

Cisco Blockchain Platform on the AWS Cloud

Quick Start Reference Deployment

November 2018

*Cisco Systems
AWS Quick Start Team*

Contents

Overview.....	2
Cisco Blockchain Platform on AWS	2
Costs and Licenses.....	3
Architecture.....	3
Prerequisites	5
Technical Requirements.....	5
Specialized Knowledge	5
Deployment Options	5
Deployment Steps	6
Step 1. Prepare Your AWS Account.....	6
Step 2. Subscribe to the Cisco Blockchain Platform AMI.....	6
Step 3. Launch the Quick Start	7
Step 4. Test the Deployment	12
Security.....	12
Public Keys	12
Access to Peer Nodes.....	13
Troubleshooting.....	13

GitHub Repository	14
Additional Resources	14
Document Revisions	15

This Quick Start was created by Cisco Systems 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 reference deployment guide provides step-by-step instructions for deploying Cisco Blockchain Platform on the AWS Cloud.

Cisco Blockchain Platform on AWS

Blockchain is a shared digital ledger for recording business transactions between enterprise partners. When a transaction is executed and validated, it is appended to the end of the blockchain, making the blockchain an immutable history of all valid transactions.

In the context of software development for blockchain, smart contracts are software programs that enable complex transactions by defining data models and the methods that act on the data that you store on the blockchain. You can encapsulate shared business logic by encoding it into smart contracts.

Cisco Blockchain Platform for AWS provides an end-to-end development environment on AWS and the tools needed to build enterprise-class blockchain solutions. This Quick Start provides a fast and easy way to deploy Cisco's Blockchain network.

This deployment brings up a pre-configured enterprise blockchain network from scratch using three nodes. The nodes will automatically peer with each other to create the genesis block and immediately become ready to accept incoming requests.

The nodes use Amazon Elasticsearch Service (Amazon ES) to store transaction metadata and smart contract and smart-contract logs. This data is used by the Cisco Blockchain Platform Developer Center web user interface for display purposes.

This Quick Start is intended for developers who are focusing on the enterprise blockchain space. It was created as a streamlined approach for developers to bring up their AWS environment for testing, evaluation, and further development.

This Quick Start should not be used for an enterprise consortium deployment.

Costs and Licenses

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.

The AWS CloudFormation template for this Quick Start includes configuration parameters that you can customize. 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.

There's no cost for Cisco Blockchain Platform, although you need to be approved to receive access to the Cisco Blockchain Amazon Machine Image (AMI). If you don't currently have access approval, please send your request to cisco-aws-blockchain-eft@cisco.com.

Tip After you deploy the Quick Start, we recommend that you enable the [AWS Cost and Usage Report](#) to track costs associated with the Quick Start. This report delivers billing metrics to an Amazon Simple Storage Service (Amazon S3) bucket in your account. It provides cost estimates based on usage throughout each month, and finalizes the data at the end of the month. For more information about the report, see the [AWS documentation](#).

Architecture

Deploying this Quick Start for a new virtual private cloud (VPC) with the **default parameters** builds the following Cisco Blockchain Platform environment in the AWS Cloud.

