzerberechtigungen – Ein IAM-Benutzer oder eine -Rolle kann eine IAM-Rolle übernehmen, um vorübergehend andere Berechtigungen für eine bestimmte Aufgabe zu erhalten.
- Kontoübergreifender Zugriff – Sie können eine IAM-Rolle verwenden, um einem vertrauenswürdigen Prinzipal in einem anderen Konto den Zugriff auf Ressourcen in Ihrem Konto zu ermöglichen. Rollen stellen die primäre Möglichkeit dar, um kontoübergreifendem Zugriff zu gewähren. Bei einigen können Sie AWS-Services jedoch eine Richtlinie direkt an eine Ressource anhängen (anstatt eine Rolle als Proxy zu verwenden). Informationen zu den Unterschieden zwischen Rollen und ressourcenbasierten Richtlinien für den kontoübergreifenden Zugriff finden Sie unter Kontoübergreifender Ressourcenzugriff in IAM im IAM-Benutzerhandbuch.
- Serviceübergreifender Zugriff — Einige AWS-Services verwenden Funktionen in anderen AWS-Services. Wenn Sie beispielsweise in einem Service einen Anruf tätigen, ist es üblich, dass dieser Service Anwendungen in Amazon ausführt EC2 oder Objekte in Amazon S3 speichert. Ein Dienst

kann dies mit den Berechtigungen des aufrufenden Prinzipals mit einer Servicerolle oder mit einer serviceverknüpften Rolle tun.

- Forward Access Sessions (FAS) — Wenn Sie einen IAM-Benutzer oder eine IAM-Rolle verwenden, um Aktionen auszuführen AWS, gelten Sie als Principal. Bei einigen Services könnte es Aktionen geben, die dann eine andere Aktion in einem anderen Service initiieren. FAS verwendet die Berechtigungen des Prinzipals, der einen aufruft AWS-Service, in Kombination mit der Anfrage, Anfragen an AWS-Service nachgelagerte Dienste zu stellen. FAS-Anfragen werden nur gestellt, wenn ein Dienst eine Anfrage erhält, für deren Abschluss Interaktionen mit anderen AWS-Services oder Ressourcen erforderlich sind. In diesem Fall müssen Sie über Berechtigungen zum Ausführen beider Aktionen verfügen. Einzelheiten zu den Richtlinien für FAS-Anfragen finden Sie unter Zugriffssitzungen weiterleiten.
- Servicerolle – Eine Servicerolle ist eine IAM-Rolle, die ein Service übernimmt, um Aktionen in Ihrem Namen auszuführen. Ein IAM-Administrator kann eine Servicerolle innerhalb von IAM erstellen, ändern und löschen. Weitere Informationen finden Sie unter Erstellen einer Rolle zum Delegieren von Berechtigungen an einen AWS-Service im IAM-Benutzerhandbuch.
- Dienstbezogene Rolle — Eine dienstbezogene Rolle ist eine Art von Servicerolle, die mit einer verknüpft ist. AWS-Service Der Service kann die Rolle übernehmen, um eine Aktion in Ihrem Namen auszuführen. Servicebezogene Rollen erscheinen in Ihrem Dienst AWS-Konto und gehören dem Dienst. Ein IAM-Administrator kann die Berechtigungen für Service-verknüpfte Rollen anzeigen, aber nicht bearbeiten.

- Auf Amazon ausgeführte Anwendungen EC2 — Sie können eine IAM-Rolle verwenden, um temporäre Anmeldeinformationen für Anwendungen zu verwalten, die auf einer EC2 Instance ausgeführt werden und AWS API-Anfragen stellen AWS CLI . Dies ist dem Speichern von Zugriffsschlüsseln innerhalb der EC2 Instance vorzuziehen. Um einer EC2 Instanz eine AWS Rolle zuzuweisen und sie allen ihren Anwendungen zur Verfügung zu stellen, erstellen Sie ein Instanzprofil, das an die Instanz angehängt ist. Ein Instanzprofil enthält die Rolle und ermöglicht Programmen, die auf der EC2 Instanz ausgeführt werden, temporäre Anmeldeinformationen abzurufen. Weitere Informationen finden Sie im IAM-Benutzerhandbuch unter Verwenden einer IAM-Rolle, um Berechtigungen für Anwendungen zu gewähren, die auf EC2 Amazon-Instances ausgeführt werden.

Verwalten des Zugriffs mit Richtlinien

Sie kontrollieren den Zugriff, AWS indem Sie Richtlinien erstellen und diese an AWS Identitäten oder Ressourcen anhängen. Eine Richtlinie ist ein Objekt, AWS das, wenn es einer Identität oder

Ressource zugeordnet ist, deren Berechtigungen definiert. AWS wertet diese Richtlinien aus, wenn ein Prinzipal (Benutzer, Root-Benutzer oder Rollensitzung) eine Anfrage stellt. Die Berechtigungen in den Richtlinien legen fest, ob eine Anforderung zugelassen oder abgelehnt wird. Die meisten Richtlinien werden AWS als JSON-Dokumente gespeichert. Weitere Informationen zu Struktur und Inhalten von JSON-Richtliniendokumenten finden Sie unter Übersicht über JSON-Richtlinien im IAM-Benutzerhandbuch.

Administratoren können mithilfe von AWS JSON-Richtlinien angeben, wer Zugriff auf was hat. Das heißt, welcher Prinzipal Aktionen für welche Ressourcen und unter welchen Bedingungen ausführen kann.

Standardmäßig haben Benutzer, Gruppen und Rollen keine Berechtigungen. Ein IAM-Administrator muss IAM-Richtlinien erstellen, die Benutzern die Berechtigung erteilen, Aktionen für die Ressourcen auszuführen, die sie benötigen. Der Administrator kann dann die IAM-Richtlinien zu Rollen hinzufügen, und Benutzer können die Rollen annehmen.

IAM-Richtlinien definieren Berechtigungen für eine Aktion unabhängig von der Methode, die Sie zur Ausführung der Aktion verwenden. Angenommen, es gibt eine Richtlinie, die Berechtigungen für die `iam:GetRole`-Aktion erteilt. Ein Benutzer mit dieser Richtlinie kann Rolleninformationen von der AWS Management Console AWS CLI, der oder der AWS API abrufen.

Identitätsbasierte Richtlinien

Identitätsbasierte Richtlinien sind JSON-Berechtigungsrichtliniendokumente, die Sie einer Identität anfügen können, wie z. B. IAM-Benutzern, -Benutzergruppen oder -Rollen. Diese Richtlinien steuern, welche Aktionen die Benutzer und Rollen für welche Ressourcen und unter welchen Bedingungen ausführen können. Informationen zum Erstellen identitätsbasierter Richtlinien finden Sie unter Definieren benutzerdefinierter IAM-Berechtigungen mit vom Kunden verwalteten Richtlinien im IAM-Benutzerhandbuch.

