IBM Blockchain Platform for AWS

Quick Start Reference Deployment

November 2018

IBM
AWS Quick Start Reference Team

Contents
Overview ........................................................................................................................................... 2
Costs and Licenses .......................................................................................................................... 3
Architecture ..................................................................................................................................... 3
Prerequisites ..................................................................................................................................... 5
  Specialized Knowledge .................................................................................................................. 5
  Technical Requirements ................................................................................................................ 5
Deployment Options ....................................................................................................................... 7
Deployment Steps .......................................................................................................................... 7
  Step 1. Prepare Your AWS Account ............................................................................................... 7
  Step 2. Launch the Quick Start ..................................................................................................... 8
  Step 3. Test the Deployment ......................................................................................................... 14
  Step 4. Configuring the IBP for AWS distributed peer ............................................................... 14
Best Practices Using IBM Blockchain Platform for AWS .............................................................. 15
Limitations ........................................................................................................................................ 15
Security ........................................................................................................................................... 16
AWS Identity and Access Management (IAM) ............................................................................... 16
Operating System Security ........................................................................................................... 16
Security Groups ............................................................................................................................. 16
Peer Security ........................................................................................................................................... 17
Data residency ........................................................................................................................................ 17
Key management ..................................................................................................................................... 17
TLS .......................................................................................................................................................... 17
Troubleshooting .................................................................................................................................... 18
Git Repository ......................................................................................................................................... 18
Additional Resources .............................................................................................................................. 18
Document Revisions .............................................................................................................................. 19

This Quick Start deployment guide was created by IBM in collaboration with Amazon Web Services (AWS).

Quick Starts are automated reference deployments that use AWS CloudFormation templates to deploy key technologies on AWS, following AWS best practices.

Overview

This Quick Start deployment guide provides step-by-step instructions for deploying an IBM Blockchain Platform (IBP) distributed peer into the AWS Cloud. A blockchain is a growing list of records, or blocks, that are linked and secured by using cryptography. It can be used as a distributed ledger. Peers, a key component of blockchain networks, maintain a copy of the ledger, endorse transactions, and host smart contracts. By using the IBP for AWS, you can deploy peers on AWS and can store ledger data in your AWS environment.

This Quick Start is for IT infrastructure decision makers and system administrators who want to rapidly configure, deploy, and run AWS-hosted IBP distributed peers that are connected to an IBM Blockchain Platform network.

The IBP distributed peers in AWS leverage the connection profile, Hyperledger Fabric certificate authorities (CAs), and the ordering service of an existing Enterprise Plan or Starter Plan network on IBM Blockchain Platform to process transactions.

Hyperledger Fabric uses container technology to host smart contracts called chaincode, which provide the logic of the blockchain system. For block storage, Apache CouchDB is a JSON document store. Google LevelDB is a key-value storage library that provides an ordered mapping between string keys and string values.
For more information about IBM Blockchain Platform, see the IBM Blockchain Platform documentation.

Costs and Licenses
This Quick Start deploys the IBP for AWS environment using AWS CloudFormation templates. You can use the template to build a new virtual private cloud (VPC) for your IBP for AWS cluster, or deploy the IBP for AWS cluster into an existing VPC.

You are responsible for the cost of the AWS services used while running this Quick Start reference deployment. There is no additional cost for using the Quick Start.

The AWS CloudFormation template for this Quick Start includes customizable configuration parameters. Some of these settings, such as instance type, will affect the cost of deployment. For cost estimates, see the pricing pages for each AWS service you will be using. Prices are subject to change.

To use the solution that is deployed by the Quick Start, you must accept the IBM license agreement. The IBP for AWS software license agreement contains more details about licensing terms. When you launch the Quick Start, you’ll be asked to read and agree to the terms of the agreement. IBP for AWS is currently offered as a Community Edition, free of charge.

IBM Blockchain Platform does not provide support for this offering. If you encounter any issues that are related to your distributed peer, refer to the free blockchain resources and support forums.

Architecture
Deploying this Quick Start for a new or existing virtual private cloud (VPC) builds the following IBP for AWS environment in the AWS Cloud.
The Quick Start sets up the following:

- A highly available architecture that spans two Availability Zones.*
- A VPC configured with public subnets according to AWS best practices. This provides you with your own virtual network on AWS.*
- An internet gateway to allow access to the internet. This gateway is used by the bastion hosts to send and receive traffic.*
- In the public subnets, an IBP for AWS cluster, which includes two peers across two Availability Zones (one peer in each subnet).
- In each public subnet, a peer container with either an embedded LevelDB database or a secondary CouchDB container.