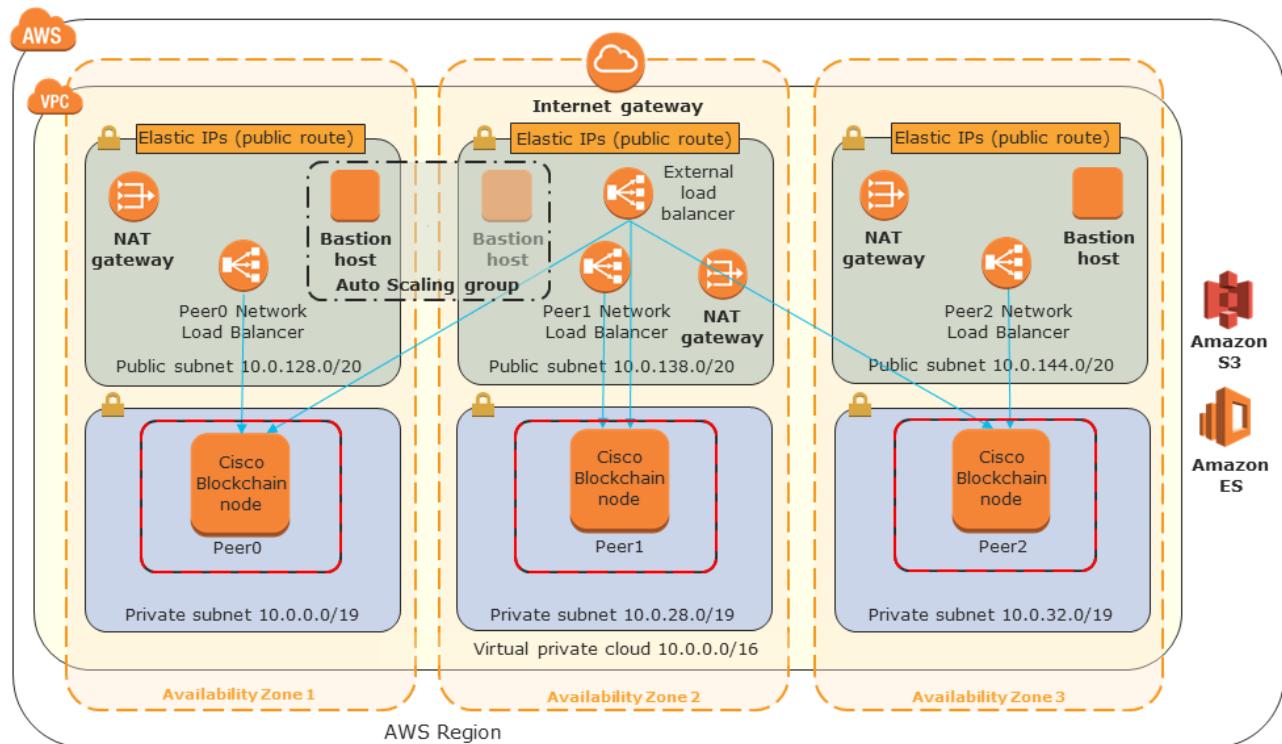


Figure 1: Quick Start architecture for Cisco Blockchain Platform on AWS

The Quick Start sets up the following:

- A VPC configured with public and private subnets according to AWS best practices, that spans three Availability Zones for high availability and quorum.*
- An internet gateway to allow access to the internet.*
- In the public subnets, managed NAT gateways to allow outbound internet access for resources in the private subnets.*
- In the public subnets, a Linux bastion host in an Auto Scaling group to allow inbound Secure Shell (SSH) access to Amazon Elastic Compute Cloud (Amazon EC2) instances in public and private subnets.*
- One Cisco Blockchain Platform node in the private subnet of each of the three Availability Zones.
- Security groups for each instance, which restrict access to only the necessary protocols and ports.
- Network Load Balancers in the public subnets to reach each Cisco Blockchain Platform node in the private subnets.
- Private Amazon S3 bucket to share certificates across the nodes.

* 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

Technical Requirements

- Cisco Blockchain Platform AMI, with access approval to Cisco Blockchain. Request access by contacting cisco-aws-blockchain-eft@cisco.com.
- AWS account with permissions to create a VPC with Amazon EC2.

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 Elastic Block Store \(Amazon EBS\)](#)
- [Amazon VPC](#)
- [Amazon S3](#)
- [Elastic Load Balancing \(ELB\)](#)
- [Amazon ES](#)
- [AWS CloudFormation](#)

We also recommend that you read the [Cisco Blockchain whitepaper](#).

