User Guide

AWS Launch Wizard

AWS Launch Wizard: User Guide

Table of Contents

AWS Launch Wizard for Active Directory

AWS Launch Wizard for Active Directory is a service that applies AWS cloud application best practices to guide you through setting up a new Active Directory infrastructure, or adding domain controllers to an existing infrastructure either in the AWS Cloud or on premises. The deployment environment includes various resources such as a new or existing VPC, security groups, and AWS Identity and Access Management (IAM) roles. You can set up a new Active Directory infrastructure with domain controllers on Amazon EC2 instances, add domain controllers on Amazon EC2 instances to extend your existing Active Directory infrastructure, or use AWS Directory Service for Microsoft Active Directory for a managed service experience.

Launch Wizard reduces the time that it takes to set up an Active Directory infrastructure and deploy self-managed domain controllers to the cloud or on premises. You input your domain controller requirements, including number of nodes and connectivity, on the service console, and AWS Launch Wizard identifies the right AWS resources to deploy your self-managed domain controllers. AWS Launch Wizard provides an estimated cost of deployment, and gives you the ability to modify your resources and instantly view the updated cost assessment. When you approve, AWS Launch Wizard provisions and configures the selected resources in a few hours to create fully-functioning, production-ready domain controllers.

After you deploy your self-managed domain controllers, they are ready to use and can be accessed on the Amazon Elastic Compute Cloud (Amazon EC2) console.

Supported operating systems

AWS Launch Wizard for Active Directory supports the Windows Server 2022 operating system.

Features of AWS Launch Wizard

AWS Launch Wizard provides the following features:

- Simple application deployment
- AWS resource selection
- Cost estimation
- SNS notification
- Early input validation

- Application resource groups for easy discoverability

Simple application deployment

AWS Launch Wizard makes it efficient for you to deploy self-managed domain controllers and AWS Directory Service for Microsoft Active Directory on AWS. When you enter the domain controller requirements, AWS Launch Wizard deploys the necessary AWS resources for a production-ready environment. This means that you do not have to manage separate infrastructure pieces or spend time provisioning and configuring your domain controllers.

AWS resource selection

Launch Wizard considers the number of Active Directory users to determine the best instance type, EBS volumes, and other resources for your domain controllers. You can modify the recommended defaults.

Cost estimation

Launch Wizard provides a cost estimate for the complete deployment that is itemized for each individual resource being deployed. The estimated cost automatically updates each time you change a resource type configuration in the wizard. However, the provided estimates are only for general comparisons. They are based on on-Demand costs and actual costs may be lower.

SNS notification

You can provide an SNS topic that allows Launch Wizard to send you notifications and alerts about the status of a deployment.

Early input validation

You can take advantage of your existing infrastructure, such as VPC or security groups, with Launch Wizard. This may lead to deployment failures if your existing infrastructure does not meet certain deployment prerequisites. If these requirements are not met, the deployment will fail. If you are in a later stage of a deployment, this failure can take more than an hour to detect. To detect these types of issues early in the application deployment process, Launch Wizard's validation framework verifies key infrastructure specifications before provisioning. Verification takes approximately 15 minutes. If necessary, you can take appropriate actions to adjust your VPC configuration.

> **Note**
>
> Some validations, such as for Active Directory credentials, require Application Wizard to launch a t2.large EC2 instance in your account for a few minutes. After it runs the necessary validations, Launch Wizard terminates the instance.

Application resource groups for easy discoverability

Launch Wizard creates a resource group for all of the AWS resources created for your domain controllers. You can manage the resources through the Amazon EC2 console or with Systems Manager. When you access Systems Manager through Launch Wizard, the resources are automatically filtered for you based on your resource group.

Components

Self-managed domain controllers deployed with Launch Wizard include the following components:

- A **virtual private cloud (VPC)** configured with public and private subnets across two Availability Zones. A public subnet is a subnet whose traffic is routed to an internet gateway. If a subnet does not have a route to the internet gateway, then it is a private subnet. The VPC provides the network infrastructure for your domain controller environment.
- **Amazon EC2 instances** on which to provision your domain controllers.
- An **internet gateway** to provide access to the internet.
- In the public subnets, **network address translation (NAT) gateways** for outbound internet access. If you are deploying in your preexisting VPC, Launch Wizard uses the existing NAT gateway in your VPC. For more information about NAT gateways, see NAT Gateways.
- **Elastic IP addresses** associated with the NAT gateway and RDGW instances. For more information about Elastic IP addresses, see Elastic IP Addresses.
- **CloudFormation** templates and **PowerShell** configuration scripts to perform the domain controller configuration steps.
- **Security groups** to ensure the secure flow of traffic between the instances deployed in the VPC. For more information, see Security Groups for Your VPC.
- **AWS Secrets Manager** to protect secrets required to generate and store your Active Directory Administrator credentials.

- **Amazon CloudWatch Logs** to monitor, store, and access your log files produced by CloudFormation.
- Amazon Kinesis Agent for Microsoft Windows to gather, parse, transform, and stream logs, events, and metrics to Amazon CloudWatch Logs. For more information, see What Is Amazon Kinesis Agent for Microsoft Windows?

Requirements

Your account must be configured as specified in the following table to deploy self-managed domain controllers using Launch Wizard.

To add domain controllers to an existing infrastructure, you must create a VPC peering connection between the two VPCs for an existing Active Directory in AWS. If you are using an existing Active Directory on premises, you must use AWS Direct Connect. To ensure that instances in the VPCs can communicate with each other, you can use either Direct Connect or VPC Private Link. For more information about VPC connectivity, see VPN connections.

Resource	Minimum number of resources required for deployment
Virtual private clouds (VPCs)	1
VPC security groups	3
AWS Identity and Access Management (IAM) roles	2
General purpose EC2 instances	Existing VPC: 1
	New Active Directory infrastructure: 2
AWS Secrets Manager secrets	2

If you have an existing environment that uses these resources and you think that deploying domain controllers in this environment using Launch Wizard may exceed your default quotas, you can request service quota increases for these resources. For default quotas, see AWS service quotas.

Related services

The following services are used when you deploy self-managed domain controllers with AWS Launch Wizard:

- CloudFormation
- Amazon Simple Notification Service (SNS)
- Amazon CloudWatch Logs
- AWS Secrets Manager

CloudFormation

CloudFormation is a service for modeling and setting up your AWS resources, enabling you to spend more time focusing on your applications that run in AWS . You create a template that describes all of the AWS resources that you want to use (for example, EC2 instances), and CloudFormation provisions and configures those resources for you. With Launch Wizard, you don’t have to sift through CloudFormation templates to deploy your application. Instead, Launch Wizard combines infrastructure provisioning and configuration (with an CloudFormation template and PowerShell scripts) to provision a new Active Directory infrastructure or additional domain controllers in your account. For more information, see the *AWS CloudFormation User Guide*.

Amazon Simple Notification Service (SNS)

Amazon Simple Notification Service (Amazon SNS) is a highly available, durable, secure, fully managed publish/subscribe messaging service that provides topics for high-throughput, push-based, many-to-many messaging. Using Amazon SNS topics, your publisher systems can fan out messages to a large number of subscriber endpoints and send notifications to end users using mobile push, SMS, and email. You can use Amazon SNS topics for your Launch Wizard deployments to stay up to date on deployment progress. For more information, see the *Amazon Simple Notification Service Developer Guide*.

Amazon CloudWatch Logs

Amazon CloudWatch Logs enables you to centralize the logs from all of your systems, applications, and AWS services that you use, in a single, highly scalable service. You can then easily view them, search them for specific error codes or patterns, filter them based on specific fields, or archive them securely for future analysis. Amazon CloudWatch Logs enables you to see all of your logs, regardless of their source, as a single and consistent flow of events ordered by time, and you can query them and sort them based on other dimensions, group them by specific fields, create custom computations with a powerful query language, and visualize log data in dashboards. Launch Wizard streams provisioning logs from all of the AWS log sources that you can view on the CloudWatch console.

AWS Secrets Manager

With AWS Secrets Manager you can replace hard-coded credentials in your code, including passwords, with an API call to Secrets Manager to programmatically retrieve the secret. This helps ensure the secret can't be compromised by someone examining your code. Also, you can configure Secrets Manager to automatically rotate the secret for you according to a specified schedule. Launch Wizard uses Secrets Manager to join your domain controllers to Active Directory and promote them.

How AWS Launch Wizard Active Directory works

AWS Launch Wizard provides a complete solution to provision self-managed domain controllers on Amazon EC2 instances, or AWS Directory Service for Microsoft Active Directory, in the AWS Cloud. You select **Microsoft Active Directory** in the wizard and provide the specifications, such as the required number of vCPUs or memory. Based on the infrastructure requirements that you enter, Launch Wizard automatically provisions the appropriate AWS resources in the cloud. For example, Launch Wizard recommends an appropriate instance type from the amount of vCPUs that you specify, then deploys and configures the instances.

Launch Wizard provides an estimated cost of deployment. You can modify your resources and instantly view an updated cost assessment. Once you approve, Launch Wizard validates the inputs and flags inconsistencies. After you resolve the inconsistencies, Launch Wizard provisions the resources and configures them. The result is a ready-to-use Active Directory infrastructure and domain controllers.

AWS Launch Wizard performs the following tasks to provision Active Directory domain controllers.

- Sets up the VPC, including private and public subnets in two Availability Zones.*
- Configures two NAT gateways in the public subnets.*
- Configures private and public routes.*
- Enables ingress traffic into the VPC for administrative access to Remote Desktop Gateway, if specified.
- Uses Secrets Manager to store Domain Administrator credentials.

- Configures security groups and rules for traffic between instances.
- Sets up and configures Active Directory sites and subnets.
- Sets up and deploys Active Directory Certificate Services with a new Active Directory infrastructure.

* If you deploy Launch Wizard into an existing VPC, the tasks in this list marked by asterisks are skipped.

Topics

- Deployment path
- Implementation details
- Domain controller launch limits
- AWS Regions

Deployment path

Launch Wizard supports the following deployment path for provisioning self-managed domain controllers or AWS Managed Microsoft AD.

Deploy and manage your own domain controllers on Amazon EC2 instances

Launch Wizard builds the AWS Cloud infrastructure, and sets up and configures Active Directory Domain Services (AD DS) and Active Directory-integrated DNS on the AWS Cloud. For self-managed domain controllers, you handle all AD DS maintenance and monitoring tasks. You can deploy the domain controllers or AWS Managed Microsoft AD into a new or existing VPC infrastructure.

Implementation details

This section describes how Launch Wizard implements an Active Directory Domain Services (AD DS) deployment in the AWS Cloud. It includes details about how to use Amazon Virtual Private Cloud (Amazon VPC) to define your networks in the cloud, and information about domain controller placement, Active Directory Sites and Services configuration, and how DNS and DHCP work in a VPC.

Topics

- VPC

- Security groups
- Remote Desktop Gateway
- Active Directory
- Self-managed domain controller architecture

VPC

You can define a virtual network topology that closely resembles a traditional on-premises network using Amazon VPC. A VPC can span multiple Availability Zones place independent infrastructure in physically separate locations. A multi-Availability Zone deployment results in high availability and fault tolerance. Launch Wizard provisions domain controllers in two Availability Zones to provide highly available, low latency access to AD DS services in the AWS Cloud.

Launch Wizard can build a new VPC for the deployment, or deploy into an existing VPC. To accommodate highly available AD DS in the AWS Cloud, Launch Wizard builds (or requires, in the case of existing VPCs) a base Amazon VPC configuration that complies with the following AWS best practices:

- Domain controllers must be placed in a minimum of two Availability Zones to provide high availability.
- Domain controllers and other non-internet facing servers must be placed in private subnets.
- Launched instances require internet access to connect to the CloudFormation endpoint during the bootstrapping process. To support this configuration, public subnets are used to host NAT gateways for outbound internet access. Remote Desktop Gateways are also deployed into the public subnets for remote administration. Other components such as reverse proxy servers can be placed into these public subnets, if needed.

This VPC architecture uses two Availability Zones, each with its own distinct public and private subnets. We recommend that you leave plenty of unallocated address space to support the growth of your environment over time and to reduce the complexity of your VPC subnet design. Launch Wizard uses a default VPC configuration that provides plenty of address space by using the minimum number of private and public subnets. By default, Launch Wizard uses the following CIDR ranges.

VPC	10.0.0.0/16	
Private subnets A	10.0.0.0/20	
	Availability Zone 1	10.0.0.0/20
	Availability Zone 2	10.0.16.0/20
	Availability Zone 3	10.0.32.0/20
Public subnets	Availability Zone 1	10.0.128.0/20

In addition, Launch Wizard provisions spare capacity for additional subnets to support your environment as it grows or changes over time. If you have sensitive workloads that must be completely isolated from the internet, you can create new VPC subnets using these optional address spaces.

Security groups

Amazon EC2 instances must be associated with a security group, which acts as a stateful firewall. You control the network traffic entering or leaving the security group, and you can create rules that are defined by protocol, port number, and source/destination IP address, or other security groups. By default, all egress traffic from a security group is permitted. However, ingress traffic must be configured to allow the desired traffic to reach your instances.

The Securing the Microsoft Platform on Amazon Web Services whitepaper explains the different methods for securing your AWS infrastructure. Recommendations include providing isolation between application tiers by using security groups. We recommend that you tightly control ingress traffic in order to reduce the attack surface of your Amazon EC2 instances.

If you are deploying and managing your own AD DS installation, domain controllers and member servers will require several security group rules to allow traffic for services. These rules include AD DS replication, user authentication, Windows Time services, and Distributed File System (DFS). You should also consider restricting these rules to specific IP subnets that are used within your VPC.

For a detailed list of port mappings used by CloudFormation, see the Security best practices in this guide.

For a complete list of ports, see Active Directory and Active Directory Domain Services Port Requirements in the Microsoft TechNet Library and How to configure a firewall for Active Directory domains and trusts for forest trusts. For guidance on implementing rules, see Adding Rules to a Security Group in the *Amazon EC2 User Guide*.

Remote Desktop Gateway

When you design your architecture for highly available AD DS, you should also design for highly available and secure remote access. Launch Wizard optionally allows for deployment of a Remote Desktop (RD) Gateway server to manage your AD DS instances.

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 Amazon EC2 instances, without the need for a virtual private network (VPN) connection. This configuration reduces the attack surface of your Windows-based Amazon EC2 instances, while providing a remote administration solution for administrators.

> ⚠ **Important**
>
> Never open up RDP to the entire internet even temporarily or for testing purposes. Always restrict ports and source traffic to the minimum necessary to support the functionality of the application.

Active Directory

This section provides information about key design considerations specific to a Launch Wizard deployment of Active Directory Domain Services (AD DS) domain controllers on AWS.

Active Directory deployment topics

- Highly available directory domain services
- Active Directory DNS and DHCP inside the VPC
- DNS settings on Windows Servers instances
- Active Directory Certificate Services

Highly available directory domain services

Launch Wizard deploys two domain controllers in your AWS environment in two Availability Zones. This design provides fault tolerance and prevents a single domain controller failure from affecting the availability of the AD DS.

To strengthen the high availability of your architecture and help mitigate the impact of a possible disaster, each domain controller deployed by Launch Wizard is a global catalog server and an Active Directory DNS server.

When you choose to deploy self-managed domain controllers to the AWS Cloud, Launch Wizard automatically builds out an Active Directory Sites and Services configuration that supports a highly available AD DS architecture.

For information about creating sites, adding global catalog servers, and creating and managing site links, see the Microsoft Active Directory Sites and Services documentation.

Active Directory DNS and DHCP inside the VPC

Dynamic Host Configuration Protocol (DHCP) services are provided by default for your instances within a VPC. DHCP scopes do not need to be managed; they are created for the VPC subnets you define when you deploy your solution. These DHCP services cannot be disabled, so you must use them rather than deploying your own DHCP server.

The VPC also provides an internal DNS server. This DNS provides instances with basic name resolution services for internet access and is crucial for access to AWS service endpoints, such as CloudFormation and Amazon S3 during bootstrapping.

Amazon-provided DNS server settings will be assigned to instances launched into the VPC based on a DHCP options set. DHCP options sets are used within an Amazon VPC to define scope options, such as the domain name or the name servers that should be handed to your instances via DHCP. Amazon-provided DNS is used only for public DNS resolution.

Because Amazon-provided DNS cannot be used to provide name resolution services for Active Directory, you must ensure that domain-joined Windows instances are configured to use Active Directory DNS.

Launch Wizard statically assigns Active Directory DNS server addresses on Windows instances. You can alternatively specify them using a custom DHCP options set. This allows you to assign your Active Directory DNS suffix and DNS server IP addresses as the name servers within the VPC through DHCP.

Note

The IP addresses in the `domain-name-servers` field are always returned in the same order. If the first DNS server in the list fails, instances should fall back to the second IP and continue to resolve host names successfully. However, during normal operations, the first DNS server listed will always handle DNS requests. If you want to ensure that DNS queries are distributed evenly across multiple servers, you should consider statically configuring DNS server settings on your instances.

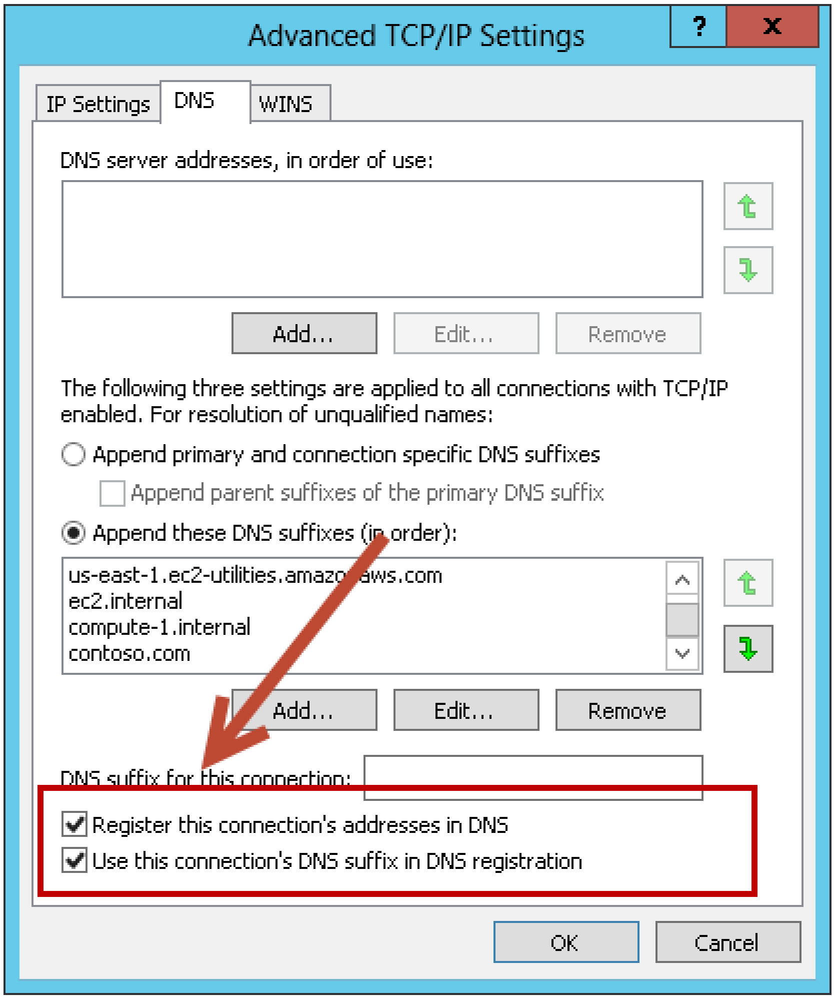
For more information about creating a custom DHCP options set and associating it with your VPC, see Working with DHCP Options Sets in the *Amazon VPC User Guide*.

Note

If you choose to deploy self-managed domain controllers in the AWS Cloud, Launch Wizard adds the DNS suffix for your domain to the DNS suffixes list. The DNS settings on the local server point to the IP address of the first domain controller for all of the domain controllers in the infrastructure.

DNS settings on Windows Servers instances

To ensure that domain-joined Windows instances automatically register host (A) and reverse lookup (PTR) records with Active Directory-integrated DNS, set the properties of the network connection as shown in the following image.



Advanced TCP/IP Settings

DNS server addresses, in order of use:

The following three settings are applied to all connections with TCP/IP enabled. For resolution of unqualified names:

○ Append primary and connection specific DNS suffixes

□ Append parent suffixes of the primary DNS suffix

● Append these DNS suffixes (in order):

us-east-1.ec2-utilities.amazonaws.com
ec2.internal
compute-1.internal
contoso.com

DNS suffix for this connection:

☑ Register this connection's addresses in DNS

☑ Use this connection's DNS suffix in DNS registration

The default configuration for a network connection is set to automatically register the connections address in DNS. In other words, the **Register this connection's addresses in DNS** option is selected for you automatically. This takes care of host (A) record dynamic registration. However, if you do not also select the second option, **Use this connection's DNS suffix in DNS registration**, dynamic registration of PTR records will not occur.

If you have a small number of instances in the VPC, you may choose to manually configure the network connection. For larger fleets, you can push this setting out to all of your Windows instances by using Active Directory Group Policy. For instructions about how to do this, see IPv4 and IPv6 Advanced DNS Tab in the Microsoft TechNet Library.

Active Directory Certificate Services

Launch Wizard sets up and deploys Active Directory Certificate Services (AD CS) with a new Active Directory infrastructure to issue and manage digital certificates in systems that use public key technologies. For more information about AD CS, see the Microsoft documentation.

Self-managed domain controller architecture

The Launch Wizard self-managed domain controller deployment sets up the following architecture.

- Domain controllers are deployed into two private VPC subnets in separate Availability Zones, which makes AD DS highly available.
- NAT gateways are deployed to public subnets, providing outbound internet access for instances in private subnets.
- Remote Desktop gateways are deployed in an Auto Scaling group in one Availability Zone to allow access to the domain controllers.

Launch Wizard deploys AWS resources, including a Systems Manager Automation document. When the second node is deployed, it initiates running the Automation document through Amazon EC2 user data. The automation workflow deploys the required components, finalizes the configuration to create a new AD forest, and promotes instances in two Availability Zones to Active Directory domain controllers.

To view architectural diagrams showing best practices for setting up an AD DS environment, see Active Directory Domain Services on AWS.

Domain controller launch limits

A single Launch Wizard deployment for Active Directory launches two domain controllers per each AWS Region. If you want to add more domain controllers, you can create additional Launch Wizard for Active Directory deployments to add them to the same Active Directory infrastructure. For more information, see Extend on-premises Active Directory to an existing VPC.

AWS Regions

Launch Wizard uses various AWS services during the provisioning of the application's environment. Not every workload is supported in all AWS Regions. For a current list of Regions where the workload can be provisioned, see AWS Launch Wizard workload availability.

Get started with AWS Launch Wizard for Active Directory

This section contains information to set up your environment for Launch Wizard to deploy domain controllers.

Topics

Accessing AWS Launch Wizard Active Directory

You can launch AWS Launch Wizard from the AWS Launch Wizard console located at https://console.aws.amazon.com/launchwizard.

Specialized knowledge

This deployment requires a moderate level of familiarity with AWS services. If you're new to AWS, see Getting Started Resource Center and AWS Training and Certification. These sites provide materials for learning how to design, deploy, and operate your infrastructure and applications on the AWS Cloud.

This Launch Wizard deployment assumes familiarity with Active Directory concepts and usage.

Amazon Web Services account

Sign up for an AWS account

If you do not have an AWS account, complete the following steps to create one.

To sign up for an AWS account

1. Open https://portal.aws.amazon.com/billing/signup.
2. Follow the online instructions.

   Part of the sign-up procedure involves receiving a phone call or text message and entering a verification code on the phone keypad.

When you sign up for an AWS account, an *AWS account root user* is created. The root user has access to all AWS services and resources in the account. As a security best practice, assign administrative access to a user, and use only the root user to perform tasks that require root user access.

AWS sends you a confirmation email after the sign-up process is complete. At any time, you can view your current account activity and manage your account by going to https://aws.amazon.com/ and choosing **My Account**.

Create a user with administrative access

After you sign up for an AWS account, secure your AWS account root user, enable AWS IAM Identity Center, and create an administrative user so that you don't use the root user for everyday tasks.

Secure your AWS account root user

1. Sign in to the AWS Management Console as the account owner by choosing **Root user** and entering your AWS account email address. On the next page, enter your password.

   For help signing in by using root user, see Signing in as the root user in the *AWS Sign-In User Guide*.

2. Turn on multi-factor authentication (MFA) for your root user.

   For instructions, see Enable a virtual MFA device for your AWS account root user (console) in the *IAM User Guide*.

Create a user with administrative access

1. Enable IAM Identity Center.

   For instructions, see Enabling AWS IAM Identity Center in the *AWS IAM Identity Center User Guide*.

2. In IAM Identity Center, grant administrative access to a user.

   For a tutorial about using the IAM Identity Center directory as your identity source, see Configure user access with the default IAM Identity Center directory in the *AWS IAM Identity Center User Guide*.

Sign in as the user with administrative access

- To sign in with your IAM Identity Center user, use the sign-in URL that was sent to your email address when you created the IAM Identity Center user.

  For help signing in using an IAM Identity Center user, see Signing in to the AWS access portal in the *AWS Sign-In User Guide*.

Assign access to additional users

1. In IAM Identity Center, create a permission set that follows the best practice of applying least-privilege permissions.

   For instructions, see Create a permission set in the *AWS IAM Identity Center User Guide*.
2. Assign users to a group, and then assign single sign-on access to the group.

   For instructions, see Add groups in the *AWS IAM Identity Center User Guide*.

Technical requirements

Before you start the Launch Wizard deployment, review the following information and make sure that your account is properly configured. Otherwise, deployment might fail.

Service Quotas

If necessary, request service quota increases for the resources deployed by Launch Wizard. You might need to request increases if your existing deployment currently uses these resources and if this Launch Wizard deployment could result in exceeding the default quotas. The Service Quotas console displays your usage and quotas for some aspects of some services. For more information, see What is Service Quotas? and AWS service quotas.

IAM permissions

Before deploying the Launch Wizard application, you must sign in to the AWS Management Console with IAM permissions for the resources that the templates deploy. The *AdministratorAccess* managed policy within IAM provides sufficient permissions, although your organization may choose to use a custom policy with more restrictions. For more information, see AWS managed policies for job functions.

Active Directory deployment options

This section contains information on what configuration is performed for deployment of domain controllers into a new or existing VPC. You can deploy a new Active Directory infrastructure on Amazon EC2, deploy a new AWS Managed Microsoft AD, or extend an existing on-premises Active Directory into the AWS Cloud.

Active Directory configurations

When you use Launch Wizard to deploy Active Directory, the following key operations are performed. These operations result in the creation of new records or entries in Active Directory.

- When you create a new Active Directory domain, Launch Wizard creates two new Amazon EC2 instances and promotes the servers to domain controllers in your domain.
- When you extend an existing Active Directory domain, Launch Wizard creates two new Amazon EC2 instances and optionally joins them to the domain.
- When you create an AWS Managed Microsoft AD, Launch Wizard deploys the managed directory.
- All deployment types create ingress and egress rules to communicate with your domain controllers.

On-premises Active Directory through Direct Connect

If you are deploying domain controllers to extend an on-premises Active Directory into an existing VPC, ensure that the following prerequisites are in place.

- Make sure that you have connectivity between your AWS account and your on-premises network. You can establish a dedicated network connection from your on-premises network to your AWS account with Direct Connect. For more information, see the AWS Direct Connect documentation.
- The domain functional level of your Active Directory domain controller must be Windows Server 2012 or later.
- The IP addresses of your DNS server must be either in the same VPC CIDR range as the one in which your Launch Wizard domain controllers will be created, or in the private IP address range.
- The firewall on the Active Directory domain controllers should allow the connections from the VPC from which you will create the Launch Wizard deployment. At a minimum, your configuration should include the ports mentioned in How to configure a firewall for Active Directory domains and trusts.

You can optionally perform the following step.

- Establish DNS resolution across your environments. For options on how to set this up, see How to Set Up DNS Resolution Between On-Premises Networks and AWS using Directory Service and Amazon Route 53 or How to Set Up DNS Resolution Between On-Premises Networks and AWS Using Directory Service and Microsoft Active Directory.

Deploy Active Directory to a new VPC (Console)

You can use AWS Launch Wizard to deploy Active Directory to a new virtual private cloud (VPC) as a self-managed directory on Amazon Elastic Compute Cloud instances, extend your existing active directory into a new VPC with Amazon EC2 instances, or create an AWS Directory Service for Microsoft Active Directory directory in a new VPC.

Contents

Deploy self-managed Active Directory to a new VPC

The following steps guide you through an Active Directory deployment with AWS Launch Wizard after you have launched it from the console for a new VPC.

1. On the Launch Wizard Console's landing page, use the **Choose application** button. This opens the Choose application wizard where you are prompted to select the type of application that you want to deploy.
2. Select **Active Directory**, select **Deploy self-managed AD into a new VPC**, then select **Create deployment.**
3. Review and acknowledge the required IAM permissions are met before proceeding. For more information, see Identity and Access Management for AWS Launch Wizard.
4. On the **Configure application settings** page, you are prompted to enter the specifications for the new deployment. The following tabs provide information about the specification fields of the deployment model.

General settings

- **Deployment name**. Enter a unique application name for your deployment.
- **Amazon Simple Notification Service (Amazon SNS) topic ARN — optional**. Specify an Amazon SNS topic where Launch Wizard can send notifications and alerts. For more information, see the Amazon Simple Notification Service Developer Guide.
- **Deactivate rollback on failed deployment**. By default, if a deployment fails, your provisioned resources will be deleted. You can enable this setting during deployment to prevent this behavior.
- **Tags - optional**. Enter a key and value to assign metadata to your deployment. For help with tagging, see Tagging Your Amazon EC2 Resources.

Network configuration

Parameter label (name)	Default value	Description
Availability zones (AvailabilityZones)	***Requires input***	List of Availability Zones (AZs) to use for the subnets in the VPC.
Number of availability zones (NumberOfAZs)	2	Number of Availability Zones to use in the VPC. This must match your selections in the list of Availability Zones parameter.
VPC CIDR (VPCCIDR)	10.0.0.0/16	CIDR Block for the VPC.
Create a DHCP options set (DHCPOptionSet)	Yes	Creates and associates a new DHCP Options Set to your VPC.
Private subnet 1 CIDR (PrivateSubnet1CIDR)	10.0.0.0/19	CIDR block for private subnet 1 located in Availability Zone 1.

Parameter label (name)	Default value	Description
Private subnet 2 CIDR (PrivateSubnet2CIDR)	10.0.32.0/19	CIDR block for private subnet 2 located in Availability Zone 2.
(Optional) Private subnet 3 CIDR (PrivateSubnet3CIDR)	***Blank string***	CIDR block for private subnet 3 located in Availability Zone 3.
Public subnet 1 CIDR (PublicSubnet1CIDR)	10.0.128.0/20	CIDR Block for the public subnet 1 located in Availability Zone 1.
Public subnet 2 CIDR (PublicSubnet2CIDR)	10.0.144.0/20	CIDR Block for the public subnet 2 located in Availability Zone 2.
(Optional) Public subnet 3 CIDR (PublicSubnet3CIDR)	***Blank string***	CIDR Block for the public subnet 3 located in Availability Zone 3.

Amazon EC2 configuration

Parameter label (name)	Default value	Description
Domain controller 1 NetBIOS name (ADServer 1NetBIOSName)	DC1	NetBIOS name of the first Active Directory domain controller (between 1-15 characters).
Domain controller 1 private IP address (ADServer 1PrivateIP)	10.0.0.10	Fixed private IP for the first Active Directory domain controller located in Availability Zone 1.

Parameter label (name)	Default value	Description
Domain controller 2 NetBIOS name (ADServer 2NetBIOSName)	DC2	NetBIOS name of the second Active Directory domain controller (between 1-15 characters).
Domain controller 2 private IP address (ADServer 2PrivateIP)	10.0.32.10	Fixed private IP for the second Active Directory domain controller located in Availability Zone 2.
SYSVOL and NTDS and data drive size (DataDriv eSizeGiB)	10	Size of SYSVOL and NTDS data drive in GiB.
Key pair name (KeyPairN ame)	***Requires input***	Public/private key pairs allow you to securely connect to your instance after it launches.

Microsoft Active Directory Domain Services configuration

Parameter label (name)	Default value	Description
Domain admin user name (DomainAdminUser)	Admin	User name for the account that will be added as a Domain Administrator. This is separate from the default "Administrator" account.
Domain admin password (DomainAdminPassword)	***Requires input***	Password for the previously named account. Must be at least 8 characters containin g letters, numbers and symbols.

Parameter label (name)	Default value	Description
Domain DNS name (DomainDNSName)	example.com	Fully qualified domain name (FQDN) of the forest root domain. For example, example.com.
Domain NetBIOS name (DomainNetBIOSName)	example	NetBIOS name of the domain (between 1 to 15 characters) for users of earlier versions of Windows. For example, EXAMPLE.
Create Default OUs (CreateDefaultOUs)	No	Domain Elevated Accounts, Domain Users, Domain Computers, Domain Servers, Domain Service Accounts, and Domain Groups OUs and set the default users and computers containers to Domain Users and Domain Computers.
Set new tombstone lifetime (TombstoneLifetime)	180	The number of days before a deleted object, not recoverable by Active Directory natively, is permanently removed.
Set new deleted objects lifetime (DeletedObjectLifetime)	180	The number of days a deleted Active Directory object is restorable from the Active Directory Recycle Bin, with no loss of information.

Microsoft Active Directory Certificate Services configuration

Parameter label (name)	Default value	Description
Certificate authority (CA) deployment type (PKI)	No	Deploy two-tier (Offline Root with Subordinate Enterprise CA) or one-tier (Enterprise Root CA) PKI Infrastructure.
CA data drive size (CaDataDriveSizeGiB)	2	Size of the data drive in GiB for the CA instance(s).
CA AMI ID (CaAmi)	/aws/service/ami-windows-latest/Windows_Server-2022-English-Full-Base	The Systems Manager Parameter Store value used to provision the enterprise root CA.
Offline root CA NetBIOS name (Only Used For two-tier PKI) (OrCaServerNetBIOSName)	ORCA1	NetBIOS name of the offline root CA server, used only for two-tier PKI (between 1-15 characters).
Enterprise root or subordinate CA NetBIOS name (EntCaServerNetBIOSName)	ENTCA1	NetBIOS name of the enterprise root (one-tier) or subordinate CA server (two-tier). The value must be 1-15 characters.
CA key length (CaKeyLength)	2048	CA(s) cryptographic provider key length.
CA hash algorithm (CaHashAlgorithm)	SHA256	CA(s) hash algorithm for signing certificates.

Parameter label (name)	Default value	Description
Offline root CA certificate validity period (only used for two-tier PKI) (OrCaValidityPeriodUnits)	10	Validity period in years for the offline root CA certificate (used only for two-tier PKI).
Enterprise root or subordinate CA certificate validity period (CaValidityPeriodUnits)	5	Validity period in years for the enterprise root or subordinate CA certificate.
Use Amazon S3 for CA CRL location (UseS3ForCRL)	No	Store CA CRL(s) in an S3 bucket.
CA CRL Amazon S3 bucket name (S3CRLBucketName)	examplebucket	S3 bucket name for CA CRL(s) storage. Bucket name can include numbers, lowercase letters, uppercase letters, and hyphens (-). It cannot start or end with a hyphen (-).

Microsoft Remote Desktop Gateway configuration

Parameter label (name)	Default value	Description
Number of RDGW hosts (NumberOfRDGWHosts)	1	Enter the number of Remote Desktop Gateway hosts to create.
Allowed Remote Desktop Gateway external access CIDR (RDGWCIDR)	***Requires input***	Allowed CIDR block for external access to the Remote Desktop Gateways.

5. When you are satisfied with your application settings, choose **Next**. If you don't want to complete the configuration, choose **Cancel**. When you choose **Cancel**, all of the selections

on the specification page are lost and you are returned to the landing page. To return to the previous screen, choose **Previous**.

6. On the **Configure infrastructure settings** page, you are prompted to define the infrastructure settings for the new deployment. The following tab provides information about the input fields.

   Storage and compute

   You can choose to select your instances, or to use AWS recommended resources. If you choose to use AWS recommended resources, you have the option of defining your performance needs. If you don't select either option, default values are assigned. Launch Wizard will display the estimated charges incurred to deploy the application based on suggested infrastructure and also based on static values.

   - **Based on infrastructure suggestion**. Launch Wizard displays the suggested resources for the deployment. You can specify your performance requirements of the resources to update the recommendation.
     - **Number of instance cores**. Choose the number of CPU cores for your infrastructure. The default value assigned is 4.
     - **Network performance**. Choose your preferred network performance in Gbps.
     - **Memory (GB)**. Choose the amount of RAM that you want to attach to your EC2 instances. The default value assigned is 4 GB.
     - **Recommended resources**. Launch Wizard displays the system-recommended resources based on your infrastructure selections. If you want to change the recommended resources, select different infrastructure settings.
     - **Estimated on-demand cost to deploy additional resources**. Launch Wizard displays the estimated charges incurred to deploy the resources.

   - **Based on static values**. You can specify specific instance types for the resources used in your deployment. If you don't select either option, default values are assigned.
     - **Instance type**. You can choose your instance type from the dropdown list, or you can use AWS recommended resources.
     - **Estimated on-demand cost to deploy additional resources**. Launch Wizard displays the estimated charges incurred to deploy the resources.

7. When you are satisfied with your infrastructure settings, select **Next**. If you don't want to complete the configuration, select **Cancel**. When you select **Cancel**, all of the selections on the specification page are lost and you are returned to the landing page. To go to the previous screen, select **Previous**.
8. On the **Review and deploy** page, review your configuration details. If you want to make changes, select **Previous**. To stop, select **Cancel**. When you select **Cancel**, all of the selections on the specification page are lost and you are returned to the landing page. When you choose **Deploy**, you agree to the terms of the **Acknowledgment**. Launch Wizard validates the inputs and notifies you if you need to address any issues.
9. When validation is complete, Launch Wizard deploys your AWS resources and configures your application. Launch Wizard provides you with status updates about the progress of the deployment on the **Deployments** page. From the **Deployments** page, you can view the list of current and previous deployments.
10. When your deployment is ready, a notification informs you that your application is successfully deployed. If you have set up an Amazon SNS notification, you are also alerted through Amazon SNS. You can manage and access all of the resources related to your application by selecting the deployment, and then selecting **Manage** from the **Actions** dropdown list.
11. When the application is deployed, you can access your EC2 instances through the Amazon EC2 console.

Extend an existing Active Directory to a new VPC

The following steps guide you through an Active Directory deployment with AWS Launch Wizard after you have launched it from the console for a new VPC.

1. On the Launch Wizard Console's landing page, use the **Choose application** button. This opens the Choose application wizard where you are prompted to select the type of application that you want to deploy.
2. Select **Active Directory**, select **Extend on-premises AD into a new VPC**, then select **Create deployment.**
3. Review and acknowledge the required IAM permissions are met before proceeding. For more information, see Identity and Access Management for AWS Launch Wizard.
4. On the **Configure application settings** page, you are prompted to enter the specifications for the new deployment. The following tabs provide information about the specification fields of the deployment model.

General settings

- **Deployment name**. Enter a unique application name for your deployment.
- **Amazon Simple Notification Service (Amazon SNS) topic ARN — optional**. Specify an Amazon SNS topic where Launch Wizard can send notifications and alerts. For more information, see the Amazon Simple Notification Service Developer Guide.
- **Deactivate rollback on failed deployment**. By default, if a deployment fails, your provisioned resources will be deleted. You can enable this setting during deployment to prevent this behavior.
- **Tags - optional**. Enter a key and value to assign metadata to your deployment. For help with tagging, see Tagging Your Amazon EC2 Resources.

Network configuration

Parameter label (name)	Default value	Description
Availability zones (AvailabilityZones)	***Requires input***	List of Availability Zones (AZs) to use for the subnets in the VPC.
Number of Availability Zones (NumberOfAZs)	2	Number of Availability Zones to use in the VPC. This must match your selections in the list of Availability Zones parameter.
VPC CIDR (VPCCIDR)	10.0.0.0/16	CIDR Block for the VPC.
Private subnet 1 CIDR (PrivateSubnet1CIDR)	10.0.0.0/19	CIDR block for private subnet 1 located in Availability Zone 1.
Private subnet 2 CIDR (PrivateSubnet2CIDR)	10.0.32.0/19	CIDR block for private subnet 2 located in Availability Zone 2.

Parameter label (name)	Default value	Description
(Optional) Private subnet 3 CIDR (PrivateSubnet3CIDR)	***Blank string***	CIDR block for private subnet 3 located in Availability Zone 3.
Public subnet 1 CIDR (PublicSubnet1CIDR)	10.0.128.0/20	CIDR Block for the public subnet 1 located in Availability Zone 1.
Public subnet 2 CIDR (PublicSubnet2CIDR)	10.0.144.0/20	CIDR Block for the public subnet 2 located in Availability Zone 2.
(Optional) Public subnet 3 CIDR (PublicSubnet3CIDR)	***Blank string***	CIDR Block for the public subnet 3 located in Availability Zone 3.

Amazon EC2 configuration

Parameter label (name)	Default value	Description
Domain controller 1 NetBIOS name (ADServer 1NetBIOSName)	DC3	NetBIOS name of the first additional Active Directory domain controlle r (between 1-15 character s).
Domain controller 1 private IP address (ADServer 1PrivateIP)	10.0.0.11	Fixed private IP for the first additional Active Directory domain controller located in subnet 1.

Parameter label (name)	Default value	Description
Domain controller 2 NetBIOS name (ADServer 2NetBIOSName)	DC4	NetBIOS name of the second additional Active Directory domain controller (between 1-15 characters).
Domain controller 2 private IP address (ADServer 2PrivateIP)	10.0.32.11	Fixed private IP for the second additional Active Directory domain controller located in subnet 2.
SYSVOL and NTDS and data drive size (DataDriveSizeGiB)	10	Size of SYSVOL and NTDS data drive in GiB.
Key pair name (KeyPairName)	***Requires input***	Public/private key pairs allow you to securely connect to your instance after it launches.

Microsoft Active Directory Domain Services configuration

Parameter label (name)	Default value	Description
DNS Server 1 IP address (ExistingDomainController1IP)	10.0.0.10	The IP address of the first DNS server that can resolve the domain. You must have connectivity from the VPC to the DNS server.

Parameter label (name)	Default value	Description
DNS Server 2 IP address (ExistingDomainController2IP)	10.0.32.10	The IP address of the second DNS server that can resolve the domain. You must have connectivity from the VPC to the DNS server.
Domain DNS name (DomainDNSName)	example.com	Fully qualified domain name (FQDN) of the domain you would like to join and promote to. For example, example.com.
Domain NetBIOS name (DomainNetBIOSName)	example	NetBIOS name of the domain (between 1 to 15 characters) you would like to join and promote to for users of earlier versions of Windows. For example, EXAMPLE.

Microsoft Remote Desktop Gateway configuration

Parameter label (name)	Default value	Description
Local administrator user name (AdminUser)	StackAdmin	User name for the new local administrator account This is separate from the default "Administrator" account.

Parameter label (name)	Default value	Description
Local administrator password (AdminPassword)	***Requires input***	Password for the new local administrator account containing letters, numbers and symbols.
Number of RDGW hosts (NumberOfRDGWHosts)	1	Enter the number of Remote Desktop Gateway hosts to create.
Allowed Remote Desktop Gateway external access CIDR (RDGWCIDR)	***Requires input***	Allowed CIDR block for external access to the Remote Desktop Gateways.

5. When you are satisfied with your application settings, choose **Next**. If you don't want to complete the configuration, choose **Cancel**. When you choose **Cancel**, all of the selections on the specification page are lost and you are returned to the landing page. To return to the previous screen, choose **Previous**.
6. On the **Configure infrastructure settings** page, you are prompted to define the infrastructure settings for the new deployment. The following tab provides information about the input fields.

   Storage and compute

   You can choose to select your instances, or to use AWS recommended resources. If you choose to use AWS recommended resources, you have the option of defining your performance needs. If you don't select either option, default values are assigned. Launch Wizard will display the estimated charges incurred to deploy the application based on suggested infrastructure and also based on static values.

   - **Based on infrastructure suggestion**. Launch Wizard displays the suggested resources for the deployment. You can specify your performance requirements of the resources to update the recommendation.
     - **Number of instance cores**. Choose the number of CPU cores for your infrastructure. The default value assigned is 4.
     - **Network performance**. Choose your preferred network performance in Gbps.

    - **Memory (GB)**. Choose the amount of RAM that you want to attach to your EC2 instances. The default value assigned is 4 GB.
    - **Recommended resources**. Launch Wizard displays the system-recommended resources based on your infrastructure selections. If you want to change the recommended resources, select different infrastructure settings.
    - **Estimated on-demand cost to deploy additional resources**. Launch Wizard displays the estimated charges incurred to deploy the resources.

  - **Based on static values**. You can specify specific instance types for the resources used in your deployment. If you don't select either option, default values are assigned.
    - **Instance type**. You can choose your instance type from the dropdown list, or you can use AWS recommended resources.
    - **Estimated on-demand cost to deploy additional resources**. Launch Wizard displays the estimated charges incurred to deploy the resources.

7. When you are satisfied with your infrastructure settings, select **Next**. If you don't want to complete the configuration, select **Cancel**. When you select **Cancel**, all of the selections on the specification page are lost and you are returned to the landing page. To go to the previous screen, select **Previous**.
8. On the **Review and deploy** page, review your configuration details. If you want to make changes, select **Previous**. To stop, select **Cancel**. When you select **Cancel**, all of the selections on the specification page are lost and you are returned to the landing page. When you choose **Deploy**, you agree to the terms of the **Acknowledgment**. Launch Wizard validates the inputs and notifies you if you need to address any issues.
9. When validation is complete, Launch Wizard deploys your AWS resources and configures your application. Launch Wizard provides you with status updates about the progress of the deployment on the **Deployments** page. From the **Deployments** page, you can view the list of current and previous deployments.
10. When your deployment is ready, a notification informs you that your application is successfully deployed. If you have set up an Amazon SNS notification, you are also alerted through Amazon SNS. You can manage and access all of the resources related to your application by selecting the deployment, and then selecting **Manage** from the **Actions** dropdown list.
11. When the application is deployed, you can access your EC2 instances through the Amazon EC2 console.

Deploy AWS Directory Service for Microsoft Active Directory to a new VPC

The following steps guide you through an Active Directory deployment with AWS Launch Wizard after you have launched it from the console for a new VPC.

1. On the Launch Wizard Console's landing page, use the **Choose application** button. This opens the Choose application wizard where you are prompted to select the type of application that you want to deploy.
2. Select **Active Directory**, select **Deploy AWS Managed Microsoft AD into a new VPC**, then select **Create deployment.**
3. Review and acknowledge the required IAM permissions are met before proceeding. For more information, see Identity and Access Management for AWS Launch Wizard.
4. On the **Configure application settings** page, you are prompted to enter the specifications for the new deployment. The following tabs provide information about the specification fields of the deployment model.

   General settings

   - **Deployment name**. Enter a unique application name for your deployment.
   - **Amazon Simple Notification Service (Amazon SNS) topic ARN — optional**. Specify an Amazon SNS topic where Launch Wizard can send notifications and alerts. For more information, see the Amazon Simple Notification Service Developer Guide.
   - **Deactivate rollback on failed deployment**. By default, if a deployment fails, your provisioned resources will be deleted. You can enable this setting during deployment to prevent this behavior.
   - **Tags - optional**. Enter a key and value to assign metadata to your deployment. For help with tagging, see Tagging Your Amazon EC2 Resources.

   Network configuration

Parameter label (name)	Default value	Description
Availability zones (AvailabilityZones)	***Requires input***	List of Availability Zones to use for the subnets in the VPC. Note: The logical

Parameter label (name)	Default value	Description
		order is preserved and only two Availability Zones are used for this deployment.
Number of Availability Zones (NumberOfAZs)	2	Number of Availabil ity Zones to use in the VPC. This must match your selections in the list of Availability Zones parameter.
VPC CIDR (VPCCIDR)	10.0.0.0/16	CIDR Block for the VPC.
Create a DHCP options set (DHCPOptionSet)	Yes	Creates and associates a new DHCP Options Set to your VPC.
Private subnet 1 CIDR (PrivateSubnet1CIDR)	10.0.0.0/19	CIDR block for private subnet 1 located in Availability Zone 1.
Private subnet 2 CIDR (PrivateSubnet2CIDR)	10.0.32.0/19	CIDR block for private subnet 2 located in Availability Zone 2.
(Optional) Private subnet 3 CIDR (PrivateSubnet3CIDR)	***Blank string***	CIDR block for private subnet 3 located in Availability Zone 3.
Public subnet 1 CIDR (PublicSubnet1CIDR)	10.0.128.0/20	CIDR Block for the public subnet 1 located in Availability Zone 1.
Public subnet 2 CIDR (PublicSubnet2CIDR)	10.0.144.0/20	CIDR Block for the public subnet 2 located in Availability Zone 2.

Parameter label (name)	Default value	Description
(Optional) Public subnet 3 CIDR (PublicSubnet3CIDR)	***Blank string***	CIDR Block for the public subnet 3 located in Availability Zone 3.

Amazon EC2 configuration

Parameter label (name)	Default value	Description
Key pair name (KeyPairName)	***Requires input***	Public/private key pairs allow you to securely connect to your instance after it launches.

Microsoft Active Directory configuration

Parameter label (name)	Default value	Description
Domain DNS name (DomainDNSName)	example.com	Fully qualified domain name (FQDN) of the forest root domain. For example, example.com.
Domain NetBIOS name (DomainNetBIOSName)	example	NetBIOS name of the domain (between 1 to 15 characters) for users of earlier versions of Windows. For example, EXAMPLE.

Parameter label (name)	Default value	Description
Admin account password (DomainAdminPassword)	***Requires input***	Password for the Admin account. Must be at least 8 characters containing letters, numbers and symbols.
AWS Managed Microsoft AD edition (ADEdition)	Enterprise	The AWS Managed Microsoft AD Edition you wish to deploy.

Microsoft Windows Server management instance

Parameter label (name)	Default value	Description
Deploy management server (MgmtServer)	TRUE	Deploys an EC2 instance to act as a management server.
Data drive size (MgmtDataDriveSizeGiB)	2	Size of the management server data drive in GiB.
Management server NetBIOS name (MgmtServerNetBIOSName)	MGMT1	NetBIOS name of the management server (between 1-15 characters).

Microsoft Active Directory Certificate Services configuration

Parameter label (name)	Default value	Description
Certificate authority (CA) deployment type (PKI)	No	Deploy two-tier (Offline Root with Subordinate Enterprise CA) or one-tier (Enterprise Root CA) PKI Infrastructure.
CA data drive size (CaDataDriveSizeGiB)	2	Size of the data drive in GiB for the CA instance(s).
Offline root CA NetBIOS name (Only Used For two-tier PKI) (OrCaServerNetBIOSName)	ORCA1	NetBIOS name of the offline root CA server, used only for two-tier PKI (between 1-15 characters).
Enterprise root or subordinate CA NetBIOS name (EntCaServerNetBIOSName)	ENTCA1	NetBIOS name of the enterprise root (one-tier) or subordinate CA server (two-tier). The value must be 1-15 characters.
CA key length (CaKeyLength)	2048	CA(s) cryptographic provider key length.
CA hash algorithm (CaHashAlgorithm)	SHA256	CA(s) hash algorithm for signing certificates.
Offline root CA certificate validity period (only used for two-tier PKI) (OrCaValidityPeriodUnits)	10	Validity period in years for the root CA certificate (used only for two-tier PKI).

Parameter label (name)	Default value	Description
Enterprise root or subordinate CA certificate validity period (CaValidityPeriodUnits)	5	Validity period in years for the subordinate CA certificate.
Use S3 for CA CRL location (UseS3ForCRL)	No	Store CA CRL(s) in an S3 bucket.
CA CRL S3 bucket name (S3CRLBucketName)	examplebucket	S3 bucket name for CA CRL(s) storage. Quick Start bucket name can include numbers, lowercase letters, uppercase letters, and hyphens (-). It cannot start or end with a hyphen (-).

Microsoft Remote Desktop Gateway configuration

Parameter label (name)	Default value	Description
Number of RDGW hosts (NumberOfRDGWHosts)	1	Enter the number of Remote Desktop Gateway instances to create.
Allowed Remote Desktop Gateway external access CIDR (RDGWCIDR)	***Requires input***	Allowed CIDR block for external access to the Remote Desktop Gateways.

5. When you are satisfied with your application settings, choose **Next**. If you don't want to complete the configuration, choose **Cancel**. When you choose **Cancel**, all of the selections on the specification page are lost and you are returned to the landing page. To return to the previous screen, choose **Previous**.

6. On the **Configure infrastructure settings** page, you are prompted to define the infrastructure settings for the new deployment. The following tab provides information about the input fields.

Storage and compute

You can choose to select your instances, or to use AWS recommended resources. If you choose to use AWS recommended resources, you have the option of defining your performance needs. If you don't select either option, default values are assigned. Launch Wizard will display the estimated charges incurred to deploy the application based on suggested infrastructure and also based on static values.

- **Based on infrastructure suggestion**. Launch Wizard displays the suggested resources for the deployment. You can specify your performance requirements of the resources to update the recommendation.
  - **Number of instance cores**. Choose the number of CPU cores for your infrastructure. The default value assigned is 4.
  - **Network performance**. Choose your preferred network performance in Gbps.
  - **Memory (GB)**. Choose the amount of RAM that you want to attach to your EC2 instances. The default value assigned is 4 GB.
  - **Recommended resources**. Launch Wizard displays the system-recommended resources based on your infrastructure selections. If you want to change the recommended resources, select different infrastructure settings.
  - **Estimated on-demand cost to deploy additional resources**. Launch Wizard displays the estimated charges incurred to deploy the resources.

- **Based on static values**. You can specify specific instance types for the resources used in your deployment. If you don't select either option, default values are assigned.
  - **Instance type**. You can choose your instance type from the dropdown list, or you can use AWS recommended resources.
  - **Estimated on-demand cost to deploy additional resources**. Launch Wizard displays the estimated charges incurred to deploy the resources.

7. When you are satisfied with your infrastructure settings, select **Next**. If you don't want to complete the configuration, select **Cancel**. When you select **Cancel**, all of the selections on the specification page are lost and you are returned to the landing page. To go to the previous screen, select **Previous**.
8. On the **Review and deploy** page, review your configuration details. If you want to make changes, select **Previous**. To stop, select **Cancel**. When you select **Cancel**, all of the selections

on the specification page are lost and you are returned to the landing page. When you choose **Deploy**, you agree to the terms of the **Acknowledgment**. Launch Wizard validates the inputs and notifies you if you need to address any issues.

9. When validation is complete, Launch Wizard deploys your AWS resources and configures your application. Launch Wizard provides you with status updates about the progress of the deployment on the **Deployments** page. From the **Deployments** page, you can view the list of current and previous deployments.
10. When your deployment is ready, a notification informs you that your application is successfully deployed. If you have set up an Amazon SNS notification, you are also alerted through Amazon SNS. You can manage and access all of the resources related to your application by selecting the deployment, and then selecting **Manage** from the **Actions** dropdown list.
11. When the application is deployed, you can access your EC2 instances through the Amazon EC2 console.

Deploy Active Directory to an existing VPC (Console)

You can use AWS Launch Wizard to deploy Active Directory to an existing virtual private cloud (VPC) as a self-managed directory on Amazon Elastic Compute Cloud instances, extend your existing active directory into an existing VPC with Amazon EC2 instances, or create an AWS Directory Service for Microsoft Active Directory directory in an existing VPC.

Contents

Deploy self-managed Active Directory to an existing VPC

The following steps guide you through an Active Directory deployment with AWS Launch Wizard after you have launched it from the console for an existing VPC.

1. On the Launch Wizard Console's landing page, use the **Choose application** button. This opens the Choose application wizard where you are prompted to select the type of application that you want to deploy.

2. Select **Active Directory**, select **Deploy self-managed AD into an existing VPC**, then select **Create deployment.**
3. Review and acknowledge the required IAM permissions are met before proceeding. For more information, see Identity and Access Management for AWS Launch Wizard.
4. You are prompted to enter the specifications for the new deployment. The following tabs provide information about the specification fields of the deployment model.

   General settings

   - **Deployment name**. Enter a unique application name for your deployment.
   - **Amazon Simple Notification Service (Amazon SNS) topic ARN — optional**. Specify an Amazon SNS topic where Launch Wizard can send notifications and alerts. For more information, see the Amazon Simple Notification Service Developer Guide.
   - **Deactivate rollback on failed deployment**. By default, if a deployment fails, your provisioned resources will be deleted. You can enable this setting during deployment to prevent this behavior.
   - **Tags - optional**. Enter a key and value to assign metadata to your deployment. For help with tagging, see Tagging Your Amazon EC2 Resources.

   Network configuration

Parameter label (name)	Default value	Description
VPC CIDR (VPCCIDR)	10.0.0.0/16	CIDR Block for the VPC.
VPC ID (VPCID)	***Requires input***	ID of the VPC (for example, vpc-abcd0123).
Create a DHCP options set (DHCPOptionSet)	Yes	Creates and associates a new DHCP Options Set to your VPC.
Subnet 1 ID (PrivateS ubnet1ID)	***Requires input***	ID of subnet 1 in Availabil ity Zone 1 (for example, subnet-abcd0123).

Parameter label (name)	Default value	Description
Subnet 2 ID (PrivateSubnet2ID)	***Requires input***	ID of subnet 2 in Availability Zone 2 (for example, subnet-01234abcd).

Amazon EC2 configuration

Parameter label (name)	Default value	Description
Domain controller 1 NetBIOS name (ADServer1NetBIOSName)	DC1	NetBIOS name of the first Active Directory domain controller (between 1-15 characters).
Domain controller 1 private IP address (ADServer1PrivateIP)	10.0.0.10	Fixed private IP for the first Active Directory domain controller located in Availability Zone 1.
Domain controller 2 NetBIOS name (ADServer2NetBIOSName)	DC2	NetBIOS name of the second Active Directory domain controller (between 1-15 characters).
Domain controller 2 private IP address (ADServer2PrivateIP)	10.0.32.10	Fixed private IP for the second Active Directory domain controller located in Availability Zone 2.
SYSVOL and NTDS Data Drive Size (DataDriveSizeGiB)	10	Size of SYSVOL and NTDS data drive in GiB.

Parameter label (name)	Default value	Description
KMS key for Amazon EBS encryption (EbsEncryptionKmsKeyId)	alias/aws/ebs	The identifier of the KMS key to use for Amazon EBS encryption. You can specify the KMS key using any of the following; key ID, key alias, key ARN, alias ARN.
Key pair name (KeyPairName)	***Requires input***	Public/private key pairs allow you to securely connect to your instance after it launches.
AMI ID (LatestAmiId)	/aws/service/ami-windows-latest/Windows_Server-2022-English-Full-Base	Systems Manager parameter value for latest Windows Server AMI.

Microsoft Active Directory Domain Services configuration

Parameter label (name)	Default value	Description
Domain admin user name (DomainAdminUser)	Admin	User name for the account that will be added as a Domain Administrator. This is separate from the default "Administrator" account.
Domain admin password (DomainAdminPassword)	***Requires input***	Password for the account named above. Must be at least 8 characters containing letters, numbers and symbols.

Parameter label (name)	Default value	Description
Domain DNS name (DomainDNSName)	example.com	Fully qualified domain name (FQDN) of the forest root domain. For example, example.com.
Domain NetBIOS name (DomainNetBIOSName)	example	NetBIOS name of the domain (between 1 to 15 characters) for users of earlier versions of Windows. For example, EXAMPLE.
Create Default OUs (CreateDefaultOUs)	No	Domain Elevated Accounts, Domain Users, Domain Computers, Domain Servers, Domain Service Accounts, and Domain Groups OUs and set the default users and computers containers to Domain Users and Domain Computers.
Set new tombstone lifetime (TombstoneLifetime)	180	The number of days before a deleted object, not recoverable by Active Directory natively, is permanently removed.
Set new deleted objects lifetime (DeletedObjectLifetime)	180	The number of days a deleted Active Directory object is restorable from the Active Directory Recycle Bin, with no loss of information.

Microsoft Active Directory Certificate Services configuration

Parameter label (name)	Default value	Description
Certificate authority (CA) deployment type (PKI)	No	Deploy two-tier (Offline Root with Subordinate Enterprise CA) or one-tier (Enterprise Root CA) PKI Infrastructure.
CA data drive size (CaDataDriveSizeGiB)	2	Size of the data drive in GiB for the CA instance(s).
Offline root CA NetBIOS name (Only Used For two-tier PKI) (OrCaServerNetBIOSName)	ORCA1	NetBIOS name of the offline root CA server (used only for two-tier PKI) (between 1-15 characters).
Enterprise root or subordinate CA NetBIOS name (EntCaServerNetBIOSName)	ENTCA1	NetBIOS name of the enterprise root (one-tier) or subordinate CA server (two-tier). The value must be 1-15 characters.
CA key length (CaKeyLength)	2048	CA(s) cryptographic provider key length.
CA hash algorithm (CaHashAlgorithm)	SHA256	CA(s) hash algorithm for signing certificates.
Offline root CA certificate validity period (only used for two-tier PKI) (OrCaValidityPeriodUnits)	10	Validity period in years for the offline root CA certificate (used only for two-tier PKI).

Parameter label (name)	Default value	Description
Enterprise root or subordinate CA certificate validity period (CaValidityPeriodUnits)	5	Validity period in years for the enterprise root or subordinate CA certificate.
Use Amazon S3 for CA CRL location (UseS3ForCRL)	No	Store CA CRL(s) in an S3 bucket.
CA CRL Amazon S3 bucket name (S3CRLBucketName)	examplebucket	S3 bucket name for CA CRL(s) storage. Bucket name can include numbers, lowercase letters, uppercase letters, and hyphens (-). It cannot start or end with a hyphen (-).

Microsoft Remote Desktop Gateway configuration

Parameter label (name)	Default value	Description
Number of RDGW hosts (NumberOfRDGWHosts)	1	Enter the number of Remote Desktop Gateway instances to create.
Allowed Remote Desktop Gateway external access CIDR (RDGWCIDR)	***Requires input***	Allowed CIDR block for external access to the Remote Desktop Gateways.

5. When you are satisfied with your application settings, choose **Next**. If you don't want to complete the configuration, choose **Cancel**. When you choose **Cancel**, all of the selections on the specification page are lost and you are returned to the landing page. To return to the previous screen, choose **Previous**.
6. On the **Configure infrastructure settings** page, you are prompted to define the infrastructure settings for the new deployment. The following tab provides information about the input fields.

Storage and compute

You can choose to select your instances, or to use AWS recommended resources. If you choose to use AWS recommended resources, you have the option of defining your performance needs. If you don't select either option, default values are assigned. Launch Wizard will display the estimated charges incurred to deploy the application based on suggested infrastructure and also based on static values.

- **Based on infrastructure suggestion**. Launch Wizard displays the suggested resources for the deployment. You can specify your performance requirements of the resources to update the recommendation.
  - **Number of instance cores**. Choose the number of CPU cores for your infrastructure. The default value assigned is 4.
  - **Network performance**. Choose your preferred network performance in Gbps.
  - **Memory (GB)**. Choose the amount of RAM that you want to attach to your EC2 instances. The default value assigned is 4 GB.
  - **Recommended resources**. Launch Wizard displays the system-recommended resources based on your infrastructure selections. If you want to change the recommended resources, select different infrastructure settings.
  - **Estimated on-demand cost to deploy additional resources**. Launch Wizard displays the estimated charges incurred to deploy the resources.
- **Based on static values**. You can specify specific instance types for the resources used in your deployment. If you don't select either option, default values are assigned.
  - **Instance type**. You can choose your instance type from the dropdown list, or you can use AWS recommended resources.
  - **Estimated on-demand cost to deploy additional resources**. Launch Wizard displays the estimated charges incurred to deploy the resources.

7. When you are satisfied with your infrastructure settings, select **Next**. If you don't want to complete the configuration, select **Cancel**. When you select **Cancel**, all of the selections on the specification page are lost and you are returned to the landing page. To go to the previous screen, select **Previous**.
8. On the **Review and deploy** page, review your configuration details. If you want to make changes, select **Previous**. To stop, select **Cancel**. When you select **Cancel**, all of the selections

on the specification page are lost and you are returned to the landing page. When you choose **Deploy**, you agree to the terms of the **Acknowledgment**. Launch Wizard validates the inputs and notifies you if you need to address any issues.

9. When validation is complete, Launch Wizard deploys your AWS resources and configures your application. Launch Wizard provides you with status updates about the progress of the deployment on the **Deployments** page. From the **Deployments** page, you can view the list of current and previous deployments.
10. When your deployment is ready, a notification informs you that your application is successfully deployed. If you have set up an Amazon SNS notification, you are also alerted through Amazon SNS. You can manage and access all of the resources related to your application by selecting the deployment, and then selecting **Manage** from the **Actions** dropdown list.
11. When the application is deployed, you can access your EC2 instances through the Amazon EC2 console.

Extend on-premises Active Directory to an existing VPC

The following steps guide you through an Active Directory deployment with AWS Launch Wizard after you have launched it from the console for an existing VPC.

1. On the Launch Wizard console's landing page, use the **Choose application** button. This opens the Choose application wizard where you are prompted to select the type of application that you want to deploy.
2. Select **Active Directory**, select **Extend on-premises AD into an existing VPC**, then select **Create deployment.**
3. Review and acknowledge that the required IAM permissions are met before proceeding. For more information, see Identity and Access Management for AWS Launch Wizard.
4. When prompted, enter the specifications for the new deployment. The following tabs provide information about the specification fields of the deployment model.

   General settings

   - **Deployment name**. Enter a unique application name for your deployment.
   - **Amazon Simple Notification Service (Amazon SNS) topic ARN — optional**. Specify an Amazon SNS topic where Launch Wizard can send notifications and alerts. For more information, see the Amazon Simple Notification Service Developer Guide.

- **Deactivate rollback on failed deployment**. By default, if a deployment fails, your provisioned resources will be deleted. You can enable this setting during deployment to prevent this behavior.
- **Tags - optional**. Enter a key and value to assign metadata to your deployment. For help with tagging, see Tagging Your Amazon EC2 Resources.

Network configuration

Parameter label (name)	Default value	Description
Parameter label (name)	Default value	Description
VPC CIDR (VPCCIDR)	10.0.0.0/16	CIDR Block for the VPC.
VPC ID (VPCID)	***Requires input***	ID of the VPC (for example, vpc-abcd0123).
Subnet 1 ID (Subnet1ID)	***Requires input***	ID of subnet 1 in Availabil ity Zone 1 (for example, subnet-abcd0123).
Subnet 2 ID (Subnet2ID)	***Requires input***	ID of subnet 2 in Availabil ity Zone 2 (for example, subnet-01234abcd).
Exiting domain controllers Security Group ID (Existing DomainControllersSG)	sg-1234567890abcdef0	Security Group ID for existing domain controllers Security Group. (Used only when JoinAndPromote equals Yes).

Amazon EC2 configuration

Parameter label (name)	Default value	Description
Domain controller 1 NetBIOS name (ADServer 1NetBIOSName)	DC3	NetBIOS name of the first additional Active Directory domain controller (between 1-15 characters).
Domain controller 1 private IP address (ADServer 1PrivateIP)	10.0.0.11	Fixed private IP for the first additional Active Directory domain controller located in subnet 1.
Domain controller 2 NetBIOS name (ADServer 2NetBIOSName)	DC4	NetBIOS name of the second additional Active Directory domain controller (between 1-15 characters).
Domain controller 2 private IP address (ADServer 2PrivateIP)	10.0.32.11	Fixed private IP for the second additional Active Directory domain controller located in subnet 2.
SYSVOL and NTDS Data Drive Size (DataDriveSizeGiB)	10	Size of SYSVOL and NTDS data drive in GiB.
KMS key for EBS Encryption (EbsEncryptionKmsKeyId)	alias/aws/ebs	The identifier of the KMS key to use for Amazon EBS encryption. You can specify the KMS key using any of the following; key ID, key alias, key ARN, alias ARN.

Parameter label (name)	Default value	Description
Key pair name (KeyPairName)	***Requires input***	Public/private key pairs allow you to securely connect to your instance after it launches.
AMI ID (LatestAmiId)	/aws/service/ami-windows-latest/Windows_Server-2022-English-Full-Base	AWS Systems Manager parameter value for latest Windows Server AMI.

Microsoft Active Directory Domain Services configuration

Parameter label (name)	Default value	Description
Join and Promote to domain controllers (JoinAndPromote)	No	Do you want to join and promote these instances to be Active Directory domain controllers.
DNS Server 1 IP address (ExistingDomainController1IP)	10.0.0.10	The IP address of the first DNS server that can resolve the domain. You must have connectivity from the VPC to the DNS server.
DNS Server 2 IP address (ExistingDomainController2IP)	10.0.32.10	The IP address of the second DNS server that can resolve the domain. You must have connectivity from the VPC to the DNS server.

Parameter label (name)	Default value	Description
Domain DNS name (DomainDNSName)	example.com	Fully qualified domain name (FQDN) of the domain you would like to join and promote to. For example, example.com.
Domain NetBIOS name (DomainNetBIOSName)	example	NetBIOS name of the domain (between 1 to 15 characters) you would like to join and promote to for users of earlier versions of Windows. For example, EXAMPLE.

5. When you are satisfied with your application settings, choose **Next**. If you don't want to complete the configuration, choose **Cancel**. When you choose **Cancel**, all of the selections on the specification page are lost and you are returned to the landing page. To return to the previous screen, choose **Previous**.
6. On the **Configure infrastructure settings** page, you are prompted to define the infrastructure settings for the new deployment. The following tab provides information about the input fields.

   Storage and compute

   You can choose to select your instances, or to use AWS recommended resources. If you choose to use AWS recommended resources, you have the option of defining your performance needs. If you don't select either option, default values are assigned. Launch Wizard will display the estimated charges incurred to deploy the application based on suggested infrastructure and also based on static values.

   - **Based on infrastructure suggestion**. Launch Wizard displays the suggested resources for the deployment. You can specify your performance requirements of the resources to update the recommendation.
     - **Number of instance cores**. Choose the number of CPU cores for your infrastructure. The default value assigned is 4.
     - **Network performance**. Choose your preferred network performance in Gbps.

    - **Memory (GB)**. Choose the amount of RAM that you want to attach to your EC2 instances. The default value assigned is 4 GB.
    - **Recommended resources**. Launch Wizard displays the system-recommended resources based on your infrastructure selections. If you want to change the recommended resources, select different infrastructure settings.
    - **Estimated on-demand cost to deploy additional resources**. Launch Wizard displays the estimated charges incurred to deploy the resources.

  - **Based on static values**. You can specify specific instance types for the resources used in your deployment. If you don't select either option, default values are assigned.
    - **Instance type**. You can choose your instance type from the dropdown list, or you can use AWS recommended resources.
    - **Estimated on-demand cost to deploy additional resources**. Launch Wizard displays the estimated charges incurred to deploy the resources.

7. When you are satisfied with your infrastructure settings, select **Next**. If you don't want to complete the configuration, select **Cancel**. When you select **Cancel**, all of the selections on the specification page are lost and you are returned to the landing page. To go to the previous screen, select **Previous**.
8. On the **Review and deploy** page, review your configuration details. If you want to make changes, select **Previous**. To stop, select **Cancel**. When you select **Cancel**, all of the selections on the specification page are lost and you are returned to the landing page. When you choose **Deploy**, you agree to the terms of the **Acknowledgment**. Launch Wizard validates the inputs and notifies you if you need to address any issues.
9. When validation is complete, Launch Wizard deploys your AWS resources and configures your application. Launch Wizard provides you with status updates about the progress of the deployment on the **Deployments** page. From the **Deployments** page, you can view the list of current and previous deployments.
10. When your deployment is ready, a notification informs you that your application is successfully deployed. If you have set up an Amazon SNS notification, you are also alerted through Amazon SNS. You can manage and access all of the resources related to your application by selecting the deployment, and then selecting **Manage** from the **Actions** dropdown list.
11. When the application is deployed, you can access your EC2 instances through the Amazon EC2 console.

Deploy AWS Directory Service for Microsoft Active Directory to an existing VPC

The following steps guide you through an Active Directory deployment with AWS Launch Wizard after you have launched it from the console for an existing virtual private cloud (VPC).

1. On the Launch Wizard Console's landing page, use the **Choose application** button. This opens the Choose application wizard where you are prompted to select the type of application that you want to deploy.
2. Select **Active Directory**, select **Deploy AWS Managed Microsoft AD into an existing VPC**, then select **Create deployment.**
3. Review and acknowledge the required IAM permissions are met before proceeding. For more information, see Identity and Access Management for AWS Launch Wizard.
4. You are prompted to enter the specifications for the new deployment. The following tabs provide information about the specification fields of the deployment model.

   General settings

   - **Deployment name**. Enter a unique application name for your deployment.
   - **Amazon Simple Notification Service (Amazon SNS) topic ARN — optional**. Specify an Amazon SNS topic where Launch Wizard can send notifications and alerts. For more information, see the Amazon Simple Notification Service Developer Guide.
   - **Deactivate rollback on failed deployment**. By default, if a deployment fails, your provisioned resources will be deleted. You can enable this setting during deployment to prevent this behavior.
   - **Tags - optional**. Enter a key and value to assign metadata to your deployment. For help with tagging, see Tagging Your Amazon EC2 Resources.

   Network Configuration

Parameter label (name)	Default value	Description
VPC CIDR (VPCCIDR)	10.0.0.0/16	CIDR Block for the VPC.
VPC ID (VPCID)	***Requires input***	ID of the VPC (for example, vpc-abcd0123).