* The template that deploys the Quick Start into an existing VPC skips the tasks marked by asterisks and prompts you for your existing VPC configuration.
Prerequisites

Specialized Knowledge

Before you deploy this Quick Start, we recommend that you become familiar with the following AWS services. (If you are new to AWS, see *Getting Started with AWS*.)

- Amazon EC2
- Amazon EBS
- Amazon VPC
- AWS CloudFormation

These instructions also presume general familiarity with the *IBM Blockchain Platform* and *Hyperledger Fabric*.

Technical Requirements

You must have either created a new IBM Blockchain Platform Enterprise Plan or Starter Plan network that uses Fabric v1.1.0 or v1.2.x, or you must currently be a member of an existing IBM Blockchain Platform v1.1.0 or v1.2.x network. If not,

- Follow the instructions for creating a new network.
- Follow the instructions for joining an existing network.

The Starter or Enterprise Plan network should have at least one organization and one channel defined. The IBP distributed peer will join this organization and channel.

In the AWS Quick Start template, you will need to choose LevelDB or CouchDB as the peer database. Select the database type of the existing blockchain network.

Register Your Distributed Peer Identity

Before the distributed peer can join the network, you need to add a new peer identity to your network on IBM Blockchain Platform. Complete the following steps to enroll and register a distributed peer node by adding a user on the Certificate Authority panel.

1. Log in to the Network Monitor of your network on IBM Blockchain Platform and choose Certificate Authority.

2. Choose *Add User* at the top of the panel. In the panel that opens, fill out the following fields to register your distributed peer to the network:
   - **ID**: The user name of your peer, which is referred to as your enroll ID when you configure your peer. Save this value for future usage.
- **Secret**: The password of your peer, which is referred to as your enroll Secret when you configure your peer. **Save this Value** for future usage.

- **Type**: Select peer for this field.

- **Affiliation**: This is the affiliation under your organization—org1, for example—that your distributed peer will belong to. Select an existing affiliation from the drop-down list or enter a new one.

- **Maximum Enrollments**: You can use this field to limit the number of times you can enroll or generate certificates using this identity. If not specified, the value defaults to unlimited enrollments.

Be sure to save the value of the **ID** and **Secret** for later when you configure your peer.

After you fill in the fields, click **Submit** to register the distributed peer. The registered peer is then listed in the table as an identity on the network.

**Note**  Because the AWS Quick Start template creates two distributed peer VPC instances by default, repeat this step twice to generate two distributed peer identities.

**Retrieve Your Distributed Peer Configuration Information**

You need to provide the API endpoints of your IBP network to your distributed peer during configuration. These endpoints allow a distributed peer to find and connect to the network on IBM Blockchain Platform.

On the **Overview** screen of your Network Monitor, click the **Distributed Peer Configuration** button.

The panel that opens displays the values of the following fields, which you can paste into the Quick Start template when you deploy a distributed peer:

- Organization Membership Service Provider (MSP)
- Certificate Authority (CA) Name
- Certificate Authority (CA) URL

You can copy and paste the fields individually, or you can choose **Download** to save them as a JSON file.
**Note:** If you download the information to a JSON file, you can copy and paste the value of each field into the Helm Chart, with the exception of the **Certificate Authority (CA) TLS Certificate**. Before you paste the TLS certificate into the Helm Chart, you must convert it to Privacy Enhanced Mail (PEM format) by issuing the following command:

```
echo -e "<CERT>" > catls.pem
```

Replace `<CERT>` with the certificate string from the JSON file. Then later, when you configure the distributed peer, copy the contents of the `catls.pem` file into the **Certificate Authority (CA) TLS Certificate** field in the Helm Chart.

**Deployment Options**

This Quick Start provides two deployment options:

- **Deploy IBP for AWS into a new VPC** (end-to-end deployment). This option builds a new AWS environment consisting of the VPC, subnets, NAT gateways, security groups, bastion hosts, and other infrastructure components, and then deploys IBP for AWS into this new VPC.

- **Deploy IBP for AWS into an existing VPC**. This option provisions IBP for AWS in your existing AWS infrastructure.

The Quick Start provides separate templates for these options. It also lets you configure CIDR blocks, instance types, and distributed peer settings, as discussed later in this guide.

**Deployment Steps**

**Step 1. Prepare Your AWS Account**

1. If you don’t already have an AWS account, create one at [https://aws.amazon.com](https://aws.amazon.com) by following the on-screen instructions.