Deployment Options

This Quick Start provides two deployment options:

- **Deploy Cisco Blockchain Platform 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 Cisco Blockchain Platform components into this new VPC.
- **Deploy Cisco Blockchain Platform into an existing VPC**. This option provisions Cisco Blockchain Platform components in your existing AWS infrastructure.

The Quick Start provides separate templates for these options. It also lets you configure CIDR blocks, instance types, and Cisco Blockchain Platform 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> 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 Cisco Blockchain Platform on AWS. This Quick Start requires a region with three Availability Zones. For more information, see the [AWS Global Infrastructure](#) webpage.
3. Create a [key pair](#) in your preferred region.
4. If necessary, [request a service limit increase](#) in the region of your choice for Amazon EC2 resources like EC2 instances, Elastic Load Balancers, and Elastic IP addresses. For this Quick Start deployment, we recommend having five Elastic IP addresses. You might need to request service limit increases if you already have an existing deployment that uses many of these resources, and you think you might exceed the [default limit](#) with this deployment.

Step 2. Subscribe to the Cisco Blockchain Platform AMI

This Quick Start uses AWS Marketplace software from Cisco and requires that you accept the terms within the AWS account where the Quick Start will be deployed.

1. [Request and get access](#) to the Cisco Blockchain Platform AMI for your AWS account.
2. Log in to the AWS Marketplace at <https://aws.amazon.com/marketplace>.
3. Open the page for the [Cisco Blockchain Platform AMI](#).
4. Choose **Continue**.
5. Choose **Manual Launch**, and then choose **Accept Software Terms**. For detailed subscription instructions, see the [AWS Marketplace documentation](#).

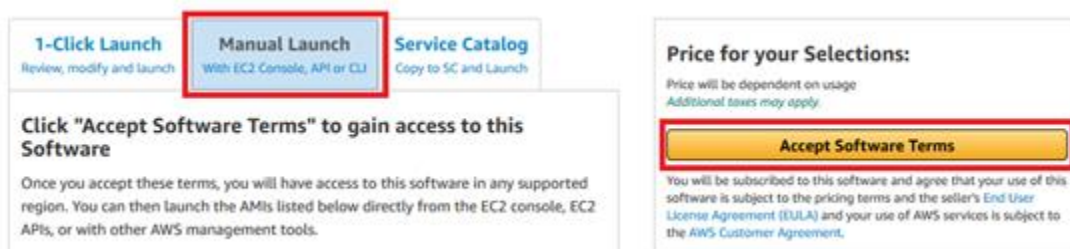


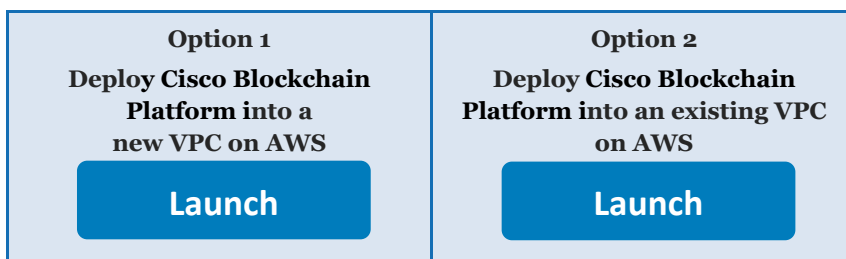
Figure 2: Accepting software terms in AWS Marketplace

6. When the subscription process is complete, exit out of AWS Marketplace without further action. **Do not** provision the software from AWS Marketplace—the Quick Start will deploy the AMI for you.

Step 3. 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.



Important If you're deploying Cisco Blockchain Platform into an existing VPC, make sure that your VPC has three private 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](#). You will be prompted for your VPC settings when you launch the Quick Start.

Each deployment takes about one hour 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 Cisco Blockchain Platform 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**.

