Microsoft SharePoint Server 2019 on the AWS Cloud

Quick Start Reference Deployment

AWS Quick Start team

May 2016
Last updated: December 2019 (see revisions)

This guide is also available in HTML format at https://docs.aws.amazon.com/quickstart/latest/sharepoint/.
Contents

About This Guide ........................................................................................................................................... 3
About Quick Starts ........................................................................................................................................... 4
Overview ......................................................................................................................................................... 4
SharePoint Server 2019 on AWS ...................................................................................................................... 4
Cost and Licenses ............................................................................................................................................. 4
AWS Services .................................................................................................................................................... 6
Architecture ...................................................................................................................................................... 7
VPC Configuration ........................................................................................................................................... 8
Active Directory Domain Services .................................................................................................................. 10
Remote Administration ...................................................................................................................................... 11
Deployment Steps ............................................................................................................................................ 12
Step 1. Prepare an AWS Account .................................................................................................................... 12
Step 2. Download the SharePoint Software .................................................................................................... 14
Step 3. Configure AWS prerequisites ................................................................................................................ 15
Step 4. Prepare the SQL Server Cluster for Always On Availability Groups ................................................ 16
Step 5. Launch the Quick Start ........................................................................................................................ 23
  Option 1: Deploy SharePoint into a new VPC ............................................................................................... 24
  Option 2: Deploy a SharePoint farm into an existing VPC ......................................................................... 28
  Option 3: Deploy a single SharePoint server into an existing VPC ............................................................... 30
Step 6. Create Initial Content .......................................................................................................................... 33
Step 7. Make the SharePoint Databases Highly Available ................................................................................ 36
  Enable Full Recovery Mode ........................................................................................................................... 37
  Back Up the Databases ................................................................................................................................ 39
  Add the Databases to the Availability Group ................................................................................................. 39
  Provide Log Access to the Secondary Database Server ............................................................................... 42
  Set Up Multiple-Subnet Failover .................................................................................................................... 43
Step 8. Test Automatic Failover ........................................................................................................................ 43
Troubleshooting ................................................................................................................................. 44
Additional Resources .......................................................................................................................... 49
Appendix A: Server Role Architecture .............................................................................................. 51
  Single-Server Topology ...................................................................................................................... 51
  Multiple-Server Topology .................................................................................................................... 51
    Load Balancing ................................................................................................................................. 53
    Database Tier .................................................................................................................................. 53
Intranet SharePoint Server Farm on AWS ............................................................................................ 54
Security ................................................................................................................................................ 55
  Security Groups .................................................................................................................................. 56
  Network ACLs ..................................................................................................................................... 56
  Secure Extranet Publishing ................................................................................................................... 57
EC2 Instance Types ................................................................................................................................ 58
GitHub Repository .................................................................................................................................. 60
Document Revisions ............................................................................................................................. 60

About This Guide
This Quick Start reference deployment guide discusses architectural considerations and configuration steps for building a Microsoft SharePoint Server 2019 environment on the Amazon Web Services (AWS) cloud. It also provides links for viewing and launching AWS CloudFormation templates that automate the deployment.

This guide is for IT infrastructure architects, administrators, and DevOps professionals who are planning to implement or extend SharePoint Server 2019 on the AWS Cloud. The guide requires basic familiarity with SharePoint Server architecture and management. For more information about SharePoint Server, including general guidance and best practices, consult the Microsoft SharePoint product documentation.
About Quick Starts

Quick Starts are automated reference deployments for key enterprise workloads on the AWS Cloud. Each Quick Start launches, configures, and runs the AWS compute, network, storage, and other services required to deploy a specific workload on AWS, using AWS best practices for security and availability.

Overview

SharePoint Server 2019 on AWS

The Amazon Web Services (AWS) cloud provides a suite of infrastructure services that enable you to deploy SharePoint Server 2019 securely, affordably, and with high availability. Running SharePoint Server on the AWS Cloud gives you flexibility and agility, and you can fully customize and extend SharePoint for your business processes.

This Quick Start implementation guide walks you through the steps to automatically deploy an enterprise SharePoint Server 2019 architecture in your own AWS account. The automatic deployment, including Active Directory and SQL Server, takes approximately 1-2 hours.

Cost 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 the Quick Start itself.

The AWS CloudFormation template for the SharePoint Server 2019 Quick Start includes configuration parameters that you can customize, and some settings, such as the instance types and the number of instances, can greatly affect the cost of the deployment.

AWS has published a whitepaper that shows how to estimate the cost of your SharePoint deployment. You have a wide array of options for building your SharePoint farm, and it’s not possible to cover them all in that whitepaper or in this guide. The following table offers a model based on some key assumptions. You can open an example in the Simple Monthly Calculator to change the configuration and revise any of these estimates to fit your scenario.

- It assumes that you launch the Quick Start AWS CloudFormation template with the default parameters. The traditional topology architecture shown in Figure 2 includes 10 instances.
- It assumes 15 TiB of outbound data traffic per month (based on 50 MiB per day for 20,000 users). This accounts for about $1,300 of the monthly cost.
• It assumes storage and backups for about 5 TiB of data. This accounts for about $2,100 of the monthly cost.

<table>
<thead>
<tr>
<th>Model</th>
<th>Up-front cost</th>
<th>Monthly cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>On-Demand Instances, license Windows Server from AWS, use free trial SharePoint Server and SQL Server licenses</td>
<td>$0</td>
<td>$8,900</td>
</tr>
</tbody>
</table>

This approach represents an average of 40 percent savings over the typical cost to deploy an on-premises SharePoint solution. You can get an idea of the savings you may see for your specific deployment by using the [AWS TCO Calculator](#).

![Figure 1: Use the AWS Simple Monthly Calculator to estimate the deployment costs for SharePoint Server](aws.png)

For more information about instance pricing, see [Instance Purchasing Options](#) in the AWS documentation. Please note that AWS prices are subject to change.

This SharePoint Quick Start (using free trial licenses for SQL Server and SharePoint Server) is most appropriate for a trial or proof-of-concept project.

By default, this Quick Start installs the evaluation edition of SharePoint Server 2019 and SQL Server provided by Microsoft. For production environments, you can license SharePoint Server and SQL Server through the [Microsoft License Mobility through Software Assurance](#) program, and use your own product key during deployment. This Quick Start is not appropriate for development or test environments in which you leverage your existing Microsoft Developer Network (MSDN) licenses, because it does not support Dedicated Hosts or Dedicated Instances. For details, see the [MSDN on AWS](#) page.
AWS Services

The core AWS components used by this Quick Start include the following AWS services. (If you are new to AWS, see Getting Started with AWS.)

- **AWS CloudFormation** – AWS CloudFormation gives you an easy way to create and manage a collection of related AWS resources, and provision and update them in an orderly and predictable way. You use a template to describe all the AWS resources (e.g., EC2 instances) that you want. You don’t have to individually create and configure the resources or figure out dependencies—AWS CloudFormation handles all of that.

- **Amazon VPC** – The Amazon Virtual Private Cloud (Amazon VPC) service lets you provision a private, isolated section of the AWS Cloud where you can launch AWS services and other resources in a virtual network that you define. You have complete control over your virtual networking environment, including selection of your own IP address range, creation of subnets, and configuration of route tables and network gateways.

- **Amazon EC2** – The Amazon Elastic Compute Cloud (Amazon EC2) service enables you to launch virtual machine instances with a variety of operating systems. You can choose from existing Amazon Machine Images (AMIs) or import your own virtual machine images.

- **NAT Gateway** – NAT Gateway is an AWS managed service that controls network address translation (NAT) gateway resources. A NAT gateway is a type of network address translation (NAT) device that enables instances in a private subnet to connect to the internet or to other AWS services, but prevents the internet from connecting to those instances.

- **IAM** – AWS Identity and Access Management (IAM) enables you to securely control access to AWS services and resources for your users. With IAM, you can manage users, security credentials such as access keys, and permissions that control which AWS resources users can access, from a central location.