Parameter label (name)	Default value	Description
Create a DHCP options set (DHCPOptionSet)	Yes	Creates and associates a new DHCP Options Set to your VPC.
Subnet 1 ID (PrivateSubnet1ID)	***Requires input***	ID of subnet 1 in Availability Zone 1 (for example, subnet-abcd0123).
Subnet 2 ID (PrivateSubnet2ID)	***Requires input***	ID of subnet 2 in Availability Zone 2 (for example, subnet-01234abcd).

AWS Managed Microsoft AD configuration

Parameter label (name)	Default value	Description
Domain DNS name (DomainDNSName)	example.com	Fully qualified domain name (FQDN) of the forest root domain. For example, example.com.
Domain NetBIOS name (DomainNetBIOSName)	example	NetBIOS name of the domain (Between 1 to 15 characters) for users of earlier versions of Windows. For example, EXAMPLE.
Admin account password (DomainAdminPassword)	***Requires input***	Password for the Admin account. Must be at least 8 characters containing letters, numbers and symbols.

Parameter label (name)	Default value	Description
AWS Managed Microsoft AD edition (ADEdition)	Enterprise	The AWS Managed Microsoft AD Edition you wish to deploy.

Management instance

Parameter label (name)	Default value	Description
Deploy management server (MgmtServer)	TRUE	Deploys an EC2 instance to act as a management server.
Management Server SSM Parameter Value for latest AMI ID (MgmtAmi)	/aws/service/ami-windows-latest/Windows_Server-2022-English-Full-Base	Management Server SSM Parameter Value to grab the latest AMI ID.
Data drive size (MgmtDataDriveSizeGiB)	2	Size of the management server data drive in GiB.
Management server NetBIOS name (MgmtServerNetBIOSName)	MGMT1	NetBIOS name of the Management Server server (between 1-15 characters).
Key pair name (KeyPairName)	***Requires input***	Public/private key pairs allow you to securely connect to your instance after it launches.

Microsoft Active Directory Certificate Services configuration

Parameter label (name)	Default value	Description
Certificate authority (CA) deployment type (PKI)	No	Deploy two-tier (Offline Root with Subordinate Enterprise CA) or one-tier (Enterprise Root CA) PKI Infrastructure.
CA AMI ID (CaAmi)	/aws/service/ami-windows-latest/Windows_Server-2022-English-Full-Base	The Systems Manager Parameter Store value used to provision the enterprise root CA.
CA data drive size (CaDataDriveSizeGiB)	2	Size of the data drive in GiB for the CA instance(s).
Offline root CA NetBIOS name (Only Used For two-tier PKI) (OrCaServerNetBIOSName)	ORCA1	NetBIOS name of the offline root CA server, used only for two-tier PKI (between 1-15 characters).
Enterprise root or subordinate CA NetBIOS name (EntCaServerNetBIOSName)	ENTCA1	NetBIOS name of the enterprise root (one-tier) or subordinate CA server (two-tier). The value must be 1-15 characters.
CA key length (CaKeyLength)	2048	CA(s) cryptographic provider key length.
CA hash algorithm (CaHashAlgorithm)	SHA256	CA(s) hash algorithm for signing certificates.

Parameter label (name)	Default value	Description
Offline root CA certificate validity period (only used for two-tier PKI) (OrCaValidityPeriodUnits)	10	Validity period in years for the offline root CA certificate (used only for two-tier PKI).
Enterprise root or subordinate CA certificate validity period (CaValidityPeriodUnits)	5	Validity period in years for the enterprise root or subordinate CA certificate.
Use S3 for CA CRL location (UseS3ForCRL)	No	Store CA CRL(s) in an S3 bucket.
CA CRL S3 bucket name (S3CRLBucketName)	examplebucket	S3 bucket name for CA CRL(s) storage. Bucket name can include numbers, lowercase letters, uppercase letters, and hyphens (-). It cannot start or end with a hyphen (-).

5. When you are satisfied with your application settings, choose **Next**. If you don't want to complete the configuration, choose **Cancel**. When you choose **Cancel**, all of the selections on the specification page are lost and you are returned to the landing page. To return to the previous screen, choose **Previous**.

6. On the **Configure infrastructure settings** page, you are prompted to define the infrastructure settings for the new deployment. The following tab provides information about the input fields.

   Storage and compute

      You can choose to select your instances, or to use AWS recommended resources. If you choose to use AWS recommended resources, you have the option of defining your performance needs. If you don't select either option, default values are assigned. Launch Wizard will display the estimated charges incurred to deploy the application based on suggested infrastructure and also based on static values.

- **Based on infrastructure suggestion**. Launch Wizard displays the suggested resources for the deployment. You can specify your performance requirements of the resources to update the recommendation.
  - **Number of instance cores**. Choose the number of CPU cores for your infrastructure. The default value assigned is 4.
  - **Network performance**. Choose your preferred network performance in Gbps.
  - **Memory (GB)**. Choose the amount of RAM that you want to attach to your EC2 instances. The default value assigned is 4 GB.
  - **Recommended resources**. Launch Wizard displays the system-recommended resources based on your infrastructure selections. If you want to change the recommended resources, select different infrastructure settings.
  - **Estimated on-demand cost to deploy additional resources**. Launch Wizard displays the estimated charges incurred to deploy the resources.

- **Based on static values**. You can specify specific instance types for the resources used in your deployment. If you don't select either option, default values are assigned.
  - **Instance type**. You can choose your instance type from the dropdown list, or you can use AWS recommended resources.
  - **Estimated on-demand cost to deploy additional resources**. Launch Wizard displays the estimated charges incurred to deploy the resources.

7. When you are satisfied with your infrastructure settings, select **Next**. If you don't want to complete the configuration, select **Cancel**. When you select **Cancel**, all of the selections on the specification page are lost and you are returned to the landing page. To go to the previous screen, select **Previous**.
8. On the **Review and deploy** page, review your configuration details. If you want to make changes, select **Previous**. To stop, select **Cancel**. When you select **Cancel**, all of the selections on the specification page are lost and you are returned to the landing page. When you choose **Deploy**, you agree to the terms of the **Acknowledgment**. Launch Wizard validates the inputs and notifies you if you need to address any issues.
9. When validation is complete, Launch Wizard deploys your AWS resources and configures your application. Launch Wizard provides you with status updates about the progress of the deployment on the **Deployments** page. From the **Deployments** page, you can view the list of current and previous deployments.

10. When your deployment is ready, a notification informs you that your application is successfully deployed. If you have set up an Amazon SNS notification, you are also alerted through Amazon SNS. You can manage and access all of the resources related to your application by selecting the deployment, and then selecting **Manage** from the **Actions** dropdown list.
11. When the application is deployed, you can access your EC2 instances through the Amazon EC2 console.

Deploy Active Directory to a new or existing VPC (AWS CLI)

You can use the AWS Launch Wizard CreateDeployment API operation to deploy Active Directory. To create a deployment, you must provide values for various *specifications*. Specifications are a collection of settings that define how your deployment should be created and configured. A workload will have one or more deployment patterns with differing required and optional specifications.

If you want to use the **Clone deployment** action on your deployment, you must create your deployment using the Launch Wizard console.

Prerequisites for deploying Active Directory with the AWS CLI

Before deploying Active Directory with the AWS CLI, ensure you have met the following prerequisites:

- Install and configure the AWS CLI. For more information, see Install or update to the latest version of the AWS CLI.
- Complete the steps in the previous section titled **Set up**. Some deployment patterns have requirements that must be met for a deployment to be successful.

Create an Active Directory deployment with the AWS CLI

You can create a deployment for your Active Directory application using the `CreateDeployment` Launch Wizard API operation.

To create a deployment for Active Directory using the AWS CLI

1. List the available workload names using the ListWorkloads Launch Wizard API operation.

   The following example shows listing the available workloads:

```
aws launchwizard list-workloads --region us-east-1
{
    "workloads": [
        {
            "displayName": "Remote Desktop Gateway",
            "workloadName": "RDGW"
        },
        {
            "displayName": "MS SQL Server",
            "workloadName": "SQL"
        },
        {
            "displayName": "SAP",
            "workloadName": "SAP"
        },
        {
            "displayName": "Microsoft Active Directory",
            "workloadName": "MicrosoftActiveDirectory"
        }
        ...
    ]
}
```

2. Specify the desired workload name with the ListWorkloadDeploymentPatterns operation to describe the supported values for the deployment pattern names.

   The following example lists the available workload patterns for a given workload:

```
aws launch-wizard list-workload-deployment-patterns --workload-
name MicrosoftActiveDirectory --region us-east-1
{
    "workloadDeploymentPatterns": [
        {
            "deploymentPatternName": "adAwsManagedExistingVpc",
            "description": "Example description.",
            "displayName": "ExampleDisplayName",
            "status": "ACTIVE",
            "workloadName": "MicrosoftActiveDirectory",
            "workloadVersionName": "2024-05-03-00-00-00"
        },
        ...
    ]
}
```

3. Use the workload and deployment pattern names you discovered with the [GetWorkloadDeploymentPattern](#) operation to list the specification details.

   The following example lists the workload specifications of a given workload and deployment pattern:

```
aws launchwizard get-workload-deployment-pattern --workload-
name MicrosoftActiveDirectory --deployment-pattern-name adAwsManagedExistingVpc --
region us-east-1
{
    "workloadDeploymentPattern": {
        "deploymentPatternName": "adAwsManagedExistingVpc",
        "description": "Example description.",
        "displayName": "ExampleDisplayName",
        "specifications": [
            {
                "description": "Enter an SNS topic for AWS Launch Wizard to send
 notifications and alerts.",
                "name": "AWS:LaunchWizard:TopicArn",
                "required": "No"
            },
            {
                "description": "When a deployment fails, your provisioned resources
 will be deleted/rolled back by default. If deactivated, the provisioned resources
 will be deleted when you delete your deployment from the Launch Wizard console.",
                "name": "AWS:LaunchWizard:DisableRollbackFlag",
                "required": "No"
            },
            {
                "allowedValues": [
                    "true",
                    "false"
                ],
                "description": "Cloud Watch Application Insights monitoring",
                "name": "SetupAppInsightsMonitoring",
                "required": "Yes"
            },
            ...
        ]
    }
}
```

4. With the workload specifications retrieved, you must provide values for any specification name with a `required` value of `Yes`. You can also provide any optional specifications you require for your deployment. We recommend that you pass inputs to the `specifications` parameter for your deployment as a file for easier usage.

   Your JSON file's format should resemble the following:

   ```
   {
     "ExampleName1": "ExampleValue1",
     "ExampleName2": "ExampleValue2",
     "ExampleName3": "ExampleValue3"
   }
   ```

5. With the specifications file created, you can create a deployment for your chosen workload and deployment pattern.

   The following example creates a deployment with specifications defined in a file:

   ```
   aws launch-wizard create-deployment --workload-name MicrosoftActiveDirectory --
   deployment-pattern-name adAwsManagedExistingVpc --name ExampleDeploymentName --
   region us-east-1 --specifications file://specifications.json
   ```

Manage application resources with AWS Launch Wizard for Active Directory

After you deploy your self-managed domain controllers, you can manage them by following these steps.

1. From the navigation pane, choose **Deployments**.
2. From the **Deployments** page, select **Actions**. You can select to do the following:

   1. **Manage resources on the EC2 console**. You are taken to the Amazon EC2 console, where you can view and manage your domain controller resources. For example, you can view and manage EC2, Amazon EBS, Active Directory, VPC, subnets, NAT Gateways, and Elastic IPs.
   2. **View resource group with SSM**. You are taken to the Systems Manager console to view your resource groups.

3. **View CloudWatch application logs**. You are taken to CloudWatch Logs, where you can monitor, store, and access your Active Directory application.
4. **View your CloudFormation template**. This is the CloudFormation template created by your most recent deployment, and it can be accessed through the CloudFormation console. For help with finding and using your CloudFormation template, see Viewing CloudFormation Stack Data and Resources on the AWS Management Console.

3. To delete a deployment, select the application that you want to delete and select **Delete**. You are prompted to confirm your action.

> ⚠ **Important**
>
> You lose all specification settings for the domain controllers when you delete a deployment. AWS Launch Wizard attempts to delete only the AWS resources that it created in your account as part of the deployment. If you created resources outside of Launch Wizard, for example resources that reside in a VPC created by Launch Wizard, the deletion may fail. Launch Wizard does not delete any Active Directory objects in your Active Directory, nor any of the records in your DNS server. Launch Wizard has no control over your Active Directory domain user password over time, which is required to clean up Active Directory objects or DNS records. We recommend that you remove these entries from your Active Directory after Launch Wizard deletes the deployment. For key operations performed against your Active Directory resulting in new records or entries, see Active Directory configurations.

4. To further investigate details regarding your domain controller resources, select the **Application name**. You can then view the **Deployment events** and **Summary** details for your application by using the tabs at the top of the page.

Post-deployment steps for AWS Launch Wizard Active Directory

Post-deployment steps for AWS Launch Wizard for Active Directory.

Run Windows Updates

To ensure that the operating systems on deployed servers and installed applications have the latest Microsoft updates, run Windows Update on each server.

1. For each deployed server, create an RDP session.

2. Open the **Settings** application.
3. Open **Update & Security**.
4. Click **Check for updates**.
5. Install any updates, and restart your server, if necessary.

High availability and security best practices for AWS Launch Wizard for Active Directory

The domain controller architecture created by AWS Launch Wizard supports AWS best practices for high availability and security as promoted by the AWS Well-Architected Framework.

Topics

- High availability
- Security in Launch Wizard for Active Directory

High availability

With Amazon EC2, you can set the location of instances in multiple locations composed of AWS Regions and Availability Zones. Regions are dispersed and located in separate geographic areas. Availability Zones are distinct locations within a Region that are engineered to be isolated from failures in other Availability Zones. Availability Zones provide inexpensive, low-latency network connectivity to other Availability Zones in the same Region.

When you launch your instances in different Regions, you can set your domain controllers to be closer to specific customers, or to meet legal or other requirements. When you launch your instances in different Availability Zones, you can protect your domain controllers from the failure of a single location.

Security in Launch Wizard for Active Directory

Launch Wizard creates a number of security groups and rules for you. When your directory resources are launched, they must be associated with a security group, which acts as a stateful firewall. You have complete control over the network traffic entering or leaving the security group. You can also build granular rules that are scoped by protocol, port number, and source or destination IP address or subnet. By default, all outbound traffic from a security group is permitted.

Inbound traffic, on the other hand, permits traffic from the VPC used for the deployment and resources that Launch Wizard deploys. You might require additional configuration to allow appropriate traffic to reach your resources.

The Securing the Microsoft Platform on Amazon Web Services whitepaper discusses the different methods for securing your AWS infrastructure. Recommendations include providing isolation between application tiers using security groups. We recommend that you tightly control inbound traffic to reduce the attack surface of your EC2 instances.

Troubleshoot AWS Launch Wizard for Active Directory

Each deployment in your account in the same AWS Region can be uniquely identified by the name specified at the time of a deployment. The deployment name can be used to view the details related to the deployment on the **Deployments** page of the Launch Wizard console.

This section describes steps to help you troubleshoot deploying domain controllers with Launch Wizard for Active Directory.

Contents

Launch Wizard provisioning events

Launch Wizard captures events from CloudFormation to track the status of an ongoing application deployment. If an application deployment fails, you can view the deployment events for this application by selecting **Deployments** from the navigation pane. A failed event shows a status of **Failed** along with a failure message.

CloudWatch Logs

Launch Wizard streams provisioning logs from all of the AWS log sources, such as CloudFormation and PowerShell DSC scripts to CloudWatch Logs. You can view the CloudWatch Logs for a given application name on the CloudWatch console for the log group name `LaunchWizard-`*`APPLICATION_NAME`* and log stream `ApplicationLaunchLog`.

CloudFormation stack

Launch Wizard uses CloudFormation to provision the infrastructure resources of an application. CloudFormation stacks can be found in your account using the CloudFormation `describe-stacks` API. Launch Wizard launches various stacks in your account for validation and application resource creation. The following are the relevant filters for the `describe-stacks` API.

- Application Resources
  - `LaunchWizard-`*`APPLICATION_NAME`*. This stack includes all of the resource creation events for resources created by the deployment.
  - `LaunchWizard-`*`STACK_NAME`*`-`*`TEMPLATE_NAME`*. This log includes all of the logs from each PowerShell script run from within the instance.

You can view the status of these CloudFormation stacks. If any of them fail, you can view the cause of failure.

AWS Launch Wizard for Remote Desktop Gateway

AWS Launch Wizard for Remote Desktop Gateway (RD Gateway) guides you through the sizing, configuration, and deployment of RD Gateway on the AWS Cloud. RD Gateway uses the Remote Desktop Protocol (RDP) over HTTPS to establish a secure, encrypted connection between remote users and Amazon Elastic Compute Cloud instances running Windows, without needing to configure a virtual private network (VPN). This helps reduce the attack surface on your Windows instances while providing a remote administration solution for administrators.

Deployment options

This Launch Wizard application provides two deployment options:

- **Deploy RD Gateway into a new VPC (end-to-end deployment).** Builds a new AWS environment consisting of a VPC, subnets, NAT gateways, security groups, and other infrastructure components, and then deploys RD Gateway into this new VPC.
- **Deploy RD Gateway into an existing VPC.** Provisions standalone RD Gateway instances in your existing AWS infrastructure.

AWS Launch Wizard provides separate templates for these two deployment types. You can also configure CIDR blocks, instance types, and RD Gateway settings.

AWS Regions

Launch Wizard uses various AWS services during the provisioning of the application's environment. Not every workload is supported in all AWS Regions. For a current list of Regions where the workload can be provisioned, see AWS Launch Wizard workload availability.

Features

AWS Launch Wizard provides the following features:

- Simple application deployment
- Application Resource Groups for discoverability
- AWS resource selection

- Cost estimation
- SNS notification
- Early input validation

Simple application deployment

AWS Launch Wizard makes it more efficient for you to deploy third-party applications on AWS, such as Remote Desktop Gateway. When you input the application requirements, AWS Launch Wizard deploys the necessary AWS resources for a production-ready application. This means that you do not have to manage separate infrastructure pieces or spend as much time provisioning and configuring your Remote Desktop Gateway application.

Application Resource Groups for discoverability

Launch Wizard creates a Resource Group for all of the AWS resources created for your Remote Desktop Gateway application. You can manage the resources through the Amazon EC2 console or with AWS Systems Manager. When you access Systems Manager through Launch Wizard, the resources are automatically filtered for you based on your Resource Group. You can manage, patch, and maintain your Remote Desktop Gateway applications in Systems Manager.

AWS resource selection

Launch Wizard considers performance, memory, bandwidth, and other application features to determine the most appropriate instance type for your Remote Desktop Gateway application. You can modify the recommended defaults.

Cost estimation

Launch Wizard provides a cost estimate for a complete deployment. The cost estimate is itemized for each individual resource to deploy. The estimated cost automatically updates each time you change a resource type configuration in the wizard. The provided estimates are for general comparisons only. The estimates are based on On-Demand costs and actual costs may be lower.

SNS notification

You can provide an Amazon SNS topic so that Launch Wizard will send you notifications and alerts about the status of a deployment.

Early input validation

Launch Wizard performs the following resource limit validations at the AWS account level:

- VPC
- Internet gateway
- Number of AWS CloudFormation stacks

Components

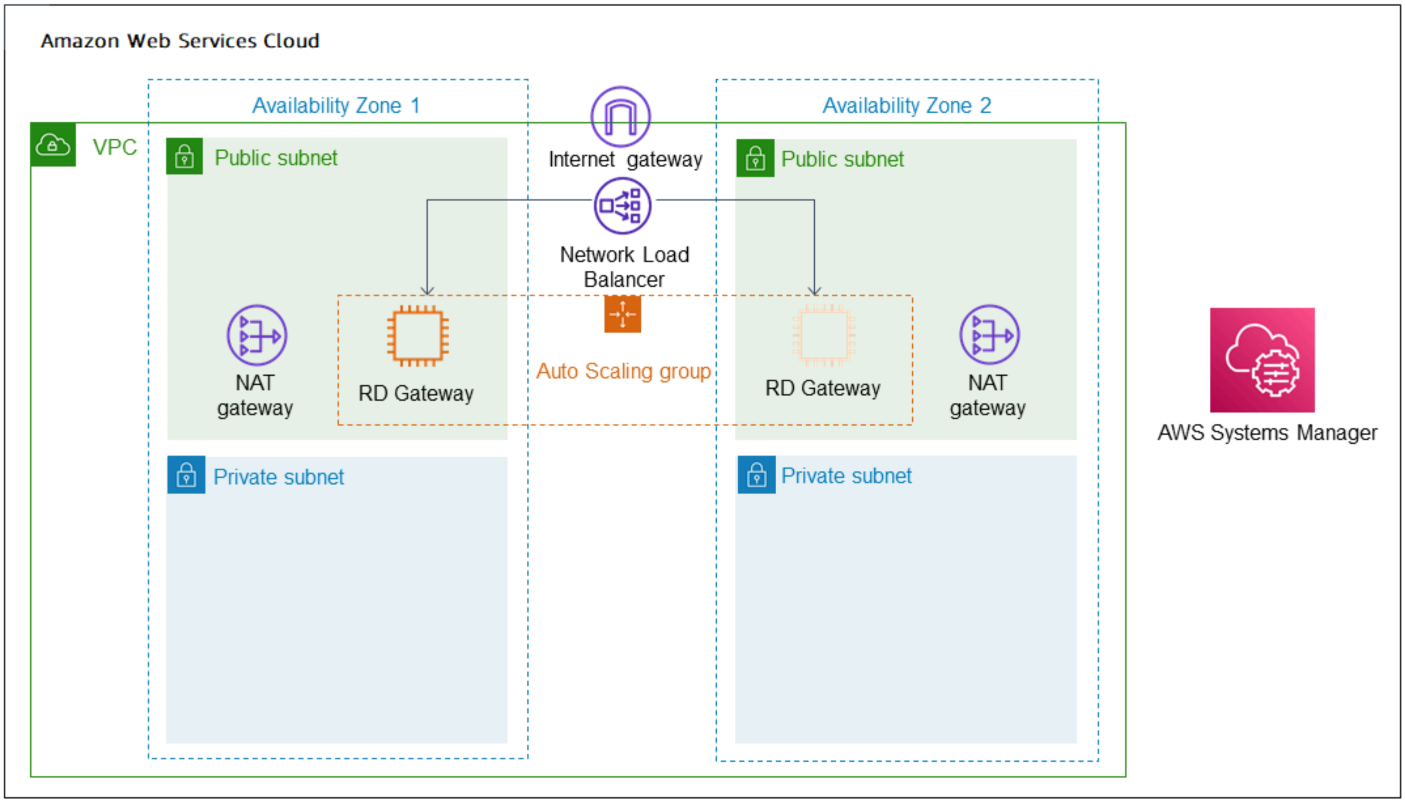
An RD Gateway application deployed with Launch Wizard includes the following components:

- A highly available architecture that spans two Availability Zones.
- In each public subnet, up to four RD Gateway instances in an Auto Scaling group to provide secure remote access to instances in the private subnets. Each instance is assigned an Elastic IP address so it’s reachable directly from the internet.
- A Network Load Balancer to provide RDP access to the RD Gateway instances.
- A security group for Windows instances that will host the RD Gateway role, with an ingress rule permitting TCP port 3389 from your administrator IP address. After deployment, you’ll modify the security group ingress rules to configure administrative access through TCP port 443 instead.
- An empty application tier for instances in private subnets. If more tiers are required, you can create additional private subnets with unique CIDR ranges.
- AWS Systems Manager Parameter Store to securely store credentials used for accessing the RD Gateway instances.
- AWS Systems Manager to automate the deployment of the RD Gateway Auto Scaling group.
- Self-signed SSL certificate and configuration of Remote Desktop Connection Authorization Policies (RD CAPs) and RD Gateway.
- Resource Groups that contain all the resources created with Launch Wizard.

Additionally, a new VPC deployment includes the following components:

- A VPC configured with public and private subnets according to AWS best practices, to provide you with your own virtual network on AWS.
- An internet gateway to allow access to the internet. This gateway is used by the RD Gateway instances to send and receive traffic.

- Managed network address translation (NAT) gateways to allow outbound internet access for resources in the private subnets.



Get Started with AWS Launch Wizard for Remote Desktop Gateway

This section contains information to help you set up your environment to deploy RD Gateway with Launch Wizard. When your environment is set up, you can deploy RD Gateway application with Launch Wizard by following the steps and parameter specification details provided in this section.

Topics to help you get started:

- Access AWS Launch Wizard
- Specialized knowledge
- Amazon Web Services account
- Service Quotas
- Amazon Elastic Compute Cloud key pairs

- AWS Identity and Access Management permissions

Access AWS Launch Wizard

You can launch AWS Launch Wizard from the AWS Launch Wizard console located at https://console.aws.amazon.com/launchwizard.

Specialized knowledge

This deployment requires a moderate level of familiarity with AWS services. If you're new to AWS, see Getting Started Resource Center and AWS Training and Certification. These sites provide materials for learning how to design, deploy, and operate your infrastructure and applications on the AWS Cloud.

This Launch Wizard assumes familiarity with Remote Desktop Gateway.

Amazon Web Services account

Sign up for an AWS account

If you do not have an AWS account, complete the following steps to create one.

To sign up for an AWS account

1. Open https://portal.aws.amazon.com/billing/signup.
2. Follow the online instructions.

   Part of the sign-up procedure involves receiving a phone call or text message and entering a verification code on the phone keypad.

   When you sign up for an AWS account, an *AWS account root user* is created. The root user has access to all AWS services and resources in the account. As a security best practice, assign administrative access to a user, and use only the root user to perform tasks that require root user access.

AWS sends you a confirmation email after the sign-up process is complete. At any time, you can view your current account activity and manage your account by going to https://aws.amazon.com/ and choosing **My Account**.

Create a user with administrative access

After you sign up for an AWS account, secure your AWS account root user, enable AWS IAM Identity Center, and create an administrative user so that you don't use the root user for everyday tasks.

Secure your AWS account root user

1. Sign in to the AWS Management Console as the account owner by choosing **Root user** and entering your AWS account email address. On the next page, enter your password.

   For help signing in by using root user, see Signing in as the root user in the *AWS Sign-In User Guide*.

2. Turn on multi-factor authentication (MFA) for your root user.

   For instructions, see Enable a virtual MFA device for your AWS account root user (console) in the *IAM User Guide*.

Create a user with administrative access

1. Enable IAM Identity Center.

   For instructions, see Enabling AWS IAM Identity Center in the *AWS IAM Identity Center User Guide*.

2. In IAM Identity Center, grant administrative access to a user.

   For a tutorial about using the IAM Identity Center directory as your identity source, see Configure user access with the default IAM Identity Center directory in the *AWS IAM Identity Center User Guide*.

Sign in as the user with administrative access

- To sign in with your IAM Identity Center user, use the sign-in URL that was sent to your email address when you created the IAM Identity Center user.

  For help signing in using an IAM Identity Center user, see Signing in to the AWS access portal in the *AWS Sign-In User Guide*.

Assign access to additional users

1. In IAM Identity Center, create a permission set that follows the best practice of applying least-privilege permissions.

   For instructions, see Create a permission set in the *AWS IAM Identity Center User Guide*.
2. Assign users to a group, and then assign single sign-on access to the group.

   For instructions, see Add groups in the *AWS IAM Identity Center User Guide*.

Service Quotas

If necessary, request service quota increases for the following resources. You might need to request increases if your existing deployment currently uses these resources and if this Launch Wizard deployment could result in exceeding the default quotas. The Service Quotas console displays your usage and quotas for some aspects of some services. For more information, see What is Service Quotas? and AWS service quotas.

Existing VPC Service Quotas:

Resource	Default quota	This deployment uses
Elastic IP Addresses	5 per Region	2
AWS Identity and Access Management (IAM) security groups	300 per account	1
IAM roles	1,000 per account	1
Auto Scaling groups	200 per Region	1
Amazon EC2 On-Demand Instances (Standard)	5 per Region	1-4

New VPC Service Quotas:

Resource	Default quota	This deployment uses
VPCs	5 per Region	1
Elastic IP Addresses	5 per Region	2
Internet Gateway	5 per Region	1
AWS Identity and Access Management (IAM) security groups	300 per account	1
IAM roles	1,000 per account	1
Auto Scaling groups	200 per Region	1
Amazon EC2 On-Demand Instances (Standard)	5 per Region	1-4

Amazon Elastic Compute Cloud key pairs

Ensure that at least one Amazon EC2 key pair exists in your AWS account in the Region where you plan to deploy the Launch Wizard application. Note the key pair name because you will use it during deployment. To create a key pair, see Amazon EC2 key pairs and EC2 instances.

For testing or proof-of-concept purposes, we recommend creating a new key pair instead of using one that’s already being used by a production instance.

AWS Identity and Access Management permissions

Before deploying the Launch Wizard application, you must sign in to the AWS Management Console with IAM permissions for the resources that the templates deploy. The *AdministratorAccess* managed policy within IAM provides sufficient permissions, although your organization may choose to use a custom policy with more restrictions. For more information, see AWS managed policies for job functions.

Deploy standalone Remote Desktop Gateway into a new VPC (Console)

The following steps guide you through a Remote Desktop Gateway deployment with AWS Launch Wizard after you have launched it from the console.

1. When you select **Choose application** from the AWS Launch Wizard landing page, you are directed to the Choose application wizard where you are prompted to select the type of application that you want to deploy.
2. Select **Microsoft Remote Desktop Gateway**, select **Deploy into a new VPC**, then select **Create deployment.**
3. You are prompted to enter the specifications for the new deployment. The following tabs provide information about the specification fields of the deployment model.

   General

   - **Deployment name**. Enter a unique application name for your deployment.
   - **Amazon Simple Notification Service (SNS) topic ARN — optional**. Specify an Amazon SNS topic where AWS Launch Wizard can send notifications and alerts. For more information, see the Amazon Simple Notification Service Developer Guide.
   - **Deactivate rollback on failed deployment**. By default, if a deployment fails, your provisioned resources will be deleted. You can enable this setting during deployment to prevent this behavior.
   - **Tags - optional**. Enter a key and value to assign metadata to your deployment. For help with tagging, see Tagging Your EC2 Resources.

   Network configuration

   - **Key pair name**. Select an existing key pair from the dropdown list or create a new one. If you select **Create new key pair name**, you are directed to the Amazon EC2 console. From there, under **Network and Security**, choose **Key Pairs**. Choose **Create a new key pair**, enter a name for the key pair, and then choose **Download Key Pair**.

⚠ **Important**

This is the only opportunity for you to save the private key file. Download it and save it in a safe place. You must provide the name of your key pair when you launch an instance and provide the corresponding private key each time that you connect to the instance. Return to the Launch Wizard console and choose the refresh button next to the **Key Pairs** dropdown list. The newly created key pair appears in the dropdown list. For more information about key pairs, see Amazon EC2 Key Pairs and Windows Instances.

- **Availability Zone (AZ) configuration:** You must choose at least two Availability Zones. Deployment will create a highly available architecture that spans these Availability Zones.
- **VPC Settings:** Launch Wizard creates your VPC in this case. The following shows Input fields that define VPC configuration.

Parameter label (name)	Default value	Description
VPC tenancy	default	The allowed tenancy of instances launched into the VPC.
VPC CIDR	10.0.0.0/16	CIDR block for the VPC.
Private subnet 1 CIDR	10.0.0.0/19	CIDR block for private subnet 1 located in Availability Zone 1.
Private subnet 2 CIDR	10.0.32.0/19	CIDR block for private subnet 2 located in Availability Zone 2.
Public subnet 1 CIDR	10.0.128.0/20	CIDR Block for the public DMZ subnet 1 located in Availability Zone 1.

Parameter label (name)	Default value	Description
Public subnet 2 CIDR	10.0.144.0/20	CIDR Block for the public DMZ subnet 2 located in Availability Zone 2.
Allowed Remote Desktop Gateway external access CIDR	***Requires input***	Allowed CIDR block for external access to the Remote Desktop Gateways.

Microsoft Remote Desktop Gateway configuration

Parameter label (name)	Default value	Description
Number of RDGW hosts	1	Enter the number of Remote Desktop Gateway hosts to create.
Admin user name	StackAdmin	User name for the new local administrator account.
Admin password	***Requires input***	Password for the administr ative account. Must be at least 8 characters containin g letters, numbers, and symbols.

4. When you are satisfied with your infrastructure selections, select **Next**. If you don't want to complete the configuration, select **Cancel**. When you select **Cancel**, all of the selections on the specification page are lost and you are returned to the landing page. To go to the previous screen, select **Previous**.

5. After configuring your application, you are prompted to define the infrastructure requirements for the new deployment on the **Define infrastructure requirements** page. The following tabs provide information about the input fields.

Compute

- **Infrastructure requirements based on instance type**. You can choose to select your instances or use AWS recommended resources. If you choose to use AWS recommended resources, you have the option of defining your performance needs. If no selections are made, default values are assigned.
- **Number of instance cores**. Choose the number of CPU cores for your infrastructure. The default value assigned is 4.
- **Network performance**. Choose your preferred network performance in Gbps.
- **Memory (GB)**. Choose the amount of RAM that you want to attach to your EC2 instances. The default value assigned is 4 GB.
- **Recommended resources**. Launch Wizard displays the system-recommended resources based on your infrastructure selections. If you want to change the recommended resources, select different infrastructure requirements.
- **Infrastructure requirements based on instance type**. You can choose to select your instance or use AWS recommended resources. If no selections are made, default values are assigned.
- **Instance type**. Select your preferred instance type from the dropdown list.

6. When you are satisfied with your infrastructure selections, select **Next**. If you don't want to complete the configuration, select **Cancel**. When you select **Cancel**, all of the selections on the specification page are lost and you are returned to the landing page. To go to the previous screen, select **Previous**.
7. On the **Review and deploy** page, review your configuration details. If you want to make changes, select **Previous**. To stop, select **Cancel**. When you select **Cancel**, all of the selections on the specification page are lost and you are returned to the landing page. When you choose **Deploy**, you agree to the terms of the **Acknowledgment**. Launch Wizard validates the inputs and notifies you of any issues you must address.
8. When validation is complete, Launch Wizard deploys your AWS resources and configures your **Microsoft Remote Desktop Gateway** application. Launch Wizard provides you with status updates about the progress of the deployment on the **Deployments** page. From the **Deployments** page, you can view the list of current and previous deployments
9. When your deployment is ready, a notification informs you that your **Remote Desktop Gateway** application is successfully deployed. If you have set up an Amazon SNS notification, you are also alerted through Amazon SNS. To manage and access all of the resources related

to your application, select the deployment, and from the **Actions** dropdown list, select **Manage**.

10. When the application is deployed, you can access your EC2 instances through the Amazon EC2 console.

Deploy standalone Remote Desktop Gateway into an existing VPC (Console)

The following steps guide you through a Remote Desktop Gateway deployment with AWS Launch Wizard after you have launched it from the console.

1. When you select **Choose application** from the AWS Launch Wizard landing page, you are directed to the Choose application wizard where you are prompted to select the type of application that you want to deploy.
2. Select **Microsoft Remote Desktop Gateway**, select **Deploy into an existing VPC**, then select **Create deployment.**
3. You are prompted to enter the specifications for the new deployment. The following tabs provide information about the specification fields of the deployment model.

   General

   - **Deployment name**. Enter a unique application name for your deployment.
   - **Amazon Simple Notification Service (SNS) topic ARN — optional**. Specify an Amazon SNS topic where AWS Launch Wizard can send notifications and alerts. For more information, see the Amazon Simple Notification Service Developer Guide.
   - **Deactivate rollback on failed deployment**. By default, if a deployment fails, your provisioned resources will be deleted. You can enable this setting during deployment to prevent this behavior.
   - **Tags - optional**. Enter a key and value to assign metadata to your deployment. For help with tagging, see Tagging Your Amazon EC2 Resources.

   Network configuration

   Key pair name. Select an existing key pair from the dropdown list or create a new one. If you select **Create new key pair name**, you are directed to the Amazon EC2 console. From there, under **Network and Security**, choose **Key Pairs**. Choose **Create a new key pair**, enter a name for the key pair, and then choose **Download Key Pair**.

> ⚠ **Important**
>
> This is the only opportunity for you to save the private key file. Download it and save it in a safe place. You must provide the name of your key pair when you launch an instance and provide the corresponding private key each time that you connect to the instance. Return to the Launch Wizard console and choose the refresh button next to the **Key Pairs** dropdown list. The newly created key pair appears in the dropdown list. For more information about key pairs, see Amazon EC2 Key Pairs and Windows Instances.

VPC Settings:

- **Select Virtual Private Cloud (VPC)** option. Choose the VPC that you want to use from the dropdown list. Your VPC must be associated at least two public subnets for HA deployments.

  To add a new public subnet

  If a subnet's traffic is routed to an internet gateway, the subnet is known as a public subnet. If, however, a subnet doesn't have a route to the internet gateway, the subnet is known as a private subnet. To use an existing VPC that does not have a public subnet, you can add a new public subnet using the following steps.

  - Follow the steps in Creating a Subnet in the Amazon VPC User Guide using the existing VPC you intend to use AWS Launch Wizard.
  - To add an internet gateway to your VPC, follow the steps in Attaching an Internet Gateway in the Amazon VPC User Guide.
  - To configure your subnets to route internet traffic through the internet gateway, follow the steps in Creating a Custom Route Table in the Amazon VPC User Guide. Use IPv4 format (0.0.0.0/0) for Destination.
  - The public subnet should have the “auto-assign public IPv4 address” setting enabled. To enable this setting, follow the steps in Modifying the Public IPv4 Addressing Attribute for Your Subnet in the Amazon VPC User Guide.

- **Availability Zone (AZ) configuration:** You must choose at least two Availability Zones. The deployment will create a highly available architecture that spans these Availability Zones.
- **Allowed Remote Desktop Gateway external access CIDR:** You must specify a CIDR block for allowing external RDP access to the Remote Desktop Gateways on TCP port 3389.

Microsoft Remote Desktop Gateway configuration

Parameter label (name)	Default value	Description
Number of RDGW hosts	1	Enter the number of Remote Desktop Gateway hosts to create.
Admin user name	StackAdmin	User name for the new local administrator account.
Admin password	***Requires input***	Password for the administr ative account. Must be at least 8 characters containin g letters, numbers, and symbols.

4. When you are satisfied with your infrastructure selections, select **Next**. If you don't want to complete the configuration, select **Cancel**. When you select **Cancel**, all of the selections on the specification page are lost and you are returned to the landing page. To go to the previous screen, select **Previous**.
5. After configuring your application, you are prompted to define the infrastructure requirements for the new deployment on the **Define infrastructure requirements** page. The following tabs provide information about the input fields.

   Compute

   - **Infrastructure requirements based on instance type**. You can choose to select your instances, or to use AWS recommended resources. If you choose to use AWS recommended resources, you have the option of defining your performance needs. If no selections are made, default values are assigned.

- **Number of instance cores**. Choose the number of CPU cores for your infrastructure. The default value assigned is 4.
- **Network performance**. Choose your preferred network performance in Gbps.
- **Memory (GB)**. Choose the amount of RAM that you want to attach to your EC2 instances. The default value assigned is 4 GB.
- **Recommended resources**. Launch Wizard displays the system-recommended resources based on your infrastructure selections. If you want to change the recommended resources, select different infrastructure requirements.
- **Infrastructure requirements based on instance type**. You can choose to select your instance or use AWS recommended resources. If no selections are made, default values are assigned.
- **Instance type**. Select your preferred instance type from the dropdown list.

6. When you are satisfied with your infrastructure selections, select **Next**. If you don't want to complete the configuration, select **Cancel**. When you select **Cancel**, all of the selections on the specification page are lost and you are returned to the landing page. To go to the previous screen, select **Previous**.
7. On the **Review and deploy** page, review your configuration details. If you want to make changes, select **Previous**. To stop, select **Cancel**. When you select **Cancel**, all of the selections on the specification page are lost and you are returned to the landing page. When you choose **Deploy**, you agree to the terms of the **Acknowledgment**. Launch Wizard validates the inputs and notifies you of any issues you must address.
8. When validation is complete, Launch Wizard deploys your AWS resources and configures your **Microsoft Remote Desktop Gateway** application. Launch Wizard provides you with status updates about the progress of the deployment on the **Deployments** page. From the **Deployments** page, you can view the list of current and previous deployments
9. When your deployment is ready, a notification informs you that your **Remote Desktop Gateway** application is successfully deployed. If you have set up an Amazon SNS notification, you are also alerted through Amazon SNS. You can manage and access all of the resources related to your application by selecting the deployment, and then selecting **Manage** from the **Actions** dropdown list.
10. When the application is deployed, you can access your EC2 instances through the Amazon EC2 console.

Deploy Remote Desktop Gateway to a new or existing VPC (AWS CLI)

You can use the AWS Launch Wizard CreateDeployment API operation to deploy Remote Desktop Gateway. To create a deployment, you must provide values for various *specifications*. Specifications are a collection of settings that define how your deployment should be created and configured. A workload will have one or more deployment patterns with differing required and optional specifications.

If you want to use the **Clone deployment** action on your deployment, you must create your deployment using the Launch Wizard console.

Prerequisites for deploying Remote Desktop Gateway with the AWS CLI

Before deploying Remote Desktop Gateway with the AWS CLI, ensure you have met the following prerequisites:

- Install and configure the AWS CLI. For more information, see Install or update to the latest version of the AWS CLI.
- Complete the steps in the previous section titled **Set up**. Some deployment patterns have requirements that must be met for a deployment to be successful.

Create a Remote Desktop Gateway deployment with the AWS CLI

You can create a deployment for your Remote Desktop Gateway application using the `CreateDeployment` Launch Wizard API operation.

To create a deployment for Remote Desktop Gateway using the AWS CLI

1. List the available workload names using the ListWorkloads Launch Wizard API operation.

   The following example shows listing the available workloads:

   ```
   aws launchwizard list-workloads --region us-east-1
   {
       "workloads": [
           {
               "displayName": "Remote Desktop Gateway",
               "workloadName": "RDGW"
   ```

```
        },
        {
            "displayName": "MS SQL Server",
            "workloadName": "SQL"
        },
        {
            "displayName": "SAP",
            "workloadName": "SAP"
        },
        {
            "displayName": "Microsoft Active Directory",
            "workloadName": "MicrosoftActiveDirectory"
        }
        ...
    ]
}
```

2. Specify the desired workload name with the ListWorkloadDeploymentPatterns operation to describe the supported values for the deployment pattern names.

   The following example lists the available workload patterns for a given workload:

```
aws launch-wizard list-workload-deployment-patterns --workload-name RDGW --
region us-east-1
{
    "workloadDeploymentPatterns": [
        {
            "deploymentPatternName": "RDGWExistingVpc",
            "description": "Example description.",
            "displayName": "ExampleDisplayName",
            "status": "ACTIVE",
            "workloadName": "RDGW",
            "workloadVersionName": "2024-05-03-00-00-00"
        },
        ...
    ]
}
```

3. Use the workload and deployment pattern names you discovered with the GetWorkloadDeploymentPattern operation to list the specification details.

   The following example lists the workload specifications of a given workload and deployment pattern:

```
aws launchwizard get-workload-deployment-pattern --workload-name RDGW --deployment-
pattern-name RDGWExistingVpc --region us-east-1
{
    "workloadDeploymentPattern": {
        "deploymentPatternName": "RDGWExistingVpc",
        "description": "Example description.",
        "displayName": "ExampleDisplayName",
        "specifications": [
            {
                "description": "Enter an SNS topic for AWS Launch Wizard to send
 notifications and alerts.",
                "name": "AWS:LaunchWizard:TopicArn",
                "required": "No"
            },
            {
                "description": "When a deployment fails, your provisioned resources
 will be deleted/rolled back by default. If deactivated, the provisioned resources
 will be deleted when you delete your deployment from the Launch Wizard console.",
                "name": "AWS:LaunchWizard:DisableRollbackFlag",
                "required": "No"
            },
            {
                "allowedValues": [
                    "true",
                    "false"
                ],
                "description": "Cloud Watch Application Insights monitoring",
                "name": "SetupAppInsightsMonitoring",
                "required": "Yes"
            },
            ...
        ]
    }
}
```

4. With the workload specifications retrieved, you must provide values for any specification `name` with a `required` value of `Yes`. You can also provide any optional specifications you require for your deployment. We recommend that you pass inputs to the `specifications` parameter for your deployment as a file for easier usage.

   Your JSON file's format should resemble the following:

```
{
  "ExampleName1": "ExampleValue1",
  "ExampleName2": "ExampleValue2",
  "ExampleName3": "ExampleValue3"
}
```

5. With the specifications file created, you can create a deployment for your chosen workload and deployment pattern.

   The following example creates a deployment with specifications defined in a file:

   ```
   aws launch-wizard create-deployment --workload-name RDGW --deployment-pattern-
   name RDGWExistingVpc --name ExampleDeploymentName --region us-east-1 --
   specifications file://specifications.json
   ```

Post-deployment steps

The following are the recommended post-deployment steps for Remote Desktop Gateway on AWS.

Topics

- Complete the configuration of your AWS environment
- Install the root certificate
- Configure the Remote Desktop Connection Client
- Run Windows Updates

Complete the configuration of your AWS environment

After AWS Launch Wizard finishes the application deployment, follow these steps:

1. Create security groups for your Windows instances that will be located in private VPC subnets. Create an ingress rule permitting TCP port 3389 from the RD Gateway security group, CIDR range, or IP address. Associate these groups with instances as they are launched into the private subnets.
2. Make sure that your administrative clients can resolve the name for the RD Gateway endpoint (for example, `win-1a2b3c4d5e6.example.com`). You can create an A (Host) record in DNS

that maps the FQDN to the RD gateway's Elastic IP or public IP address. For testing purposes, you can configure this mapping in the local host's file on the machine.

3. Configure administrative clients with the proper configuration settings. This includes installing the root certificate from each RD Gateway server on the client machines (see the next section for instructions). When you use the CloudFormation templates, the default location for the root certificate will be `c:\servername.cer` on each RD Gateway server.
4. Modify the RD Gateway security group. Remove the ingress rule permitting TCP port 3389. Create a new ingress rule permitting TCP port 443 from your administrator's IP address.
5. Make sure that instances in private subnets are associated with a security group containing ingress rules permitting the RD Gateway server IP address to connect through TCP port 3389.
6. Configure the Remote Desktop connection for administrative clients, as described later in this section.

Install the root certificate

This Launch Wizard deployment implements a self-signed certificate on the RD gateway instances. After deployment, you must install the root certificate on your administrative clients before you configure the RDP client to connect to your RD gateway instances. The root certificate will automatically be stored as `c:\servername.cer`.

To distribute this file to administrator workstations and install it, follow these steps:

1. Open a Command Prompt window using administrative credentials.
2. Type `mmc` and press **Enter**.
3. In the Console Root window, on the **File** menu, choose **Add/Remove Snap In**.
4. In the **Add Standalone Snap-in** dialog box, choose **Certificates**, and then choose **Add**.
5. In the **Certificates snap-in** dialog box, choose **Computer account**, and then choose **Next**.
6. In the **Select Computer** dialog box, choose **Finish**.
7. In the **Add Standalone Snap-in** dialog box, choose **Close**.
8. On the **Add/Remove Snap-in** dialog box, choose **OK**.
9. In the Console Root window, expand **Certificates (Local Computer)**.
10. Under **Certificates (Local Computer)**, expand **Trusted Root Certification Authorities**.
11. Open the context (right-click) window for **Certificates**, and choose **All Tasks > Import**.
12. Navigate to the root certificate (e.g., `RDGW1.cer`) to complete the installation.

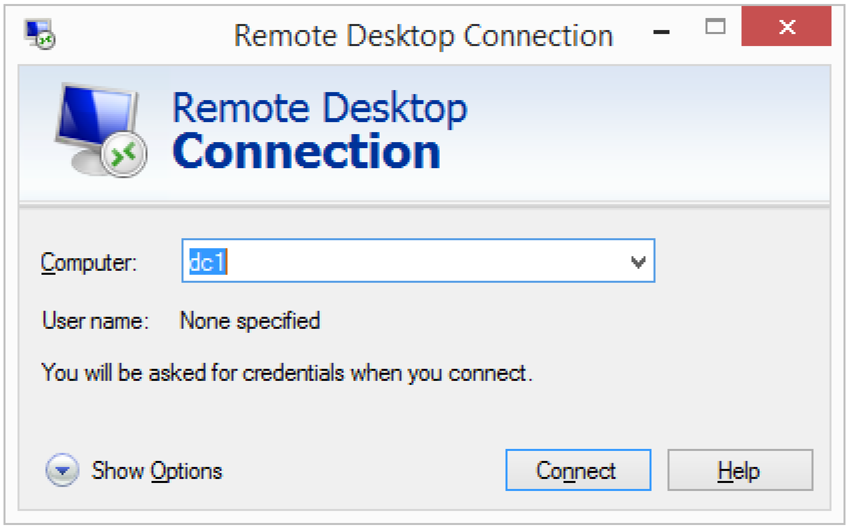
Note

The root certificate will be stored as `c:\servername.cer` on each RD gateway when deploying servers using the CloudFormation templates.

Configure the Remote Desktop Connection Client

1. Start the Remote Desktop Connection client.

2. In the computer name field, type the name or IP address of the Windows instance you want to connect to. Keep in mind that this instance needs to be reachable only from the RD gateway, not from the client machine.



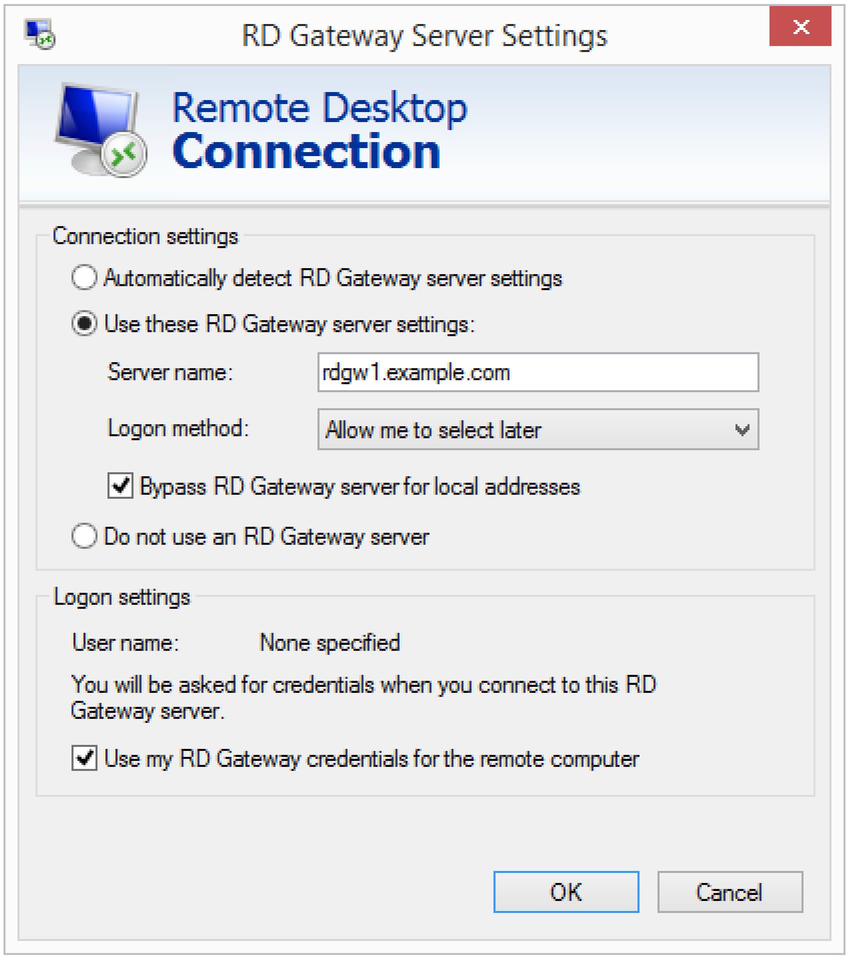
Computer: dc1

User name: None specified

You will be asked for credentials when you connect.

3. Choose Show Options. On the Advanced tab, choose Settings.

4. Choose Use these RD Gateway server settings. For server name, specify the FQDN of the RD gateway. If the RD gateway and the server you want to connect to are in the same domain, choose Use my RD Gateway credentials for the remote computer, and then choose OK.



Connection settings

○ Automatically detect RD Gateway server settings

● Use these RD Gateway server settings:

Server name: rdgw1.example.com

Logon method: Allow me to select later

☑ Bypass RD Gateway server for local addresses

○ Do not use an RD Gateway server

Logon settings

User name: None specified

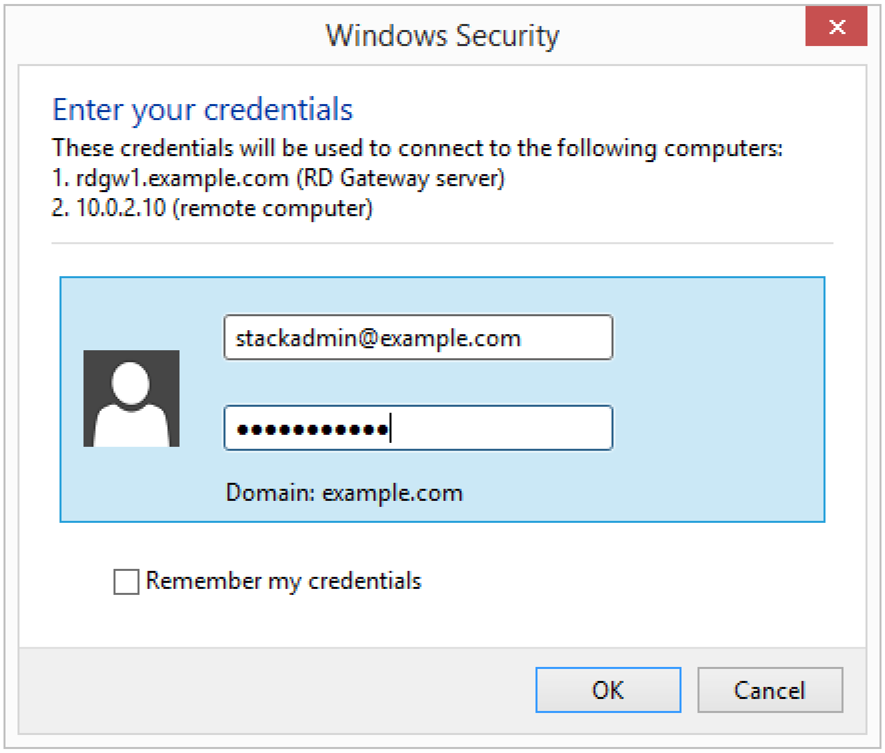
You will be asked for credentials when you connect to this RD Gateway server.

☑ Use my RD Gateway credentials for the remote computer

Note

The FQDN server name of the RD Gateway host must match the certificate and the DNS record (or local HOSTS file entry). Otherwise, the secure connection will generate warnings and might fail.

5. Enter your credentials, and then choose OK to connect to the server. You can supply the same set of credentials for the RD gateway and the destination server, as shown. If your servers are not joined to the domain, you will need to authenticate twice: once for the RD gateway and once for the destination server. If your servers aren't joined to the domain, when prompted for the RD Gateway server credentials, provide the Admin User Name and Admin Password credentials you set in when you deployed with Launch Wizard. Check the Remember my credentials box. (Otherwise, if you're connecting from a Windows computer, you'll get prompted for your credentials repeatedly, and will be blocked from entering your remote computer credentials.)



Enter your credentials

These credentials will be used to connect to the following computers:
1. rdgw1.example.com (RD Gateway server)
2. 10.0.2.10 (remote computer)

stackadmin@example.com

•••••••••••

Domain: example.com

□ Remember my credentials

Run Windows Updates

In order to ensure the deployed servers' operating systems and installed applications have the latest Microsoft updates, run Windows Update on each server.

1. Create an RDP session to the Remote Desktop Gateway server(s).
2. Open the **Settings** application.
3. Open **Update & Security**.
4. Click **Check for updates**.
5. Install any updates and reboot if necessary.

Best practices

The following are the recommended best practices for using Remote Desktop Gateway on AWS.

Topics

- The Principle of Least Privilege
- VPC Configuration

- Network Access Control Lists
- Security groups
- Initial Remote Administration Architecture
- SSL Certificates
- Connection and Resource Authorization Policies

The Principle of Least Privilege

When considering remote administrative access to your environment, it is important to follow the principle of least privilege. This principle refers to users having the fewest possible permissions necessary to perform their job functions. This helps reduce the attack surface of your environment, making it much harder for an adversary to exploit. An attack surface can be defined as the set of exploitable vulnerabilities in your environment, including the network, software, and users who are involved in the ongoing operation of the system.

Following the principle of least privilege, we recommend reducing the attack surface of your environment by exposing the absolute minimal set of ports to the network while also restricting the source network or IP address that will have access to your Amazon Elastic Compute Cloud instances.

In addition to the functionality that exists in the Windows platform, there are several AWS capabilities to help you implement the principle of least privilege, such as subnets, security groups, and trusted ingress CIDR blocks.

VPC Configuration

Amazon Virtual Private Cloud (Amazon VPC) lets you provision a private, isolated section of the AWS Cloud where you can launch AWS resources in a virtual network that you define. With Amazon VPC, you can define a virtual network topology closely resembling a traditional network that you might operate on your own premises. You control your virtual networking environment. This includes the selection of your own IP address range, creation of subnets, and configuration of route tables and network gateways.

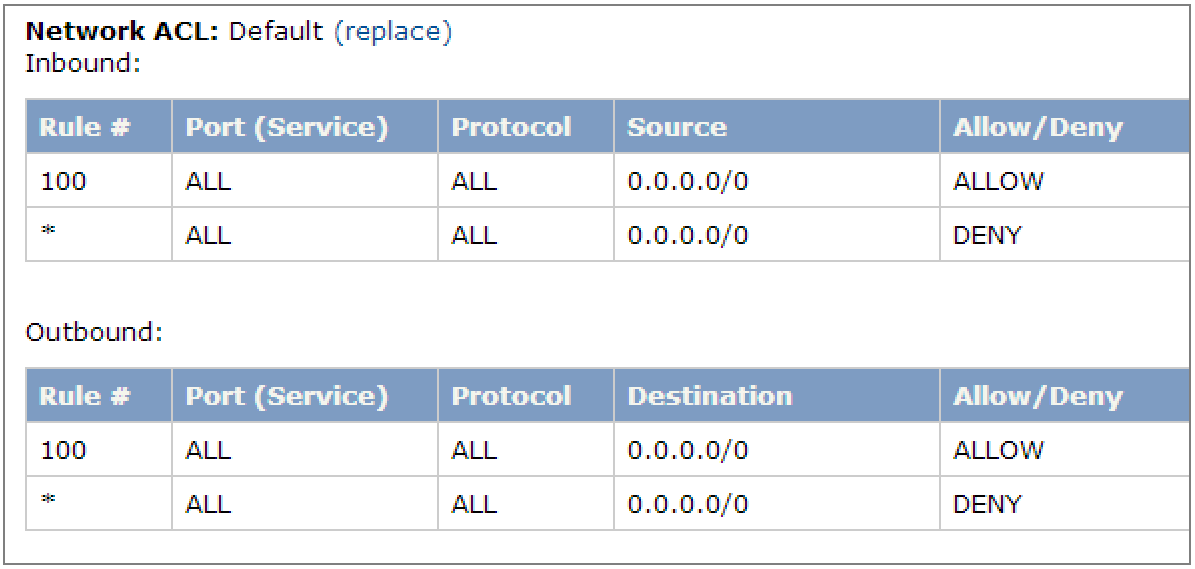
When deploying Windows architecture on the AWS Cloud, we recommend a VPC configuration that supports the following requirements:

- Place critical workloads in a minimum of two Availability Zones to provide high availability.

- Place instances into individual tiers. For example, in a Microsoft SharePoint deployment, you should have separate tiers for web servers, application servers, database servers, and domain controllers. Traffic between these groups can be controlled to adhere to the principle of least privilege.
- Deploy RD Gateways 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.

Network Access Control Lists

A network access control list (ACL) is a set of permissions that you can attach to any network subnet in a VPC to provide stateless filtering of traffic. You can use network ACLs for inbound or outbound traffic, as they provide an effective way to place a CIDR block or individual IP addresses on a deny list. These ACLs can contain ordered rules to allow or deny traffic based on IP protocol, service port, or source or destination IP address. The following image shows the default ACL configuration for a VPC subnet, which is also used by this Launch Wizard deployment:



Network ACL: Default (replace)
Inbound:

Rule #	Port (Service)	Protocol	Source	Allow/Deny
100	ALL	ALL	0.0.0.0/0	ALLOW
*	ALL	ALL	0.0.0.0/0	DENY

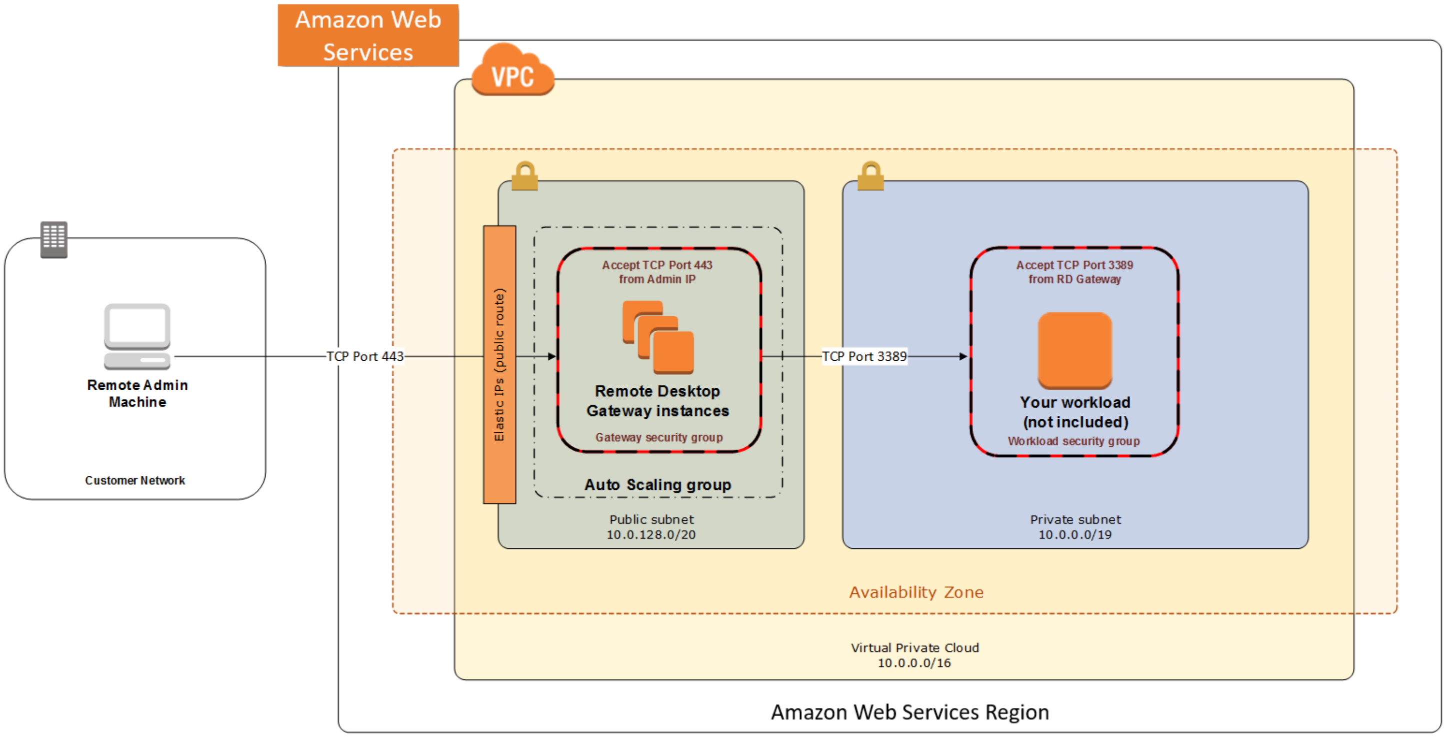
Outbound:

Rule #	Port (Service)	Protocol	Destination	Allow/Deny
100	ALL	ALL	0.0.0.0/0	ALLOW
*	ALL	ALL	0.0.0.0/0	DENY

You can keep the default network ACL configuration, or you can configure more specific rules to restrict traffic between subnets at the network level. For example, you could set a rule that would allow inbound administrative traffic on TCP port 3389 from a specific set of IP addresses. In either case, you must implement security group rules to permit access from users connecting to RD Gateways and between tiered groups of Amazon EC2 instances.

Security groups

All instances are required to belong to one or more security groups. Security groups allow you to set policies to control open ports and provide isolation between application tiers. In a VPC, every instance runs behind a stateful firewall with all ports closed by default. The security group contains rules responsible for opening inbound and outbound ports on that firewall. While security groups act as an instance-level firewall, they can also be associated with multiple instances, providing isolation between application tiers in your environment. For example, you can create a security group for all your web servers that will allow traffic on TCP port 3389, but only from members of the security group containing your RD Gateway servers. The following diagram illustrates this configuration:


Amazon Web Services
VPC
Remote Admin Machine
Customer Network
TCP Port 443
Elastic IPs (public route)
Accept TCP Port 443 from Admin IP
Remote Desktop Gateway instances
Gateway security group
Auto Scaling group
Public subnet 10.0.128.0/20
TCP Port 3389
Accept TCP Port 3389 from RD Gateway
Your workload (not included)
Workload security group
Private subnet 10.0.0.0/19
Availability Zone
Virtual Private Cloud 10.0.0.0/16
Amazon Web Services Region

Notice that inbound connections from the internet are only permitted over TCP port 443 to the RD Gateways. The RD Gateways have an Elastic IP address assigned and have direct access to the internet. The remaining Windows instances are deployed into private subnets and are assigned private IP addresses only. Security group rules allow only the RD Gateways to initiate inbound connections for remote administration to TCP port 3389 for instances in the private subnets.

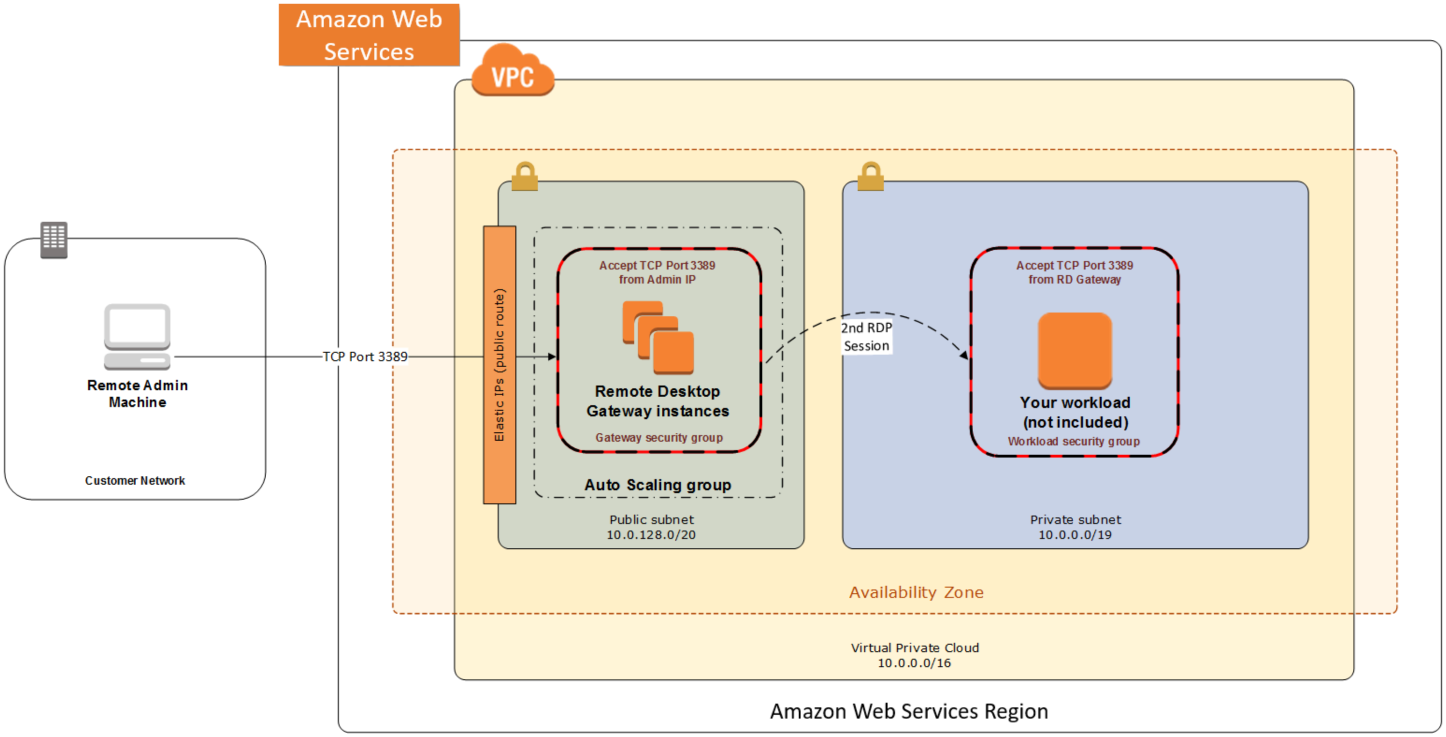
In this architecture, RDP connections are established over HTTPS to the RD Gateway and proxied to backend instances on the standard RDP TCP port 3389. This configuration helps you reduce the attack surface on your Windows instances while allowing administrators to establish connections to all your instances through a single gateway.

It's possible to provide remote administrative access to all your Windows instances through one RD Gateway, but we recommend placing gateways in each Availability Zone for redundancy. This Launch Wizard deployment implements this best practice for you.

Initial Remote Administration Architecture

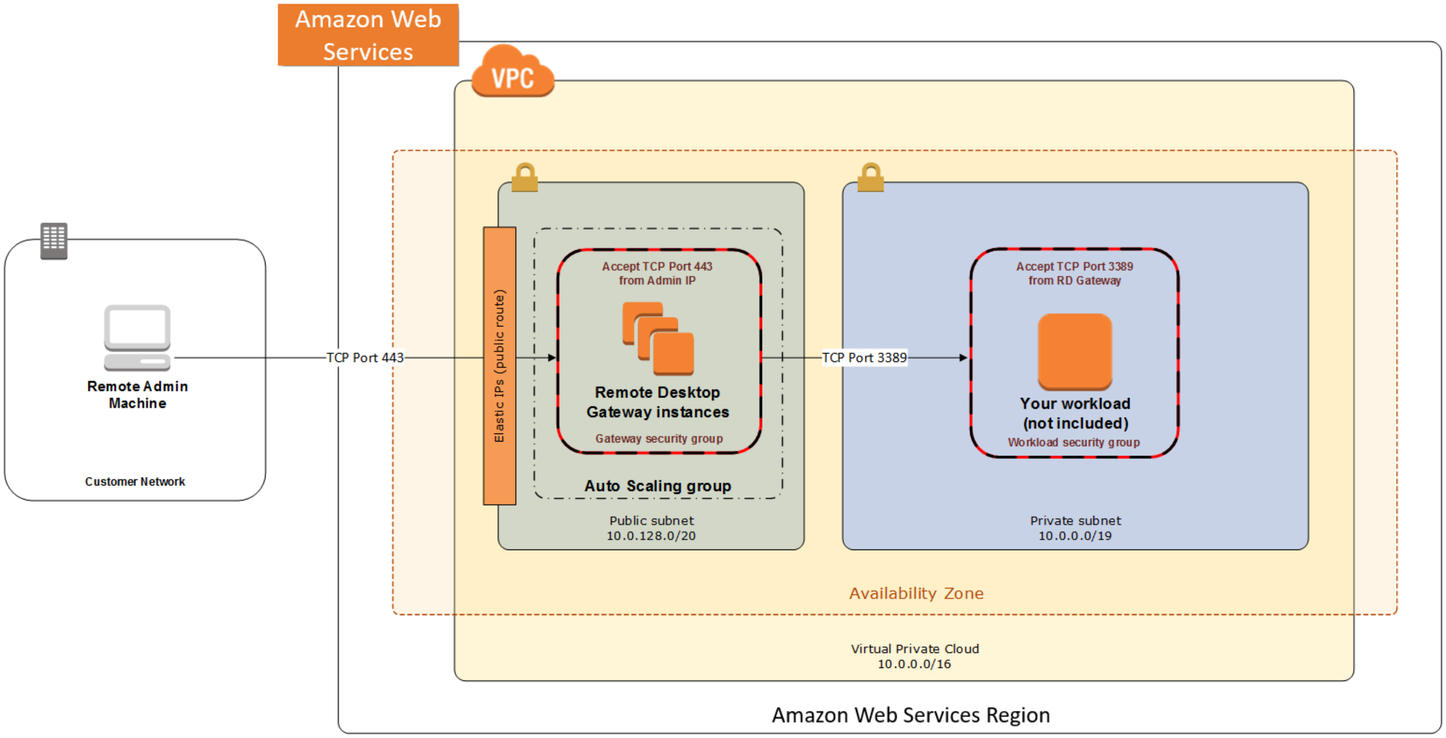
In an initial RD Gateway configuration, the servers in the public subnet will need an inbound security group rule permitting TCP port 3389 from the administrator's source IP address or subnet. Windows instances sitting behind the RD Gateway in a private subnet will be in their own isolated tier. For example, a group of web server instances in a private subnet may be associated with their own web tier security group. This security group will need an inbound rule allowing connections from the RD Gateway on TCP port 3389.