Identitätsbasierte Richtlinien können weiter als Inline-Richtlinien oder verwaltete Richtlinien kategorisiert werden. Inline-Richtlinien sind direkt in einen einzelnen Benutzer, eine einzelne Gruppe oder eine einzelne Rolle eingebettet. Verwaltete Richtlinien sind eigenständige Richtlinien, die Sie mehreren Benutzern, Gruppen und Rollen in Ihrem System zuordnen können AWS-Konto. Zu den verwalteten Richtlinien gehören AWS verwaltete Richtlinien und vom Kunden verwaltete Richtlinien. Informationen dazu, wie Sie zwischen einer verwalteten Richtlinie und einer Inline-Richtlinie wählen, finden Sie unter Auswählen zwischen verwalteten und eingebundenen Richtlinien im IAM-Benutzerhandbuch.

Ressourcenbasierte Richtlinien

Ressourcenbasierte Richtlinien sind JSON-Richtliniendokumente, die Sie an eine Ressource anfügen. Beispiele für ressourcenbasierte Richtlinien sind IAM-Rollen-Vertrauensrichtlinien und Amazon-S3-Bucket-Richtlinien. In Services, die ressourcenbasierte Richtlinien unterstützen, können Service-Administratoren sie verwenden, um den Zugriff auf eine bestimmte Ressource zu steuern. Für die Ressource, an welche die Richtlinie angehängt ist, legt die Richtlinie fest, welche Aktionen ein bestimmter Prinzipal unter welchen Bedingungen für diese Ressource ausführen kann. Sie müssen in einer ressourcenbasierten Richtlinie einen Prinzipal angeben. Zu den Prinzipalen können Konten, Benutzer, Rollen, Verbundbenutzer oder gehören. AWS-Services

Ressourcenbasierte Richtlinien sind Richtlinien innerhalb dieses Diensts. Sie können AWS verwaltete Richtlinien von IAM nicht in einer ressourcenbasierten Richtlinie verwenden.

Zugriffskontrolllisten () ACLs

Zugriffskontrolllisten (ACLs) steuern, welche Principals (Kontomitglieder, Benutzer oder Rollen) über Zugriffsberechtigungen für eine Ressource verfügen. ACLs ähneln ressourcenbasierten Richtlinien, verwenden jedoch nicht das JSON-Richtliniendokumentformat.

Amazon S3 und Amazon VPC sind Beispiele für Dienste, die Unterstützung ACLs bieten. AWS WAF Weitere Informationen finden Sie unter Übersicht über ACLs die Zugriffskontrollliste (ACL) im Amazon Simple Storage Service Developer Guide.

Weitere Richtlinientypen

AWS unterstützt zusätzliche, weniger verbreitete Richtlinientypen. Diese Richtlinientypen können die maximalen Berechtigungen festlegen, die Ihnen von den häufiger verwendeten Richtlinientypen erteilt werden können.

- Berechtigungsgrenzen – Eine Berechtigungsgrenze ist ein erweitertes Feature, mit der Sie die maximalen Berechtigungen festlegen können, die eine identitätsbasierte Richtlinie einer IAM-Entität (IAM-Benutzer oder -Rolle) erteilen kann. Sie können eine Berechtigungsgrenze für eine Entität festlegen. Die daraus resultierenden Berechtigungen sind der Schnittpunkt der identitätsbasierten Richtlinien einer Entität und ihrer Berechtigungsgrenzen. Ressourcenbasierte Richtlinien, die den Benutzer oder die Rolle im Feld `Principal` angeben, werden nicht durch Berechtigungsgrenzen eingeschränkt. Eine explizite Zugriffsverweigerung in einer dieser Richtlinien setzt eine Zugriffserlaubnis außer Kraft. Weitere Informationen über Berechtigungsgrenzen finden Sie unter Berechtigungsgrenzen für IAM-Entitäten im IAM-Benutzerhandbuch.

- Dienststeuerungsrichtlinien (SCPs) — SCPs sind JSON-Richtlinien, die die maximalen Berechtigungen für eine Organisation oder Organisationseinheit (OU) in festlegen. AWS Organizations AWS Organizations ist ein Dienst zur Gruppierung und zentralen Verwaltung mehrerer Objekte AWS-Konten , die Ihrem Unternehmen gehören. Wenn Sie alle Funktionen in einer Organisation aktivieren, können Sie Richtlinien zur Servicesteuerung (SCPs) auf einige oder alle Ihre Konten anwenden. Das SCP schränkt die Berechtigungen für Entitäten in Mitgliedskonten ein, einschließlich der einzelnen Root-Benutzer des AWS-Kontos Entitäten. Weitere Informationen zu Organizations und SCPs finden Sie unter Richtlinien zur Servicesteuerung im AWS Organizations Benutzerhandbuch.
- Ressourcenkontrollrichtlinien (RCPs) — RCPs sind JSON-Richtlinien, mit denen Sie die maximal verfügbaren Berechtigungen für Ressourcen in Ihren Konten festlegen können, ohne die IAM-Richtlinien aktualisieren zu müssen, die jeder Ressource zugeordnet sind, deren Eigentümer Sie sind. Das RCP schränkt die Berechtigungen für Ressourcen in Mitgliedskonten ein und kann sich auf die effektiven Berechtigungen für Identitäten auswirken, einschließlich der Root-Benutzer des AWS-Kontos, unabhängig davon, ob sie zu Ihrer Organisation gehören. Weitere Informationen zu Organizations RCPs, einschließlich einer Liste AWS-Services dieser Support-Leistungen RCPs, finden Sie unter Resource Control Policies (RCPs) im AWS Organizations Benutzerhandbuch.
- Sitzungsrichtlinien – Sitzungsrichtlinien sind erweiterte Richtlinien, die Sie als Parameter übergeben, wenn Sie eine temporäre Sitzung für eine Rolle oder einen verbundenen Benutzer programmgesteuert erstellen. Die resultierenden Sitzungsberechtigungen sind eine Schnittmenge der auf der Identität des Benutzers oder der Rolle basierenden Richtlinien und der Sitzungsrichtlinien. Berechtigungen können auch aus einer ressourcenbasierten Richtlinie stammen. Eine explizite Zugriffsverweigerung in einer dieser Richtlinien setzt eine Zugriffserlaubnis außer Kraft. Weitere Informationen finden Sie unter Sitzungsrichtlinien im IAM-Benutzerhandbuch.

Mehrere Richtlinientypen

Wenn mehrere auf eine Anforderung mehrere Richtlinientypen angewendet werden können, sind die entsprechenden Berechtigungen komplizierter. Informationen darüber, wie AWS bestimmt wird, ob eine Anfrage zulässig ist, wenn mehrere Richtlinientypen betroffen sind, finden Sie im IAM-Benutzerhandbuch unter Bewertungslogik für Richtlinien.