- On the **Specify Details** page, change the stack name if needed. 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 Cisco Blockchain Platform into a new VPC](#)
- [Parameters for deploying Cisco Blockchain Platform into an existing VPC](#)
- **Option 1: Parameters for deploying Cisco Blockchain Platform into a new VPC**

[View template](#)

Network configuration (new VPC):

Parameter label (name)	Default	Description
Availability Zones (AvailabilityZones)	<i>Requires input</i>	The list of Availability Zones to use for the subnets in the VPC. The Quick Start uses three Availability Zones from your list and preserves the logical order you specify.
Number of used Availability Zones (NumberOfAZs)	3	Number of Availability Zones to use in the VPC. This must match your selections in the list of Availability Zones parameter.
Private subnet 1 CIDR (PrivateSubnet1CIDR)	10.0.0.0/20	The CIDR block for the private subnet located in Availability Zone 1.
Private subnet 2 CIDR (PrivateSubnet2CIDR)	10.0.16.0/20	The CIDR block for the private subnet located in Availability Zone 2.
Private subnet 3 CIDR (PrivateSubnet3CIDR)	10.0.32.0/20	The CIDR block for the private subnet located in Availability Zone 3.
Public subnet 1 CIDR (PublicSubnet1CIDR)	10.0.128.0/20	The CIDR block for the public (DMZ) subnet located in Availability Zone 1.
Public subnet 2 CIDR (PublicSubnet2CIDR)	10.0.144.0/20	The CIDR block for the public (DMZ) subnet located in Availability Zone 2.
Public subnet 3 CIDR (PublicSubnet3CIDR)	10.0.160.0/20	The CIDR block for the public (DMZ) subnet located in Availability Zone 3.
Allowed SSH access CIDR (RemoteAccessCIDR)	<i>Requires input</i>	The allowed CIDR block for external SSH access.
VPC CIDR (VPCCIDR)	10.0.0.0/16	The CIDR block for the VPC.

Linux bastion configuration:

Parameter label (name)	Default	Description
Bastion AMI operating system (BastionAMIOS)	Amazon-Linux-HVM	The Linux distribution for the AMI to be used for the bastion instances.
Bastion instance type (BastionInstanceType)	t2.micro	Amazon EC2 instance type for the bastion instances.
Number of bastion hosts (NumBastionHosts)	1	The number of bastion hosts to create. You can create up to four bastion hosts.
Bastion tenancy (BastionTenancy)	default	The tenancy attribute for the instances launched into the VPC. By default, all instances in the VPC run as shared-tenancy instances. Set this parameter to dedicated to run them as single-tenancy instances instead. For more information, see Dedicated Instances in the <i>Amazon EC2 User Guide</i> .
Enable banner (EnableBanner)	false	To display a banner when connecting via SSH to the bastion, set this parameter to true.
Bastion banner (BastionBanner)	https://aws-quickstart.s3.amazonaws.com/quickstart-linux-bastion/scripts/banner_message.txt	The banner text to display upon login.
Enable TCP forwarding (EnableTCPForwarding)	false	Setting this value to true will enable TCP forwarding (SSH tunneling). This can be useful but it is also a security risk, so we recommend that you keep the default (disabled) setting unless required.
Enable X11 forwarding (EnableX11Forwarding)	false	Setting this value to true will enable X Windows over SSH. X11 forwarding can be very useful but it is also a security risk, so we recommend that you keep the default (disabled) setting unless required.

Cisco Blockchain configuration:

Parameter label (name)	Default	Description
Key pair name (KeyPairName)	<i>Requires input</i>	The name of an existing public/private key pair, which allows you to securely connect to your instance after it launches.
Allowed Blockchain access CIDR (BlockChainAccessCIDR)	<i>Requires input</i>	Allowed CIDR block for external blockchain access.
Blockchain instance type (InstanceType)	t2.xlarge	Amazon EC2 instance type for the blockchain instances.
Elasticsearch version (ElasticsearchVersion)	6.3	User-defined Amazon ES version. For more information, see the Amazon ES documentation .