- **Amazon S3** – Amazon Simple Storage Service (Amazon S3) provides developers and IT teams with secure, durable, highly scalable, cost-effective object storage. Amazon S3 is easy to use and includes a web services interface to store and retrieve any amount of data from anywhere on the web. Object storage is not appropriate for workloads that require incremental data insertions, such as databases. However, Amazon S3 is an excellent service for storing snapshots of Amazon Elastic Block Store (Amazon EBS) volumes.

- **AWS Systems Manager** - AWS Systems Manager gives you visibility and control of your infrastructure on AWS. Systems Manager provides a unified user interface so that you
can view operational data from multiple AWS services, and it allows you to automate operational tasks across your AWS resources. With Systems Manager, you can do the following:

- Group resources, such as EC2 instances, S3 buckets, or Amazon Relational Database Service (Amazon RDS) instances, by application
- View operational data for monitoring and troubleshooting
- Take action on your groups of resources

Systems Manager simplifies resource and application management, shortens the time to detect and resolve operational problems, and makes it easy to operate and manage your infrastructure securely at scale.

Architecture

There are a number of ways to design the topology of your SharePoint farm depending on your requirements. In SharePoint 2016, Microsoft added a feature called MinRole, which helped simplify deployment, performance, and reliability of SharePoint farms. This approach is used for the multiple-server CloudFormation template provided by this Quick Start. (The single-server option is deployed as a custom role so that it can run more services on its own.) Both the multiple-server and single-server topologies are covered in detail in Appendix A.

Deploying this Quick Start with the default parameters builds the following highly available SharePoint environment based on the multiple-server topology in the AWS Cloud.
The AWS CloudFormation templates provided with this Quick Start deploy the SharePoint servers in this diagram. Active Directory and SQL Server are deployed via their corresponding Quick Starts. See the Deployment Steps section for more details. The template deploys a highly available architecture that includes redundant servers for SharePoint Server 2019 in two Availability Zones.

The following sections describe these components of the architecture in more detail. For more information about the server role architecture, including a detailed discussion of multiple-server and single-server topologies, see Appendix A.

VPC Configuration

When deploying a Windows-based architecture on the AWS Cloud, we recommend a VPC configuration that supports the following requirements:

- Critical workloads should be placed in a minimum of two Availability Zones to provide high availability.
- Internal application servers and other non-internet facing servers should be placed in private subnets to prevent direct access to these instances from the internet.
Remote Desktop Gateways (RD Gateways) should be deployed into public subnets in each Availability Zone for remote administration. Other components, such as reverse proxy servers, can also be placed into these public subnets if needed.

For details on the VPC design used in this reference, see the Active Directory Domain Services Quick Start deployment guide.

Based on these best practices, the VPC Quick Start deploys the following base-level VPC framework to support the SharePoint Server 2019 infrastructure:

![VPC Architecture Diagram](attachment:image.png)

**Figure 3: VPC architecture on the AWS Cloud**

As shown in Figure 3, NAT gateways are deployed into the public subnets. The public subnets have a route to the internet directly through the internet gateway attached to the VPC.

Instances that will be deployed in the private subnets have no direct route to the internet. Instead, instances in private subnets use private routes to send internet traffic to the NAT.
gateways in the public subnets. This architecture isolates your critical workloads from direct internet access.

**Active Directory Domain Services**

To provide user authentication and authorization, the Microsoft SharePoint servers in this reference architecture use Active Directory Domain Services (Active Directory DS). As you deploy your environment, you should place at least one domain controller in a private subnet in each Availability Zone for redundancy and high availability.

Notice that in Figure 4, we’ve now included a domain controller in the Active Directory tier in each Availability Zone.

There are two ways to use Active Directory DS in the AWS Cloud:

- **Cloud only** – This is the architecture shown in Figure 4. This type of architecture means that your entire Active Directory forest exists only within the AWS Cloud. With a cloud-only Active Directory DS architecture, there are no on-premises domain controllers.

- **Hybrid** – The hybrid architecture takes advantage of your existing Active Directory DS environment. You can extend your private, on-premises network to AWS so the
resources in the cloud can utilize your existing Active Directory infrastructure. In a hybrid architecture, we recommend that you also deploy domain controllers for your existing Active Directory forest to the AWS Cloud. We recommend this configuration primarily to help ensure that the application servers deployed in AWS remain functional and available in the event of an on-premises outage.

The [Quick Start for AD DS on AWS](https://aws.amazon.com/quickstart/) covers our best practices and recommendations for deploying Active Directory DS on AWS. The process outlined in this SharePoint Quick Start asks you to launch the Active Directory DS Quick Start, which deploys the Active Directory environment based on those best practices.

**Remote Administration**

As we design the architecture for a highly available SharePoint farm, we should also design for highly available and secure remote access. We can do this by deploying an RD Gateway in each Availability Zone. In the case of an Availability Zone outage, this architecture allows access to the resources that may have failed over to the other Availability Zone.

The RD Gateway uses the Remote Desktop Protocol (RDP) over HTTPS to establish a secure, encrypted connection between remote administrators on the internet and Windows-based EC2 instances, without needing to configure a virtual private network (VPN) connection. This allows you to reduce the attack surface on your Windows-based instances while providing a remote administration solution for administrators.
The AWS CloudFormation templates provided in this Quick Start automatically deploy the architecture described in the Quick Start for Remote Desktop Gateway on AWS. After you have launched your SharePoint infrastructure using a deployment option in this guide, you will initially connect to your instances using a standard RDP TCP port 3389 connection. You can then follow the steps in the Quick Start for Remote Desktop Gateway to secure future connections via HTTPS.

Deployment Steps

To build the SharePoint environment shown in Figure 2 on the AWS Cloud, follow these steps.

**Step 1. Prepare an 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. Part of the sign-up process involves receiving a phone call and entering a PIN using the phone keypad.

2. Use the Region selector in the navigation bar to choose the AWS Region where you want to deploy SharePoint on AWS.
Amazon EC2 locations are composed of Regions and Availability Zones. For a current list of the Availability Zones and the services that are available in them, see the Region Table. Each Region includes at least two Availability Zones, which are isolated from each other with respect to power, network backbone, etc. Deploying your cloud applications across two Availability Zones helps you achieve high availability, even in the face of natural disasters that might impact a single Availability Zone.

**Tip** Consider choosing a Region closest to your data center or corporate network to reduce network latency between systems running on AWS and the systems and users on your corporate network.

3. Create a key pair in your preferred Region. To do this, in the navigation pane of the Amazon EC2 console, choose Key Pairs, Create Key Pair, type a name, and then choose Create.

![Figure 6: Creating a key pair](image)

Amazon EC2 uses public-key cryptography to encrypt and decrypt login information. To be able to log in to your instances, you must create a key pair. With Windows instances, we use the key pair to obtain the administrator password via the Amazon EC2 console, and then log in using Remote Desktop Protocol (RDP), as explained in the step-by-step instructions in the Amazon Elastic Compute Cloud User Guide.

4. Verify that you have available Elastic IP addresses in your account. The AWS CloudFormation stack you will launch will automatically create Elastic IP addresses as needed for the SharePoint architecture. Each AWS account has a default quota of five
addresses. To ensure beforehand that the AWS CloudFormation template will not fail because you’ve reached this quota, we recommend that you manually create two Elastic IP addresses in your AWS account and then delete them before you launch the Microsoft SharePoint AWS CloudFormation stack.

5. If necessary, request a service quota increase for the instance types used for the deployment. You might need to request an increase if you already have an existing deployment that uses the same instance types as your SharePoint architecture or if you need additional Elastic IP addresses. To do this, on the Service Quotas console, for each instance type that you want a service quota increase, choose Request quota increase, and then complete the fields in the quota increase form. It can take a few days for the new service quota to become effective.

Step 2. Download the SharePoint Software
To access the SharePoint Server 2019 installation media, you need to place the contents of the SharePoint installation ISO file in a .zip file. When creating this .zip file, you have the option of “slip streaming” additional cumulative updates into the file, which means that these updates will be automatically installed when SharePoint is installed. This can help you deploy SharePoint to a specific update level when it is first installed. The following steps show you how to create the .zip file and upload it to Amazon S3.