Using this architecture, an administrator can use a traditional RDP connection to an RD Gateway to configure the local server. The RD Gateway can also be used as a bastion host (jump box). This means that when an RDP connection is established to the desktop of the RD Gateway, an administrator can start a new RDP client session to initiate a connection to an instance in a private subnet, as illustrated in the following diagram:


Amazon Web Services
VPC
Remote Admin Machine
Customer Network
TCP Port 3389
Elastic IPs (public route)
Accept TCP Port 3389 from Admin IP
Remote Desktop Gateway instances
Gateway security group
Auto Scaling group
Public subnet 10.0.128.0/20
2nd RDP Session
Accept TCP Port 3389 from RD Gateway
Your workload (not included)
Workload security group
Private subnet 10.0.0.0/19
Availability Zone
Virtual Private Cloud 10.0.0.0/16
Amazon Web Services Region

Although this architecture works well for initial administration, it is not recommended for the long term. To further secure connections and reduce the number of RDP sessions required to administer the servers in the private subnets, the inbound rule should be changed to permit TCP port 443. The RD Gateway service should be installed and configured with an SSL certificate and Remote Desktop Connection Authorization Policies (RD CAP).

This Launch Wizard deployment sets up a standard TCP port 3389 connection from the administrator's IP address. You must follow the post-deployment steps to modify the security group for RD Gateway to use a single inbound rule permitting TCP port 443. This modification will allow a Transport Layer Security (TLS) encrypted RDP connection to be proxied through the gateway over TCP port 443 directly to one or more Windows instances in private subnets on TCP port 3389. This configuration increases the security of the connection and also prevents the need to initiate an RDP session to the desktop of the RD Gateway. The following diagram illustrates this configuration:


Amazon Web Services
VPC
Remote Admin Machine
Customer Network
TCP Port 443
Elastic IPs (public route)
Accept TCP Port 443 from Admin IP
Remote Desktop Gateway instances
Gateway security group
Auto Scaling group
Public subnet 10.0.128.0/20
TCP Port 3389
Accept TCP Port 3389 from RD Gateway
Your workload (not included)
Workload security group
Private subnet 10.0.0.0/19
Availability Zone
Virtual Private Cloud 10.0.0.0/16
Amazon Web Services Region

SSL Certificates

The RD Gateway role uses Transport Layer Security (TLS) to encrypt communications over the internet between administrators and gateway servers. To support TLS, a valid X.509 SSL

certificate must be installed on each RD Gateway. Certificates can be acquired in a number of ways, including:

- Your own PKI infrastructure, such as a Microsoft Enterprise Certificate Authority (CA)
- Certificates issued by a public CA, such as Verisign or Digicert
- Self-signed certificates

For smaller test environments, implementing a self-signed certificate is a straightforward process that helps you get up and running quickly. This Launch Wizard deployment automatically generates a self-signed certificate for RD Gateway.

However, if you have a large number of varying administrative devices that need to establish a connection to your gateways, we recommend using a public certificate.

For an RDP client to establish a secure connection with an RD Gateway, the following certificate and DNS requirements must be met:

- The issuing CA of the certificate installed on the gateway must be trusted by the RDP client. For example, the root CA certificate must be installed in the client machine's Trusted Root Certification Authorities store.
- The subject name used on the certificate installed on the gateway must match the DNS name used by the client to connect to the server; for example, rdgw1.example.com.
- The client must be able to resolve the hostname (for example, rdgw1.example.com) to the Elastic IP address of the RD Gateway. This will require a Host (A) record in DNS.

There are various considerations when choosing the right CA to obtain an SSL certificate. For example, a public certificate may be ideal, because the issuing CA will be widely trusted by the majority of client devices that need to connect to your gateways. However, you may want to use your own PKI infrastructure to ensure that only the machines that are part of your organization will trust the issuing CA.

Connection and Resource Authorization Policies

Users must meet specific requirements to connect to RD Gateway instances:

- *Connection Authorization Policies* – Remote Desktop Connection Authorization Policies (RD CAPs) allow you to specify who can connect to an RD Gateway instance. For example, you can select a group of users from your domain, such as Domain Admins.

- *Resource Authorization Policies* – Remote Desktop Resource Authorization Policies (RD RAPs) allow you to specify the internal Windows instances that remote users can connect to through an RD Gateway instance. For example, you can choose specific computers joined to a domain, which administrators can connect to through the RD Gateway.

This Launch Wizard deployment automatically sets up Connection and Resource Authorization Policies.

Troubleshoot AWS Launch Wizard for Remote Desktop Gateway

Each application in your account in the same AWS Region can be uniquely identified by the application name specified at the time of a deployment. The application name can be used to view the details related to the application launch.

Contents

Launch Wizard provisioning events

Launch Wizard captures events from CloudFormation to track the status of an ongoing application deployment. If an application deployment fails, you can access the CloudFormation console to view the deployment events for this application by selecting **Deployments** from the navigation pane. A failed event shows a status of **Failed** along with a failure message.

CloudFormation stack

Launch Wizard uses CloudFormation to provision the infrastructure resources of an application. You can view the status of these CloudFormation stacks, and if any of the stacks fail, you can view the cause of the failure. CloudFormation stacks can be found in your account using the CloudFormation describe-stacks API or by accessing the stack in the CloudFormation console. The following can be used with the `describe-stacks` API for the `--stack-name` argument:

- **Application resources**

  `LaunchWizard-`*`APPLICATION_NAME`*. This stack also has nested stacks for VPC and the RDGW node.

Application launch quotas

Launch Wizard allows three active applications with the status of `in progress` at one time. The combined maximum amount of `in progress` and `completed` active applications is 25 for any given application type. If you want to increase this limit, contact Support.

Enable termination protection

If you encounter errors when you deploy Remote Desktop Gateway with Launch Wizard, and the log information provided by Launch Wizard or CloudFormation is not sufficient to determine your issue, you must connect to the instance within the Amazon EC2 Auto Scaling group to determine the cause of the failure. When you connect to an instance to troubleshoot deployment failures, a common cause is the deployment scripts failing on the operating system. The following error messages in CloudFormation can indicate that the deployment scripts failed:

-
  ```
  Received 1 FAILURE signal(s) out of 1. Unable to satisfy 100%
   MinSuccessfulInstancesPercent requirement
  ```

-
  ```
  WaitCondition received failed message: ‘Error: Failed in function <script function
   name>. Return code 1 , warnings: <any warnings>’ for uniqueId: <Resource/wait
   condition name>
  ```

-
  ```
  <Resource name> timed out. Failed to receive 1 resource signal(s) within the
   specified duration
  ```

-
  ```
  Unparsable WaitCondition data
  ```

You can only connect to an EC2 instance if it is not terminated. Launch Wizard terminates instances on stack creation failure by default. You can enable the **Deactivate rollback on failed deployment** setting during deployment to prevent this behavior. If the setting was not enabled, you can still prevent the instance from getting terminated by updating the termination settings of that instance from the EC2 console before the CloudFormation stack gets rolled back.

Note

When you enable **Deactivate rollback on failed deployment**, you continue to incur AWS charges for the stack. Ensure that you delete the stack when you finish troubleshooting.

To find the EC2 instances from the Launch Wizard deployment

1. Access the CloudFormation console at https://console.aws.amazon.com/cloudformation.
2. Choose the CloudFormation stack of the Launch Wizard deployment, and choose the **Resources tab**.
3. Choose the resource with type **AWS::AutoScaling::AutoScalingGroup**.
4. Select the **instance management** tab. This page will have a link to the EC2 console, which lists the instances in the Launch Wizard deployment.

You can update the termination settings to disable termination of the instances from the EC2 console. From the **Instances** page, select an instance and choose **Action** > **Instance Settings** > **Change Termination Protection**. Then choose **Yes, Enable**.

After you have determined the root cause, disable the termination protection before you delete the deployment in Launch Wizard.

Errors

Your requested instance type is not supported in your requested Availability Zone

- **Cause:** This failure might occur during the launch of your RD Gateway instances.
- **Solution:** You must choose a different Availability Zone and retry the deployment from the initial page of the Launch Wizard console.

EC2 instance stabilization error

- **Cause:** Failure can occur if an EC2 instance fails to stabilize. When this happens, the EC2 instance is unable to communicate to the CloudFormation service to signal completions, resulting in `WaitCondition` errors.
- **Solution:** `WaitCondition` errors are often transient EC2 failures and retrying the deployment may succeed. For additional assistance, contact Support.

Permission errors

- **Cause:** Insufficient IAM permissions could be the cause of various failures in the RD Gateway deployment. Errors caused by insufficient permissions may occur within the EC2 instances as scripts are run during the application deployment. Other errors may return a verbose message indicating there are insufficient permissions similar to the following:

  ```
  User: arn:aws:iam::123456789098:user/test-user is not authorized to perform:
   elasticloadbalancing:CreateTargetGroup on resource: arn:aws:elasticloadbalancing:us-
  east-1:123456789098:targetgroup/myTargetGroup/*)
  ```

- **Solution:** Before deploying the Launch Wizard application, you must sign in to the AWS Management Console with IAM permissions for the resources that Launch Wizard will deploy. The *AdministratorAccess* managed policy within IAM provides sufficient permissions, although your organization may choose to use a custom policy with more restrictions.

AWS Launch Wizard for SAP

AWS Launch Wizard for SAP is a service that guides you through the sizing, configuration, and deployment of SAP applications on AWS, and follows AWS cloud application best practices.

AWS Launch Wizard reduces the time it takes to deploy SAP applications on AWS. You input your application requirements, including the database (SAP HANA or SAP ASE) settings, SAP landscape settings, and deployment details on the service console, and Launch Wizard identifies the appropriate AWS resources to deploy and run your SAP application. Launch Wizard provides an estimated cost of deployment, which allows you to modify your resources and instantly view the updated cost. When you finalize your settings, Launch Wizard provisions and configures the selected resources. It then optionally installs SAP application/database software using customer-provided software.

You can create deployments from the Launch Wizard console or AWS Launch Wizard APIs. For more information, see Get started with AWS Launch Wizard for SAP.

After you deploy an SAP application, you can access it from the Amazon EC2 console. You can manage your SAP applications with AWS Systems Manager.

Supported deployments and features of AWS Launch Wizard

Supported deployments

AWS Launch Wizard currently supports the deployment of AWS resources for the following SAP systems and patterns. SAP HANA database software and supported SAP application software are optionally installed and provided by the customer.

- **SAP HANA database on a single Amazon EC2 instance.** Deploy SAP HANA in a single-node, scale-up architecture, with up to 24TB of memory.
- **SAP NetWeaver on SAP HANA system on a single Amazon EC2 instance.** Deploy an SAP application on the same Amazon EC2 instance as your SAP HANA database.
- **SAP NetWeaver on SAP ASE database on a single Amazon EC2 instance.** Deploy an SAP application on the same Amazon EC2 instance as your SAP ASE database.
- **SAP HANA database on multiple EC2 instances.** Deploy SAP HANA in a multi-node, scale-out architecture.

- **SAP NetWeaver on SAP HANA system on multiple EC2 instances.** Deploy an SAP NetWeaver system using a distributed deployment model, which includes an ASCS/PAS server, single/multiple SAP HANA servers running SAP HANA databases, and multiple application servers.
- **SAP NetWeaver on SAP ASE system on multiple EC2 instances.** Deploy an SAP NetWeaver system using a distributed deployment model, which includes an ASCS/PAS server, multiple application servers, and single SAP ASE database server.
- **Cross-AZ SAP HANA database high availability setup.** Deploy SAP HANA with high availability configured across two Availability Zones.
- **Cross-AZ SAP NetWeaver system setup.** Deploy Amazon EC2 instances for ASCS/ERS and SAP HANA databases across two Availability Zones, and spread the deployment of application servers across them.
- **SUSE/RHEL cluster setup** For SAP HANA and NetWeaver on HANA high availability deployments, Launch Wizard for SAP configures SUSE/RHEL clustering when you provide SAP software and specify the deployment of SAP database or application software. For SAP HANA databases, clustering is enabled between the ASCS and ERS nodes.

AWS Launch Wizard provides the following features:

- Instance selection and configuration
- AWS resource selection
- Cost estimation
- Reusable infrastructure settings
- SNS notification
- Application resource groups
- AWS Data Provider for SAP
- AWS Backint Agent for SAP HANA
- Custom deployment configuration scripts
- Application software installation
- Creation of AWS Service Catalog products
- AWS Systems Manager for SAP
- AWS Regions

Instance selection and configuration

When you input the application requirements, Launch Wizard deploys the necessary AWS resources for a production-ready application. This means that you do not have to figure out how to select the right instances and configure them to run supported SAP applications.

AWS resource selection

Launch Wizard considers CPU/Memory or SAPS requirements that you provide to determine the most appropriate instance types and other resources for your SAP application. You can modify the recommended defaults.

Cost estimation

Launch Wizard provides a cost estimate for the complete deployment that is itemized for each individual resource being deployed. The estimated cost automatically updates each time you change a resource type configuration in the wizard. The provided estimates are only for general comparisons. They are based on On-Demand Instance costs. Actual costs may be lower.

Reusable infrastructure settings

You can save the settings for your AWS infrastructure for the SAP landscape to reuse when you want to deploy SAP systems that function similarly within a landscape. For example, a development configuration can be created for the first development instance, which can later be reused to deploy other development systems.

Some example scenarios for which DevOps and SAP architecture teams can create templates include:

- Organize the SAP systems within a landscape.
- Save infrastructure settings, including VPC, subnets, key pairs, and security groups to ensure that systems that must be deployed with the same settings are correctly deployed.
- Set up connectivity between the systems using the same configuration template so they can communicate with each other when security groups are created with Launch Wizard.
- Use the same GID for SAPSYS group across different configuration templates to ensure that SAP transport files systems are mounted properly.

SNS notification

You can provide an SNS topic so that Launch Wizard will send you notifications and alerts about the status of a deployment.

Application resource groups

Launch Wizard creates a resource group for all of the AWS resources created for your SAP system. You can manage the resources through the Amazon EC2 console or by using Systems Manager.

AWS Data Provider for SAP

Deploying and running the Amazon Web Services (AWS) Data Provider for SAP is a prerequisite for running SAP systems on AWS. Launch Wizard automatically deploys AWS Data Provider for SAP on every Amazon EC2 instance that it launches. AWS Data Provider for SAP is a tool that collects performance-related data from AWS services. It makes this data available to SAP applications to help monitor and improve the performance of business transactions. AWS Data Provider for SAP uses operating system, network, and storage data that is most relevant to the operation of the SAP infrastructure. Its data sources include Amazon EC2 and Amazon CloudWatch.

AWS Backint Agent for SAP HANA

Launch Wizard deploys and configures AWS Backint Agent for SAP HANA, an SAP-certified backup and restore application for SAP HANA workloads running on Amazon EC2 instances in the cloud. Launch Wizard supports the deployment and configuration of Backint Agent for single-node, multi-node, and high availability deployments for supported SAP HANA and SAP NetWeaver on SAP HANA applications.

You have the option to choose fully-managed backup or self-managed backup when deploying SAP applications using Launch Wizard for SAP workflow. Launch Wizard for SAP deploys AWS Backint agent for AWS Backup if fully-managed backup is selected or AWS Backint agent for Amazon S3 if self-managed backup is selected as your backup method.

Once the deployment is complete, you must maintain AWS Backint Agent for SAP HANA with latest releases and updated configurations. For more information, see AWS Backint Agent for SAP HANA.

Custom deployment configuration scripts

You can provide custom pre-deployment and post-deployment configuration scripts that can run on various instance tiers, such as SAP HANA Database, Primary Application Server, and Enqueue

Replication Server during the pre-deployment and post-deployment configuration phases. Launch Wizard uses a standalone component manager application (AWSTOE) to run the scripts. For more information, see Custom deployment configuration scripts.

Application software installation

Launch Wizard can install SAP application software that you have made available on Amazon S3, including SAP NetWeaver ABAP on SAP HANA and SAP ASE databases, SAP NetWeaver JAVA on SAP HANA and SAP ASE databases, SAP Solution Manager on SAP HANA and SAP ASE databases, SAP S/4HANA, and SAP BW/4HANA. For more details about which operating systems and database versions are supported for each deployment pattern, see SAP applications. For supported software versions and installation details, see Make SAP application software available for AWS Launch Wizard to deploy SAP.

Creation of AWS Service Catalog products

AWS Launch Wizard can create AWS Service Catalog products from successful deployments. The AWS Service Catalog products contain CloudFormation templates and associated application configuration scripts, which are stored in Amazon S3. You can use the AWS Service Catalog products, along with integrations offered by AWS Service Catalog, with third-party products, such as ServiceNow, Jira, or Terraform. Or, you can use the CloudFormation templates and application configuration scripts saved in Amazon S3 to deploy SAP applications that meet the requirements of organizational deployment and governance policies.

In addition to supporting deployments using CloudFormation templates, AWS Service Catalog, and multiple deployment tools supported by AWS Service Catalog, AWS Launch Wizard creates a point-in-time snapshot of the code used to deploy and configure SAP applications at the time of the deployment. You can use the code in its current form for consistent repeated deployments, or you can use the code as a baseline and update it to meet specific application requirements.

AWS Systems Manager for SAP

You can register SAP HANA databases and SAP applications based on SAP HANA database with AWS Systems Manager for SAP. It enables you to configure managed backups with AWS Backup for SAP HANA at the time of deployment with AWS Launch Wizard for SAP. These newly deployed applications have access to the management and operational capability that offered by AWS Systems Manager for SAP.

- SAP HANA single-node, SAP HANA high availability, and SAP NetWeaver on SAP HANA are supported. For more information, see Supported versions for SAP deployments.
- S/4HANA, S/4HANA Foundation, NetWeaver 7.5X, and BW/4HANA are the supported software stacks for SAP NetWeaver on SAP HANA deployments.
- This feature is available in all commercial regions where AWS Launch Wizard for SAP and AWS Systems Manager for SAP supported backup for SAP HANA with AWS Backup is available. For more information, see Supported Regions.

AWS Regions

Launch Wizard uses various AWS services during the provisioning of the application's environment. Not every workload is supported in all AWS Regions. For a current list of Regions where the workload can be provisioned, see AWS Launch Wizard workload availability.

Components

An SAP application deployed with Launch Wizard includes the following components.

SAP applications:

- **SAP HANA Database** supports the following:
  - Single instance deployment
  - Distributed instance deployment in a single Availability Zone
  - Cross-Availability Zone, high-availability deployment
- **SAP applications based on SAP NetWeaver on SAP HANA database** supports the following:
  - Single instance deployment
  - Distributed instance deployment
  - cross-Availability Zone, high-availability deployment
- **SAP applications based on SAP NetWeaver on SAP ASE database** supports the following:
  - Single instance deployment
  - Distributed instance deployment in a single Availability Zone
- **SAP Web Dispatcher** supports the following:
  - All SAP deployment patterns, including with other SAP applications

Security groups

Launch Wizard creates optional security groups to ensure that all of the systems sharing the same configuration template can communicate with each other and with systems and end users who access the SAP systems from an IP CIDR range, an external IP address, or security groups. For more information about how Launch Wizard creates security groups and how they are configured, see Security groups in AWS Launch Wizard for SAP.

SAP transport group configuration

You can create an SAP transport file system, or attach an existing transport file system that was created as part of a previous deployment with AWS Launch Wizard. Transport file systems are created with Amazon Elastic File System. For more information, see Amazon Elastic File System setup for transport directory.

Related services

The following AWS services are used when you deploy an SAP application with AWS Launch Wizard.

Services

- CloudFormation
- Amazon Virtual Private Cloud security groups
- Amazon Elastic File System
- AWS Systems Manager
- Amazon Simple Notification Service (SNS)
- Amazon Route 53
- AWS Backint Agent for SAP HANA
- AWS Task Orchestrator and Executor
- Amazon FSx for NetApp ONTAP
- Elastic Load Balancing
- AWS Systems Manager for SAP

CloudFormation

CloudFormation is a service that helps you model and set up your AWS resources, and lets you spend more time focusing on your applications that run in AWS. You create a template that describes all of the AWS resources that you want (for example, Amazon EC2 instances or Amazon RDS DB instances), and CloudFormation takes care of provisioning and configuring those resources for you. With AWS Launch Wizard for SAP, you don't need to build CloudFormation templates to deploy your application. Instead, AWS Launch Wizard combines infrastructure provisioning and application configuration (code that runs on EC2 instances to configure the application) into a unified CloudFormation template. The CloudFormation template is then invoked by AWS Launch Wizard's backend service to provision an application in your account.

Amazon Virtual Private Cloud security groups

Amazon Virtual Private Cloud security groups act as a virtual firewall for your instance to control inbound and outbound traffic. When you launch an instance in a VPC, you can assign up to five security groups to the instances. AWS Launch Wizard displays the security groups that will be assigned to the EC2 instances that run the SAP applications. This allows the components to communicate.

Amazon Elastic File System

Amazon EFS provides file storage in the AWS Cloud. With Amazon EFS, you can create a file system, mount the file system on an Amazon EC2 instance, and then read and write data to and from your file system. For more information, see Amazon Elastic File System setup for transport directory.

AWS Systems Manager

AWS Systems Manager is an AWS service that you can use to view and control your infrastructure on AWS. Using the AWS Systems Manager console, you can view operational data from multiple AWS services and automate operational tasks across your AWS resources. Systems Manager helps you maintain security and compliance by scanning your managed instances and reporting on, or taking corrective action on, any policy violations that it detects.

Amazon Simple Notification Service (SNS)

Amazon Simple Notification Service (SNS) is a highly available, durable, secure, fully managed pub/sub messaging service that provides topics for high-throughput, push-based, many-to-many messaging. Using Amazon SNS topics, your publisher systems can fan out messages to a large number of subscriber endpoints and send notifications to end users using mobile push, SMS, and email. You can use SNS topics for your Launch Wizard deployments to stay up-to-date on deployment progress. For more information, see the *Amazon Simple Notification Service Developer Guide*.

Amazon Route 53

Amazon Route 53 is a highly available and scalable Domain Name System (DNS) web service. You can use Route 53 to perform three main functions in any combination: domain registration, DNS routing, and health checking. Launch Wizard integrates with Route 53 hosted zones, which are containers for records. The records contain information about how you want to route traffic for a specific domain, such as example.com, and its subdomains (acme.example.com, zenith.example.com). There are two types of hosted zones: public and private hosted zones. We recommend that you use private hosted zones for SAP applications unless an application must be directly accessible from the internet.

AWS Backint Agent for SAP HANA

AWS Backint Agent for SAP HANA is an SAP-certified backup and restore application for SAP HANA workloads running on Amazon EC2 instances in the cloud. AWS Backint Agent runs as a standalone application that integrates with your existing workflows to back up your SAP HANA database to Amazon S3 and to restore it using SAP HANA Cockpit, SAP HANA Studio, and SQL commands. AWS Backint Agent supports full, incremental, and differential backup of SAP HANA databases. Additionally, you can back up log files and catalogs to Amazon S3. AWS Backint Agent runs on an SAP HANA database server, where backups and catalogs are transferred from the SAP HANA database to the AWS Backint Agent. The AWS Backint Agent stores your files in the S3 bucket that is specified in the agent configuration file. To restore your SAP HANA database server, SAP HANA reads the catalog files stored in your S3 bucket using the AWS Backint Agent. It then initiates a request to restore the required files from S3.

AWS Task Orchestrator and Executor

AWS Task Orchestrator and Executor is component management application used to orchestrate complex workflows, modify system configurations, and test your systems without writing code. This application uses a declarative document schema. As a standalone application it does not require additional server setup. It can run on any cloud infrastructure and on premises. AWS Launch Wizard uses this application to orchestrate the download of the pre- and post-configuration scripts, and to run them.

Amazon FSx for NetApp ONTAP

Amazon FSx for NetApp ONTAP is a fully managed service that provides highly reliable, scalable, high-performing, and feature-rich file storage built on NetApp's popular ONTAP file system. You can now deploy and operate SAP HANA on AWS with Amazon FSx for NetApp ONTAP. For more information, see Amazon FSx for NetApp ONTAP.

Elastic Load Balancing

Elastic Load Balancing can be deployed as an optional component to load balance internet or intranet traffic between one or more SAP Web Dispatcher instances. Launch Wizard for SAP supports both Application Load Balancer and Network Load Balancer resources. For more information, see What is Elastic Load Balancing? in the Elastic Load Balancing User Guide.

AWS Systems Manager for SAP

AWS Systems Manager for SAP is a secure end-to-end management solution for resources on AWS. It provides automation capabilities to help you manage and operate your SAP applications on AWS more efficiently with features such as as managed backups with AWS Backup for SAP HANA and graceful start/stop of SAP HANA.

Supported versions for SAP deployments

Topics

- Operating systems
- Databases
- SAP applications

Operating systems

The following table provides details of the operating systems supported by Launch Wizard for SAP deployments.

Operating system	Supported deployment patterns
Red Hat Enterprise Linux (RHEL)* 8.4, 8.6, 8.8, 9.0, 9.2, and 9.4	All

Operating system	Supported deployment patterns
SUSE Linux Enterprise Server for SAP Applications 12 SP5, 15 SP3, 15 SP4, 15 SP5, and 15 SP6	All
SUSE Linux Enterprise Server 12 SP5, 15 SP5, and 15 SP6	All, *except high availability patterns*
Bring Your Own Subscription Amazon Machine Image	All

**RHEL is available with high availability and update services on AWS Marketplace.*

Note

Operating system versions are supported on the basis of SAP component types. For example, ASCS and ERS components for high availability are supported on SUSE Linux Enterprise Server for SAP Applications and Red Hat Enterprise Linux for SAP Solutions.

Databases

The following table provides details of the database versions supported by Launch Wizard for SAP deployments.

Database	Versions	Service Pack Stack
SAP HANA	2.0	SP05 Rev59, SP06, SP07, and SP08
SAP ASE	16	SP4 PL04

For more information on the supported operating systems for SAP service pack stacks, see SAP Note 2235581 (requires access to the SAP portal).

SAP applications

The following table provides details of SAP applications supported by Launch Wizard for SAP deployments.

For more information on the supported operating systems for SAP service pack stacks, see SAP Product Availability Matrix (requires access to the SAP portal). The versions in the following tables link to the relevant sections of the SAP Product Availability Matrix.

Database support

- Applications supported with SAP HANA database
- Applications supported with SAP ASE database
- Supported versions of SAP Web Dispatcher

Applications supported with SAP HANA database

Applications	Versions
SAP NetWeaver on ABAP	750 and 752
SAP BW4/HANA	2.0, 2021 and 2023
SAP S4/HANA	1909, 2020, 2021, 2022, 2023
SAP S4/HANA Foundation	2021, 2022, 2023
SAP Solution Manager	7.2
SAP NetWeaver on JAVA	750

Applications supported with SAP ASE database

Applications	Versions
SAP NetWeaver on ABAP	750 and 752
SAP NetWeaver on JAVA	750

Applications	Versions
SAP Solution Manager	7.2

Supported versions of SAP Web Dispatcher

Launch Wizard for SAP supports SAP Web Dispatcher version 7.93. SAP Web Dispatcher is downward compatible however as the newest version can be used with all older backend systems. For more information, see SAP Note 908097 in the SAP documentation.

How AWS Launch Wizard for SAP works

AWS Launch Wizard provisions and configures the infrastructure required to run SAP HANA database and SAP NetWeaver based SAP applications on SAP HANA or SAP ASE database on AWS. You select the SAP deployment pattern and provide the specifications, such as operating system, instance size, and vCPU/memory. Or, Launch Wizard can make these selections for you according to SAP Standard Application Benchmarks. You have the option to manually choose the instance. Based on your selections, Launch Wizard automatically provisions the necessary AWS resources in the cloud.

Launch Wizard recommends Amazon EC2 instances by evaluating the SAP Standard Application Benchmarks or vCPU/memory requirements against the performance of Amazon EC2 instances supported by AWS. When new EC2 instances are released and certified for SAP, the sizing feature of Launch Wizard will take them into consideration when proposing recommendations.

Launch Wizard maintains a mapping rule engine built on the list of certified EC2 instances that are supported by SAP. When you enter your vCPU/memory or SAPS requirements, Launch Wizard recommends an Amazon EC2 instance that is certified for SAP workloads and offers performance that is no less than your input requirements. For certain workloads, such as SAP HANA in a production environment, Launch Wizard recommends instances based on the official SAP recommendations for SAP HANA database workloads. For workloads in a non-production environment, Launch Wizard recommends Amazon EC2 instances that meet SAP recommended requirements; however, the recommended instances are not enforced. You can change the instance types after deployment, or you can override the recommendation by making manual selections.

In addition to launching instances based on the SAP system information that you provide, such as SAP System Number and SAP System Identifier (SAP SID), Launch Wizard performs the following operations:

- Configures the operating system
- Configures hostname
- Attaches security groups so that the systems in the cluster that use the same configuration template, and also external systems, can communicate with the SAP systems that will be deployed on these instances.

Launch Wizard provides an estimated cost of deployment. You can modify your resources and instantly view an updated cost assessment. After you approve the deployment, Launch Wizard validates the inputs and flags inconsistencies. After you resolve the inconsistencies, Launch Wizard provisions and configures the resources. The result is a ready-to-use SAP application.

Launch Wizard creates a CloudFormation stack according to your infrastructure needs. For more information, see Working With Stacks in the *AWS CloudFormation User Guide*.

AWS Launch Wizard implements SAP deployments as follows.

Deployment aspects

- Storage for SAP systems
- Amazon Elastic File System setup for transport directory
- Amazon Elastic File System setup for SAP Central Services instances configured for high availability
- Bring your own image (BYOI)
- Specify private IP address
- Configuration settings
- Custom deployment configuration scripts
- Manual cleanup activities
- Default Quotas
- AWS Regions and Endpoints

Storage for SAP systems

Storage capacity and performance are key aspects of any SAP system installation. Launch Wizard provides storage type options for the SAP NetWeaver application tier, SAP HANA database tiers, and SAP ASE database tiers.

Amazon Elastic Block Store (Amazon EBS) volumes are included in the architecture to provide durable, high-performance storage. Amazon EBS volumes are network-attached disk storage, which you can create and attach to EC2 instances. When attached, you can create a file system on top of these volumes, run a database, or use them in any way that you would use a block device. Amazon EBS volumes are placed in a specific Availability Zone, where they are automatically replicated to protect you from the failure of a single component.

General Purpose EBS Volumes offer storage for a broad range of workloads. These volumes deliver single-digit millisecond latencies and the ability to burst to 3,000 IOPS for extended periods of time. Between a minimum of 100 IOPS (at 33.33 GiB and below) and a maximum of 16,000 IOPS (at 5,334 GiB and above), baseline performance scales linearly at 3 IOPS per GiB of volume size.

Provisioned IOPS Amazon EBS volumes offer storage with consistent and low-latency performance. They are backed by solid state drives (SSDs) and designed for applications with I/O intensive workloads, such as databases. Amazon EBS-optimized instances, such as the R4 instance type, deliver dedicated throughput between Amazon EC2 and Amazon EBS.

By default, Launch Wizard deploys Amazon EBS volumes for the SAP HANA database that meet the storage KPIs for SAP as listed in Storage Configuration for SAP HANA.

For NetWeaver database stacks, you can choose between a `gp2`, `gp3`, `io1`, or `io2` volume for the `usr/sap/`**`SAPSID`** and `/sapmnt` (for non-HA deployment architectures) file systems, whereas other configurations are deployed with `gp3` volumes. The `gp3` volumes are used by default.

Launch Wizard also supports the use of Amazon FSx for NetApp ONTAP for SAP HANA databases. FSx for ONTAP file systems can be used for `data`, `log`, and `shared` (`hana-shared` and `usr-sap`) file systems. For more information, see SAP HANA on AWS with Amazon FSx for NetApp ONTAP.

In an SAP landscape, development occurs in the development system and is then imported into the QA and follow-on systems. For this import to occur successfully, a shared file system is required for SAP systems in the landscape. Amazon EFS is used to create the SAP Transport file system that is shared between multiple SAP systems in the landscape.

Amazon Elastic File System setup for transport directory

The SAP transport directory is a shared file system between SAP systems (for example, Development, Quality, and Production) that are part of the same SAP Transport Domain for releasing and importing SAP transports. To avoid a single point of failure, Launch Wizard creates a file system with Amazon Elastic File System or reuses existing file systems. It mounts the file systems on the SAP systems that you select based on the role of the system. The transport file system is mounted on all of the applications servers included in the deployment.

When systems within the same SAP Transport Domain are created in one VPC and need to be attached to SAP systems in other VPCs (for example, if Development and Quality are deployed in a VPC tagged as Non_Prod, and Production is deployed in a VPC tagged as Prod), a prerequisite for using VPC Peering/Transit Gateway is that you must enable the VPCs to be able to communicate. This allows Launch Wizard to attach the transport directory created in one VPC to instance(s) in other VPCs using a mount target in the same Availability Zone or other Availability Zones, as applicable. If the VPCs are not permitted to communicate, then the deployment will fail when it attempts to mount the transport file system created in one VPC to systems in another VPC.

Note

When a transport files system is created with Amazon Elastic File System, Launch Wizard considers it a shared resource and will not delete it when you delete the deployment or if the deployment is rolled back.

Amazon Elastic File System setup for SAP Central Services instances configured for high availability

The SAP Central Services instances that make up a NetWeaver high availability deployment, ABAP Central Server (ASCS) and Enqueue Replication Server (ERS) instances, must contain the following file systems to be highly available: `/sapmnt`, `/usr/sap/`**`<SAPSID>`**`/ASCS`**`<XX>`**, and `/usr/sap/`**`<SAPSID>`**`/ERS`**`<XX>`**. These file systems are built with Amazon EFS to avoid a single point of failure for the SAP system. Launch Wizard creates these file systems for the NetWeaver high availability pattern using a single Amazon Elastic File System.

The following table contains information about how a single Amazon EFS is configured and mounted on an ASCS, ERS, Primary Application Server (PAS), and Additional Application Server (AAS).

EFS ID	EFS DNS name	Instance mounted on	File System name	Server mounted on
`fs-123A456B`	`fs-123A456B.ef Region>.amazons.com`	`fs-123A456B.ef Region>.amazons.com:/SAPMNT- <SAPSID>`	/sapmnt	SAP ASCS, ERS, Primary and Additional Application servers
`fs-123A456B`	`fs-123A456B.ef Region>.amazons.com`	`fs-123A456B.ef Region>.amazons.com:/ASCS- <SAPSID>`	`/usr/sap/<SAPSID>/ASCS<XX>`	SAP ASCS Server
`fs-123A456B`	`fs-123A456B.ef Region>.amazons.com`	`fs-123A456B.ef Region>.amazons.com:/ERS- <SAPSID>`	`/usr/sap/<SAPSID>/ERS<XX>`	SAP ERS Server

Bring your own image (BYOI)

You can bring your own images to deploy and configure EC2 instances for SAP with AWS Launch Wizard. During launch, in order to continue with a deployment, Launch Wizard verifies whether the operating system version selected on the front end matches the operating system version of the instance. If the versions do not match, the deployment fails with an error.

> **Note**
>
> If your own image has a volume size less than 50GB, Launch Wizard automatically creates a root volume with 50GB. If the image has a volume size bigger than 50GB, then Launch Wizard creates the root volume with that size.

When building your own image, consider the following:

- Launch Wizard configures the operating systems with OS-level parameters and utilities required by SAP

- Refer to SAP installation documents to ensure that operating system prerequisites are in place so that Launch Wizard deployments do not fail.
- Launch Wizard accesses standard repositories provided by OS vendors. Do not block access to them.
- Deployments by Launch Wizard use OS utilities and programs, such as zipper, yum, grep, printf, awk, sed, autofs, python, saptune, and tuned-profiles in the deployment script to configure SAP application and database servers. We recommend that you do not delete standard utilities.

Specify private IP address

You can specify available IP addresses that are already approved by your internal security and governance for each Amazon EC2 instance in your SAP deployment. The SAP environment is accessible as soon as the deployment is successful.

Launch Wizard, by default, auto-selects available IP addresses when a custom IP address is not provided.

When specifying a custom IP address, verify that it is within the range of the subnet of the instance that you are deploying.

Configuration settings

The following configuration settings are applied when deploying an SAP application with Launch Wizard.

Setting	Applies to
SSM Agent	All SAP systems and patterns
EBS volumes for SAP application tier	All SAP systems and patterns
EBS volumes for SAP HANA database, log and backup file systems	All SAP systems and patterns
EBS volumes for SAP ASE database, log and backup file systems	All SAP systems and patterns

Setting	Applies to
EFS volumes for `/hana/shared` and `/backup`	
EFS volumes for SAP transport file systems	All SAP systems and patterns
EFS volumes for SAP central services: `sapmnt`, `/usr/sap/<SID>/ASCS<XX>` , and `/usr/sap/<SID>/ERS<XX`	ASCS and ERS systems
OS parameters required based on the operating system chosen for database	All SAP systems and patterns
Security groups created and assigned for accessing the SAP system	All SAP systems and patterns
SSM Session Manager to remotely access the server for administrator activities	All SAP systems and patterns
Time zone settings at the OS level	All SAP systems and patterns

Custom deployment configuration scripts

You can use custom shell scripts during the pre-deployment and post-deployment configuration phases. You provide the scripts stored on Amazon S3 or locally. During provisioning, Launch Wizard installs the AWSTOE application. When there are custom scripts to run, Launch Wizard creates an AWSTOE document that downloads the scripts from the location specified and then runs the scripts. The success of the custom scripts is a customer responsibility. Check the CloudWatch log streams for detailed execution logs or failure information after the scripts are deployed.

The number of configuration scripts you can use depends on the deployment model. For SAP HANA deployments, you can use one script, which runs on all of the HANA instances (both primary and worker nodes). For NetWeaver stack on SAP HANA database, the following script limits apply:

- *NetWeaver stack on SAP HANA or SAP ASE single-instance deployment* — Because all tiers are installed on the same database instance, you can use only one script.

- *NetWeaver stack on SAP HANA distributed-instance deployment* — You can use one script per each instance tier selected, including for ASCS/SCS Server and Primary Application Server (PAS), Database (DB) Server, and Additional App Servers (AAS).
- *NetWeaver stack on SAP HANA high availability deployment* — You can use one script per each instance tier selected, including for Primary Application Server (PAS), ABAP System Central Services (ASCS) Server, Database (DB) Server, Additional App Servers (AAS), and Enqueue Replication Server (ERS).

Pre-deployment configuration scripts

Pre-deployment configuration scripts run after the instances are launched and the baseline Launch Wizard configuration tasks, such as deploying Amazon CloudWatch, Amazon EC2 Systems Manager agents, and the AWS CLI, are complete. If you want to run multiple pre-deployment configuration scripts, Launch Wizard runs them in parallel on each EC2 instance in the order in which they are specified. Pre-deployment configuration scripts can be used to perform tasks such as OS hardening or deploying security and logging software. The maximum runtime for all pre-deployment configuration scripts on a single EC2 instance is 45 minutes.

Post-deployment configuration scripts

Post-deployment configuration scripts run when Launch Wizard completes configuration tasks specific to the application on all of the instances in a deployment. Before the provisioning process completes, post-configuration scripts run on all of the specified instance tiers. Launch Wizard uses SSM and AWS Lambda to trigger running post-deployment scripts on all selected SAP instances in the order in which they are specified. They can be used to perform tasks such as installing monitoring and management software, and for updating your DNS with entries for the newly deployed SAP servers and the domains joining them. The maximum runtime for all post-deployment configuration scripts on a single instance is 2 hours.

Manual cleanup activities

If you choose to delete a deployment, or a deployment fails during the deployment phase and rolls back, Launch Wizard deletes the Amazon EC2 and Amazon EBS volumes that it launches as part of the deployment. It also removes the AWSTOE application. The following resources are considered shared resources and are created without the deletion flag.

- The Amazon Elastic File System file system that is created for the SAP transport files system `/usr/sap/trans`.

- The Amazon Elastic File System that is created for storing SAP software and media.
- The security groups that you create.

These resources must be manually verified to ensure that they are not being used by other systems in the landscape. They must then be manually deleted from either the Amazon Elastic File System or Amazon EC2 consoles, or by using APIs.

Default Quotas

To view the default quotas for AWS Launch Wizard, see AWS Launch Wizard Endpoints and Quotas.

AWS Regions and Endpoints

To view the service endpoints for AWS Launch Wizard, see AWS Launch Wizard Endpoints and Quotas.

Get started with AWS Launch Wizard for SAP

This topic contains information to help you set up your environment and deploy AWS resources with Launch Wizard, such as:

- How to create an IAM policy and attach it to your IAM user identity
- Configuration settings to apply to your environment
- How to deploy an SAP application from the AWS Management Console

Getting started topics

- Set up for AWS Launch Wizard for SAP
- Deploy an SAP application with AWS Launch Wizard
- Monitor Launch Wizard for SAP deployments
- Deploying SAP Web Dispatcher
- AWS Launch Wizard for SAP tutorials

Set up for AWS Launch Wizard for SAP

This section describes the prerequisites that you must verify to deploy an SAP application with AWS Launch Wizard.

Prerequisites

- General prerequisites
- AWS Identity and Access Management (IAM)

General prerequisites

The following general prerequisites must be met to deploy an application with Launch Wizard.

- You must create a VPC that consists of private subnet(s) in a minimum of two Availability Zones. The subnets must have outbound internet access. For more information on how to create and set up a VPC, see Getting Started with Amazon VPC in the *Amazon VPC User Guide*.
- You must create a user or role and attach the **AmazonLaunchWizardFullAccessV2** policy. See the following sections for the steps to attach the policy to the user or role.
- When using AWS Backup to back up databases on Amazon EC2 instances,
  1. You must set up the required permissions in the role `AmazonEC2RoleForLaunchWizard` for Amazon EC2 to backup and restore SAP HANA database when setting up AWS Systems Manager for SAP with fully-managed backup for SAP HANA with AWS Backup.

     The policies (that need to be attached to the role `AmazonEC2RoleForLaunchWizard`) containing these required permissions are:

     - `AWSBackupDataTransferAccess`
     - `AWSBackupRestoreAccessForSAPHANA`
     - `AWSBackupServiceRolePolicyForBackup`

     For more information, see Set up required permissions for Amazon EC2 instance for backup and restore of SAP HANA database .
  2. If you intend to assign one or more backup plans through LaunchWizard, ensure your account has the role `AWSBackupDefaultServiceRole` to ensure the HANA database is successfully assigned to the chosen backup plan and that the resulting managed backups are successful. This role is not required if you do not choose a backup plan though the LaunchWizard workflow.

- To run custom pre- and post-configuration deployment scripts, you must add the permissions listed in Add permissions to run custom pre- and post-deployment configuration scripts to the `AmazonEC2RoleForLaunchWizard` role.
- If you want to install SAP software, you must download the software from the SAP Software Download page and upload it to an Amazon S3 bucket. For steps on how to download the software and upload it to an Amazon S3 bucket, see Make SAP HANA software available for AWS Launch Wizard to deploy a HANA database.
- Depending on the operating system version you want to use for the SAP deployment, an SAP Marketplace subscription may be required. For a complete list of supported operating system versions, see Operating systems.

AWS Identity and Access Management (IAM)

Establishing the AWS Identity and Access Management (IAM) role and setting up users with the required permissions is typically performed by **an IAM administrator** for your organization. The steps are as follows:

- A one-time creation of IAM roles that Launch Wizard uses to deploy SAP systems on AWS.
- The creation of users or roles who can grant permission for Launch Wizard to deploy applications.

Launch Wizard for SAP IAM topics

- Sign up for an AWS account
- Create a user with administrative access
- One-time creation of IAM role
- Enable users to use Launch Wizard
- Add permissions to use AWS KMS keys
- Add permissions to run custom pre- and post-deployment configuration scripts
- Add permissions to save deployment artifacts to Amazon S3

Sign up for an AWS account

If you do not have an AWS account, complete the following steps to create one.

To sign up for an AWS account

1. Open https://portal.aws.amazon.com/billing/signup.
2. Follow the online instructions.

   Part of the sign-up procedure involves receiving a phone call or text message and entering a verification code on the phone keypad.

   When you sign up for an AWS account, an *AWS account root user* is created. The root user has access to all AWS services and resources in the account. As a security best practice, assign administrative access to a user, and use only the root user to perform tasks that require root user access.

AWS sends you a confirmation email after the sign-up process is complete. At any time, you can view your current account activity and manage your account by going to https://aws.amazon.com/ and choosing **My Account**.

Create a user with administrative access

After you sign up for an AWS account, secure your AWS account root user, enable AWS IAM Identity Center, and create an administrative user so that you don't use the root user for everyday tasks.

Secure your AWS account root user

1. Sign in to the AWS Management Console as the account owner by choosing **Root user** and entering your AWS account email address. On the next page, enter your password.

   For help signing in by using root user, see Signing in as the root user in the *AWS Sign-In User Guide*.
2. Turn on multi-factor authentication (MFA) for your root user.

   For instructions, see Enable a virtual MFA device for your AWS account root user (console) in the *IAM User Guide*.

Create a user with administrative access

1. Enable IAM Identity Center.

For instructions, see Enabling AWS IAM Identity Center in the *AWS IAM Identity Center User Guide*.

2. In IAM Identity Center, grant administrative access to a user.

   For a tutorial about using the IAM Identity Center directory as your identity source, see Configure user access with the default IAM Identity Center directory in the *AWS IAM Identity Center User Guide*.

Sign in as the user with administrative access

- To sign in with your IAM Identity Center user, use the sign-in URL that was sent to your email address when you created the IAM Identity Center user.

  For help signing in using an IAM Identity Center user, see Signing in to the AWS access portal in the *AWS Sign-In User Guide*.

Assign access to additional users

1. In IAM Identity Center, create a permission set that follows the best practice of applying least-privilege permissions.

   For instructions, see Create a permission set in the *AWS IAM Identity Center User Guide*.

2. Assign users to a group, and then assign single sign-on access to the group.

   For instructions, see Add groups in the *AWS IAM Identity Center User Guide*.

One-time creation of IAM role

On the **Choose Application** page of Launch Wizard, under **Permissions**, Launch Wizard displays the IAM role required for the Amazon EC2 instances created by Launch Wizard to access other AWS services on your behalf. When you select **Next**, Launch Wizard attempts to discover the IAM role in your account. If the role exists in your account, it is attached to the instance profile for the Amazon EC2 instances that Launch Wizard launches from your account. If the role does not exist, Launch Wizard attempts to create the role with the same name, `AmazonEC2RoleForLaunchWizard`.

The `AmazonEC2RoleForLaunchWizard` role is comprised of two IAM managed policies: `AmazonSSMManagedInstanceCore` and `AmazonEC2RolePolicyForLaunchWizard`. The

`AmazonEC2RoleForLaunchWizard` role is used by the instance profile for the Amazon EC2 instances that Launch Wizard launches into your account as part of the deployment.

If you want to deploy AWS Backint Agent as a backup and restore solution for your application, you must attach a policy to the `AmazonEC2RoleForLaunchWizard` so that Launch Wizard can perform Backint Agent operations on your behalf. The required policy and instructions can be found in Step 2 of the Backint Agent IAM documentation. During a deployment, Launch Wizard provides the policy as well as the steps to update the role, taking user specifications into account.

After the IAM roles are created, the IAM administrator can either continue with the deployment process or optionally delegate the application deployment process to another user, as described in the following section. At this point in the IAM set up process, the IAM administrator can exit the Launch Wizard service.

Enable users to use Launch Wizard

To deploy an SAP system with Launch Wizard, your user must have the permissions provided by the **AmazonLaunchWizardFullAccessV2** policy. The following guidance is provided for IAM administrators to provide permissions for users to access and deploy applications from Launch Wizard using the **AmazonLaunchWizardFullAccessV2** policy.

To provide access, add permissions to your users, groups, or roles:

- Users and groups in AWS IAM Identity Center:

  Create a permission set. Follow the instructions in Create a permission set in the *AWS IAM Identity Center User Guide*.
- Users managed in IAM through an identity provider:

  Create a role for identity federation. Follow the instructions in Create a role for a third-party identity provider (federation) in the *IAM User Guide*.
- IAM users:
  - Create a role that your user can assume. Follow the instructions in Create a role for an IAM user in the *IAM User Guide*.
  - (Not recommended) Attach a policy directly to a user or add a user to a user group. Follow the instructions in Adding permissions to a user (console) in the *IAM User Guide*.

⚠ **Important**

You must log in with the user or assume the role associated with this IAM policy when you use Launch Wizard.

Add permissions to use AWS KMS keys

AWS Launch Wizard uses AWS default encryption keys to encrypt Amazon EBS volumes. In addition, Launch Wizard supports the use of KMS keys created and maintained in AWS KMS. You can choose to either create new keys or use preexisting keys to encrypt your EBS volumes. You must add permissions to the KMS key policy for your key so that Launch Wizard can use your KMS key for encryption.

How to add permissions to your KMS key policy so that Launch Wizard can use your key for encryption

1. Sign in to the AWS Management Console and open the AWS Key Management Service (AWS KMS) console at https://console.aws.amazon.com/kms.
2. To change the AWS Region, use the Region selector in the upper-right corner of the page.
3. Choose **Customer managed keys** in the left navigation pane.
4. Select the alias of the KMS key that you want to use to encrypt your EBS volumes.
5. Under **Key users**, choose **Add**.
6. Select the check box next to `AmazonEC2RoleForLaunchWizard` and the role your users assume with Launch Wizard full access permissions.
7. Choose **Add**. Verify that `AmazonEC2RoleForLaunchWizard` and the user or role with Launch Wizard full access permissions appear in the **Key users** list.

Add permissions to run custom pre- and post-deployment configuration scripts

To run custom pre- and post-configuration deployment scripts, you must add the following permissions to the `AmazonEC2RoleForLaunchWizard` role. The following steps guide you through the process of adding the required permissions for using custom scripts to the `AmazonEC2RoleForLaunchWizard` role.

1. Sign in to the AWS Management Console and open the IAM console at https://console.aws.amazon.com/iam/.

2. In the navigation pane, choose Policies, Create policy.
3. On the **Create policy** page, choose **JSON**, then copy and paste the following policy into the **JSON** tab. Enter the S3 paths where your scripts are stored.

   JSON

   ```
   {
       "Version":"2012-10-17",
       "Statement": [
           {
               "Sid": "VisualEditor0",
               "Effect": "Allow",
               "Action": [
                   "s3:GetObject",
                   "s3:GetBucketLocation"
               ],
               "Resource": [
                   "arn:aws:s3:::<S3bucket1>/<S3prefix1>/<script1>",
                   "arn:aws:s3:::<S3bucket2>/<S3prefix2>/<script2>",
                   "arn:aws:s3:::<S3bucket1>",
                   "arn:aws:s3:::<S3bucket2>"
               ]
           }
       ]
   }
   ```

4. Choose **Next: Tags** and create any tags you require.
5. Choose **Next: Review** and enter a **Name** for the policy.
6. Choose **Create Policy**.
7. Verify that the correct policy is listed, and then choose **Policy actions**.
8. Choose **Attach**.
9. Search for the policy named **AmazonEC2RoleForLaunchWizard** and select the check box to the left of the policy name.
10. Choose **Attach policy**.

If the pre- or post-deployment configuration deployment scripts are expected to run additional AWS services, the permissions to use the services must also be manually added as policy to the `AmazonEC2RoleForLaunchWizard`.

Add permissions to save deployment artifacts to Amazon S3

To create AWS Service Catalog products from successful deployments, which include CloudFormation templates and application configuration scripts, you must provide access to an Amazon S3 location to save the generated artifacts.

The following steps guide you through adding the required permissions for saving deployment artifacts to Amazon S3. These permissions are required in addition the ones provided by the `AmazonLaunchWizardFullAccessV2` role. If the S3 bucket that you want to use to save deployment artifacts does not contain the prefix `launchwizard` in its name, you must perform the following steps to attach the required policy to the IAM role that will be used for performing the deployments.

1. Sign in to the AWS Management Console and open the IAM console at https://console.aws.amazon.com/iam/.
2. In the navigation pane, choose Policies, Create policy.
3. On the **Create policy** page, choose **JSON**, then copy and paste the following policy into the **JSON** tab. Enter the S3 path where you want to store your artifacts in the policy.

   JSON

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

      {
          "Sid": "SaveLaunchWizardDeploymentArtifacts",
          "Effect": "Allow",
          "Action": [
            "s3:PutObject"
          ],
          "Resource": [
              "arn:aws:s3:::${bucketName}/${bucketFolder}*"
          ]
      }
    ]
  }
```

4. Choose **Next: Tags** and create any tags you require.
5. Choose **Next: Review** and enter a **Name** for the policy.

6. Choose **Create Policy**.
7. Verify that the correct policy is listed, and then choose **Policy actions**.
8. Choose **Attach**.
9. Search for the role your users assume with Launch Wizard full access permissions and select the check box to the left of the policy name.
10. Choose **Attach policy**.

Deploy an SAP application with AWS Launch Wizard

This section contains steps for deploying an SAP application with Launch Wizard. It includes steps for various deployment paths for NetWeaver stack on SAP HANA database and SAP HANA database.

Topics

- Deploying an SAP application (Console)
- Deploying an SAP application (AWS CLI)

Deploying an SAP application (Console)

You can deploy an SAP application using the AWS Launch Wizard console.

Topics

- Access AWS Launch Wizard
- Deploy an SAP application with AWS Launch Wizard
- Clone deployment

Access AWS Launch Wizard

You can launch AWS Launch Wizard from the AWS Launch Wizard console located at https://console.aws.amazon.com/launchwizard.

Deploy an SAP application with AWS Launch Wizard

The following steps guide you through deploying an SAP application with AWS Launch Wizard after you have launched it from the console.

Create a deployment

1. From the AWS Launch Wizard landing page, choose **Create deployment**.
2. Choose **SAP**.
3. Under **Permissions**, Launch Wizard displays the AWS Identity and Access Management (IAM) roles required for Launch Wizard to access other AWS services on your behalf. For more information about these roles and setting up IAM for Launch Wizard, see Identity and Access Management for AWS Launch Wizard. Choose **Next**.

Define infrastructure

On the **Define infrastructure** page, define your deployment name and infrastructure settings.

1. Under the **General** subheading, define the following:
   - **Deployment name**. Enter a unique application name for your deployment.
   - **Description (Optional)**. Provide an optional description of your deployment.
   - **Enable rollback on failed deployment**. By default, if a deployment fails, your provisioned resources will not be rolled back/deleted. This default configuration helps you to troubleshoot errors at the resource level as you debug deployment issues. If you want your provisioned resources to be immediately deleted if a deployment fails, select the check box.
   - **Create an AWS Service Catalog product**. Select the check box to package and export CloudFormation templates and associated application configuration scripts to Amazon S3 and create an AWS Service Catalog product. You use these scripts to deploy and configure AWS infrastructure resources for SAP applications. If you select this option, the templates and scripts are saved to the specified Amazon S3 path. You can use the saved CloudFormation templates and AWS Service Catalog products for repeated deployments of the SAP applications using CloudFormation, AWS Service Catalog, and third-party applications integrated with AWS Service Catalog.
   - **Tags (Optional)**. Enter a key and value to assign metadata to your deployment. For help with tagging, see Tagging Your Amazon EC2 Resources.
2. Under the **Infrastructure – SAP landscape** subheading, configure the following infrastructure settings for your SAP landscape.

Configuration options

- Under **Configuration type**, choose whether to **Create new configuration** or **Apply saved configuration**.
- Enter the following information:
  - **Configuration name**. Enter a name or short description to identify your configuration. You can save this configuration for future use.
  - **Deployment environment**. (**Create new configuration**, only) Choose whether to deploy into a **Production** or **Non-Production** environment.

Configuration details

If you choose to create a new configuration, enter the following information.

- **Key pair name**. Choose an existing key pair from the dropdown list or select the link to create one. If you select **Create new key pair name**, you are directed to the Amazon EC2 console. From the Amazon EC2 console, under **Network and Security**, choose **Key Pairs**. Choose **Create a new key pair**, enter a name for the key pair, and then choose **Download Key Pair**.

  > ⚠ **Important**
  >
  > This is the only time that you can save the private key file, so download and save it in a safe place. You must specify the name of your key pair when you launch an instance, and provide the corresponding private key each time that you connect to the instance.

  Return to the Launch Wizard console, and choose the refresh button next to the **Key Pair name** dropdown list. The new key pair appears in the dropdown list. For more information about key pairs, see Amazon EC2 Key Pairs.
- **Virtual Private Cloud**. Choose a VPC from the dropdown list or select the **Create VPC** link. If you select **Create VPC**, you are redirected to the VPC console to create a VPC.
- **Availability Zone and private subnets**. You can deploy into one or two Availability Zones with up to two private subnets per Availability Zone. Different requirements are needed for different systems in the landscape. You must select two Availability Zones

with a required primary and optional secondary subnet for each Availability Zone. These selections are used for each deployment according to the deployment model that you selected.

From the dropdown lists, choose the **Availability Zones** within which you want to deploy your SAP systems and choose the private subnets. The private subnets must have outbound connectivity to the internet and to other AWS services, such as Amazon S3, CloudFormation, and CloudWatch Logs. They must also be able to access the Linux repositories required for instance configuration.

For high availability deployments, the following subnets must share a common route table:

- subnet 1 in Availability Zone 1 and subnet 1 in Availability Zone 2
- subnet 2 in Availability Zone 1 and subnet 2 in Availability Zone 2

To create a private subnet

- If a subnet doesn't have a route to an internet gateway, the subnet is known as a private subnet. Use the following procedure to create a private subnet. We recommend that you enable the outbound connectivity for each of your selected private subnets using a NAT gateway. To enable outbound connectivity from private subnets with public subnets, create a NAT Gateway in your chosen public subnet. Then, follow the steps in Updating Your Route Table for each of your private subnets.
  - Follow the steps in Creating a Subnet in the Amazon VPC User Guide using the existing VPC that you will use in Launch Wizard.
  - When you create a VPC, it includes a main route table by default. On the **Route Tables** page in the Amazon VPC console, you can view the main route table for a VPC by looking for **Yes** in the **Main** column. The main route table controls the routing for all subnets that are not explicitly associated with any other route table. If the main route table for your VPC has an outbound route to an internet gateway, then any subnet created using the previous step, by default, becomes a public subnet. To ensure the subnets are private, you may need to create separate route tables for your private subnets. These route tables must not contain any routes to an internet gateway. Alternatively, you can create a custom route table for your public subnet and remove the internet gateway entry from the main route table.

- **Verify Connectivity**. Select the check box to verify that your private subnets have outbound internet connectivity.
- **Security groups**. You can choose already existing security groups or Launch Wizard can create security groups that will be assigned to the EC2 instances that Launch Wizard deploys. If you choose already existing security groups, you must ensure that all of the necessary ports required to access the SAP and SAP HANA databases are open. If you choose to allow Launch Wizard to create the security groups, the security groups are created to enable the components of the cluster to communicate. Systems that are deployed with the same configuration settings can also communicate.

  If you choose an existing security group, Launch Wizard displays the security groups that will be assigned to the EC2 instances that Launch Wizard deploys. This enables the components to communicate and systems that are deployed with the same configuration settings to communicate.
- **Connectivity to external systems or users**. If you allowed Launch Wizard to create the security groups, then choose the **Connection type** and **Value** of the IP address or security groups required to access the SAP systems. These values can be a network segment from which the end users access the SAP systems, or downstream/upstream systems assigned a different security group in AWS or on premises.
- **Proxy setting**. During the launch process, the deployed Amazon EC2 instances require outbound internet access in order to:
  - Access the operating system (SUSE/RHEL) repositories.
  - Access AWS services, such as Amazon S3, CloudWatch and Systems Manager.

  An internet gateway is typically configured for outbound internet access. If you want to route internet traffic through a proxy server, enter the proxy server details. When proxy server information is provided, Launch Wizard will make the necessary environment changes to the EC2 instances during launch so that outbound internet traffic is routed through the proxy server.
  - **PROXY**. Enter the proxy server name and port, for example `http://10.0.0.140:3128` or `https://10.0.0.140.3128`.
  - **NO_PROXY**. When a proxy server is used for outbound communication, the `NO_PROXY` environment variable is used to route traffic without using the proxy for the following types of traffic:

    - local communication
    - traffic to other instances within the VPC
    - traffic to other AWS services for which VPC endpoints are created

    Enter a list of comma-separated values to denote hostnames, domain names, or a combination of both.

  We recommend that you add all AWS service endpoints (if defined) to the NO_PROXY environment variable so that a private connection between the VPC and the service endpoint can be established in the AWS VPN. For more information on AWS service endpoints, see AWS service endpoints.

  NO_PROXY is an optional parameter. If no value is entered, the following default URLs are added to the environment. Values entered for NO_PROXY at a later time are added to this list.