2. Use the region selector in the navigation bar to choose the AWS Region where you want to deploy the Remote Peer on AWS.

3. Create a key pair in your preferred region.

4. If necessary, [request a service limit increase](https://aws.amazon.com) for the Amazon EC2 m4.xlarge instance type. You might need to do this if you already have an existing deployment that uses this instance type, and you think you might exceed the default limit with this deployment.
Step 2. Launch the Quick Start

**Note** You are responsible for the cost of the AWS services used while running this Quick Start reference deployment. There is no additional cost for using this Quick Start. For full details, see the pricing pages for each AWS service you will be using in this Quick Start. Prices are subject to change.

1. Choose one of the following options to launch the AWS CloudFormation template into your AWS account. For help choosing an option, see [deployment options](#) earlier in this guide.

<table>
<thead>
<tr>
<th>Option 1</th>
<th>Option 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Deploy IBP for AWS into a new VPC on AWS</td>
<td>Deploy IBP for AWS into an existing VPC on AWS</td>
</tr>
<tr>
<td><img src="#" alt="Launch" /></td>
<td><img src="#" alt="Launch" /></td>
</tr>
</tbody>
</table>

**Important**

If you’re deploying IBP for AWS into an existing VPC, make sure that your VPC has two public subnets in different Availability Zones for the database instances. These subnets require NAT gateways or NAT instances in their route tables, to allow the instances to download packages and software without exposing them to the internet. You will also need the domain name option configured in the DHCP options as explained in the [Amazon VPC documentation](#).

Also, be sure to create a security group tied to your existing VPC and add inbound rules on ports 22 and 7051 to this security group. TCP connections on port 22 allow for Secure Shell (SSH) access to the generated instance; TCP connections on port 7051 allow for external gRPC access to the peer instance (external gRPC access is needed for operating the peer using the Fabric Tools CLI and Fabric SDKs). You will be prompted for these VPC settings when you launch the Quick Start.

Each deployment takes 10-15 minutes to complete.

2. Check the region that’s displayed in the upper-right corner of the navigation bar, and change it if necessary. This is where the network infrastructure for the distributed peer will be built. The template is launched in the US East (Ohio) Region by default.
3. On the **Select Template** page, keep the default setting for the template URL, and then choose **Next**.

4. On the **Specify Details** page, change the stack name if needed. To use the Quick Start in your environment, you must accept the IBP for AWS license by clicking **I agree** next to License Agreement. Review the parameters for the template. Provide values for the parameters that require input. For all other parameters, review the default settings and customize them as necessary. When you finish reviewing and customizing the parameters, choose **Next**.

In the following tables, parameters are listed by category and described separately for the two deployment options:

- **Parameters for deploying IBM Blockchain Platform for AWS into a new VPC**
- **Parameters for deploying IBM Blockchain Platform for AWS into an existing VPC**

**Option 1: Parameters for deploying IBM Blockchain Platform for AWS into a new VPC**

*View template*

**Software License Agreement:**

<table>
<thead>
<tr>
<th>Parameter label (name)</th>
<th>Default</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>License agreement (LicenseAgreement)</td>
<td>-</td>
<td>I have read and agree to the license terms for IBM Blockchain Platform (<a href="https://ibm.biz/BdY3AY">https://ibm.biz/BdY3AY</a>).</td>
</tr>
</tbody>
</table>

**Network Configuration:**

<table>
<thead>
<tr>
<th>Parameter label (name)</th>
<th>Default</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Availability Zones (AvailabilityZones)</td>
<td><strong>Requires input</strong></td>
<td>The two Availability Zones to use for the subnets in the VPC. The Quick Start preserves the logical order you specify.</td>
</tr>
<tr>
<td>Allowed SSH access CIDR (RemoteAccessCIDR)</td>
<td><strong>Requires input</strong></td>
<td>Allowed CIDR block for external SSH access to the IBM Blockchain peer instances. It can be set to 0.0.0.0/0 to allow access from anywhere (not recommended).</td>
</tr>
<tr>
<td>Peer endpoint access CIDR (PeerEndpointAccessCIDR)</td>
<td><strong>Requires input</strong></td>
<td>Allowed CIDR block for external gRPC access to the IBM Blockchain peer instances. Typically set to 0.0.0.0/0 to allow access from anywhere.</td>
</tr>
</tbody>
</table>
Amazon EC2 Configuration:

<table>
<thead>
<tr>
<th>Parameter label (name)</th>
<th>Default</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Instance type</td>
<td>m4.xlarge</td>
<td>Type of EC2 instance for the peer instances.</td>
</tr>
<tr>
<td>Key pair name</td>
<td>Requires input</td>
<td>Name of an existing EC2 key pair within the AWS Region. The key pair allows you to connect securely to your instance after it launches. When you created an AWS account, this is the key pair you created in your preferred region.</td>
</tr>
</tbody>
</table>

IBM Blockchain Configuration:

<table>
<thead>
<tr>
<th>Parameter label (name)</th>
<th>Default</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>IBM Blockchain version</td>
<td>1.2.1</td>
<td>IBM Blockchain version to deploy.</td>
</tr>
<tr>
<td>Type of state database</td>
<td>CouchDB</td>
<td>The type of database to use for storing blockchain state. This selection should match the State Database type used by the rest of the network.</td>
</tr>
<tr>
<td>Peer volume size</td>
<td>100</td>
<td>Size of the Amazon EBS Volume Size used to store persistent data (ledger, state database, MSP) for the peer in GiBs.</td>
</tr>
<tr>
<td>Peer 1 enroll ID</td>
<td>Requires input</td>
<td>The registered ID the first peer will use to enroll.</td>
</tr>
<tr>
<td>Peer 1 enroll secret</td>
<td>Requires input</td>
<td>The secret the first peer will use to enroll.</td>
</tr>
<tr>
<td>Peer 2 enroll ID</td>
<td>Requires input</td>
<td>The registered ID the second peer will use to enroll.</td>
</tr>
<tr>
<td>Peer 2 enroll secret</td>
<td>Requires input</td>
<td>The secret the second peer will use to enroll.</td>
</tr>
<tr>
<td>Certificate Authority</td>
<td>Requires input</td>
<td>The URL for the Fabric Certificate Authority to use for enrolling the peer.</td>
</tr>
<tr>
<td>Certificate Authority</td>
<td>Requires input</td>
<td>The name of the Fabric Certificate Authority to use for enrolling the peer.</td>
</tr>
<tr>
<td>Certificate Authority</td>
<td>Requires input</td>
<td>PEM-encoded certificate chain that issued the TLS certificate for the Fabric Certificate Authority.</td>
</tr>
<tr>
<td>Organization MSP</td>
<td>Requires input</td>
<td>The MSP identifier for the organization to which this peer belongs.</td>
</tr>
</tbody>
</table>
AWS Quick Start Configuration:

<table>
<thead>
<tr>
<th>Parameter label (name)</th>
<th>Default</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Quick Start S3 bucket name (QSS3BucketName)</td>
<td>aws-quickstart</td>
<td>The S3 bucket you have created for your copy of Quick Start assets, if you decide to customize or extend the Quick Start for your own use. The bucket name can include numbers, lowercase letters, uppercase letters, and hyphens, but should not start or end with a hyphen.</td>
</tr>
<tr>
<td>Quick Start S3 key prefix (QSS3KeyPrefix)</td>
<td>quickstart-ibm-fabric/</td>
<td>The S3 key name prefix used to simulate a folder for your copy of Quick Start assets, if you decide to customize or extend the Quick Start for your own use. This prefix can include numbers, lowercase letters, uppercase letters, hyphens, and forward slashes.</td>
</tr>
</tbody>
</table>

- **Option 2: Parameters for deploying an IBM Blockchain Platform for AWS into an existing VPC**

  View template

Software License Agreement:

<table>
<thead>
<tr>
<th>Parameter label (name)</th>
<th>Default</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>License agreement (LicenseAgreement)</td>
<td>-</td>
<td>I have read and agree to the license terms for IBM Blockchain Platform (<a href="http://ibm.biz/BdY3AY">http://ibm.biz/BdY3AY</a>).</td>
</tr>
</tbody>
</table>

Network Configuration:

<table>
<thead>
<tr>
<th>Parameter label (name)</th>
<th>Default</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>VPC ID (VPCID)</td>
<td>Requires input</td>
<td>The ID of your existing VPC for deployment.</td>
</tr>
<tr>
<td>Availability Zone 1 (AvailabilityZone1)</td>
<td>Requires input</td>
<td>The Availability Zone in which to deploy the first peer node.</td>
</tr>
<tr>
<td>Subnet ID 1 (SubnetID1)</td>
<td>Requires input</td>
<td>The ID of the subnet to use for the first peer node.</td>
</tr>
<tr>
<td>Availability Zone 2 (AvailabilityZone2)</td>
<td>Requires input</td>
<td>The Availability Zone in which to deploy the second peer node.</td>
</tr>
<tr>
<td>Subnet ID 2 (SubnetID2)</td>
<td>Requires input</td>
<td>The ID of the subnet to use for the second peer node.</td>
</tr>
</tbody>
</table>
### Amazon EC2 Configuration:

<table>
<thead>
<tr>
<th>Parameter label (name)</th>
<th>Default</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Instance type (InstanceType)</td>
<td>m4.xlarge</td>
<td>Type of EC2 instance for the Workload instances.</td>
</tr>
<tr>
<td>Key pair name (KeyPairName)</td>
<td>Requires input</td>
<td>Name of an existing EC2 key pair within the AWS Region. The key pair allows you to connect securely to your instance after it launches. When you created an AWS account, this is the key pair you created in your preferred region.</td>
</tr>
<tr>
<td>Security group (SecurityGroup)</td>
<td>Requires input</td>
<td>The ID of an existing EC2 security group within the AWS region; this security group should allow ingress on ports 22, 7051, and optionally 7053.</td>
</tr>
</tbody>
</table>

### IBM Blockchain Configuration:

<table>
<thead>
<tr>
<th>Parameter label (name)</th>
<th>Default</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>IBM Blockchain version (IBMBlockchainVersion)</td>
<td>1.2.1</td>
<td>IBM Blockchain version to deploy.</td>
</tr>
<tr>
<td>Type of state database (KeyPairName)</td>
<td>CouchDB</td>
<td>The type of database to use for storing blockchain state. This selection should match the State Database type used by the rest of the network.</td>
</tr>
<tr>
<td>Peer volume size (PeerVolumeSize)</td>
<td>100</td>
<td>Size of the Amazon EBS Volume Size used to store persistent data (ledger, state database, MSP) for the peer in GiBs.</td>
</tr>
<tr>
<td>Peer 1 enroll ID (Peer1EnrollID)</td>
<td>Requires input</td>
<td>The registered ID the first peer will use to enroll.</td>
</tr>
<tr>
<td>Peer 1 enroll secret (Peer1EnrollSecret)</td>
<td>Requires input</td>
<td>The secret the first peer will use to enroll.</td>
</tr>
<tr>
<td>Peer 2 enroll ID (Peer2EnrollID)</td>
<td>Requires input</td>
<td>The registered ID the second peer will use to enroll.</td>
</tr>
<tr>
<td>Peer 2 enroll secret (Peer2EnrollSecret)</td>
<td>Requires input</td>
<td>The secret the second peer will use to enroll.</td>
</tr>
<tr>
<td>Certificate Authority (CA) URL (CAUrl)</td>
<td>Requires input</td>
<td>The URL for the Fabric Certificate Authority to use for enrolling the peer.</td>
</tr>
<tr>
<td>Certificate Authority (CA) Name (CAName)</td>
<td>Requires input</td>
<td>The name of the Fabric Certificate Authority to use for enrolling the peer.</td>
</tr>
<tr>
<td>Certificate Authority (CA) TLS Certificate Chain (CATLSCertificateChain)</td>
<td>Requires input</td>
<td>PEM-encoded certificate chain that issued the TLS certificate for the Fabric Certificate Authority.</td>
</tr>
<tr>
<td>Parameter label (name)</td>
<td>Default</td>
<td>Description</td>
</tr>
<tr>
<td>--------------------------------------------</td>
<td>---------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Organization MSP (MspID)</td>
<td>Requires input</td>
<td>The MSP identifier for the organization to which this peer belongs.</td>
</tr>
</tbody>
</table>

**AWS Quick Start Configuration:**

<table>
<thead>
<tr>
<th>Parameter label (name)</th>
<th>Default</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Quick Start S3 bucket name (QSS3BucketName)</td>
<td>aws-quickstart</td>
<td>The S3 bucket you have created for your copy of Quick Start assets, if you decide to customize or extend the Quick Start for your own use. The bucket name can include numbers, lowercase letters, uppercase letters, and hyphens, but should not start or end with a hyphen.</td>
</tr>
<tr>
<td>Quick Start S3 key prefix (QSS3KeyPrefix)</td>
<td>quickstart-ibm-fabric/</td>
<td>The S3 key name prefix used to simulate a folder for your copy of Quick Start assets, if you decide to customize or extend the Quick Start for your own use. This prefix can include numbers, lowercase letters, uppercase letters, hyphens, and forward slashes.</td>
</tr>
</tbody>
</table>

5. On the **Options** page, you can specify tags (key-value pairs) for resources in your stack and set advanced options. When you’re done, choose **Next**.