1. Download the evaluation edition from Microsoft Download Center.

—or—

Obtain the ISO file from MSDN, if you have an MSDN account with licenses for server software.

2. Open the ISO file (in Windows, double-click the file to mount the ISO and open it), and copy the contents to a temporary location on your system.

3. (Optional) To include any additional updates, use the slip-streaming option:
   a. Download the updates that you want to include.
   b. Open a command prompt, and change the current path to the folder where you downloaded the updates.
   c. To extract the updates to the updates folder in your installation media temporary folder, run the following command:

   ```
   UpdateFileName.exe /extract:PathToUpdatesFolder
   ```

   where:
UpdateFileName is the name of the file you downloaded for the update (for example, sts2019-kb4464556-fullfile-x64-g1b.exe).

PathToUpdatesFolder is the path to the folder called Updates that exists in the installation media that you extracted in Step 2. For example, if you extracted the media to C:\SPInstall, the path is C:\SPInstall\Updates.

4. Select all of the files and folders in the directory on your system to which you extracted the installation media, open the context (right-click) menu, and choose Send To, Compressed (zipped) folder. This creates a .zip file of your installation media that can be used for the installation.

5. Sign in to your AWS account and open the Amazon S3 console.

6. Choose Create Bucket.

7. Complete the Create a Bucket dialog box:
   a. In the Bucket Name box, enter a globally unique name for the bucket (for example, your account name).
   b. In the Region list, select the AWS Region where you plan to launch the Quick Start.
   c. Choose Create.

   The console displays your new bucket in the Buckets pane.

8. Choose the new bucket to navigate to it.


10. In the Upload – Select Files and Folders dialog box, choose Add Files.

11. In the file selection dialog box, browse to the .zip file that you created earlier, and choose Open, Start Upload.

Make a note of the bucket name and the file name that you used for your zip file. You will use these later when you launch the SharePoint stack.

Step 3. Configure AWS prerequisites

Note This step is not required if you are deploying the master stack into a new VPC.

Before you launch the SharePoint stack, you need to configure several prerequisites. If you have deployed the Active Directory Domain Services and SQL Server Quick Starts, they will have provisioned the following components:
• A VPC with two private subnets where SharePoint will run
• A Windows Active Directory server that runs the Domain Name Server (DNS) service
• A SQL Server 2016 or 2017 instance (or a highly available instance that has an Always On availability group configured with a listener attached to it)

If you have not yet deployed these prerequisites, deploy the Active Directory Domain Services and SQL Server Quick Starts. Details and instructions are in the deployment guides:

• Active Directory Domain Services on AWS
• SQL Server on AWS

Step 4. Prepare the SQL Server Cluster for Always On Availability Groups

**Note**  This step is not required if you are deploying the master stack into a new VPC.

The SQL Server on AWS Quick Start deploys two SQL Server instances that are ready to be configured as Always On availability groups. To simplify how SharePoint will deploy its databases we need to configure this availability group with an appropriate listener before we deploy SharePoint.

1. Establish an RDP session to the WSFCNODE1 instance. Start SQL Server Management Studio, and then choose **Connect** to connect to the local server.
2. To create an availability group, we need an empty database to seed it with first. To create this, in **Object Explorer**, open the context (right-click) menu for the **Databases** node, and choose **New Database**. For the database name, enter **AAGSeed**, and leave all other settings at default values. Choose **OK** at the bottom of the window to create the database.

3. To make a backup of the new database, expand the **Databases** node, open the context (right-click) menu for the database name, choose **Tasks**, and then choose **Back Up**. Keep the default settings, and then choose **OK** to perform the backup.
4. When the databases have been backed up, in **Object Explorer**, open the context (right-click) menu for **AlwaysOn High Availability**, and then choose **New Availability Group Wizard**.

5. Provide a name for the availability group, and choose **Next**.

   In this example, we use **SharepointAG** as the name of the availability group.
6. Select the database that you created and backed up, and then choose **Next**.

7. On the **Specify Replicas** page, choose **Add Replica**, and add **WSFCNODE2** as a replica. Make sure that the check boxes for automatic failover and synchronous replication are selected.
8. Choose the **Listener** tab. Provide a listener DNS name, the port number to listen on (which will be 1433), and the IP address for each WSFC node. Based on the template default settings, the IP addresses should be **10.0.0.102** for WSFCNODE1, and **10.0.64.102** for WSFCNODE2. After you fill out the page as shown in Figure 12, choose **Next**.
9. On the Select Data Synchronization page, choose Automatic seeding, and then choose Next.
10. Accept the default settings on the remaining pages of the wizard, and then choose **Next** and **Finish** to build the availability group. Make sure that the wizard has completed successfully before continuing to the next step.
11. To ensure that the DNS records for your availability group listener can be resolved in the secondary Active Directory site in Availability Zone 2, force Active Directory replication from DC1 to DC2 by connecting to DC1 and running the command `repadmin /syncall /A /e /P`.

![Administrator: Command Prompt](image)

**Figure 15: Forcing Active Directory replication from Availability Zone 1 to Availability Zone 2**

The endpoint should now resolve as `spdb.[fully qualified domain name]` (e.g., `spdb.example.com`), and we will use this as the database server name for the SharePoint stack.

**Step 5. Launch the Quick Start**

The automated AWS CloudFormation template deploys SharePoint in multiple Availability Zones into a VPC. There are three deployment options.

1. Choose one of the following options to deploy the AWS CloudFormation template into your AWS account.
   
   - **Option 1**: Deploy SharePoint into a new VPC
     
     ![Launch Quick Start](image)
     
   - **Option 2**: Deploy a SharePoint farm into an existing VPC
     
     ![Launch Quick Start](image)
     
   - **Option 3**: Deploy a single SharePoint server into an existing VPC
     
     ![Launch Quick Start](image)

2. On the **Select Template** page, keep the default setting for the template URL, and then choose **Next**.

3. On the **Specify Details** page, review the parameters for the template. Provide values for the required parameters based on the selected option.
In the following tables, parameters are listed by category and described separately for the three deployment options:

- Parameters for deploying SharePoint into a new VPC
- Parameters for deploying a SharePoint farm into an existing VPC
- Parameters for deploying a single SharePoint server into an existing VPC

When you finish reviewing and customizing the parameters, choose Next.

Option 1: Deploy SharePoint into a new VPC

View template

Network configuration:

<table>
<thead>
<tr>
<th>Parameter label (name)</th>
<th>Default</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Key pair name (KeyPairName)</td>
<td>Requires input</td>
<td>Public/private key pair, which 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>Availability Zones (AvailabilityZones)</td>
<td>Requires input</td>
<td>List of Availability Zones to use for the subnets in the VPC. Only two Availability Zones are used for this deployment, and the logical order is preserved.</td>
</tr>
<tr>
<td>Private subnet 1 CIDR (PrivateSubnet1CIDR)</td>
<td>10.0.0.0/19</td>
<td>CIDR block for the Active Directory Server tier located in Availability Zone 1.</td>
</tr>
<tr>
<td>Private subnet 2 CIDR (PrivateSubnet2CIDR)</td>
<td>10.0.64.0/19</td>
<td>CIDR block for the Active Directory Server tier located in Availability Zone 2.</td>
</tr>
<tr>
<td>Public subnet 1 CIDR (PublicSubnet1CIDR)</td>
<td>10.0.32.0/20</td>
<td>CIDR block for the public (DMZ) subnet located in Availability Zone 1.</td>
</tr>
<tr>
<td>Public subnet 2 CIDR (PublicSubnet2CIDR)</td>
<td>10.0.96.0/20</td>
<td>CIDR block for the public (DMZ) subnet located in Availability Zone 2.</td>
</tr>
<tr>
<td>VPC CIDR (VPCCIDR)</td>
<td>10.0.0.0/16</td>
<td>CIDR block for the VPC.</td>
</tr>
</tbody>
</table>

Microsoft SharePoint configuration:

<table>
<thead>
<tr>
<th>Parameter label (name)</th>
<th>Default</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SharePoint binary bucket (SPBinaryBucket)</td>
<td>Requires input</td>
<td>The name of the bucket that contains your SharePoint binaries.</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>SharePoint binary key (SPBinaryKey)</td>
<td>Requires input</td>
<td>The path (key) to the installer .zip file in the S3 bucket.</td>
</tr>
<tr>
<td>SharePoint product key (SPProductKey)</td>
<td>M692G-8N2JP-GG8B2-2W2P7-YY7J6</td>
<td>The product key to use with SharePoint.</td>
</tr>
<tr>
<td>Farm (Farm)</td>
<td>Deploy a SharePoint farm</td>
<td>Choose whether you want to deploy a SharePoint farm or a single SharePoint server.</td>
</tr>
<tr>
<td>SharePoint Server NetBIOS name (SPServerNetBIOSName)</td>
<td>SPServer</td>
<td>The NetBIOS name (for a single-server deployment) or name prefix (for a farm deployment) of the SharePoint server or servers (up to 14 characters).</td>
</tr>
<tr>
<td>DNS name central admin (DNSPrefixCentralAdmin)</td>
<td>spadmin</td>
<td>The domain name to use for the central admin account (without the domain suffix; use admin, not admin.domain.com).</td>
</tr>
<tr>
<td>DNS name main site (DNSPrefixMainSite)</td>
<td>sharepoint</td>
<td>The domain name to use for the main SharePoint site (without the domain suffix; use site, not site.domain.com).</td>
</tr>
<tr>
<td>DNS server IP address (DNSServerIP)</td>
<td>10.0.0.10</td>
<td>The IP address of a DNS server on the network that will allow DNS entries to be created as the admin account.</td>
</tr>
</tbody>
</table>

**Microsoft Active Directory configuration:**

<table>
<thead>
<tr>
<th>Parameter label (name)</th>
<th>Default</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Domain controller 1 instance type (ADServer1InstanceType)</td>
<td>m4.xlarge</td>
<td>EC2 instance type for the first Active Directory instance.</td>
</tr>
<tr>
<td>Domain controller 1 NetBIOS name (ADServer1NetBIOSName)</td>
<td>DC1</td>
<td>NetBIOS name of the first Active Directory server (up to 15 characters).</td>
</tr>
<tr>
<td>Domain controller 1 private IP address (ADServer1PrivateIP)</td>
<td>10.0.0.10</td>
<td>Fixed private IP address for the first Active Directory server located in Availability Zone 1.</td>
</tr>
<tr>
<td>Domain controller 2 instance type (ADServer2InstanceType)</td>
<td>m4.xlarge</td>
<td>EC2 instance type for the second Active Directory instance.</td>
</tr>
<tr>
<td>Domain controller 2 NetBIOS name (ADServer2NetBIOSName)</td>
<td>DC2</td>
<td>NetBIOS name of the second Active Directory server (up to 15 characters).</td>
</tr>
<tr>
<td>Domain controller 2 private IP address (ADServer2PrivateIP)</td>
<td>10.0.64.10</td>
<td>Fixed private IP address for the second Active Directory server located in Availability Zone 2.</td>
</tr>
</tbody>
</table>
### Parameter label (name) | Default | Description
--- | --- | ---
Domain admin user name (DomainAdminUser) | StackAdmin | User name for the account that will be added as Domain Administrator. This is separate from the default Administrator account.

**Domain admin password (DomainAdminPassword)** | Requires input | Password for the domain admin user. Must be at least 8 characters and contain letters, numbers, and symbols.

Domain DNS name (DomainDNSName) | example.com | Fully qualified domain name (FQDN) of the forest root domain.

Domain NetBIOS name (DomainNetBIOSName) | example | NetBIOS name of the domain (up to 15 characters) for users of earlier versions of Windows.

### Microsoft SQL Server configuration:

| Parameter label (name) | Default | Description |
--- | --- | ---
Service account name (SQLServiceAccount) | sqlsa | User name for the SQL Server service account. This account is a Domain User. |
Service account password (SQLServiceAccountPassword) | Requires input | Password for the SQL Server service account. Must be at least 8 characters and contain letters, numbers, and symbols. |
Amazon-provided SQL Server license (SQLLicenseProvided) | no | Choose whether to license SQL Server from AWS Marketplace or provide your own license post-deployment. |
Data volume size (Volume1Size) | 500 | Volume size for the SQL Data drive, in GiB. |
Data volume type (Volume1Type) | gp2 | Volume type for the SQL Data drive. |
Data volume IOPS (Volume1IOPS) | 1000 | IOPS for the SQL Data drive. Use only when volume type is io1. |
Logs volume size (Volume2Size) | 500 | Volume size for the SQL Logs drive, in GiB. |
Logs volume type (Volume2Type) | gp2 | Volume type for the SQL Logs drive. |
Logs volume IOPS (Volume2IOPS) | 1000 | IOPS for the SQL Logs drive (only used when volume type is io1). |
TempDB volume size (Volume3Size) | 500 | Volume size for the SQL temporary database drive, in GiB. |
### Parameter label (name) | Default | Description
---|---|---
TempDB volume type (Volume3Type) | gp2 | Volume type for the SQL temporary database drive.  
TempDB volume IOPS (Volume3Iops) | 1000 | IOPS for the SQL temporary database drive. Use only when volume type is io1.
Availability group name (AvailabilityGroupName) | SQLAG1 | NetBIOS name of the SQL database or availability group (up to 15 characters).

**WSFC configuration:**

<table>
<thead>
<tr>
<th>Parameter label (name)</th>
<th>Default</th>
<th>Description</th>
</tr>
</thead>
</table>
| File server instance type (WSFCFileServerInstanceType) | m5.large | EC2 instance type for a file server for witness and replication folders.  
| File server private IP address (WSFCFileServerPrivateIP) | 10.0.0.200 | Primary private IP address for the file server located in Availability Zone 1.  
| WSFC node 1 instance type (WSFCNode1InstanceType) | r4.xlarge | EC2 instance type for the first WSFC node.  
| WSFC node 1 NetBIOS name (WSFCNode1NetBIOSName) | WSFCNode1 | NetBIOS name of the first WSFC node (up to 15 characters).  
| WSFC node 1 private IP address 1 (WSFCNode1PrivateIP1) | 10.0.0.100 | Primary private IP address for the first WSFC node located in Availability Zone 1.  
| WSFC node 1 private IP address 2 (WSFCNode1PrivateIP2) | 10.0.0.101 | Secondary private IP address for the WSFC cluster on the first WSFC node.  
| WSFC node 1 private IP address 3 (WSFCNode1PrivateIP3) | 10.0.0.102 | Third private IP address for the availability group listener on the first WSFC node.  
| WSFC node 2 instance type (WSFCNode2InstanceType) | r4.xlarge | EC2 instance type for the second WSFC node.  
| WSFC node 2 NetBIOS name (WSFCNode2NetBIOSName) | WSFCNode2 | NetBIOS name of the second WSFC node (up to 15 characters).  
| WSFC node 2 private IP address 1 (WSFCNode2PrivateIP1) | 10.0.64.100 | Primary private IP address for the second WSFC node located in Availability Zone 2.
### Microsoft RD Gateway configuration:

<table>
<thead>
<tr>
<th>Parameter label (name)</th>
<th>Default</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of RD Gateway hosts</td>
<td>1</td>
<td>The number of RD Gateway hosts to create (up to four).</td>
</tr>
<tr>
<td>Allowed RD Gateway external access CIDR</td>
<td>Requires input</td>
<td>Allowed CIDR block for external access to the RD Gateway hosts.</td>
</tr>
<tr>
<td>RD Gateway server instance type</td>
<td>m4.large</td>
<td>EC2 instance type for the RD Gateway instance.</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</td>
<td>aws-quickstart</td>
<td>S3 bucket name for the Quick Start assets. Quick Start bucket name can include numbers, lowercase letters, uppercase letters, and hyphens (-). It cannot start or end with a hyphen (-).</td>
</tr>
<tr>
<td>Quick Start S3 Key Prefix</td>
<td>quickstart-microsoft-sharepoint/</td>
<td>S3 key prefix for the Quick Start assets. Quick Start key prefix can include numbers, lowercase letters, uppercase letters, hyphens (-), and forward slash (/).</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>Key pair name</td>
<td>Requires input</td>
<td>Public/private key pair that allows you to securely connect to your instance after it launches.</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>VPC ID (VPCID)</td>
<td>Requires input</td>
<td>ID of the VPC (e.g., vpc-0343606e).</td>
</tr>
<tr>
<td>VPC CIDR (VPCCIDR)</td>
<td>10.0.0.0/16</td>
<td>CIDR block for the VPC.</td>
</tr>
<tr>
<td>Private subnet 1 ID (Subnet1ID)</td>
<td>Requires input</td>
<td>ID of the first private subnet into which SharePoint will be deployed (e.g., subnet-a0246dcd).</td>
</tr>
<tr>
<td>Private subnet 2 ID (Subnet2ID)</td>
<td>Requires input</td>
<td>ID of the second private subnet into which SharePoint will be deployed (e.g., subnet-b58c3d67).</td>
</tr>
<tr>
<td>DNS server IP address (DNSServerIP)</td>
<td>10.0.0.10</td>
<td>The IP address of a DNS server on the network that will allow DNS entries to be created as the admin account.</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>SharePoint server instance type (SPServerInstanceType)</td>
<td>m4.xlarge</td>
<td>EC2 instance type for the SharePoint servers.</td>
</tr>
<tr>
<td>Windows Server 2019 full base AMI (WS2019FULLBASE)</td>
<td>/aws/service/ami-windows-latest/Windows_Server-2019-English-Full-Base</td>
<td>If you’re replacing the Amazon-provided AMI for Windows Server 2019 with your own AMI, specify the AWS Systems Manager Parameter Store parameter namespace that provides your AMI ID.</td>
</tr>
<tr>
<td>SharePoint server NetBIOS name (SPServerNetBIOSName Prefix)</td>
<td>SPServer</td>
<td>NetBIOS name prefix of the SharePoint servers (up to 14 characters).</td>
</tr>
</tbody>
</table>

**Microsoft Active Directory configuration:**

<table>
<thead>
<tr>
<th>Parameter label (name)</th>
<th>Default</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Domain DNS name (DomainDNSName)</td>
<td>example.com</td>
<td>Fully qualified domain name (FQDN) of the forest root domain.</td>
</tr>
<tr>
<td>Domain NetBIOS name (DomainNetBIOSName)</td>
<td>example</td>
<td>NetBIOS name of the domain (up to 15 characters) for users of earlier versions of Windows.</td>
</tr>
<tr>
<td>Admin user name (ADAdminUserName)</td>
<td>Requires input</td>
<td>The user name of an account with rights to the domain (without the domain name).</td>
</tr>
<tr>
<td>Admin password (ADAdminPassword)</td>
<td>Requires input</td>
<td>The password of the Active Directory administrator account.</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>Security group ID (ADMemberSecurityGroup)</td>
<td>Requires input</td>
<td>The ID of the security group that will allow access to the domain controllers for domain join and other actions.</td>
</tr>
</tbody>
</table>

**Microsoft SQL Server configuration:**

<table>
<thead>
<tr>
<th>Parameter label (name)</th>
<th>Default</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Security group ID (SQLServerAccessSecurityGroup)</td>
<td>Requires input</td>
<td>The ID of the security group that will allow communication with the server or servers running SQL Server.</td>
</tr>
<tr>
<td>Database server name (SPDatabaseName)</td>
<td>Requires input</td>
<td>The name of the server running SQL Server that will store the SharePoint databases.</td>
</tr>
<tr>
<td>SQL Server admin user name (SQLAdminUserName)</td>
<td>Requires input</td>
<td>The user name of an account with rights to the server running SQL Server (without the domain name).</td>
</tr>
<tr>
<td>SQL Server admin password (SQLAdminPassword)</td>
<td>Requires input</td>
<td>The password of the SQL Server admin account.</td>
</tr>
</tbody>
</table>

**Microsoft SharePoint configuration**

<table>
<thead>
<tr>
<th>Parameter label (name)</th>
<th>Default</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SharePoint binary bucket (SPBinaryBucket)</td>
<td>Requires input</td>
<td>The name of the bucket that contains your SharePoint binaries.</td>
</tr>
<tr>
<td>SharePoint binary key (SPBinaryKey)</td>
<td>Requires input</td>
<td>The path (key) to the installer .zip file in the S3 bucket.</td>
</tr>
<tr>
<td>SharePoint product key (SPProductKey)</td>
<td>M692G-8N2JP-GG8B2-2W2P7-YY7J6</td>
<td>The product key to use with SharePoint. The default key is for SharePoint 2019 Enterprise Trial.</td>
</tr>
<tr>
<td>DNS name central admin (DNSPrefixCentralAdmin)</td>
<td>spadmin</td>
<td>The domain name to use for the central admin account (without the domain suffix; use admin, not admin.domain.com).</td>
</tr>
<tr>
<td>DNS prefix main site (DNSPrefixMainSite)</td>
<td>sharepoint</td>
<td>The domain name to use for the main SharePoint site (without the domain suffix; use site, not site.domain.com).</td>
</tr>
</tbody>
</table>

**Option 3: Deploy a single SharePoint server into an existing VPC**

View template

**Network configuration:**
<table>
<thead>
<tr>
<th>Parameter label (name)</th>
<th>Default</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Key pair name (KeyPairName)</td>
<td>Requires input</td>
<td>Public/private key pair that allows you to securely connect to your instance after it launches.</td>
</tr>
<tr>
<td>VPCID (VPCID)</td>
<td>Requires input</td>
<td>ID of the VPC (e.g., vpc-0343606e).</td>
</tr>
<tr>
<td>VPC CIDR (VPCCIDR)</td>
<td>10.0.0.0/16</td>
<td>CIDR block for the VPC.</td>
</tr>
<tr>
<td>Private subnet ID (SubnetID)</td>
<td>Requires input</td>
<td>ID of the private subnet into which SharePoint will be deployed (e.g., subnet-a0246dcd).</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>SharePoint server instance type (SPServerInstanceType)</td>
<td>m4.xlarge</td>
<td>EC2 instance type for the SharePoint server.</td>
</tr>
<tr>
<td>Windows Server 2019 full base AMI (WS2019FULLBase)</td>
<td>/aws/service/ami-windows-latest/Windows_Server-2019-English-Full-Base</td>
<td>If you’re replacing the Amazon-provided AMI for Windows Server 2019 with your own AMI, specify the AWS Systems Manager Parameter Store parameter namespace that provides your AMI ID.</td>
</tr>
<tr>
<td>SharePoint server NetBIOS name (SPServerNetBIOSName)</td>
<td>SP1</td>
<td>NetBIOS name of the SharePoint server (up to 15 characters).</td>
</tr>
</tbody>
</table>

**Microsoft Active Directory configuration:**

<table>
<thead>
<tr>
<th>Parameter label (name)</th>
<th>Default</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Domain DNS name (DomainDNSName)</td>
<td>example.com</td>
<td>Fully qualified domain name (FQDN) of the domain to join.</td>
</tr>
<tr>
<td>Domain NetBIOS name (DomainNetBIOSName)</td>
<td>example</td>
<td>NetBIOS name of the domain (up to 15 characters) for users of earlier versions of Windows.</td>
</tr>
<tr>
<td>Admin user name (ADAdminUserName)</td>
<td>Requires input</td>
<td>The username of an account with rights to the domain (without the domain name).</td>
</tr>
<tr>
<td>Admin password (ADAdminPassword)</td>
<td>Requires input</td>
<td>The password of the Active Directory administrator account.</td>
</tr>
<tr>
<td>Security group ID (ADMemberSecurityGroup)</td>
<td>Requires input</td>
<td>The ID of the security group that will allow access to the domain controllers for domain join and other actions.</td>
</tr>
</tbody>
</table>

**Microsoft SQL Server configuration:**
### Parameter label (name) | Default | Description
--- | --- | ---
Security group ID (SQLServerAccessSecurityGroup) | Requires input | The ID of the security group that will allow communication with the server (or servers) running SQL Server.

Database server name (SPDatabaseName) | Requires input | The name of the server running SQL Server that will store the SharePoint databases.