  ```
  NO_PROXY="localhost,127.0.0.1,169.254.169.254,.internal,{VPC_CIDR_RANGE}"
  ```

  Default NO_PROXY URL details

  - **localhost**—loopback hostname
  - **127.0.0.1**—loopback adapter IP
  - **169.254.169.254**—EC2 metadata link-local address
  - **.internal**—default DNS for the VPC
  - **{VPC_CIDR_RANGE}**—CIDR block of the VPC, for example, 10.0.0.0/24
- **Time zone**. Choose the time zone settings to configure the timezone on the instances from the dropdown list.
- **EBS encryption**. From the dropdown list, choose whether or not to enable EBS encryption for all of the EBS volumes that are created for the SAP systems. For more information, see Amazon EBS Encryption.
- **Domain name (DNS) settings (Optional)**. Select **Domain Name** or **Route 53** from the **DNS type** dropdown list.
  - If you select **Domain Name**, you have the option to enter a domain name to maintain a Fully Qualified Domain Name (FQDN) in the `/etc/hosts` file for each instance that is launched and configured by Launch Wizard.

- If you select **Route 53**, select a Route 53 hosted zone from the dropdown list. Launch Wizard will create a DNS entry for each EC2 instance launched.

  > **Note**
  >
  > Before you use a Route 53 hosted zone, verify that the hosted zone is integrated with the VPC, and that the VPC DHCP options are correctly set up.

- **SAP landscape settings**. Enter the system settings for your SAP landscape.
  - **SAP System Admin User ID**. Enter the user ID for the SAP system administrator.
  - **SAP System Admin Group ID**. Enter the group ID for SAPSYS. We recommend that you replicate this number across all of your SAP systems because SAPSYS GID must be the same between systems that are part of the transport domain.
  - **SAPINST Group ID**. Enter the group ID for the SAPINST.
- **Simple Notification Service (SNS) topic ARN (Optional)**. Specify an SNS topic where Launch Wizard can send notifications and alerts. For more information, see the Amazon Simple Notification Service Developer Guide. You can also choose **Create SNS topic** and then create one in the Amazon SNS console. After you create an SNS topic, you can enter it in the Launch Wizard SNS field.
- After you specify the infrastructure settings, choose **Next**.

Application and deployment settings

The following steps show the deployment paths for **NetWeaver stack on SAP HANA database** and **SAP HANA database**. Please follow the deployment steps for your deployment path.

Topics

NetWeaver stack on SAP HANA database

Application settings

On the **Configure application settings** page, enter your NetWeaver stack on SAP HANA database application settings.

1. **Application type**. Select **NetWeaver stack on SAP HANA database**. This configuration includes:
   - NetWeaver stack for a single instance , distributed instance, or multi-AZ for high availability (HA) deployment.
   - EC2 instances for the NetWeaver application tier
   - EC2 instances for SAP HANA database and optional SAP HANA database install
2. **General settings – SAP system**. Enter the settings for your SAP system.
   - **SAP System ID (SAPSID)**. An identifier for your system. The ID must be a three character, alphanumeric string.
   - **EBS Volume Type for NetWeaver application stack instances**. Choose which volume type to use for the NW application file system `/usr/sap/`**`SAPSID`** from the dropdown list.
   - **Transport Domain Controller**. Specify whether the SAP system will be the domain controller for the SAP landscape. If not, select the transport file system of the domain controller to be mounted.
   - **SAP Web Dispatcher**. Specify whether to deploy SAP Web Dispatcher to load balance incoming web connections for your SAP application server instances.
3. **General Settings – SAP HANA**. Enter the settings for your SAP HANA installation.
   - **SAP HANA System ID.** Enter the identifier for your SAP HANA database. The ID must be a three character, alphanumeric string.
   - **SAP HANA Instance number.** Enter the instance number to be used for the SAP HANA installation and setup. The ID must be a two-digit number.
   - **EBS Volume Type for SAP HANA**. Select the EBS volume types to use for **SAP HANA Data**, **SAP HANA Logs**, and **SAP Others** from the dropdown lists.

> **Note**
>
> gp3 volumes are not supported for HANA production databases running on Xen instances (X1, X1e, R4, and R3). When you deploy HANA databases with Xen instances after choosing **Production** as the **Deployment environment** under the **Configuration options**, gp2 volumes will be used to set up SAP HANA Data and Logs on the instances you selected for the HANA database.

- Select *Make this selection to use Amazon FSx for NetApp ONTAP for all SAP HANA database file systems, except root, backup, and media file systems.*

  Your chosen Amazon EBS volume type is used for the application layer.

4. After you enter your application settings, choose **Next**.

(Use the tab for **Single instance deployment**, **Distributed instance deployment**, or **High availability deployment**, depending on your configuration)

Single instance deployment

On the **Configure deployment model** page, enter the deployment details for a single instance deployment.

1. **Deployment details**. Launch Wizard supports single instance deployments, distributed instance deployments, and high availability deployments. Select **Single instance deployment**.
2. **ASCS, PAS, and DB on one EC2 instance**. Enter the deployment settings for your instance.
   - **Instance details**.
     - Under **Instance sizing**, choose whether to use **AWS/Marketplace/Community images** or **Bring your own images (BYOI)**.
     - **Operating System**. Select a supported operating system version for the ASCS instance. For a complete list of operating system versions supported for ASCS, see Operating systems.
     - **AMI ID**. For BYOI, select the AMI that you want to use from the dropdown.
     - **Host name**. Enter the host name for the EC2 instance.

    - **Private IP address**. Choose whether to use an **Auto-assigned (default)** IP address or a **Custom IP address**.
        - **Auto-assign (default)**. When you select this option, an IP addressed will be assigned for you. This is the default option.
        - **Private IP address**. When you select this option, you can enter a single IP address. Verify that this IP address is within the subnet range of the instance you are launching.
    - **Auto Recovery**. Auto recovery is an Amazon EC2 feature to increase instance availability. Select the check box to enable EC2 automatic recovery for the instance. For more information, see Recover Your Instance in the Amazon EC2 User Guide.
    - **SAP Web Dispatcher ID (SID)**. The SID to use for SAP Web Dispatcher. This value must be unique throughout your SAP system's landscape.
    - **SAP Web Dispatcher Admin User ID**. The user ID number for the SAP Web Dispatcher administrator user (sid-adm).
- Under **Instance sizing**, choose whether to **Use AWS recommended resources** or **Choose instance**.
    - **Use AWS recommended resources**.
        - **Infrastructure requirements**. Choose the requirements for your recommended resources from the dropdown list.
            - **Based on CPU/Memory**. If you select this option, enter the required number of vCPU **Cores** and **Memory**. Amazon EC2 supports up to 1920 logical processors. If the amount of memory required exceeds 4TB, dedicated hosts are required.
            - **SAPS (SAP Application Performance Standard)**. If you select this option, enter the **SAPS** rating for the SAP certified instance types.
    - **Choose your instance**.
        - **Instance type**. Choose the instance type from the dropdown list.
    - **Recommended Resources**. AWS Launch Wizard displays the **Estimated monthly cost of operation** based on your instance sizing selections and the EBS volumes that will be created and attached to the launched instances. This is an estimate of AWS costs to deploy additional resources and does not include any image costs, EC2 reservations, applicable taxes, or discounts.

- Under **Amazon FSx for NetApp ONTAP storage**, enter your desired volume sizes for SAP HANA data, log, and other file systems. The displayed default values are based on your selection of the instance type.

3. **Pre- and post-deployment configuration scripts — optional**

   You can run pre- and post-deployment configuration scripts during application provisioning. For more information about how Launch Wizard accesses and deploys these scripts, see Custom deployment configuration scripts.

   Pre-deployment configuration script — optional

   - **Deployment settings**. In the event of a configuration script failure or time out, select whether to ignore all failures and proceed with the deployment. If you do not select this option, when the configuration scripts fail or time out, Launch Wizard will roll back the workload and delete all of the AWS resources created by Launch Wizard. Pre-deployment configuration scripts must finish running in 45 minutes or less.
   - **Configuration script**. Choose to use a script located in Amazon S3 and enter the URL path of the script, or enter a script manually by uploading a script file. To remove the configuration script, choose **Remove script**.

   Post-deployment configuration script — optional

   - **Deployment settings**. In the event of a configuration script failure or time out, select whether to ignore all failures and proceed with the deployment. If you do not select this option, when the configuration scripts fail or time out, Launch Wizard will roll back the workload and delete all of the AWS resources created by Launch Wizard. Post-deployment configuration scripts must finish running in 2 hours or less.
   - **Configuration script**. Choose to use a script located in Amazon S3 and enter the URL path of the script, or enter a script manually by uploading a script file. To remove the configuration script, choose **Remove script**.

4. After you have entered your deployment settings, choose **Next**.

(See the **SAP software installation settings** tab.)

Distributed instance deployment

On the **Configure SAP HANA deployment model** page, enter the deployment details for a distributed instance deployment.

1. **Deployment details**. Launch Wizard supports single instance deployments, distributed instance deployments, and high availability deployments. Select **Distributed instance deployment**.
2. **ASCS/SCS Server and Primary Application Server (PAS)**. Enter the deployment settings for your instance.
   - **Instance details**.
     - Under **Instance sizing**, choose whether to use **AWS/Marketplace/Community images** or **Bring your own images (BYOI)**.
       - **Operating System**. Select a supported operating system version for the ASCS and PAS instances. For a complete list of operating system versions supported for ASCS, see Operating systems.
       - **AMI ID**. For BYOI, select the AMI that you want to use from the dropdown.
     - **Host name**. Enter the host name for the EC2 instances.
     - **Auto Recovery**. Auto recovery is an Amazon EC2 feature to increase instance availability. Select the check box to enable EC2 automatic recovery for the instance. For more information, see Recover Your Instance in the Amazon EC2 User Guide.
   - Under **Instance sizing**, choose whether to Use **AWS recommended resources** or **Choose your instance**.
     - **Use AWS recommended resources**.
       - **Infrastructure requirements**. Choose the requirements for your recommended resources from the dropdown list.
         - **Based on CPU/Memory**. If you select this option, enter the required number of vCPU **Cores** and **Memory**. Amazon EC2 supports up to 1920 logical processors. If the amount of memory required exceeds 4TB, dedicated hosts are required.
         - **SAPS (SAP Application Performance Standard)**. If you select this option, enter the **SAPS** rating for the SAP certified instance types.
     - **Choose your instance**.
       - **Instance type**. Choose the instance type from the dropdown list.

    - **Recommended Resources**. AWS Launch Wizard displays the **Estimated monthly cost of operation** based on your instance sizing selections. This is an estimate of AWS costs to deploy additional resources and does not include any applicable taxes or discounts.

3. **Settings for Database (DB) Server**. Enter the deployment settings for your instance.

    - **Instance details**.
        - Under **Instance sizing**, choose whether to use **AWS/Marketplace/Community images** or **Bring your own images (BYOI)**.
            - **Operating System**. Select a supported operating system version for the ASCS and PAS instances. For a complete list of operating system versions supported for ASCS, see Operating systems.
            - **AMI ID**. For BYOI, select the AMI that you want to use from the dropdown.
        - **Scale up and Scale out**. Select an upgrade strategy for your system hardware to upgrade for increased data and workload.
            - **Scale-up deployment**.If you choose this deployment upgrade model, enter the host name for the database
            - **Scale-out deployment**. If you choose this deployment upgrade model, enter the **SAP HANA master host name**, the **Number of worker nodes**, and the **Worker node hostname prefix** under **Instance sizing**.
    - Under **Instance sizing**, choose whether to **Use AWS recommended resources** or **Choose instance**.
        - **Use AWS recommended resources**.
            - **Define requirements**. Choose the requirements for your recommended resources from the dropdown list.
                - **Based on CPU/Memory**. If you select this option, enter the required number of vCPU **Cores** and **Memory**. Amazon EC2 supports up to 1920 logical processors. If the amount of memory required exceeds 4TB, dedicated hosts are required.
                - **SAPS (SAP Application Performance Standard)**. If you select this option, enter the **SAPS** rating for the SAP certified instance types.
        - **Instance type**. Choose the instance type from the dropdown list.
        - **Private IP address**. Choose whether to use an **Auto-assigned (default)** IP address or a **Custom IP address**.
            - **Auto-assign (default)**. When you select this option, an IP addressed will be assigned for you. This is the default option.

       - **Private IP address**. When you select this option, you can enter a single IP address. If you have selected multiple worker nodes, enter the IP addresses to assign to the instance for each selected node. Separate more than one IP address with commas. Verify that the IP addresses are within the subnet range of the instance you are launching. You must enter the same number of IP addresses as the number of nodes selected.
     - **Auto Recovery**. Auto recovery is an Amazon EC2 feature to increase instance availability. Select the check box to enable EC2 automatic recovery for the instance. For more information, see Recover Your Instance in the Amazon EC2 User Guide.
     - **Recommended Resources**. AWS Launch Wizard displays the **Estimated monthly cost of operation** based on your instance sizing selections. This is an estimate of AWS costs to deploy additional resources and does not include any applicable taxes or discounts.
   - Under **Amazon FSx for NetApp ONTAP storage**, enter your desired volume sizes for SAP HANA data, log, and other file systems. The displayed default values are based on your selection of the instance type.

4. **Settings for SAP Web Dispatcher**. Enter the deployment settings for your SAP Web Dispatcher instance.

   - **Instance details**.
     - Under **Image type**, choose whether to use **AWS/Marketplace/Community images** or **Bring your own images (BYOI)**.
       - **Operating System**. Select a supported operating system version for the SAP Web Dispatcher instances. For a complete list of operating system versions supported for SAP Web Dispatcher, see Operating systems.
       - **AMI ID**. For BYOI, select the AMI that you want to use from the dropdown.
     - **Private subnet**. The private subnet to provision SAP Web Dispatcher instances in.
     - **SAP Web Dispatcher ID (SID)**. The SID to use for SAP Web Dispatcher. This value must be unique throughout your SAP system's landscape.
     - **SAP Web Dispatcher Admin User ID**. The user ID number for the SAP Web Dispatcher administrator user (sid-adm).
     - **Hostname**. The hostname to use for the EC2 instance where SAP Web Dispatcher is deployed.
   - **Private IP address**. Choose whether to use an **Auto-assigned (default)** IP address or a **Custom IP address**.

    - **Auto-assign IP address (default)**. When you select this option, an IP addressed will be assigned for you. This is the default option.
    - **Custom IP address**. When you select this option, you can enter a single IP address.
    - **Auto Recovery**. Auto recovery is an Amazon EC2 feature to increase instance availability. Select the check box to enable EC2 automatic recovery for the instance. For more information, see Recover Your Instance in the Amazon EC2 User Guide.
  - Under **Instance sizing**, choose whether to **Based on infrastructure requirements** or **Based on instance type**.
    - **Based on infrastructure requirements**
      - **Infrastructure requirements**. Choose the requirements for your recommended resources from the dropdown list.
        - **Based on CPU/Memory**. If you select this option, enter the required number of vCPU **Cores** and **Memory**. Amazon EC2 supports up to 1920 logical processors. If the amount of memory required exceeds 4TB, dedicated hosts are required.
          - **vCPU**. The amount of vCPUs cores required by the instance.
          - **Memory (GB)**. The amount of memory required for each instance.
        - **SAPS (SAP Application Performance Standard)**. If you select this option, enter the **SAPS** rating for the SAP certified instance types.
          - **SAPS**. We will recommend SAP certified instance types which offer SAPS ratings for the value you enter.
      - **Based on instance type**
        - **Instance type**. Choose the instance type from the dropdown list.
    - **Recommended Resources**. AWS Launch Wizard displays the **Estimated monthly cost of operation** based on your instance sizing selections. This is an estimate of AWS costs to deploy additional resources and does not include any applicable taxes or discounts.

5. **Settings for Additional App Servers (AAS) - *optional***. Enter the deployment settings for your AAS instances.

  - **Instance details**.
    - **Number of Additional App Servers (AAS)**. Enter the number of additional application servers.
    - **Naming convention for host name**. Enter the naming convention for the host name.

    - **Auto Recovery**. Auto recovery is an Amazon EC2 feature to increase instance availability. Select the check box to enable EC2 automatic recovery for the instance. For more information, see Recover Your Instance in the Amazon EC2 User Guide.

- Under **Instance sizing**, choose whether to Use **AWS recommended resources** or **Choose your instance**.
    - **Use AWS recommended resources**.
        - **Define requirements**. Choose the requirements for your recommended resources from the dropdown list.
            - **Based on CPU/Memory**. If you select this option, enter the required number of vCPU **Cores** and **Memory**. Amazon EC2 supports up to 1920 logical processors. If the amount of memory required exceeds 4TB, dedicated hosts are required.
            - **SAPS (SAP Application Performance Standard)**. If you select this option, enter the **SAPS** rating for the SAP certified instance types.
    - **Choose your instance**.
        - **Instance type**. Choose the instance type from the dropdown list.
    - **Recommended Resources**. AWS Launch Wizard displays the **Estimated monthly cost of operation** based on your instance sizing selections. This is an estimate of AWS costs to deploy additional resources and does not include any applicable taxes or discounts.

6. **Pre- and post-deployment configuration scripts — optional**

   You can run pre- and post-deployment configuration scripts during application provisioning. For more information about how Launch Wizard accesses and deploys these scripts, see Custom deployment configuration scripts.

   Pre-deployment configuration script — optional

   - **Deployment settings**. In the event of a configuration script failure or time out, choose whether to proceed with the deployment. If you do not select this option, then when the configuration scripts fail or time out, Launch Wizard will roll back the workload and delete all of the AWS resources created by Launch Wizard. Pre-deployment configuration scripts must finish running in 45 minutes or less.
   - **Configuration script**. You can add one or more configuration scripts depending on the number of servers you select to run scripts during the launch phase.

  - For each pre-deployment configuration script that you want to run, choose to use a script located in Amazon S3 and enter the URL path of the script, or upload a script file.
  - Select the servers to run the pre-deployment configuration scripts during the launch phase. You can choose to run pre-deployment scripts on **ASCS/SCS Server and Primary Application Server (PAS)**, **Database (DB) Server**, and **Additional App Servers (AAS)**. You can add a script for each server selected.
  - To remove a configuration script, choose **Remove script**. To add more configuration scripts, choose **Add another script**.

Post-deployment configuration script — optional

- **Deployment settings**. In the event of a configuration script failure, choose whether to proceed with the deployment. If you do not select this option, then when the configuration scripts fail or time out, Launch Wizard will roll back the workload and delete all of the AWS resources created by Launch Wizard. Post-deployment configuration scripts must finish running in 2 hours or less.
- **Configuration script**. You can add one or more configuration scripts depending on the number of servers you select to run scripts during the post-deployment phase.
  - For each post-deployment configuration script that you want to run, choose to use a script located in Amazon S3 and enter the URL path of the script, or upload a script file.
  - Select the servers to run the post-deployment configuration scripts when an EC2 instance has been configured for use. You can choose to run the post-deployment scripts on **ASCS/SCS Server and Primary Application Server (PAS)**, **Database (DB) Server**, and **Additional App Servers (AAS)**. You can add a script for each server selected.
  - To remove a configuration script, choose **Remove script**. To add more configuration scripts, choose **Add another script**.

7. After you have entered your additional settings, choose **Next**.

(See the **SAP software installation settings** tab.)

High availability deployment

On the **Configure SAP HANA deployment model** page, enter the deployment details for the high availability deployment.

1. **Deployment details**. Launch Wizard supports single instance deployments, distributed instance deployments, and high availability deployments. Select **High availability deployment**.
2. **Settings for ABAP System Central Services (ASCS) server**. Enter the deployment settings for your instance.
   - **Instance details**.
     - Under **Image type**, choose whether to use **AWS/Marketplace/Community images** or **Bring your own images (BYOI)**.
       - **Operating System**. Select a supported operating system version for the ASCS instances. For a complete list of operating system versions supported for ASCS, see Operating systems.
       - **AMI ID**. For BYOI, select the AMI that you want to use from the dropdown list.
     - **Host name**. Enter the host name for the EC2 instance.
     - **ASCS instance number**. Enter the instance number for the SAP installation and setup, and to open up ports for security groups.
   - Under **Instance sizing**, choose whether to **Use AWS recommended resources** or **Choose your instance**.
     - **Use AWS recommended resources**.
       - **Based on CPU/Memory**. If you select this option, enter the required number of vCPU **Cores** and **Memory**. Amazon EC2 supports up to 1920 logical processors. If the amount of memory required exceeds 4 TB, dedicated hosts are required.
       - **SAPS (SAP Application Performance Standard)**. If you select this option, enter the **SAPS** rating for the SAP certified instance type.
     - **Choose your instance**.
       - **Instance type**. Choose the instance type from the dropdown list.
   - **Recommended Resources**. AWS Launch Wizard displays the **Estimated monthly cost of operation** based on your instance sizing selections. This is an estimate of AWS costs to deploy additional resources. It does not include any applicable taxes or discounts.

    - Under **Amazon FSx for NetApp ONTAP storage**, enter your desired volume sizes for SAP HANA data, log, and other file systems. The displayed default values are based on your selection of the instance type.

3. **Settings for Enqueue Replication Server (ERS)**. Enter the deployment settings for your ERS.

    - **Instance details**.
        - Under **Instance sizing**, choose whether to use **AWS/Marketplace/Community images** or **Bring your own images (BYOI)**.
            - **Operating System**. Select a supported operating system version for the ERS instance.
            - **AMI ID**. For BYOI, select the AMI that you want to use from the dropdown list.
        - **Host name**. Enter the host name for the EC2 instance.
        - **ERS instance number**. Enter the instance number for the SAP installation and setup, and to open up ports for security groups.
    - Under **Instance sizing**, choose whether to **Use AWS recommended resources** or **Choose your instance**.
        - **Use AWS recommended resources**.
            - **Based on CPU/Memory**. If you select this option, enter the required number of vCPU **Cores** and **Memory**. Amazon EC2 supports up to 1920 logical processors. If the amount of memory required exceeds 4TB, dedicated hosts are required.
            - **SAPS (SAP Application Performance Standard)**. If you select this option, enter the **SAPS** rating for the SAP certified instance type.
        - **Choose your instance**.
            - **Instance type**. Choose the instance type from the dropdown list.
    - **Recommended Resources**. AWS Launch Wizard displays the **Estimated monthly cost of operation** based on your instance sizing selections. This is an estimate of AWS costs to deploy additional resources and does not include any applicable taxes or discounts.

4. **Settings for database (DB) Server**. Enter the deployment settings for your database.

    - Under **Instance sizing**, choose whether to use **AWS/Marketplace/Community images** or **Bring your own images (BYOI)**.
        - **Instance details.**

    - **Operating System**. Select a supported operating system version for the ERS instance.
    - **AMI ID**. For BYOI, select the AMI that you want to use from the dropdown list.
- **Primary and secondary instance details**. Enter details for both the primary and secondary instances.
  - **SAP HANA host name**. Enter the host name for the SAP HANA primary and secondary instances.
  - **Server site name**. Enter the primary and secondary site name for the SAP HANA system replication.

  Private IP address settings

  - **Primary instance details**

    Private IP address. Choose whether to use an **Auto-assigned (default)** IP address or a **Custom IP address** for your primary instance.

    - **Auto-assign (default)**. When you select this option, an IP addressed will be assigned for you. This is the default option.
    - **Private IP address**. When you select this option, you can enter a single IP address. Verify that this IP address is within the subnet range of the instance you are launching.
  - **Secondary instance details**

    Private IP address. Choose whether to use an **Auto-assigned (default)** IP address or a **Custom IP address** for your secondary instance.

    - **Auto-assign (default)**. When you select this option, an IP addressed will be assigned for you. This is the default option.
    - **Private IP address**. When you select this option, you can enter a single IP address. Verify that this IP address is within the subnet range of the instance you are launching.
- **Overlay IP address**. Enter the overlay IP address to assign to the active node. The IP address should be outside of the VPC CIDR and must not be used by any other HA cluster. It is configured to always point to the active SAP HANA node.
- **Pacemaker tag name**. Enter the tag to assign to each EC2 instance. This tag is used by the pacemaker component of SLES HAE and RHEL for SAP high availability solutions and must not be used by any other EC2 instance in your account.

- Under **Instance sizing**, choose whether to **Use AWS recommended resources** or **Choose your instance**.
    - **Use AWS recommended resources**.
        - **Based on CPU/Memory**. If you select this option, enter the required number of vCPU **Cores** and **Memory**. Amazon EC2 supports up to 1920 logical processors. If the amount of memory required exceeds 4 TB, dedicated hosts are required.
        - **SAPS (SAP Application Performance Standard)**. If you select this option, enter the **SAPS** rating for the SAP certified instance type.
    - **Choose your instance**.
        - **Instance type**. Choose the instance type from the dropdown list.
- **Recommended Resources**. AWS Launch Wizard displays the **Estimated monthly cost of operation** based on your instance sizing selections. This is an estimate of AWS costs to deploy additional resources. It does not include any applicable taxes or discounts.

5. **Primary Application Server (PAS)**. Enter the deployment settings for your instance.

    - **Instance details**.
        - Under **Image type**, choose whether to use **AWS/Marketplace/Community images** or **Bring your own images (BYOI)**.
            - **Operating System**. Select a supported operating system version for the ERS instance.
            - **AMI ID**. For BYOI, select the AMI that you want to use from the dropdown list.
        - **Host name**. Enter the host name for the EC2 instance.
        - **Auto Recovery**. Auto recovery is an Amazon EC2 feature to increase instance availability. Select the check box to enable Amazon EC2 automatic recovery for the instance. For more information, see Recover Your Instance in the Amazon EC2 User Guide.
    - Under **Instance sizing**, choose whether to **Use AWS recommended resources** or **Choose your instance**.
        - **Use AWS recommended resources**.
            - **Define requirements**. Choose the requirements for your recommended resources from the dropdown list.

         - **Based on CPU/Memory**. If you select this option, enter the required number of vCPU **Cores** and **Memory**. Amazon EC2 supports up to 1920 logical processors. If the amount of memory required exceeds 4TB, dedicated hosts are required.
         - **SAPS (SAP Application Performance Standard)**. If you select this option, enter the **SAPS** rating for the SAP certified instance types.
     - **Choose your instance**.
       - **Instance type**. Choose the instance type from the dropdown list.
     - **Recommended Resources**. AWS Launch Wizard displays the **Estimated monthly cost of operation** based on your instance sizing selections. This is an estimate of AWS costs to deploy additional resources. It does not include any applicable taxes or discounts.

6. **Settings for SAP Web Dispatcher**. Enter the deployment settings for your SAP Web Dispatcher instances.

   - **Instance details**.
     - Under **Image type**, choose whether to use **AWS/Marketplace/Community images** or **Bring your own images (BYOI)**.
       - **Operating System**. Select a supported operating system version for the SAP Web Dispatcher instances. For a complete list of operating system versions supported for SAP Web Dispatcher, see Operating systems.
       - **AMI ID**. For BYOI, select the AMI that you want to use from the dropdown.
     - **SAP Web Dispatcher ID (SID)**. The SID to use for SAP Web Dispatcher. This value must be unique throughout your SAP system's landscape.
     - **SAP Web Dispatcher Admin User ID**. The user ID number for the SAP Web Dispatcher administrator user (sid-adm).
   - Enter the following information for both the primary and secondary instance:
     - **Private subnet**. The private subnet to provision SAP Web Dispatcher instances in.
     - **Hostname**. The hostname to use for the EC2 instance where SAP Web Dispatcher is deployed.
     - **Private IP address**. Choose whether to use an **Auto-assigned (default)** IP address or a **Custom IP address**.
       - **Auto-assign IP address (default)**. When you select this option, an IP addressed will be assigned for you. This is the default option.
       - **Custom IP address**. When you select this option, you can enter a single IP address.

        - **Auto Recovery**. Auto recovery is an Amazon EC2 feature to increase instance availability. Select the check box to enable EC2 automatic recovery for the instance. For more information, see Recover Your Instance in the Amazon EC2 User Guide.

    - Under **Instance sizing**, choose whether to **Based on infrastructure requirements** or **Based on instance type**.

        - **Based on infrastructure requirements**

            - **Infrastructure requirements**. Choose the requirements for your recommended resources from the dropdown list.

                - **Based on CPU/Memory**. If you select this option, enter the required number of vCPU **Cores** and **Memory**. Amazon EC2 supports up to 1920 logical processors. If the amount of memory required exceeds 4TB, dedicated hosts are required.

                    - **vCPU**. The amount of vCPUs cores required by the instance.

                    - **Memory (GB)**. The amount of memory required for each instance.

                - **SAPS (SAP Application Performance Standard)**. If you select this option, enter the **SAPS** rating for the SAP certified instance types.

                    - **SAPS**. We will recommend SAP certified instance types which offer SAPS ratings for the value you enter.

            - **Based on instance type**

                - **Instance type**. Choose the instance type from the dropdown list.

        - **Recommended Resources**. AWS Launch Wizard displays the **Estimated monthly cost of operation** based on your instance sizing selections. This is an estimate of AWS costs to deploy additional resources and does not include any applicable taxes or discounts.

7. **Settings for Additional App Servers (AAS) - *optional***. Enter the deployment settings for your AAS instances.

    - **Instance details**

        - **Number of Additional App Servers (AAS)**. Enter the number of additional application servers.

        - **Naming convention for host name**. Enter the naming convention for the host name.

        - **Auto Recovery**. Auto recovery is an Amazon EC2 feature to increase instance availability. Select the check box to enable Amazon EC2 automatic recovery for the instance. For more information, see Recover Your Instance in the Amazon EC2 User Guide.

- Under **Instance sizing**, choose whether to **Use AWS recommended resources** or **Choose your instance**.
    - **Use AWS recommended resources**.
        - **Infrastructure requirements**. Choose the requirements for your recommended resources from the dropdown list.
            - **Based on CPU/Memory**. If you select this option, enter the required number of vCPU **Cores** and **Memory**. Amazon EC2 supports up to 1920 logical processors. If the amount of memory required exceeds 4 TB, dedicated hosts are required.
            - **SAPS (SAP Application Performance Standard)**. If you select this option, enter the **SAPS** rating for the SAP certified instance types.
    - **Choose your instance**.
        - **Instance type**. Choose the instance type from the dropdown list.
    - **Recommended Resources**. AWS Launch Wizard displays the **Estimated monthly cost of operation** based on your instance sizing selections. This is an estimate of AWS costs to deploy additional resources. It does not include any applicable taxes or discounts.

8. **Pre- and post-deployment configuration scripts — optional**

    You can run pre- and post-deployment configuration scripts during application provisioning. For more information about how Launch Wizard accesses and deploys these scripts, see Custom deployment configuration scripts.

    Pre-deployment configuration script — optional

    - **Deployment settings**. Choose whether to proceed with the deployment if a configuration script fails or times out. If you do not select this option, if the configuration scripts fail or time out, Launch Wizard will roll back the workload and delete all of the AWS resources created by Launch Wizard. Pre-deployment configuration scripts must finish running in 45 minutes or less.
    - **Configuration script**. You can add one or more configuration scripts depending on the number of servers that you select to run scripts during the launch phase.
        - For each pre-deployment configuration script that you want to run, choose to use a script located in Amazon S3 and enter the URL path of the script, or upload a script file.
        - Select the servers to run the pre-deployment configuration scripts during the launch phase. You can choose to run pre-deployment scripts on **Primary Application**

Server (PAS), **ABAP System Central Services (ASCS) Server**, **Database (DB) Server**, **Additional App Servers (AAS)**, and **Enqueue Replication Server (ERS)**. You can add a script for each server selected.

- To remove a configuration script, choose **Remove script**. To add more configuration scripts, choose **Add another script**.

Post-deployment configuration script — optional

- **Deployment settings**. Choose whether to proceed with the deployment if a configuration script fails. If you do not select this option, if the configuration scripts fail or time out, Launch Wizard will roll back the workload and delete all of the AWS resources created by Launch Wizard. Post-deployment configuration scripts must finish running in 2 hours or less.
- **Configuration script**. You can add one or more configuration scripts depending on the number of servers that you select to run scripts during the post-deployment phase.
  - For each post-deployment configuration script that you want to run, choose to use a script located in Amazon S3 and enter the URL path of the script, or upload a script file.
  - Select the servers to run the post-deployment configuration scripts when an EC2 instance has been configured for use. You can choose to run the post-deployment scripts on **Primary Application Server (PAS)**, **ABAP System Central Services (ASCS) Server**, **Database (DB) Server**, **Additional App Servers (AAS)**, and **Enqueue Replication Server (ERS)**. You can add a script for each server selected.
  - To remove a configuration script, choose **Remove script**. To add more configuration scripts, choose **Add another script**.

9. After you have entered all of your deployment settings, choose **Next**.

(See the **SAP software installation settings** tab.)

SAP software installation settings

On the **Configure SAP application software installation** page, enter the software installation details for a single instance, distributed instance, or high availability deployment.

1. **SAP application software**. Choose whether to install the SAP installation software.

- If you choose **No**, choose whether to install HANA software. If you want to install HANA software, enter the **S3 location for HANA media** and the **HANA password**. Then, proceed to **Step 6**. If you don't want to install HANA software, proceed to **Step 9**.
- If you choose **Yes**, provide the information listed in the following steps.

2. **Application** and **Version**. If you choose to install the SAP application software, select the supported application and version of the software you want to install. The following configuration fields will change based on your application software and version selections. For supported application versions, see SAP applications.

3. **Load balancer for SAP Web Dispatcher**. You can select **Add Load balancer for SAP Web Dispatcher** to launch a load balancer to distribute incoming traffic to your SAP Web Dispatcher instances.

   - **Load balancer type**. Choose the type of load balancer to deploy. For more information, see Load balancers for SAP Web Dispatcher.
   - **Scheme**. Choose whether the load balancer should be internet-facing or intranet-facing. For more information, see Architectures for SAP Web Dispatcher.
   - **Load balancer secure communication**. You can enable this option to configure an HTTPS/TLS listener for your load balancer. This option will terminate the HTTPS/TLS connection at the load balancer. For more information, see Enable HTTPS communication.
     - **ACM certificate ARN**. Enter the ARN of a certificate in AWS Certificate Manager (ACM) to use for the load balancer HTTPS/TLS listener.
   - **Availability Zones(AZ) and subnet**. The public or private subnets to deploy the load balancer in.
     - **Availability Zone 1**. Choose the first availability zone to use.
     - **Availability Zone 2**. Choose the second availability zone to use.
     - **Load balancer security group**. Choose the security group to assign to the load balancer.

4. **SAP application software location**. In order to install the SAP application software, Launch Wizard requires access to the relevant software and files. For instructions to provide Launch Wizard access to the application software and associated files, see Make SAP application software available for AWS Launch Wizard to deploy SAP.

   - **SAPCAR location**. Enter the Amazon S3 path where the SAPCAR is located.

- **Software Provisioning Manager (SWPM) location**. Enter the Amazon S3 path where the SWPM is located.
- **Kernel software location**. Enter the Amazon S3 path where the unextracted kernel with media label is located.
- **Installation export location**. Enter the Amazon S3 path where the installation export is located.
- **HANA database software location**. Enter the Amazon S3 path where the SAP HANA database software is located.
- **SAP HANA client software location**. Enter the Amazon S3 path where the SAP HANA client software is located.

5. **Installation details**

   The following fields may vary according to the application selected.

   - **Schema name** and **Master password**. Enter the schema name and password to use for the HANA database.
   - **PAS instance number**. Enter the PAS instance number.
   - **ASCS virtual host name**. Enter the ASCS virtual host name used to set up high availability.
   - **ASCS virtual IP address**. Enter the ASCS virtual IP address.
   - **Enqueue Replication Server (ERS) instance number**. Enter the instance number to use for the ERS instance.
   - **Enqueue Replication Server (ERS) virtual IP address**. Enter the virtual IP address used to set up high availability.
   - **Enqueue Replication Server (ERS) virtual host name**. Enter the virtual host name used to set up high availability.
   - **ASCS instance number**. Enter the ASCS instance number.
   - **Database installation**. Choose whether or not to install the HANA database.
   - **Database virtual host name**. Enter the database virtual host name used to set up high availability.
   - **Software**. Select the software type that you want to install. You can install SQL or SAP software.
   - **Host name**. Enter the Central Instance, ASCS, ASCS virtual IP, or Enqueue Replication Server (ERS) host name.

6. **Additional installation details**. Select the parameter name and values to use for your software installation. The following fields may vary according to the application selected.

   - **Number of batch processes**. Enter the maximum number of batch processes.
   - **Number of dialog processes**. Enter the maximum number of dialog processes.
   - **UID for SAP host agent**. Enter the UID for the SAP host agent.
   - **Create a DBA Cockpit user**. Choose whether to create a DBA Cockpit user.

7. **AWS Backint Agent**. Select the check box to install AWS Backint Agent. For more information, see AWS Backint Agent for SAP HANA.

   a. **S3 file path**. Select or enter the Amazon S3 location to store the SAP HANA backup files.

   b. **AWS KMS key ARN**. Select the ARN of the KMS key that can be used by AWS Backint Agent to encrypt the backup files. For more information, see the AWS Backint Agent for SAP documentation.

   c. **Agent version**. Select the AWS Backint Agent version you want to install.

8. **Additional preferences.**

   a. When you use AWS Backint Agent, the HANA backup files are stored in Amazon S3, which eliminates the requirement for local EBS backup volumes. If you want Launch Wizard to provision local EBS backup volumes for file-based backups that can be configured manually after deployments, select the check box.

   b. By default, a Launch Wizard deployment rolls back when the AWS Backint Agent installation fails. If you want to continue with a Launch Wizard deployment when the AWS Backint Agent installation fails, select the check box. This option does not apply to high availability deployments.

9. **IAM permissions**. To deploy an application successfully, Launch Wizard must be allowed to perform operations in other AWS services on your behalf. To do this, the Launch Wizard IAM role, `AmazonEC2RoleForLaunchWizard`, must have permissions attached to perform these operations, which include AWS Backint Agent operations, running pre- and post-deployment configuration scripts, and downloading the SAP installation media from Amazon S3. If the required policy is not attached to the Launch Wizard role, the Launch Wizard deployment can fail. Select the check box to verify that you have attached the required permissions before deploying.

For steps to attach the required permissions to `AmazonEC2RoleForLaunchWizard`, see [AWS Identity and Access Management (IAM)](#) in this guide.

10. Choose **Deploy**

(See the **Review** tab)

Review

- On the **Review** page, review your infrastructure, application, and deployment model settings. If you are satisfied with your selections, choose **Deploy**. If you want to change settings, choose **Previous**.
- When you choose **Deploy**, you are redirected to the **Deployments** page, where you can view the status of your deployment, and also the deployment details.

SAP HANA database

Application settings

On the **Configure application settings** page, enter your SAP HANA database application settings.

1. **Application type**. Select **SAP HANA database**. This configuration includes:
   - EC2 instances for an SAP HANA database
   - Optional installation of SAP HANA database software
2. **General Settings – SAP HANA**. Enter the settings for your SAP HANA database installation.
   - **SAP HANA System ID (SID)**. Enter the SAP HANA system ID for your system. The HANASID must be different from SAPSID if you are configuring a single instance deployment.
   - **SAP HANA Instance number**. Enter the instance number to use for your SAP HANA system. This must be a two-digit number from 00 through 99.
   - **EBS Volume Type for SAP HANA**. Select the EBS volume types that you want to use for **SAP HANA Data**, **SAP HANA Logs**, and **SAP Others** from the dropdown lists.

> **Note**
>
> gp3 volumes are not supported for HANA production databases running on Xen instances (X1, X1e, R4, and R3). When you deploy HANA databases with Xen instances after choosing **Production** as the **Deployment environment** under the **Configuration options**, gp2 volumes will be used to set up SAP HANA Data and Logs on the instances you selected for the HANA database.

- Select *Make this selection to use Amazon FSx for NetApp ONTAP for all SAP HANA database file systems, except root, backup, and media file systems.*

  Your chosen Amazon EBS volume type is used for the application layer.
- **SAP HANA software install**. Select whether you want to download the SAP HANA software.
  - If you select **Yes**, enter the Amazon S3 location where the SAP HANA software is located. The S3 bucket must have the prefix “launchwizard” in the bucket name to ensure that the Launch Wizard IAM role policy for EC2 has read-only access to the bucket. For steps to set up the folder structure for your S3 bucket, see Make SAP HANA software available for AWS Launch Wizard to deploy a HANA database. Enter a password to use for your SAP HANA installation.
    - **AWS Backint Agent.** Select the check box if you want to deploy AWS Backint Agent for backup and recover along with the application. For more information about AWS Backint Agent, see AWS Backint Agent for SAP HANA.
      - **S3 URI.** Enter the URI of the S3 bucket where you want to store your SAP HANA backup files. For example, `s3://`**`<bucket-name>`**.
      - **S3 Encryption (AWS KMS key ARN).** Select the ARN of the KMS key that AWS Backint Agent can use to encrypt the backup files stored in your Amazon S3 bucket.
      - **Agent version.** Select the version number of the agent that you want to install. If you do not enter a version number, the latest published version of the agent is installed.
      - **Additional Backint preferences.**
        - If you selected to use AWS Backint agent, the agent backs up files to S3, which removes the requirement for EBS backup volumes. Select this check box to provision local EBS backup volumes for file-level backups.

        - By default, Launch Wizard rolls back a deployment when the AWS Backint Agent installation fails. Select the check box if you want Launch Wizard to continue with non-HA application deployments when the Backint installation fails.

      - **Verify that you have attached the required policy for Backint operations to the following role.** Select this check box after you have attached the required policy to the `AmazonEC2RoleForLaunchWizard`. This policy allows Launch Wizard to perform Backint Agent operations on your behalf. The policy and instructions to attach the policy to the role are provided by Launch Wizard during deployment. This information can also be found in [Step 2 of the Backint Agent IAM documentation](#).

    - If you select **No**, only the AWS infrastructure is provisioned so you can manually deploy an SAP HANA database post deployment .

3. After you enter your application settings, choose **Next**.

(Use the tab for **Single instance deployment**, **Multiple instance deployment**, or **High availability deployment**, depending on your configuration)

Single instance deployment

On the **Configure deployment model** page, enter the deployment details for the SAP HANA database deployment.

1. **Deployment model**. Launch Wizard supports single instance deployments, multiple instance deployments, and high availability deployments. Select **Single instance deployment**.

2. **Settings for SAP HANA database on one instance**

   - **Instance details.**

     - Under **Image type**, choose whether to use **AWS/Marketplace/Community images** or **Bring your own images (BYOI)**.

       - **Operating System**. Select a supported operating system version for the ERS instance.

       - **AMI ID**. For BYOI, select the AMI that you want to use from the dropdown.

     - **Host name**. Enter the host name for the EC2 instance.

    - **Auto Recovery**. Auto recovery is an Amazon EC2 feature to increase instance availability. Select the check box to enable EC2 automatic recovery for the instance. For more information, see Recover Your Instance in the Amazon EC2 User Guide.

  - Under **Instance sizing**, choose **Use AWS recommended resources** or **Choose your instance**.
    - **Use AWS recommended resources**.
      - **Define requirements**. Choose the requirements for your recommended resources from the dropdown list.
        - **Based on CPU/Memory**. If you select this option, enter the required number of vCPU **Cores** and **Memory**. Amazon EC2 supports up to 1920 logical processors. If the amount of memory required exceeds 4TB, dedicated hosts are required.
        - **SAPS (SAP Application Performance Standard)**. If you select this option, enter the **SAPS** rating for the SAP certified instance types.
    - **Choose your instance**.
      - **Instance type**. Choose the instance type from the dropdown list.
    - **Recommended Resources**. Launch Wizard displays the **Estimated monthly cost of operation** based on your instance sizing selections. This is an estimate of AWS costs to deploy additional resources and does not include applicable taxes or discounts.
  - Under **Amazon FSx for NetApp ONTAP storage**, enter your desired volume sizes for SAP HANA data, log, and other file systems. The displayed default values are based on your selection of the instance type.

3. **Pre- and post-deployment configuration scripts — optional**

   You can run pre- and post-deployment configuration scripts during application provisioning. For more information about how Launch Wizard accesses and deploys these scripts, see Custom deployment configuration scripts.

   Pre-deployment configuration script — optional

   - **Deployment settings**. In the event of a configuration script failure or time out, select whether to ignore all failures and proceed with the deployment. If you do not select this option, when the configuration scripts fail or time out, Launch Wizard will roll back the workload and delete all of the AWS resources created by Launch Wizard. Pre-deployment configuration scripts must finish running in 45 minutes or less.

- **Configuration script**. Choose to use a script located in Amazon S3 and enter the URL path of the script, or enter a script manually by uploading a script file. To remove the configuration script, choose **Remove script**.

Post-deployment configuration script — optional

- **Deployment settings**. In the event of a configuration script failure or time out, select whether to ignore all failures and proceed with the deployment. If you do not select this option, when the configuration scripts fail or time out, Launch Wizard will roll back the workload and delete all of the AWS resources created by Launch Wizard. Post-deployment configuration scripts must finish running in 2 hours or less.
- **Configuration script**. Choose to use a script located in Amazon S3 and enter the URL path of the script, or enter a script manually by uploading a script file. To remove the configuration script, choose **Remove script**.

4. After you enter your deployment settings, choose Next.

(See the **Review** tab)

Multiple instance deployment

On the **Configure deployment model** page, enter the deployment details for the SAP HANA database deployment.

1. **Deployment model**. Launch Wizard supports single instance deployments, multiple instance deployments, and high availability deployments. Select **Multiple instance deployment**.
2. **SAP HANA on multiple EC2 instances**

   - **Instance details.**
     - Under **Instance sizing**, choose whether to use **AWS/Marketplace/Community images** or **Bring your own images (BYOI)**.
       - **Operating System**. Select a supported operating system version for the SAP HANA servers.
       - **AMI ID**. For BYOI, select the AMI that you want to use from the dropdown.

- Under **Instance sizing**, choose **Use AWS recommended resources** or **Choose your instance**.
  - **Use AWS recommended resources**.
    - **Infrastructure requirements**. Choose the requirements for your recommended resources from the dropdown list.
      - **Based on CPU/Memory**. If you select this option, enter the required number of vCPU **Cores** and **Memory**. Amazon EC2 supports up to 1920 logical processors. If the amount of memory required exceeds 4TB, dedicated hosts are required.
      - **SAPS (SAP Application Performance Standard)**. If you select this option, enter the **SAPS** rating for the SAP certified instance types.
  - **Choose your instance**.
    - **Instance type**. Choose the instance type from the dropdown list.
  - **Host Name for SAP system**. Enter the host name for the EC2 instance.
  - **Number of worker nodes**. Enter the number of EC2 instances to be configured as worker nodes for this SAP HANA system.
  - **Worker node hostname prefix**. Enter the hostname prefix for the worker nodes.
  - **Auto Recovery**. Auto recovery is an Amazon EC2 feature to increase instance availability. Select the check box to enable EC2 automatic recovery for the instance. For more information, see Recover Your Instance in the Amazon EC2 User Guide.
  - **Recommended Resources**. Launch Wizard displays the **Estimated monthly cost of operation** based on your instance sizing selections. This is an estimate of AWS costs to deploy additional resources and does not include applicable taxes or discounts.
- Under **Amazon FSx for NetApp ONTAP storage**, enter your desired volume sizes for SAP HANA data, log, and other file systems. The displayed default values are based on your selection of the instance type.

3. **Pre- and post-deployment configuration scripts — optional**

   You can run pre- and post-deployment configuration scripts during application provisioning. For more information about how Launch Wizard accesses and deploys these scripts, see Custom deployment configuration scripts.

Pre-deployment configuration script — optional

- **Deployment settings**. In the event of a configuration script failure or time out, select whether to ignore all failures and proceed with the deployment. If you do not select this option, when the configuration scripts fail or time out, Launch Wizard will roll back the workload and delete all of the AWS resources created by Launch Wizard. Pre-deployment configuration scripts must finish running in 45 minutes or less.
- **Configuration script**. Choose to use a script located in Amazon S3 and enter the URL path of the script, or enter a script manually by uploading a script file. To remove the configuration script, choose **Remove script**.

Post-deployment configuration script — optional

- **Deployment settings**. In the event of a configuration script failure or time out, select whether to ignore all failures and proceed with the deployment. If you do not select this option, when the configuration scripts fail or time out, Launch Wizard will roll back the workload and delete all of the AWS resources created by Launch Wizard. Post-deployment configuration scripts must finish running in 2 hours or less.
- **Configuration script**. Choose to use a script located in Amazon S3 and enter the URL path of the script, or enter a script manually by uploading a script file. To remove the configuration script, choose **Remove script**.

4. After you enter your deployment settings, choose **Next**.

(See **Review** tab)

High availability deployment

On the **Configure deployment model** page, enter the deployment details for the SAP HANA database deployment.

1. **Deployment model**. Launch Wizard supports single instance deployments, multiple instance deployments, and high availability deployments. Select **High availability deployment**.
2. **Instance details.**

   - Under **Instance details**, choose whether to use **AWS/Marketplace/Community images** or **Bring your own images (BYOI)**.

  - **Operating System**. Select a supported operating system version for the SAP HANA servers.
  - **AMI ID**. For BYOI, select the AMI that you want to use from the dropdown.
- **Primary and secondary instance details**. Enter details for both the primary and secondary instances.
  - **SAP HANA host name**. Enter the host name for the SAP HANA primary and secondary instances.
  - **Server site name**. Enter the primary and secondary site name for the SAP HANA system replication.
- **Overlay IP address**. Enter the overlay IP address to assign to the active node. The IP address should be outside of the VPC CIDR and must not be used by any other HA cluster. It is configured to always point to the active SAP HANA node.
- **Pacemaker tag name**. Enter the tag to assign to each EC2 instance. This tag is used by the pacemaker component of SLES HAE and RHEL for SAP high availability solutions and must not be used by any other EC2 instance in your account.

- Under **Instance sizing**, choose **Use AWS recommended resources** or **Choose your instance**.
  - **Use AWS recommended resources**.
    - **Infrastructure requirements**. Choose the requirements for your recommended resources from the dropdown list.
      - **Based on CPU/Memory**. If you select this option, enter the required number of vCPU **Cores** and **Memory**. Amazon EC2 supports up to 1920 logical processors. If the amount of memory required exceeds 4TB, dedicated hosts are required.
      - **SAPS (SAP Application Performance Standard)**. If you select this option, enter the **SAPS** rating for the SAP certified instance types.
  - **Choose your instance**.
    - **Instance type**. Choose the instance type from the dropdown list.
  - **Recommended Resources**. Launch Wizard displays the **Estimated monthly cost of operation** based on your instance sizing selections. This is an estimate of AWS costs to deploy additional resources and does not include applicable taxes or discounts.

- Under **Amazon FSx for NetApp ONTAP storage**, enter your desired volume sizes for SAP HANA data, log, and other file systems. The displayed default values are based on your selection of the instance type.

3. **Pre- and post-deployment configuration scripts — optional**

   You can run pre- and post-deployment configuration scripts during application provisioning. For more information about how Launch Wizard accesses and deploys these scripts, see Custom deployment configuration scripts.

   Pre-deployment configuration script — optional

   - **Deployment settings**. In the event of a configuration script failure or time out, select whether to ignore all failures and proceed with the deployment. If you do not select this option, when the configuration scripts fail or time out, Launch Wizard will roll back the workload and delete all of the AWS resources created by Launch Wizard. Pre-deployment configuration scripts must finish running in 45 minutes or less.
   - **Configuration script**. Choose to use a script located in Amazon S3 and enter the URL path of the script, or enter a script manually by uploading a script file. To remove the configuration script, choose **Remove script**.

   Post-deployment configuration script — optional

   - **Deployment settings**. In the event of a configuration script failure or time out, select whether to ignore all failures and proceed with the deployment. If you do not select this option, when the configuration scripts fail or time out, Launch Wizard will roll back the workload and delete all of the AWS resources created by Launch Wizard. Post-deployment configuration scripts must finish running in 2 hours or less.
   - **Configuration script**. Choose to use a script located in Amazon S3 and enter the URL path of the script, or enter a script manually by uploading a script file. To remove the configuration script, choose **Remove script**.

4. After you enter your deployment settings, choose **Next**.

(See **Review** tab)

Review

- On the **Review** page, review your infrastructure, application, and deployment model settings. If you are satisfied with your selections, choose **Deploy** . If you want to change settings, choose **Previous**.
- When you choose **Deploy** , you are redirected to the **Deployments page**, where you can view the status of your deployment, and also the deployment details.

NetWeaver stack on SAP ASE database

Application settings

On the **Configure application settings** page, enter your NetWeaver stack on SAP ASE database application settings.

1. **Application type**. Select **NetWeaver stack on SAP ASE database**. This configuration includes:
   - NetWeaver stack for a single instance.
   - EC2 instances for the NetWeaver application tier
   - SAP NetWeaver ABAP or JAVA software install.
2. **General settings – SAP system**. Enter the settings for your SAP system.
   - **SAP Application ID**. Select the application you want to deploy from the dropdown. The options include, SAP NetWeaver ABAP, SAP NetWeaver JAVA, and SAP Solution Manager.
   - **SAP System ID (SAPSID)**. An identifier for your system. The ID must be a three character, alphanumeric string.
   - **SAP System Admin User ID**. The user ID number for the SAP system admin (`<sid>adm`). The minimum number is 100, and the maximum allowed number is 65536.
   - **EBS Volume Type for NetWeaver application stack instances**. Choose which volume type to use for the NetWeaver application file system `/usr/sap/`**`SAPSID`** from the dropdown list.
   - **Transport Domain Controller**. Specify whether the SAP system will be the domain controller for the SAP landscape. If not, select the transport file system of the domain controller to be mounted.

3. **General Settings – SAP Adaptive Server Enterprise (ASE)**. Enter the settings for your SAP ASE database.

   - **SAP ASE User ID number.** Enter the user ID number for the SAP ASE database admin (`syb<SAPSID>`). The minimum number is 100, and the maximum allowed number is 65536.
   - **EBS Volume Type for SAP ASE filesystems**. Select the EBS volume types to use for **SAP ASE Data**, **SAP ASE Logs**, and **SAP ASE Backup** filesystems from the dropdown lists.

4. After you enter your application settings, choose **Next**.

Use the tab for **Single instance deployment** for further information.

Single instance deployment

On the **Configure deployment model** page, enter the deployment details for a single instance deployment.

1. **ASCS, PAS, and DB on one EC2 instance**. Enter the deployment settings for your instance.

   - **Instance details**.
     - Under **Image type**, choose whether to use **AWS/Marketplace/Community image** or **Bring your own image (BYOI)**.
     - **Operating System and version**. Select a supported operating system version for the ASCS instance. For a complete list of operating system versions supported for ASCS, see Operating systems.
     - **AMI ID**. For BYOI, select the AMI that you want to use from the dropdown.
     - **Host name**. Enter the host name for the EC2 instance.
     - **Private IP address**. Choose whether to use an **Auto-assigned (default)** IP address or a **Custom IP address**.
       - **Auto-assign (default)**. When you select this option, an IP addressed will be assigned for you. This is the default option.
       - **Private IP address**. When you select this option, you can enter a single IP address. Verify that this IP address is within the subnet range of the instance you are launching.

    - **Auto Recovery**. Auto recovery is an Amazon EC2 feature to increase instance availability. Select the check box to enable EC2 automatic recovery for the instance. For more information, see Recover Your Instance in the Amazon EC2 User Guide.

  - **Instance sizing**.

    Under **Instance sizing type**, choose **Based on infrastructure requirements** or **Based on instance type**.

    - **Use AWS recommended resources**.
      - **Infrastructure requirements**. Choose the requirements for your recommended resources from the dropdown list.
        - **Based on CPU/Memory**. If you select this option, enter the required number of vCPU **Cores** and **Memory**. Amazon EC2 supports up to 1920 logical processors. If the amount of memory required exceeds 4TB, dedicated hosts are required.
        - **SAPS (SAP Application Performance Standard)**. If you select this option, enter the **SAPS** rating for the SAP certified instance types.
    - **Choose your instance**.
      - **Instance type**. Choose the instance type from the dropdown list.
  - **Storage Sizing**. Enter the size, IOPS, and throughput for the data, log, and backup filesystems. You can have upto 6 data filesystems, 1 log filesystem, and 1 backup filesystem.
  - **Recommended Resources**. AWS Launch Wizard displays the **Estimated monthly cost of operation** based on your instance sizing selections and the EBS volumes that will be created and attached to the launched instances. This is an estimate of AWS costs to deploy additional resources and does not include any image costs, EC2 reservations, applicable taxes, or discounts.

2. **Pre- and post-deployment configuration scripts — optional**

   You can run pre- and post-deployment configuration scripts during application provisioning. For more information about how Launch Wizard accesses and deploys these scripts, see Custom deployment configuration scripts.

   Pre-deployment configuration script — optional

   - **Deployment settings**. In the event of a configuration script failure or time out, select whether to ignore all failures and proceed with the deployment. If you do not select this option, when the configuration scripts fail or time out, Launch Wizard will roll back the workload and delete all of the AWS resources created by Launch Wizard. Pre-deployment configuration scripts must finish running in 45 minutes or less.

- **Configuration script**. Choose to use a script located in Amazon S3 and enter the URL path of the script, or enter a script manually by uploading a script file. To remove the configuration script, choose **Remove script**.

Post-deployment configuration script — optional

- **Deployment settings**. In the event of a configuration script failure or time out, select whether to ignore all failures and proceed with the deployment. If you do not select this option, when the configuration scripts fail or time out, Launch Wizard will roll back the workload and delete all of the AWS resources created by Launch Wizard. Post-deployment configuration scripts must finish running in 2 hours or less.
- **Configuration script**. Choose to use a script located in Amazon S3 and enter the URL path of the script, or enter a script manually by uploading a script file. To remove the configuration script, choose **Remove script**.

3. After you have entered your deployment settings, choose **Next**.

See the **SAP software installation settings** tab for further information.

SAP software installation settings

On the **Configure SAP application software installation** page, enter the software installation details for a single instance.

1. **SAP application software**. Choose whether to install the SAP installation software.

   If you choose **Yes**, provide the information listed in the following steps.
2. **Application** and **Version**. If you choose to install the SAP application software, select the supported application and version of the software you want to install. The following configuration fields will change based on your application software and version selections. For supported application versions, see SAP applications.
3. **SAP application software location**. In order to install the SAP application software, Launch Wizard requires access to the relevant software and files. For instructions to provide Launch Wizard access to the application software and associated files, see Make SAP application software available for AWS Launch Wizard to deploy SAP.

   - **SAPCAR location**. Enter the Amazon S3 path where the SAPCAR is located.
   - **Software Provisioning Manager (SWPM) location**. Enter the Amazon S3 path where the SWPM is located.
   - **Kernel software location**. Enter the Amazon S3 path where the unextracted kernel with media label is located.
   - **Installation export location**. Enter the Amazon S3 path where the installation export is located.
   - **SAP ASE database software location**. Enter the Amazon S3 path where the SAP ASE database software is located.

4. **Installation details**

   The following fields may vary according to the application selected.

   - **Master password**. Enter the password to use for the SAP ASE database.
   - **PAS instance number**. Enter the PAS instance number.
   - **ASCS instance number**. Enter the ASCS instance number.

5. **Additional installation details**. Select the parameter name and values to use for your software installation. The following fields may vary according to the application selected.

   - **Number of batch processes**. Enter the maximum number of batch processes.
   - **Number of dialog processes**. Enter the maximum number of dialog processes.
   - **UID for SAP host agent**. Enter the UID for the SAP host agent.

6. **IAM permissions**. To deploy an application successfully, Launch Wizard must be allowed to perform operations in other AWS services on your behalf. To do this, the Launch Wizard IAM role, `AmazonEC2RoleForLaunchWizard`, must have permissions attached to perform these operations, which include AWS Backint Agent operations, running pre- and post-deployment configuration scripts, and downloading the SAP installation media from Amazon S3. If the required policy is not attached to the Launch Wizard role, the Launch Wizard deployment can fail. Select the check box to verify that you have attached the required permissions before deploying.

   For steps to attach the required permissions to `AmazonEC2RoleForLaunchWizard`, see AWS Identity and Access Management (IAM) in this guide.

7. Choose **Deploy**

See the **Review** tab for futher information.

Multiple instance deployment

On the **Configure deployment model** page, enter the deployment details for an SAP ASE deployment.

1. **Settings for ASCS/SCS and PAS Server**. Enter the deployment settings for your application instance.

   - **Instance details**.
     - Under **Image type**, choose whether to use **AWS/Marketplace/Community image** or **Bring your own image (BYOI)**.
     - **Operating System and version**. Select a supported operating system version for the ASCS instance. For a complete list of operating system versions supported for ASCS, see Operating systems.
     - **AMI ID**. For BYOI, select the AMI that you want to use from the dropdown.
     - **Host name**. Enter the host name for the EC2 instance.
     - **Private IP address**. Choose whether to use an **Auto-assigned (default)** IP address or a **Custom IP address**.
       - **Auto-assign (default)**. When you select this option, an IP addressed will be assigned for you. This is the default option.
       - **Private IP address**. When you select this option, you can enter a single IP address. Verify that this IP address is within the subnet range of the instance you are launching.
     - **Auto Recovery**. Auto recovery is an Amazon EC2 feature to increase instance availability. Select the check box to enable EC2 automatic recovery for the instance. For more information, see Recover Your Instance in the Amazon EC2 User Guide.
   - **Instance sizing**.

     Under **Instance sizing type**, choose **Based on infrastructure requirements** or **Based on instance type**.
     - **Use AWS recommended resources**.
       - **Infrastructure requirements**. Choose the requirements for your recommended resources from the dropdown list.

        - **Based on CPU/Memory**. If you select this option, enter the required number of vCPU **Cores** and **Memory**. Amazon EC2 supports up to 1920 logical processors. If the amount of memory required exceeds 4TB, dedicated hosts are required.
        - **SAPS (SAP Application Performance Standard)**. If you select this option, enter the **SAPS** rating for the SAP certified instance types.
    - **Choose your instance**.
        - **Instance type**. Choose the instance type from the dropdown list.
- **Recommended Resources**. AWS Launch Wizard displays the **Estimated monthly cost of operation** based on your instance sizing selections and the EBS volumes that will be created and attached to the launched instances. This is an estimate of AWS costs to deploy additional resources and does not include any image costs, EC2 reservations, applicable taxes, or discounts.

2. **Settings for ASE database instance**. Enter the deployment settings for your application instance.

- **Instance details**.
    - Under **Image type**, choose whether to use **AWS/Marketplace/Community image** or **Bring your own image (BYOI)**.
    - **Operating System and version**. Select a supported operating system version for the ASCS instance. For a complete list of operating system versions supported for ASCS, see Operating systems.
    - **AMI ID**. For BYOI, select the AMI that you want to use from the dropdown.
    - **Host name**. Enter the host name for the EC2 instance.
    - **Private IP address**. Choose whether to use an **Auto-assigned (default)** IP address or a **Custom IP address**.
        - **Auto-assign (default)**. When you select this option, an IP addressed will be assigned for you. This is the default option.
        - **Private IP address**. When you select this option, you can enter a single IP address. Verify that this IP address is within the subnet range of the instance you are launching.
    - **Auto Recovery**. Auto recovery is an Amazon EC2 feature to increase instance availability. Select the check box to enable EC2 automatic recovery for the instance. For more information, see Recover Your Instance in the Amazon EC2 User Guide.
- **Instance sizing**.

Under **Instance sizing type**, choose **Based on infrastructure requirements** or **Based on instance type**.

- **Use AWS recommended resources**.
    - **Infrastructure requirements**. Choose the requirements for your recommended resources from the dropdown list.
        - **Based on CPU/Memory**. If you select this option, enter the required number of vCPU **Cores** and **Memory**. Amazon EC2 supports up to 1920 logical processors. If the amount of memory required exceeds 4TB, dedicated hosts are required.
        - **SAPS (SAP Application Performance Standard)**. If you select this option, enter the **SAPS** rating for the SAP certified instance types.
- **Choose your instance**.
    - **Instance type**. Choose the instance type from the dropdown list.

- **Storage Sizing**. Enter the size, IOPS, and throughput for the data, log, and backup filesystems. You can have upto 6 data filesystems, 1 log filesystem, and 1 backup filesystem.
- **Recommended Resources**. AWS Launch Wizard displays the **Estimated monthly cost of operation** based on your instance sizing selections and the EBS volumes that will be created and attached to the launched instances. This is an estimate of AWS costs to deploy additional resources and does not include any image costs, EC2 reservations, applicable taxes, or discounts.

3. **Pre- and post-deployment configuration scripts — optional**

   You can run pre- and post-deployment configuration scripts during application provisioning. For more information about how Launch Wizard accesses and deploys these scripts, see Custom deployment configuration scripts.

   Pre-deployment configuration script — optional

   - **Deployment settings**. In the event of a configuration script failure or time out, select whether to ignore all failures and proceed with the deployment. If you do not select this option, when the configuration scripts fail or time out, Launch Wizard will roll back the workload and delete all of the AWS resources created by Launch Wizard. Pre-deployment configuration scripts must finish running in 45 minutes or less.

- **Configuration script**. Choose to use a script located in Amazon S3 and enter the URL path of the script, or enter a script manually by uploading a script file. To remove the configuration script, choose **Remove script**.

Post-deployment configuration script — optional

- **Deployment settings**. In the event of a configuration script failure or time out, select whether to ignore all failures and proceed with the deployment. If you do not select this option, when the configuration scripts fail or time out, Launch Wizard will roll back the workload and delete all of the AWS resources created by Launch Wizard. Post-deployment configuration scripts must finish running in 2 hours or less.
- **Configuration script**. Choose to use a script located in Amazon S3 and enter the URL path of the script, or enter a script manually by uploading a script file. To remove the configuration script, choose **Remove script**.

4. After you have entered your deployment settings, choose **Next**.

See the **SAP software installation settings** tab for further information.

Review

- On the **Review** page, review your infrastructure, application, and deployment model settings. If you are satisfied with your selections, choose **Deploy**. If you want to change settings, choose **Previous**.
- When you choose **Deploy**, you are redirected to the **Deployments** page, where you can view the status of your deployment, and also the deployment details.

Clone deployment

You can now clone your SAP deployments created after April 21, 2022.

1. Sign in to https://console.aws.amazon.com/launchwizard.
2. In the Deployments pane on the left, select **SAP**.
3. Choose an existing deployment from the list of deployments and select **Actions** > **Clone deployment**.

Cloning inputs

With a cloned deployment, the following inputs must be provided.

- Enter a unique name for the cloned deployment.
- For SAP landscape infrastructure, you must define the configuration type.
  - You can **Create new configuration** by entering a new **Configuration name** and checking the **Verify connectivity** box.
  - To use the same configuration, select **Apply saved configuration** and choose a configuration from the list.
- The application and database credentials are not carried over. Enter your application and database passwords when prompted.

Deploying an SAP application (AWS CLI)

You can deploy, describe, and delete SAP applications you create using Launch Wizard with the AWS CLI. For more information on the AWS Launch Wizard APIs, see the AWS Launch Wizard API reference.

Topics

- Prerequisites
- AWS CLI examples

Prerequisites

The following requirements must be met before you can use the AWS CLI to create Launch Wizard deployments.

- Install or update the AWS CLI. For more information, see Install or update the latest version of the AWS CLI.
- Complete the getting started requirements in Set up for AWS Launch Wizard for SAP.

AWS CLI examples

The following examples demonstrate how you can use the Launch Wizard API operations with the AWS CLI.

Create a deployment

You can create a deployment for your SAP application using the `CreateDeployment` Launch Wizard API operation. You can use the ListWorkloadDeploymentPatterns operation to discover the supported values for the `--workload-name` and `--deployment-pattern-name` parameters. SAP applications deployed using this API operation can't be cloned from the AWS Launch Wizard console. For more information, see CreateDeployment.

> **Tip**
>
> You can pass inputs to the `specifications` parameter for your deployment as a file for easier usage. For more information on the available specifications for each deployment pattern, including examples, see Deployment specifications.

```
$ aws launch-wizard create-deployment --workload-name SAP --deployment-pattern-
name SapHanaSingle --name ExampleName --region us-east-1 --specifications
 file://hana-single-specifications.json

{
    "deploymentId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
}
```

Delete a deployment

You can delete an SAP deployment using the `DeleteDeployment` Launch Wizard API operation. For more information, see DeleteDeployment.

```
$ aws launch-wizard delete-deployment --deployment-id a1b2c3d4-5678-90ab-cdef-
EXAMPLE11111 --region us-east-1

{
    "status": "DELETE_INITIATING",
    "statusReason": "Finished processing DeleteApp request"
}
```

Get deployment details

You can get deployment details for an SAP deployment using the `GetDeployment` Launch Wizard API operation. For more information, see GetDeployment.

```
$ aws launch-wizard get-deployment --deployment-id a1b2c3d4-5678-90ab-cdef-
EXAMPLE11111 --region us-east-1

{
    "deployment": {
        "name": "ExampleName",
        "id": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
        "workloadName": "SAP",
        "patternName": "SapHanaSingle",
        "status": "COMPLETED",
        "createdAt": "2023-10-10T17:18:49.150000-07:00",
        "specifications": {
            "ApplicationSecurityGroupId": "sg-1234567890abcdef0",
            "AvailabilityZone1PrivateSubnet1Id": "subnet-1234567890abcdef0",
            "CreateSecurityGroup": "No",
            "DatabaseAmiId": "ami-1234567890abcdef0",
            "DatabaseAutomaticRecovery": "Yes",
            "DatabaseDataVolumeType": "gp3",
            "DatabaseHostCount": "1",
            "DatabaseInstanceType": "r3.2xlarge",
            "DatabaseLogVolumeType": "gp3",
            "DatabaseOperatingSystem": "SuSE-Linux-12-SP5-HVM",
            "DatabaseOthersVolumeType": "gp3",
            "DatabasePrimaryHostname": "sapci",
            "DatabaseSecurityGroupId": "sg-1234567890abcdef0",
            "DatabaseSystemId": "HDB",
            "DisableDeploymentRollback": "true",
            "Ec2InstanceRoleName": "AmazonEC2RoleForLaunchWizard",
            "EnableCloudwatchLogs": "Yes",
            "EnableEbsVolumeEncryption": "Yes",
            "EnvironmentType": "production",
            "InstallDatabaseSoftware": "No",
            "KeyPairName": "canary-test",
            "NewSecurityGroupRules": "[{\"type\":\"ip\",\"value\":
\"10.0.0.0/32\"}]",
            "PrimaryAZ": "us-east-1a",
            "SapInstanceNumber": "00",
            "SaveDeploymentArtifacts": "false",
            "SnsTopicArn": "arn:aws:sns:us-east-1:111122223333:ExampleTopic",
            "Timezone": "UTC",
            "VpcId": "vpc-1234567890abcdef0"
        },
        "resourceGroup": "LaunchWizard-SapHanaSingle-ExampleGroup-abcd1234"
    }
}
```

Get workload details

You can get workload details for an SAP deployment using the `GetWorkload` Launch Wizard API operation. For more information, see GetWorkload.

```
$ aws launch-wizard get-workload --workload-name SAP --region us-east-1

{
    "workload": {
        "workloadName": "SAP",
        "displayName": "SAP",
        "description": "AWS Launch Wizard for SAP is a service that guides you
 through the sizing, configuration, and deployment of SAP applications on AWS.",
        "documentationUrl": https://docs.aws.amazon.com/launchwizard/latest/
userguide/launch-wizard-sap.html ,
        "iconUrl": "https://example.com/example.png",
        "status": "ACTIVE"
    }
}
```

List deployments

You can list an SAP deployment using the `ListDeployments` Launch Wizard API operation. For more information, see ListDeployments.

```
$ aws launch-wizard list-deployments --filter
 name=DEPLOYMENT_STATUS,values=IN_PROGRESS --region us-east-1

{
    "deployments": [
        {
            "name": "ExampleName",
            "id": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
            "workloadName": "SAP",
            "patternName": "SapHanaSingle",
            "status": "IN_PROGRESS",
            "createdAt": "2023-04-24T13:10:09.857000-07:00"
        }
    ]
}
```

List deployment events

You can list SAP deployment events using the `ListDeploymentEvents` Launch Wizard API operation. For more information, see ListDeploymentEvents.

```
$ aws launch-wizard list-deployment-events --deployment-id a1b2c3d4-5678-90ab-cdef-
EXAMPLE11111 --region us-east-1

{
    "deploymentEvents": [
        {
            "name": "Create secure parameter",
            "description": "Creates a new secure parameter",
            "status": "COMPLETED",
            "statusReason": "",
            "timestamp": "2021-12-22T16:35:17.227000-08:00"
        },
        {
            "name": "Create resource group",
            "description": "Creates a resource group with all the application
 resources",
            "status": "COMPLETED",
            "statusReason": "",
            "timestamp": "2021-12-22T16:35:17.909000-08:00"
        },
        {
            "name": "Delete application resource group",
            "description": "Deletes the application resource group",
            "status": "COMPLETED",
            "statusReason": "",
            "timestamp": "2023-05-10T15:50:51.156000-07:00"
        },
        {
            "name": "Delete secure parameters",
            "description": "Deletes a secure parameter",
            "status": "COMPLETED",
            "statusReason": "",
            "timestamp": "2023-05-10T15:50:51.451000-07:00"
        }
    ]
}
```

List workloads

You can list workload details for SAP deployments using the `ListWorkloads` Launch Wizard API operation. For more information, see ListWorkloads.

```
$ aws launch-wizard list-workloads --region us-east-1

{
    "workloads": [
        {
            "displayName": "SAP",
            "workloadName": "SAP"
        },
        {
            "displayName": "Exchange Server",
            "workloadName": "ExchangeServer"
        },
        {
            "displayName": "MS SQL Server",
            "workloadName": "SQL"
        },
        {
            "displayName": "Amazon EKS",
            "workloadName": "EKS"
        },
        {
            "displayName": "Microsoft Active Directory",
            "workloadName": "MicrosoftActiveDirectory"
        },
        {
            "displayName": "Microsoft IIS",
            "workloadName": "IIS"
        },
        {
            "displayName": "Remote Desktop Gateway",
            "workloadName": "RDGW"
        }
    ]
}
```

List workload deployment patterns

You can list the available patterns for SAP workloads using the `ListWorkloadDeploymentPatterns` Launch Wizard API operation. For more information, see ListWorkloadDeploymentPatterns.

```
$ aws launch-wizard list-workload-deployment-patterns --workload-name SAP --
region us-east-1

{
    "workloadDeploymentPatterns": [
        {
            "workloadName": "SAP",
            "deploymentPatternName": "SapHanaHA",
            "workloadVersionName": "2023-10-30-23-00-00",
            "displayName": "Cross-AZ SAP HANA database high availability setup",
            "description": "Deploy SAP HANA with high availability configured across
 two Availability Zones.",
            "status": "ACTIVE"
        },
        {
            "workloadName": "SAP",
            "deploymentPatternName": "SapHanaMulti",
            "workloadVersionName": "2023-10-30-23-00-00",
            "displayName": "SAP HANA database on multiple EC2 instances",
            "description": "Deploy SAP HANA in a multi-node, scale-out
 architecture.",
            "status": "ACTIVE"
        },
        {
            "workloadName": "SAP",
            "deploymentPatternName": "SapHanaSingle",
            "workloadVersionName": "2023-10-30-23-00-00",
            "displayName": "SAP HANA database on a single Amazon EC2 instance",
            "description": "Deploy SAP HANA in a single-node, scale-up architecture,
 with up to 24TB of memory.",
            "status": "ACTIVE"
        },
        {
            "workloadName": "SAP",
            "deploymentPatternName": "SapNWOnHanaHA",
            "workloadVersionName": "2023-10-30-23-00-00",
            "displayName": "Cross-AZ SAP NetWeaver system setup",
            "description": "Deploy Amazon EC2 instances for ASCS/ERS and SAP HANA
 databases across two Availability Zones, and spread the deployment of application
 servers across them.",
            "status": "ACTIVE"
        },
        {
            "workloadName": "SAP",
            "deploymentPatternName": "SapNWOnHanaMulti",
            "workloadVersionName": "2023-10-30-23-00-00",
            "displayName": "SAP NetWeaver system on multiple EC2 instances",
            "description": "Deploy an SAP NetWeaver system using a distributed
 deployment model, which includes an ASCS/PAS server, single/multiple SAP HANA
 servers running SAP HANA databases, and multiple application servers.",
            "status": "ACTIVE"
        },
        {
            "workloadName": "SAP",
            "deploymentPatternName": "SapNWOnHanaSingle",
            "workloadVersionName": "2023-10-30-23-00-00",
            "displayName": "SAP NetWeaver on SAP HANA system on a single Amazon EC2
 instance",
            "description": "Deploy an SAP application on the same Amazon EC2
 instance as your SAP HANA Database.",
            "status": "ACTIVE"
        }
    ]
}
```

Monitor Launch Wizard for SAP deployments

You can monitor your Launch Wizard for SAP deployments using the AWS Launch Wizard console and AWS CLI.

Topics

- Monitoring SAP deployments (Console)
- Monitor SAP deployments (AWS CLI)

Monitoring SAP deployments (Console)

Once you have deployed an application, you can monitor it using the Launch Wizard console.

To monitor SAP deployments using the console

1. Access the AWS Launch Wizard console located at https://console.aws.amazon.com/launchwizard.
2. On the left panel, under **Deployments**, choose **SAP**.
3. Under **Application name**, choose the application's name that you are deploying.
4. You can now review the information in the **Details** pane and the **Deployment events** for your application.

Monitor SAP deployments (AWS CLI)

You can monitor your Launch Wizard for SAP deployments using the AWS CLI.

Prerequisites

The following prerequisites are required in order to use the AWS CLI to monitor your Launch Wizard deployment.

- Install or update the AWS CLI, see Install or update the latest version of the AWS CLI.
- Determine the deployment name used for the deployment to monitor. The name was specified during the deployment creation wizard, or as the input for the `CreateDeployment` operation. You can discover the name using the Launch Wizard console, or the `GetDeployment` operation.

Monitoring SAP deployments (AWS CLI)

You can use the `DescribeLogStreams` operation to find the available log streams for the deployment. Once you have the log stream names, you can use the **GetEventLogs** operation to list log events for your deployment related to the log stream you specify.

To monitor SAP deployments using the AWS CLI

1. List the log streams available for the deployment. The streams include the logs and scripts that are run on instances launched for the deployment.

```
$ aws logs describe-log-streams --region us-east-1 --log-group-name
 LaunchWizard-DeploymentName

"logStreams": [
        {
            "logStreamName": "/var/lib/amazon/ssm/packages/AWSSAP-
Backint/2.0.4.768/aws-backint-agent-install-20231027153332.log",
            "creationTime": 1698420842081,
            "firstEventTimestamp": 1698420837015,
            "lastEventTimestamp": 1698420842015,
            "lastIngestionTime": 1698420842277,
            "uploadSequenceToken": "111122223333EXAMPLE",
            "arn": "arn:aws:logs:us-east-1:111122223333:log-group:LaunchWizard-
b2gtehPp:log-stream:/var/lib/amazon/ssm/packages/AWSSAP-Backint/2.0.4.768/aws-
backint-agent-install-20231027153332.log",
            "storedBytes": 0
        },
        {
            "logStreamName": "i-1234567890abcdef0",
            "creationTime": 1698418980895,
            "firstEventTimestamp": 1698418975818,
            "lastEventTimestamp": 1698420271819,
            "lastIngestionTime": 1698420276842,
            "uploadSequenceToken": "111122223333EXAMPLE",
            "arn": "arn:aws:logs:us-east-1:111122223333:log-group:LaunchWizard-
b2gtehPp:log-stream:i-1234567890abcdef0",
            "storedBytes": 0
        },
        ...
]
```

2. Get the log events for the relevant log stream. By default, this operation returns as many log events as can fit in a response size of 1 MB (up to 10,000 log events). You can get additional log events by specifying one of the tokens in a subsequent call.