So funktioniert Amazon Managed Blockchain (AMB) Access Polygon mit IAM

Bevor Sie IAM verwenden, um den Zugriff auf AMB Access Polygon zu verwalten, sollten Sie sich darüber informieren, welche IAM-Funktionen für die Verwendung mit AMB Access Polygon verfügbar sind.

IAM-Funktionen, die Sie mit Amazon Managed Blockchain (AMB) Access Polygon verwenden können

IAM-Feature	Unterstützung für AMB Access Polygon
Identitätsbasierte Richtlinien	Ja
Ressourcenbasierte Richtlinien	Nein
Richtlinienaktionen	Ja
Richtlinienressourcen	Nein
Bedingungsschlüssel für die Richtlinie	Nein
ACLs	Nein
ABAC (Tags in Richtlinien)	Nein
Temporäre Anmeldeinformationen	Nein
Hauptberechtigungen	Nein
Servicerollen	Nein
Serviceverknüpfte Rollen	Nein

Einen allgemeinen Überblick darüber, wie AMB Access Polygon und andere mit den meisten IAM-Funktionen AWS-Services funktionieren, finden Sie im IAM-Benutzerhandbuch unter AWS Dienste, die mit IAM funktionieren.

Identitätsbasierte Richtlinien für AMB Access Polygon

Unterstützt Richtlinien auf Identitätsbasis: Ja

Identitätsbasierte Richtlinien sind JSON-Berechtigungsrichtliniendokumente, die Sie einer Identität anfügen können, wie z. B. IAM-Benutzern, -Benutzergruppen oder -Rollen. Diese Richtlinien steuern, welche Aktionen die Benutzer und Rollen für welche Ressourcen und unter welchen Bedingungen ausführen können. Informationen zum Erstellen identitätsbasierter Richtlinien finden Sie unter Definieren benutzerdefinierter IAM-Berechtigungen mit vom Kunden verwalteten Richtlinien im IAM-Benutzerhandbuch.

Mit identitätsbasierten IAM-Richtlinien können Sie angeben, welche Aktionen und Ressourcen zugelassen oder abgelehnt werden. Darüber hinaus können Sie die Bedingungen festlegen, unter denen Aktionen zugelassen oder abgelehnt werden. Sie können den Prinzipal nicht in einer identitätsbasierten Richtlinie angeben, da er für den Benutzer oder die Rolle gilt, dem er zugeordnet ist. Informationen zu sämtlichen Elementen, die Sie in einer JSON-Richtlinie verwenden, finden Sie in der IAM-Referenz für JSON-Richtlinienelemente im IAM-Benutzerhandbuch.

Beispiele für identitätsbasierte Richtlinien für AMB Access Polygon

Beispiele für identitätsbasierte Richtlinien von AMB Access Polygon finden Sie unter. Beispiele für identitätsbasierte Richtlinien für Amazon Managed Blockchain (AMB) Access Polygon

Ressourcenbasierte Richtlinien in AMB Access Polygon

Unterstützt ressourcenbasierte Richtlinien: Nein

Ressourcenbasierte Richtlinien sind JSON-Richtliniendokumente, die Sie an eine Ressource anfügen. Beispiele für ressourcenbasierte Richtlinien sind IAM-Rollen-Vertrauensrichtlinien und Amazon-S3-Bucket-Richtlinien. In Services, die ressourcenbasierte Richtlinien unterstützen, können Service-Administratoren sie verwenden, um den Zugriff auf eine bestimmte Ressource zu steuern. Für die Ressource, an welche die Richtlinie angehängt ist, legt die Richtlinie fest, welche Aktionen ein bestimmter Prinzipal unter welchen Bedingungen für diese Ressource ausführen kann. Sie müssen in einer ressourcenbasierten Richtlinie einen Prinzipal angeben. Zu den Prinzipalen können Konten, Benutzer, Rollen, Verbundbenutzer oder gehören. AWS-Services

Um kontoübergreifenden Zugriff zu ermöglichen, können Sie ein gesamtes Konto oder IAM-Entitäten in einem anderen Konto als Prinzipal in einer ressourcenbasierten Richtlinie angeben. Durch das Hinzufügen eines kontoübergreifenden Auftraggebers zu einer ressourcenbasierten Richtlinie ist nur die halbe Vertrauensbeziehung eingerichtet. Wenn sich der Prinzipal und die Ressource unterscheiden AWS-Konten, muss ein IAM-Administrator des vertrauenswürdigen Kontos auch der Prinzipalentität (Benutzer oder Rolle) die Berechtigung zum Zugriff auf die Ressource erteilen. Sie

erteilen Berechtigungen, indem Sie der juristischen Stelle eine identitätsbasierte Richtlinie anfügen. Wenn jedoch eine ressourcenbasierte Richtlinie Zugriff auf einen Prinzipal in demselben Konto gewährt, ist keine zusätzliche identitätsbasierte Richtlinie erforderlich. Weitere Informationen finden Sie unter Kontoübergreifender Ressourcenzugriff in IAM im IAM-Benutzerhandbuch.

Richtlinienaktionen für AMB Access Polygon

Unterstützt Richtlinienaktionen: Ja

Administratoren können mithilfe von AWS JSON-Richtlinien angeben, wer auf was Zugriff hat. Das heißt, welcher Prinzipal Aktionen für welche Ressourcen und unter welchen Bedingungen ausführen kann.

Das Element `Action` einer JSON-Richtlinie beschreibt die Aktionen, mit denen Sie den Zugriff in einer Richtlinie zulassen oder verweigern können. Richtlinienaktionen haben normalerweise denselben Namen wie der zugehörige AWS API-Vorgang. Es gibt einige Ausnahmen, z. B. Aktionen, die nur mit Genehmigung durchgeführt werden können und für die es keinen passenden API-Vorgang gibt. Es gibt auch einige Operationen, die mehrere Aktionen in einer Richtlinie erfordern. Diese zusätzlichen Aktionen werden als abhängige Aktionen bezeichnet.

Schließen Sie Aktionen in eine Richtlinie ein, um Berechtigungen zur Durchführung der zugeordneten Operation zu erteilen.

Eine Liste der AMB Access Polygon-Aktionen finden Sie unter Von Amazon Managed Blockchain (AMB) Access Polygon definierte Aktionen in der Service Authorization Reference.

Richtlinienaktionen in AMB Access Polygon verwenden vor der Aktion das folgende Präfix:

```
managedblockchain:
```

Um mehrere Aktionen in einer einzigen Anweisung anzugeben, trennen Sie sie mit Kommata:

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

Sie können auch Platzhalter verwenden, um mehrere Aktionen anzugeben. Beispielsweise können Sie alle Aktionen festlegen, die mit dem Wort `InvokeRpcPolygon` beginnen, einschließlich der folgenden Aktion:

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

Beispiele für identitätsbasierte Richtlinien von AMB Access Polygon finden Sie unter. Beispiele für identitätsbasierte Richtlinien für Amazon Managed Blockchain (AMB) Access Polygon

Richtlinienressourcen für AMB Access Polygon

Unterstützt politische Ressourcen: Nein

Administratoren können mithilfe von AWS JSON-Richtlinien angeben, wer Zugriff auf was hat. Das heißt, welcher Prinzipal Aktionen für welche Ressourcen und unter welchen Bedingungen ausführen kann.