SQL Server admin user name (SQLAdminUserName) | Requires input | The user name of an account with rights to the server running SQL Server (without the domain name).

SQL Server admin password (SQLAdminPassword) | Requires input | The password of the SQL Server admin account.

---

### Microsoft SharePoint configuration

| Parameter label (name) | Default | Description |
--- | --- | ---|
SharePoint binary bucket (SPBinaryBucket) | Requires input | The name of the S3 bucket that contains your SharePoint binaries. |
SharePoint binary key (SPBinaryKey) | Requires input | The path (key) to the installer .zip file in the S3 bucket. |
SharePoint product key (SPProductKey) | M692G-8N2JP-GG8B2-2W2P7-YY7J6 | The product key to use with SharePoint. The default key is for SharePoint 2019 Enterprise Trial. |

4. 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**.

5. 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.

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

7. Monitor the status of the stack. If you have deployed into a new VPC, you will see that it spins off three separate sub-stacks. When the status of the **SharePoint Master** stack is **CREATE_COMPLETE**, the SharePoint environment is ready. The total process takes approximately 1-2 hours.
Step 6. Create Initial Content

Steps 6-8 walk you through testing high availability and automatic failover of your SharePoint servers; this does not apply to the single-server option. We’ll assume that you’ve used the default parameter values in the AWS CloudFormation template, that you deployed a SharePoint farm rather than a single server, and that the server name prefix value was SPServer. In this scenario, we’ll assume that the SharePoint farm is hosting an internal site, and we’ll set up a modern team site to validate our test.

After you have successfully launched the stack, create an RDP connection into the environment through one of the RD Gateway instances. You can retrieve the Elastic IP address for each RD Gateway instance from the Amazon EC2 console.

1. Establish an RDP session to the primary SharePoint server (SPServer1 if you deployed a SharePoint farm with the default parameters, or SP if you deployed a single SharePoint server with the default parameters). From the start menu, select SharePoint 2019 Central Administration. When prompted to run this as an administrator, choose OK for this prompt. A web browser will open to http://spadmin.example.com. When prompted for credentials, use the domain admin user name and password.
2. On the central administration home page, choose **Create site collections**.

3. Provide a **Title** for your site, and then choose the **Collaboration** tab and select the **team site** template.
You will also need to define a **Primary Site Collection Administrator** on this page. You can use the domain administrator account for this value, or any other valid user in the domain. When you finish filling out the form, choose **OK**.

4. Now that you have created a team site, navigate to **http://sharepoint.example.com**. When prompted for credentials, enter those of the user you selected as the site collection administrator in the step above. If the site loads, your SharePoint server is now ready for use.
Figure 19: The modern team site that was just created

Step 7. Make the SharePoint Databases Highly Available

Note This step is not required if deploying the master stack into a new VPC.
Enable Full Recovery Mode

1. Establish an RDP session to the WSFCNODE1 instance. Start SQL Server Management Studio, and then choose **Connect** to connect to the local server.

![Figure 20: Connecting to WSFCNODE1](image-url)
2. For the four Search databases, and the Usage database, you will need to enable Full Recovery mode before you can add them to the availability group. In **Object Explorer**, expand the **Databases** node.

3. For each database, choose the context (right-click) menu, and then choose **Properties**. In the **Database Properties** dialog box, choose the **Options** page, and then change **Recovery Model** to **Full**.

![Figure 21: Setting databases to full recovery mode](image)
Back Up the Databases

Next, make a backup of each SharePoint database. They will all share a common prefix that begins with the name of your CloudFormation stack.

1. To make a backup, choose the context (right-click) menu for the database name, choose "Tasks", and then choose "Back Up".

![Figure 22: Backing up a database](image)

2. Keep the default settings, and then choose "OK" to perform the backup.

Add the Databases to the Availability Group

Now add these databases to the availability group that you prepared in step 4.

1. In "Object Explorer", expand "Always On High Availability, Availability Groups".

2. Choose the context (right click) menu for the availability group, and choose "Add Database".
Figure 23: Select Add Database for the availability group
3. Select all the SharePoint databases to add to the group, and choose **Next**.

![Select Databases](image)

**Figure 24:** Selecting databases to include in the availability group

4. Connect to WSFCNODE2 to allow the databases to be added to the group there, and then choose the **Connect** button, and choose **Next**.

5. Choose **Automatic seeding** to tell SQL Server to automatically copy the databases to the replica.
Figure 25: Selecting automatic seeding to populate the secondary replicas

6. Ensure that the validation checks pass, and then choose **Next, Finish**.

The databases are now part of the availability group.

**Provide Log Access to the Secondary Database Server**

Now we need to ensure that the SharePoint service accounts have access to log in to the secondary database server if there is a failover event.
To do this, follow the instructions on the Microsoft [How to transfer logins and passwords between instances of SQL Server](#) support page. When you run the generated script against the secondary server, you might see some errors for accounts that already exist. This is normal. The script will, however, add the user logins that do not already exist on the secondary server. If you add additional server accounts or users at a later stage, be sure to run this script again to add them to the secondary server.

**Set Up Multiple-Subnet Failover**

1. After the logins have been copied to the secondary server, the last step is to tell SharePoint that multiple-subnet failover is required. To enable multiple-subnet failover for the SharePoint databases, on the `SPServer1` server, run Windows PowerShell with administrative permissions and execute the following PowerShell code.

```powershell
Add-PSSnapin Microsoft.SharePoint.PowerShell
$multiNetDBs = Get-SPDatabase | ?{$_MultiSubnetFailover -ne $true}
foreach ($db in $multiNetDBs) {
    $db.MultiSubnetFailover = $true
    $db.Update()
}
```

**Step 8. Test Automatic Failover**

After your externally facing SharePoint site is available, you can test automatic failover. The primary database server should be the first WSFC node, the Network Load Balancer for the back end (central admin site) distributes HTTP requests across `SPServer1` and `SPServer2`, and the Network Load Balancer for the front end distributes HTTP requests across `SPServer3` and `SPServer4`. To verify that automatic failover is functional, on the Amazon EC2 console, forcibly stop the first WSFC node, `SPServer1`, and `SPServer3`. You can stop the instances simultaneously to perform this test, as shown in Figure 27.
After you simulate a failure by stopping the instances, the SharePoint databases should fail over automatically to the second WSFC node. The load balancers should detect that SPServer1 and SPServer3 have both gone offline and should direct HTTP traffic to SPServer2 and SPServer4, respectively. You can revisit the site and the central admin site in your web browser to confirm that everything is still working.

**Troubleshooting**

When you deploy the Quick Start, if you encounter a CREATE_FAILED error instead of the CREATE_COMPLETE status code, we recommend that you re-launch 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.

**Important** When you set Rollback on failure to No, you’ll continue to incur AWS charges for this stack. Please make sure to delete the stack when you’ve finished troubleshooting.
The following table lists specific CREATE_FAILED error messages you might encounter while creating the stack in AWS CloudFormation.

<table>
<thead>
<tr>
<th>Error message</th>
<th>Possible cause</th>
<th>What to do</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>API: ec2: RunInstances Not authorized for images: ami-ID</strong></td>
<td>The template is referencing an AMI that has expired.</td>
<td>We refresh AMIs on a regular basis, but our schedule isn’t always synchronized with AWS AMI updates. If you get this error message, notify us, and we’ll update the template with the new AMI ID. If you’d like to fix the template yourself, you can <a href="#">download it</a> and update the Mappings section with the latest AMI ID for your Region.</td>
</tr>
<tr>
<td><strong>We currently do not have sufficient instance-type capacity in the AZ you requested</strong></td>
<td>One of the instance types is currently not available.</td>
<td>Switch to an instance type that supports higher capacity, or complete the <a href="#">Request quota increase</a> form on the <a href="#">Service Quotas</a> console to increase the Amazon EC2 quota for the instance type or Region. Quota increases are tied to the Region they were requested for.</td>
</tr>
<tr>
<td><strong>Instance ID did not stabilize</strong></td>
<td>You have exceeded your IOPS for the Region.</td>
<td>Request a quota increase by completing the <a href="#">Request quota increase</a> form on the <a href="#">Service Quotas</a> console.</td>
</tr>
<tr>
<td><strong>System Administrator password must contain at least 8 characters</strong></td>
<td>The master password contains $ or other special characters.</td>
<td>Check the password parameters before you re-launch the Quick Start. The passwords must be at least 8 characters, consisting of uppercase and lowercase letters and numbers. Follow the <a href="#">guidelines for complex passwords</a>, and avoid using special characters such as @ or $.</td>
</tr>
</tbody>
</table>

If failure is signaled or a wait condition or resource signal times out, review the configuration attempts in AWS Systems Manager State Manager.