```
$ userprompt;aws logs get-log-events --region us-east-1 --log-group-name
 LaunchWizard-DeploymentName --log-stream-name LogStreamName

{
    "events": [
        {
            "timestamp": 1698418975818,
            "message": "[ 10/27/2023 03:02:45 PM] [        main:  29] [   INFO]
 process stated.",
            "ingestionTime": 1698418981051
        },
        {
            "timestamp": 1698418975818,
            "message": "[ 10/27/2023 03:02:45 PM] [        main:  30] [   INFO]
 python version 3.8",
            "ingestionTime": 1698418981051
        },
        ...
    ],
    "nextBackwardToken":
 "b/31961209122358285602261756944988674324553373268216709120",
    "nextForwardToken":
 "f/31961209122447488583055879464742346735121166569214640130",
}
```

3. Repeat these steps as necessary to review all of the relevant log streams.

If you find that your deployment has experienced issues, or you want to know more about other logs generated during an SAP deployment, see Troubleshoot AWS Launch Wizard for SAP.

Deploying SAP Web Dispatcher

AWS Launch Wizard supports the deployment of SAP Web Dispatcher as an optional component for Netweaver stack on HANA deployments. SAP Web Dispatcher is deployed in front of your SAP Application Servers to act as the entry point for HTTP(S) request traffic destined for your SAP Application Servers. SAP Web Dispatcher accepts or rejects the request traffic that arrives. Accepted traffic is load balanced among your Application Servers. You can use SAP Web Dispatcher in systems with the following application stacks:

- Advanced Business Application Programming (ABAP) only
- Java only
- ABAP and Java (dual-stack)

Topics

- Architectures for SAP Web Dispatcher
- Post-deployment configuration activities

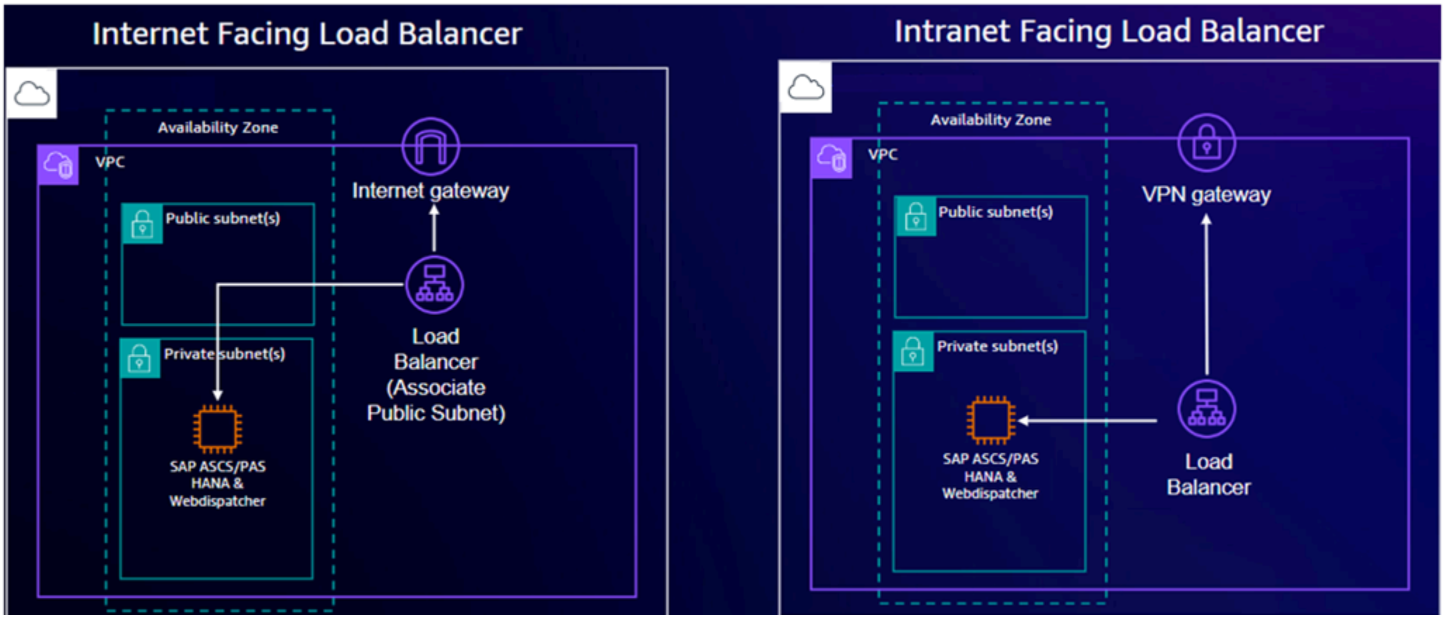
Architectures for SAP Web Dispatcher

SAP Web Dispatcher is available for singe instance, multiple instance, and high availability deployments of Netweaver stack on HANA. The deployment type you specify affects the placement of the component in your architecture.

Single instance deployment

Launch Wizard deploys the component as a standalone component on the same instance where the SAP application and database are deployed.

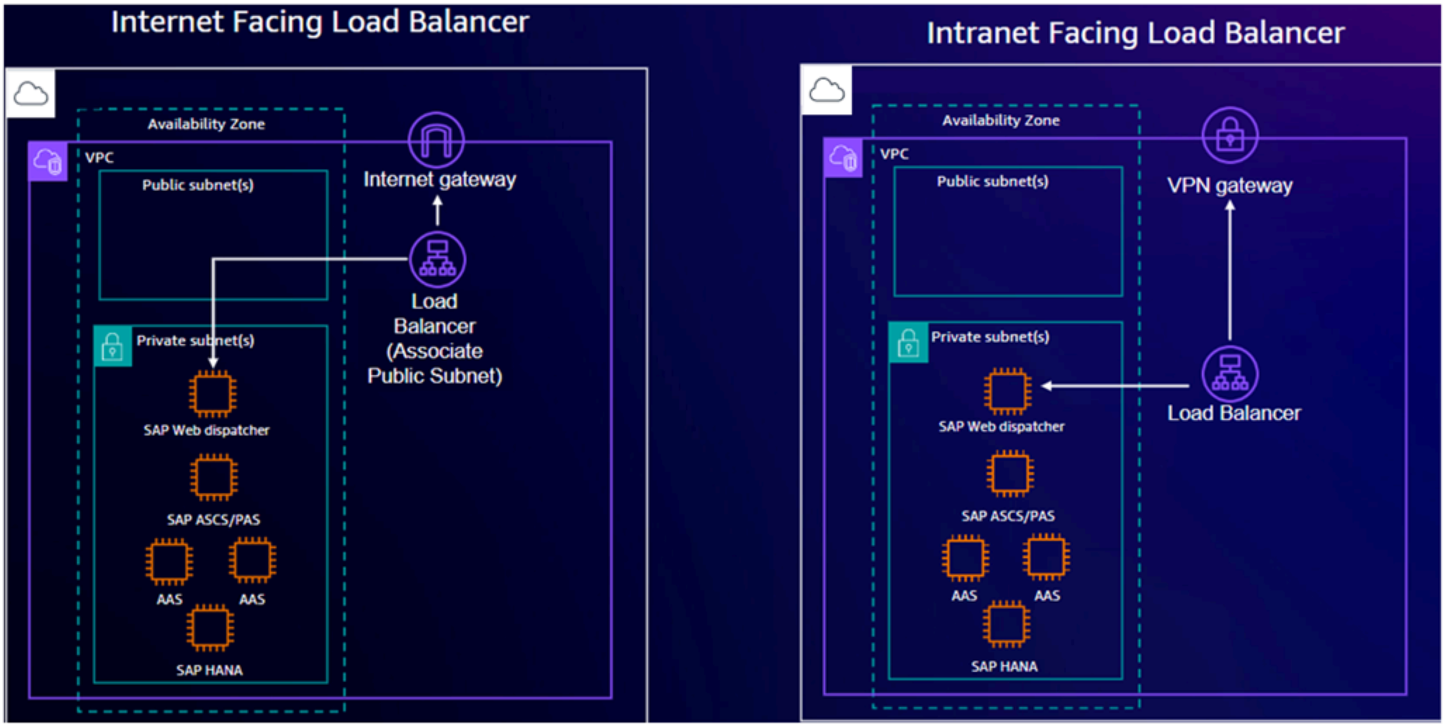
The following diagram depicts an SAP Web Dispatcher deployment using a single instance.


Internet Facing Load Balancer
Availability Zone
VPC
Public subnet(s)
Private subnet(s)
SAP ASCS/PAS HANA & Webdispatcher
Internet gateway
Load Balancer (Associate Public Subnet)
Intranet Facing Load Balancer
Availability Zone
VPC
Public subnet(s)
Private subnet(s)
SAP ASCS/PAS HANA & Webdispatcher
VPN gateway
Load Balancer

Distributed instances deployment

Launch Wizard deploys the component on a separate instance in the same Availability Zone (AZ) where the SAP application and database components are deployed.

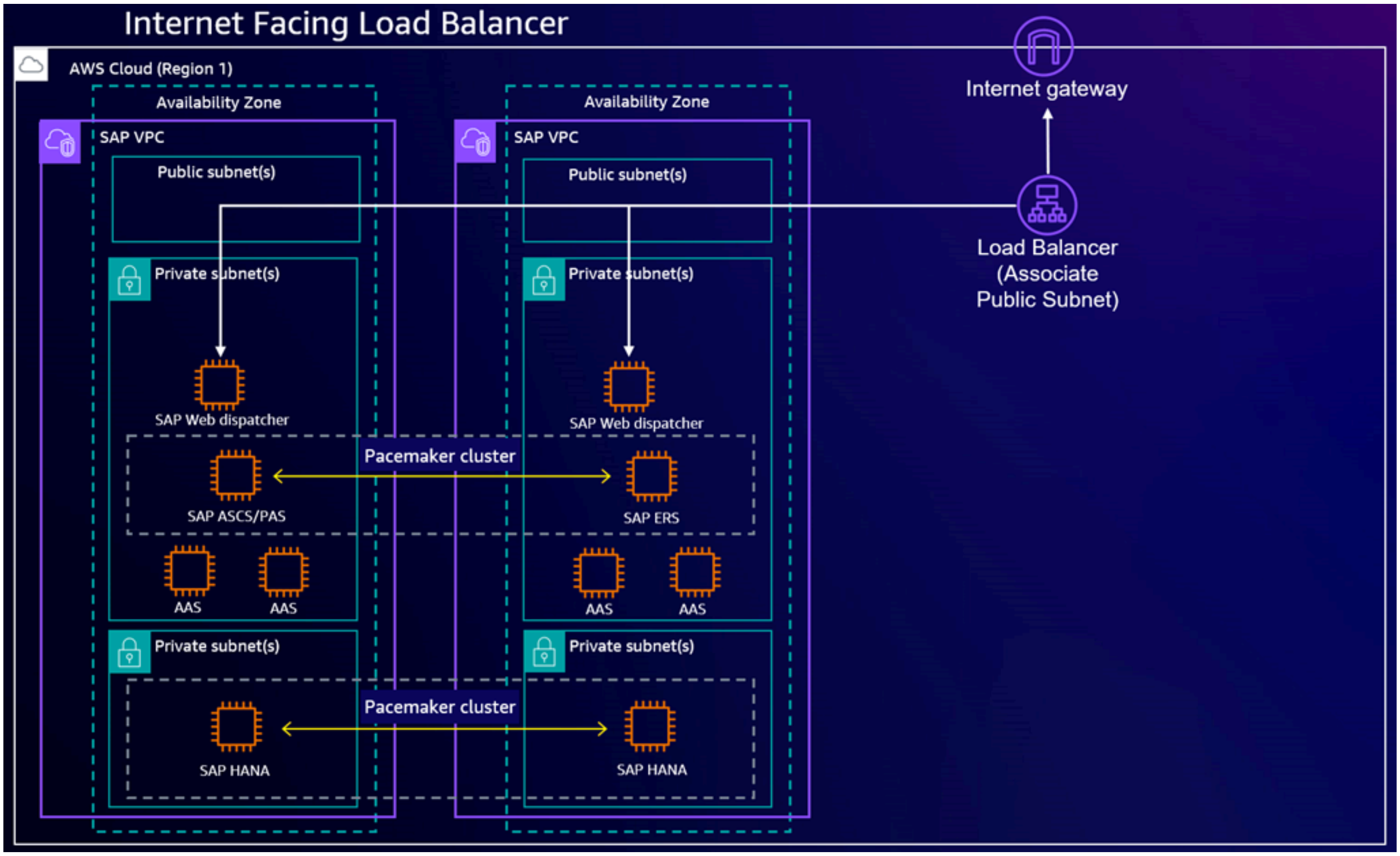
The following diagram depicts an SAP Web Dispatcher deployment using a multiple instances.



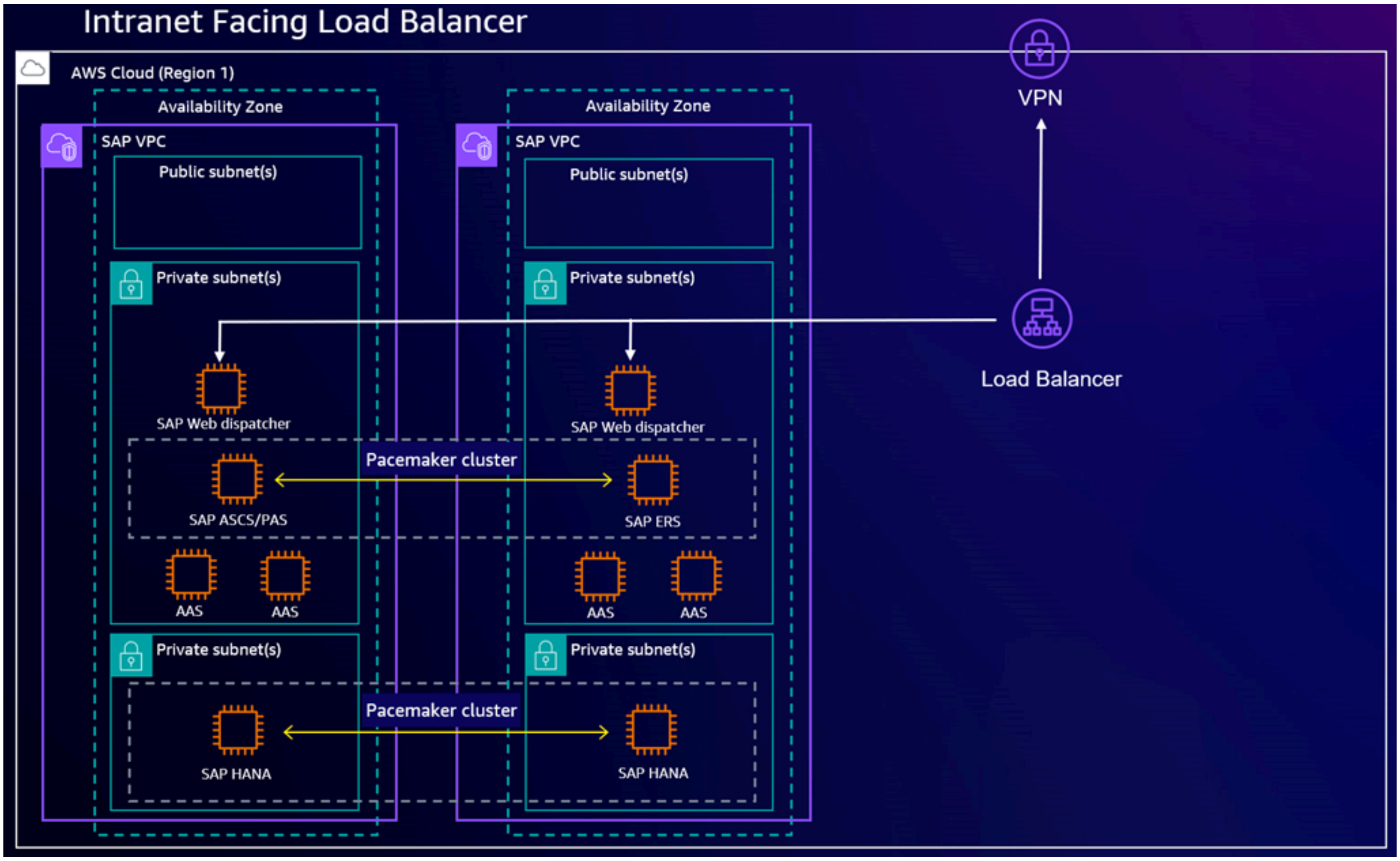
High availability (HA) deployment

Launch Wizard deploys the component on two Amazon EC2 instances, each in a different Availability Zone (AZ). Each AZ also the SAP application and database components. For more information, see High Availability of the SAP Web Dispatcher in the SAP documentation.

The following diagram depicts a highly available SAP Web Dispatcher deployment using multiple instances behind an internet-facing load balancer.



The following diagrams depicts a highly available SAP Web Dispatcher deployment using multiple instances behind an intranet-facing load balancer.



Load balancers for SAP Web Dispatcher

You can optionally deploy an Application Load Balancer or Network Load Balancer with all deployment patterns. The load balancer can be used to accept internet or intranet traffic based on your application requirements. For more information about Elastic Load Balancing, see What is Elastic Load Balancing? in the *Elastic Load Balancing User Guide*.

Network Load Balancer operate at the TCP layer and can handle traffic such as the RFC protocol for system interfaces and File Transfer Protocol (FTP). If your applications need additional context such as HTTP headers, or you plan to integrate other AWS services in your architecture, consider using an Application Load Balancer. Deploying an Application Load Balancer allows you to integrate various other services such as AWS WAF, AWS Certificate Manager (ACM), and AWS CloudFormation.

In Launch Wizard, you will have the option to implement the SSL/TLS termination at load balancer. You must first request a public SSL in ACM or import your own SSL Certificate into ACM to use this option. If you need to do end-to-end HTTPS encryption, you can follow the post-deployment configuration activities. For more information on configuring your deployed resources to support HTTS traffic, see Post-deployment configuration activities.

Post-deployment configuration activities

After your Launch Wizard for SAP deployment with the SAP Web Dispatcher component completes, you must perform several manual configurations to finalize the deployment. These additional configurations are in the customer portion of the AWS Shared Responsibility Model. You should ensure that the changes you make meet your specific security requirements.

Topics

- Validate HTTP(S) listeners are set up
- Activate HTTP(S) services
- Validate target group checks are set up
- Validate SAP Web Dispatcher functionality
- Enable HTTPS communication

Validate HTTP(S) listeners are set up

HTTP(S) listeners must be set up in the SAP System. You can check whether the Internet Communication Framework (ICF) is configured according to your requirements (transaction SMICM for ABAP). All HTTP(S) listeners must use the correct port settings and be in the **Active** status. For more information, see Displaying and Changing Services in the SAP documentation.

Activate HTTP(S) services

For SAP Web Dispatcher and load balancing to function properly, you must activate the following services in the HTTP service tree (transaction SICF for ABAP):

- /sap/public/icman
- /sap/public/icf_info/*
- /sap/public/ping

For ABAP installations, you must activate **/sap/public/ping** to allow load balancers to perform health checks through SAP Web Dispatcher. This prevents the routing of traffic to unhealthy application servers.

For Java installations, you must use **/startPage** as the starting point for the health check endpoint. Once you have full installed and configured the Portal Usage Type, you can adjust this value to **/irj/portal**.

For more information, see Operating SAP Web Dispatcher in the SAP documentation.

Validate target group checks are set up

After you configure load balancing, the target group for your load balancer might end up with unhealthy SAP Web Dispatcher endpoints. You can reregister your SAP Web Dispatcher instances with the correct ports to ensure the load balancer is properly routing traffic. For more information, see Register or deregister targets in the *Elastic Load Balancing User Guide*.

Validate SAP Web Dispatcher functionality

After you configure and validate the related SICF services and validate that the load balancer target groups are healthy, you can validate SAP Web Dispatcher with a web browser.

To access SAP Web Dispatcher

1. Open a web browser on a device that can access the instance running SAP Web Dispatcher.
2. Access your SAP Web Dispatcher web console, replacing values as necessary:

   ```
   http://load-balancer-dns-endpoint:listener-port/sap/wdisp/admin/public/default.html
   ```

3. For **user**, enter **webadm**.
4. For **password**, enter the password you specified in the Launch Wizard deployment.
5. Login to the web console.
6. Choose **Monitor Application Servers** and ensure that you can see all of your Application Servers and that they are using port 80.
7. Choose **Monitor Server Groups** and ensure that you can see all of your server groups.

   For more information, see Area Menu in the SAP documentation.

Enable HTTPS communication

To provide you with the most flexibility to meet your own requirements, SAP Web Dispatcher is deployed behind an Application Load Balancer with only the HTTP protocol enabled by default.

Launch Wizard can implement SSL/TLS termination at the load balancer during deployment, or you can implement end-to-end encryption after the deployment completes.

Enable SSL/TLS termination

With SSL/TLS termination, HTTPS traffic from the end user is decrypted at the load balancer. This traffic is then forwarded to SAP Web Dispatcher and your application servers using the HTTP protocol. Launch Wizard can configure SSL/TLS termination at the load balancer during deployment. To use this option, you will need to specify a load balancer and ACM certificate while configuring the deployment. For more information, see Deploy an SAP application with AWS Launch Wizard.

Enable end-to-end HTTPS encryption

With end-to-end HTTPS encryption, traffic is encrypted to the load balancer and then traffic is re-encrypted at the SAP Web Dispatcher and Application Server instances. You must obtain a certificate from a 3rd party provider before following this procedure.

To configure end-to-end encryption

1. Apply your own certificate to your application servers.

   a. If you have a SAP ABAP application server, apply your certificate to it. For more information, see Configuring the ABAP Platform to Support TLS in the SAP documentation.

   b. If you have a SAP NetWeaver Java application server, apply your certificate to it. For more information, see Configuring Transport Layer Security on SAP NetWeaver AS for Java in the SAP documentation.

2. Apply your own certificate to the SAP Web Dispatcher instance. For more information, see Configure SAP Web Dispatcher to Support SSL in the SAP documentation.

3. Import the certificate that you used in the previous steps into ACM. For more information, see Importing a certificate in the *AWS Certificate Manager User Guide*.

4. Create a listener for your Load Balancer.

   a. If you use Application Load Balancer, you create a HTTPS listener with your certificate imported into ACM as the default certificate. For more information, see Create an HTTPS listener for your Application Load Balancer in the *User Guide for Application Load Balancers*.

   b. If you use Network Load Balancer, you create a TLS Listener. For more information, see TLS listeners for your Network Load Balancer in *User Guide for Network Load Balancers*.

5. Configure an alias or CNAME DNS record for your load balancer using your preferred domain name. For example, your domain name might resemble the following:

   ```
   example.yourdomain.com
   ```

   a. **Example**

      <caption>

      If you use Amazon Route 53, create an Alias record. For more information, see Creating records by using the Amazon Route 53 console in the *Amazon Route 53 Developer Guide*. </caption>

   b. If you use a different DNS provider, create a CNAME record with the provider. For more information, refer to your DNS provider's documentation.

6. Confirm the configuration is working by accessing your endpoint by the DNS name over HTTPS.

   a. For ABAP systems, your URL with the custom DNS name might resemble the following:

      ```
      https://example.yourdomain.com/sap/public/ping
      ```

   b. For Java systems, your URL with the custom DNS name might resemble the following:

      ```
      https://example.yourdomain.com/startPage
      ```

AWS Launch Wizard for SAP tutorials

The following tutorials can help you get started with deploying an application with AWS Launch Wizard.

Topics

Deploy SAP HANA with AWS Launch Wizard

For more information about how to deploy an SAP HANA database on AWS that applies AWS and SAP best practices, watch the following video.

Deploy SAP HANA on AWS in under two hours with AWS Launch Wizard

Deploy SAP S/4HANA with high availability

For more information about how to automate the configuration of a high availability SAP S/4HANA application on AWS using AWS Launch Wizard, watch the following video.

How to deploy SAP S/4HANA with high availability using AWS Launch Wizard

Automate a high availability configuration for SAP HANA

For more information about how to automate a high availability configuration for an SAP HANA database on AWS and then test the failover of the system with AWS Launch Wizard, watch the following video.

Demonstrating failover of SAP S/4HANA with high availability on AWS

Manage application resources with AWS Launch Wizard for SAP

After you have deployed an SAP application, you can manage and update it as follows:

Topics

- Manage deployments
- Delete infrastructure configuration

Manage deployments

1. From the left navigation pane, choose **SAP**.
2. Under the **Deployments** tab, select the check box next to the application that you want to manage, and then choose **Actions**. You can do the following:

    1. **Manage resources on the EC2 console**. You are directed to the Amazon EC2 console, where you can view and manage your SAP application resources, such as Amazon EC2, Amazon EBS, Amazon VPC, Subnets, NAT Gateways, and Elastic IPs.

   2. **View resource group with Systems Manager**. In the Systems Manager console, you can manage your application with built-in integrations through resource groups. Launch Wizard automatically tags your deployment with resource groups. When you access Systems Manager through Launch Wizard, the resources are automatically filtered for you based on your resource group. You can manage, patch, and maintain your applications in Systems Manager.
   3. **View CloudWatch application logs.** You are directed to the CloudWatch dashboard, where you can view your logs.
   4. **View CloudFormation template.** You are directed to the AWS CloudFormation to view the templates created for this deployment.
   5. **View Service Catalog product.** You are directed to the AWS Service Catalog console to view the AWS Service Catalog product that was created for this deployment.

3. Select the check box next to the application that you want to manage, and then choose **Manage Application**:

   - You are redirected to the **Application Detail** page in Application Manager if the deployment is complete, and the application is supported and onboarded to AWS Systems Manager for SAP.
   - You are redirected to the **Register Application** page in Application Manager if the deployment is complete, the application is supported but not onboarded to AWS Systems Manager for SAP.
   - **Manage Application** is disabled if the deployment is not complete, or if the application is unsupported by AWS Systems Manager for SAP.

4. To delete a deployment, select the application that you want to delete, and select **Delete**. You are prompted to confirm the deletion.

   > ⚠ **Important**
   >
   > When you delete a deployment, Launch Wizard attempts to delete only the AWS resources it created in your account as part of the deployment. Launch Wizard considers certain resources, such as security groups, infrastructure configuration templates created during a deployment, and EFS file systems created for a transport directory, as shared resources between multiple deployments. Shared resources are not deleted when you delete a deployment.

5. For more information about your application resources, choose the **Application name**. You can then view the **Deployment events** and **Summary** details for your application using the tabs at the top of the page.

Delete infrastructure configuration

1. From the left navigation pane, choose **SAP**.
2. Under **Saved infrastructure configurations** tab, select the configuration name you want to delete, and then choose **Delete**. You are prompted to confirm the deletion.

> ⚠ **Important**
>
> When you delete an infrastructure configuration, it will not be available for future deployments. Resources created from the configuration, such as VPCs, availability groups, subnets, and key pair names are not deleted.

3. For more information about an infrastructure configuration, choose the **Configuration name**.

Make SAP HANA software available for AWS Launch Wizard to deploy a HANA database

This section describes steps to download the SAP HANA software and upload it to Amazon S3 to make it available for Launch Wizard to deploy a HANA database.

Topics

- Download SAP HANA software
- Upload SAP HANA software to Amazon S3

Download SAP HANA software

To download the SAP HANA software, go to the **SAP Software Downloads** page and download the installation files directly to your local drive.

1. Navigate to the SAP Software Downloads page and log in to your account.
2. Under **Installation and Upgrades**, choose **Access Downloads**>**A-Z index**.

3. Choose **H** in the **Installations and Upgrades** window, and select **SAP HANA Platform Edition** from the list.

4. Choose **SAP HANA Platform Edition**>**Installation**.

5. In the **Downloads** window, find the revision you want to download and download each file to your local drive.

> **Note**
>
> If you do not have access to the software and believe you should, contact the SAP Global Support Customer Interaction Center.

> **Important**
>
> Do not extract the downloaded HANA software. Instead, stage the files in your Amazon S3 bucket as is. Launch Wizard will extract the media and install the software for you.

Upload SAP HANA software to Amazon S3

To upload the SAP HANA software to your Amazon S3 bucket, you must create and set up your destination bucket.

Set up destination bucket

1. Navigate to the Amazon S3 console at https://console.aws.amazon.com/s3.

2. Choose **Create Bucket**.

3. In the **Create Bucket** dialog box, provide a name for your new S3 bucket with the prefix `launchwizard`. Choose the **AWS Region** where you want to create the S3 bucket, which should be a Region that is close to your location, and then choose **Create Bucket**. For detailed information about bucket names and Region selection, see Create a Bucket in the **Amazon S3 Getting Started Guide**.

4. Choose the bucket that you created and, from the **Overview** tab, **Create folder**s to organize your SAP HANA downloads. We recommend that you create a folder for each version of SAP HANA.

5. To add the unextracted SAP HANA files to the appropriate folder, choose **Upload** from the **Overview** tab.

If the path for the specific version of SAP HANA software is `s3://`**`launchwizard`**`hanamedia/SP23` or `s3://`**`launchwizard`**`hanamedia/SP24`, then use this path in the Amazon S3 URL for SAP HANA software (HANAInstallMedia) parameter.

Note

We recommend that you place only the main SAP HANA installation files in the S3 bucket. Do not place multiple SAP HANA versions in the same folder. SAP provides the software as a single .zip file or as multiple files depending on the SAP HANA version (one .exe file and multiple .rar files). Upload them to the version-specific folder that you created.

Make SAP application software available for AWS Launch Wizard to deploy SAP

This section describes steps to upload the SAP application software to Amazon S3 to make it available for Launch Wizard to deploy SAP.

AWS Launch Wizard supports the following software versions. To install a software version, you must provide the SAP software files to Launch Wizard by downloading them from the SAP Support Portal and then uploading them to Amazon S3 (storage class - Standard). To access and use the files for installation, Launch Wizard requires them to be formatted according to the Amazon S3 file path syntax listed in the following table.

Note

The software versions and CD numbers listed in the following table should be used as a reference for all of the software components required to deploy SAP, as well as for how to format the Amazon S3 path to make the software available for Launch Wizard to deploy SAP. Launch Wizard supports NetWeaver 7.50, NetWeaver 7.52, S/4 HANA 1909, S/4 HANA 2020, and BW/4HANA 2.0. You can source the latest SAP software using a script or determine the latest CD numbers of supported applications to use from SAP manually.

- For more information about running a pre-deployment configuration script to source the latest SAP software, refer to the software_download portion of the **aws-sap-automation** repository.
- For more information about finding the latest software from SAP, refer to SAP Maintenance Planner or SAP Software Downloads.

Databases

- Making software available for SAP HANA based applications
- Making software available for SAP ASE based applications

Making software available for SAP HANA based applications

Note

SAP Host Agent 7.22 PL62 or a later version is recommended in a high availability setup for SAP HANA site replication to avoid a known issue with the host agent.

NetWeaver 7.52

CD name	Versions	CD number	Amazon S3 file path
SWPM	SWPM 1.0 latest version	SWPM20SP16_2-80003424.SAR	S3://**<Your SAP software bucket>/<Path representing NW version>/**SWPM
SAPCAR	SAPCAR_1200-70007716.EXE	N/A	S3://**<Your SAP software bucket>/<Path representing**

CD name	Versions	CD number	Amazon S3 file path
			NW version>/ SAPCAR
Exports	NW 7.52	51051806_part1.exe 51051806_part2.rar	S3://**<Your SAP software bucket>/<Path representing NW version>/** Exports
Kernel components	NW 7.53 and later	igsexe_0-70005417.sar igshelper_17-10010245.sar SAPEXE_66-70006642.SAR SAPEXEDB_66-70006641.SAR SAPHOSTAGENT59_59-80004822.SAR	S3://**<Your SAP software bucket>/<Path representing NW version>/** Kernel
SAP HANA Client	2.5	IMDB_CLIENT20_005_111-80002082.SAR	S3://**<Your SAP software bucket>/<Path representing NW version>/** HANA_Client_Software

CD name	Versions	CD number	Amazon S3 file path
SAP Web Dispatcher	7.93	See Note 908097 in the SAP documentation.	S3://**Your SAP software bucket</**webdisp/

The following HANA DB versions are supported (ZIP files only).

Note

The CD versions are for reference only. Use the latest versions available on SAP Software Center.

CD name	Versions	CD number	Amazon S3 file path
hana-20-sp05	51058046	S3://**<Your SAP software bucket>/<Path representing NW version>/**HANA_DB_Software	
hana-20-sp06	51056431	S3://**<Your SAP software bucket>/<Path representing NW version>/**HANA_DB_Software	
hana-20-sp07	51057071	S3://**<Your SAP software bucket>/<Path**	

CD name	Versions	CD number	Amazon S3 file path
		representing NW version>/ HANA_DB_ Software	
hana-20-sp08	51058521	S3://**<Your SAP software bucket>/<Path representing NW version>/** HANA_DB_ Software	

NetWeaver 7.50

CD name	Versions	CD number	Amazon S3 file path
SWPM	SWPM 1.0 latest version	SWPM10SP42_1-20009701.SAR	S3://**<Your SAP software bucket>/<Path representing NW version>/** SWPM
SAPCAR	SAPCAR_1010-70006178.exe	N/A	S3://**<Your SAP software bucket>/<Path representing NW version>/** SAPCAR
Exports	NW 7.50	51050829_3.ZIP	S3://**<Your SAP software bucket>/<Path representing**

CD name	Versions	CD number	Amazon S3 file path
			NW version>/ Exports
Kernel components	NW 7.53 and later	igsexe_12-80003187.sar igshelper_17-10010245.sar SAPEXE_700-80002573.SAR SAPEXEDB_700-80002572.SAR SAPHOSTAGENT49_49-20009394.SAR	S3://**<Your SAP software bucket>/<Path representing NW version>/** Kernel
SAP HANA Client	2.5	IMDB_CLIENT20_005_111-80002082.SAR	S3://**<Your SAP software bucket>/<Path representing NW version>/** HANA_Client_Software
SAP Web Dispatcher	7.93	See Note 908097 in the SAP documentation.	S3://**Your SAP software bucket</** webdisp/

The following HANA DB versions are supported (ZIP files only).

> **Note**
>
> The CD versions are for reference only. Use the latest versions available on SAP Software Center.

CD name	Versions	CD number	Amazon S3 file path
hana-20-sp05	51058046	S3://**<Your SAP software bucket>**/**<Path representing NW version>**/HANA_DB_Software	
hana-20-sp06	51056431	S3://**<Your SAP software bucket>**/**<Path representing NW version>**/HANA_DB_Software	
hana-20-sp07	51057071	S3://**<Your SAP software bucket>**/**<Path representing NW version>**/HANA_DB_Software	
hana-20-sp08	51058521	S3://**<Your SAP software bucket>**/**<Path representing NW version>**/	

CD name	Versions	CD number	Amazon S3 file path
		HANA_DB_ Software	

NetWeaver 750 (JAVA)

CD name	Versions	CD number	Amazon S3 file path
SWPM	SWPM 1.0 latest version	SWPM10SP42_1-20009701.SAR	S3://**<Your SAP software bucket>/<Path representing NW version>/**SWPM
SAPCAR	Latest	N/A	S3://**<Your SAP software bucket>/<Path representing NW version>/**SAPCAR
Exports	NW 7.50	51055106.ZIP	S3://**<Your SAP software bucket>/<Path representing NW version>/**Exports
Kernel components	NW 7.53 and later	igsexe_12-80003187.sar igshelper_17-10010245.sar	S3://**<Your SAP software bucket>/<Path representing NW version>/**Kernel

CD name	Versions	CD number	Amazon S3 file path
		SAPEXE_700-80002573.SAR SAPEXEDB_700-80002572.SAR SAPHOSTAGENT49_49-20009394.SAR SAPJVM8_89-80000202.SAR	
SAP HANA Client	2.5	IMDB_CLIENT20_005_111-80002082.SAR	S3://**<Your SAP software bucket>/<Path representing NW version>/**HANA_Client_Software
SAP Web Dispatcher	7.93	See Note 908097 in the SAP documentation.	S3://**Your SAP software bucket</**webdisp/

The following HANA DB versions are supported (ZIP files only).

Note

The CD versions are for reference only. Use the latest versions available on SAP Software Center.

CD name	Versions	CD number	Amazon S3 file path
hana-20-sp05	51056441	S3://**<Your SAP software bucket>/<Path representing NW version>/** HANA_DB_ Software	
hana-20-sp06	51058046	S3://**<Your SAP software bucket>/<Path representing NW version>/** HANA_DB_ Software	
hana-20-sp07	51057071	S3://**<Your SAP software bucket>/<Path representing NW version>/** HANA_DB_ Software	
hana-20-sp08	51058521	S3://**<Your SAP software bucket>/<Path representing NW version>/** HANA_DB_ Software	

BW/4HANA 2023

CD name	Versions	CD number	Amazon S3 file path
SWPM	SWPM 2.0 latest version	SWPM20SP19_1-80003424.SAR	S3://**<Your SAP software bucket>/<Path representing NW version>/** SWPM
SAPCAR	SAPCAR_1010-70006178.exe	N/A	S3://**<Your SAP software bucket>/<Path representing NW version>/** SAPCAR
Exports	BW4HANA300	BW4HANA400_INST_EXPORT_1.zip through BW4HANA400_INST_EXPORT_9.zip	S3://**<Your SAP software bucket>/<Path representing NW version>/** Exports
Kernel components	785 or later	igsexe_0-70005417.sar igshelper_17-10010245.sar SAPEXE_101-70007807.SAR	S3://**<Your SAP software bucket>/<Path representing NW version>/** Kernel

CD name	Versions	CD number	Amazon S3 file path
		SAPEXEDB_101-70007806.SAR SAPHOSTAGENT54_54-80004822.SAR	
SAP HANA Client	2.22	IMDB_CLIENT20_022_32-80002082.SAR	S3://**<Your SAP software bucket>/<Path representing NW version>/**HANA_Client_Software
SAP Web Dispatcher	7.93	See Note 908097 in the SAP documentation.	S3://**Your SAP software bucket</**webdisp/

The following HANA DB version is supported.

Note

*The CD versions are for reference only. Use the latest versions available on SAP Software Center.

CD name	Versions	CD number	Amazon S3 file path
SAP HANA Database software	hana-20-sp05	51056441	S3://**<Your SAP software bucket>/<Path representing**

CD name	Versions	CD number	Amazon S3 file path
			NW version>/ HANA_DB_ Software
	hana-20-sp06	51056431	S3://**<Your SAP software bucket>/<Path representing NW version>/** HANA_DB_ Software
	hana-20-sp07	51057071	S3://**<Your SAP software bucket>/<Path representing NW version>/** HANA_DB_ Software

BW/4HANA 2021

CD name	Versions	CD number	Amazon S3 file path
SWPM	SWPM 2.0 latest version	SWPM20SP10_3-80003424.SAR	S3://**<Your SAP software bucket>/<Path representing NW version>/** SWPM
SAPCAR	SAPCAR_1010-70006178.exe	N/A	S3://**<Your SAP software bucket>/<Path representing**

CD name	Versions	CD number	Amazon S3 file path
			NW version>/ SAPCAR
Exports	BW4HANA300	BW4HANA300_INST_EXPORT_1.zip through BW4HANA300_INST_EXPORT_8.zip	S3://**<Your SAP software bucket>/<Path representing NW version>/** Exports
Kernel components	785 or later	igsexe_0-70005417.sar igshelper_17-10010245.sar SAPEXE_50-80005374.SAR SAPEXEDB_50-80005373.SAR SAPHOSTAGENT54_54-80004822.SAR	S3://**<Your SAP software bucket>/<Path representing NW version>/** Kernel
SAP HANA Client	2.11	IMDB_CLIENT20_011_14-80002082.SAR	S3://**<Your SAP software bucket>/<Path representing NW version>/** HANA_Client_Software

CD name	Versions	CD number	Amazon S3 file path
SAP Web Dispatcher	7.93	See Note 908097 in the SAP documentation.	S3://**Your SAP software bucket<**/webdisp/

The following HANA DB version is supported.

Note

*The CD versions are for reference only. Use the latest versions available on SAP Software Center.

CD name	Versions	CD number	Amazon S3 file path
SAP HANA Database software	hana-20-sp05	51056441	S3://**<Your SAP software bucket>/<Path representing NW version>/**HANA_DB_Software
	hana-20-sp06	51056431	S3://**<Your SAP software bucket>/<Path representing NW version>/**HANA_DB_Software
	hana-20-sp07	51057071	S3://**<Your SAP software bucket>/<Path**

CD name	Versions	CD number	Amazon S3 file path
			representing NW version>/ HANA_DB_ Software

BW/4HANA 2.0

CD name	Versions	CD number	Amazon S3 file path
SWPM	SWPM 2.0 latest version	SWPM20SP07_0-80003424.SAR	S3://**<Your SAP software bucket>/<Path representing NW version>/** SWPM
SAPCAR	SAPCAR_1010-70006178.exe	N/A	S3://**<Your SAP software bucket>/<Path representing NW version>/** SAPCAR
Exports	BW4HANA 2.0	BW4HANA200_INST_EXPORT_1.zip through BW4HANA200_INST_EXPORT_7.zip	S3://**<Your SAP software bucket>/<Path representing NW version>/** Exports
Kernel components	NW 7.77	igsexe_12-80003187.sar	S3://**<Your SAP software bucket>/<Path representing**

CD name	Versions	CD number	Amazon S3 file path
		igshelper_17-10010245.sar SAPEXE_300-80004393.SAR SAPEXEDB_300-80004392.SAR SAPHOSTAGENT49_49-20009394.SAR	**NW version>/**Kernel
SAP HANA Client	2.5	IMDB_CLIENT20_005_111-80002082.SAR	S3://**<Your SAP software bucket>/<Path representing NW version>/**HANA_Client_Software
SAP Web Dispatcher	7.93	See Note 908097 in the SAP documentation.	S3://**Your SAP software bucket</**webdisp/

The following HANA DB version is supported.

Note

*The CD versions are for reference only. Use the latest versions available on SAP Software Center.

CD name	Versions	CD number	Amazon S3 file path
SAP HANA database software	hana-20-sp05	51056441	S3://**<Your SAP software bucket>/<Path representing NW version>/** HANA_DB_ Software
	hana-20-sp06	51056431	S3://**<Your SAP software bucket>/<Path representing NW version>/** HANA_DB_ Software
	hana-20-sp07	51057071	S3://**<Your SAP software bucket>/<Path representing NW version>/** HANA_DB_ Software

S/4HANA 2023

CD name	Versions	CD number	Amazon S3 file path
SWPM	SWPM 2.0 latest version	SWPM20SP16_0-80003424.SAR	S3://**<Your SAP software bucket>/<Path representing NW version>/** SWPM

CD name	Versions	CD number	Amazon S3 file path
SAPCAR	Latest	N/A	S3://**<Your SAP software bucket>/<Path representing NW version>/** SAPCAR
Exports	S4Core 108	S4CORE108_INST_EXPORT_1.zip through S4CORE108_INST_EXPORT_30.zip	S3://**<Your SAP software bucket>/<Path representing NW version>/** Exports
Kernel components	785 or later	igsexe_4-70005417.sar igshelper_17-10010245.sar SAPEXE_60-70007807.SAR SAPEXEDB_60-70007806.SAR SAPHOSTAGENT62_62-80004822.SAR	S3://**<Your SAP software bucket>/<Path representing NW version>/** Kernel

CD name	Versions	CD number	Amazon S3 file path
SAP HANA Client	2.11	IMDB_CLIENT20_011_14-80002082.SAR	S3://**<Your SAP software bucket>/<Path representing NW version>/**HANA_Client_Software
SAP Web Dispatcher	7.93	See Note 908097 in the SAP documentation.	S3://**Your SAP software bucket</**webdisp/

The following HANA DB version is supported.

Note

*The CD versions are for reference only. Use the latest versions available on SAP Software Center.

CD name	Versions	CD number	Amazon S3 file path
SAP HANA database software	hana-20-sp07	51057071	S3://**<Your SAP software bucket>/<Path representing NW version>/**HANA_DB_Software

S/4HANA Foundations 2023

CD name	Versions	CD number	Amazon S3 file path
SWPM	SWPM 2.0 latest version	SWPM20SP16_0-80003424.SAR	S3://**<Your SAP software bucket>/<Path representing NW version>/** SWPM
SAPCAR	Latest	N/A	S3://**<Your SAP software bucket>/<Path representing NW version>/** SAPCAR
Exports	S4Core 108	S4FND108_INST_EXPORT_1.zip through S4FND108_INST_EXPORT_9.zip	S3://**<Your SAP software bucket>/<Path representing NW version>/** Exports
Kernel components	785 or later	igsexe_4-70005417.sar igshelper_17-10010245.sar SAPEXE_60-70007807.SAR SAPEXEDB_60-70007806.SAR	S3://**<Your SAP software bucket>/<Path representing NW version>/** Kernel

CD name	Versions	CD number	Amazon S3 file path
		SAPHOSTAGENT62_62-80004822.SAR	
SAP HANA Client	2.11	IMDB_CLIENT20_011_14-80002082.SAR	S3://**<Your SAP software bucket>/<Path representing NW version>/**HANA_Client_Software
SAP Web Dispatcher	7.93	See Note 908097 in the SAP documentation.	S3://**Your SAP software bucket</**webdisp/

The following HANA DB version is supported.

> **Note**
>
> *The CD versions are for reference only. Use the latest versions available on SAP Software Center.

CD name	Versions	CD number	Amazon S3 file path
SAP HANA database software	hana-20-sp07	51057071	S3://**<Your SAP software bucket>/<Path representing NW version>/**HANA_DB_Software

S/4HANA 2022

CD name	Versions	CD number	Amazon S3 file path
SWPM	SWPM 2.0 latest version	SWPM20SP10_3-80003424.SAR	S3://**<Your SAP software bucket>/<Path representing NW version>/** SWPM
SAPCAR	Latest	N/A	S3://**<Your SAP software bucket>/<Path representing NW version>/** SAPCAR
Exports	S4Core 105	S4CORE107_INST_EXPORT_1.zip through S4CORE107_INST_EXPORT_30.zip	S3://**<Your SAP software bucket>/<Path representing NW version>/** Exports
Kernel components	785 or later	igsexe_0-70005417.sar igshelper_17-10010245.sar SAPEXE_66-70006642.SAR SAPEXEDB_66-70006641.SAR	S3://**<Your SAP software bucket>/<Path representing NW version>/** Kernel

CD name	Versions	CD number	Amazon S3 file path
		SAPHOSTAGENT59_59-80004822.SAR	
SAP HANA Client	2.11	IMDB_CLIENT20_011_14-80002082.SAR	S3://**<Your SAP software bucket>/<Path representing NW version>/**HANA_Client_Software
SAP Web Dispatcher	7.93	See [Note 908097](#) in the SAP documentation.	S3://**Your SAP software bucket</**webdisp/

The following HANA DB version is supported.

Note

*The CD versions are for reference only. Use the latest versions available on SAP Software Center.

CD name	Versions	CD number	Amazon S3 file path
SAP HANA database software	hana-20-sp05	51056441	S3://**<Your SAP software bucket>/<Path representing NW version>/**HANA_DB_Software

CD name	Versions	CD number	Amazon S3 file path
	hana-20-sp06	51056431	S3://**<Your SAP software bucket>/<Path representing NW version>/** HANA_DB_ Software
	hana-20-sp07	51057071	S3://**<Your SAP software bucket>/<Path representing NW version>/** HANA_DB_ Software

S/4HANA Foundations 2022

CD name	Versions	CD number	Amazon S3 file path
SWPM	SWPM 2.0 latest version	SWPM20SP10_3-80003424.SAR	S3://**<Your SAP software bucket>/<Path representing NW version>/** SWPM
SAPCAR	Latest	N/A	S3://**<Your SAP software bucket>/<Path representing NW version>/** SAPCAR

CD name	Versions	CD number	Amazon S3 file path
Exports	S4Core 105	S4FND107_INST_EXPORT_1.zip through S4FND107_INST_EXPORT_9.zip	S3://**<Your SAP software bucket>/<Path representing NW version>/** Exports
Kernel components	785 or later	igsexe_0-70005417.sar igshelper_17-10010245.sar SAPEXE_66-70006642.SAR SAPEXEDB_66-70006641.SAR SAPHOSTAGENT59_59-80004822.SAR	S3://**<Your SAP software bucket>/<Path representing NW version>/** Kernel
SAP HANA Client	2.11	IMDB_CLIENT20_011_14-80002082.SAR	S3://**<Your SAP software bucket>/<Path representing NW version>/** HANA_Client_Software

CD name	Versions	CD number	Amazon S3 file path
SAP Web Dispatcher	7.93	See Note 908097 in the SAP documentation.	S3://**Your SAP software bucket</**webdisp/

The following HANA DB version is supported.

> **Note**
>
> *The CD versions are for reference only. Use the latest versions available on SAP Software Center.

CD name	Versions	CD number	Amazon S3 file path
SAP HANA database software	hana-20-sp05	51056441	S3://**<Your SAP software bucket>/<Path representing NW version>/**HANA_DB_Software
	hana-20-sp06	51056431	S3://**<Your SAP software bucket>/<Path representing NW version>/**HANA_DB_Software
	hana-20-sp07	51057071	S3://**<Your SAP software bucket>/<Path**

CD name	Versions	CD number	Amazon S3 file path
			representing NW version>/ HANA_DB_Software

S/4HANA 2021

CD name	Versions	CD number	Amazon S3 file path
SWPM	SWPM 2.0 latest version	SWPM20SP10_3-80003424.SAR	S3://**<Your SAP software bucket>/<Path representing NW version>/** SWPM
SAPCAR	SAPCAR_1010-70006178.exe	N/A	S3://**<Your SAP software bucket>/<Path representing NW version>/** SAPCAR
Exports	S4Core 106	S4CORE106_INST_EXPORT_1.zip through S4CORE106_INST_EXPORT_28.zip	S3://**<Your SAP software bucket>/<Path representing NW version>/** Exports
Kernel components	785 or later	igsexe_0-70005417.sar	S3://**<Your SAP software bucket>/<Path representing**

CD name	Versions	CD number	Amazon S3 file path
		igshelper_17-10010245.sar SAPEXE_50-80005374.SAR SAPEXEDB_50-80005373.SAR SAPHOSTAGENT54_54-80004822.SAR	**NW version>/** Kernel
SAP HANA Client	2.11	IMDB_CLIENT20_011_14-80002082.SAR	S3://**<Your SAP software bucket>/<Path representing NW version>/** HANA_Client_Software
SAP Web Dispatcher	7.93	See Note 908097 in the SAP documentation.	S3://**Your SAP software bucket</** webdisp/

The following HANA DB version is supported.

Note

*The CD versions are for reference only. Use the latest versions available on SAP Software Center.

CD name	Versions	CD number	Amazon S3 file path
SAP HANA database software	hana-20-sp05	51056441	S3://**<Your SAP software bucket>/<Path representing NW version>/** HANA_DB_ Software
	hana-20-sp06	51056431	S3://**<Your SAP software bucket>/<Path representing NW version>/** HANA_DB_ Software
	hana-20-sp07	51057071	S3://**<Your SAP software bucket>/<Path representing NW version>/** HANA_DB_ Software

S/4HANA Foundations 2021

CD name	Versions	CD number	Amazon S3 file path
SWPM	SWPM 2.0 latest version	SWPM20SP10_3-80003424.SAR	S3://**<Your SAP software bucket>/<Path representing NW version>/** SWPM

CD name	Versions	CD number	Amazon S3 file path
SAPCAR	Latest	N/A	S3://**<Your SAP software bucket>/<Path representing NW version>/** SAPCAR
Exports	S4Core 105	S4FND106_INST_EXPORT_1.zip through S4FND106_INST_EXPORT_8.zip	S3://**<Your SAP software bucket>/<Path representing NW version>/** Exports
Kernel components	785 or later	igsexe_0-70005417.sar igshelper_17-10010245.sar SAPEXE_66-70006642.SAR SAPEXEDB_66-70006641.SAR SAPHOSTAGENT59_59-80004822.SAR	S3://**<Your SAP software bucket>/<Path representing NW version>/** Kernel

CD name	Versions	CD number	Amazon S3 file path
SAP HANA Client	2.11	IMDB_CLIENT20_011_14-80002082.SAR	S3://**<Your SAP software bucket>/<Path representing NW version>/**HANA_Client_Software
SAP Web Dispatcher	7.93	See Note 908097 in the SAP documentation.	S3://**Your SAP software bucket</**webdisp/

The following HANA DB version is supported.

Note

*The CD versions are for reference only. Use the latest versions available on SAP Software Center.

CD name	Versions	CD number	Amazon S3 file path
SAP HANA database software	hana-20-sp05	51056441	S3://**<Your SAP software bucket>/<Path representing NW version>/**HANA_DB_Software
	hana-20-sp06	51056431	S3://**<Your SAP software bucket>/<Path**

CD name	Versions	CD number	Amazon S3 file path
			representing NW version>/ HANA_DB_ Software
	hana-20-sp07	51057071	S3://**<Your SAP software bucket>/<Path representing NW version>/** HANA_DB_ Software

S/4HANA 2020

CD name	Versions	CD number	Amazon S3 file path
SWPM	SWPM 2.0 latest version	SWPM20SP07_0-80003424.SAR	S3://**<Your SAP software bucket>/<Path representing NW version>/** SWPM
SAPCAR	SAPCAR_1010-70006178.exe	N/A	S3://**<Your SAP software bucket>/<Path representing NW version>/** SAPCAR
Exports	S4Core 105	S4CORE105_INST_EXPORT_1.zip through S4CORE105	S3://**<Your SAP software bucket>/<Path representing**

CD name	Versions	CD number	Amazon S3 file path
		_INST_EXPORT_24.zip	**NW version>/** Exports
Kernel components	NW 7.77	igsexe_0-70005417.sar igshelper_17-10010245.sar SAPEXE_15-70005283.SAR SAPEXEDB_15-70005282.SAR SAPHOSTAGENT49_49-20009394.SAR	S3://**<Your SAP software bucket>/<Path representing NW version>/** Kernel
SAP HANA Client	2.5	IMDB_CLIENT20_005_111-80002082.SAR	S3://**<Your SAP software bucket>/<Path representing NW version>/** HANA_Client_Software
SAP Web Dispatcher	7.93	See Note 908097 in the SAP documentation.	S3://**Your SAP software bucket</** webdisp/

The following HANA DB version is supported.

Note

*The CD versions are for reference only. Use the latest versions available on SAP Software Center.

CD name	Versions	CD number	Amazon S3 file path
SAP HANA database software	hana-20-sp05	51056441	S3://**<Your SAP software bucket>/<Path representing NW version>/**HANA_DB_Software
	hana-20-sp06	51056431	S3://**<Your SAP software bucket>/<Path representing NW version>/**HANA_DB_Software
	hana-20-sp07	51057071	S3://**<Your SAP software bucket>/<Path representing NW version>/**HANA_DB_Software

S/4HANA 1909

CD name	Versions	CD number	Amazon S3 file path
SWPM	SWPM 2.0 latest version	SWPM20SP07_0-80003424.SAR	S3://**<Your SAP software bucket>/<Path representing NW version>/** SWPM
SAPCAR	SAPCAR_1010-70006178.exe	N/A	S3://**<Your SAP software bucket>/<Path representing NW version>/** SAPCAR
Exports	S4Core 104	S4CORE104_INST_EXPORT_1.zip through S4CORE104_INST_EXPORT_25.zip	S3://**<Your SAP software bucket>/<Path representing NW version>/** Exports
Kernel components	NW 7.77	igsexe_12-80003187.sar igshelper_17-10010245.sar SAPEXE_300-80004393.SAR	S3://**<Your SAP software bucket>/<Path representing NW version>/** Kernel

CD name	Versions	CD number	Amazon S3 file path
		SAPEXEDB_300-80004392.SAR SAPHOSTAGENT49_49-20009394.SAR	
SAP HANA Client	2.5	IMDB_CLIENT20_005_111-80002082.SAR	S3://**<Your SAP software bucket>/<Path representing NW version>/**HANA_Client_Software
SAP Web Dispatcher	7.93	See Note 908097 in the SAP documentation.	S3://**Your SAP software bucket</**webdisp/

The following HANA DB version is supported.

> **Note**
>
> *The CD versions are for reference only. Use the latest versions available on SAP Software Center.

CD name	Versions	CD number	Amazon S3 file path
SAP HANA database software	hana-20-sp05	51056441	S3://**<Your SAP software bucket>/<Path representing**

CD name	Versions	CD number	Amazon S3 file path
			NW version>/ HANA_DB_Software
	hana-20-sp06	51056431	S3://**<Your SAP software bucket>/<Path representing NW version>/** HANA_DB_Software
	hana-20-sp07	51057071	S3://**<Your SAP software bucket>/<Path representing NW version>/** HANA_DB_Software

Solution Manager 7.2

CD name	Versions	CD number	Amazon S3 file path
SWPM	SWPM 1.0 latest version	SWPM10SP42_1-20009701.SAR	S3://**<Your SAP software bucket>/<Path representing SolutionManager version>**/SWPM
SAPCAR	SAPCAR_1010-70006178.exe	N/A	S3://**<Your SAP software bucket>/<Path**

CD name	Versions	CD number	Amazon S3 file path
			representing SolutionManager version>/SAPCAR
Exports	SAP Solution Manager 7.2	51054655_1.ZIP…51054655_4.ZIP igsexe_12-80003187.sar igshelper_17-10010245.sar	S3://**<Your SAP software bucket>/<Path representing SolutionManager version>**/Exports
Kernel components	NW 7.53 and later	SAPEXE_700-80002573.SAR SAPEXEDB_700-80002572.SAR SAPHOSTAGENT49_49-20009394.SAR SAPJVM8_89-80000202.SAR	S3://**<Your SAP software bucket>/<Path representing SolutionManager version>**/Kernel

CD name	Versions	CD number	Amazon S3 file path
SAP HANA Client	2.5	IMDB_CLIENT20_005_111-80002082.SAR	S3://**<Your SAP software bucket>/<Path representing SolutionManager version>/**HANA_Client_Software
SAP Web Dispatcher	7.93	See Note 908097 in the SAP documentation.	S3://**Your SAP software bucket</**webdisp/

The following HANA DB versions are supported (ZIP files only).

> **Note**
>
> The CD versions are for reference only. Use the latest versions available on SAP Software Center.

CD name	Versions	CD number	Amazon S3 file path
hana-20-sp05	51058046	S3://**<Your SAP software bucket>/<Path representing NW version>/**HANA_DB_Software	

CD name	Versions	CD number	Amazon S3 file path
hana-20-sp06	51056431	S3://**<Your SAP software bucket>/<Path representing NW version>/**HANA_DB_Software	
hana-20-sp07	51057071	S3://**<Your SAP software bucket>/<Path representing NW version>/**HANA_DB_Software	
hana-20-sp08	51058521	S3://**<Your SAP software bucket>/<Path representing NW version>/**HANA_DB_Software	

Making software available for SAP ASE based applications

NetWeaver 7.52

CD name	Versions	CD number	Amazon S3 file path
SWPM	SWPM 1.0 latest version	SWPM10SP38_4-20009701.SAR	S3://**<Your SAP software bucket>/<Path representing**

CD name	Versions	CD number	Amazon S3 file path
			NW version>/SWPM
SAPCAR	Latest	SAPCAR_1115-70006178.EXE	S3://**<Your SAP software bucket>/<Path representing NW version>/**SAPCAR
Exports	NW 7.52	51051806_part1.exe 51051806_part2.rar	S3://**<Your SAP software bucket>/<Path representing NW version>/**Exports

CD name	Versions	CD number	Amazon S3 file path
Kernel components	NW 7.53 and later	igsexe_12-80003187.sar igshelper_17-10010245.sar SAPEXE_700-80002573.SAR SAPEXEDB_1000-80002616.SAR SAPHOSTAGENT61_61-80004822.SAR SAPJVM8_95-80000202.SAR	S3://**<Your SAP software bucket>/<Path representing NW version>/**Kernel

The following SAP ASE DB versions are supported (ZIP files only).

Note

The CD versions are for reference only. Use the latest versions available on SAP Software Center.

CD name	Versions	CD number	Amazon S3 file path
SAP ASE Database software	SAP ASE 16.0.04.04	51056521_1.ZIP	S3://**<Your SAP software bucket>/<Path representing NW version>/**SAPASE_DB_Software

NetWeaver 7.50

CD name	Versions	CD number	Amazon S3 file path
SWPM	SWPM 1.0 latest version	SWPM10SP38_4-20009701.SAR	S3://**<Your SAP software bucket>/<Path representing NW version>/**SWPM
SAPCAR	Latest	SAPCAR_1115-70006178.EXE	S3://**<Your SAP software bucket>/<Path representing NW version>/**SAPCAR
Exports	NW 7.50	51050829_3.ZIP	S3://**<Your SAP software bucket>/<Path representing NW version>/**Exports

CD name	Versions	CD number	Amazon S3 file path
Kernel components	NW 7.53 and later	igsexe_12-80003187.sar igshelper_17-10010245.sar SAPEXE_700-80002573.SAR SAPEXEDB_1000-80002616.SAR SAPHOSTAGENT61_61-80004822.SAR SAPJVM8_95-80000202.SAR	S3://**<Your SAP software bucket>/<Path representing NW version>/**Kernel

The following SAP ASE DB versions are supported (ZIP files only).

Note

The CD versions are for reference only. Use the latest versions available on SAP Software Center.

CD name	Versions	CD number	Amazon S3 file path
SAP ASE Database software	SAP ASE 16.0.04.04	51056521_1.ZIP	S3://**<Your SAP software bucket>/<Path representing NW version>/** SAPASE_DB_Software

NetWeaver 750 (JAVA)

CD name	Versions	CD number	Amazon S3 file path
SWPM	SWPM 1.0 latest version	SWPM10SP42_1-20009701.SAR	S3://**<Your SAP software bucket>/<Path representing NW version>/** SWPM
SAPCAR	Latest	SAPCAR_1115-70006178.EXE	S3://**<Your SAP software bucket>/<Path representing NW version>/** SAPCAR
Exports	NW 7.50	51055106.ZIP	S3://**<Your SAP software bucket>/<Path representing NW version>/** Exports

CD name	Versions	CD number	Amazon S3 file path

`Kernel components`	NW 7.53 and later	`igsexe_12-80003187.sar`
		`igshelper_17-10010245.sar`
		`SAPEXE_700-80002573.SAR`
		`SAPEXEDB_1000-80002616.SAR`
		`SAPHOSTAGENT61_61-80004822.SAR`
		`SAPJVM8_95-80000202.SAR`
			`S3://`**`<Your SAP software bucket>/<Path representing NW version>/`**`Kernel`

The following SAP ASE DB versions are supported (ZIP files only).

Note

The CD versions are for reference only. Use the latest versions available on SAP Software Center.

CD name	Versions	CD number	Amazon S3 file path
SAP ASE Database software	SAP ASE 16.0.04.04	51056521_1.ZIP	S3://**<Your SAP software bucket>/<Path representing NW version>/**SAPASE_DB_Software

Solution Manager 7.2

CD name	Versions	CD number	Amazon S3 file path
SWPM	SWPM 1.0 latest version	SWPM10SP42_1-20009701.SAR	S3://**<Your SAP software bucket>/<Path representing SolutionManager version>**/SWPM
SAPCAR	Latest	SAPCAR_1115-70006178.EXE	S3://**<Your SAP software bucket>/<Path representing SolutionManager version>**/SAPCAR
Exports	SAP Solution Manager 7.2	51054655_1.ZIP 51054655_2.ZIP 51054655_3.ZIP 51054655_4.ZIP	S3://**<Your SAP software bucket>/<Path representing SolutionManager**

CD name	Versions	CD number	Amazon S3 file path
			version>/ Exports
Kernel components	NW 7.53 and later	igsexe_12-80003187.sar igshelper_17-10010245.sar SAPEXE_700-80002573.SAR SAPEXEDB_1000-80002616.SAR SAPHOSTAGENT61_61-80004822.SAR SAPJVM8_95-80000202.SAR	S3://**<Your SAP software bucket>/<Path representing NW version>/** Kernel

The following SAP ASE DB versions are supported (ZIP files only).

Note

The CD versions are for reference only. Use the latest versions available on SAP Software Center.

CD name	Versions	CD number	Amazon S3 file path
SAP ASE Database software	SAP ASE 16.0.04.04	51056521_1.ZIP	S3://**<Your SAP software bucket>/<Path representing NW version>/**SAPASE_DB_Software

Repeat SAP application deployments using deployment artifacts created with AWS Launch Wizard

This section contains information about how to repeat deployments using deployment artifacts created with Launch Wizard. The artifacts include AWS Service Catalog products and AWS CloudFormation templates.

Deployment artifact topics

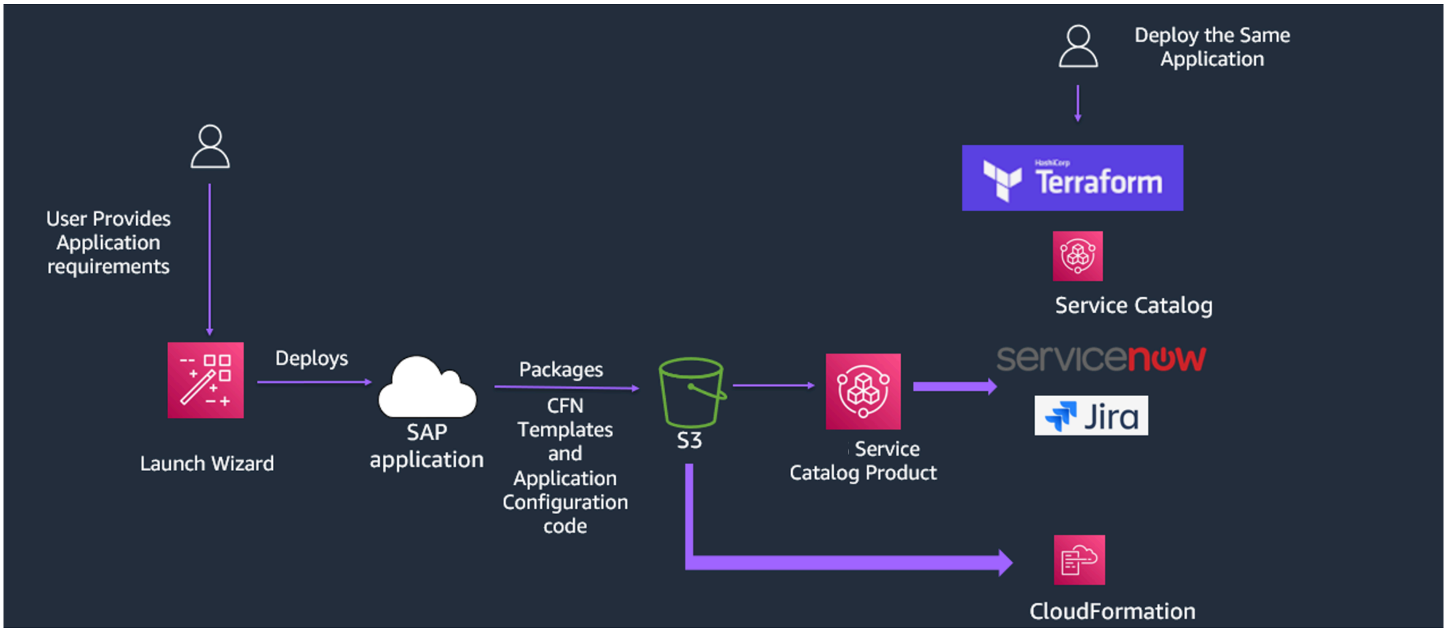
- How AWS Launch Wizard integration with AWS Service Catalog works
- Launch AWS Service Catalog products created with AWS Launch Wizard
- Launch AWS Service Catalog products with ServiceNow
- Launch AWS Service Catalog products with Jira
- Launch AWS Service Catalog products with Terraform
- Launch CloudFormation templates created in Launch Wizard

How AWS Launch Wizard integration with AWS Service Catalog works

AWS Launch Wizard creates AWS Service Catalog products from successful deployments. The AWS Service Catalog products contain CloudFormation templates and associated application configuration scripts, which are stored in Amazon S3. You can use the AWS Service Catalog products, along with integrations offered by AWS Service Catalog, with third-party products, such as ServiceNow, Jira, or Terraform. Or, you can use the CloudFormation templates and application configuration scripts saved in Amazon S3 to deploy SAP applications that meet the requirements of organizational deployment and governance policies.

In addition to supporting deployments using CloudFormation templates, AWS Service Catalog, and multiple deployment tools supported by AWS Service Catalog, AWS Launch Wizard creates a point-in-time snapshot of the code used to deploy and configure SAP applications at the time of the deployment. You can use the code "as is" for consistent repeated deployments, or you can use the code as a baseline and update it to meet specific application requirements.

AWS Launch Wizard creates a default Launch Wizard portfolio and products within the portfolio. An AWS Service Catalog product is created for each deployment and given a name that corresponds to the Launch Wizard deployment name.



Deploying SAP applications with Launch Wizard, CloudFormation, AWS Service Catalog, and third-party applications

Launch AWS Service Catalog products created with AWS Launch Wizard

This section contains information to help you set up for and access AWS Service Catalog products created with AWS Launch Wizard to launch those products. It also contains information about how to create a launch constraint so that you don't have to use your own IAM credentials to launch and manage AWS Service Catalog products.

Topics for launching AWS Service Catalog products

Set up to launch AWS Service Catalog products created with AWS Launch Wizard

This section provides the required steps to grant permissions to the user group. This requirement must be met to access AWS Service Catalog products created with Launch Wizard to launch those products.

Grant AWS Service Catalog permissions to the user group

1. Navigate to the [AWS Identity and Access Management console](https://console.aws.amazon.com/iam/).
2. Choose **User groups** from the left navigation pane.
3. Choose **Create group.**
4. For **User group name**, enter `Endusers`.
5. Enter `AWSServiceCatalog` in the search box to filter the policy list.
6. Select the check box next to the **AWSServiceCatalogEndUserFullAccess** policy. You can optionally choose **AWSServiceCatalogEndUserReadOnlyAccess** if you prefer to grant the user only read-only access. Choose **Create group**
7. To add a new user to the group, in the left navigation pane, choose **Users**.
8. Choose **Add user**.
9. Enter a **User name**.
10. Select **AWS Management Console access**.
11. Choose **Next: Permissions**.
12. Choose **Add user to group**.
13. Select the check box next to the **Endusers** group, then choose **Next:Tags**.
14. Choose **Next: Review**. On the **Review** page, choose **Create user**. Download or copy the credentials, then choose **Close**.

Create a launch constraint

A launch constraint specifies the AWS Identity and Access Management role that AWS Service Catalog assumes when a user launches a product. It is associated with products in the portfolio. If you do not use launch constraints, you must launch and manage products using your own IAM credentials. These credentials must have permissions to use CloudFormation, AWS Service Catalog, and any other AWS services used by the products. Using a launch constraint allows you to limit the permissions of a user to the minimum required for a product.

To create a launch constraint, complete the steps in the following procedure. Perform Step 2 for each of the following listed policies.

Create the launch role

AWS Service Catalog launch constraint policy 1

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "applicationinsights:*",
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": "resource-groups:List*",
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "route53:ChangeResourceRecordSets",
                "route53:GetChange",
                "route53:ListResourceRecordSets",
                "route53:ListHostedZones",
                "route53:ListHostedZonesByName"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "s3:ListAllMyBuckets",
                "s3:ListBucket",
                "s3:GetBucketLocation"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "kms:ListKeys",
                "kms:ListAliases"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "cloudwatch:List*",
                "cloudwatch:Get*",
                "cloudwatch:Describe*"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "ec2:CreateInternetGateway",
                "ec2:CreateNatGateway",
                "ec2:CreateVpc",
                "ec2:CreateKeyPair",
                "ec2:CreateRoute",
                "ec2:CreateRouteTable",
                "ec2:CreateSubnet"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "ec2:AllocateAddress",
                "ec2:AllocateHosts",
                "ec2:AssignPrivateIpAddresses",
                "ec2:AssociateAddress",
                "ec2:CreateDhcpOptions",
                "ec2:CreateEgressOnlyInternetGateway",
                "ec2:CreateNetworkInterface",
                "ec2:CreateVolume",
                "ec2:CreateVpcEndpoint",
                "ec2:CreateTags",
                "ec2:DeleteTags",
                "ec2:RunInstances",
                "ec2:StartInstances",
                "ec2:ModifyInstanceAttribute",
                "ec2:ModifySubnetAttribute",
                "ec2:ModifyVolumeAttribute",
                "ec2:ModifyVpcAttribute",
                "ec2:AssociateDhcpOptions",
                "ec2:AssociateSubnetCidrBlock",
                "ec2:AttachInternetGateway",
                "ec2:AttachNetworkInterface",
                "ec2:AttachVolume",
                "ec2:DeleteDhcpOptions",
                "ec2:DeleteInternetGateway",
                "ec2:DeleteKeyPair",
                "ec2:DeleteNatGateway",
                "ec2:DeleteSecurityGroup",
                "ec2:DeleteVolume",
                "ec2:DeleteVpc",
                "ec2:DetachInternetGateway",
                "ec2:DetachVolume",
                "ec2:DeleteSnapshot",
                "ec2:AssociateRouteTable",
                "ec2:AssociateVpcCidrBlock",
                "ec2:DeleteNetworkAcl",
                "ec2:DeleteNetworkInterface",
                "ec2:DeleteNetworkInterfacePermission",
                "ec2:DeleteRoute",
                "ec2:DeleteRouteTable",
                "ec2:DeleteSubnet",
                "ec2:DetachNetworkInterface",
                "ec2:DisassociateAddress",
                "ec2:DisassociateVpcCidrBlock",
                "ec2:GetLaunchTemplateData",
                "ec2:ModifyNetworkInterfaceAttribute",
                "ec2:ModifyVolume",
                "ec2:AuthorizeSecurityGroupEgress",
                "ec2:GetConsoleOutput",
                "ec2:GetPasswordData",
                "ec2:ReleaseAddress",
                "ec2:ReplaceRoute",
                "ec2:ReplaceRouteTableAssociation",
                "ec2:RevokeSecurityGroupEgress",
                "ec2:RevokeSecurityGroupIngress",
                "ec2:DisassociateIamInstanceProfile",
                "ec2:DisassociateRouteTable",
                "ec2:DisassociateSubnetCidrBlock",
                "ec2:ModifyInstancePlacement",
                "ec2:DeletePlacementGroup",
                "ec2:CreatePlacementGroup",
                "elasticfilesystem:DeleteFileSystem",
                "elasticfilesystem:DeleteMountTarget",
                "ds:AddIpRoutes",
                "ds:CreateComputer",
                "ds:CreateMicrosoftAD",
                "ds:DeleteDirectory"
            ],
            "Resource": "*"
        }
    ]
}
```

Service Catalog launch constraint policy 2

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "cloudformation:DescribeStack*",
                "cloudformation:Get*",
                "cloudformation:ListStacks",
                "cloudformation:SignalResource",
                "cloudformation:DeleteStack"
            ],
            "Resource": [
                "arn:aws:cloudformation:*:*:stack/*/*",
                "arn:aws:cloudformation:*:*:stack/ApplicationInsights*/*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "ec2:StopInstances",
                "ec2:TerminateInstances"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "iam:CreateInstanceProfile",
                "iam:DeleteInstanceProfile",
                "iam:RemoveRoleFromInstanceProfile",
                "iam:AddRoleToInstanceProfile"
            ],
            "Resource": [
                "arn:aws:iam::*:role/service-role/AmazonEC2RoleForLaunchWizard*",
                "arn:aws:iam::*:instance-profile/*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "iam:PassRole"
            ],
            "Resource": [
                "arn:aws:iam::*:role/service-role/AmazonEC2RoleForLaunchWizard*",
                "arn:aws:iam::*:role/service-role/
AmazonLambdaRoleForLaunchWizard*",
                "arn:aws:iam::*:instance-profile/*"
            ],
            "Condition": {
                "StringEqualsIfExists": {
                    "iam:PassedToService": [
                        "lambda.amazonaws.com",
                        "ec2.amazonaws.com"
                    ]
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "autoscaling:AttachInstances",
                "autoscaling:CreateAutoScalingGroup",
                "autoscaling:CreateLaunchConfiguration",
                "autoscaling:DeleteAutoScalingGroup",
                "autoscaling:DeleteLaunchConfiguration",
                "autoscaling:UpdateAutoScalingGroup",
                "logs:CreateLogStream",
                "logs:DeleteLogGroup",
                "logs:DeleteLogStream",
                "logs:DescribeLog*",
                "logs:PutLogEvents",
                "resource-groups:CreateGroup",
                "resource-groups:DeleteGroup",
                "sns:ListSubscriptionsByTopic",
                "sns:Publish",
                "ssm:DeleteDocument",
                "ssm:DeleteParameter*",
                "ssm:DescribeDocument*",
                "ssm:GetDocument",
                "ssm:PutParameter"
            ],
            "Resource": [
                "arn:aws:resource-groups:*:*:group/*",
                "arn:aws:sns:*:*:*",
                "arn:aws:autoscaling:*:*:autoScalingGroup:*:autoScalingGroupName/
LaunchWizard*",

 "arn:aws:autoscaling:*:*:launchConfiguration:*:launchConfigurationName/
LaunchWizard*",
                "arn:aws:ssm:*:*:parameter/LaunchWizard*",
                "arn:aws:ssm:*:*:document/LaunchWizard*",
                "arn:aws:logs:*:*:log-group:*:*:*",
                "arn:aws:logs:*:*:log-group:LaunchWizard*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": "ssm:SendCommand",
            "Resource": "*",
            "Condition": {
                "ForAllValues:StringLike": {
                    "aws:TagKeys": "LaunchWizard*"
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "logs:DeleteLogStream",
                "logs:GetLogEvents",
                "logs:PutLogEvents",
                "ssm:AddTagsToResource",
                "ssm:DescribeDocument",
                "ssm:GetDocument",
                "ssm:ListTagsForResource",
                "ssm:RemoveTagsFromResource"
            ],
            "Resource": [
                "arn:aws:logs:*:*:log-group:*:*:*",
                "arn:aws:logs:*:*:log-group:LaunchWizard*",
                "arn:aws:ssm:*:*:parameter/LaunchWizard*",
                "arn:aws:ssm:*:*:document/LaunchWizard*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "autoscaling:Describe*",
                "cloudformation:DescribeAccountLimits",
                "cloudformation:DescribeStackDriftDetectionStatus",
                "cloudformation:List*",
                "cloudformation:GetTemplateSummary",
                "cloudformation:ValidateTemplate",
                "ds:Describe*",
                "ds:ListAuthorizedApplications",
                "ec2:Describe*",
                "ec2:Get*",
                "iam:GetRole",
                "iam:GetRolePolicy",
                "iam:GetUser",
                "iam:GetPolicyVersion",
                "iam:GetPolicy",
                "iam:List*",
                "logs:CreateLogGroup",
                "logs:GetLogDelivery",
                "logs:GetLogRecord",
                "logs:ListLogDeliveries",
                "resource-groups:Get*",
                "resource-groups:List*",
                "servicequotas:GetServiceQuota",
                "servicequotas:ListServiceQuotas",
                "sns:ListSubscriptions",
                "sns:ListTopics",
                "ssm:CreateDocument",
                "ssm:DescribeAutomation*",
                "ssm:DescribeInstanceInformation",
                "ssm:DescribeParameters",
                "ssm:GetAutomationExecution",
                "ssm:GetCommandInvocation",
                "ssm:GetParameter*",
                "ssm:GetConnectionStatus",
                "ssm:ListCommand*",
                "ssm:ListDocument*",
                "ssm:ListInstanceAssociations",
                "ssm:SendAutomationSignal",
                "ssm:StartAutomationExecution",
                "ssm:StopAutomationExecution",
                "tag:Get*"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": "logs:GetLog*",
            "Resource": [
                "arn:aws:logs:*:*:log-group:*:*:*",
                "arn:aws:logs:*:*:log-group:LaunchWizard*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "cloudformation:List*",
                "cloudformation:Describe*"
            ],
            "Resource": "arn:aws:cloudformation:*:*:stack/LaunchWizard*/"
        },
        {
            "Effect": "Allow",
            "Action": [
                "iam:CreateServiceLinkedRole"
            ],
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "iam:AWSServiceName": [
                        "autoscaling.amazonaws.com",
                        "application-insights.amazonaws.com",
                        "events.amazonaws.com"
                    ]
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "sqs:TagQueue",
                "sqs:GetQueueUrl",
                "sqs:AddPermission",
                "sqs:ListQueues",
                "sqs:DeleteQueue",
                "sqs:GetQueueAttributes",
                "sqs:ListQueueTags",
                "sqs:CreateQueue",
                "sqs:SetQueueAttributes"
            ],
            "Resource": "arn:aws:sqs:*:*:*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "cloudwatch:PutMetricAlarm",
                "iam:GetInstanceProfile",
                "cloudwatch:DeleteAlarms",
                "cloudwatch:DescribeAlarms"
            ],
            "Resource": [
                "arn:aws:cloudwatch:*:*:alarm:*",
                "arn:aws:iam::*:instance-profile/*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "cloudformation:CreateStack",
                "route53:ListHostedZones",
                "ec2:CreateSecurityGroup",
                "ec2:AuthorizeSecurityGroupIngress",
                "elasticfilesystem:DescribeFileSystems",
                "elasticfilesystem:CreateFileSystem",
                "elasticfilesystem:CreateMountTarget",
                "elasticfilesystem:DescribeMountTargets",
                "elasticfilesystem:DescribeMountTargetSecurityGroups"
            ],
            "Resource": "*"
        }
    ]
}
```

Service Catalog launch constraint policy 3

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "s3:GetObject",
                "s3:PutObject"
            ],
            "Resource": [
                "arn:aws:s3:::launchwizard*",
                "arn:aws:s3:::launchwizard*/*",
                "arn:aws:s3:::aws-sap-data-provider/config.properties"
            ]
        },
        {
            "Effect": "Allow",
            "Action": "cloudformation:TagResource",
            "Resource": "*",
            "Condition": {
                "ForAllValues:StringLike": {
                    "aws:TagKeys": "LaunchWizard*"
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "s3:CreateBucket",
                "s3:PutBucketVersioning",
                "s3:DeleteBucket",
                "lambda:CreateFunction",
                "lambda:DeleteFunction",
                "lambda:GetFunction",
                "lambda:GetFunctionConfiguration",
                "lambda:InvokeFunction"
            ],
            "Resource": [
                "arn:aws:lambda:*:*:function:*",
                "arn:aws:s3:::launchwizard*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "dynamodb:CreateTable",
                "dynamodb:DescribeTable",
                "dynamodb:DeleteTable"
            ],
            "Resource": "arn:aws:dynamodb:*:*:table/*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "secretsmanager:CreateSecret",
                "secretsmanager:DeleteSecret",
                "secretsmanager:TagResource",
                "secretsmanager:UntagResource",
                "secretsmanager:PutResourcePolicy",
                "secretsmanager:DeleteResourcePolicy",
                "secretsmanager:ListSecretVersionIds",
                "secretsmanager:GetSecretValue"
            ],
            "Resource": "arn:aws:secretsmanager:*:*:secret:*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "secretsmanager:GetRandomPassword",
                "secretsmanager:ListSecrets"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "ssm:CreateOpsMetadata"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": "ssm:DeleteOpsMetadata",
            "Resource": "arn:aws:ssm:*:*:opsmetadata/aws/ssm/LaunchWizard*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "sns:CreateTopic",
                "sns:DeleteTopic",
                "sns:Subscribe",
                "sns:Unsubscribe"
            ],
            "Resource": "arn:aws:sns:*:*:*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "fsx:UntagResource",
                "fsx:TagResource",
                "fsx:DeleteFileSystem",
                "fsx:ListTagsForResource"
            ],
            "Resource": "*",
            "Condition": {
                "StringLike": {
                    "aws:ResourceTag/Name": "LaunchWizard*"
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "fsx:CreateFileSystem"
            ],
            "Resource": "*",
            "Condition": {
                "StringLike": {
                    "aws:RequestTag/Name": [
                        "LaunchWizard*"
                    ]
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "fsx:DescribeFileSystems"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "s3:GetObject"
            ],
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "s3:ExistingObjectTag/servicecatalog:provisioning": "true"
                }
            }
        }
    ]
}
```

1. Sign in to the AWS Management Console and open the IAM console at https://console.aws.amazon.com/iam/.
2. Perform the following substeps individually for each of the three policies previously listed.

    a. In the left navigation pane, choose **Policies** > **Create policy**.

    b. On the **Create policy** page, choose the **JSON** tab.

    c. Copy each of the previous policies and paste each into the **Policy Document** JSON text box, replacing the placeholder text.

    d. Choose **Next: Tags** > **Next: Review**.

    e. Enter a **Policy Name**.

    f. Choose **Create policy**.