Das JSON-Richtlinienelement `Resource` gibt die Objekte an, auf welche die Aktion angewendet wird. Anweisungen müssen entweder ein – `Resource`oder ein `NotResource`-Element enthalten. Als bewährte Methode geben Sie eine Ressource mit dem zugehörigen Amazon-Ressourcennamen (ARN) an. Sie können dies für Aktionen tun, die einen bestimmten Ressourcentyp unterstützen, der als Berechtigungen auf Ressourcenebene bezeichnet wird.

Verwenden Sie für Aktionen, die keine Berechtigungen auf Ressourcenebene unterstützen, z. B. Auflistungsoperationen, einen Platzhalter (*), um anzugeben, dass die Anweisung für alle Ressourcen gilt.

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

Eine Liste der AMB Access Polygon-Ressourcentypen und ihrer ARNs Eigenschaften finden Sie unter Resources Defined by Amazon Managed Blockchain (AMB) Access Polygon in der Service Authorization Reference. Informationen darüber, mit welchen Aktionen Sie den ARN jeder Ressource angeben können, finden Sie unter Von Amazon Managed Blockchain (AMB) Access Polygon definierte Aktionen.

Beispiele für identitätsbasierte Richtlinien von AMB Access Polygon finden Sie unter. Beispiele für identitätsbasierte Richtlinien für Amazon Managed Blockchain (AMB) Access Polygon

Schlüssel zur Richtlinienbedingung für AMB Access Polygon

Unterstützt dienstspezifische Richtlinien-Bedingungsschlüssel: Nein

Administratoren können mithilfe von AWS JSON-Richtlinien angeben, wer Zugriff auf was hat. Das heißt, welcher Prinzipal kann Aktionen für welche Ressourcen und unter welchen Bedingungen ausführen.

Das Element `Condition` (oder `Condition` block) ermöglicht Ihnen die Angabe der Bedingungen, unter denen eine Anweisung wirksam ist. Das Element `Condition` ist optional. Sie können bedingte Ausdrücke erstellen, die Bedingungsoperatoren verwenden, z. B. ist gleich oder kleiner als, damit die Bedingung in der Richtlinie mit Werten in der Anforderung übereinstimmt.

Wenn Sie mehrere `Condition`-Elemente in einer Anweisung oder mehrere Schlüssel in einem einzelnen `Condition`-Element angeben, wertet AWS diese mittels einer logischen `AND`-Operation aus. Wenn Sie mehrere Werte für einen einzelnen Bedingungsschlüssel angeben, AWS wertet die Bedingung mithilfe einer logischen `OR` Operation aus. Alle Bedingungen müssen erfüllt werden, bevor die Berechtigungen der Anweisung gewährt werden.

Sie können auch Platzhaltervariablen verwenden, wenn Sie Bedingungen angeben. Beispielsweise können Sie einem IAM-Benutzer die Berechtigung für den Zugriff auf eine Ressource nur dann gewähren, wenn sie mit dessen IAM-Benutzernamen gekennzeichnet ist. Weitere Informationen finden Sie unter IAM-Richtlinienelemente: Variablen und Tags im IAM-Benutzerhandbuch.

AWS unterstützt globale Bedingungsschlüssel und dienstspezifische Bedingungsschlüssel. Eine Übersicht aller AWS globalen Bedingungsschlüssel finden Sie unter Kontextschlüssel für AWS globale Bedingungen im IAM-Benutzerhandbuch.

Eine Liste der AMB Access Polygon-Bedingungsschlüssel finden Sie unter Bedingungsschlüssel für Amazon Managed Blockchain (AMB) Access Polygon in der Service Authorization Reference. Informationen zu den Aktionen und Ressourcen, mit denen Sie einen Bedingungsschlüssel verwenden können, finden Sie unter Von Amazon Managed Blockchain (AMB) Access Polygon definierte Aktionen.

Beispiele für identitätsbasierte Richtlinien von AMB Access Polygon finden Sie unter. Beispiele für identitätsbasierte Richtlinien für Amazon Managed Blockchain (AMB) Access Polygon

ACLs in AMB Access Polygon

Unterstützt ACLs: Nein

Zugriffskontrolllisten (ACLs) steuern, welche Principals (Kontomitglieder, Benutzer oder Rollen) über Zugriffsberechtigungen für eine Ressource verfügen. ACLs ähneln ressourcenbasierten Richtlinien, verwenden jedoch nicht das JSON-Richtliniendokumentformat.

ABAC mit AMB Access Polygon

Unterstützt ABAC (Tags in Richtlinien): Nein

Die attributbasierte Zugriffskontrolle (ABAC) ist eine Autorisierungsstrategie, bei der Berechtigungen basierend auf Attributen definiert werden. In AWS werden diese Attribute als Tags bezeichnet. Sie können Tags an IAM-Entitäten (Benutzer oder Rollen) und an viele AWS Ressourcen anhängen. Das Markieren von Entitäten und Ressourcen ist der erste Schritt von ABAC. Anschließend entwerfen Sie ABAC-Richtlinien, um Operationen zuzulassen, wenn das Tag des Prinzipals mit dem Tag der Ressource übereinstimmt, auf die sie zugreifen möchten.

ABAC ist in Umgebungen hilfreich, die schnell wachsen, und unterstützt Sie in Situationen, in denen die Richtlinienverwaltung mühsam wird.

Um den Zugriff auf der Grundlage von Tags zu steuern, geben Sie im Bedingungselement einer Richtlinie Tag-Informationen an, indem Sie die Schlüssel `aws:ResourceTag/key-name`, `aws:RequestTag/key-name`, oder Bedingung `aws:TagKeys` verwenden.

Wenn ein Service alle drei Bedingungsschlüssel für jeden Ressourcentyp unterstützt, lautet der Wert für den Service Ja. Wenn ein Service alle drei Bedingungsschlüssel für nur einige Ressourcentypen unterstützt, lautet der Wert Teilweise.

Weitere Informationen zu ABAC finden Sie unter Definieren von Berechtigungen mit ABAC-Autorisierung im IAM-Benutzerhandbuch. Um ein Tutorial mit Schritten zur Einstellung von ABAC anzuzeigen, siehe Attributbasierte Zugriffskontrolle (ABAC) verwenden im IAM-Benutzerhandbuch.

Verwenden temporärer Anmeldeinformationen mit AMB Access Polygon

Unterstützt temporäre Anmeldeinformationen: Nein

Einige funktionieren AWS-Services nicht, wenn Sie sich mit temporären Anmeldeinformationen anmelden. Weitere Informationen, einschließlich Informationen, die mit temporären Anmeldeinformationen AWS-Services funktionieren AWS-Services , finden Sie im IAM-Benutzerhandbuch unter Diese Option funktioniert mit IAM.