1. On the Systems Manager console, choose **Compliance**.
   
   You will see a list of all State Manager compliance resources.
2. Look for one of the following compliance types and identify whether there are non-compliant resources for it:
   
   a. Single-server deployments will see Custom:QSSPSingle.
   
   b. Multiple-server deployments will see Custom:QSSPFarmFarm, Custom:QSSPFarmApp, and Custom:QSSPFarmWFE.

3. If there are non-compliant resources, choose the number with the red X next to it, and scroll down the page to see a list of instances that do not meet the compliance requirement. (This occurs when the server could not be configured correctly.)

4. Select the instance you want to investigate. A new browser tab will open.

5. To see which resources on the server were not configured, on the instance detail page, on the Configuration compliance tab, choose Configuration status in the filter box, and then choose NonCompliant.
Figure 29: Filtering to show non-compliant resources

6. To find detailed logs, on the **Associations** tab, choose the entry with the **AWS-ApplyDSCMofs** document name, and then choose **View Output**. Take note of the association ID here, which is needed in the next step.

Figure 30: Finding the association ID

7. On the Systems Manager console, choose **State Manager**.
8. In the list that is displayed, choose the association ID from step 6 to view its execution history.

9. On the **Execution History** tab, you can see every configuration run that has been attempted by this Automation document (State Manager configurations run every 30 minutes). Select the first item in the list, and then find the instance ID that was failing configuration.

10. Choose **Output**.
11. The console will show only the first 2,500 characters from the log, which is not likely to be useful. To view the logs, choose Amazon S3, select the RunPowerShell folder, and open the file stdout. The file will show you the full output of the configuration script, including any errors, which will help you troubleshoot.

For additional information, see Troubleshooting AWS CloudFormation on the AWS website.

Additional Resources

AWS services

- AWS CloudFormation
  https://aws.amazon.com/documentation/cloudformation/
- Amazon EC2
- Amazon VPC
  https://aws.amazon.com/documentation/vpc/
- AWS Systems Manager
  https://aws.amazon.com/systems-manager/

Microsoft SharePoint Server

- Configure SQL Server AlwaysOn Availability Groups for SharePoint Server
  https://docs.microsoft.com/en-us/sharepoint/administration/configure-an-alwayson-availability-group
- Failover Clustering and Always On Availability Groups (SQL Server)
  https://docs.microsoft.com/en-us/sql/database-engine/availability-
Deploying Microsoft software on AWS

- Microsoft on AWS
  https://aws.amazon.com/microsoft/

- Securing the Microsoft Platform on Amazon Web Services

- Microsoft Licensing Mobility
  https://aws.amazon.com/windows/mslicensemobility/

- MSDN on AWS
  https://aws.amazon.com/windows/msdn/

- Windows and .NET Developer Center on AWS
  https://aws.amazon.com/net/

Quick Start reference deployments

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

- Microsoft Active Directory on AWS
  https://docs.aws.amazon.com/quickstart/latest/active-directory-ds/

- Microsoft Remote Desktop Gateway on AWS
  https://docs.aws.amazon.com/quickstart/latest/rd-gateway/

- Microsoft SQL Server with WSFC on AWS
  https://docs.aws.amazon.com/quickstart/latest/sql/
Appendix A: Server Role Architecture

Single-Server Topology
A single-server deployment of SharePoint 2019 is ideal for development environments and for small-scale testing of SharePoint where multiple servers might be excessive.

![Single-server SharePoint topology](image)

In this topology, all SharePoint Service Apps and Instances will execute on the one server though, with no load balancing being required. The database server, however, runs on a different server than SharePoint. This is the simplest way to set up an environment for experimenting with SharePoint.

Multiple-Server Topology
When you deploy SharePoint to support a production workload that will have many users across the organization, a single-server deployment is not robust enough to account for the load or to maintain availability in the event of an incident. The multiple-server template uses four SharePoint servers in to different roles to distribute the load.
In this model, many of the back-end related activities of a SharePoint farm are handled by a second application server (or group of servers) so that the front-end servers can focus on serving end-user requests. Both sets of SharePoint servers directly communicate with the database server or Always On availability group.

To simplify the configuration of multiple-server farms, SharePoint includes the MinRole feature. MinRole provides pre-defined roles, each of which runs a group of services, which ensures that the SharePoint farm runs more smoothly and is easier to maintain.

The multiple-server topology in this Quick Start leverages the following roles in MinRole:

- Front end: Front-End with Distributed Cache
- Application servers: Application with Search

This allows an even distribution of services while adhering to the Microsoft best practices for farm topology. For more information on what MinRole is and how it works in detail, including documentation on what services run on each of these roles, see Overview of MinRole Server Roles in SharePoint Servers 2016 and 2019.

With two servers in both roles (to support high availability), the resulting architecture is shown in the following diagram.
Figure 35: The multiple-server topology

To ensure the highest levels of availability, the servers are put in to different Availability Zones. This helps ensure that in an event where an entire Availability Zone becomes impaired, the servers in the second Availability Zone will still be running.

Load Balancing
The multiple-server version of the Quick Start deploys two Network Load Balancers. The first load balancer is for the front-end servers, which run the main SharePoint web applications. The second is for the application servers, where the central administration website runs.

The Network Load Balancer directs traffic to a healthy node in the target group. The load balancers poll for the availability of a connection on port 80 every 30 seconds. If a node fails three consecutive checks, the load balancer will not direct any requests to it until it is again registered as healthy.

Database Tier
The database server role stores content and service data so that your SharePoint farm can utilize SQL Server in a number of ways. For small or medium-sized environments, you may be able to place all your databases on a single server. For larger-sized farms, you can spread your databases across multiple SQL Server instances or clusters of SQL Server instances.
We recommend using SQL Server Enterprise in your SharePoint deployment, as it meets the performance, high availability, and reliability requirements for an enterprise application.

Amazon Machine Images (AMIs) for SQL Server Express, SQL Server Web Edition, and SQL Server Standard are available for launch on AWS. To install SQL Server 2016 or 2017 Enterprise Edition on AWS, you can use Microsoft License Mobility through Software Assurance to bring your own license into the cloud.

In the Quick Start for Microsoft WSFC and SQL Server Always On on AWS, we provide an example of how you can deploy an Always On availability group to provide high availability for your databases. Our default SQL Server configuration uses the r4.2xlarge instance type, which is a memory-optimized instance with 8 vCPUs, 61 GiB of memory, and 1 x 100 GiB of SSD instance storage. Additionally, we provide highly performant and durable storage in the form of Elastic Block Store (Amazon EBS) volumes.

**Intranet SharePoint Server Farm on AWS**

All the architecture diagrams shown up to this point represent an isolated Microsoft SharePoint farm. For this scenario, users who are in the domain and network where SharePoint is running can access the content, but it is closed off to anyone outside of that domain and network. To allow access for a corporate network, the architecture needs to include private connectivity from the on-premises environment. Figure 36 shows a typical topology for an intranet SharePoint server farm running on the AWS Cloud.
As shown in Figure 36, we’ve added a VPN gateway to the VPC. To enable internal network connectivity to the VPC, we’ve created a VPN tunnel from the customer gateway (an IPsec-capable device) to the VPN gateway running in the VPC.

In addition, AWS offers the AWS Direct Connect service, which allows you to create a direct network connection from your data center into the AWS Cloud. In either case, once you have internal network connectivity into the VPC from your on-premises environment, you can simply provision internal Elastic Load Balancing to spread incoming traffic to front-end servers across each Availability Zone. Elastic Load Balancing will also provide high availability in the event of a server failure. If a web front-end server is unavailable, requests will be sent to one that is online.