3. In the left navigation pane, choose **Roles**, then choose **Create role**.
4. Under **Select type of trusted entity**, choose **AWS service** > **Service Catalog**.
5. Select the **Service Catalog** use case, then choose **Next:Permissions**.
6. Search for the three policies that you added in Step 2 and select the check boxes next to them.
7. Choose **Next: Tags**.
8. Choose **Next: Review**.
9. Enter `LaunchWizardServiceCatalogProductsLaunchRole` for the **Role name**.
10. Choose **Create role**.

Create launch constraint

1. Navigate to the AWS Service Catalog console.
2. In the left navigation pane, under **Administration**, choose **Portfolios**.
3. Choose the portfolio named **Launch Wizard Service Catalog portfolio**, which is the default portfolio.
4. Under **Constraints**, choose **Create Constraints**.
5. Select the **Product** to which to apply the constraint.
6. Select **Launch** as the **Constraint type**.
7. Select the IAM role that you created in the procedure for creating a launch role.
8. Choose **Create**.

Access AWS Service Catalog products created with AWS Launch Wizard

Perform the following steps to access AWS Service Catalog products created with AWS Launch Wizard.

In the AWS Service Catalog administrator console, the **Portfolio details** page lists the portfolio settings. From this page, you can manage the products in a portfolio, grant users access to products, and apply `TagOptions` and constraints. You can manage products from the **Products** page.

Access Service Catalog products as a Service Catalog Admin user

1. Navigate to the AWS Service Catalog console.

2. In the left navigation pane, under **Administration**, choose **Portfolios**.
3. Choose the portfolio named **AWS Launch Wizard Products**, which is the default portfolio created by Launch Wizard.
4. Choose **AWS Launch Wizard products**.
5. The product created by Launch Wizard using CloudFormation templates and user inputs is named **[LW Deployment Name]-[Deployment Type]**. You can create a new version by choosing **Create new version**.
6. You can associate tags or apply product-specific tags as needed.

Access Service Catalog products as an IAM user

1. Navigate to the AWS Service Catalog console.
2. In the left navigation pane. under **Home**, choose **Products**.
3. Search for the Launch Wizard SAP product that you saved from the Launch Wizard deployment, and select it. The product, won't be visible to any user who has not been granted access to it. To grant access to the product, see Granting Access to Users.
4. Choose **Launch product**.
5. You will be directed to the AWS Service Catalog **Launching** page, which resembles CloudFormation. Most of the parameters are specified using your defaults. Enter or replace the default values as you require, including passwords and SAPSIDs.
6. After you verify the parameters, choose **Launch product** to start the creation of the CloudFormation stack.

AWS Service Catalog deployment errors

For AWS Service Catalog deployments completed prior to February 7, 2022, perform the following steps to remove the `AmazonLambdaRolePolicyForLaunchWizardSAP` policy from the `AmazonLambdaRoleForLaunchWizard` role, and add a new inline policy. Deployments completed after February 7, 2022 do not require you to perform these steps.

1. Sign in to the AWS Management Console and open the IAM console at https://console.aws.amazon.com/iam/.
2. Choose **Roles** from the left navigation pane.
3. Search for the `AmazonLambdaRoleForLaunchWizard`. Select the policy to view the attached permissions.

4. Check whether the `AmazonLambdaRolePolicyForLaunchWizardSAP` policy is attached to this role. If it is attached, remove the policy by selecting the check box next to it, and choose **Remove**.

5. Add the following inline policy by choosing **Add permissions**>**Create inline policy**, and entering the policy in the **JSON** tab of the **Create policy** wizard.

   JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "ssm:GetParameter"
      ],
      "Resource": "arn:aws:ssm:*::parameter/LaunchWizard*"
    },
    {
      "Effect": "Allow",
      "Action": [
        "ssm:GetDocument",
        "ssm:sendCommand"
      ],
      "Resource": "arn:aws:ssm:*::document/AWS-RunShellScript"
    },
    {
      "Effect": "Allow",
      "Action": [
        "ssm:SendCommand"
      ],
      "Resource": [
      "arn:aws:ec2:*:111122223333:instance/*"
      ],
      "Condition": {
        "StringLike": {
          "ssm:resourceTag/LaunchWizardApplicationType": "*"
        }
      }
    }
  ]
}
```

6. Choose **Review policy**, enter a name for the policy, and choose **Create policy**.

Launch AWS Service Catalog products with ServiceNow

ServiceNow users can natively browse and provision AWS Service Catalog products created with AWS Launch Wizard by using the AWS Management Connector for ServiceNow.

Prerequisites for using ServiceNow to launch products:

- You must create a deployment using Launch Wizard by choosing the **Create an AWS Service Catalog product** option in the infrastructure settings in Launch Wizard. For more information, see Define infrastructure.
- You must install the AWS Service Catalog Connector for ServiceNow. For details about how to install the Connector, see AWS Service Management Connector for ServiceNow.
- You must complete the set up steps to launch AWS Service Catalog products.
- You must create a launch constraint.

For more information about how to integrate AWS products into your ServiceNow Portal using the AWS Service Catalog Connector, watch the following video.

Integrate AWS Products into Your ServiceNow Portal via the AWS Service Management Connector

Launch AWS Service Catalog products with Jira

AWS Service Catalog products created with AWS Launch Wizard can be integrated with Jira workflows. You can use the AWS Service Catalog Connector for Jira to natively provision and operate AWS Service Catalog products created with Launch Wizard by using Atlassian's Jira Service Management. This workflow simplifies product request actions for Jira Service Management users and provides Jira Service Management governance and oversight over AWS products.

To use Jira to launch products, you must follow these prerequisites:

- Create a deployment using Launch Wizard by choosing the **Create an AWS Service Catalog product** option in the infrastructure settings in Launch Wizard. For more information, see Define infrastructure.

- Install the AWS Service Catalog Connector for Jira. For information about how to install the Connector, see AWS Service Management Connector for ServiceNow.
- Complete the set up steps to launch AWS Service Catalog products.
- Complete the steps to create a launch constraint.

For more information about how to integrate AWS products into your Jira Service Management portal using the AWS Service Catalog Connector, watch the following video.

Integrate AWS Products into Your Jira Service Management Portal

Launch AWS Service Catalog products with Terraform

The official HashiCorp AWS provider supports AWS Service Catalog resources. You can launch products created with Launch Wizard and saved to AWS Service Catalog using Terraform. Or, you can integrate the products with their existing Terraform workflows. Administrators can create AWS Service Catalog portfolios and add Launch Wizard products to them using Terraform.

Prerequisites for using Terraform to launch products:

- You must create a deployment using Launch Wizard by choosing the **Create an AWS Service Catalog product** option in the infrastructure settings in Launch Wizard. For more information, see Define infrastructure.
- The Terraform user that authenticates the AWS account must have access to the AWS Service Catalog products. For more information, see AWS Provider in the Terraform documentation.
- The IAM user that authenticates the AWS account must have permissions to use the AWS Service Catalog products created by Launch Wizard. For steps to grant access to users, see Granting Access to Users in the *AWS Service Catalog User Guide*.

The Terraform resource named `aws_servicecatalog_product` is used to launch the AWS Service Catalog product created with Launch Wizard.

Example Terraform script

The following example Terraform script launches a single node HANA database instance with a single node HANA product (`prod-abc1234546`) created with Launch Wizard using the product version ID (`pa-xyz12345`). In this example, the hostname for HANA and the SID for HANA DB are passed to override the defaults, and the remaining parameters are set to the defaults in the AWS Service Catalog product.

```
terraform {
  required_providers {
    aws = {
      source  = "hashicorp/aws"
      version = "~> 3.54.0"
    }
  }
}
provider "aws" {
  profile = "default"
  region  = "us-east-1"
}
resource "random_id" "id" {
  byte_length = 8
}
#Confirm user can launch product  - No launch paths has many reasons for failure.
resource "aws_servicecatalog_provisioned_product" "singlenodehana" {
  name = "tef-${random_id.id.hex}"
  product_id = "prod-abc1234546"
  provisioning_artifact_id = "pa-xyz12345"
  provisioning_parameters {
        key = "HANASID"
        value = "HDB"
  }
  provisioning_parameters {
        key = "HANAHostname"
        value = "saphanadev"
  }
tags = {
    TFLaunched= "True"
  }
}
```

Note that the environment variables authentication mechanism is used in this example.

Launch CloudFormation templates created in Launch Wizard

You can launch CloudFormation stacks from the CloudFormation templates that you saved from your successful Launch Wizard deployments. Perform the following steps to find and launch your CloudFormation templates created with Launch Wizard.

To create a launch constraint, complete the steps in the following procedure. Perform Step 2 for each of the following listed policies.

Attach required policies to IAM user

Service Catalog launch constraint policy 1

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "applicationinsights:*",
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": "resource-groups:List*",
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "route53:ChangeResourceRecordSets",
                "route53:GetChange",
                "route53:ListResourceRecordSets",
                "route53:ListHostedZones",
                "route53:ListHostedZonesByName"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "s3:ListAllMyBuckets",
                "s3:ListBucket",
                "s3:GetBucketLocation"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "kms:ListKeys",
                "kms:ListAliases"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "cloudwatch:List*",
                "cloudwatch:Get*",
                "cloudwatch:Describe*"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "ec2:CreateInternetGateway",
                "ec2:CreateNatGateway",
                "ec2:CreateVpc",
                "ec2:CreateKeyPair",
                "ec2:CreateRoute",
                "ec2:CreateRouteTable",
                "ec2:CreateSubnet"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "ec2:AllocateAddress",
                "ec2:AllocateHosts",
                "ec2:AssignPrivateIpAddresses",
                "ec2:AssociateAddress",
                "ec2:CreateDhcpOptions",
                "ec2:CreateEgressOnlyInternetGateway",
                "ec2:CreateNetworkInterface",
                "ec2:CreateVolume",
                "ec2:CreateVpcEndpoint",
                "ec2:CreateTags",
                "ec2:DeleteTags",
                "ec2:RunInstances",
                "ec2:StartInstances",
                "ec2:ModifyInstanceAttribute",
                "ec2:ModifySubnetAttribute",
                "ec2:ModifyVolumeAttribute",
                "ec2:ModifyVpcAttribute",
                "ec2:AssociateDhcpOptions",
                "ec2:AssociateSubnetCidrBlock",
                "ec2:AttachInternetGateway",
                "ec2:AttachNetworkInterface",
                "ec2:AttachVolume",
                "ec2:DeleteDhcpOptions",
                "ec2:DeleteInternetGateway",
                "ec2:DeleteKeyPair",
                "ec2:DeleteNatGateway",
                "ec2:DeleteSecurityGroup",
                "ec2:DeleteVolume",
                "ec2:DeleteVpc",
                "ec2:DetachInternetGateway",
                "ec2:DetachVolume",
                "ec2:DeleteSnapshot",
                "ec2:AssociateRouteTable",
                "ec2:AssociateVpcCidrBlock",
                "ec2:DeleteNetworkAcl",
                "ec2:DeleteNetworkInterface",
                "ec2:DeleteNetworkInterfacePermission",
                "ec2:DeleteRoute",
                "ec2:DeleteRouteTable",
                "ec2:DeleteSubnet",
                "ec2:DetachNetworkInterface",
                "ec2:DisassociateAddress",
                "ec2:DisassociateVpcCidrBlock",
                "ec2:GetLaunchTemplateData",
                "ec2:ModifyNetworkInterfaceAttribute",
                "ec2:ModifyVolume",
                "ec2:AuthorizeSecurityGroupEgress",
                "ec2:GetConsoleOutput",
                "ec2:GetPasswordData",
                "ec2:ReleaseAddress",
                "ec2:ReplaceRoute",
                "ec2:ReplaceRouteTableAssociation",
                "ec2:RevokeSecurityGroupEgress",
                "ec2:RevokeSecurityGroupIngress",
                "ec2:DisassociateIamInstanceProfile",
                "ec2:DisassociateRouteTable",
                "ec2:DisassociateSubnetCidrBlock",
                "ec2:ModifyInstancePlacement",
                "ec2:DeletePlacementGroup",
                "ec2:CreatePlacementGroup",
                "elasticfilesystem:DeleteFileSystem",
                "elasticfilesystem:DeleteMountTarget",
                "ds:AddIpRoutes",
                "ds:CreateComputer",
                "ds:CreateMicrosoftAD",
                "ds:DeleteDirectory"
            ],
            "Resource": "*"
        }
    ]
}
```

Service Catalog launch constraint policy 2

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "cloudformation:DescribeStack*",
                "cloudformation:Get*",
                "cloudformation:ListStacks",
                "cloudformation:SignalResource",
                "cloudformation:DeleteStack"
            ],
            "Resource": [
                "arn:aws:cloudformation:*:*:stack/*/*",
                "arn:aws:cloudformation:*:*:stack/ApplicationInsights*/*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "ec2:StopInstances",
                "ec2:TerminateInstances"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "iam:CreateInstanceProfile",
                "iam:DeleteInstanceProfile",
                "iam:RemoveRoleFromInstanceProfile",
                "iam:AddRoleToInstanceProfile"
            ],
            "Resource": [
                "arn:aws:iam::*:role/service-role/AmazonEC2RoleForLaunchWizard*",
                "arn:aws:iam::*:instance-profile/*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "iam:PassRole"
            ],
            "Resource": [
                "arn:aws:iam::*:role/service-role/AmazonEC2RoleForLaunchWizard*",
                "arn:aws:iam::*:role/service-role/
AmazonLambdaRoleForLaunchWizard*",
                "arn:aws:iam::*:instance-profile/*"
            ],
            "Condition": {
                "StringEqualsIfExists": {
                    "iam:PassedToService": [
                        "lambda.amazonaws.com",
                        "ec2.amazonaws.com"
                    ]
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "autoscaling:AttachInstances",
                "autoscaling:CreateAutoScalingGroup",
                "autoscaling:CreateLaunchConfiguration",
                "autoscaling:DeleteAutoScalingGroup",
                "autoscaling:DeleteLaunchConfiguration",
                "autoscaling:UpdateAutoScalingGroup",
                "logs:CreateLogStream",
                "logs:DeleteLogGroup",
                "logs:DeleteLogStream",
                "logs:DescribeLog*",
                "logs:PutLogEvents",
                "resource-groups:CreateGroup",
                "resource-groups:DeleteGroup",
                "sns:ListSubscriptionsByTopic",
                "sns:Publish",
                "ssm:DeleteDocument",
                "ssm:DeleteParameter*",
                "ssm:DescribeDocument*",
                "ssm:GetDocument",
                "ssm:PutParameter"
            ],
            "Resource": [
                "arn:aws:resource-groups:*:*:group/*",
                "arn:aws:sns:*:*:*",
                "arn:aws:autoscaling:*:*:autoScalingGroup:*:autoScalingGroupName/
LaunchWizard*",

 "arn:aws:autoscaling:*:*:launchConfiguration:*:launchConfigurationName/
LaunchWizard*",
                "arn:aws:ssm:*:*:parameter/LaunchWizard*",
                "arn:aws:ssm:*:*:document/LaunchWizard*",
                "arn:aws:logs:*:*:log-group:*:*:*",
                "arn:aws:logs:*:*:log-group:LaunchWizard*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": "ssm:SendCommand",
            "Resource": "*",
            "Condition": {
                "ForAllValues:StringLike": {
                    "aws:TagKeys": "LaunchWizard*"
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "logs:DeleteLogStream",
                "logs:GetLogEvents",
                "logs:PutLogEvents",
                "ssm:AddTagsToResource",
                "ssm:DescribeDocument",
                "ssm:GetDocument",
                "ssm:ListTagsForResource",
                "ssm:RemoveTagsFromResource"
            ],
            "Resource": [
                "arn:aws:logs:*:*:log-group:*:*:*",
                "arn:aws:logs:*:*:log-group:LaunchWizard*",
                "arn:aws:ssm:*:*:parameter/LaunchWizard*",
                "arn:aws:ssm:*:*:document/LaunchWizard*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "autoscaling:Describe*",
                "cloudformation:DescribeAccountLimits",
                "cloudformation:DescribeStackDriftDetectionStatus",
                "cloudformation:List*",
                "cloudformation:GetTemplateSummary",
                "cloudformation:ValidateTemplate",
                "ds:Describe*",
                "ds:ListAuthorizedApplications",
                "ec2:Describe*",
                "ec2:Get*",
                "iam:GetRole",
                "iam:GetRolePolicy",
                "iam:GetUser",
                "iam:GetPolicyVersion",
                "iam:GetPolicy",
                "iam:List*",
                "logs:CreateLogGroup",
                "logs:GetLogDelivery",
                "logs:GetLogRecord",
                "logs:ListLogDeliveries",
                "resource-groups:Get*",
                "resource-groups:List*",
                "servicequotas:GetServiceQuota",
                "servicequotas:ListServiceQuotas",
                "sns:ListSubscriptions",
                "sns:ListTopics",
                "ssm:CreateDocument",
                "ssm:DescribeAutomation*",
                "ssm:DescribeInstanceInformation",
                "ssm:DescribeParameters",
                "ssm:GetAutomationExecution",
                "ssm:GetCommandInvocation",
                "ssm:GetParameter*",
                "ssm:GetConnectionStatus",
                "ssm:ListCommand*",
                "ssm:ListDocument*",
                "ssm:ListInstanceAssociations",
                "ssm:SendAutomationSignal",
                "ssm:StartAutomationExecution",
                "ssm:StopAutomationExecution",
                "tag:Get*"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": "logs:GetLog*",
            "Resource": [
                "arn:aws:logs:*:*:log-group:*:*:*",
                "arn:aws:logs:*:*:log-group:LaunchWizard*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "cloudformation:List*",
                "cloudformation:Describe*"
            ],
            "Resource": "arn:aws:cloudformation:*:*:stack/LaunchWizard*/"
        },
        {
            "Effect": "Allow",
            "Action": [
                "iam:CreateServiceLinkedRole"
            ],
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "iam:AWSServiceName": [
                        "autoscaling.amazonaws.com",
                        "application-insights.amazonaws.com",
                        "events.amazonaws.com"
                    ]
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "sqs:TagQueue",
                "sqs:GetQueueUrl",
                "sqs:AddPermission",
                "sqs:ListQueues",
                "sqs:DeleteQueue",
                "sqs:GetQueueAttributes",
                "sqs:ListQueueTags",
                "sqs:CreateQueue",
                "sqs:SetQueueAttributes"
            ],
            "Resource": "arn:aws:sqs:*:*:*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "cloudwatch:PutMetricAlarm",
                "iam:GetInstanceProfile",
                "cloudwatch:DeleteAlarms",
                "cloudwatch:DescribeAlarms"
            ],
            "Resource": [
                "arn:aws:cloudwatch:*:*:alarm:*",
                "arn:aws:iam::*:instance-profile/*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "cloudformation:CreateStack",
                "route53:ListHostedZones",
                "ec2:CreateSecurityGroup",
                "ec2:AuthorizeSecurityGroupIngress",
                "elasticfilesystem:DescribeFileSystems",
                "elasticfilesystem:CreateFileSystem",
                "elasticfilesystem:CreateMountTarget",
                "elasticfilesystem:DescribeMountTargets",
                "elasticfilesystem:DescribeMountTargetSecurityGroups"
            ],
            "Resource": "*"
        }
    ]
}
```

Service Catalog launch constraint policy 3

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "s3:GetObject",
                "s3:PutObject"
            ],
            "Resource": [
                "arn:aws:s3:::launchwizard*",
                "arn:aws:s3:::launchwizard*/*",
                "arn:aws:s3:::aws-sap-data-provider/config.properties"
            ]
        },
        {
            "Effect": "Allow",
            "Action": "cloudformation:TagResource",
            "Resource": "*",
            "Condition": {
                "ForAllValues:StringLike": {
                    "aws:TagKeys": "LaunchWizard*"
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "s3:CreateBucket",
                "s3:PutBucketVersioning",
                "s3:DeleteBucket",
                "lambda:CreateFunction",
                "lambda:DeleteFunction",
                "lambda:GetFunction",
                "lambda:GetFunctionConfiguration",
                "lambda:InvokeFunction"
            ],
            "Resource": [
                "arn:aws:lambda:*:*:function:*",
                "arn:aws:s3:::launchwizard*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "dynamodb:CreateTable",
                "dynamodb:DescribeTable",
                "dynamodb:DeleteTable"
            ],
            "Resource": "arn:aws:dynamodb:*:*:table/*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "secretsmanager:CreateSecret",
                "secretsmanager:DeleteSecret",
                "secretsmanager:TagResource",
                "secretsmanager:UntagResource",
                "secretsmanager:PutResourcePolicy",
                "secretsmanager:DeleteResourcePolicy",
                "secretsmanager:ListSecretVersionIds",
                "secretsmanager:GetSecretValue"
            ],
            "Resource": "arn:aws:secretsmanager:*:*:secret:*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "secretsmanager:GetRandomPassword",
                "secretsmanager:ListSecrets"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "ssm:CreateOpsMetadata"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": "ssm:DeleteOpsMetadata",
            "Resource": "arn:aws:ssm:*:*:opsmetadata/aws/ssm/LaunchWizard*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "sns:CreateTopic",
                "sns:DeleteTopic",
                "sns:Subscribe",
                "sns:Unsubscribe"
            ],
            "Resource": "arn:aws:sns:*:*:*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "fsx:UntagResource",
                "fsx:TagResource",
                "fsx:DeleteFileSystem",
                "fsx:ListTagsForResource"
            ],
            "Resource": "*",
            "Condition": {
                "StringLike": {
                    "aws:ResourceTag/Name": "LaunchWizard*"
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "fsx:CreateFileSystem"
            ],
            "Resource": "*",
            "Condition": {
                "StringLike": {
                    "aws:RequestTag/Name": [
                        "LaunchWizard*"
                    ]
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "fsx:DescribeFileSystems"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "s3:GetObject"
            ],
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "s3:ExistingObjectTag/servicecatalog:provisioning": "true"
                }
            }
        }
    ]
}
```

1. Sign in to the AWS Management Console and open the IAM console at https://console.aws.amazon.com/iam/.
2. Perform the following substeps for each of the three policies listed above.

   a. In the left navigation pane, choose **Policies** > **Create policy**.

   b. On the **Create policy** page, choose the **JSON** tab.

   c. Copy each policy above and paste it into the **Policy Document** JSON text field, replacing the placeholder text (perform these substeps individually for each of the three policies listed above).

   d. Choose **Next: Tags** > **Next: Review**.

   e. Enter a **Policy Name**.

   f. Choose **Create policy**.

3. Attach the three policies you just created to the IAM user you use to launch CloudFormation templates.

Find and launch your templates

1. Navigate to the Amazon S3 console.
2. Locate the name of the location within the Amazon S3 bucket that you specified when you defined the infrastructure for your Launch Wizard deployment.
3. Under the folder that you specified, locate and choose a new folder named **`<LaunchWizardDeploymentName>-<TimeStamp>`**. This is the folder to which the Launch Wizard service copies the CloudFormation templates and deployment artifacts.
4. After you choose the new folder, you will see an `sap/` folder and a JSON file named **`<LaunchWizardDeploymentName>-<DeploymentType>`**`-template.json`. This is the root CloudFormation template file. Select the check box next to this file and choose **Copy URL**.
5. Navigate to the CloudFormation console to create a stack with the URL that you copied.

For more information about CloudFormation templates, see Working with AWS CloudFormation templates in the *AWS CloudFormation User Guide*.

Deploy SAP applications with AWS Launch Wizard for SAP using a proxy server

AWS Launch Wizard for SAP launches and configures Amazon EC2 instances to deploy an SAP system on AWS. The launched instances must have outbound connectivity to internet to download operating system patches and communicate with several AWS services. You can setup this connection via an internet gateway or a proxy server in a public subnet.

The following is an example on how to configure a Squid proxy server for deploying SAP applications on AWS with Launch Wizard.

Topics

Setup

Configure your Squid proxy server with the following steps.

1. Choose any Linux-based AMI. In this example, we have selected SLES 12 SP5 for SAP AMI.
2. Verify that your server is hosted on a public subnet and is attached to a public IP address.
3. Add AWS services to the `allowed_list` file.

    a. In the Squid server configuration file `/etc/squid/squid.conf`, create an `allowed_list` path using the `acl` command.

    ```
    acl whitelist dstdomain '/etc/squid/allowed_list'
    ```

    b. In the `allowed_list` file, add the domains of all the services listed in the following table.

    c. Run the `rcsquid restart` command for the changes to take effect.

Service name	Domains to be allowed
Amazon DynamoDB	`.dynamodb.<<region>>.amazonaws.com` , `.dynamodb-fips.<<region>>.amazonaws.com`
Amazon EFS	`.elasticfilesystem.<<region>>.amazonaws.com` , `.elasticfilesystem-fips.<<region>>.amazonaws.com`
Amazon EBS	`.com.amazonaws.<<region>>.ebs`
Amazon EC2	`.api.ec2.<<region>>.aws` , `.ec2.<<region>>.amazonaws.com` , `.ec2-fips.<<region>>.amazonaws.com` , `.ec2messages.<<region>>.amazonaws.com` , `.169.254.169.254`
Amazon FSx	`.fsx.<region>.amazonaws.com`

Service name	Domains to be allowed
AWS Lambda	.com.amazonaws.<<region>>.lambda , .lambda.<<region>>.amazonaws.com , .lambda-fips.<<region>>.amazonaws.com , .lambda.<<region>>.api.aws
Amazon Route 53	.route53.amazonaws.com
Amazon CloudWatch	.com.amazonaws.<<region>>.evidently , .com.amazonaws.<<region>>.evidently-dataplane , .com.amazonaws.<<region>>.monitoring , .com.amazonaws.<<region>>.rum , .com.amazonaws.<<region>>.rum-dataplane , .com.amazonaws.<<region>>.synthetics , .com.amazonaws.<<region>>.events.monitoring.<<region>>.amazonaws.com , .logs.<<region>>.amazonaws.com , .monitoring-fips.<<region>>.amazonaws.com
CloudFormation	.cloudformation.<<region>>.amazonaws.com , .cloudformation-fips.<<region>>.amazonaws.com , .com.amazonaws.<<region>>.cloudformation
AWS KMS	.com.amazonaws.<<region>>.kms , .kms.<<region>>.amazonaws.com , .kms-fips.<<region>>.amazonaws.com

Service name	Domains to be allowed
AWS Secrets Manager	`.secretsmanager.<<region>>.amazonaws.com` , `.com.amazonaws.<<region>>.secretsmanager`
AWS Identity and Access Management	`.iam.amazonaws.com` , `.iam-fips.amazonaws.com`
AWS Systems Manager	`.ssm.<<region>>.amazonaws.com` , `.ssmmessages.<<region>>.amazonaws.com` , `amazon-ssm-us-east-1.s3.us-east-1.amazonaws.com`
Amazon S3	`.s3.amazonaws.com` , `<<S3_bucket_for_HANA_backint_backups>>.s3.<<region>>.amazonaws.com` , `.s3.<<region>>.amazonaws.com` , `.s3.dualstack.us-east-1.amazonaws.com`
AWS CLI	`awscli.amazonaws.com` .
SUSE infrastructure for SLES	`.smt-ec2.susecloud.net` , `.54.225.105.144` , `.54.197.240.216` , `.107.22.231.220` , `.34.197.223.242`
SUSE packages	`.scc.suse.com`
REDHAT repository	`.rhui.<<region>>.aws.ce.redhat.com`
Python packages	`.files.pythonhosted.org` , `.pypi.org` , `.python.org`
Amazon Cognito	`.cognito-identity.us-east-1.amazonaws.com`
Amazon Security Token Service	`.sts.amazonaws.com`

Run Launch Wizard

After you complete the initial setup, you can begin deploying your SAP application using Launch Wizard. For more information, see Deploy an SAP application with AWS Launch Wizard.

To connect your SAP deployment on Launch Wizard with the Squid proxy server, enter the IP address of the server. To add the server address, go to Step 2 Define infrastructure > Infrastructure - SAP landscape > Security groups > **Proxy server address - optional.**

The *No proxy* setting contains the list of whitelisted domains and IP addresses that do not pass through the proxy server.

In the *No proxy setting - optional* field, you must include the following IP addresses:

- Localhost - `127.0.0.1`
- Internal
- Amazon EC2 instance metadata- `169.254.169.254`

> **Note**
>
> Include the hostnames of ASCS, ERS, primary SAP HANA, and secondary SAP HANA instances in the *No proxy setting - optional* field, if you are deploying an SAP system with high availability using RHEL operating system. This will enable the cluster to communicate with all the nodes as well as perform any failover or failback operations.

Amazon EC2 connection

Your Amazon EC2 instance must be connected to the SUSE repository servers on AWS. Add the following IP addresses to the route tables of the associated Amazon EC2 instances. For more information, see Add and remove routes from a route table. The *Target* of these routes should be the NAT gateway of your subnet. For more information, see Add a NAT Gateway to an Existing VPC.

- `34.197.223.242/32`
- `54.197.240.216/32`
- `54.225.105.144/32`
- `107.22.231.220/32`

Troubleshoot

To resolve any connectivity issues with the Squid proxy server, use the following steps.

1. Login to your Squid proxy server.
2. Open the `access.log` file located at `/var/log/squid/access.log`.
3. Search for the **TCP_DENIED** message in the `access.log` file. The message displays an address that is not allowed in the proxy configuration.
4. Add the address to the `squid.conf` file and restart the Squid server for the changes to take effect.
5. You can now start over your SAP deployment with Launch Wizard.

> **Note**
>
> The troubleshooting steps are only applicable to the Squid proxy server. The location of the `log` file varies with the type of proxy server.

Security groups in AWS Launch Wizard for SAP

This section describes the security groups that Launch Wizard for SAP creates and assigns to the database and application instances. It also describes how the entries in the outbound and inbound communication rules for database and application security groups are updated.

Topics

- Security groups
- Connectivity to external systems and users

Security groups

A security group acts as a virtual firewall that controls the traffic for one or more instances. When you allow Launch Wizard to create security groups, it creates a set of security groups and assigns them to the SAP database and application instances to allow for inbound traffic. Security groups use the following naming conventions:

- `<Infrastructure_Configuration_Name>_App_SecurityGroup`

- `<Infrastructure_Configuration_Name>_DB_SecurityGroup`
- `WD_Security_Group`
- `WD_LB_Security_Group`

<Infrastructure_Configuration_Name>_App_SecurityGroup

`<Infrastructure_Configuration_Name>_App_SecurityGroup` is configured as follows to allow inbound access to the database servers.

Source	Protocol	Port Range
All instances attached to this security group	all	
All instances attached to the DB security group	TCP	1-65535

This configuration allows:

- inbound communication on all TCP ports from all of the SAP application servers deployed using the same configuration name
- inbound communication on all TCP ports from all of the database servers deployed using the same configuration name.

<Infrastructure_Configuration_Name >_DB_SecurityGroup

`<Infrastructure_Configuration_Name>_DB_SecurityGroup` is configured as follows to allow inbound access to the database servers.

Source	Protocol	Port Range
All instances attached to this security group	all	
All instances attached to the App security group	TCP	1-65535

Source	Protocol	Port Range
All instances attached to the App security group	UDP	111
All instances attached to the App security group	UDP	2049
All instances attached to the App security group	UDP	4000-4002

This configuration allows:

- inbound communication on all TCP ports from all of the SAP database servers deployed using the same configuration name.
- inbound communication on all TCP ports from all of the SAP application servers deployed using the same configuration name.
- inbound communication on UDP 111,2049 and 4000 to 4002 from all the SAP application servers deployed using the same configuration name.

WD_Security_Group

`WD_Security_Group` is configured as follows to allow inbound access to SAP Web Dispatcher servers.

Deployment type	Source	Protocol	Port range
All	*ID of the WD_Security_Group*	all	1-65535
All	Input	TCP	1-65535
Distributed instances deployment	*ID of the security group for the SAP transport directory*	TCP	2049

Deployment type	Source	Protocol	Port range
High availability (HA) deployment	*ID of the security group for the SAP transport directory in **Availability Zone 1***	TCP	2049
High availability (HA) deployment	*ID of the security group for the SAP transport directory in **Availability Zone 2***	TCP	2049

`WD_Security_Group` is configured as follows to allow the following outbound access from SAP Web Dispatcher servers.

Deployment type	Destination	Protocol	Port range
All	*ID of the security group for the SAP application server*	TCP	8000-8197

WD_LB_Security_Group

`WD_LB_Security_Group` is configured as follows to allow the following inbound access to the load balancer for SAP Web Dispatcher servers.

Deployment type	Source	Protocol	Port range
All	Input	TCP	1-65535

`WD_LB_Security_Group` is configured as follows to allow the following outbound access from the load balancer for SAP Web Dispatcher servers.

Deployment type	Destination	Protocol	Port range
All	*ID of the WD_Security_Group*	all	8000-8097
All	*ID of the WD_LB_Security_Group*	all	1-65535

Connectivity to external systems and users

CIDR/IP address and security group entries are entered in the infrastructure configuration. This allows access to SAP systems by front end users and upstream/downstream systems that are running in that CIDR block, or by end users (IP address) or systems assigned to those security groups. Port ranges are included in the rule definition that allow inbound access so that you can reuse the infrastructure configuration and deploy SAP systems with an instance number 00 to 99. Each entry in the outbound and inbound communication rules for a database security group, created either by the service or provided by the user, are updated as follows.

Source	Protocol	Port Range
Input	TCP	22
Input	TCP	1128 - 1129
Input	TCP	4300 - 4399
Input	TCP	8000 - 8099
Input	TCP	8443
Input	TCP	30013 - 39913
Input	TCP	30015 - 39915
Input	TCP	30017 - 39917
Input	TCP	30041 - 39941

Source	Protocol	Port Range
Input	TCP	30044 - 39944
Input	TCP	50013 - 59914

Each entry in the outbound and inbound communication rules for the application security group, created either by the service or by the user, are updated as follows.

Source	Protocol	Port Range
Input	TCP	22
Input	TCP	3200 - 3399
Input	TCP	8080
Input	TCP	8443
Input	TCP	3600-3699
Input	TCP	4237

Note

- When the deployment is complete, you can update the security group information by adjusting the port range and source information.
- Launch Wizard considers a security group that it created as a shared resource. It does not delete the security group if you delete a deployment or if a deployment is rolled back.

Troubleshoot AWS Launch Wizard for SAP

Each application in your account in the same AWS Region can be uniquely identified by the application name specified at the time of a deployment. The application name can be used to view the details related to the application launch.

Contents

Launch Wizard provisioning events

Launch Wizard captures events from SSM Automation and CloudFormation to track the status of an ongoing application deployment. If an application deployment fails, you can view the deployment events for this application by selecting **Deployments** from the navigation pane. A failed event shows a status of **Failed** along with a failure message.

CloudWatch Logs

Launch Wizard streams provisioning logs from all of the AWS log sources, such as CloudFormation, SSM, and CloudWatch Logs. You can access CloudWatch logs for your SAP deployment with the following steps.

1. Sign in to console.aws.amazon.com and go to AWS Launch Wizard.
2. Under **Deployments** on the left panel, go to **SAP** and you can see the list of your SAP deployments.
3. Select the failed deployment for which you want to verify the logs.
4. Choose **Actions** > **View/Manage resources** > **View CloudWatch application logs**.
5. You can now view the detailed logs and log streams that provide additional information on the SAP application type that failed during deployment.

CloudFormation stack

Launch Wizard uses CloudFormation to provision the infrastructure resources of an application. Launch Wizard launches various stacks in your account for validation and application resource creation. You can verify the stacks via AWS console or AWS CLI.

Console

1. Sign in to console.aws.amazon.com and go to AWS Launch Wizard.
2. Under **Deployments** on the left panel, go to **SAP** and you can see the list of your SAP deployments.
3. Select the failed deployment for which you want to verify the stacks.
4. Choose **Actions** > **View/Manage resources** > **View CloudFormation template** .
5. You can now view all the stacks and their current status. To see more details on any stack, select a **Stack name**.
6. You are now on the **Stack details** page of your selected stack. Choose **Events** from the top menu bar to view the cause of the failure.

CLI

CloudFormation stacks can be found in your account using the CloudFormation describe-stacks API. The following are the relevant filters for the describe-stacks API.

- **Application resources**

  `LaunchWizard-APPLICATION_NAME`.

You can view the status of these CloudFormation stacks. If any of them fail, you can view the cause of the failure.

Pre- and post-deployment configuration scripts

Can't find the output of my scripts

- **Cause:** Customizations are key scripts that you want to run on the EC2 instances and the logs from script deployments are not included with the provisioning logs.

- **Solution:** The logs for scripts that run on EC2 instances are included in the CloudWatch log group that Launch Wizard creates in your account for the workload. The CloudWatch log group can be identified as `LaunchWizard-APPLICATION_NAME` . You can find the following logs in this log group.
  - `lw-customization/<instance-id>/preDeploymentConfiguration` — For pre-deployment configuration scripts that run on the specified EC2 instance.
  - `lw-customization/<instance-id>/postDeploymentConfiguration` — For post-deployment configuration scripts that run on the specified EC2 instance.

Application launch quotas

Launch Wizard allows for a maximum of 25 active applications for any given application type. Up to three applications can be `in progress` at a time. If you want to increase this limit, contact Support.

Instance level logs

To check the progress of a deployment, you can log in to an instance as soon its instance state is listed as **running**. When the deployment is finished, the log files are moved to `/tmp`.

By default, your provisioned Amazon EC2 instances are retained when a deployment fails. If you created your Launch Wizard deployment with these default settings, you can navigate to the following paths for further evaluation.

Directory	Purpose
`/root/install`	The working directory of Launch Wizard SAP deployment.
`/root/install/scripts`	The home directory of Launch Wizard SAP deployment. It contains all the scripts called by Launch Wizard.
`/root/install/scripts/log`	All the logs related to the deployment (`install.log` file).
`/tmp/`	Based on the SAP components that are deployed on an Amazon EC2 instance, Launch

Directory	Purpose
	Wizard creates a folder in this directory for SAP software application deployment logs.
`/var/log/messages`	The unhandled exceptions of an Amazon EC2 instance.
`/var/log/zypper.log`	All the logs for SLES operating system package installation failures.
`/var/log/yum.log`	All the logs for RHEL operating system package installation failures.
`/var/log/pacemaker`	All the logs for pacemaker cluster.
`/var/log/pacemaker/pacemaker.log`	
`/var/log/cluster/corosync.log`	

SAP application software deployment logs

Depending on which SAP components are deployed on an instance, Launch Wizard creates a folder in `/tmp` to log all of the SAP software application deployment logs. If a database component is deployed on an instance, the folder name in the file will be `NW_ABAP_DB`. If an application server is deployed, the folder name will be `NW_ABAP_APP`. For single node deployments, there will be multiple folders, such as `NW_ABAP_DB` and `NW_ABAP_CI`, which represent the different components deployed on the instance.

Errors

Your requested instance type is not supported in your requested Availability Zone

- **Cause:** This failure might occur during the launch of your instance, or during the validation of the instances that Launch Wizard launches in your selected subnets.
- **Solution:** For this scenario, you must choose a different Availability Zone and retry the deployment from the initial page of the Launch Wizard console.

Infrastructure template already exists

- **Cause:** This failure occurs when you choose to create a new infrastructure configuration and then navigate back to the first step in the wizard to review or adjust any settings. Launch Wizard has already registered the configuration template, so choosing **Next** results in the error "Template name already exists. Select a new template name."
- **Solution:**

  Perform one of the following actions to continue with your deployment.

  - Change the name of the configuration template and continue.
  - Choose another template and continue.
  - Delete the template causing the error by navigating to the **Saved Infrastructure Setting** tab under **Deployments – SAP**, and then continue with your configuration using the same configuration name.

AWS Systems Manager for SAP

An Internal Error Occurred

- **Cause:** For users using AWS Systems Manager for the first time, the CloudFormation resource (`AWS::SystemsManagerSAP::Application`) can fail with a message `An Internal Error Occurred` due to issues during the SLR (service-linked role) `AWSSSMForSAPServiceLinkedRolePolicy` creation.
- **Solution:**
  1. Use the IAM console to ensure that `AWSSSMForSAPServiceLinkedRolePolicy` is in your account.
  2. Retry the Launch Wizard deployment to complete the registration successfully.
  3. If errors persist, contact Support

For more information, see Troubleshooting AWS Systems Manager for SAP.

Support

If your deployment is failing after following the troubleshooting steps listed here, we recommend you to create a support case with the following information.

```
           [Error description]:<Provide a brief description of the error.>

           [Deployment information]: Provide information about the failed deployment.
           Account number: <AWS account number>
           Deployment name: <Enter deployment name>
           Deployment type: <Single-instance/Multi-instance/High availability>
           SAP HANA version: <Enter SAP HANA database version>
           SAP application: <Enter SAP application name>
           OS type: <Enter operating system>
           OS version: <Enter operating system version>
           Amazon EC2 instance family: <Enter Amazon EC2 instance family>
           Amazon EC2 instance type: <Enter Amazon EC2 instance type>
           If used proxy: <Yes/No>
           AMI type: <BYOI/BYOS/Marketplace>
           Instances retained: <Yes/No>
           FailedStackID (optional):

           [Required logs] Provide the following logs. Based on the scenario and state
of deployment, some logs may not be available.
           /root/install/scripts/log/
           /tmp/install.log
           /tmp/inputs.json
           /var/log/cloud-init.log
           /var/log/hdblcm.log (If SAP HANA install is selected)
           /tmp/NW directory (If SAP HANA install is selected)

           If you haven't retained your Amazon EC2 instance, provide the logs
extracted from CloudWatch logs.

           [Troubleshooting]
           Provide the details of the troubleshooting steps that you carried out and
the results from them.
```

For more information, see Creating a support case.

AWS Launch Wizard for SQL Server

AWS Launch Wizard is a service that guides you through the sizing, configuration, and deployment of Microsoft SQL Server applications on AWS, following the AWS Well-Architected Framework. AWS Launch Wizard supports both single instance and high availability (HA) application deployments.

AWS Launch Wizard reduces the time it takes to deploy SQL Server solutions to the cloud. You input your application requirements, including performance, number of nodes, and connectivity, on the service console. AWS Launch Wizard identifies the right AWS resources to deploy and run your SQL Server application. AWS Launch Wizard provides an estimated cost of deployment, and you can modify your resources and instantly view the updated cost assessment. When you approve, AWS Launch Wizard provisions and configures the selected resources in a few hours to create a fully-functioning production-ready SQL Server application. It also creates custom AWS CloudFormation templates, which can be reused and customized for subsequent deployments.

Once deployed, your SQL Server application is ready to use and can be accessed on the EC2 console. You can manage your SQL Server application with AWS SSM.

Supported operating systems and SQL versions

AWS Launch Wizard supports the following operating systems and SQL Server versions:

Deployments on Windows

- Windows Server 2025/2022/2019
- Enterprise and Standard Editions of Microsoft SQL Server 2025/2022/2019
- Developer Edition of Microsoft SQL Server 2025/2022

Amazon FSx for Failover Clustering (FCI) deployments on Windows

- Windows Server 2025/2022/2019
- Enterprise and Standard Editions of Microsoft SQL Server 2025/2022/2019

  CUs are installed at the same time as public AMIs for SQL license-included AMIs. CUs and service packs are not installed for license-included Windows AMIs and BYOL AMIs.

Deployments on Ubuntu

- Ubuntu 18.04
- Enterprise and Standard Edition of Microsoft SQL Server 2019

Features of AWS Launch Wizard

AWS Launch Wizard provides the following features:

- Simple application deployment
- AWS resource selection
- Cost estimation
- Reusable code templates
- SNS notification
- Always On Availability Groups (SQL Server)
- Dedicated Hosts (deployment on Windows)
- Early input validation
- Application resource groups for easy discoverability
- One-click monitoring
- Amazon FSx for Failover Clustering (FCI)

Simple application deployment

AWS Launch Wizard makes it easy for you to deploy third-party applications on AWS, such as Microsoft SQL Server. When you input the application requirements, AWS Launch Wizard deploys the necessary AWS resources for a production-ready application. This means that you do not have to manage separate infrastructure pieces or spend time provisioning and configuring your SQL Server application.

AWS resource selection

Launch Wizard considers performance, memory, bandwidth, and other application features to determine the best instance type, EBS volumes, and other resources for your SQL Server application. You can modify the recommended defaults.

Cost estimation

Launch Wizard provides a cost estimate for a complete deployment. The cost estimate is itemized for each individual resource to deploy. The estimated cost automatically updates each time you change a resource type configuration in the wizard. The provided estimates are for general comparisons only. The estimates are based on On-Demand costs and actual costs may be lower.

Reusable code templates

Launch Wizard creates a CloudFormation stack that can be reused to customize and replicate your infrastructure in multiple environments. Code in the template helps you provision resources. You can access and use the templates created by your Launch Wizard deployment from the CloudFormation console. For more information about CloudFormation stacks, see Working with stacks.

SNS notification

You can provide an SNS topic so that Launch Wizard will send you notifications and alerts about the status of a deployment.

Always On Availability Groups (SQL Server)

Always On Availability Groups (AG) is a Microsoft SQL Server feature that is supported by the AWS SQL Server installation. AG augments the availability of a set of user databases. An availability group supports a failover environment for a discrete set of user databases, known as availability databases. If one of these databases fails, another database takes over its workload with no impact on availability. Always On Availability improves database availability, enabling more efficient resource usage. For more information about the concepts and benefits of Always On Availability, see Always On Availability Groups (SQL Server).

Dedicated Hosts (deployment on Windows)

You can deploy SQL Server Always On Availability Groups (AG) or basic availability groups on your Dedicated Hosts to leverage your existing SQL Server Licenses (BYOL). From the Launch Wizard console, select **Dedicated Host** tenancy, and then select the Dedicated Hosts for your VPC. For more information about Amazon EC2 Dedicated Hosts, see Dedicated Hosts.

Early input validation

You can leverage your existing infrastructure (such as VPC or Active Directory) with Launch Wizard. This may lead to deployment failures if your existing infrastructure does not meet certain deployment prerequisites. For example, for a SQL Server Always On deployment in your existing VPC, the VPC must have at least one public subnet and two private subnets. It must also have outbound connectivity to Amazon S3, Systems Manager, and AWS CloudFormation service endpoints. If these requirements are not met, the deployment will fail. If you are in a later stage of a deployment, this failure can take more than an hour to detect. To detect these types of issues early in the application deployment process, Launch Wizard's validation framework verifies key application and infrastructure specifications before provisioning. Verification takes approximately 15 minutes. If necessary, you can take appropriate actions to adjust your VPC configuration.

Launch Wizard performs the following infrastructure validations:

Resource limit validations at the AWS account level:

- VPC
- Internet gateway
- Number of CloudFormation stacks

Additionally, Launch Wizard performs the following application-specific validations:

- Active Directory credentials (deployment on Windows)
- Public subnet outbound connectivity
- Private subnet outbound connectivity
- Custom Windows AMIs:
  - SQL Server installed and running on instance
  - Compliant versions of Windows and SQL Server
- Dedicated Hosts (deployment on Windows)
  - AMIs are filtered according to the billing code. This filtering behavior is the result of restrictions described in the Dedicated Host restrictions page.
  - Supported instance type
  - Sufficient capacity to launch number of nodes and instances

- Selected subnet and corresponding Dedicated Host are in the same Availability Zone for any additional nodes beyond the primary and first secondary nodes

> **Note**
>
> Some validations, for example for valid Active Directory credentials, require Application Wizard to launch a `t2.large` EC2 instance in your account for a few minutes. After it runs the necessary validations, Launch Wizard terminates the instance.

Application resource groups for easy discoverability

Launch Wizard creates a resource group for all of the AWS resources created for your SQL Server application. You can manage the resources through the EC2 console or with Systems Manager. When you access Systems Manager through Launch Wizard, the resources are automatically filtered for you based on your resource group. You can manage, patch, and maintain your SQL Server applications in Systems Manager.

One-click monitoring

Launch Wizard integrates with CloudWatch Application Insights to provide a one-click monitoring setup experience for deploying SQL Server HA workloads on AWS. When you select the option to set up monitoring and insights with Application Insights on the Launch Wizard console, Application Insights automatically sets up relevant metrics, logs, and alarms on CloudWatch, and starts monitoring newly deployed workloads. You can view automated insights and detected problems, along with the health of your SQL Server HA workloads, on the CloudWatch console.

Counters that you can configure using Application Insights include:

- Mirrored Write Transaction/sec
- Recovery Queue Length
- Transaction delay
- Windows Event Logs on CloudWatch

You can also get automated insights when a failover event or problem, such as a restricted access to query a target database, is detected on your workload.

Amazon FSx for Failover Clustering (FCI)

Launch Wizard uses Amazon FSx to provide Failover Clustering for SQL Server deployments. Failover Clustering is a high availability solution for SQL that puts all database and log files in shared storage (Amazon FSx). The Amazon FSx file share spans multiple Availability Zones and is highly redundant, which allows for automatic failover between SQL nodes in the event of failure.

Launch Wizard offers two storage options for your FCI deployments: Amazon FSx for Windows or Amazon FSx for NetApp ONTAP. If you choose NetApp ONTAP as the storage type for FCI, License Manager creates the user name `FSXAdmin` and a password during the deployment. The user name and password are stored in AWS Secrets Manager to manage ONTAP.

Related services

The following services are used when you deploy a SQL Server application with AWS Launch Wizard:

- CloudFormation
- Amazon Simple Notification Service (SNS)
- Amazon CloudWatch Application Insights
- Linux-only technologies

CloudFormation

CloudFormation is a service for modeling and setting up your AWS resources, enabling you to spend more time focusing on your applications that run in AWS. You create a template that describes all of the AWS resources that you want to use (for example, Amazon EC2 instances or Amazon RDS DB instances), and CloudFormation takes care of provisioning and configuring those resources for you. With Launch Wizard, you don’t have to sift through CloudFormation templates to deploy your application. Instead, Launch Wizard combines infrastructure provisioning and configuration (with a CloudFormation template) and application configuration (with code that runs on EC2 instances to configure the application) into a unified SSM Automation document. The SSM document is then invoked by Launch Wizard’s backend service to provision a SQL Server application in your account. For more information, see the *AWS CloudFormation User Guide*.

Amazon Simple Notification Service (SNS)

Amazon Simple Notification Service (SNS) is a highly available, durable, secure, fully managed pub/sub messaging service that provides topics for high-throughput, push-based, many-to-many messaging. Using Amazon SNS topics, your publisher systems can fan out messages to a large number of subscriber endpoints and send notifications to end users using mobile push, SMS, and email. You can use SNS topics for your Launch Wizard deployments to stay up-to-date on deployment progress. For more information, see the *Amazon Simple Notification Service Developer Guide*.

Amazon CloudWatch Application Insights

Amazon CloudWatch Application Insights facilitates observability for .NET and SQL Server applications. It can help you set up the best monitors for your application resources to continuously analyze data for signs of problems with your applications. Application Insights, which is powered by Sagemaker and other AWS technologies, provides automated dashboards that show potential problems with monitored applications, helping you to quickly isolate ongoing issues with your applications and infrastructure. The enhanced visibility into the health of your applications that Application Insights provides can help you reduce your mean time to repair (MTTR) so that you don't have to pull in multiple teams and experts to troubleshoot your application issues.

Linux-only technologies

The following key technologies are used when you deploy a SQL Server application with Amazon Launch Wizard to the Linux platform.

- **Pacemaker** is an open source cluster resource manager (CRM), which is a system that coordinates managed resources and services made highly available by a cluster.
- **Corosync** is an open source program that provides cluster membership and messaging capabilities, often referred to as the messaging layer, to client servers. In contrast to Pacemaker, which allows you to control cluster behavior, Corosync makes it possible for servers to communicate as a cluster.
- **Transact-SQL** is an extension to the SQL language. It is used to interact with relational databases. Transact-SQL is platform-agnostic and can be used to configure the AlwaysOn Availability Group and listener.
- **Fencing** is used to isolate a malfunctioning server from the cluster in order to protect and secure the synced resources. The recommended solution to use in the case of a malfucntioning server

is the "Shoot the other node in the head" (STONITH) method. STONITH is a fencing technique that isolates a failed node so that it does not disrupt a computer cluster. The STONITH method fences failed nodes by resetting or powering down the failed node. Fencing is also used when a clustered service cannot be stopped. In this case, the cluster uses fencing to force the whole node offline, which makes it safe to start the service from a different server. Fencing can be performed at two levels: the node or resource level. Launch Wizard only supports node-level fencing.

Default quotas

Launch Wizard allows for a maximum of 50 active applications (with status `in progress` or `completed`) for any given application type. If you want to increase this limit, contact Support. Launch Wizard supports three parallel, in-progress deployments per account.

AWS Regions

Launch Wizard uses various AWS services during the provisioning of the application's environment. Not every workload is supported in all AWS Regions. For a current list of Regions where the workload can be provisioned, see AWS Launch Wizard workload availability.

Components

Topics

- Windows
- Linux

Windows

A SQL Server application deployed on Windows with Launch Wizard includes the following components:

- A **virtual private cloud (VPC)** configured with public and private subnets across two Availability Zones. A public subnet is a subnet whose traffic is routed to an internet gateway. If a subnet does not have a route to the internet gateway, then it is a private subnet. The VPC provides the network infrastructure for your SQL Server deployment. You can choose an optional third Availability Zone for additional SQL cluster nodes, as shown below.

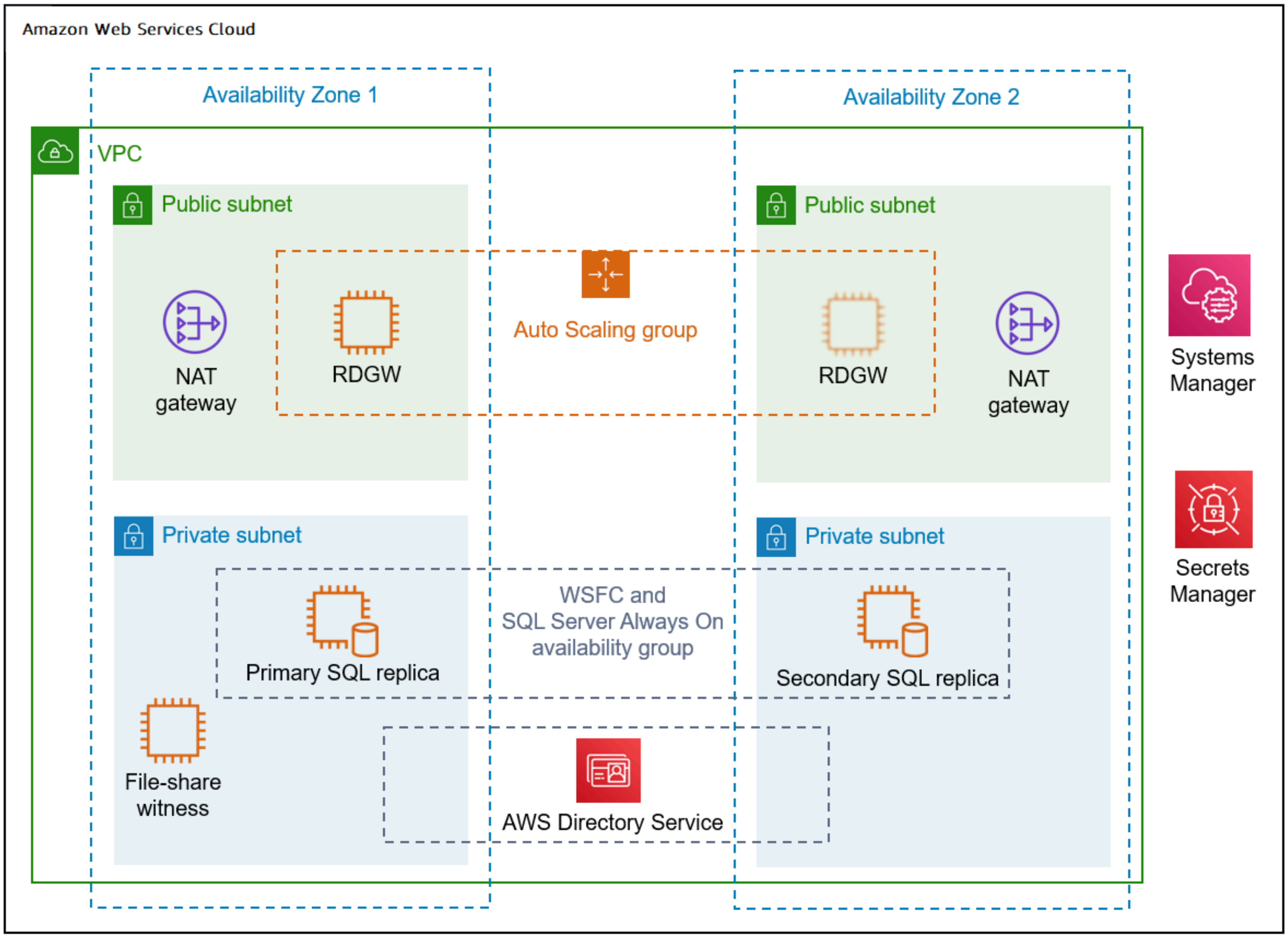
- An **internet gateway** to provide access to the internet.
- In the public subnets, **Windows Server-based Remote Desktop Gateway (RDGW) instances and network address translation (NAT) gateways** for outbound internet access. If you are deploying in your preexisting VPC, Launch Wizard uses the existing NAT gateway in your VPC. For more information about NAT gateways, see NAT Gateways.
- **Elastic IP addresses** associated with the NAT gateway and RDGW instances. For more information about Elastic IP addresses, see Elastic IP addresses.
- In the private subnets, **Active Directory domain controllers**.
- In the private subnets, **Windows Server-based instances as Windows Server Failover Clustering (WSFC) nodes**. For more information, see Windows Server Failover Clustering with SQL Server.
- **SQL Server Enterprise edition with SQL Server Always On Availability Groups on each WSFC node**. This architecture provides redundant databases and a witness server to ensure that a quorum can vote for the node to be promoted to the controlling resource. The default architecture mirrors an on-premises architecture of two SQL Server instances spanning two subnets placed in two different Availability Zones. For more information about SQL Server Always On Availability Groups, see Overview of Always On Availability Groups (SQL Server).
- **Security groups** to ensure the secure flow of traffic between the instances deployed in the VPC. For more information, see Security Groups for Your VPC.

> **Note**
>
> If you choose to deploy SQL Server Always On through Launch Wizard into your existing VPC, there is an additional mandatory check box on the console to indicate whether VPC and public/private subnet requirements have been met.

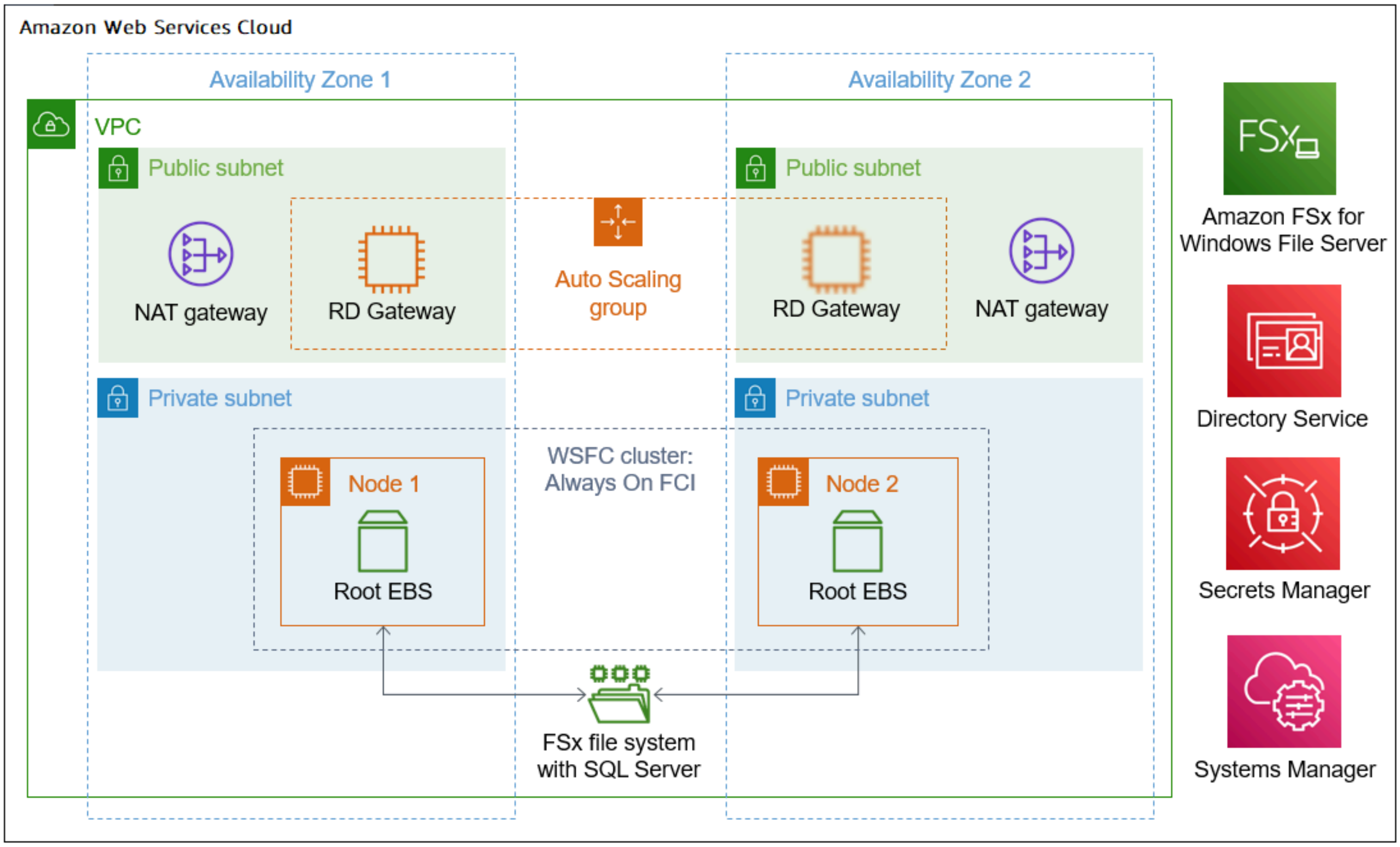
- **Amazon FSx** to provide highly available and redundant storage across Availability Zones for clustering.

> **Note**
>
> Launch Wizard uses two Availability Zones.

You can build a SQL HA installation, as shown in the following diagram.



You can also choose to build an architecture with SQL Server Always On FCI, as shown in the following diagram.



Linux

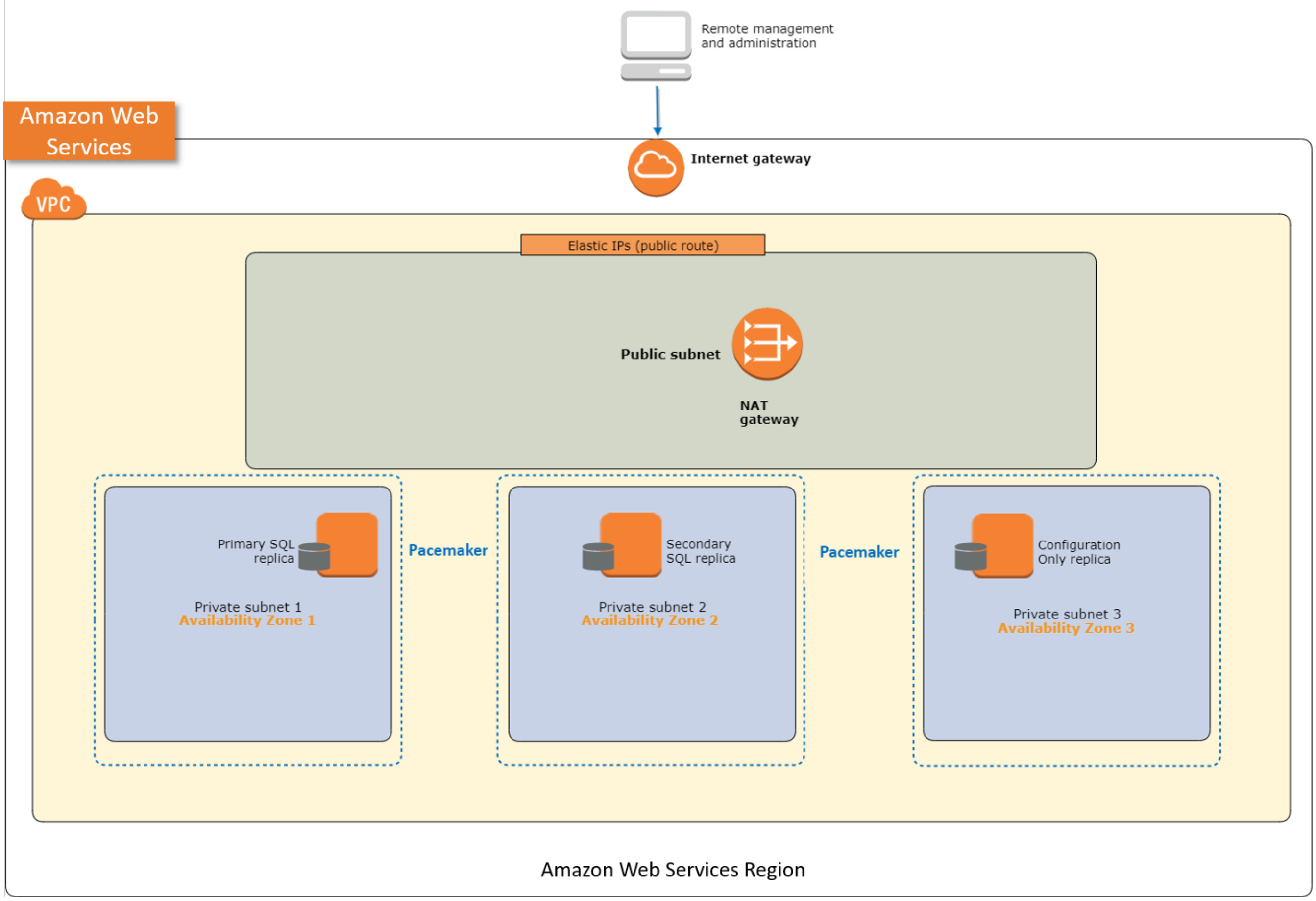
A SQL Server application deployed on Linux with Launch Wizard includes the following components:

- A **virtual private cloud (VPC)** configured with public and private subnets across three Availability Zones. A public subnet is a subnet whose traffic is routed to an internet gateway. If a subnet does not have a route to the internet gateway, then it is a private subnet. The VPC provides the network infrastructure for your SQL Server deployment.
- An **internet gateway** to provide access to the internet.
- In the public subnets, **network address translation (NAT)** for outbound internet access. If you are deploying in your preexisting VPC, Launch Wizard uses the existing NAT gateway in your VPC. For more information about NAT gateways, see NAT Gateways.
- Two of the private subnets each run a SQL Server **replica node**. One acts as the primary node, and the other as secondary node. The third private subnet is used to run the configuration replica. Launch Wizard deployments on Linux use Pacemaker as the cluster resource manager. Pacemaker differs from Windows Server Failover Cluster (WSFC), which is used for Windows

deployments, in terms of how it handles quorum. For Always On availability groups (AG) on Linux, arbitration happens in SQL Server where the metadata is stored. This is where the configuration-only replica is relevant. In order to maintain quorum and enable automatic failovers, Launch Wizard sets up a third node that acts as the configuration-only replica.

- **Security groups** to ensure the secure flow of traffic between the instances deployed in the VPC. For more information, see Security Groups for Your VPC.

The high-level architecture of a SQL Server high availability solution on Linux is similar to the architecture for deployment on Windows. The main differences are the low-level components and technologies. The architecture for Linux deployments provides redundant databases and a configuration-only replica node to verify that a quorum can vote for the node to be promoted to the controlling resource. The default architecture mirrors an on-premises architecture of two SQL Server instances spanning two subnets placed in two different Availability Zones. For more information about SQL Server Always On Availability Groups (AG), see Overview of Always On Availability Groups (SQL Server) in the Microsoft documentation.



Get started with AWS Launch Wizard for SQL Server

This section contains information to help you set up your environment to deploy SQL Server with Launch Wizard, including:

- Active Directory permissions
- How to create an IAM policy and assign the permissions
- OS and SQL version requirements
- Configuration settings

When your environment is set up, you can deploy a SQL Server Always On application with Launch Wizard by following the steps and parameter specification details provided in this section.

Topics

AWS Identity and Access Management (IAM)

The following steps to establish the AWS Identity and Access Management (IAM) role and set up the user for permissions are typically performed by an IAM administrator for your organization.

Topics

Sign up for an AWS account

Sign up for an AWS account

If you do not have an AWS account, complete the following steps to create one.

To sign up for an AWS account

1. Open https://portal.aws.amazon.com/billing/signup.
2. Follow the online instructions.

   Part of the sign-up procedure involves receiving a phone call or text message and entering a verification code on the phone keypad.

   When you sign up for an AWS account, an *AWS account root user* is created. The root user has access to all AWS services and resources in the account. As a security best practice, assign administrative access to a user, and use only the root user to perform tasks that require root user access.

AWS sends you a confirmation email after the sign-up process is complete. At any time, you can view your current account activity and manage your account by going to https://aws.amazon.com/ and choosing **My Account**.

Create a user with administrative access

After you sign up for an AWS account, secure your AWS account root user, enable AWS IAM Identity Center, and create an administrative user so that you don't use the root user for everyday tasks.

Secure your AWS account root user

1. Sign in to the AWS Management Console as the account owner by choosing **Root user** and entering your AWS account email address. On the next page, enter your password.