Sie verwenden temporäre Anmeldeinformationen, wenn Sie sich mit einer anderen AWS Management Console Methode als einem Benutzernamen und einem Passwort anmelden. Wenn

Sie beispielsweise AWS über den Single Sign-On-Link (SSO) Ihres Unternehmens darauf zugreifen, werden bei diesem Vorgang automatisch temporäre Anmeldeinformationen erstellt. Sie erstellen auch automatisch temporäre Anmeldeinformationen, wenn Sie sich als Benutzer bei der Konsole anmelden und dann die Rollen wechseln. Weitere Informationen zum Wechseln von Rollen finden Sie unter Wechseln von einer Benutzerrolle zu einer IAM-Rolle (Konsole) im IAM-Benutzerhandbuch.

Mithilfe der AWS API AWS CLI oder können Sie temporäre Anmeldeinformationen manuell erstellen. Sie können diese temporären Anmeldeinformationen dann für den Zugriff verwenden AWS. AWS empfiehlt, temporäre Anmeldeinformationen dynamisch zu generieren, anstatt langfristige Zugriffsschlüssel zu verwenden. Weitere Informationen finden Sie unter Temporäre Sicherheitsanmeldeinformationen in IAM.

Serviceübergreifende Prinzipalberechtigungen für AMB Access Polygon

Unterstützt Forward Access Sessions (FAS): Nein

Wenn Sie einen IAM-Benutzer oder eine IAM-Rolle verwenden, um Aktionen auszuführen AWS, gelten Sie als Principal. Bei einigen Services könnte es Aktionen geben, die dann eine andere Aktion in einem anderen Service initiieren. FAS verwendet die Berechtigungen des Prinzipals, der einen aufruft AWS-Service, kombiniert mit der Anforderung, Anfragen an nachgelagerte Dienste AWS-Service zu stellen. FAS-Anfragen werden nur gestellt, wenn ein Dienst eine Anfrage erhält, für deren Abschluss Interaktionen mit anderen AWS-Services oder Ressourcen erforderlich sind. In diesem Fall müssen Sie über Berechtigungen zum Ausführen beider Aktionen verfügen. Einzelheiten zu den Richtlinien für FAS-Anfragen finden Sie unter Zugriffssitzungen weiterleiten.

Servicerollen für AMB Access Polygon

Unterstützt Servicerollen: Nein

Eine Servicerolle ist eine IAM-Rolle, die ein Service annimmt, um Aktionen in Ihrem Namen auszuführen. Ein IAM-Administrator kann eine Servicerolle innerhalb von IAM erstellen, ändern und löschen. Weitere Informationen finden Sie unter Erstellen einer Rolle zum Delegieren von Berechtigungen an einen AWS-Service im IAM-Benutzerhandbuch.

> **Warning**
>
> Durch das Ändern der Berechtigungen für eine Servicerolle kann die Funktionalität von AMB Access Polygon beeinträchtigt werden. Bearbeiten Sie Servicerollen nur, wenn AMB Access Polygon Sie dazu anleitet.

Mit Diensten verknüpfte Rollen für AMB Access Polygon

Unterstützt serviceverknüpfte Rollen: Ja

Eine serviceverknüpfte Rolle ist eine Art von Servicerolle, die mit einer verknüpft ist. AWS-Service Der Service kann die Rolle übernehmen, um eine Aktion in Ihrem Namen auszuführen. Dienstbezogene Rollen werden in Ihrem Dienst angezeigt AWS-Konto und gehören dem Dienst. Ein IAM-Administrator kann die Berechtigungen für Service-verknüpfte Rollen anzeigen, aber nicht bearbeiten.

Details zum Erstellen oder Verwalten von serviceverknüpften Rollen finden Sie unter AWS -Services, die mit IAM funktionieren. Suchen Sie in der Tabelle nach einem Service mit einem `Yes` in der Spalte Service-linked role (Serviceverknüpfte Rolle). Wählen Sie den Link Yes (Ja) aus, um die Dokumentation für die serviceverknüpfte Rolle für diesen Service anzuzeigen.

Beispiele für identitätsbasierte Richtlinien für Amazon Managed Blockchain (AMB) Access Polygon

Standardmäßig sind Benutzer und Rollen nicht berechtigt, AMB Access Polygon-Ressourcen zu erstellen oder zu ändern. Sie können auch keine Aufgaben mithilfe der API AWS Management Console, AWS Command Line Interface (AWS CLI) oder AWS ausführen. Ein IAM-Administrator muss IAM-Richtlinien erstellen, die Benutzern die Berechtigung erteilen, Aktionen für die Ressourcen auszuführen, die sie benötigen. Der Administrator kann dann die IAM-Richtlinien zu Rollen hinzufügen, und Benutzer können die Rollen annehmen.

Informationen dazu, wie Sie unter Verwendung dieser beispielhaften JSON-Richtliniendokumente eine identitätsbasierte IAM-Richtlinie erstellen, finden Sie unter Erstellen von IAM-Richtlinien (Konsole) im IAM-Benutzerhandbuch.

Einzelheiten zu den von AMB Access Polygon definierten Aktionen und Ressourcentypen, einschließlich des Formats ARNs für die einzelnen Ressourcentypen, finden Sie unter Aktionen, Ressourcen und Bedingungsschlüssel für Amazon Managed Blockchain (AMB) Access Polygon in der Service Authorization Reference.

Themen

- Bewährte Methoden für Richtlinien
- Verwenden der AMB Access Polygon-Konsole
- Gewähren der Berechtigung zur Anzeige der eigenen Berechtigungen für Benutzer

- Zugreifen auf Polygon-Netzwerke

Bewährte Methoden für Richtlinien

Identitätsbasierte Richtlinien legen fest, ob jemand AMB Access Polygon-Ressourcen in Ihrem Konto erstellen, darauf zugreifen oder sie löschen kann. Dies kann zusätzliche Kosten für Ihr verursachen AWS-Konto. Befolgen Sie beim Erstellen oder Bearbeiten identitätsbasierter Richtlinien die folgenden Anleitungen und Empfehlungen:

- Beginnen Sie mit AWS verwalteten Richtlinien und wechseln Sie zu Berechtigungen mit den geringsten Rechten — Verwenden Sie die AWS verwalteten Richtlinien, die Berechtigungen für viele gängige Anwendungsfälle gewähren, um damit zu beginnen, Ihren Benutzern und Workloads Berechtigungen zu gewähren. Sie sind in Ihrem verfügbar. AWS-Konto Wir empfehlen Ihnen, die Berechtigungen weiter zu reduzieren, indem Sie vom AWS Kunden verwaltete Richtlinien definieren, die speziell auf Ihre Anwendungsfälle zugeschnitten sind. Weitere Informationen finden Sie unter AWS -verwaltete Richtlinien oder AWS -verwaltete Richtlinien für Auftrags-Funktionen im IAM-Benutzerhandbuch.
- Anwendung von Berechtigungen mit den geringsten Rechten – Wenn Sie mit IAM-Richtlinien Berechtigungen festlegen, gewähren Sie nur die Berechtigungen, die für die Durchführung einer Aufgabe erforderlich sind. Sie tun dies, indem Sie die Aktionen definieren, die für bestimmte Ressourcen unter bestimmten Bedingungen durchgeführt werden können, auch bekannt als die geringsten Berechtigungen. Weitere Informationen zur Verwendung von IAM zum Anwenden von Berechtigungen finden Sie unter Richtlinien und Berechtigungen in IAM im IAM-Benutzerhandbuch.
- Verwenden von Bedingungen in IAM-Richtlinien zur weiteren Einschränkung des Zugriffs – Sie können Ihren Richtlinien eine Bedingung hinzufügen, um den Zugriff auf Aktionen und Ressourcen zu beschränken. Sie können beispielsweise eine Richtlinienbedingung schreiben, um festzulegen, dass alle Anforderungen mithilfe von SSL gesendet werden müssen. Sie können auch Bedingungen verwenden, um Zugriff auf Serviceaktionen zu gewähren, wenn diese für einen bestimmten Zweck verwendet werden AWS-Service, z. AWS CloudFormation B. Weitere Informationen finden Sie unter IAM-JSON-Richtlinienelemente: Bedingung im IAM-Benutzerhandbuch.
- Verwenden von IAM Access Analyzer zur Validierung Ihrer IAM-Richtlinien, um sichere und funktionale Berechtigungen zu gewährleisten – IAM Access Analyzer validiert neue und vorhandene Richtlinien, damit die Richtlinien der IAM-Richtliniensprache (JSON) und den bewährten IAM-Methoden entsprechen. IAM Access Analyzer stellt mehr als

100 Richtlinienprüfungen und umsetzbare Empfehlungen zur Verfügung, damit Sie sichere und funktionale Richtlinien erstellen können. Weitere Informationen finden Sie unter Richtlinienvalidierung mit IAM Access Analyzer im IAM-Benutzerhandbuch.

- Multi-Faktor-Authentifizierung (MFA) erforderlich — Wenn Sie ein Szenario haben, das IAM-Benutzer oder einen Root-Benutzer in Ihrem System erfordert AWS-Konto, aktivieren Sie MFA für zusätzliche Sicherheit. Um MFA beim Aufrufen von API-Vorgängen anzufordern, fügen Sie Ihren Richtlinien MFA-Bedingungen hinzu. Weitere Informationen finden Sie unter Sicherer API-Zugriff mit MFA im IAM-Benutzerhandbuch.

Weitere Informationen zu bewährten Methoden in IAM finden Sie unter Bewährte Methoden für die Sicherheit in IAM im IAM-Benutzerhandbuch.

Verwenden der AMB Access Polygon-Konsole

Um auf die Amazon Managed Blockchain (AMB) Access Polygon-Konsole zugreifen zu können, benötigen Sie ein Mindestmaß an Berechtigungen. Diese Berechtigungen müssen es Ihnen ermöglichen, die AMB Access Polygon-Ressourcen in Ihrem aufzulisten und anzuzeigen. AWS-Konto Wenn Sie eine identitätsbasierte Richtlinie erstellen, die strenger ist als die mindestens erforderlichen Berechtigungen, funktioniert die Konsole nicht wie vorgesehen für Entitäten (Benutzer oder Rollen) mit dieser Richtlinie.

Sie müssen Benutzern, die nur die API AWS CLI oder die API aufrufen, keine Mindestberechtigungen für die Konsole gewähren. AWS Stattdessen sollten Sie nur Zugriff auf die Aktionen zulassen, die der API-Operation entsprechen, die die Benutzer ausführen möchten.

Um sicherzustellen, dass Benutzer und Rollen die AMB Access Polygon-Konsole weiterhin verwenden können, fügen Sie den Entitäten auch das AMB Access Polygon `ConsoleAccess` oder die `ReadOnly` AWS verwaltete Richtlinie hinzu. Weitere Informationen finden Sie unter Hinzufügen von Berechtigungen zu einem Benutzer im IAM-Benutzerhandbuch.

Gewähren der Berechtigung zur Anzeige der eigenen Berechtigungen für Benutzer

In diesem Beispiel wird gezeigt, wie Sie eine Richtlinie erstellen, die IAM-Benutzern die Berechtigung zum Anzeigen der eingebundenen Richtlinien und verwalteten Richtlinien gewährt, die ihrer Benutzeridentität angefügt sind. Diese Richtlinie beinhaltet Berechtigungen zum Ausführen dieser Aktion auf der Konsole oder programmgesteuert mithilfe der OR-API. AWS CLI AWS

```
{
```

```
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "ViewOwnUserInfo",
            "Effect": "Allow",
            "Action": [
                "iam:GetUserPolicy",
                "iam:ListGroupsForUser",
                "iam:ListAttachedUserPolicies",
                "iam:ListUserPolicies",
                "iam:GetUser"
            ],
            "Resource": ["arn:aws:iam::*:user/${aws:username}"]
        },
        {
            "Sid": "NavigateInConsole",
            "Effect": "Allow",
            "Action": [
                "iam:GetGroupPolicy",
                "iam:GetPolicyVersion",
                "iam:GetPolicy",
                "iam:ListAttachedGroupPolicies",
                "iam:ListGroupPolicies",
                "iam:ListPolicyVersions",
                "iam:ListPolicies",
                "iam:ListUsers"
            ],
            "Resource": "*"
        }
    ]
}
```

Zugreifen auf Polygon-Netzwerke

> **Note**
>
> Um auf die öffentlichen Endpunkte für das Polygon zuzugreifen `mainnet` und JSON-RPC-Aufrufe `mainnet` zu tätigen, benötigen Sie Benutzeranmeldedaten (`AWS_ACCESS_KEY_IDundAWS_SECRET_ACCESS_KEY`), die über die entsprechenden IAM-Berechtigungen für AMB Access Polygon verfügen.

Example IAM-Richtlinie für den Zugriff auf alle Polygon-Netzwerke

In diesem Beispiel wird einem IAM-Benutzer AWS-Konto Zugriff auf alle Polygon-Netzwerke gewährt.

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

Example IAM-Richtlinie für den Zugriff auf das Polygon Mainnet-Netzwerk

In diesem Beispiel wird einem IAM-Benutzer AWS-Konto Zugriff auf das Polygon Mainnet-Netzwerk gewährt.

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

Fehlerbehebung bei Identität und Zugriff auf Amazon Managed Blockchain (AMB) Access Polygon

Verwenden Sie die folgenden Informationen, um häufig auftretende Probleme zu diagnostizieren und zu beheben, die bei der Arbeit mit AMB Access Polygon und IAM auftreten können.