Parameter label (name)	Default	Description
Elasticsearch instance type (ESInstanceType)	t2.small. elasticsearch	Instance type for Amazon ES.

AWS Quick Start configuration:

Parameter label (name)	Default	Description
Quick Start S3 bucket name (QSS3BucketName)	aws-quickstart	S3 bucket name for the Quick Start assets. This string can include numbers, lowercase letters, uppercase letters, and hyphens (-). It cannot start or end with a hyphen (-).
Quick Start S3 key prefix (QSS3KeyPrefix)	quickstart-cisco- blockchainplatform/	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.

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

[View template](#)

Network configuration:

Parameter label (name)	Default	Description
VPC ID (VPCID)	<i>Requires input</i>	ID of the VPC (e.g., vpc-0343606e).
Public subnet 1 ID (PublicSubnet1ID)	<i>Requires input</i>	ID of the public subnet 1 that you want to provision the first bastion into (e.g., subnet-a0246dcd).
Public subnet 2 ID (PublicSubnet2ID)	<i>Requires input</i>	ID of the public subnet 2 you want to provision the second bastion into (e.g., subnet-e3246d8e).
Public subnet 3 ID (PublicSubnet3ID)	<i>Requires input</i>	ID of the public subnet 3 you want to provision the second bastion into (e.g., subnet-b1f4a2cd).
Private subnet 1 ID (PrivateSubnet1ID)	<i>Requires input</i>	ID of the private subnet 1 in Availability Zone 1 (e.g., subnet-a0246dcd).
Private subnet 2 ID (PrivateSubnet2ID)	<i>Requires input</i>	ID of the private subnet 2 in Availability Zone 2 (e.g., subnet-e3246d8e).
Private subnet 3 ID (PrivateSubnet3ID)	<i>Requires input</i>	ID of the private subnet 3 in Availability Zone 3 (e.g., subnet-b1f4a2cd).
Allowed SSH access CIDR (RemoteAccessCIDR)	<i>Requires input</i>	Allowed CIDR block for external SSH access.

Cisco Blockchain configuration:

Parameter label (name)	Default	Description
Key pair name (KeyPairName)	<i>Requires input</i>	Enter a public/private key pair. If you do not have one in this region, please create it before continuing.
Allowed Blockchain access CIDR (BlockChainAccessCIDR)	<i>Requires input</i>	Allowed CIDR block for external blockchain access.
Blockchain instance type (InstanceType)	t2.xlarge	Amazon EC2 instance type for the blockchain instances.
Elasticsearch version (ElasticsearchVersion)	6.3	User-defined Amazon ES version. For more information, see the Amazon ES documentation .
Elasticsearch instance type (ESInstanceType)	t2.small. elasticsearch	Instance type for Amazon ES.

AWS Quick Start configuration:

Parameter label (name)	Default	Description
Quick Start S3 bucket name (QSS3BucketName)	aws-quickstart	S3 bucket name for the Quick Start assets. This string can include numbers, lowercase letters, uppercase letters, and hyphens (-). It cannot start or end with a hyphen (-).
Quick Start S3 key prefix (QSS3KeyPrefix)	quickstart-cisco- blockchainplatform/	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.

- 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**.
- 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.
- Choose **Create** to deploy the stack.
- Monitor the status of the stack. When the status is **CREATE_COMPLETE**, Cisco Blockchain Platform is ready.
- The DNS names of the blockchain nodes are displayed on the **Outputs** tab of the **Blockchain Node Template** stack using the key names `Peer0NLB`, `Peer1NLB`, and `Peer2NLB`. These parameters are required for further testing and usage of the created Cisco Blockchain network.

Step 4. Test the Deployment

After the Cisco Blockchain Platform stack installation is complete, you have two options for testing the deployment:

- SDK
- Cisco Blockchain Platform Developer Center

SDK

Cisco Blockchain Platform Software Development Kit (SDK) is a nodejs tool kit available for developers to test the blockchain network and to develop Blockchain Client applications that interact with the nodes to submit transactions and query data from the blockchain.