Security

As with any enterprise application deployment, a Microsoft SharePoint Server farm on AWS should implement strict security controls. AWS provides a comprehensive set of security features that allow you to control the flow of traffic through your VPC, associated subnets, and ultimately to each Amazon EC2 instance. These features allow you to reduce the attack surface of your environment while providing both end-user access to SharePoint content.
and applications, and administrator access for securely managing the Windows Server infrastructure. These security features and approaches are covered in this section.

**Security Groups**
When launched, Amazon EC2 instances must be associated with at least one security group, which acts as a stateful firewall. You have complete control over the network traffic entering or leaving your security groups, and you can build granular rules that are scoped by protocol, port number, and source/destination IP address or subnet. By default, all traffic egressing a security group is permitted. Ingress traffic, on the other hand, must be configured to allow the appropriate traffic to reach your instances.

The [Securing the Microsoft Platform on Amazon Web Services](https://aws.amazon.com) whitepaper discusses the different methods for securing your AWS infrastructure in detail. Recommendations include providing isolation between application tiers using security groups. We recommend that you tightly control ingress traffic in order to reduce the attack surface of your Amazon EC2 instances.

**Network ACLs**
A network access control list (ACL) is a set of permissions that can be attached to any network subnet in a VPC to provide stateless filtering of traffic. Network ACLs can be used for inbound or outbound traffic, and provide an effective way to blacklist a CIDR block or individual IP addresses. These ACLs can contain ordered rules to allow or deny traffic based upon IP protocol, service port, or source or destination IP address. Figure 37 shows the default ACL configuration for a VPC subnet.

<table>
<thead>
<tr>
<th>Rule #</th>
<th>Port (Service)</th>
<th>Protocol</th>
<th>Source</th>
<th>Allow/Deny</th>
</tr>
</thead>
<tbody>
<tr>
<td>100</td>
<td>ALL</td>
<td>ALL</td>
<td>0.0.0.0/0</td>
<td>ALLOW</td>
</tr>
<tr>
<td>*</td>
<td>ALL</td>
<td>ALL</td>
<td>0.0.0.0/0</td>
<td>DENY</td>
</tr>
</tbody>
</table>

![Figure 37: Default network ACL configuration for a VPC subnet](https://aws.amazon.com)
You may choose to keep the default network ACL configuration or lock it down with more specific rules to restrict traffic between subnets at the network level. Typically, network ACLs will mirror your security group rules. One benefit of multiple layers of network security (security groups and network ACLs) is that each layer can be managed by a separate group in your organization. If a server administrator inadvertently exposes unnecessary network ports on a security group, a network administrator could supersede this configuration by blocking that traffic at the network ACL layer.

**Secure Extranet Publishing**

Some organizations may use SharePoint Server to host a publicly accessible extranet. In this scenario, you can add another layer of security by placing reverse proxy servers into your public subnet to provide additional security and threat management. In this configuration, the public subnet acts like the DMZ that you would typically use in a physical network environment. Web page requests from internet-based users would be sent to these reverse proxy servers, which would then establish a connection to your web front-end servers that are running in a private subnet.

Figure 38 shows an example of publishing SharePoint web front-end servers, located in a private subnet, through a reverse proxy server deployed into a public subnet.
A benefit of this architecture is that it provides the ability to pre-authenticate users at the perimeter of your network while shielding your internal SharePoint servers from the public internet. Several third-party appliances and applications can be used for this task. Microsoft’s Web Application Proxy role in Windows Server 2019 also provides support for publishing your SharePoint resources to the internet.

The AWS CloudFormation template provided by this Quick Start does not set up an environment for extranet publishing, but after the deployment, you may choose to add reverse proxy servers and configure the environment that’s illustrated in Figure 38.

**EC2 Instance Types**

Properly planning for capacity and sizing servers is a key aspect of every enterprise application deployment. As such, it is important that you choose the appropriate Amazon EC2 instance type for each server role in your Microsoft SharePoint deployment. Since each deployment is different, you will need to follow Microsoft’s detailed guidance on how to properly size your environment based on the number of users and workloads involved. As a starting point, consider the minimum requirements for each server role.

The following values are based on minimum requirements for all server roles operating in a three-tier farm.

<table>
<thead>
<tr>
<th>Role</th>
<th>Processor</th>
<th>RAM</th>
<th>Boot volume</th>
</tr>
</thead>
<tbody>
<tr>
<td>Web front-end server / front-end server</td>
<td>64-bit, 4 cores</td>
<td>12 GiB</td>
<td>80 GiB</td>
</tr>
<tr>
<td>Application server / batch processing / back end</td>
<td>64-bit, 4 cores</td>
<td>12 GiB</td>
<td>80 GiB</td>
</tr>
<tr>
<td>Database server (fewer than 1,000 users)</td>
<td>64-bit, 4 cores</td>
<td>8 GiB</td>
<td>80 GiB</td>
</tr>
<tr>
<td>Database server (between 1,000 and 10,000 users)</td>
<td>64-bit, 8 cores</td>
<td>16 GiB</td>
<td>80 GiB</td>
</tr>
</tbody>
</table>

The Quick Start uses the following instance types by default. These provide additional capacity over the absolute minimum requirements as a starting point.

<table>
<thead>
<tr>
<th>Role</th>
<th>EC2 instance type</th>
<th>Boot volume</th>
</tr>
</thead>
<tbody>
<tr>
<td>Web front-end server / front-end server</td>
<td>m4.xlarge (4 vCPU, 16 GiB memory)</td>
<td>100 GiB (EBS/GP2)</td>
</tr>
<tr>
<td>Application server / batch processing / back end</td>
<td>m4.xlarge (4 vCPU, 16 GiB memory)</td>
<td>100 GiB (EBS/GP2)</td>
</tr>
<tr>
<td>Database server</td>
<td>r4.2xlarge (8 vCPU, 61 GiB memory)</td>
<td>100 GiB (EBS/GP2)</td>
</tr>
</tbody>
</table>

Amazon Elastic Block Store (Amazon EBS) volumes are used as the boot volume for each instance. Notice that we use the EBS General Purpose (gp2) volume type. This is an SSD-
backed EBS volume that is used as the default boot volume type for all EC2 instances. These gp2 volumes provide a consistent baseline of 3 IOPS/GiB and are burstable up to 3,000 IOPS.

When you launch the AWS CloudFormation template in this guide, you'll be given the opportunity to adjust these instance types.
GitHub Repository

You can visit our GitHub repository to download the templates and scripts for this Quick Start, to post your feedback, and to share your customizations with others.

Document Revisions

<table>
<thead>
<tr>
<th>Date</th>
<th>Changes</th>
<th>In sections</th>
</tr>
</thead>
</table>
| December 2019 | • Updated to include multiple deployment options  
                 • Moved parameters from Appendix B to Deployment Steps | Changes throughout templates and guide |
| July 2019     | • Support for SharePoint Server 2019  
                 • Configuration management with Systems Manager State Manager  
                 • Topologies changed to be single server, and multiple-server based on MinRole  
                 • Updated instructions on adding databases to Always On availability groups  
                 • Added new prerequisites to support PowerShell DSC based deployment via Systems Manager  
                 • Removed need for public S3 bucket for distribution of binaries, added support for slipstream updates | Changes throughout templates and guide |
| March 2018     | Updated Active Directory to use the Windows Server 2016 AMI; added Quick Start configuration parameters                                      | Appendix B                       |
| May 2016       | Added:  
                 • Support for SharePoint Server 2016  
                 • Dedicated streamlined topology servers  
                 • Simpler installation media consumption process  
                 • Updated parameter names  
                 • New parameter labels and grouping  
                 • Support for latest AD stack and NAT gateways | Changes throughout templates and guide |
| April 2015     | Added information about testing high availability and automatic failover of SharePoint servers                                              | Steps 4-6                        |
| March 2015     | Optimized the underlying VPC design to support expansion and to reduce complexity                                                       | Architecture diagram and template updates |
| August 2014    | Initial publication                                                                                                                    | –                                |