Themen

Ich bin nicht berechtigt, eine Aktion in AMB Access Polygon durchzuführen

Wenn Sie eine Fehlermeldung erhalten, dass Sie nicht zur Durchführung einer Aktion berechtigt sind, müssen Ihre Richtlinien aktualisiert werden, damit Sie die Aktion durchführen können.

Der folgende Beispielfehler tritt auf, wenn der IAM-Benutzer `mateojackson` versucht, über die Konsole Details zu einer fiktiven *`my-example-widget`*-Ressource anzuzeigen, jedoch nicht über `managedblockchain::`*`GetWidget`*-Berechtigungen verfügt.

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

In diesem Fall muss die Richtlinie für den Benutzer `mateojackson` aktualisiert werden, damit er mit der `managedblockchain::`*`GetWidget`*-Aktion auf die *`my-example-widget`*-Ressource zugreifen kann.

Wenn Sie Hilfe benötigen, wenden Sie sich an Ihren AWS Administrator. Ihr Administrator hat Ihnen Ihre Anmeldeinformationen zur Verfügung gestellt.

Ich bin nicht berechtigt, iam auszuführen: PassRole

Wenn Sie die Fehlermeldung erhalten, dass Sie nicht autorisiert sind, die `iam:PassRole` Aktion auszuführen, müssen Ihre Richtlinien aktualisiert werden, damit Sie eine Rolle an AMB Access Polygon übergeben können.

Einige AWS-Services ermöglichen es Ihnen, eine bestehende Rolle an diesen Dienst zu übergeben, anstatt eine neue Servicerolle oder eine dienstverknüpfte Rolle zu erstellen. Hierzu benötigen Sie Berechtigungen für die Übergabe der Rolle an den Dienst.

Der folgende Beispielfehler tritt auf, wenn ein IAM-Benutzer mit dem Namen `marymajor` versucht, die Konsole zu verwenden, um eine Aktion in AMB Access Polygon auszuführen. Die Aktion erfordert jedoch, dass der Service über Berechtigungen verfügt, die durch eine Servicerolle gewährt werden. Mary besitzt keine Berechtigungen für die Übergabe der Rolle an den Dienst.

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

In diesem Fall müssen die Richtlinien von Mary aktualisiert werden, um die Aktion `iam:PassRole` ausführen zu können.

Wenn Sie Hilfe benötigen, wenden Sie sich an Ihren Administrator. AWS Ihr Administrator hat Ihnen Ihre Anmeldeinformationen zur Verfügung gestellt.

Ich möchte Personen außerhalb von mir den Zugriff AWS-Konto auf meine AMB Access Polygon-Ressourcen ermöglichen

Sie können eine Rolle erstellen, die Benutzer in anderen Konten oder Personen außerhalb Ihrer Organisation für den Zugriff auf Ihre Ressourcen verwenden können. Sie können festlegen, wem die Übernahme der Rolle anvertraut wird. Für Dienste, die ressourcenbasierte Richtlinien oder Zugriffskontrolllisten (ACLs) unterstützen, können Sie diese Richtlinien verwenden, um Personen Zugriff auf Ihre Ressourcen zu gewähren.

Weitere Informationen dazu finden Sie hier:

- Informationen darüber, ob AMB Access Polygon diese Funktionen unterstützt, finden Sie unter. So funktioniert Amazon Managed Blockchain (AMB) Access Polygon mit IAM
- Informationen dazu, wie Sie Zugriff auf Ihre Ressourcen gewähren können, AWS-Konten die Ihnen gehören, finden Sie im IAM-Benutzerhandbuch unter Gewähren des Zugriffs für einen IAM-Benutzer in einem anderen AWS-Konto , den Sie besitzen.
- Informationen dazu, wie Sie Dritten Zugriff auf Ihre Ressourcen gewähren können AWS-Konten, finden Sie AWS-Konten im IAM-Benutzerhandbuch unter Gewähren des Zugriffs für Dritte.
- Informationen dazu, wie Sie über einen Identitätsverbund Zugriff gewähren, finden Sie unter Gewähren von Zugriff für extern authentifizierte Benutzer (Identitätsverbund) im IAM-Benutzerhandbuch.

- Informationen zum Unterschied zwischen der Verwendung von Rollen und ressourcenbasierten Richtlinien für den kontoübergreifenden Zugriff finden Sie unter Kontoübergreifender Ressourcenzugriff in IAM im IAM-Benutzerhandbuch.

Protokollieren von Amazon Managed Blockchain (AMB) Access Polygon-Ereignissen mithilfe von AWS CloudTrail

> **Note**
>
> Amazon Managed Blockchain (AMB) Access Polygon unterstützt keine Verwaltungsereignisse.

Amazon Managed Blockchain läuft auf AWS CloudTrail, einem Service, der eine Aufzeichnung der Aktionen eines Benutzers, einer Rolle oder eines AWS Dienstes in Managed Blockchain bereitstellt. CloudTrail erfasst, wer die AMB Access Polygon-Endpunkte für Managed Blockchain als Ereignisse auf Datenebene aufgerufen hat.

Wenn Sie einen ordnungsgemäß konfigurierten Trail erstellen, der für den Empfang der gewünschten Ereignisse auf der Datenebene abonniert ist, können Sie kontinuierlich Ereignisse im Zusammenhang mit AMB Access Polygon an einen S3-Bucket senden. CloudTrail Anhand der von gesammelten Informationen können Sie feststellen CloudTrail, ob eine Anfrage an einen der AMB Access Polygon-Endpunkte gestellt wurde, von welcher IP-Adresse die Anfrage kam, wer die Anfrage gestellt hat, wann sie gestellt wurde und weitere zusätzliche Informationen.

Weitere Informationen CloudTrail dazu finden Sie im AWS CloudTrail Benutzerhandbuch.

Informationen zu AMB Access Polygon finden Sie unter CloudTrail

CloudTrail ist auf Ihrem aktiviert, AWS-Konto wenn Sie es erstellen. Sie müssen jedoch die Ereignisse auf der Datenebene konfigurieren, um zu sehen, wer die AMB Access Polygon-Endpunkte aufgerufen hat.

Für eine fortlaufende Aufzeichnung der Ereignisse in Ihrem System AWS-Konto, einschließlich der Ereignisse für AMB Access Polygon, erstellen Sie einen Trail. Ein Trail ermöglicht die Übermittlung CloudTrail von Protokolldateien an einen S3-Bucket. Wenn Sie einen Trail in der Konsole anlegen, gilt dieser für alle AWS-Regionen-Regionen. Der Trail protokolliert Ereignisse aus allen unterstützten Regionen in der AWS Partition und übermittelt die Protokolldateien an den von Ihnen angegebenen S3-Bucket. Darüber hinaus können Sie andere konfigurieren, AWS-Services um die in den CloudTrail Protokollen gesammelten Ereignisdaten weiter zu analysieren und entsprechend zu handeln. Weitere Informationen finden Sie hier:

- Wird verwendet CloudTrail , um Polygon JSON zu verfolgen- RPCs
- Übersicht zum Erstellen eines Trails
- CloudTrail unterstützte Dienste und Integrationen
- Konfiguration von Amazon SNS SNS-Benachrichtigungen für CloudTrail
- Empfangen von CloudTrail Protokolldateien aus mehreren Regionen und Empfangen von CloudTrail Protokolldateien von mehreren Konten

Durch die Analyse der CloudTrail Datenereignisse können Sie überwachen, wer die AMB Access Polygon-Endpunkte aufgerufen hat.