   For help signing in by using root user, see Signing in as the root user in the *AWS Sign-In User Guide*.
2. Turn on multi-factor authentication (MFA) for your root user.

   For instructions, see Enable a virtual MFA device for your AWS account root user (console) in the *IAM User Guide*.

Create a user with administrative access

1. Enable IAM Identity Center.

   For instructions, see Enabling AWS IAM Identity Center in the *AWS IAM Identity Center User Guide*.
2. In IAM Identity Center, grant administrative access to a user.

   For a tutorial about using the IAM Identity Center directory as your identity source, see Configure user access with the default IAM Identity Center directory in the *AWS IAM Identity Center User Guide*.

Sign in as the user with administrative access

- To sign in with your IAM Identity Center user, use the sign-in URL that was sent to your email address when you created the IAM Identity Center user.

  For help signing in using an IAM Identity Center user, see Signing in to the AWS access portal in the *AWS Sign-In User Guide*.

Assign access to additional users

1. In IAM Identity Center, create a permission set that follows the best practice of applying least-privilege permissions.

   For instructions, see Create a permission set in the *AWS IAM Identity Center User Guide*.
2. Assign users to a group, and then assign single sign-on access to the group.

   For instructions, see Add groups in the *AWS IAM Identity Center User Guide*.

Assign permissions to use Launch Wizard

To deploy a SQL Server Always On application with Launch Wizard, your user must have the permissions provided by the `AmazonLaunchWizardFullAccessV2` policy. The following guidance is provided for IAM administrators to provide permissions for users to access and deploy applications from Launch Wizard using the `AmazonLaunchWizardFullAccessV2` policy.

To provide access, add permissions to your users, groups, or roles:

- Users and groups in AWS IAM Identity Center:

  Create a permission set. Follow the instructions in Create a permission set in the *AWS IAM Identity Center User Guide*.
- Users managed in IAM through an identity provider:

  Create a role for identity federation. Follow the instructions in Create a role for a third-party identity provider (federation) in the *IAM User Guide*.
- IAM users:
  - Create a role that your user can assume. Follow the instructions in Create a role for an IAM user in the *IAM User Guide*.
  - (Not recommended) Attach a policy directly to a user or add a user to a user group. Follow the instructions in Adding permissions to a user (console) in the *IAM User Guide*.

⚠ **Important**

Log in with the user associated with the above policy when you use Launch Wizard.

One-time creation of IAM Role

On the **Choose Application** page of Launch Wizard, under **Permissions**, Launch Wizard displays the IAM role required for the Amazon EC2 instances created by Launch Wizard to access other AWS services on your behalf. When you select **Next**, Launch Wizard attempts to discover the IAM role in your account. If the role exists, it is attached to the instance profile for the EC2 instances that Launch Wizard will launch into your account. If the role does not exist, Launch Wizard attempts to create the role with the same name, `AmazonEC2RoleForLaunchWizard`. This role is comprised of two IAM managed policies: `AmazonSSMManagedInstanceCore` and `AmazonEC2RolePolicyForLaunchWizard`. After the role is created, the IAM administrator can delegate the application deployment process to another user who, in turn, must have the Launch Wizard IAM managed policy described in the following section.

AWS Secrets Manager permissions

Launch Wizard uses AWS Secrets Manager to manage your domain and SQL Server account passwords. Your username and password is stored in Secrets Manager and is retrieved during the build process. The following resource policy is added to the secret so that the

AmazonEC2RoleForLaunchWizard IAM role used by Launch Wizard can retrieve the secret. For more information about Secrets Manager, see the AWS Secrets Manager User Guide.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": {
            "AWS":
                "arn:aws:iam::111122223333:role/service-role/
AmazonEC2RoleForLaunchWizard"
            },
            "Action": [
                "secretsmanager:GetSecretValue",
                "secretsmanager:CreateSecret",
                "secretsmanager:GetRandomPassword"
            ],
            "Resource": "*"
        }
    ]
}
```

Active Directory (Windows deployment)

Launch Wizard can deploy SQL Server using AWS Directory Service for Microsoft Active Directory (AWS Managed Microsoft AD), or your Self Managed Active Directory.

AWS Managed Active Directory

If you are deploying SQL Server into an existing VPC with an existing Active Directory, Launch Wizard uses your Managed Active Directory (AD) domain user credentials to set up a fully functional SQL Server Always On Availability Group in the Active Directory. Launch Wizard supports this deployment option only for AWS Managed Active Directory. Your Managed Active Directory does not have to be in the same VPC as the one in which SQL Server Always On is deployed. If it is in a different VPC than the one in which SQL Server Always On is deployed, verify that you set up connectivity between the two VPCs. The domain user requires the following permissions in the Active Directory Default organizational unit (OU) to enable Launch Wizard to perform the deployment successfully:

- `Reset password`
- `Write userAccountControl`
- `Create user accounts`
- `Create computer objects`
- `Read all properties`
- `Modify permissions`

The following key operations are performed against your Active Directory by Launch Wizard. These operations result in the creation of new records or entries in Active Directory.

- SQL Server service user added as a new Active Directory user if it does not already exist in Active Directory.
- SQL Server instance and Remote Desktop Gateway Access instance joined to the Active Directory domain.
- `CreateChild` role added to Windows Server Failover Cluster as part of `ActiveDirectoryAccessRule`.
- `FullControl` role added to SQL Server Service user as part of `FileSystemRights`.

Self Managed Active Directory

If you are deploying SQL Server into an existing VPC across multiple Availability Zones and connecting to a Self Managed Active Directory or deploying SQL Server into an existing VPC on a single node and connecting to a Self Managed Active Directory, verify the following prerequisites.

- If your Self Managed Active Directory resides in another network than where you are deploying SQL Server, make sure you have connectivity between your VPC and the Self Managed Active Directory network. You must also be able to connect to any DNS servers you specify during deployment from your VPC. For more information, see Network-to-Amazon VPC connectivity options.
- Your SQL Server resources must be able to perform DNS resolution from within the VPC to any DNS servers you specify. For options on how to set this up, see How to Set Up DNS Resolution Between On-Premises Networks and AWS Using Directory Service and Amazon Route 53 or How

to Set Up DNS Resolution Between On-Premises Networks and AWS Using Directory Service and Microsoft Active Directory.

- The domain functional level of your Active Directory domain controller must be Windows Server 2019 or later.
- The firewall on the Active Directory domain controllers should allow the connections from the Amazon VPC from which you will create the Launch Wizard deployment. At a minimum, your configuration should include the ports mentioned in How to configure a firewall for Active Directory domains and trusts.
- The domain user requires the following permissions in the Active Directory Default organizational unit (OU) to enable Launch Wizard to perform the deployment successfully:
  - `Reset password`
  - `Write userAccountControl`
  - `Create user accounts`
  - `Create computer objects`
  - `Read all properties`
  - `Modify permissions`

Requirements for Windows and Linux AMIs

Launch Wizard has requirements for using custom Windows and Linux AMIs as well as Windows license-included AMIs in certain deployment scenarios.

Requirements for using Windows license-included AMIs (deployment on Windows)

When you use Windows license-included AMIs, note the following:

- You can use Windows license-included AMIs with SQL Bring-Your-Own-License (BYOL).
- Your SQL media must meet certain requirements to use Windows license-included AMIs with SQL BYOL. The SQL media must be:
  - An ISO file.
  - Hosted in an Amazon S3 bucket prefixed with `LaunchWizard-*`.
  - Included in a folder within the Amazon S3 bucket.
  - Included in a public folder so that Launch Wizard can download and install the media.

Requirements for using custom Windows AMIs (deployment on Windows)

We recommend that you use Amazon Windows license-included AMIs whenever possible. There are scenarios for which you may want to use a custom Windows AMI. For example, you may have existing licenses (BYOL), or you may have made changes to one of our public images and re-imaged it.

If you use Amazon Windows license-included AMIs, you are not required to perform any pre-checks on the AMI to ensure that it meets Launch Wizard requirements.

Launch Wizard relies on user data to begin the process of configuring SQL Server or RGW instances to launch in your account. For more information, see User Data Scripts. By default, all AWS Windows AMIs have user data execution enabled for the initial launch. To ensure that your custom AMIs are set up to run the User Data script at launch, follow the AWS recommended method to prepare your AMIs using EC2Launch v2. For more information about how to prepare your custom AMI using the options to `Shutdown with Sysprep` or `Shutdown without Sysprep`, see Create a Standard Amazon Machine Image Using Sysprep or EC2Launch v2 and Sysprep. If you want to directly enable user data as part of the custom AMI creation process, follow the steps for `Subsequent Reboots` or `Starts` under Run commands on your EC2 instance at launch.

If you use a custom Windows AMI, the volume drive letter for the root partition should be `C:` because EC2Launch v2 relies on this configuration to install the components.

While not exhaustive, the following requirements cover most of the configurations whose alteration might impact the successful deployment of a SQL Server Always On application using Launch Wizard.

Support matrix

SQL Server Version	Windows Server 2019	Windows Server 2022	Windows Server 2025
SQL Server 2019	YES	YES	YES
SQL Server 2022	YES	YES	YES
SQL Server 2025	YES	YES	YES

OS and SQL requirements

- Windows Server 2019 (Datacenter) (64-bit only)
- Windows Server 2022 (Datacenter) (64-bit only)
- MBR-partitioned volumes and GUID Partition Table (GPT) partitioned volumes that are formatted using the NTFS file system
- English language pack only
- SQL Server Enterprise Edition 2019 or Standard Edition 2019
- SQL Server Enterprise Edition 2022, Standard Edition 2022, or Developer Edition 2022
- SQL Server Standard Developer Edition 2025
- SQL Server Enterprise Developer Edition 2025
- The root volume drive for the custom AMI should be `C:`
- SQL Server is installed on the root drive

AWS software and drivers

- EC2Launch v2
- AWS SSM (SSM agent must be installed)
- AWS Tools for Windows PowerShell
- Network drivers (SRIOV, ENA)
- Storage drivers (NVMe, AWS PV)

Requirements for using custom Linux AMIs (deployment on Linux)

There are occasions when you may want to use a custom Linux AMI. For example, you may have existing licenses (BYOL), or you may have made changes to one of our public images and re-imaged it.

If you use a custom Linux AMI, you must adhere to the following requirements:

- The operating system must be Ubuntu version 18.04 LTS.
- The system installer and administrator must be a sudo user and be able to log in to the cluster nodes using SSH.
- SQL Server for Linux must be a default installation.
- The SQL Server for Linux version must be 2019.

- The latest Microsoft SQL tools must be installed.

Requirements for using Amazon FSx

Launch Wizard uses continuously available Amazon FSx file shares to host clustered databases. The Amazon FSx file shares are accessible from within an instance joined to the domain. You can either create a new Active Directory or connect to an existing Active Directory (managed or Self Managed). If you connect to an existing Active Directory, you can use preexisting security groups . The security groups must satisfy port and security requirements for FSx to communicate with the domain, as described in Using Amazon FSx with your Self Managed Microsoft Active Directory and Using Amazon FSx with AWS Directory Service for Microsoft Active Directory.

If you are using an existing AWS Managed Active Directory instance, you must specify the ID of the managed Active Directory instance for FSx to be able to join the domain. The account must have the same access rights in the domain as described in Using Amazon FSx with your Self Managed Microsoft Active Directory and Using Amazon FSx with AWS Directory Service for Microsoft Active Directory.

For Amazon FSx using NetApp ONTAP, Launch Wizard creates security groups in order to access the ONTAP file system and to set up failover clustering. For port requirements, see File System Access Control with Amazon VPC in the *Amazon FSx for NetApp ONTAP User Guide*.

> **Note**
>
> This Launch Wizard deployment relies on the instances that are being deployed to be able to connect to your ONTAP endpoint from within the VPC. For more information on the connectivity requirements, see Accessing data from within AWS in the *Amazon FSx for NetApp ONTAP User Guide*.

Backup schedule

Launch Wizard uses FSx defaults for setting up the backup schedule. You can change the default settings in the FSx console after the build completes.

The `WeeklyMaintenanceStartime` follows the format `day of the week:time`, where Monday is indicated by 1. The maintenance start time is set to begin on Saturday at 10pm.

```
WeeklyMaintenanceStartTime: '6:22:00'
```

```
DailyAutomaticBackupStartTime: '01:00'
AutomaticBackupRetentionDays: 7
```

Amazon FSx using NetApp ONTAP

Amazon FSx using NetApp ONTAP creates a new ONTAP file system for use with your Launch Wizard SQL deployment. We use the formulas in the following table to calculate volume and LUN storage for optimal performance.

These values can be modified post deployment.

Storage type	Size in GB	Sizing calculations
FSx storage	1024	Size in GB
Volume storage	870.4	85% of total storage FSx capacity
LUN storage	696.32	80% of volume storage (65% of total FSx storage)
SQL data LUN size	522.24	60% of LUN storage
SQL log LUN size	139.264	20% of SQL Data LUN size

Backup schedule for ONTAP

By default, ONTAP backups are disabled during builds. You can set your own backup schedule from the Amazon FSx console. Choose the **Backup** tab. Then, choose **Update** to update the backup settings.

> **Note**
>
> When you delete a Launch Wizard deployment that uses ONTAP, FSx creates a backup of the ONTAP volume before deleting the file system. You can delete the backup from the Amazon FSx console if it is not required. For more information, see Deleting backups in the *FSx for ONTAP User Guide*.

Configuration settings (deployment on Windows)

The following configuration settings are applied when deploying a SQL Server Always On application with Launch Wizard.

Setting	Applies to
Current EC2Launch v2 and SSM Agent	Windows Server 2022 and 2019*
Current EC2Launch and SSM Agent	Windows Server 2019*
Current AWS PV, ENA, and NVMe drivers	Windows Server 2022 and 2019
Current SRIOV drivers	Windows Server 2022 and 2019
Microsoft SQL Server: Latest service pack SQL Service configured to start automatically SQL Service running `BUILTIN\Administrators` added to the `SysAdmin` server role TCP port 1433 and UDP port 1434 open	Windows Server 2022 and 2019
Allow ICMP traffic through the firewall	Windows Server 2022 and 2019
Allow RDP traffic through host firewall	Windows Server 2022 and 2019
`RealTimeIsUniversal` registry key set	Windows Server 2022 and 2019
SQL Server FCI	Windows Server 2022 and 2019 SQL Server 2022 and 2019

* Windows Server 2019 can use either EC2Launch or EC2Launch v2 depending on what is configured in the AMI. For more information, see Supported AMIs.

The following AMI settings can impact the Launch Wizard deployment:

System Time

RealTimeIsUniversal. If disabled, Windows system time drifts when the time zone is set to a value other than UTC.

Windows Firewall

In most cases, Launch Wizard configures the correct protocols and ports. However, custom Windows Firewall rules could impact the cluster service. To ensure that your custom AMI works with Launch Wizard, see Service overview and network port requirements for Windows.

Remote Desktop

Service Start. Remote Desktop service must be enabled.

Remote Desktop Connections. Must be enabled.

Network Interface

DHCP Service Startup. DHCP service should be enabled.

DHCP on Ethernet. DHCP should be enabled.

Microsoft SQL Server

TCPIP. Must be enabled for protocols in SQL Configuration Manager.

PowerShell

Execution Policy. The execution policy in all AWS license-included AMIs is set to `Unrestricted`. We recommend that you set this policy to `Unrestricted` when you set up SQL Server Always On Availability Groups using Launch Wizard. You can change the policy when setup is complete.

Deploy an application with AWS Launch Wizard for SQL Server on Windows (Console)

Access AWS Launch Wizard

You can launch AWS Launch Wizard from the AWS Launch Wizard console.

Deploy AWS Launch Wizard on Windows

Deploy SQL Server Always On application

The following steps guide you through a SQL Server Always On application deployment with AWS Launch Wizard after you have launched it from the console.

1. When you select **Workload library** from the AWS Launch Wizard landing page, you are directed to the **Workload library** wizard, where you are prompted to select the type of application that you want to deploy. Select **Microsoft SQL Server**, then **Launch new deployment**.
2. From the **Choose deployment pattern** list of available deployment patterns, choose **SQL Server Always On - Windows** and then **Configure deployment**.
3. Under **Review Permissions**, Launch Wizard displays the AWS Identity and Access Management (IAM) role required for Launch Wizard to access other AWS services on your behalf. For more information about setting up IAM for Launch Wizard, see AWS Identity and Access Management (IAM). Choose **Next** .
4. On the **Configure application settings** page, select the **Operating System** on which you want to install SQL Server — in this case, **Windows**.
5. **Deployment model**. Choose **High availability deployment** to deploy your SQL Server Always On application across multiple Availability Zones or **Single instance deployment** to deploy your SQL Server application on a single node.
6. You are prompted to enter the specifications for the new deployment. The following tabs provide information about the specification fields.

   General

   - **Deployment name**. Enter a unique application name for your deployment.
   - (Optional) **Simple Notification Service (SNS) topic ARN**. Specify an SNS topic where AWS Launch Wizard can send notifications and alerts. For more information, see the *Amazon Simple Notification Service Developer Guide*.
   - (Optional for HA deployments) **CloudWatch application monitoring**. Select the check box to set up monitors and automated insights for your deployment using CloudWatch Application Insights. For more information, see the Amazon CloudWatch User Guide.
   - **Enable rollback on failed deployment**. By default, if a deployment fails, your provisioned resources will not be rolled back/deleted. This default configuration helps

you to troubleshoot errors at the resource level as you debug deployment issues. If you want your provisioned resources to be immediately deleted if a deployment fails, select the check box.

Connectivity

Enter specifications for how you want to connect to your instance and configure your Virtual Private Cloud (VPC).

Key pair name

- Select an existing key pair from the dropdown list or create a new one. If you select **Create new key pair name**, you are directed to the Amazon EC2 console. From there, under **Network and Security**, choose **Key Pairs**. Choose **Create a new key pair**, enter a name for the key pair, and then choose **Download Key Pair**.

> ⚠ **Important**
>
> This is the only opportunity for you to save the private key file. Download it and save it in a safe place. You must provide the name of your key pair when you launch an instance and provide the corresponding private key each time that you connect to the instance.

Return to the Launch Wizard console and choose the refresh button next to the **Key Pairs** dropdown list. The newly created key pair appears in the dropdown list. For more information about key pairs, see Amazon EC2 Key Pairs and Windows Instances.

Tenancy model (HA deployments only)

Select your preferred tenancy. Each instance that you launch into a VPC has a tenancy attribute. The **Shared** tenancy option means that the instance runs on shared hardware. The **Dedicated Host (HA deployments)** tenancy option means that the instance runs on a Dedicated Host, which is an isolated server with configurations that you can control. For more information, see Dedicated Hosts.

Virtual Private Cloud (VPC). Choose whether you want to use an existing VPC or create a new VPC.

- **Select Virtual Private Cloud (VPC)** option. Choose the VPC that you want to use from the dropdown list. If you choose to enable Remote Desktop Gateway access on single-node deployments, then your VPC must include one public subnet and one private subnet. It must include at least two private subnets for HA deployments . Your VPC must be associated with a DHCP Options Set to enable DNS translations to work. The private subnets must have outbound connectivity to the internet and other AWS services (S3, CFN, SSM, Logs). We recommend that you enable this connectivity with a NAT Gateway. For more information about NAT Gateways, see NAT Gateways in the Amazon VPC User Guide.
  - **Public Subnet**. If you choose to enable Remote Desktop Gateway access on single-node deployments, then your VPC must include one public subnet and one private subnet. It must include at least two private subnets for HA deployments. Choose a public subnet for your VPC from the dropdown list. To continue, you must select the check box that indicates that the public subnet has been set up and the private subnets have outbound connectivity enabled.

    To add a new public subnet

    If a subnet's traffic is routed to an internet gateway, the subnet is known as a public subnet. If, however, a subnet doesn't have a route to the internet gateway, the subnet is known as a private subnet. To use an existing VPC that does not have a public subnet, you can add a new public subnet using the following steps.

    - Follow the steps in Creating a Subnet in the Amazon VPC User Guide using the existing VPC you intend to use AWS Launch Wizard.
    - To add an internet gateway to your VPC, follow the steps in Attaching an Internet Gateway in the Amazon VPC User Guide.
    - To configure your subnets to route internet traffic through the internet gateway, follow the steps in Creating a Custom Route Table in the Amazon VPC User Guide. Use IPv4 format (0.0.0.0/0) for Destination.
    - The public subnet should have the “auto-assign public IPv4 address” setting enabled. To enable this setting, follow the steps in Modifying the Public IPv4 Addressing Attribute for Your Subnet in the Amazon VPC User Guide.
  - **Availability Zone (AZ) configuration**. You must choose at least two Availability Zones for High Availability (HA) deployments and one Availability Zone for single-node deployments, with one private subnet for each zone that you select. For HA

deployments, select the **Availability Zones** within which you want to deploy your **primary** and **secondary** SQL nodes. Depending on the number of secondary nodes that you plan to use to set up a SQL Server Always On deployment, you may have to specify a **private subnet** for each of them. Cross-Region replication is not supported.

To create a private subnet

If a subnet doesn't have a route to an internet gateway, the subnet is known as a private subnet. To create a private subnet, you can use the following steps. We recommend that you enable the outbound connectivity for each of your selected private subnets using a NAT Gateway. To enable outbound connectivity from private subnets with public subnet, see the steps in Creating a NAT Gateway to create a NAT Gateway in your chosen public subnet. Then, follow the steps in Updating Your Route Table for each of your chosen private subnets.

- Follow the steps in Creating a Subnet in the Amazon VPC User Guide using the existing VPC you will use in AWS Launch Wizard.
- When you create a VPC, it includes a main route table by default. On the **Route Tables** page in the Amazon VPC console, you can view the main route table for a VPC by looking for Yes in the Main column. The main route table controls the routing for all subnets that are not explicitly associated with any other route table. If the main route table for your VPC has an outbound route to an internet gateway, then any subnet created using the previous step, by default, becomes a public subnet. To ensure the subnets are private, you may need to create separate route table(s) for your private subnets. These route tables must not contain any routes to an internet gateway. Alternatively, you can create a custom route table for your public subnet and remove the internet gateway entry from the main route table.

If you selected **Dedicated host** tenancy, you must select a Dedicated Host for each Availability Zone. If you have not allocated any dedicated hosts to your account, you can choose **Create new dedicated host** to do so from the EC2 console.

- **Remote Desktop Gateway preferences (single-node deployments only)**. When you select **Set up Remote Desktop Gateway**, enter the public subnet into which to deploy the RDGW instance.
- (Optional) **Remote Desktop Gateway access**. Select **Custom IP** from the dropdown list. Enter the CIDR block. If you do not specify any value for the Custom IP parameter, Launch Wizard does not set the inbound RDP access (Port 3389) from any IP. You can choose to do this later by modifying the security group settings via the Amazon

EC2 console. See Adding a Rule for Inbound RDP Traffic to a Windows Instance for instructions on adding a rule that allows inbound RDP traffic to your RDGW instance.

- **Create new Virtual Private Cloud (VPC)** option. Launch Wizard creates your VPC. You can optionally enter a **VPC name tag**. If you selected **Dedicated Host** tenancy for high availability deployments, select a primary and secondary Dedicated Host. If you haven't allocated any Dedicated Hosts to your account, select **Create a new dedicated host**. You will be directed to the EC2 console to create the new host.
  - **Remote Desktop Gateway preferences (single-node deployments only)**. When you select **Set up Remote Desktop Gateway**, only the Remote Desktop Gateway access information will be taken from the VPC.
  - (Optional) **Remote Desktop Gateway access**. Select **Custom IP** from the dropdown list. Enter the CIDR block. If you do not specify any value for the Custom IP parameter, Launch Wizard does not set the inbound RDP access (Port 3389) from any IP. You can choose to do this later by modifying the security group settings via the Amazon EC2 Console. See Adding a Rule for Inbound RDP Traffic to a Windows Instance for instructions on adding a rule that allows inbound RDP traffic to your RDGW instance.

Active Directory

You can connect to an existing Active Directory or, for high availability deployments, you can create a new one. If you selected the **Create new Virtual Private Cloud (VPC)** option for high availability deployments, you must select **Create a new Active Directory**.

Connecting to existing AWS Managed Active Directory or Self Managed Active Directory

From the dropdown list, select whether you want to use **AWS Managed Active Directory**, or **Self Managed Active Directory**. If you select **Self Managed Active Directory**, select the check box to verify that you have ensured a connection between the Active Directory and the VPC.

Follow the steps for granting permissions in the Active Directory Default Organizational Unit (OU).

- **Domain user name and password**. Enter the user name and password for your directory. For required permissions for the domain user, see Active Directory (Windows deployment). Launch Wizard stores the password in AWS Secrets Manager as a secure string parameter. It does not store the password on the service side. To create a functional SQL Server Always On deployment, it reads from AWS Secrets Manager.

- **DNS address**. Enter the IP address of the DNS servers to which you are connecting. These servers must be reachable from within the VPC that you selected.
- **Optional DNS address**. If you would like to use a backup DNS server, enter the IP address of the DNS server that you want to use as backup. These servers must be reachable from within the VPC that you selected.
- **Domain DNS name**. Enter the Fully Qualified Domain Name (FQDN) of the forest root domain used for the Active Directory. When you choose to create a new Active Directory, Launch Wizard creates a domain admin user on your Active Directory.

Creating a new AWS Managed Active Directory through Launch Wizard

- **Domain user name and password**. The domain user name is preset to “admin.” Enter a password for your directory. Launch Wizard stores the password in AWS Secrets Manager as a secure string parameter. It does not store the password on the server side. To create a functional SQL Server Always On deployment, it reads from AWS Secrets Manager.
- **Domain DNS name**. Enter a Fully Qualified Domain Name (FQDN) of the forest root domain used for the Active Directory. When you choose to create a new Active Directory, Launch Wizard creates a domain admin user on your Active Directory.

Connecting to a Self Managed Active Directory through Launch Wizard

Launch Wizard allows you to connect to a Self Managed Active Directory environment during deployment. For more information, see Self Managed Active Directory.

SQL Server

When you use an existing Active Directory, you have the option of using an existing SQL Server service account or creating a new account. If you create a new Active Directory account, you must create a new SQL Server account.

- **User name and password**. If you are using an existing SQL Server service account, provide your user name and password. This SQL Server service account should be part of the Managed Active Directory in which you are deploying. If you are creating a new SQL Server service account through Launch Wizard, enter a user name for the SQL Server service account. Create a complex Password that is at least 8 characters long, and then reenter the password to verify it. See Password Policy for more information.

- **SQL Server install type**. Select the version of SQL Server Enterprise that you want to deploy. You can select an AMI from either the License-included AMI or Custom AMI dropdown lists.
- **License-included AMI**. Choose an AMI for your SQL Server deployment which determines the version and edition of Windows Server and SQL Server that will be deployed.
- (Optional) **tempdb configuration**. To improve performance, you can opt for the SQL Server tempdb system database to reside on a local NVMe SSD ephemeral storage device, also called the (instance store volume). NVMe SSD instance store volumes are available only on instance types that provide these local storage devices. Additionally, only data that changes frequently should ever reside on these volumes. They are not intended to store data long-term. For more information, see Amazon EC2 instance store.
- (Optional) **Additional SQL Server settings**. You can specify the following:
  - **Nodes**. Enter a **Primary SQL node name** and a **Secondary SQL node name (HA deployments only)**.
  - **Additional secondary SQL node (HA deployments only, maximum of 5)**. Enter a secondary **Node name**, and select the **Access type**, the **Private subnet**, and the **Dedicated host**, if applicable, for the additional secondary SQL node from the dropdown lists. You can add more secondary nodes by selecting **Add additional secondary node**.
  - (Optional, HA deployments only) **Witness node**. For improved fault tolerance, select the check box to add a file share quorum witness node.
  - **Additional naming**. Enter a **Database name**. For HA deployments, enter an **Availability group name**, a **Listener name**, and a **Windows cluster virtual network name**.

7. When you are satisfied with your configuration selections, select **Next**. If you don't want to complete the configuration, select **Cancel**. When you select **Cancel**, all of the selections on the specification page are lost and you are returned to the landing page. To go to the previous screen, select **Previous**.

8. After configuring your application, you are prompted to define the infrastructure requirements for the new deployment on the **Define infrastructure requirements** page. The following tabs provide information about the input fields.

Define infrastructure requirements

You can choose to select your instances and volume types, or to use AWS recommended resources. If you choose to use AWS recommended resources, you have the option of defining your high availability cluster needs. If no selections are made, default values are assigned.

- **Number of instance cores**. Choose the number of CPU cores for your infrastructure. The default value assigned is 4.
- **Network performance**. Choose your preferred network performance in Gbps.
- **Memory (GB)**. Choose the amount of RAM that you want to attach to your EC2 instances. The default value assigned is 4 GB.
- **Type of storage drive**. Select the storage drive type for the SQL data and tempdb volumes. If you chose to place your tempdb on local storage, only the SQL data will be on the storage drive you select. The default value assigned is SSD.
- **SQL Server throughput**. Select the sustained SQL Server throughput that you need.
- **Recommended resources**. Launch Wizard displays the system-recommended resources based on your infrastructure selections. If you want to change the recommended resources, select different infrastructure requirements.

Infrastructure requirements based on instance type

You can choose to select your instance and volume type, or to use AWS recommended resources. If no selections are made, default values are assigned.

- **Instance type**. Select your preferred instance type from the dropdown list.
- **Volume type**. Choose your preferred EBS volume type. For more information about volume types, see Amazon EBS volume types.

Drive letters and volume size

- **Drive letter**. Select the storage drive letter for **Root drive**, **Logs**, **Data**, and **Backup** volumes.

> **Important**
>
> For custom AMIs, Launch Wizard assumes the root volume drive is `C:`.

- **Volume size**. Select the size of the SQL Server data volume in Gb for **Root drive**, **Logs**, **Data**, and **Backup** volumes. SQL Server logs and data will be staged on the same data volume for this deployment. Make sure that you select an adequate size for the data volume.

> **Note**
>
> For Launch Wizard deployments created after January 2023, IMDSv1 is disabled on all instances. If your software or scripts use IMDSv1, you will have to meet the requirements to use IMDSv2. For more information, see Use IMDSv2.

(Optional) Tags

You can provide optional custom tags for the resources Launch Wizard creates on your behalf. For example, you can set different tags for EC2 instances, EBS volumes, VPC, and subnets. If you select **All**, you can assign a common set of tags to your resources. Launch Wizard assigns tags with a fixed key `LaunchWizardResourceGroupID` and value that corresponds to the ID of the AWS resource group created for a deployment. Launch Wizard does not support custom tagging for root volumes.

Estimated on-demand cost to deploy additional resources

AWS Launch Wizard provides an estimate for application charges incurred to deploy the selected resources. The estimate updates each time you change a resource type in the Wizard. The provided estimates are only for general comparisons. They are based upon On-Demand costs and your actual costs may be lower.

9. When you are satisfied with your infrastructure selections, select **Next**. If you don't want to complete the configuration, select **Cancel**. When you select **Cancel**, all of the selections on the specification page are lost and you are returned to the landing page. To go to the previous screen, select **Previous**.

10. On the **Review and deploy** page, review your configuration details. If you want to make changes, select **Previous**. To stop, select **Cancel**. When you select **Cancel**, all of the selections

on the specification page are lost and you are returned to the landing page. When you choose **Deploy**, you agree to the terms of the **Acknowledgment**.

11. Launch Wizard validates the inputs and notifies you of any issues you must address.
12. When validation is complete, Launch Wizard deploys your AWS resources and configures your SQL Server Always On application. Launch Wizard provides you with status updates about the progress of the deployment on the **Deployments** page. From the **Deployments** page, you can view the list of current and previous deployments.
13. When your deployment is ready, a notification informs you that your SQL Server application is successfully deployed. If you have set up an SNS notification, you are also alerted through SNS. You can manage and access all of the resources related to your SQL Server Always On application by selecting the deployment, and then selecting **Manage** from the **Actions** dropdown list.
14. When the SQL Server Always On application is deployed, you can access your Amazon EC2 instances through the EC2 console. You can also use AWS SSM to manage your SQL Server Always On application for future updates and patches through built-in integration via resource groups.

Deploy SQL Failover Clustering application

The following steps guide you through a SQL Failover Clustering application deployment with AWS Launch Wizard after you have launched it from the console.

1. When you select **Workload library** from the AWS Launch Wizard landing page, you are directed to the **Workload library** wizard, where you are prompted to select the type of application that you want to deploy. Select **Microsoft SQL Server**, then **Launch new deployment**.
2. From the **Choose deployment pattern** list of available deployment patterns, choose **SQL Server Failover Clustering - Windows** and then **Configure deployment**.
3. Under **Review Permissions**, Launch Wizard displays the AWS Identity and Access Management (IAM) role required for Launch Wizard to access other AWS services on your behalf. For more information about setting up IAM for Launch Wizard, see AWS Identity and Access Management (IAM). Choose **Next** .
4. On the **Configure application settings** page, select the **Operating System** on which you want to install SQL Server — in this case, **Windows**.

5. **Deployment model**. Choose **High availability deployment**, and then choose **Always On Failover Cluster Instances** to deploy a SQL Server Failover Clustering (FCI) application across multiple Availability Zones.

6. You are prompted to enter the specifications for the new deployment The following tabs provide information about the specification fields.

   General

   - **Deployment name**. Enter a unique application name for your deployment.
   - (Optional) **Simple Notification Service (SNS) topic ARN**. Specify an SNS topic where AWS Launch Wizard can send notifications and alerts. For more information, see the *Amazon Simple Notification Service Developer Guide*.
   - (Optional for HA deployments) **CloudWatch application monitoring**. Select the check box to set up monitors and automated insights for your deployment using CloudWatch Application Insights. For more information, see the Amazon CloudWatch User Guide.
   - **Enable rollback on failed deployment**. By default, if a deployment fails, your provisioned resources will not be rolled back/deleted. This default configuration helps you to troubleshoot errors at the resource level as you debug deployment issues. If you want your provisioned resources to be immediately deleted if a deployment fails, select the check box.

   Connectivity

   Enter the specifications for how you want to connect to your instance and configure your Virtual Private Cloud (VPC).

   Key pair name

   - Select an existing key pair from the dropdown list or create a new one. If you select **Create new key pair name**, you are directed to the Amazon EC2 console. From there, under **Network and Security**, choose **Key Pairs**. Choose **Create a new key pair**, enter a name for the key pair, and then choose **Download Key Pair**.

     > ⚠ **Important**
     >
     > This is the only opportunity for you to save the private key file. Download it and save it in a safe place. You must provide the name of your key pair when you

launch an instance and provide the corresponding private key each time that you connect to the instance.

Return to the Launch Wizard console and choose the refresh button next to the **Key Pairs** dropdown list. The newly created key pair appears in the dropdown list. For more information about key pairs, see Amazon EC2 Key Pairs and Windows Instances.

Tenancy model (HA deployments only)

Select your preferred tenancy. Each instance that you launch into a VPC has a tenancy attribute. The **Shared** tenancy option means that the instance runs on shared hardware. The **Dedicated Host (HA deployments)** tenancy option means that the instance runs on a Dedicated Host, which is an isolated server with configurations that you can control. For FCI deployments, select **Shared** tenancy.

Virtual Private Cloud (VPC). Choose whether you want to use an existing VPC or create a new VPC.

- **Select Virtual Private Cloud (VPC)** option. Choose the VPC that you want to use from the dropdown list. If you choose to enable Remote Desktop Gateway access, then your VPC must include at least one public subnet and two private subnets for HA deployments . Your VPC must be associated with a DHCP Options Set to enable DNS translations to work. The private subnets must have outbound connectivity to the internet and other AWS services (S3, CFN, SSM, Logs). We recommend that you enable this connectivity with a NAT Gateway. For more information about NAT Gateways, see NAT Gateways in the Amazon VPC User Guide.
  - **Public Subnet**. If you choose to enable Remote Desktop Gateway access, then your VPC must include at least one public subnet and two private subnets for HA deployments. Choose a public subnet for your VPC from the dropdown list. To continue, you must select the check box that indicates that the public subnet has been set up and the private subnets have outbound connectivity enabled.

    If a subnet's traffic is routed to an internet gateway, the subnet is known as a public subnet. If, however, a subnet doesn't have a route to the internet gateway, the subnet is known as a private subnet.

To use an existing VPC that does not have a public subnet, you can add a new public subnet using the following steps:

- Follow the steps in Creating a Subnet in the Amazon VPC User Guide using the existing VPC you intend to use AWS Launch Wizard.
- To add an internet gateway to your VPC, follow the steps in Attaching an Internet Gateway in the Amazon VPC User Guide.
- To configure your subnets to route internet traffic through the internet gateway, follow the steps in Creating a Custom Route Table in the Amazon VPC User Guide. Use IPv4 format (0.0.0.0/0) for Destination.
- The public subnet should have the “auto-assign public IPv4 address” setting enabled. To enable this setting, follow the steps in Modifying the Public IPv4 Addressing Attribute for Your Subnet in the Amazon VPC User Guide.

- **Availability Zone (AZ) configuration**. You must choose at least two Availability Zones for High Availability (HA) deployments, with one private subnet for each zone that you select. For HA deployments, select the **Availability Zones** within which you want to deploy your **primary** and **secondary** SQL nodes. Depending on the number of secondary nodes that you plan to use to set up a SQL Server Always On deployment, you may have to specify a **private subnet** for each of them. Cross-Region replication is not supported.

  To create a private subnet

  If a subnet doesn't have a route to an internet gateway, the subnet is known as a private subnet. To create a private subnet, you can use the following steps. We recommend that you enable the outbound connectivity for each of your selected private subnets using a NAT Gateway. To enable outbound connectivity from private subnets with public subnet, see the steps in Creating a NAT Gateway to create a NAT Gateway in your chosen public subnet. Then, follow the steps in Updating Your Route Table for each of your chosen private subnets.

  - Follow the steps in Creating a Subnet in the Amazon VPC User Guide using the existing VPC you will use in AWS Launch Wizard.
  - When you create a VPC, it includes a main route table by default. On the **Route Tables** page in the Amazon VPC console, you can view the main route table for a VPC by looking for Yes in the Main column. The main route table controls the routing for all subnets that are not explicitly associated with any other route table. If the

main route table for your VPC has an outbound route to an internet gateway, then any subnet created using the previous step, by default, becomes a public subnet. To ensure the subnets are private, you may need to create separate route table(s) for your private subnets. These route tables must not contain any routes to an internet gateway. Alternatively, you can create a custom route table for your public subnet and remove the internet gateway entry from the main route table.

  - **Remote Desktop Gateway preferences**. When you select **Set up Remote Desktop Gateway**, enter the public subnet into which to deploy the RDGW instance.
  - (Optional) **Remote Desktop Gateway access**. Select **Custom IP** from the dropdown list. Enter the CIDR block. If you do not specify any value for the Custom IP parameter, Launch Wizard does not set the inbound RDP access (Port 3389) from any IP. You can choose to do this later by modifying the security group settings via the Amazon EC2 console. See Adding a Rule for Inbound RDP Traffic to a Windows Instance for instructions on adding a rule that allows inbound RDP traffic to your RDGW instance.

- **Create new Virtual Private Cloud (VPC)** option. Launch Wizard creates your VPC. You can optionally enter a **VPC name tag**.
  - **Remote Desktop Gateway preferences**. When you select **Set up Remote Desktop Gateway**, only the Remote Desktop Gateway access information will be taken from the VPC.
  - (Optional) **Remote Desktop Gateway access**. Select **Custom IP** from the dropdown list. Enter the CIDR block. If you do not specify any value for the Custom IP parameter, Launch Wizard does not set the inbound RDP access (Port 3389) from any IP. You can choose to do this later by modifying the security group settings via the Amazon EC2 Console. See Adding a Rule for Inbound RDP Traffic to a Windows Instance for instructions on adding a rule that allows inbound RDP traffic to your RDGW instance.

Active Directory

You can connect to an existing Active Directory or create a new one. If you selected the **Create new Virtual Private Cloud (VPC)** option for high availability deployments, you must select **Create a new Active Directory**.

Connecting to existing AWS Managed Active Directory or Self Managed Active Directory

From the dropdown list, select whether you want to use **AWS Managed Active Directory**, or **Self Managed Active Directory**. If you select **Self Managed Active Directory**, select the check box to verify that you have ensured a connection between the Active Directory and the VPC.

Follow the steps for granting permissions in the Active Directory Default Organizational Unit (OU).

- **Domain user name and password**. Enter the user name and password for your directory. For required permissions for the domain user, see Active Directory (Windows deployment). Launch Wizard stores the password in AWS Secrets Manager as a secure string parameter. It does not store the password on the service side. To create a functional SQL Server FCI deployment, Launch Wizard reads from AWS Secrets Manager.
- **DNS address**. Enter the IP address of the DNS servers to which you are connecting. These servers must be reachable from within the VPC that you selected.
- **Optional DNS address**. If you would like to use a backup DNS server, enter the IP address of the DNS server that you want to use as backup. These servers must be reachable from within the VPC that you selected.
- **Domain DNS name**. Enter the Fully Qualified Domain Name (FQDN) of the forest root domain used for the Active Directory. When you choose to create a new Active Directory, Launch Wizard creates a domain admin user on your Active Directory.
- (Optional) **Domain User security group**. To specify an existing security group, select one from the dropdown list. The prerequisites for adding security groups can be viewed by selecting **Info**.

Creating a new AWS Managed Active Directory through Launch Wizard

- **Domain user name and password**. The domain user name is preset to “admin.” Enter a password for your directory. Launch Wizard stores the password in AWS Secrets Manager as a secure string parameter. It does not store the password on the server side. To create a functional SQL Server FCI deployment, Launch Wizard reads from AWS Secrets Manager.

- **Domain DNS name**. Enter a Fully Qualified Domain Name (FQDN) of the forest root domain used for the Active Directory. When you choose to create a new Active Directory, Launch Wizard creates a domain admin user on your Active Directory.

Connecting to a Self Managed Active Directory through Launch Wizard

Launch Wizard allows you to connect to a Self Managed Active Directory environment during deployment. For more information, see Self Managed Active Directory.

SQL Server

When you use an existing Active Directory, you have the option of using an existing SQL Server service account or creating a new account. If you create a new Active Directory account, you must create a new SQL Server account.

- **User name and password**. If you are using an existing SQL Server service account, provide your user name and password. This SQL Server service account should be part of the Managed Active Directory in which you are deploying. If you are creating a new SQL Server service account through Launch Wizard, enter a user name for the SQL Server service account. Create a complex Password that is at least 8 characters long, and then reenter the password to verify it. See Password Policy for more information.
- **SQL Server install type**. Select the version of SQL Server Enterprise that you want to deploy. You can select an AMI from either the License-included AMI or Custom AMI dropdown lists.
- **License-included AMI**. Choose an AMI for your SQL Server deployment which determines the version and edition of Windows Server and SQL Server that will be deployed.
- (Optional) **Additional SQL Server settings**. You can optionally specify the following:
  - **Nodes**. Enter a **Primary SQL node name** and a **Secondary SQL node name**.
  - **Additional naming**. Enter a **SQL Server virtual network name** and a **Windows cluster virtual network name**.

7. When you are satisfied with your configuration selections, select **Next**. If you don't want to complete the configuration, select **Cancel**. When you select **Cancel**, all of the selections on the specification page are lost and you are returned to the landing page. To go to the previous screen, select **Previous**.

8. After configuring your application, you are prompted to define the infrastructure requirements for the new deployment on the **Define infrastructure requirements** page. The following tabs provide information about the input fields.

   Define infrastructure requirements

   You can choose to select your instances and volume types, or to use AWS recommended resources. If you choose to use AWS recommended resources, you have the option of defining your high availability cluster needs. If no selections are made, default values are assigned.

   Instances

   - **Cores**. Choose the number of CPU cores for your infrastructure. The default value assigned is 4.
   - **Network performance**. Choose your preferred network performance in Gbps.
   - **Memory (GB)**. Choose the amount of RAM that you want to attach to your EC2 instances. The default value assigned is 4 GB.

   Storage and performance

   - **Type of storage drive**. The default value assigned is SSD for FCI application deployments.
   - **Average and peak IOPS**. Select the average and peak IOPS required for your FSx share.
   - **Allocated storage space**. Select the amount of storage required for your FSx drive.
   - **Recommended resources**. Launch Wizard displays the system-recommended resources based on your infrastructure selections. If you want to change the recommended resources, select different infrastructure requirements.

   Infrastructure requirements based on instance type

   You can choose to select your instance and storage capacity, or to use AWS recommended resources. If no selections are made, default values are assigned.

   - **Instance type**. Select your preferred instance type from the dropdown list.
   - **Storage capacity**. Choose your preferred EBS volume type. For more information about volume types, see Amazon EBS volume types.

- **Throughput capacity**. Select the required sustained SQL Server throughput.

> **Note**
>
> For Launch Wizard deployments created after January 2023, IMDSv1 is disabled on all instances. If your software or scripts use IMDSv1, you will have to meet the requirements to use IMDSv2. For more information, see Use IMDSv2.

Tags-Optional

You can provide optional custom tags for the resources Launch Wizard creates on your behalf. For example, you can set different tags for EC2 instances, EBS volumes, VPC, and subnets. If you select **All**, you can assign a common set of tags to your resources. Launch Wizard assigns tags with a fixed key `LaunchWizardResourceGroupID` and value that corresponds to the ID of the AWS resource group created for a deployment. Launch Wizard does not support custom tagging for root volumes.

Estimated on-demand cost to deploy additional resources

AWS Launch Wizard provides an estimate for application charges incurred to deploy the selected resources. The estimate updates each time you change a resource type in the Wizard. The provided estimates are only for general comparisons. They are based upon On-Demand costs and your actual costs may be lower.

9. When you are satisfied with your infrastructure selections, select **Next**. If you don't want to complete the configuration, select **Cancel**. When you select **Cancel**, all of the selections on the specification page are lost and you are returned to the landing page. To go to the previous screen, select **Previous**.
10. On the **Review and deploy** page, review your configuration details. If you want to make changes, select **Previous**. To stop, select **Cancel**. When you select **Cancel**, all of the selections on the specification page are lost and you are returned to the landing page. When you choose **Deploy**, you agree to the terms of the **Acknowledgment**.
11. Launch Wizard validates the inputs and notifies you of any issues you must address.
12. When validation is complete, Launch Wizard deploys your AWS resources and configures your SQL Server FCI application. Launch Wizard provides you with status updates about the progress of the deployment on the **Deployments** page. From the **Deployments** page, you can view the list of current and previous deployments.

13. When your deployment is ready, a notification informs you that your SQL Server application is successfully deployed. If you have set up an SNS notification, you are also alerted through SNS. You can manage and access all of the resources related to your SQL Server FCI application by selecting the deployment, and then selecting **Manage** from the **Actions** dropdown list.
14. When the SQL Server FCI application is deployed, you can access your Amazon EC2 instances through the EC2 console. You can also use AWS SSM to manage your SQL Server FCI application for future updates and patches through built-in integration via resource groups.

Deploy SQL Server Developer Edition application

The following steps guide you through a SQL Server Developer Edition application deployment with AWS Launch Wizard after you have launched it from the console.

1. When you select **Workload library** from the AWS Launch Wizard landing page, you are directed to the **Workload library** wizard, where you are prompted to select the type of application that you want to deploy. Select **Microsoft SQL Server**, then **Launch new deployment**.
2. From the **Choose deployment pattern** list of available deployment patterns, choose **SQL Server Developer Edition - Single Node - Windows** and then **Configure deployment**.
3. Under **Review Permissions**, Launch Wizard displays the AWS Identity and Access Management (IAM) role required for Launch Wizard to access other AWS services on your behalf. For more information about setting up IAM for Launch Wizard, see AWS Identity and Access Management (IAM). Choose **Next** .
4. On the **Configure application settings** page, you are prompted to enter the specifications for the new deployment. The following tabs provide information about the specification fields.

   General

   - **Deployment name**. Enter a unique application name for your deployment.
   - **CloudWatch application monitoring**. Select the check box to set up monitors and automated insights for your deployment using CloudWatch Application Insights. For more information, see the Amazon CloudWatch User Guide.
   - **Enable rollback on failed deployment**. By default, if a deployment fails, your provisioned resources will not be rolled back/deleted. This default configuration helps you to troubleshoot errors at the resource level as you debug deployment issues. If you

want your provisioned resources to be immediately deleted if a deployment fails, select the check box.

- **AWS Service Catalog product creation**. Select the check box to export the CloudFormation template to create AWS Service Catalog for this deployment. When enabled, you must specify an S3 bucket location to store the CloudFormation templates and application configuration scripts for Service Catalog. You can select an existing bucket or create a new one using the provided link.
- (Optional) **Amazon Simple Notification Service (SNS) topic ARN**. Specify an SNS topic where AWS Launch Wizard can send notifications and alerts. For more information, see the Amazon Simple Notification Service Developer Guide.

Connectivity

Enter specifications for how you want to connect to your instance and configure your Virtual Private Cloud (VPC).

Key pair name

- Select an existing key pair from the dropdown list or create a new one. If you select **Create new key pair name**, you are directed to the Amazon EC2 console. From there, under **Network and Security**, choose **Key Pairs**. Choose **Create a new key pair**, enter a name for the key pair, and then choose **Download Key Pair**.

> ⚠ **Important**
>
> This is the only opportunity for you to save the private key file. Download it and save it in a safe place. You must provide the name of your key pair when you launch an instance and provide the corresponding private key each time that you connect to the instance.

Return to the Launch Wizard console and choose the refresh button next to the **Key Pairs** dropdown list. The newly created key pair appears in the dropdown list. For more information about key pairs, see Amazon EC2 Key Pairs and Windows Instances.

Virtual Private Cloud (VPC). Choose whether you want to use an existing VPC or create a new VPC.

- **Select Virtual Private Cloud (VPC)** option. Choose the VPC that you want to use from the dropdown list. If you choose to enable Remote Desktop Gateway access, then your VPC must include one public subnet and one private subnet. Your VPC must be associated with a DHCP Options Set to enable DNS translations to work. The private subnets must have outbound connectivity to the internet and other AWS services (S3, CFN, SSM, Logs). We recommend that you enable this connectivity with a NAT Gateway. For more information about NAT Gateways, see NAT Gateways in the Amazon VPC User Guide.
    - **Public Subnet**. If you choose to enable Remote Desktop Gateway access, then your VPC must include one public subnet and one private subnet. Choose a public subnet for your VPC from the dropdown list. To continue, you must select the check box that indicates that the public subnet has been set up and the private subnet has outbound connectivity enabled.

      If a subnet's traffic is routed to an internet gateway, the subnet is known as a public subnet. If, however, a subnet doesn't have a route to the internet gateway, the subnet is known as a private subnet.

      To use an existing VPC that does not have a public subnet, you can add a new public subnet using the following steps:
        - Follow the steps in Creating a Subnet in the Amazon VPC User Guide using the existing VPC you intend to use for AWS Launch Wizard.
        - To add an internet gateway to your VPC, follow the steps in Attaching an Internet Gateway in the Amazon VPC User Guide.
        - To configure your subnets to route internet traffic through the internet gateway, follow the steps in Creating a Custom Route Table in the Amazon VPC User Guide. Use IPv4 format (0.0.0.0/0) for Destination.
        - The public subnet should have the “auto-assign public IPv4 address” setting enabled. To enable this setting, follow the steps in Modifying the Public IPv4 Addressing Attribute for Your Subnet in the Amazon VPC User Guide.
    - **Availability Zone (AZ) configuration**. You must choose an Availability Zone and a private subnet for that zone. Cross-Region replication is not supported.

      If a subnet doesn't have a route to an internet gateway, the subnet is known as a private subnet. To create a private subnet, you can use the following steps. We recommend that you enable the outbound connectivity for your selected private subnet using a NAT Gateway.

To enable outbound connectivity from the selected private subnet with public subnet, see the steps in Creating a NAT Gateway to create a NAT Gateway in your chosen public subnet.

Then, follow the steps in Updating Your Route Table for your chosen private subnet.

        - Follow the steps in Creating a Subnet in the Amazon VPC User Guide using the existing VPC you will use in AWS Launch Wizard.
        - When you create a VPC, it includes a main route table by default. On the **Route Tables** page in the Amazon VPC console, you can view the main route table for a VPC by looking for Yes in the Main column. The main route table controls the routing for all subnets that are not explicitly associated with any other route table. If the main route table for your VPC has an outbound route to an internet gateway, then any subnet created using the previous step, by default, becomes a public subnet. To ensure the subnets are private, you may need to create separate route table(s) for your private subnets. These route tables must not contain any routes to an internet gateway. Alternatively, you can create a custom route table for your public subnet and remove the internet gateway entry from the main route table.
    - **Remote Desktop Gateway preferences (single-node deployments only)**. When you select **Set up Remote Desktop Gateway**, enter the public subnet into which to deploy the RDGW instance.
    - (Optional) **Remote Desktop Gateway access**. Select **Custom IP** from the dropdown list. Enter the CIDR block. If you do not specify any value for the Custom IP parameter, Launch Wizard does not set the inbound RDP access (Port 3389) from any IP. You can choose to do this later by modifying the security group settings via the Amazon EC2 console. See Adding a Rule for Inbound RDP Traffic to a Windows Instance for instructions on adding a rule that allows inbound RDP traffic to your RDGW instance.

- **Create new Virtual Private Cloud (VPC)** option. Launch Wizard creates your VPC. You can optionally enter a **VPC name tag**.
    - **Remote Desktop Gateway preferences**. When you select **Set up Remote Desktop Gateway**, only the Remote Desktop Gateway access information will be taken from the VPC.

- (Optional) **Remote Desktop Gateway access**. Select **Custom IP** from the dropdown list. Enter the CIDR block. If you do not specify any value for the Custom IP parameter, Launch Wizard does not set the inbound RDP access (Port 3389) from any IP. You can choose to do this later by modifying the security group settings via the Amazon EC2 Console. See Adding a Rule for Inbound RDP Traffic to a Windows Instance for instructions on adding a rule that allows inbound RDP traffic to your RDGW instance.

(Optional) Active Directory

SQL Server Developer Edition can be deployed with or without Active Directory integration. To enable Active Directory with the SQL Server Developer Edition Single Node deployment, set the **Enable Active Directory** toggle to on.

You can connect to an existing Active Directory or, when creating a new VPC, you can create a new one. If you selected the **Create new Virtual Private Cloud (VPC)** option and want to enable Active Directory, you must select **Create a new Active Directory**.

Connecting to existing AWS Managed Active Directory or Self Managed Active Directory

From the dropdown list, select whether you want to use **AWS Managed Active Directory**, or **Self Managed Active Directory**. If you select **Self Managed Active Directory**, select the check box to verify that you have ensured a connection between the Active Directory and the VPC.

Follow the steps for granting permissions in the Active Directory.

- **Domain user name and password**. Enter the user name and password for your directory. For required permissions for the domain user, see Active Directory (Windows deployment). Launch Wizard stores the password in AWS Secrets Manager as a secure string parameter. It does not store the password on the service side. To create a functional SQL Server Developer Edition deployment, it reads from AWS Secrets Manager.
- **DNS address**. Enter the IP address of the DNS servers to which you are connecting. These servers must be reachable from within the VPC that you selected.
- **Optional DNS address**. If you would like to use a backup DNS server, enter the IP address of the DNS server that you want to use as backup. These servers must be reachable from within the VPC that you selected.

- **Domain DNS name**. Enter the Fully Qualified Domain Name (FQDN) of the forest root domain used for the Active Directory. When you choose to create a new Active Directory, Launch Wizard creates a domain admin user on your Active Directory.
- (Optional) **Organization Unit (OU) Path**. Specify the distinguished path name of the Organizational Unit (OU) within which you want to join for the Active Directory. For example: OU=org,DC=example,DC=com

Creating a new AWS Managed Active Directory through Launch Wizard

- **Domain user name and password**. The domain user name is preset to “admin.” Enter a password for your directory. Launch Wizard stores the password in AWS Secrets Manager as a secure string parameter. It does not store the password on the server side. To create a functional SQL Server Developer Edition deployment, it reads from AWS Secrets Manager.
- **Domain DNS name**. Enter a Fully Qualified Domain Name (FQDN) of the forest root domain used for the Active Directory. When you choose to create a new Active Directory, Launch Wizard creates a domain admin user on your Active Directory.
- (Optional) **Organization Unit (OU) Path**. Specify the distinguished path name of the Organizational Unit (OU) within which you want to join for the Active Directory. For example: OU=org,DC=example,DC=com

Connecting to a Self Managed Active Directory through Launch Wizard

Launch Wizard allows you to connect to a Self Managed Active Directory environment during deployment. For more information, see Self Managed Active Directory.

SQL Server configuration

When you use an existing Active Directory (if Active Directory is enabled), you have the option of using an existing SQL Server service account or creating a new account. If you create a new Active Directory account, you must create a new SQL Server account. If Active Directory is disabled, skip this section and SQL Server authentication will be configured.

- **User name and password**. If you are using an existing SQL Server service account, provide your user name and password. This SQL Server service account should be part of the Managed Active Directory in which you are deploying. If you are creating a new

SQL Server service account through Launch Wizard, enter a user name for the SQL Server service account. Create a complex password that is at least 8 characters long, and then reenter the password to verify it. See Password Policy for more information.

- **SQL Server install type**. Select the version of SQL Server that you want to deploy. You can select an AMI from either the License-included AMI or Custom AMI dropdown lists.
- **License-included AMI**. Choose an AMI for your SQL Server deployment which determines the version and edition of Windows Server that will be deployed. Available options include:
  - Windows Server 2022: Full-base AMI with Windows Server 2022
  - Windows Server 2025: Full-base AMI with Windows Server 2025
- (Optional) **tempdb configuration**. To improve performance, you can opt for the SQL Server tempdb system database to reside on a local NVMe SSD ephemeral storage device, also called the (instance store volume). NVMe SSD instance store volumes are available only on instance types that provide these local storage devices. Additionally, only data that changes frequently should ever reside on these volumes. They are not intended to store data long-term. For more information, see Amazon EC2 instance store.
- (Optional) **Additional SQL Server settings**. You can specify the following:
  - **Node name**. Enter a **SQL node name** for the single instance deployment.

5. When you are satisfied with your configuration selections, select **Next**. If you don't want to complete the configuration, select **Cancel**. When you select **Cancel**, all of the selections on the specification page are lost and you are returned to the landing page. To go to the previous screen, select **Previous**.
6. After configuring your application, you are prompted to define the infrastructure requirements for the new deployment on the **Configure infrastructure settings** page. The following tabs provide information about the input fields.

   Define SQL Server Developer Edition settings

   Configure the SQL Server Developer Edition software settings including version selection, license agreement, and installation media source. You must accept the Microsoft EULA and provide installation media either through S3 or a direct download URL.

   - **SQL Server Developer Edition Version —** Select the version of SQL Server Developer Edition you want to install:
     - SQL Server 2022 Developer Edition

- **Microsoft EULA Agreement —** You must review and agree to the license terms for the Microsoft software you are providing to AWS Launch Wizard. This is a mandatory requirement and the deployment cannot proceed without acceptance.
- **Installation method —** Choose how you want to provide the SQL Server Developer Edition installation media.
  - *Bring Your Own Media*
    - Provide a SQL Developer Edition installation media to be used for installing SQL Server Developer Edition.
    - Select the S3 bucket that starts with "launchwizard" (e.g., launchwizard-sql-media-bucket) containing your SQL Server installation files.
  - *Bring Your Own URL*

    Launch Wizard will download the SQL Server Developer Edition installation media from a customer provided public URL.

Define compute and storage requirements

You can choose to select your instances and volume types, or to use AWS recommended resources. If you choose to use AWS recommended resources, you have the option of defining your infrastructure needs. If no selections are made, default values are assigned.

- **Number of instance cores**. Choose the number of CPU cores for your infrastructure. The default value assigned is 4.
- **Network performance**. Choose your preferred network performance in Gbps.
- **Memory (GB)**. Choose the amount of RAM that you want to attach to your EC2 instance. The default value assigned is 4 GB.
- **Type of storage drive**. Select the storage drive type for the SQL data and tempdb volumes. If you chose to place your tempdb on local storage, only the SQL data will be on the storage drive you select. The default value assigned is SSD.
- **SQL Server throughput**. Select the sustained SQL Server throughput that you need.
- **Recommended resources**. Launch Wizard displays the system-recommended resources based on your infrastructure selections. If you want to change the recommended resources, select different infrastructure requirements.

Infrastructure requirements based on instance type

You can choose to select your instance and volume type, or to use AWS recommended resources. If no selections are made, default values are assigned.

- **Instance type**. Select your preferred instance type from the dropdown list.
- **Volume type**. Choose your preferred EBS volume type. For more information about volume types, see Amazon EBS volume types.

Drive letters and volume size

- **Drive letter**. Select the storage drive letter for **Root drive**, **Logs**, **Data**, and **Backup** volumes.

  > ⚠ **Important**
  >
  > For custom AMIs, Launch Wizard assumes the root volume drive is `C:`.

- **Volume size**. Select the size of the SQL Server data volume in Gb for **Root drive**, **Logs**, **Data**, and **Backup** volumes. SQL Server logs and data will be staged on the same data volume for this deployment. Make sure that you select an adequate size for the data volume.

> ⓘ **Note**
>
> For Launch Wizard deployments created after January 2023, IMDSv1 is disabled on all instances. If your software or scripts use IMDSv1, you will have to meet the requirements to use IMDSv2. For more information, see Use IMDSv2.

(Optional) Tags

You can provide optional custom tags for the resources Launch Wizard creates on your behalf. For example, you can set different tags for EC2 instances, EBS volumes, VPC, and subnets. If you select **All**, you can assign a common set of tags to your resources. Launch Wizard assigns tags with a fixed key `LaunchWizardResourceGroupID` and value that corresponds to the ID of the AWS resource group created for a deployment. Launch Wizard does not support custom tagging for root volumes.

Estimated on-demand cost to deploy additional resources

AWS Launch Wizard provides an estimate for application charges incurred to deploy the selected resources. The estimate updates each time you change a resource type in the Wizard. The provided estimates are only for general comparisons. They are based upon On-Demand costs and your actual costs may be lower.

7. When you are satisfied with your infrastructure selections, select **Next**. If you don't want to complete the configuration, select **Cancel**. When you select **Cancel**, all of the selections on the specification page are lost and you are returned to the landing page. To go to the previous screen, select **Previous**.
8. On the **Review and deploy** page, review your configuration details. If you want to make changes, select **Previous**. To stop, select **Cancel**. When you select **Cancel**, all of the selections on the specification page are lost and you are returned to the landing page. When you choose **Deploy**, you agree to the terms of the **Acknowledgment**.
9. Launch Wizard validates the inputs and notifies you of any issues you must address.
10. When validation is complete, Launch Wizard deploys your AWS resources and configures your SQL Server Developer Edition application. Launch Wizard provides you with status updates about the progress of the deployment on the **Deployments** page. From the **Deployments** page, you can view the list of current and previous deployments.
11. When your deployment is ready, a notification informs you that your SQL Server application is successfully deployed. If you have set up an SNS notification, you are also alerted through SNS. You can manage and access all of the resources related to your SQL Server Developer Edition application by selecting the deployment, and then selecting **Manage** from the **Actions** dropdown list.
12. When the SQL Server Developer Edition application is deployed, you can access your Amazon EC2 instance through the EC2 console. You can also use AWS SSM to manage your SQL Server Developer Edition application for future updates and patches with built-in integration through resource groups.

Deploy an application with AWS Launch Wizard for SQL Server on Ubuntu (Console)

Topics

Access AWS Launch Wizard

You can launch AWS Launch Wizard from the AWS Launch Wizard console.

Deploy AWS Launch Wizard on Ubuntu

The following steps guide you through a SQL Server application deployment with AWS Launch Wizard on the Ubuntu platform after you have launched it from the console. For SQL Server deployments on Ubuntu, you must use an instance type built on the Nitro System. EBS volumes are exposed as NVMe block devices on instances built with the Nitro System. Device names that are specified for NVMe EBS volumes in a block device mapping are renamed using NVMe device names (`/dev/nvme[[0-26]n1`). Launch Wizard deployments on Ubuntu do not support block devices on Xen-virtualized instances.

1. When you select **Choose application** from the AWS Launch Wizard landing page, you are directed to the **Choose application** wizard, where you are prompted to select the type of application that you want to deploy. Select **Microsoft SQL Server**, then **Create deployment**.
2. Under **Review Permissions**, Launch Wizard displays the AWS Identity and Access Management (IAM) role required for Launch Wizard to access other AWS services on your behalf. For more information about setting up IAM for Launch Wizard, see AWS Identity and Access Management (IAM). Choose **Next** .
3. On the **Configure application settings** page, select the **Operating System** on which you want to install SQL Server — in this case, **Ubuntu**.
4. **Deployment model**. Choose **High availability deployment** to deploy your SQL Server Always On application across multiple Availability Zones or **Single instance deployment** to deploy your SQL Server application on a single node.

5. You are prompted to enter specifications for the new deployment. The following tabs provide information about the input fields.

   General

   - **Deployment name**. Enter a unique application name for your deployment.
   - **Simple Notification Service (SNS) topic ARN (Optional)**. Specify an SNS topic where AWS Launch Wizard can send notifications and alerts. For more information, see the *Amazon Simple Notification Service Developer Guide*.
   - **Enable rollback on failed deployment**. By default, if a deployment fails, your provisioned resources will not be rolled back/deleted. This default configuration helps you to troubleshoot errors at the resource level as you debug deployment issues. If you want your provisioned resources to be immediately deleted if a deployment fails, select the check box.

   Connectivity

   Enter your requirements for how you want to connect to your application instance and what kind of Virtual Private Cloud (VPC) you want to set up.

   Key pair name

   - Select an existing key pair from the dropdown list or create a new one. If you select **Create new key pair name** to create a new key pair, you are directed to the Amazon EC2 console. From there, under **Network and Security**, choose **Key Pairs**. Choose **Create a new key pair**, enter a name for the key pair, and then choose **Download Key Pair**.

     > ⚠ **Important**
     >
     > This is your only opportunity to save the private key file. Download it and save it in a safe place. You must provide the name of your key pair when you launch an instance, and provide the corresponding private key each time that you connect to the instance.

     Return to the Launch Wizard console and choose the refresh button next to the **Key Pairs** dropdown list. The newly created key pair appears in the dropdown list. For more information about key pairs, see Amazon EC2 Key Pairs and Windows Instances.

Virtual Private Cloud (VPC). Choose whether you want to use an existing VPC or create a new VPC.

- **Select Virtual Private Cloud (VPC)** option. Choose the VPC that you want to use from the dropdown list. Your VPC must contain one public subnet. For HA deployments, it must also contain, at least, three private subnets. For single node deployments, it must contain one private subnet. The private subnets must have outbound connectivity to the internet and other AWS services (S3, CFN, SSM, Logs). We recommend that you enable this connectivity with a NAT Gateway. For more information about NAT Gateways, see NAT Gateways in the Amazon VPC User Guide.
  - **Public Subnet**. Your VPC must contain one public subnet. For HA deployments it must also contain three private subnets. For single node deployments, it must contain one private subnet. Choose a public subnet for your VPC from the dropdown list. To continue, you must select the check box that indicates that the public subnet has been set up and each of the selected private subnets have outbound connectivity enabled.

    To add a new public subnet

    If the traffic of a subnet is routed to an internet gateway, the subnet is known as a public subnet. If, however, a subnet doesn't have a route to the internet gateway, the subnet is known as a private subnet. To use an existing VPC that does not have a public subnet, you can add a new public subnet using the following steps.

    - Follow the steps in Creating a Subnet in the Amazon VPC User Guide using the existing VPC you intend to use AWS Launch Wizard.
    - To add an internet gateway to your VPC, follow the steps in Attaching an Internet Gateway in the Amazon VPC User Guide.
    - To configure your subnets to route internet traffic through the internet gateway, follow the steps in Creating a Custom Route Table in the Amazon VPC User Guide. Use IPv4 format (`0.0.0.0/0`) for **Destination**.
    - The public subnet should have the “auto-assign public IPv4 address” setting enabled. To enable this setting, follow the steps in Modifying the Public IPv4 Addressing Attribute for Your Subnet in the Amazon VPC User Guide.
  - **Availability Zone (AZ) configuration**. You must choose at least three Availability Zones for High Availability (HA) deployments and one Availability Zone for single-node deployments, with one private subnet for each Availability Zone that you select. From

the dropdown lists, select the **Availability Zones** within which you want to deploy your **primary**, **secondary**, and **configuration** nodes.

To create a private subnet

If a subnet doesn't have a route to an internet gateway, the subnet is known as a private subnet. To create a private subnet, perform the following steps. We recommend that you enable the outbound connectivity for each of your selected private subnets using a NAT Gateway. To enable outbound connectivity from private subnets to public subnets, see the steps in Creating a NAT Gateway to create a NAT Gateway in your chosen public subnet. Then, follow the steps in Updating Your Route Table for each of your chosen private subnets.

- Follow the steps in Creating a Subnet in the Amazon VPC User Guide using the existing VPC you will use in AWS Launch Wizard.
- When you create a VPC, it includes a main route table by default. On the **Route Tables** page in the Amazon VPC console, you can view the main route table for a VPC by looking for **Yes** in the **Main** column. The main route table controls the routing for all subnets that are not explicitly associated with any other route table. If the main route table for your VPC has an outbound route to an internet gateway, then any subnet created using the previous step, by default, becomes a public subnet. To ensure the subnets are private, you may need to create one separate route table for all of your private subnets. This route table must not contain any routes to an internet gateway. Verify that all of the private subnets have the same route table association.

- **Create new Virtual Private Cloud (VPC)** option. Launch Wizard creates your VPC. You can optionally enter a VPC name tag.

SQL Server

SQL Server configuration

- **User name and password**. By default, Launch Wizard applies the user name `sa` . This system administrator account is used for SQL Server management. Create a complex password that is at least 8 characters long, and then reenter the password to verify it. See Password Policy for more information.

- **Floating IP Address (HA and existing VPC deployments only)**. This field is available when you select a Virtual Private Cloud (VPC). The IP address that you enter is used as the endpoint for your SQL Server Availability Group listener. Launch Wizard creates a route from this IP address to the SQL primary node in your route table. Verify that the IP address is not already in use within your VPC and is outside of all of the provided subnet CIDRs.
- **Amazon Machine Image (AMI)**. Select the version of Microsoft SQL Server Enterprise to deploy. You can select an AMI from the lists of either license-included or custom AMIs.

Pacemaker cluster configuration (HA deployments only)

Pacemaker is a high-availability cluster resource manager. This software runs on a set of hosts, or cluster of nodes, to preserve integrity and minimize the downtime of selected services or resources. Pacemaker is maintained by the ClusterLabs community.

- **Pacemaker cluster name**. Enter a name to identify your pacemaker cluster.
- **Pacemaker cluster username**. By default, Launch Wizard applies the pacemaker username `hacluster`. This username is used to securely communicate between cluster nodes.
- **Pacemaker cluster password**. Create a complex password that is at least 8 characters long, and then reenter the password to verify it. See Password Policy for more information.

SQL - Pacemaker cluster connection settings (HA deployments only)

After you configure Pacemaker cluster and SQL Server, you must create a user in SQL Server to communicate with Pacemaker.

- **SQL Pacemaker user name and password**. Enter a user name for SQL Server to communication with the Pacemaker cluster. Create a complex password that is at least 8 characters long, and then reenter the password to verify it. See Password Policy for more information.
- **S3 location for node certificates**. An Amazon S3 bucket location is required by the SQL nodes to share self-signed certificates with each other. Provide the bucket or object locations and verify that the names begin with `launchwizard-`.

Additional SQL Server settings (optional)

- **Nodes**. Enter a **Primary SQL node name**, a **Secondary SQL node name**, and a **Configuration node name**.
- **Additional naming**. Enter a **Database name** and an **Availability group name**.

6. When you are satisfied with your configuration selections, select **Next**. If you don't want to complete the configuration, select **Cancel**. When you select **Cancel**, all of the selections on the specification page are lost and you are returned to the landing page. To go to the previous screen, select **Previous**.
7. After configuring your application, you are prompted to define the infrastructure requirements for the new deployment on the **Define infrastructure requirements** page. The following tabs provide information about the input fields.

Define infrastructure requirements

You can choose to select your instances, storage and performance, and volume types, or to use AWS recommended resources. If you choose to use AWS recommended resources, you have the option of defining your high availability cluster needs. If no selections are made, default values are assigned.

- **Number of instance cores**. Choose the number of CPU cores for your infrastructure. The default value assigned is 4.
- **Network performance**. Choose your preferred network performance in Gbps.
- **Memory (GB)**. Choose the amount of RAM that you want to attach to your EC2 instances. The default value assigned is 4 GB.
- **Type of storage drive**. Select the storage drive type for the SQL data and tempdb volumes. The default value assigned is SSD.
- **SQL Server throughput**. Select the sustained SQL Server throughput that you need.
- **Recommended resources**. Launch Wizard displays the system-recommended resources based on your infrastructure selections. If you want to change the recommended resources, select different infrastructure requirements.

Infrastructure requirements based on instance type

You can choose to select your instance and volume type, or to use AWS recommended resources. If no selections are made, default values are assigned.

- **Instance type**. Select your preferred instance type from the dropdown list.
- **Volume type**. Choose your preferred EBS volume type. For more information about volume types, see Amazon EBS volume types

Volume sizes

- **Volume size**. Select the size of the SQL Server data volume in Gb for **Temporary database**, **Logs**, **Data**, and **Backup** volumes. SQL Server logs and data will be staged on the same data volume for this deployment. Make sure that you select an adequate size for the data volume.

> **Note**
>
> For Launch Wizard deployments created after January 2023, IMDSv1 is disabled on all instances. If your software or scripts use IMDSv1, you will have to meet the requirements to use IMDSv2. For more information, see Use IMDSv2.

Tags-Optional

You can provide optional custom tags for the resources Launch Wizard creates on your behalf. For example, you can set different tags for EC2 instances, EBS volumes, VPC, and subnets. If you select **All**, you can assign a common set of tags to your resources. Launch Wizard assigns tags with a fixed key `LaunchWizardResourceGroupID` and value that corresponds to the ID of the AWS resource group created for a deployment. Launch Wizard does not support custom tagging for root volumes.

Estimated on-demand cost to deploy additional resources

AWS Launch Wizard provides an estimate for application charges incurred to deploy the selected resources. The estimate updates each time you change a resource type in the wizard. The provided estimates are for general comparisons only. They are based upon On-Demand costs and your actual costs may be lower.

8. When you are satisfied with your infrastructure selections, select **Next**. If you don't want to complete the configuration, select **Cancel**. When you select **Cancel**, all of the selections on the specification page are lost and you are returned to the landing page. To go to the previous screen, select **Previous**.
9. On the **Review and deploy** page, review your configuration details. If you want to make changes, select **Previous**. To stop, select **Cancel**. When you select **Cancel**, all of the selections on the specification page are lost and you are returned to the service landing page. When you choose **Deploy**, you agree to the terms of the **Note** at the bottom of the page.
10. Launch Wizard validates the inputs and notifies you if you must update a specification.
11. When validation is complete, Launch Wizard deploys your AWS resources and configures your SQL Server Always On application. Launch Wizard provides you with status updates about the progress of the deployment on the **Deployments** page. From the **Deployments** page, you can view the list of current and previous deployments.
12. When your deployment is ready, a notification informs you that your SQL Server application is successfully deployed. If you have set up an SNS notification, you are also alerted through SNS. You can manage and access all of the resources related to your SQL Server Always On application by selecting the deployment, and then selecting **Manage** from the **Actions** dropdown list.
13. When the SQL Server Always On application is deployed, you can access your Amazon EC2 instances through the EC2 console. You can also use AWS SSM to manage your SQL Server Always On application for future updates and patches through built-in integration via resource groups.

Post-deployment cluster tasks

The Launch Wizard Pacemaker implementation includes three cluster nodes: primary, secondary, and configuration only. The primary node provides the Microsoft SQL Server for Ubuntu resource and the floating IP address. To ensure that the cluster operates correctly, some administrative tasks must be performed in a specific way. If these tasks are performed incorrectly, then Pacemaker may identify the activity as a resource failure and attempt to fail over the resources to the secondary node. If the resources are failed over to the secondary node, the cluster can remain in an unknown state, which can impact user access.

There are four primary tasks: **Start Cluster**, **Stop Cluster**, **Move Resources**, and **Recovery**. These tasks must be carried out by a sudo user with an SSH connection to any of the cluster nodes. Before performing any of these tasks, verify the cluster status using `pcs resource status --all`. This command returns all cluster issues. All issues must be addressed prior to performing any administrative tasks.

Start cluster

1. Log in to a cluster node using a sudo user over an SSH connection.
2. Verify that all cluster nodes are available.
3. Verify cluster status using the following command: `pcs resource --all`.

   Address all issues before attempting to start the cluster.
4. Start all cluster nodes using the following command: `pcs cluster start --all --wait`.
5. Verify that the cluster has started using the following command: `pcs resource --all`.

   The output provides information about the cluster nodes and cluster resources. All cluster nodes should be online and all resource agents should be visible and allocated to their assigned cluster nodes.
6. Verify that the availability group listener is available by pinging the floating IP address.

Manually move cluster resources

1. Log in to a cluster node using a sudo server over an SSH connection.
2. Verify that all cluster nodes are available.
3. Verify cluster status using the following command: `pcs resource --all`.

   Address all issues before attempting to start the cluster.
4. Run the following command: `pcs resource move` **`<RESOURCE_NAME>`**`-master` **`<NODE_NAME>`** `--force`.

   This command moves the resource agent to **`<NODE_NAME>`** and starts the resource. All cluster constraints will be applied. If the Microsoft SQL Server resource agent is moved, then the availability group listener will follow.
5. Verify cluster status using the following command: `pcs resource --all`.

   The resource that was moved should be located on the **`<NODE_NAME>`**.

6. Clear temporary constraints using the following command: `pcs resource clear` **`<RESOURCE_NAME>`**.

Stop cluster

1. Log in to a cluster node using a sudo server over an SSH connection.
2. Verify that all cluster nodes are available.
3. Verify cluster status using the following command: `pcs resource --all`.

   Address all issues before attempting to start the cluster.
4. Stop the cluster using the following command: `pcs cluster stop --ALL`. This will gracefully shut down all of the cluster nodes.
5. Verify the shut down status using the following command: `pcs status --all`.

   This command should return that the cluster is no longer running.

Recovery

If a node is restarted from the operating system or the AWS Management Console, the Pacemaker node and its related services will not automatically start. This prevention protects the high availability database replicas from split-brain corruption.

The following steps are required to restore the cluster to normal operations.

1. Log in to a cluster node using a sudo server over an SSH connection.
2. Determine the node that was restarted using the following command: `pcs resource --ALL`. The restarted node will be offline.
3. Verify cluster status using the following command: `pcs resource --all`.

   Address all issues before attempting to start the cluster.
4. Start the restarted node using the following command: `pcs cluster start --`**`<NODE_NAME>`**.
5. Verify cluster status using the following command: `pcs resource --all`.

   Address all issues before attempting to start the cluster.
6. If the restarted node is the primary node of the cluster, then the Availability Group resource must be returned to the primary node.

7. Remove all temporary constraints using the following commands: `pcs resource clear` **`<AG_RESOURCE>`** and `pcs resource clear` **`<AG_LISTENER>`**.
8. Run the following command: `pcs resource move` **`<RESOURCE_NAME> <PRI_NODE_NAME>`** `--force`.

   This command moves the resources to **`<PRI_NO_NAME>`** and starts the resource. Any cluster constraints are applied. In this scenario, if the Microsoft SQL Server resource agent is moved, then the availability group listener follows.
9. Verify cluster status using the following command: `pcs resource --all`. The restarted node will be located on **`<PRI_NO_NAME>`**.

Deploy SQL Server to a new or existing VPC (AWS CLI)

You can use the AWS Launch Wizard CreateDeployment API operation to deploy SQL Server. To create a deployment, you must provide values for various *specifications*. Specifications are a collection of settings that define how your deployment should be created and configured. A workload will have one or more deployment patterns with differing required and optional specifications.

If you want to use the **Clone deployment** action on your deployment, you must create your deployment using the Launch Wizard console.

Prerequisites for deploying SQL Server with the AWS CLI

Before deploying SQL Server with the AWS CLI, ensure you have met the following prerequisites:

- Install and configure the AWS CLI. For more information, see Install or update to the latest version of the AWS CLI.
- Complete the steps in the previous section titled **Set up**. Some deployment patterns have requirements that must be met for a deployment to be successful.

Create a SQL Server deployment with the AWS CLI

You can create a deployment for your SQL Server application using the `CreateDeployment` Launch Wizard API operation.

To create a deployment for SQL Server using the AWS CLI

1. List the available workload names using the ListWorkloads Launch Wizard API operation.

   The following example shows listing the available workloads:

   ```
   aws launchwizard list-workloads --region us-east-1
   {
       "workloads": [
           {
               "displayName": "Remote Desktop Gateway",
               "workloadName": "RDGW"
           },
           {
               "displayName": "MS SQL Server",
               "workloadName": "SQL"
           },
           {
               "displayName": "SAP",
               "workloadName": "SAP"
           },
           {
               "displayName": "Microsoft Active Directory",
               "workloadName": "MicrosoftActiveDirectory"
           }
           ...
       ]
   }
   ```

2. Specify the desired workload name with the ListWorkloadDeploymentPatterns operation to describe the supported values for the deployment pattern names.

   The following example lists the available workload patterns for a given workload:

   ```
   aws launch-wizard list-workload-deployment-patterns --workload-name SQL --
   region us-east-1
   {
       "workloadDeploymentPatterns": [
           {
               "deploymentPatternName": "SQLHAAlwaysOn",
               "description": "Example description.",
               "displayName": "ExampleDisplayName",
               "status": "ACTIVE",
   ```