6. On the **Review** page, review and confirm the template settings. Under **Capabilities**, select the check box to acknowledge that the template will create IAM resources.

7. Choose **Create** to deploy the stack.

8. Monitor the status of the stack. When the status is **CREATE_COMPLETE**, the IBP for AWS cluster is ready.

9. Use the URLs displayed in the Outputs tab for the stack to view the resources that were created for each peer.
--- Outputs

<table>
<thead>
<tr>
<th>Key</th>
<th>Value</th>
<th>Description</th>
<th>Export Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Peer2PrivateDnsName</td>
<td>ip-10-0-144-48.us-west-1.compute.internal</td>
<td>The private hostname for the second peer instance</td>
<td></td>
</tr>
<tr>
<td>Peer1TLSCertificate</td>
<td><a href="mailto:ec2-user@ec2-13-57-252-40.us-west-1.compute.amazonaws.com">ec2-user@ec2-13-57-252-40.us-west-1.compute.amazonaws.com</a>/data/ibmblockchain/peer1/ls/ca.crt</td>
<td>The SCP URI for copying the root certificate which was generated during the initial setup</td>
<td></td>
</tr>
<tr>
<td>Peer2ExternalEndpoint</td>
<td>ec2-13-57-185-73.us-west-1compute.amazonaws.com:7051</td>
<td>The public endpoint for the second peer.</td>
<td></td>
</tr>
<tr>
<td>Peer1ExternalEndpoint</td>
<td>ec2-13-57-252-40.us-west-1.compute.amazonaws.com:7051</td>
<td>The public endpoint for the first peer.</td>
<td></td>
</tr>
<tr>
<td>Peer1PrivateDnsName</td>
<td>ip-10-0-132-240.us-west-1.compute.internal</td>
<td>The private hostname for the first peer instance.</td>
<td></td>
</tr>
<tr>
<td>Peer2InternalEndpoint</td>
<td>ip-10-0-144-48.us-west-1.compute.internal:7051</td>
<td>The private endpoint for the second peer.</td>
<td></td>
</tr>
<tr>
<td>Peer1PublicDnsName</td>
<td>ec2-13-57-185-73.us-west-1.compute.amazonaws.com</td>
<td>The public hostname for the first peer instance.</td>
<td></td>
</tr>
<tr>
<td>Peer2PublicDnsName</td>
<td>ec2-13-57-185-73.us-west-1.compute.amazonaws.com</td>
<td>The public hostname for the second peer instance.</td>
<td></td>
</tr>
<tr>
<td>Peer1InternalEndpoint</td>
<td>ip-10-0-132-240.us-west-1.compute.internal:7051</td>
<td>The private endpoint for the first peer.</td>
<td></td>
</tr>
<tr>
<td>Peer2TLSCertificate</td>
<td><a href="mailto:ec2-user@ec2-13-57-185-73.us-west-1compute.amazonaws.com">ec2-user@ec2-13-57-185-73.us-west-1compute.amazonaws.com</a>/data/ibmblockchain/peer2/ls/ca.crt</td>
<td>The SCP URI for copying the root certificate which was generated during the initial setup</td>
<td></td>
</tr>
</tbody>
</table>

**Figure 2: Resources listed on the Outputs tab**

### Step 3. Test the Deployment

After successful deployment, you should have a base stack with four nested stacks if you chose CouchDB, or two nested stacks if you chose LevelDB. All stacks should have a status of **running**. Upon successful stack creation, both distributed peer nodes will be running in your AWS account. To verify that they are running:

1. Connect by using SSH into your new or existing VPC by following the instructions from AWS.

2. From the command line, run `docker ps` to view the running containers. You should see two containers running if you selected CouchDB, and one container if you selected LevelDB. Assuming CouchDB was selected, the output will be similar to:

   ```
   $ docker ps
   CONTAINER ID   IMAGE                                      PORTS               NAMES
   1e143c5039c0   ibmblockchain/fabric-peer:1.1.0
                  0.0.0.0:7051->7051/tcp
   489677f79cfd   ibmblockchain/fabric-couchdb:0.4.6
                  4369/tcp, 5984/tcp, 9100/tcp
   peer
   couchdb
   ```

### Step 4. Configuring the IBP for AWS distributed peer

After the IBP for AWS distributed peer has successfully joined an Enterprise Plan or Starter Plan network, the next steps are to join the distributed peer to the channel and install chaincode on the distributed peer. Use the [Fabric SDK](https://github.com/hyperledger/fabric) or the [Fabric tools CLI commands](https://github.com/hyperledger/fabric/blob/master/commands/README.md) to perform these tasks.
Best Practices Using IBM Blockchain Platform for AWS

You are responsible for the management of security, health monitoring, logging, and resource usage on the distributed peer nodes.