Jeder Ereignis- oder Protokolleintrag enthält Informationen zu dem Benutzer, der die Anforderung generiert hat. Die Identitätsinformationen unterstützen Sie bei der Ermittlung der folgenden Punkte:

- Ob die Anfrage mit Root- oder AWS Identity and Access Management (IAM-) Benutzeranmeldedaten gestellt wurde
- Ob die Anforderung mit temporären Sicherheitsanmeldeinformationen für eine Rolle oder einen verbundenen Benutzer gestellt wurde
- Ob die Anfrage von einem anderen gestellt wurde AWS-Service

Weitere Informationen finden Sie unter CloudTrail -Element userIdentity.

Grundlegendes zu den Einträgen in der AMB Access Polygon-Protokolldatei

Bei Ereignissen auf Datenebene ist ein Trail eine Konfiguration, die die Übertragung von Ereignissen als Protokolldateien an einen bestimmten S3-Bucket ermöglicht. Jede CloudTrail Protokolldatei enthält einen oder mehrere Protokolleinträge, die eine einzelne Anfrage aus einer beliebigen Quelle darstellen. Diese Einträge enthalten Details zur angeforderten Aktion, einschließlich Datum und Uhrzeit der Aktion sowie aller zugehörigen Anforderungsparameter.

> **Note**
>
> CloudTrail Datenereignisse in den Protokolldateien sind kein geordneter Stack-Trace der AMB Access Polygon-API-Aufrufe, sodass sie nicht in einer bestimmten Reihenfolge angezeigt werden.

Wird verwendet CloudTrail , um Polygon JSON zu verfolgen- RPCs

Sie können CloudTrail damit verfolgen, wer in Ihrem Konto die AMB Access Polygon-Endpunkte aufgerufen hat und welcher JSON-RPC als Datenereignisse aufgerufen wurde. Wenn Sie einen Trail erstellen, werden Datenereignisse standardmäßig nicht protokolliert. Um aufzuzeichnen, wer die AMB Access Polygon-Endpunkte als CloudTrail Datenereignisse aufgerufen hat, müssen Sie einem Trail explizit die unterstützten Ressourcen oder Ressourcentypen hinzufügen, für die Sie Aktivitäten erfassen möchten. AMB Access Polygon unterstützt das Hinzufügen von Datenereignissen mithilfe des SDK, und. AWS Management Console AWS CLIWeitere Informationen finden Sie unter Ereignisse mithilfe erweiterter Selektoren protokollieren im AWS CloudTrail Benutzerhandbuch.

Um Datenereignisse in einem Trail zu protokollieren, verwenden Sie den put-event-selectorsVorgang, nachdem Sie den Trail erstellt haben. Verwenden Sie die `--advanced-event-selectors` Option, um die `AWS::ManagedBlockchain::Network` Ressourcentypen anzugeben, um mit der Protokollierung von Datenereignissen zu beginnen und festzustellen, wer die AMB Access Polygon-Endpunkte aufgerufen hat.

Example Eintrag aller AMB Access Polygon-Endpunkt-Anfragen Ihres Kontos im Datenereignisprotokoll

Das folgende Beispiel zeigt, wie Sie mit diesem `put-event-selectors` Vorgang alle AMB Access Polygon-Endpunktanfragen Ihres Kontos für den Trail in der Region protokollieren können. `my-polygon-trail` `us-east-1`

```
aws cloudtrail put-event-selectors \

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

Nach dem Abonnieren können Sie die Nutzung in dem S3-Bucket verfolgen, der mit dem im vorherigen Beispiel angegebenen Trail verbunden ist.

Das folgende Ergebnis zeigt einen Eintrag im CloudTrail Datenereignisprotokoll der Informationen, die von gesammelt wurden CloudTrail. Sie können feststellen, dass eine Polygon-JSON-RPC-Anfrage an einen der AMB Access Polygon-Endpunkte gestellt wurde, die IP-Adresse, von der die

Anfrage kam, wer die Anfrage gestellt hat, wann sie gestellt wurde, und weitere zusätzliche Details. Einige Werte im folgenden Beispiel wurden aus Sicherheitsgründen verschleiert, erscheinen aber vollständig in den tatsächlichen Protokolleinträgen.


```
{
        "eventVersion": "1.09",
        "userIdentity": {
            "type": "AssumedRole",
            "principalId": "AROA554UO62RJ7KSB7FAX:777777777777",
            "arn": "arn:aws:sts::111122223333:assumed-role/Admin/777777777777",
            "accountId": "111122223333"
        },
        "eventTime": "2023-04-12T19:00:22Z",
        "eventSource": "managedblockchain.amazonaws.com",
        "eventName": "gettxout",
        "awsRegion": "us-east-1",
        "sourceIPAddress": "111.222.333.444",
        "userAgent": "python-requests/2.28.1",
        "errorCode": "-",
        "errorMessage": "-",
        "requestParameters": {
            "jsonrpc": "2.0",
            "method": "gettxout",
            "params": [],
            "id": 1
        },
        "responseElements": null,
        "requestID": "DRznHHEj********",
        "eventID": "baeb232d-2c6b-46cd-992c-0e40********",
        "readOnly": true,
        "resources": [{
            "type": "AWS::ManagedBlockchain::Network",
            "ARN": "arn:aws:managedblockchain:::networks/n-polygon-mainnet"
        }],
        "eventType": "AwsApiCall",
        "managementEvent": false,
        "recipientAccountId": "111122223333",
        "eventCategory": "Data"
}
```

Dokumentenverlauf für das AMB Access Polygon User Guide

In der folgenden Tabelle werden die Dokumentationsversionen für AMB Access Polygon beschrieben.

Änderung	Beschreibung	Datum
Die Kontingente für JSON-RPC wurden aktualisiert	Die Kontingente, die AMB Access Polygon für jedes unterstützte JSON-RPC unterstützt, wurden aktualisiert.	12. April 2024
Ende der Unterstützung für das Testnet-Netzwerk in Mumbai	AMB Access Polygon hat die Unterstützung des Mumbai-Testnetzes am 15. April 2024 eingestellt.	10. April 2024
Hinzufügung des Themas Tutorials	AMB Access Polygon-Tutorials aus dem Bereich Community-Artikel von AWS re:Post.	9. April 2024
Öffentliche Vorschau	Öffentliche Vorschauversion des Amazon Managed Blockchain (AMB) Access Polygon-Service.	24. November 2023