Please see the [Cisco Blockchain for AWS resource page](#) to download the SDK and the related SDK documentation.

Cisco Blockchain Platform Developer Center

Developer Center is a modern web UI that is available as an add-on for the Quick Start deployment. Enterprise Blockchain developers will find this powerful tool helpful when developing, testing, and deploying smart contracts onto Cisco Blockchain Platform.

Please see the [Cisco Blockchain for AWS resource page](#) to download the Developer Center AWS CloudFormation template and the related documentation.

Security

Public Keys

Cisco Blockchain Platform uses public key infrastructure (PKI) for node identities and for signing transactions. Secure key management practice is important to ensure the security of the Cisco Blockchain Platform nodes, applications, and the stored data. This Quick Start creates unique self-signed keys as part of the deployment. They are stored in the private Amazon S3 bucket and minimally shared between peer nodes. SDK client applications will also need access to this private S3 bucket to create key pairs for client identities. Care should be taken to keep the S3 bucket secure.

This key management scheme is provided primarily as a developer convenience and for demonstration purposes, and should not be used for production data.

Access to Peer Nodes

Bastion hosts are available to log in to Cisco Blockchain Platform nodes for troubleshooting purposes. Care should be taken to keep the bastion key pair secure. For more information, see <https://docs.aws.amazon.com/quickstart/latest/linux-bastion/architecture.html>.

Troubleshooting

Q. I received an Elastic IP address error when creating the bastion security group.

A. We recommend having five Elastic IP addresses available within your region before deploying Cisco Blockchain Platform.

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. (For Windows, look at the log files in `%ProgramFiles%\Amazon\EC2ConfigService` and `C:\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](#) on the AWS website.

Q. I encountered a `DELETE_FAILED` error when I deleted the stack created by the Quick Start.

A. If deletion of the AWS CloudFormation stack fails with a `DELETE_FAILED` error and the Status Reason in the **Blockchain Node Template** stack shows the message, **The following resource(s) failed to delete: [S3BucketPrivate]**, try manually deleting the contents of the S3 bucket and then try deleting the stack again. You can find the S3 bucket name in the **S3PrivateBucket** key in the **Outputs** section of the Blockchain Node Template stack.

Q. I encountered a **Role deletion failed** message for the [AWSServiceRoleForAmazonElasticsearchService](#) service-linked role when I deleted the stack created by the Quick Start.

A. The Quick Start template uses a service-linked role for Amazon EC2 instances to reach Amazon ES. The service-linked role that was created during deployment might not have been removed when the stack was deleted. The work around is to manually delete that role in the IAM console.

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 links in this guide 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](#).

GitHub Repository

You can visit our [GitHub repository](#) 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 EBS
<https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/AmazonEBS.html>
- Amazon EC2
<https://aws.amazon.com/documentation/ec2/>
- Amazon VPC
<https://aws.amazon.com/documentation/vpc/>
- Amazon S3
<https://docs.aws.amazon.com/s3/>
- ELB
<https://docs.aws.amazon.com/elasticloadbalancing/>
- Amazon ES
<https://docs.aws.amazon.com/elasticsearch-service/>
- AWS CloudFormation
<https://aws.amazon.com/documentation/cloudformation/>

Cisco Blockchain documentation

- Cisco Blockchain Platform on AWS resources page
<https://docs.ciscoblockchain.io/aws-cloud>
- Cisco Blockchain whitepaper
<https://www.cisco.com/c/en/us/solutions/digital-transformation/blockchain/index.html?dtid=ossdc000283>

Quick Start reference deployments

- AWS Quick Start home page
<https://aws.amazon.com/quickstart/>

Document Revisions

Date	Change	In sections
November 2018	Initial publication	—

© 2018, Amazon Web Services, Inc. or its affiliates, and Cisco Systems. 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.