The default instance type for the distributed peer is m4.xlarge. You should optimize the instance type you choose based on your CPU, memory, and storage requirements. The distributed peer requires at least:

- 2x CPU
- 2 GiB RAM
- 4 GiB space for chaincode
- 10 GiB space for the ledger with ability to grow as the ledger expands.

These minimum resource levels are sufficient for testing and experimentation. For a production environment it is important to allocate a sufficiently large amount of storage, for example 100 GiB, but your actual amount will depend on:

- The size of your network
- The size of your transactions, which is affected by the number of signatures required
- The number of transactions

If you exhaust the storage on your peer, you must deploy a new peer with a larger file system and let it sync via your other peer on the same channel or channels.

Limitations

- The software deployed by this Quick Start is restricted for use with IBM Blockchain Platform Enterprise Plan or Starter Plan networks. The blockchain networks must be at Hyperledger Fabric v1.1.0 or v1.2.1 levels. You can find your Hyperledger Fabric version by opening the Network Preferences window in your Network Monitor.

- The database type of the distributed peer must match the database type used on the Blockchain network. When selecting the database type (LevelDB or CouchDB) for your Distributed Peer be sure to choose the same type used by the other peers on the network.

- The CouchDB Fauxton interface is not available on the IBP Remote Peer.
• IBP for AWS distributed peers that are added to an Enterprise Plan or Starter Plan network using this AWS Quick Start template are not visible in the Network Monitor UI.

• Distributed peers cannot be addressed using the Swagger UI in the Network Monitor UI.

• Distributed Peers cannot participate in Peer Gossip messaging.

Security
The AWS Cloud provides a scalable, highly reliable platform that helps customers deploy applications and data quickly and securely.

When you build systems on the AWS infrastructure, security responsibilities are shared between you and AWS. This shared model can reduce your operational burden as AWS operates, manages, and controls the components from the host operating system and virtualization layer down to the physical security of the facilities in which the services operate. In turn, you assume responsibility and management of the guest operating system (including updates and security patches), other associated applications, as well as the configuration of the AWS-provided security group firewall. For more information about security on AWS, visit the AWS Security Center.

AWS Identity and Access Management (IAM)
This solution uses an IAM role with least privileged access. It is not necessary or recommended to store SSH keys, secret keys, or access keys on the provisioned instances.

Operating System Security
The root user on cluster nodes can be accessed only by using the SSH key specified during the deployment process. AWS doesn't store these SSH keys, so if you lose your SSH key you can lose access to these instances. Operating system patches are your responsibility and should be performed on a periodic basis.

Security Groups
A security group acts as a firewall that controls the traffic for one or more instances. When you launch an instance, you associate one or more security groups with the instance. You add rules to each security group that allow traffic to or from its associated instances. You can modify the rules for a security group at any time. The new rules are automatically applied to all instances that are associated with the security group. The security groups
created and assigned to the individual instances as part of this solution are restricted as much as possible while allowing access to the various functions needed by IBP for AWS. We recommend reviewing security groups to further restrict access as needed after the cluster is up and running.

Peer Security

The distributed peers are deployed outside of IBM Blockchain Platform; therefore, you are responsible for managing the security of the distributed peers. This includes important areas of security provided by Enterprise Plan or Starter Plan networks, such as key management and transport layer security (TLS). Review the following topics when you consider security for your distributed peers.

Data residency

Data residency mandates that the processing and storage of all blockchain ledger data remains within the border of a single country. For more details on this requirement, refer to the IBM Blockchain Platform documentation.

Key management

Key management is a critical aspect of distributed peer security. If a private key is compromised or lost, hostile actors might be able to access your remote peer's data and functionality. For a production environment, IBM Blockchain Platform Enterprise Plan uses Hardware Security Module (HSM) to store the private keys of the components (for example, a peer) on the network, but when you deploy a distributed peer (outside the network) on AWS, HSM is not available. Therefore, you are responsible for managing your private keys. Although IBM Blockchain Platform generates your private keys, those keys are not stored on the Platform. It is essential to ensure that you store your keys in a secure location so that they are not compromised.

You can use Key Escrow to recover lost private keys. This needs to be set up prior to the loss of any keys. If a private key cannot be recovered, you need to get new private keys by getting a new signCert from the Certificate Authority. You should also remove and replace your admin cert from any channels that you joined.

TLS

Transport Layer Security (TLS) is embedded in the trust model of Hyperledger Fabric. All components on IBM Blockchain Platform use TLS to communicate authenticate and communicate with each other. Therefore, the network components on IBM Blockchain Platform need to be able to complete a TLS handshake with your distributed peers. This
means that you need to enable a passthrough—for example, by using whitelisting—in your web firewall from client apps to your distributed peer.

**Troubleshooting**

**Q.** I encountered a CREATE_FAILED error when I launched the Quick Start.

**A.** If AWS CloudFormation fails to create the stack, we recommend that you relaunch the template with **Rollback on failure** set to **No**. (This setting is under **Advanced** in the AWS CloudFormation console, **Options** page.) With this setting, the stack’s state will be retained and the instance will be left running, so you can troubleshoot the issue. (Look at the log files in `%ProgramFiles%\Amazon\EC2ConfigService\cfn\log`.)

| Important | When you set **Rollback on failure** to **No**, you will continue to incur AWS charges for this stack. Please make sure to delete the stack when you finish troubleshooting. |

For additional information, see [Troubleshooting AWS CloudFormation](https://aws.amazon.com/documentation/cloudformation/) on the AWS website.

**Q.** I encountered a size limitation error when I deployed the AWS CloudFormation templates.

**A.** We recommend that you launch the Quick Start templates from the location we’ve provided or from another S3 bucket. If you deploy the templates from a local copy on your computer or from a non-S3 location, you might encounter template size limitations when you create the stack. For more information about AWS CloudFormation limits, see the [AWS documentation](https://aws.amazon.com/documentation/cloudformation/).

**Git Repository**

You can visit our [GitHub repository](https://github.com/aws-samples/aws-ibm-blockchain-platform) to download the templates and scripts for this Quick Start, to post your comments, and to share your customizations with others.

**Additional Resources**

**AWS services**

- Amazon EC2
  [https://aws.amazon.com/documentation/ec2/](https://aws.amazon.com/documentation/ec2/)
- Amazon EBS
- Amazon VPC  
  [https://aws.amazon.com/documentation/vpc/](https://aws.amazon.com/documentation/vpc/)

- AWS CloudFormation  
  [https://aws.amazon.com/documentation/cloudformation/](https://aws.amazon.com/documentation/cloudformation/)

**IBM Blockchain Platform for AWS documentation**

- IBM Blockchain Platform  
  [https://console.bluemix.net/docs/services/blockchain/index.html#ibm-blockchain-platform](https://console.bluemix.net/docs/services/blockchain/index.html#ibm-blockchain-platform)

- IBM Blockchain Platform Distributed Peer  
  [https://console.bluemix.net/docs/services/blockchain/howto/remote_peer.html#remote-peer-overview](https://console.bluemix.net/docs/services/blockchain/howto/remote_peer.html#remote-peer-overview)

- Hyperledger Fabric  

**Quick Start reference deployments**

- AWS Quick Start home page  
  [https://aws.amazon.com/quickstart/](https://aws.amazon.com/quickstart/)

**Document Revisions**

<table>
<thead>
<tr>
<th>Date</th>
<th>Change</th>
<th>In sections</th>
</tr>
</thead>
<tbody>
<tr>
<td>November 2018</td>
<td>Initial publication</td>
<td>—</td>
</tr>
</tbody>
</table>
© 2018, Amazon Web Services, Inc. or its affiliates, and IBM. All rights reserved.

Notices

This document is provided for informational purposes only. It represents AWS’s current product offerings and practices as of the date of issue of this document, which are subject to change without notice. Customers are responsible for making their own independent assessment of the information in this document and any use of AWS’s products or services, each of which is provided “as is” without warranty of any kind, whether express or implied. This document does not create any warranties, representations, contractual commitments, conditions or assurances from AWS, its affiliates, suppliers or licensors. The responsibilities and liabilities of AWS to its customers are controlled by AWS agreements, and this document is not part of, nor does it modify, any agreement between AWS and its customers.

The software included with this paper is licensed under the Apache License, Version 2.0 (the "License"). You may not use this file except in compliance with the License. A copy of the License is located at http://aws.amazon.com/apache2.0/ or in the "license" file accompanying this file. This code is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.