```
            "workloadName": "SQL",
            "workloadVersionName": "2024-05-03-00-00-00"
        },
        ...
    ]
}
```

3. Use the workload and deployment pattern names you discovered with the GetWorkloadDeploymentPattern operation to list the specification details.

   The following example lists the workload specifications of a given workload and deployment pattern:

```
aws launchwizard get-workload-deployment-pattern --workload-name SQL --deployment-pattern-name SQLHAAlwaysOn --region us-east-1
{
    "workloadDeploymentPattern": {
        "deploymentPatternName": "SQLHAAlwaysOn",
        "description": "Example description.",
        "displayName": "ExampleDisplayName",
        "specifications": [
            {
                "description": "Enter an SNS topic for AWS Launch Wizard to send
 notifications and alerts.",
                "name": "AWS:LaunchWizard:TopicArn",
                "required": "No"
            },
            {
                "description": "When a deployment fails, your provisioned resources
 will be deleted/rolled back by default. If deactivated, the provisioned resources
 will be deleted when you delete your deployment from the Launch Wizard console.",
                "name": "AWS:LaunchWizard:DisableRollbackFlag",
                "required": "No"
            },
            {
                "allowedValues": [
                    "true",
                    "false"
                ],
                "description": "Cloud Watch Application Insights monitoring",
                "name": "SetupAppInsightsMonitoring",
                "required": "Yes"
            },
            ...
        ]
    }
}
```

4. With the workload specifications retrieved, you must provide values for any specification `name` with a `required` value of `Yes`. You can also provide any optional specifications you require for your deployment. We recommend that you pass inputs to the `specifications` parameter for your deployment as a file for easier usage.

   Your JSON file's format should resemble the following:

   ```
   {
     "ExampleName1": "ExampleValue1",
     "ExampleName2": "ExampleValue2",
     "ExampleName3": "ExampleValue3"
   }
   ```

5. With the specifications file created, you can create a deployment for your chosen workload and deployment pattern.

   The following example creates a deployment with specifications defined in a file:

   ```
   aws launch-wizard create-deployment --workload-name SQL --deployment-pattern-name SQLHAAlwaysOn --name ExampleDeploymentName --region us-east-1 --specifications file://specifications.json
   ```

Manage application resources with AWS Launch Wizard for SQL Server

After your SQL Server Always On application is deployed, you can manage it by following these steps.

1. From the navigation pane, under **Deployments**, choose **MS SQL Server**.
2. From the **Deployments — SQL** page, select the deployment you want to manage and then select **Actions**. You can select to do the following:
   1. **Manage resources on the EC2 console**. You are taken to the Amazon EC2 console, where you can view and manage your SQL Server Always On application resources. For example,

you can view and manage EC2, Amazon EBS, Active Directory, Amazon VPC, Subnets, NAT Gateways, and Elastic IPs. For SQL Server on Linux deployments, you can use AWS Systems Manager Session Manager to manage your deployed EC2 instances. For more information about SSM Session Manager, see AWS Systems Manager Session Manager.

2. **Access SQL Server using RDGW instance (Windows deployments)**. Connect to SQL Server via Remote Desktop Protocol. For more information, see Connecting to your Windows Instance in the *User Guide for Windows Instances*.
3. **View resource group with SSM**. You are taken to the Systems Manager console to view your resource groups.
4. **View SSM deployment template (Windows deployments)**. You are taken to the Systems Manager console to view your documents.
5. **View CloudWatch application logs**. You are taken to CloudWatch Logs, where you can monitor, store, and access your SQL Server Always On application log files.
6. **View your CloudFormation template**. This is the CloudFormation template created by your most recent deployment, and it can be accessed through the CloudFormation console. For help with finding and using your CloudFormation template, see Viewing CloudFormation Stack Data and Resources on the AWS Management Console.
7. If you have not set up monitoring for your application on CloudWatch Application Insights, you have the option to **Set up monitoring on CloudWatch Application Insights**. You are taken to the CloudWatch Application Insights console to set up monitoring for your application.

   If you have set up monitoring for your application on CloudWatch Application Insights, you can **View insights on Amazon CloudWatch**. You are taken to the application monitoring dashboards on the CloudWatch Application Insights console.

3. To delete a deployment, select the application that you want to delete and select **Delete**. You are prompted to confirm your action.

> ⚠ **Important**
>
> You lose all specification settings for the SQL Server Always On application when you delete a deployment. Launch Wizard attempts to delete only the AWS resources that it created in your account as part of the deployment. If you created resources outside of Launch Wizard, for example, resources that reside in a VPC created by Launch Wizard, the deletion may fail. Launch Wizard does not delete any Active Directory objects in your Active Directory, nor any of the records in your DNS server. Launch Wizard has no

> control over your Active Directory domain user password over time, which is required to clean up Active Directory objects or DNS records. We recommend that you remove these entries from your Active Directory after Launch Wizard deletes the deployment. For key operations performed against your Active Directory resulting in new records or entries, see AWS Managed Active Directory.

4. To drill down into details regarding your SQL Server Always On application resources, select the **Application name**. You can then view the **Deployment events** and **Configuration summary** details for your application by using the tabs at the top of the page.

Manage Launch Wizard application resources with AWS Systems Manager Application Manager

AWS Systems Manager Application Manager, a capability of AWS Systems Manager, helps you to investigate and remediate issues with your AWS resources that make up an application. Application Manager aggregates operations information from multiple AWS services and Systems Manager capabilities to a single console.

Application Manager automatically imports application resources created by Launch Wizard. From the Application Manager console, you can view operations details and perform operations tasks. You can also use runbooks, or SSM Automation documents, provided by Launch Wizard from the Application Manager console to manage or remediate issues with application components or resources.

For general information about AWS Systems Manager Application Manager, see AWS SSM Application Manager in the AWS Systems Manager User Guide.

The following information is specific to the management of Launch Wizard application resources from the Application Manager console.

Topics

Use SSM Application Manager to run Automation workflows on your Launch Wizard applications

You can perform operations tasks and remediate issues with your Launch Wizard application resources by using AWS Systems Manager Automation runbooks.

Application Manager automatically imports all of your Launch Wizard resources and lists them in the Launch Wizard category. From the Application Manager console, choose **Launch Wizard** from the list of **Applications**. Select an application to view its information. On the **Application information** page, choose **Start runbook**. A dropdown list displays all of the runbooks available for your Launch Wizard application. This list includes runbooks provided by AWS, as well as any custom runbooks you own or are shared with you.

When you select a runbook, you are taken to the SSM Automation document console, where the resource group that makes up your application is preselected.

For descriptions of the runbooks provided by **Launch Wizard**, see AWS Launch Wizard Systems Manager Automation documents.

Add custom runbooks

To add your own runbooks, you must modify the service setting value for the supported type.

1. The service setting value is a list of document Amazon Resource Names (ARNs). You can view this list using the following AWS Command Line Interface (AWS CLI) command, and adding the type to the `setting id` path.

   There are four supported types for which there are service settings:

   - `AWS-SQLServerWindows`
   - `AWS-SQLServerLinux`
   - `AWS-SAP`
   - `AWS-SelfManagedActiveDirectory`

   The following command lists the service settings for `AWS-SQLServerWindows`.

   ```
   aws ssm get-service-setting --setting-id /launchwizard/AWS-SQLServerWindows
   ```

The following is the example output.

```
{
    "ServiceSetting": {
        "SettingId": "/launchwizard/AWS-SQLServerWindows",
        "SettingValue": "arn:aws:ssm:us-east-1::document/AWSSQLServer-
Backup,arn:aws:ssm:us-east-1::document/AWSSQLServer-Restore,arn:aws:ssm:us-
east-1::document/AWSSQLServer-Index,arn:aws:ssm:us-east-1::document/AWSSQLServer-
DBCC",
        "LastModifiedDate": "2020-11-13T13:36:09.527000-05:00",
        "LastModifiedUser": "System",
        "ARN": "arn:aws:ssm:us-east-1:012345678901:servicesetting/launchwizard/AWS-
SQLServerWindows",
        "Status": "Default"
    }
}
```

2. You can modify the list of document ARNs by running the following command.

```
aws ssm update-service-setting \
    --setting-id /launchwizard/AWS-SQLServerWindows \
    --setting-value \
    "arn:aws:ssm:us-east-1::document/AWSSQLServer-Backup,arn:aws:ssm:us-
east-1::document/AWSSQLServer-Restore,arn:aws:ssm:us-east-1::document/AWSSQLServer-
Index,arn:aws:ssm:us-east-1::document/Document"
```

3. To reset the service setting value, run the following AWS CLI command. This command resets the service setting value for `AWS-SQLServerWindows`.

```
aws ssm reset-service-setting --setting-id /launchwizard/AWS-SQLServerWindows
```

The following is the example output.

```
{
    "ServiceSetting": {
        "SettingId": "/launchwizard/AWS-SQLServerWindows",
        "SettingValue": "arn:aws:ssm:us-east-1::document/AWSSQLServer-
Backup,arn:aws:ssm:us-east-1::document/AWSSQLServer-Restore,arn:aws:ssm:us-
east-1::document/AWSSQLServer-Index,arn:aws:ssm:us-east-1::document/AWSSQLServer-
DBCC",
        "LastModifiedDate": "2020-11-13T13:36:09.527000-05:00",
        "LastModifiedUser": "System",
        "ARN": "arn:aws:ssm:us-east-1:012345678901:servicesetting/launchwizard/AWS-
SQLServerWindows",
        "Status": "Default"
    }
}
```

The document lists correspond to the application type level. Therefore, when you add a new `AWS-SQLServerWindows` document, it will show up in all `AWS-SQLServerWindows` deployments. You can't add documents to a specific application.

Note

Verify that you use the correct Region for the added document ARNs.

Onboard existing applications

When you deploy an application with Launch Wizard, the resource groups that make up the application are automatically assigned metadata showing that they are provisioned by Launch Wizard. Application Manager uses this metadata to display all of your resource groups and CloudFormation stacks created by Launch Wizard on one page. When you deploy an application, Launch Wizard calls the `CreateOpsMetadata` API to assign the provisioning metadata.

Onboard existing applications

You can manually call the `CreateOpsMetadata` API using the AWS CLI so that existing application deployments appear on the Application Manager Launch Wizard page. The following example shows the `create-ops-metadata` AWS CLI command.

```
aws ssm create-ops-metadata \
    --resource-id "arn:aws:resource-groups:us-east-1:123456789012:group/LaunchWizard-
SQLHAAlwaysOn-test" \
    --metadata '{"application-type": {"Value": "AWS-SQLServerWindows"}, "provisioned-
by": {"Value": "AWS-LaunchWizard"}}'
```

You must provide the following information:

- The resource group ARN of the resource that you want to be visible on the Launch Wizard page in Application Manager.

- A metadata JSON file that contains the `application-type` and `provisioned-by` key values. The `application-type` is the application type of the deployment, for example `AWS-SQLServerWindows` or `AWS-SAP`. The `provisioned-by` value is `AWS-LaunchWizard`.

When the command is successful, the output will be an `OpsMetadataArn`. If the output is an `OpsMetadataAlreadyExistsException`, then the resource group has already been tagged.

View all `OpsMetadata` values

You can call the `ListOpsMetadata` API to view all of your `OpsMetadata` values. To display only Launch Wizard-related metadata objects, you can use filtering. The following example shows the `list-ops-metadata` AWS CLI command.

```
aws ssm list-ops-metadata \
    --filters '[{"Key":"provisioned-by","Values":["AWS-LaunchWizard"]}]' \
    --max-results 20
```

The following is the example output.

```
{
    "OpsMetadataList": [
        {
            "ResourceId": "arn:aws:resource-groups:us-east-1:123456789012:group/
LaunchWizard-SQLHAAlwaysOn-test",
            "OpsMetadataArn": "arn:aws:ssm:us-east-1:123456789012:opsmetadata/aws/ssm/
LaunchWizard-SQLHAAlwaysOn-test/appmanager",
            "LastModifiedDate": "2020-11-16T22:41:43.035000-05:00",
            "LastModifiedUser": "arn:aws:sts::123456789012:assumed-role/Admin",
            "CreationDate": "2020-11-16T22:41:43.035000-05:00"
        }
    ]
}
```

Filter by application type

The following example shows the `list-ops-metadata` AWS CLI command to filter by application type:

```
aws ssm list-ops-metadata \
    --filters '[{"Key":"application-type","Values":["AWS-SQLServerWindows","AWS-
SAP"]}]' \
    --max-results 20
```

To get information about an OpsMetadataArn object, use the following command and enter the OpsMetadataArn.

```
aws ssm get-ops-metadata \
    --ops-metadata-arn "arn:aws:ssm:us-east-1:123456789012:opsmetadata/aws/ssm/
LaunchWizard-SQLHAAlwaysOn-test/appmanager"
```

The following is the example output.

```
{
    "ResourceId": "arn:aws:resource-groups:us-east-1:123456789012:group/LaunchWizard-
SQLHAAlwaysOn-test",
    "Metadata": {
        "application-type": {
            "Value": "AWS-SQLServerWindows"
        },
        "provisioned-by": {
            "Value": "AWS-LaunchWizard"
        }
    }
}
```

Delete metadata object

You can delete the metadata object if you make a mistake when using the create-ops-metadata AWS CLI command. Run the following command, entering the OpsMetadataArn, and then run the create-ops-metadata command again.

```
aws ssm delete-ops-metadata \
    --ops-metadata-arn "arn:aws:ssm:us-east-1:123456789012:opsmetadata/aws/ssm/
LaunchWizard-SQLHAAlwaysOn-test/appmanager"
```

For more information about CreateOpsMetadata and related APIs, see the Amazon EC2 Systems Manager API Reference.

Patch management

You can automate the process of patching your Launch Wizard instances with security and other types of updates. From the **Application information** page of the Application Manager console, choose **Patch**. You are taken to the SSM Patch Manager console **Patch now** page, where patch management options for your application instances are preselected.

For more information about how Patch Manager determines which patches to install and how it installs them, see How Patch Manager operations work.

AWS Launch Wizard Systems Manager Automation documents

A Systems Manager Automation document defines the actions that Systems Manager performs on your managed instances and other AWS resources when an automation workflow runs. A document contains one or more steps that run in sequential order.

Launch Wizard provides predefined Automation documents that are maintained by AWS. This topic describes each of the predefined Automation documents provided for AWS Launch Wizard.

For more information about SSM Automation documents, see AWS SSM Automation in the *AWS Systems Manager User Guide*.

Launch Wizard-provided Automation documents:

- AWSSQLServer-DBCC
- AWSSQLServer-Backup
- AWSSQLServer-Index
- AWSSQLServer-Restore

AWSSQLServer-DBCC

The `AWSSQLServer-DBCC` Automation document includes the steps to perform database integrity checks on a specified database. You can control the type of database checks that are run. You can also adjust the execution parameters, such as specific tables to check, maximum CPU utilization, and more. For more information about the operations performed by DBCC checks, see the SQL Server documentation.

AWSSQLServer-Backup

The `AWSSQLServer-Backup` Automation document includes the steps to back up a specified database in either full, differential, or transactional mode. After the backup is completed, you can upload it to a specified folder within an S3 bucket.

The backup modes are defined as follows:

- **Full** — a complete backup of the database.
- **Differential** — the delta of changes since the last full backup.
- **Transactional** — a log of changes from the last full or differential backup, depending on the last backup type taken.

To help ensure that the `AWSSQLServer-Backup` document can successfully back up a database that resides on resources provisioned with Launch Wizard, make sure the following is in place:

- SQL Server was provisioned on a single node or with Always On availability groups (AG).
- The `@@SERVERNAME` property in SQL Server matches the hostname of the operating system where the automation runs.
- The backup file is staged to a local disk.
- The size of the backup file for uploading to an S3 bucket is 500 GB or less.

Required IAM actions that must be added to your IAM policy to successfully run `AWSSQLServer-Backup`:

- `s3:GetBucketPolicyStatus`
- `s3:PutObject`

For more information about backup modes, see the Microsoft documentation.

AWSSQLServer-Index

The `AWSSQLServer-Index` Automation document includes steps to perform index maintenance operations on a specified database. You can choose a configuration, which includes the specific actions to take based on the level of fragmentation.

For more information about index maintenance operations, see the Microsoft documentation.

AWSSQLServer-Restore

The `AWSSQLServer-Restore` Automation document includes steps to download a backup database from a specified S3 bucket and folder to local storage. You can also optionally restore the backup to a copy of the database. The default behavior is to use the latest backup, and you can specify a time range to perform a point-in-time restore. The following conditions must be met for the `AWSSQLServer-Restore` document to successfully restore a database:

- The backup to use must have been performed by the `AWSSQLServer-Backup` document.
- There must be at least one full backup that occurred during the specified time range.

Required IAM actions that must be added to your IAM policy to successfully run `AWSSQLServer-Restore`:

- `s3:GetBucketPolicyStatus`
- `s3:PutObject`

Monitoring SQL Server Always On deployments

You can monitor your SQL Server Always On deployments using Amazon CloudWatch Application Insights. When you select the option to monitor your deployment using the Launch Wizard console, Application Insights identifies and sets up key metrics, logs, and alarms across your application resources and technology stack for your Microsoft SQL Server database. Anomalies and errors are detected and correlated as Application Insights continuously monitors metrics and logs. When errors and anomalies are detected, Application Insights generates CloudWatch Events that you can use to set up notifications or take action. To help with troubleshooting, Application Insights creates automated dashboards for detected problems, which include correlated metric anomalies and log errors, along with additional insights to point you to a possible root cause. Use the automated dashboards to take remedial actions to keep your applications healthy and prevent end-user impact. You can also resolve problems with AWS SSM OpsCenter using generated OpsItems.

For Microsoft SQL Server High Availability (HA) workloads, you can use CloudWatch Application Insights to configure important counters, such as Mirrored Write Transaction/sec, Recovery Queue Length, Transaction Delay, and Windows Event Logs on CloudWatch. You can also get automated insights whenever a failover event or problem, such as restricted access to query a target database, is detected with SQL HA workloads. See the Amazon CloudWatch Application Insights documentation for a complete list of Logs and metrics supported by Application Insights.

High availability and security best practices for AWS Launch Wizard for SQL Server

The application architecture created by AWS Launch Wizard supports AWS best practices for high availability and security as promoted by the AWS Well-Architected Framework.

Topics

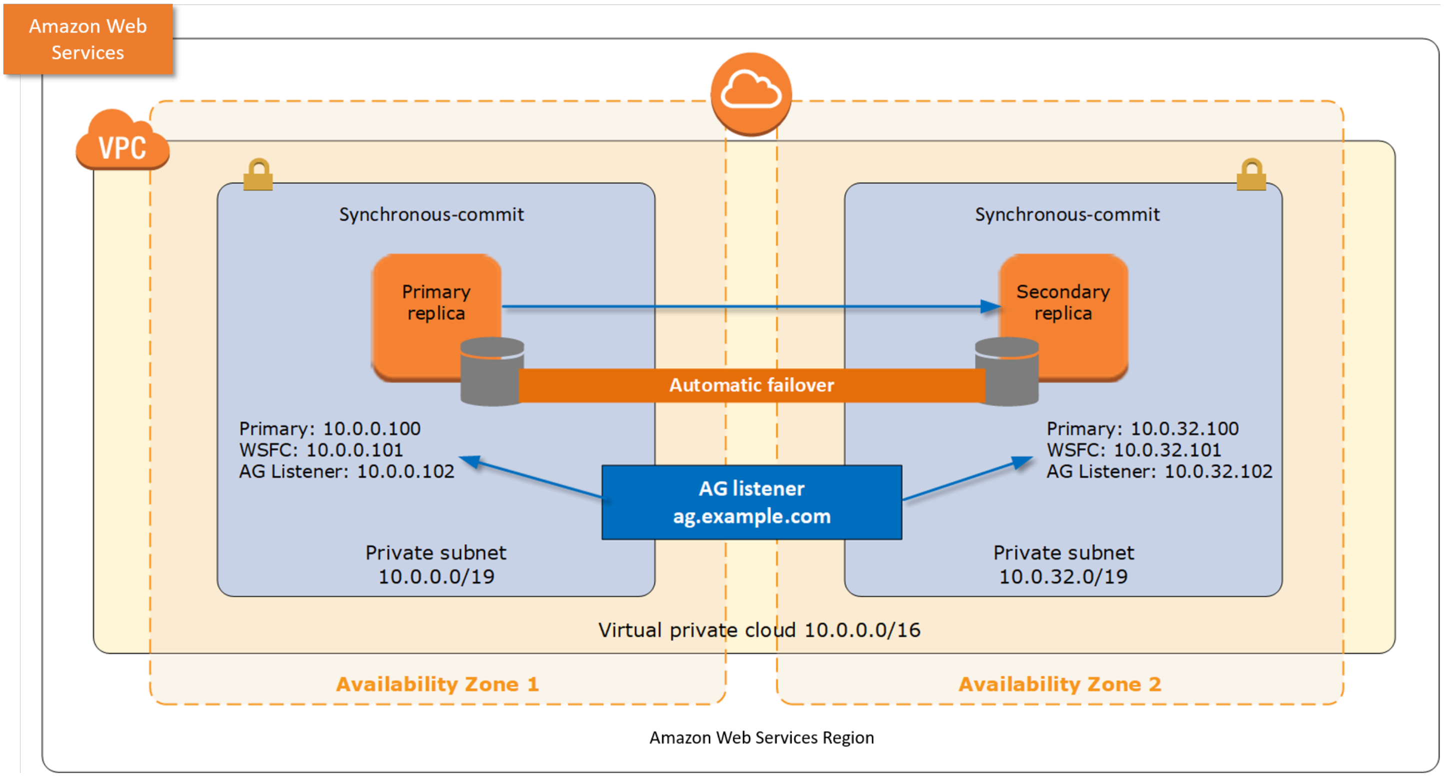
High availability

Using Amazon EC2, you can set the location of instances in multiple locations composed of AWS Regions and Availability Zones. Regions are dispersed and located in separate geographic areas. Availability Zones are distinct locations within a Region that are engineered to be isolated from failures in other Availability Zones. Availability Zones provide inexpensive, low-latency network connectivity to other Availability Zones in the same Region.

When you launch your instances in different Regions, you can set your SQL Server Always On application to be closer to specific customers, or to meet legal or other requirements. When you launch your instances in different Availability Zones, you can protect your SQL Server Always On applications from the failure of a single location.

Automatic failover

When you deploy AWS Launch Wizard with the default parameters, it configures a two-node, automatic failover cluster with a file share witness. An Always On Availability Group is deployed on this cluster with two availability replicas, as shown in the following diagram.



Launch Wizard implementation supports the following scenarios:

- Protection from the failure of a single instance
- Automatic failover between the cluster nodes
- Automatic failover between Availability Zones

The default implementation of Launch Wizard does not provide automatic failover in every case. For example, the failure of Availability Zone 1, which contains the primary node and file share witness, would prevent automatic failover to Availability Zone 2 because the cluster would fail as it loses quorum. In this scenario, you could follow manual disaster recovery steps that include restarting the cluster service and forcing quorum on the second cluster node (for example, WSFCNode2) to restore application availability. Launch Wizard also provides an option to deploy to three Availability Zones. This deployment option can mitigate the loss of quorum if a single node fails. However, you can select this option only in AWS Regions that include three or more Availability Zones. For a current list of supported Regions, see AWS Global Infrastructure.

Security groups and firewalls

Launch Wizard creates a number of security groups and rules for you. When Amazon EC2 instances are launched, they must be associated with a security group, which acts as a stateful firewall. You have complete control over the network traffic entering or leaving the security group. You can also build granular rules that are scoped by protocol, port number, and source or destination IP address or subnet. By default, all outbound traffic from a security group is permitted. Inbound 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 whitepaper discusses the different methods for securing your AWS infrastructure. Recommendations include providing isolation between application tiers using security groups. We recommend that you tightly control inbound traffic in order to reduce the attack surface of your EC2 instances.

Domain controllers and member servers require several security group rules to allow traffic for services such as AD DS replication, user authentication, Windows Time services, and Distributed File System (DFS), among others. The WSFC nodes running SQL Server must permit several additional ports to communicate with each other. Finally, instances launched into the application server tier must establish SQL client connections to the WSFC nodes.

In addition to security groups, the Windows Firewall must also be modified on the SQL Server instances. During the bootstrapping process, a script runs on each instance that opens the TCP ports 1433, 1434, 4022, 5022, 5023, and 135 on the Windows Firewall.

Troubleshoot AWS Launch Wizard for SQL Server

Each application in your account in the same AWS Region can be uniquely identified by the application name specified at the time of a deployment. The application name can be used to view the details related to the application launch.

For SQL Server deployments on Linux, you must use an instance type built on the Nitro System. EBS volumes are exposed as NVMe block devices on instances built with the Nitro System. Device names that are specified for NVMe EBS volumes in a block device mapping are renamed using NVMe device names (`/dev/nvme[[0-26]n1`). Launch Wizard deployments on Linux do not support block devices on Xen-virtualized instances.

Contents

- Pacemaker on Ubuntu (deployment on Linux)
- SQL Server Management Studio
- Errors

Active Directory objects and DNS record clean up (deployment on Windows)

When you delete a deployment, you lose all specification settings for the SQL Server Always On application. Launch Wizard attempts to delete only the AWS resources that it created in your account as part of the deployment. If you created resources outside of Launch Wizard, for example, resources in a VPC created by Launch Wizard, the deletion can fail. Launch Wizard does not delete Active Directory objects in your Active Directory, nor does it delete any of the records in your DNS server. Launch Wizard has no control over your Active Directory domain user password over time, which is required to clean up Active Directory objects or DNS records. We recommend that you remove these entries from your Active Directory after Launch Wizard deletes the deployment.

If the initial Active Directory objects or DNS records are not cleaned up, when you attempt to deploy Launch Wizard on an existing Active Directory using a deployment name that has already been used or availability group/listener/cluster name that has already been used, the deployment may fail with the following error.

Error message

```
System.Management.Automation.Remoting.PSRemotingTransportException:
Connecting to remote server xxxxxx failed with the following error
message : WinRM cannot complete the operation. Verify that the specified
computer name is valid, that the computer is accessible over the network,
and that a firewall exception for the WinRM service is enabled and allows
access from this computer. By default, the WinRM firewall exception for
public profiles limits access to remote computers within the same local
subnet.
```

To address this error, we recommend that you remove the initial entries from your Active Directory.

To clean up Active Directory Objects, run the following example PowerShell commands as a domain user with the appropriate authorization to perform these operations.

```
$Pwd = ConvertTo-SecureString $password -AsPlainText –Force
```

```
$cred = New-Object System.Management.Automation.PSCredential $domainUser,
$Pwd
```

```
$ADObject = Get-ADObject -Filter 'DNSHostName -eq “SQLnode.example.com”
```

```
Remove-ADObject -Recursive -Identity $ADObject -Credential $cred
```

To remove a DNS Record, the name of the record that you want to delete (SQL Server node name), the DNS server FQDN, and the DNS zone within which the record is residing are required. The following are example PowerShell commands to perform the DNS record removal.

```
$NodeDNS = Get-DnsServerResourceRecord -ZoneName $ZoneName -ComputerName
$DNSServer -Node $NodeToDelete -RRType A -ErrorAction SilentlyContinue
```

```
Remove-DnsServerResourceRecord -ZoneName $ZoneName -ComputerName $DNSServer
-InputObject $NodeDNS -Force
```

Launch Wizard provisioning events

Launch Wizard captures events from SSM Automation and CloudFormation to track the status of an ongoing application deployment. If an application deployment fails, you can view the deployment events for this application by selecting **Deployments** from the navigation pane. A failed event shows a status of **Failed** along with a failure message.

CloudWatch Logs

Launch Wizard streams provisioning logs from all of the AWS log sources, such as CloudFormation, SSM, and CloudWatch Logs. CloudWatch Logs for a given application name can be viewed on the CloudWatch console for the log group name `LaunchWizard-`*`APPLICATION_NAME`* and log stream `ApplicationLaunchLog`.

CloudFormation stack

Launch Wizard uses CloudFormation to provision the infrastructure resources of an application. CloudFormation stacks can be found in your account using the CloudFormation `describe-stacks` API. Launch Wizard launches various stacks in your account for validation and application resource creation. The following are the relevant filters for the `describe-stacks` API.

- **Validation**

`LaunchWizard-APPLICATION_NAME-checkCredentials-SSM_execution_id`

- **Validation**

  `LaunchWizard-APPLICATION_NAME-checkVPCConnectivity-SSM_execution_id`
- **Application resources**

  `LaunchWizard-APPLICATION_NAME`. This stack also has nested stacks for VPC, AD, the RDGW node, and SQL nodes.

You can view the status of these CloudFormation stacks. If any of them fail, you can view the cause of failure.

Pacemaker on Ubuntu (deployment on Linux)

To troubleshoot Pacemaker cluster resource issues, take the following actions as an administrator.

- Inspect the system log files for operating system errors and address the errors, as needed.
- Inspect the cluster log files for errors, including for errors that relate to Pacemaker, Corosync, or SQL Server. Check the log files carefully because the related services may provide only one or two related log entries.
- Verify resource configuration, and configuration of cluster-related functions.
  - The following commands display the configuration details:
    - To display all resources, use: `pcs resource show -full`.
    - Or, you can use: `pcs resource show <resource name>`.
  - The following command will display the cluster constraints: `pcs constraints –full`.
  - The following command displays the cluster properties: `pcs property list –all`.
- Manually start the resource with `debug-start`.
- Clear failed actions with the following command: `pcs resource cleanup <resource name>`.

SQL Server Management Studio

If you encounter issues when you attempt to add databases with SQL Server Management Studio, perform the following to add databases to the availability group:

1. Log in to the primary node using SQL Server Management Studio (SSMS) and record the name of the availability group.
2. Verify that the database that you want to add to the availability group is backed up.
3. Add the database to the availability group by running the following command in SSMS:

   ```
   ALTER AVAILABILITY GROUP ag-name ADD DATABASE db
   ```

4. Refresh the availability group and verify that the database was created.

Errors

Directory fails to create

- **Cause:** An internal service error has been encountered during the creation of the directory.
- **Solution:** Retry the operation. For this scenario, you must retry the deployment from the initial page of the Launch Wizard console.

Your requested instance type is not supported in your requested Availability Zone

- **Cause:** This failure might happen during the launch of either your RDGW instance or your SQL Server instance, or during the validation of the instances that Launch Wizard launches in your selected subnets.
- **Solution:** For this scenario, you must choose a different Availability Zone and retry the deployment from the initial page of the Launch Wizard console.

Validate connectivity for subnet. The following resource(s) failed to create: [ValidationNodeWaitCondition]

This failure can occur for multiple reasons. The following list shows known causes and solutions.

- **VPC or subnet configuration does not meet prerequisites**
  - **Cause:** This failure occurs when your VPC or subnet configuration does not meet the prerequisites documented in the VPC Connectivity Section under Deploy an application with AWS Launch Wizard for SQL Server on Windows (Console). If the failure message points to your selected public subnet, then the public subnet is not configured for outbound internet

  access. If the failure message points to one of your selected private subnets, then the specified private subnet does not have outbound connectivity.
  - **Solution:** Check that your VPC includes one public subnet and, at least, two private subnets. Your VPC must be associated with a DHCP Options Set to enable DNS translations to work. The private subnets must have outbound connectivity to the internet and other AWS services (S3, CFN, SSM, and Logs). We recommend that you enable this connectivity with a NAT Gateway. Note that, in the console, when you select a private subnet for the public subnet dropdown or you select a public subnet for the private subnet dropdown, you will encounter the same error. Please refer to the VPC Connectivity section under Deploy an application with AWS Launch Wizard for SQL Server on Windows (Console) for more information about how to configure your VPC.
- **EC2 instance stabilization error**
  - **Cause:** Failure can occur if the EC2 instance used for validation fails to stabilize. When this happens, the EC2 instance is unable to communicate to the CloudFormation service to signal completions, resulting in `WaitCondition` errors.
  - **Solution:** Please contact Support for assistance.

AWS Launch Wizard workload availability

Note

End of support notice: On May 1, 2025, AWS Launch Wizard will discontinue support for Amazon Elastic Kubernetes Service, Microsoft Internet Information Services, and Microsoft Exchange Server. After May 1, 2025, you can no longer use AWS Launch Wizard to access these workloads.

AWS Launch Wizard supports workloads based on the underlying resources it creates in an AWS Region.

Some deployments don't support every configuration and setting for every Region. For such deployments, the configurations and settings won't be listed in the Launch Wizard console for the Region.

The following table describes which workloads are available in which Regions.

Name	Code	Active Directory	RD Gateway	SAP	SQL Server
US East (N. Virginia)	`us-east-1`	✓	✓	✓	✓
US East (Ohio)	`us-east-2`	✓	✓	✓	✓
US West (N. California)	`us-west-1`	✓	✓	✓	✓
US West (Oregon)	`us-west-2`	✓	✓	✓	✓
Africa (Cape Town)	`af-south-1`	✓	✓	✓	✓

Name	Code	Active Directory	RD Gateway	SAP	SQL Server
Asia Pacific (Hong Kong)	ap-east-1	✓	✓	✓	✓
Asia Pacific (Hyderabad)	ap-south-2			✓	✓
Asia Pacific (Jakarta)	ap-southeast-3			✓	✓
Asia Pacific (Malaysia)	ap-southeast-5		✓	✓	✓
Asia Pacific (Melbourne)	ap-southeast-4			✓	✓
Asia Pacific (Mumbai)	ap-south-1	✓	✓	✓	✓
Asia Pacific (Osaka)	ap-northeast-3	✓	✓	✓	✓
Asia Pacific (Seoul)	ap-northeast-2	✓	✓	✓	✓
Asia Pacific (Singapore)	ap-southeast-1	✓	✓	✓	✓
Asia Pacific (Sydney)	ap-southeast-2	✓	✓	✓	✓
Asia Pacific (Thailand)	ap-southeast-7				
Asia Pacific (Tokyo)	ap-northeast-1	✓	✓	✓	✓

Name	Code	Active Directory	RD Gateway	SAP	SQL Server
Canada (Central)	ca-central-1	✓	✓	✓	✓
Canada West (Calgary)	ca-west-1		✓	✓	✓
China (Beijing)	cn-north-1			✓	✓
China (Ningxia)	cn-northwest-1			✓	✓
Europe (Frankfurt)	eu-central-1	✓	✓	✓	✓
Europe (Ireland)	eu-west-1	✓	✓	✓	✓
Europe (London)	eu-west-2	✓	✓	✓	✓
Europe (Milan)	eu-south-1	✓	✓	✓	✓
Europe (Paris)	eu-west-3	✓	✓	✓	✓
Europe (Spain)	eu-south-2			✓	✓
Europe (Stockholm)	eu-north-1	✓	✓	✓	✓
Europe (Zurich)	eu-central-2			✓	✓

Name	Code	Active Directory	RD Gateway	SAP	SQL Server
Israel (Tel Aviv)	`il-central-1`		✓	✓	✓
Mexico (Central)	`mx-central-1`				
Middle East (Bahrain)	`me-south-1`	✓	✓	✓	✓
Middle East (UAE)	`me-central-1`			✓	✓
South America (São Paulo)	`sa-east-1`	✓	✓	✓	✓
AWS GovCloud (US-East)	`us-gov-east-1`			✓	✓
AWS GovCloud (US-West)	`us-gov-west-1`			✓	✓

AWS Launch Wizard security

Cloud security at AWS is the highest priority. As an AWS customer, you benefit from a data center and network architecture that is built to meet the requirements of the most security-sensitive organizations.

Security is a shared responsibility between AWS and you. The shared responsibility model describes this as security *of* the cloud and security *in* the cloud:

- **Security of the cloud** – AWS is responsible for protecting the infrastructure that runs AWS services in the AWS Cloud. AWS also provides you with services that you can use securely. Third-party auditors regularly test and verify the effectiveness of our security as part of the AWS Compliance Programs. To learn about the compliance programs that apply to AWS Launch Wizard, see AWS Services in Scope by Compliance Program.
- **Security in the cloud** – Your responsibility is determined by the AWS service that you use. You are also responsible for other factors including the sensitivity of your data, your company's requirements, and applicable laws and regulations.

This documentation helps you understand how to apply the shared responsibility model when using AWS Launch Wizard. The following topics show you how to configure Launch Wizard to meet your security and compliance objectives. You also learn how to use other AWS services that help you to monitor and secure your Launch Wizard resources.

AWS Launch Wizard deploys Amazon EC2 instances into virtual private clouds. For security information for Amazon EC2 and Amazon VPC, see the security sections in the Amazon EC2 Getting Started Guide and the Amazon VPC User Guide.

This section of the Launch Wizard User Guide provides security information that pertains to AWS Launch Wizard. For security topics specific to AWS Launch Wizard for SQL Server, see Security groups and firewalls. For security topics specific to AWS Launch Wizard for SAP, see Security groups in AWS Launch Wizard for SAP.

Launch Wizard security topics

- Infrastructure security in Launch Wizard
- Resilience in Launch Wizard
- Data protection in Launch Wizard
- Identity and Access Management for AWS Launch Wizard

- Update management in Launch Wizard
- AWS managed policies for AWS Launch Wizard

Infrastructure security in Launch Wizard

As a managed service, AWS Launch Wizard is protected by the AWS global network security. For information about AWS security services and how AWS protects infrastructure, see AWS Cloud Security. To design your AWS environment using the best practices for infrastructure security, see Infrastructure Protection in *Security Pillar AWS Well‐Architected Framework*.

Resilience in Launch Wizard

The AWS global infrastructure is built around AWS Regions and Availability Zones. Regions provide multiple physically separated and isolated Availability Zones, which are connected through low-latency, high-throughput, and highly redundant networking. With Availability Zones, you can design and operate applications and databases that automatically fail over between Availability Zones without interruption. Availability Zones are more highly available, fault tolerant, and scalable than traditional single or multiple data center infrastructures.

For more information about AWS Regions and Availability Zones, see AWS Global Infrastructure.

AWS Launch Wizard sets up an application across multiple Availability Zones to ensure automatic failover between Availability Zones without interruption. Availability Zones are more highly available, fault tolerant, and scalable than traditional single or multiple datacenter infrastructures.

Data protection in Launch Wizard

The AWS shared responsibility model applies to data protection in AWS Launch Wizard. As described in this model, AWS is responsible for protecting the global infrastructure that runs all of the AWS Cloud. You are responsible for maintaining control over your content that is hosted on this infrastructure. You are also responsible for the security configuration and management tasks for the AWS services that you use. For more information about data privacy, see the Data Privacy FAQ. For information about data protection in Europe, see the AWS Shared Responsibility Model and GDPR blog post on the *AWS Security Blog*.

For data protection purposes, we recommend that you protect AWS account credentials and set up individual users with AWS IAM Identity Center or AWS Identity and Access Management (IAM).

That way, each user is given only the permissions necessary to fulfill their job duties. We also recommend that you secure your data in the following ways:

- Use multi-factor authentication (MFA) with each account.
- Use SSL/TLS to communicate with AWS resources. We require TLS 1.2 and recommend TLS 1.3.
- Set up API and user activity logging with AWS CloudTrail. For information about using CloudTrail trails to capture AWS activities, see Working with CloudTrail trails in the *AWS CloudTrail User Guide*.
- Use AWS encryption solutions, along with all default security controls within AWS services.
- Use advanced managed security services such as Amazon Macie, which assists in discovering and securing sensitive data that is stored in Amazon S3.
- If you require FIPS 140-3 validated cryptographic modules when accessing AWS through a command line interface or an API, use a FIPS endpoint. For more information about the available FIPS endpoints, see Federal Information Processing Standard (FIPS) 140-3.

We strongly recommend that you never put confidential or sensitive information, such as your customers' email addresses, into tags or free-form text fields such as a **Name** field. This includes when you work with Launch Wizard or other AWS services using the console, API, AWS CLI, or AWS SDKs. Any data that you enter into tags or free-form text fields used for names may be used for billing or diagnostic logs. If you provide a URL to an external server, we strongly recommend that you do not include credentials information in the URL to validate your request to that server.

Encryption with AWS managed keys and customer managed keys

AWS Launch Wizard for Active Directory, SQL Server, and SAP use the default AWS managed keys to encrypt Amazon EBS volumes. Launch Wizard for SAP also supports the use of customer managed keys that you have already created.

If you don't specify a customer managed key, Launch Wizard for SAP automatically creates an AWS managed key in your AWS account.

If you want to use a customer managed key for Launch Wizard for SAP, see the steps for adding permissions to your KMS key policy for Launch Wizard to use your KMS key at Add permissions to use AWS KMS keys in the *Launch Wizard for SAP User Guide*.

Creating your own customer managed CMK gives you more flexibility and control. For example, you can create, rotate, and disable customer managed keys. You can also define access controls and audit the customer managed keys that you use to protect your data. For more information about customer managed keys and AWS managed keys, see AWS KMS concepts in the AWS Key Management Service Developer Guide.

Identity and Access Management for AWS Launch Wizard

AWS Launch Wizard uses the following AWS managed policies to grant permissions to users and services.

- **AmazonEC2RolePolicyForLaunchWizard**

  AWS Launch Wizard creates an IAM role with the name **AmazonEC2RoleForLaunchWizard** in your account if the role already does not already exist in your account. If the role exists, the role is attached to the instance profile for the Amazon EC2 instances that Launch Wizard will launch into your account. This role is comprised of two IAM managed policies: **AmazonSSMManagedInstanceCore** and **AmazonEC2RolePolicyForLaunchWizard**.

  When you choose to deploy your SAP application with AWS Backint Agent for SAP HANA, you must attach the IAM inline policy provided in Step 2 of the AWS Identity and Access Management documentation for AWS Backint Agent for SAP HANA. This policy and instructions to attach the policy to the role are provided by Launch Wizard.
- **AmazonSSMManagedInstanceCore**

  This policy enables AWS Systems Manager service core functionality on Amazon EC2. For information, see Create an IAM Instance Profile for Systems Manager.
- **AmazonLaunchWizardFullAccessV2**

  This policy provides full access to AWS Launch Wizard and other required services.
- **AWSLambdaVPCAccessExecutionRole**

  This policy provides minimum permissions for a Lambda function to execute while accessing a resource within a VPC. These permissions include create, describe, delete network interfaces, and write permissions to CloudWatch Logs.
- **AmazonLambdaRolePolicyForLaunchWizardSAP**

  This policy provides minimum permissions to enable SAP provisioning scenarios on Launch Wizard. It allows invocation of Lambda functions to be able to perform certain actions, such as validation of route tables and perform pre-configuration and configuration tasks for HA mode enabling.

- To run custom pre- and post-configuration deployment scripts, you must manually add the permissions provided in Add permissions to run custom pre- and post-deployment configuration scripts to the `AmazonEC2RoleForLaunchWizard` role.
- To save generated artifacts from Launch Wizard for SAP to Amazon S3, and your S3 bucket name does not include the prefix `launchwizard`, you must attach the policy provided in Add permissions to save deployment artifacts to Amazon S3 to the IAM user.
- To grant permissions for users to launch AWS Service Catalog products created with Launch Wizard for SAP, follow the steps in Set up to launch AWS Service Catalog products created with AWS Launch Wizard.
- To grant permissions to AWS Service Catalog to create a launch constraint for users who want to launch an AWS Service Catalog product created by Launch Wizard for SAP, follow the steps in Create a launch constraint.

If you deploy domain controllers into an existing VPC with an existing Active Directory, Launch Wizard for Active Directory requires domain administrator credentials to be added to Secrets Manager in order to join your domain controllers to Active Directory and promote them. In addition, the following resource policy must be attached to the secret so that Launch Wizard can access the secret. Launch Wizard guides you through the process of attaching the required policy to your secret.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": {
            "AWS":
                "arn:aws:iam::111122223333:role/service-role/
AmazonEC2RoleForLaunchWizard"
            },
            "Action": [
                "secretsmanager:GetSecretValue",
                "secretsmanager:CreateSecret",
                "secretsmanager:GetRandomPassword"
            ],
            "Resource": "*"
        }
    ]
}
```

Update management in Launch Wizard

We recommend that you regularly patch, update, and secure the operating system and applications on your EC2 instances. You can use AWS Systems Manager Patch Manager to automate the process of installing security-related updates for both the operating system and applications. Alternatively, you can use any automatic update services or recommended processes for installing updates that are provided by the application vendor.

AWS managed policies for AWS Launch Wizard

An AWS managed policy is a standalone policy that is created and administered by AWS. AWS managed policies are designed to provide permissions for many common use cases so that you can start assigning permissions to users, groups, and roles.

Keep in mind that AWS managed policies might not grant least-privilege permissions for your specific use cases because they're available for all AWS customers to use. We recommend that you reduce permissions further by defining customer managed policies that are specific to your use cases.

You cannot change the permissions defined in AWS managed policies. If AWS updates the permissions defined in an AWS managed policy, the update affects all principal identities (users, groups, and roles) that the policy is attached to. AWS is most likely to update an AWS managed policy when a new AWS service is launched or new API operations become available for existing services.

For more information, see AWS managed policies in the *IAM User Guide*.

Managed policies:

- AWS managed policy: AmazonLaunchWizardFullAccessV2
- AWS managed policy: AmazonEC2RolePolicyForLaunchWizard

- AWS Launch Wizard updates to AWS managed policies

AWS managed policy: AmazonLaunchWizardFullAccessV2

You can attach the `AmazonLaunchWizardFullAccessV2` policy to your IAM identities.

This policy grants administrative permissions that allow full access to AWS Launch Wizard and other required services. To view the managed policy content, see the AmazonLaunchWizardFullAccessV2 page in the *AWS Managed Policy Reference Guide*.

Permissions details

This policy includes the following permissions.

- `launchwizard` – Allows all Launch Wizard actions.
- `applicationinsights` – Allows all CloudWatch Application Insights actions. This permission is required so that an application can be tracked and configured by CloudWatch Application Insights, which provides Launch Wizard with more visibility and insight into the service through functionality such as monitoring and data analysis.
- `route53` – Allows changing and listing resource record sets, listing hosted zones, and listing hosted zones by name. This is required so that scripts running on instances in your account for SAP deployments can perform these actions.
- `s3` – Allows all get or list operations for all resources, and allows for creation, deletion, and getting objects from a bucket, and putting objects in a bucket for certain Launch Wizard and SAP resources. This is required so that the Launch Wizard service can both view and update buckets and contents in Amazon S3 for tasks such as reading and storing scripts that are run on instances in its deployments.
- `kms` – Allows listing all AWS KMS keys and aliases. This is required so that Launch Wizard can view keys and aliases in your account.
- `cloudwatch` – Allows all get, list, or describe actions for all resources, and allows Launch Wizard alarms and instance profiles to be created, updated, deleted, or described. This is required so that Launch Wizard can create and manage alarms to track metrics.
- `ec2` – Allows creation of all security groups, authorization of ingress rules for all security groups, all get or describe operations, and creation of all VPCs, NAT/internet gateways, subnets, routes/route tables, and key pairs. Allows instances from the CloudFormation stacks in Launch Wizard deployments to be stopped or terminated. Allows anything called from the Launch Wizard endpoint to perform other Amazon EC2 actions. This is required so that all EC2-related resources

deployed from the Launch Wizard CloudFormation stacks can be appropriately created and managed.

- `cloudformation` – Allows all Launch Wizard and CloudWatch Application Insights CloudFormation stacks to be described and listed. Allows all get operations, all resources to be signaled, and all Launch Wizard stacks to be deleted. Allows all stacks to be created, and allows describe account limits, describe stack drift detection status, all list operations, and tagging of resources with all tag keys, starting with "LaunchWizard". This is required so that Launch Wizard can create CloudFormation stacks in your account, so that the stacks are appropriately signaled, and so that you can view and delete those stacks.
- `iam` – Allows Launch Wizard EC2 roles and instance profiles to be created and deleted and attached/detached. Allows Launch Wizard EC2 and AWS Lambda roles and instance profiles to be passed a role as long as it is passed to Lambda or EC2. Allows get operations for all roles or policies, all list operations, and all roles linked to Amazon EC2 Auto Scaling, CloudWatch Application Insights, or Amazon EventBridge to be created. This is required so that Launch Wizard can create necessary roles and attach the appropriatepolicies to them to ensure that resources in the Launch Wizard CloudFormation stacks and elsewhere in the service have the appropriate permissions.
- `autoscaling` – Allows Launch Wizard Auto Scaling groups, launch configurations, and associated tags, to be created, deleted, and updated. This is required so that the Launch Wizard SQL CloudFormation stacks can perform these actions for the RDGW nodes in its deployments.
- `logs` – Allows log groups with names beginning with `LaunchWizard` to be created and deleted. Allows log streams, log events, and tags to be created, listed, and deleted for log groups with names that begin with `LaunchWizard`. This is required so that Launch Wizard can publish logs to your account so that a you can view the events from their deployments.
- `sns` – Allows Launch Wizard Amazon SNS topics to be created, deleted, subscribed to, and unsubscribed from. Allows all Amazon SNS subscriptions to be listed and messages to be published. This is required so that the Launch Wizard Amazon SNS queues to send signals between resources and Launch Wizard Lambda functions know when to proceed with steps in their event-based workflows.
- `resource-groups` – Allows resource groups whose names begin with "LaunchWizard" to be created, deleted, or listed. This is required so that Launch Wizard resources can be grouped together in a resource group, and so that the groups can be viewed or deleted.
- `ds` – Allows creation and deletion of a Microsoft Active Directory, adding IP routes, and all describe operations. This is required so that Active Directories can be created, deleted, and viewed in Launch Wizard SQL Server deployments, and so that IP routes can be added to them.

- `sqs` – Allows all queues with "SQS" in the name to be tagged, listed, created, and deleted. Allows any queue attributes to be set and read, and for the queue URL to be read and permissions added. This is required so that Launch Wizard SAP deployments can have a queue in the deployment on which these actions can be performed.
- `elasticfilesystem` – Allows all Amazon Elastic File System (Amazon EFS) resources, and associated tags, to be created, deleted, and described. Allows mount targets to be created, deleted, and described. This is required so that Launch Wizard SAP deployments can create file systems in your account with the appropriate mount targets.
- `lambda` – Allows AWS Lambda functions with "LaunchWizard" in the name to be created, deleted, read, and invoked. This is required so that Launch Wizard SAP deployments can perform some Lambda functions at the end of CloudFormation stacks for configuration in your account or for parameter validation.
- `dynamodb` – Allows all tables with a name starting with "LaunchWizard" to be created, deleted, or described. This is required so that Launch Wizard scripts for SAP can publish events and metadata from the events of the running threads into a Amazon DynamoDB table in your account.
- `secretsmanager` – Allows all secrets with a name starting with "LaunchWizard" to be created, deleted, retrieved, and restored, all resources to be tagged or untagged, all resource policies to be created and deleted, secret version IDs to be listed, and secret values to be updated. Allows all random passwords to be generated and all secrets to be listed. This is required so that secrets can be created in your account to perform operations, such as decrypting a password in order to RDP into an instance from their deployment.
- `fsx` – Allows Amazon FSx file systems to be created by Launch Wizard. Allows describing file system properties, listing all tags on the Amazon FSx file share, adding and removing tags. Allows deleting file systems and volumes where tags include `LaunchWizard` in the CloudFormation stack-id tag.
- `servicecatalog` – Allows for the creation of AWS Service Catalog portfolios, products, and launch constraints. Allows for associated tags to be created and deleted. Allows for the association between a product and portfolio, and also the association between the IAM principal of a user and a portfolio.
- `ssm` – Allows for all get, list, tag, execute, and delete operations for all SSM resources. This is required so that Launch Wizard can create, run, and delete SSM resources on your behalf to configure your Amazon EC2 instances for application provisioning. Allows Launch Wizard to create and delete associations using the `AWS-ConfigureAWSPackage` document, which allows AWS Data Provider for SAP installations.

Note

arn:aws:s3:::launchwizard* and “arn:aws:s3:::launchwizard*/* are redundant permissions. Both permissions are present for historical purposes and do not impact security.

AWS managed policy: AmazonEC2RolePolicyForLaunchWizard

This policy grants administrative permissions that allow all AWS Launch Wizard actions to be performed. To view the managed policy content, see the AmazonEC2RolePolicyForLaunchWizard page in the *AWS Managed Policy Reference Guide*.

Permissions details

This policy includes the following permissions.

- launchwizard – Allows all Launch Wizard actions.
- ec2 – Allows starting, stopping, and rebooting instances, and attaching volumes to all instances with the LaunchWizardResourceGroupID tag. Allows replacing route table for all instances with the LaunchWizardApplicationType resource tag. Allows all resources to describe and associate IP addresses, describe instances, images, Regions, volumes, and route tables, and modify instance attributes for all resources. Allows creating tags and volumes for all resources with the LaunchWizardResourceType or LaunchWizardResourceGroupID tags.
- cloudwatch – Allows for getting and writing metrics to CloudWatch. This is required so that CloudWatch can write logs for all resources.
- s3 – Allows all get or list operations for all resources, and allows for creation, deletion, and getting objects from a bucket, and putting objects in a bucket for certain Launch Wizard and SAP resources. This is required so that the Launch Wizard service can both view and update buckets and contents in Amazon S3 for tasks such as reading and storing scripts that are run on instances in its deployments.
- ssm – Allows send commands to all Amazon EC2 instances with the LaunchWizardApplicationType resource tag. Allows getting a document. These actions are required to run the Backint install agent SSM document for SAP.
- logs – Allows all log groups or log streams for all write and read log events. This is required so that Launch Wizard can publish logs to your account so that you can view the events from their deployments.

- cloudformation – Allows all Launch Wizard and CloudWatch Application Insights CloudFormation stacks to be described and listed. Allows all get operations and for all resources to be signaled. This is required so that the stacks are appropriately signaled by CloudFormation.
- dynamodb – Allows all tables with a name starting with "LaunchWizard" to be created, deleted, or described. This is required so that Launch Wizard scripts for SAP can publish events and metadata from the events of the running threads into a Amazon DynamoDB table in your account.
- sqs – Allows sending and receiving messages from Amazon SQS queues. This is required so that Launch Wizard SAP deployments can have a queue in the deployment on which these actions can be performed.
- iam – Allows Launch Wizard EC2 roles and instance profiles to be created and deleted and attached/detached. Allows Launch Wizard EC2 and AWS Lambda roles and instance profiles to be passed a role as long as it is passed to Lambda or EC2. Allows get operations for all roles or policies, all list operations, and all roles linked to Amazon EC2 Auto Scaling, CloudWatch Application Insights, or Amazon EventBridge to be created. This is required so that Launch Wizard can create necessary roles and attach the appropriate policies to them to ensure that resources in the Launch Wizard CloudFormation stacks and elsewhere in the service have the appropriate permissions.
- fsx – Allows describing file systems and listing tags on file systems on any Amazon FSx resource tagged with the LaunchWizard tag. This is required so that Launch Wizard can retrieve the FSX DNS and administration endpoints to create the FCI SQL cluster.

AWS Launch Wizard updates to AWS managed policies

View details about updates to AWS managed policies for AWS Launch Wizard since this service began tracking these changes. For automatic alerts about changes to this page, subscribe to the RSS feed on the AWS Launch Wizard Document history page.

Change	Description	Date
AmazonEC2RolePolicyForLaunchWizard – Policy update	AWS Launch Wizard added a new permission to the policy for the InstallBackintForAWSBackup Systems Manager document.	September 25, 2024

Change	Description	Date
	It enables the Systems Manager document to install AWS Backint agent for AWS Backup.	
AmazonLaunchWizardFullAccessV2 – New policy	AWS Launch Wizard added this new policy to replace the `AmazonLaunchWizard_Fullaccess` policy. This policy grants administrative permissions that allow full access to Launch Wizard and other required services.	September 1, 2023
AmazonLaunchWizard_Fullaccess – Policy deprecation	This policy has been replaced by `AmazonLaunchWizardFullAccessV2` .	August 23, 2023

Change	Description	Date
AmazonLaunchWizard _Fullaccess – Update to an existing policy	- AWS Launch Wizard added permissions to create or update tags for Auto Scaling groups with the `arn:aws:autoscaling:*:*:autoScalingGroup:*:autoScalingGroupName/LaunchWizard*` resource. - AWS Launch Wizard added new permissions to support creating and deleting tags for Amazon Elastic File System (Amazon EFS) resources called through `launchwizard.amazonaws.com` .	February 23, 2023

Change	Description	Date
AmazonLaunchWizard _Fullaccess – Update to an existing policy	- AWS Launch Wizard added new policies to support creating and deleting tags for log groups with the `arn:aws:logs:*:*:log-group:LaunchWizard*` resource, called through `launchwizard.amazonaws.com` . - AWS Launch Wizard added new policies to support creating and deleting tags for AWS Service Catalog resources with the `arn:aws:servicecatalog:*:*:*/*` or `arn:aws:catalog:*:*:*/*` resource, called through `launchwizard.amazonaws.com` . - AWS Launch Wizard added permissions to create and delete associations that run the `AWS-ConfigureAWSPackage` document when th ey are called through `launchwizard.amazonaws.com` . This allows Launch Wizard to create associations that will install	January 12, 2023

Change	Description	Date
	the AWS Data Provider for SAP.	
AmazonEC2RolePolicyForLaunchWizard – Update to an existing policy	- AWS Launch Wizard added new policies to support FSx creation with Launch Wizard to support SQL Server ONTAP. AWS Launch Wizard will perform the `fsx:DescribeStorageVirtualMachines` action on all resources created by Launch Wizard with `LaunchWizard*` in the tag when they are called via `launchwizard.amazonaws.com` to enable this support.	May 17, 2022
AmazonLaunchWizard_Fullaccess – Update to an existing policy	- AWS Launch Wizard restricted `ssm` actions to only documents containing the `LaunchWizard` prefix and called by the Launch Wizard service to improve the security of this managed policy.	April 12, 2022

Change	Description	Date
AmazonLaunchWizard_Fullaccess – Update to an existing policy	- AWS Launch Wizard restricted `ssm:sendCommand` actions to only the `arn:aws:ec2:*:*:instance/*` resource and to resources with the tag keys `aws:cloudformation:stack-id` to improve the security of this managed policy.	February 9, 2022
AmazonLambdaRoleForLaunchWizard – Policy deprecation	- AWS Launch Wizard deprecated the `AmazonLambdaRoleForLaunchWizard` policy because it is no longer used by the service.	February 7, 2022

Change	Description	Date
AmazonEC2RolePolicyForLaunchWizard – Update to an existing policy	- AWS Launch Wizard restricted the `ec2:CreateTags` and `ec2:CreateVolume` actions to the `arn:aws:ec2:*:*:volume/*` resource to prohibit tagging of other resources with the tag keys `LaunchWizardResourceGroupID` and `LaunchWizardApplicationType` to improve the security of the managed policy.	February 7, 2022

Change	Description	Date
AmazonLaunchWizard_Fullaccess – Update to an existing policy	- AWS Launch Wizard added new policies to support the creation of AWS Service Catalog portfolios and products with Launch Wizard. AWS Launch Wizard will perform `servicecatalog:CreatePortfolio` , `servicecatalog:DescribePortfolio` , `servicecatalog:CreateConstraint` , `servicecatalog:CreateProduct` , `servicecatalog:AssociatePrincipalWithPortfolio` , `servicecatalog:CreateProvisioningArtifact` , and `servicecatalog:AssociateProductWithPortfolio` actions on AWS Service Catalog resources when they are called by Launch Wizard.	August 30, 2021

Change	Description	Date
AmazonEC2RolePolicyForLaunchWizard – Update to an existing policy	- AWS Launch Wizard added new policies to support FSx creation with Launch Wizard. AWS Launch Wizard will perform `fsx:DescribeFileSystems` and `fsx:ListTagsforResource` actions on all resources created by Launch Wizard with `LaunchWizard*` in the tag when they are called via `launchwizard.amazonaws.com` to enable this support.	May 21 2021

Change	Description	Date
AmazonLaunchWizard_Fullaccess – Update to an existing policy	- AWS Launch Wizard added new permissions to allow `PlacementGroup` for SAP HANA scale-out scenarios. AWS Launch Wizard will perform `ec2:ModifyInstancePlacement` , `ec2:DeletePlacementGroup` , and `ec2CreatePlacementGroup` actions on the database instances (in HANA and NetWeaver on HANA scenarios) in your account when they are called via `launchwizard.amazonaws.com` to enable this support. - AWS Launch Wizard added new permissions to create an SNS topic in your account, and subscribe to it, unsubscribe from it, and delete it. Permissions are restricted only to resources whose names begin with "Launch Wizard." AWS Launch Wizard will perform `sns:CreateTopic` , `sns:DeleteTopic` , `sns:Subscribe` , and	April 30, 2021

Change	Description	Date
	`sns:Unsubscribe` actions in your account when they are called via `launchwizard.amazonaws.com` to enable this support. • AWS Launch Wizard added new permissions to perform FSx operations to support SQL Server FCI on AWS Launch Wizard. Launch Wizard will perform `fsx:CreateFileSystem` , `fsx:DescribeFileSystem` , `fsx:ListTagsForResource` , `fsx:TagResource` , and `fsx:UntagResource` actions in your account when they are called via `launchwizard.amazonaws.com` to enable this support. • AWS Launch Wizard added a new permission to perform AWS Secrets Manager operations to support retrieving secret values from Secrets Manager on AWS Launch Wizard. Launch Wizard will perform the `arn:aws:s`	

Change	Description	Date
	`ecretsmanager:*:secret:LaunchWizard*` action in your account when they are called via `launchwizard.amazonaws.com` to enable this support.	
AWS Launch Wizard started tracking changes	AWS Launch Wizard started tracking changes for its AWS managed policies.	April 30, 2021

Logging AWS Launch Wizard API calls using AWS CloudTrail

AWS Launch Wizard is integrated with AWS CloudTrail, a service that provides a record of actions taken by a user, role, or an AWS service. CloudTrail captures API calls for AWS Launch Wizard as events. The calls captured include calls from the AWS Management Console and code calls to the AWS Launch Wizard API operations. Using the information collected by CloudTrail, you can determine the request that was made to AWS Launch Wizard, the IP address from which the request was made, when it was made, and additional details.

Every event or log entry contains information about who generated the request. The identity information helps you determine the following:

- Whether the request was made with root user or user credentials.
- Whether the request was made on behalf of an IAM Identity Center user.
- Whether the request was made with temporary security credentials for a role or federated user.
- Whether the request was made by another AWS service.

CloudTrail is active in your AWS account when you create the account and you automatically have access to the CloudTrail **Event history**. The CloudTrail **Event history** provides a viewable, searchable, downloadable, and immutable record of the past 90 days of recorded management events in an AWS Region. For more information, see Working with CloudTrail Event history in the *AWS CloudTrail User Guide*. There are no CloudTrail charges for viewing the **Event history**.

For an ongoing record of events in your AWS account past 90 days, create a trail or a CloudTrail Lake event data store.

CloudTrail trails

> A *trail* enables CloudTrail to deliver log files to an Amazon S3 bucket. All trails created using the AWS Management Console are multi-Region. You can create a single-Region or a multi-Region trail by using the AWS CLI. Creating a multi-Region trail is recommended because you capture activity in all AWS Regions in your account. If you create a single-Region trail, you can view only the events logged in the trail's AWS Region. For more information about trails, see Creating a trail for your AWS account and Creating a trail for an organization in the *AWS CloudTrail User Guide*.

You can deliver one copy of your ongoing management events to your Amazon S3 bucket at no charge from CloudTrail by creating a trail, however, there are Amazon S3 storage charges. For more information about CloudTrail pricing, see AWS CloudTrail Pricing. For information about Amazon S3 pricing, see Amazon S3 Pricing.

CloudTrail Lake event data stores

CloudTrail Lake lets you run SQL-based queries on your events. CloudTrail Lake converts existing events in row-based JSON format to Apache ORC format. ORC is a columnar storage format that is optimized for fast retrieval of data. Events are aggregated into *event data stores*, which are immutable collections of events based on criteria that you select by applying advanced event selectors. The selectors that you apply to an event data store control which events persist and are available for you to query. For more information about CloudTrail Lake, see Working with AWS CloudTrail Lake in the *AWS CloudTrail User Guide*.

CloudTrail Lake event data stores and queries incur costs. When you create an event data store, you choose the pricing option you want to use for the event data store. The pricing option determines the cost for ingesting and storing events, and the default and maximum retention period for the event data store. For more information about CloudTrail pricing, see AWS CloudTrail Pricing.

AWS Launch Wizard management events in CloudTrail

Management events provide information about management operations that are performed on resources in your AWS account. These are also known as control plane operations. By default, CloudTrail logs management events.

AWS Launch Wizard logs control plane operations as management events. For a list of the control plane operations, see the AWS Launch Wizard API Reference.

CloudTrail event examples

An event represents a single request from any source and includes information about the requested API operation, the date and time of the operation, request parameters, and so on. CloudTrail log files aren't an ordered stack trace of the public API calls, so events don't appear in any specific order.

Examples

- Example: CreateDeployment

- Example: DeleteDeployment
- Example: GetDeployment
- Example: GetWorkload
- Example: ListDeploymentEvents
- Example: ListDeployments
- Example: ListWorkloadDeploymentPattern
- Example: ListWorkloads

Example: CreateDeployment

The following example shows a CloudTrail log entry that demonstrates the `CreateDeployment` operation.

```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "AssumedRole",
        "principalId": "AIDAQRSTUVWXYZEXAMPLE:ExampleAssumedRoleSessionName",
        "arn": "arn:aws:sts::123456789012:assumed-role/ExampleAssumedRole/
ExampleRoleSessionName",
        "accountId": "123456789012",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "sessionContext": {
            "sessionIssuer": {
                "type": "Role",
                "principalId": "AIDAQRSTUVWXYZEXAMPLE",
                "arn": "arn:aws:iam::123456789012:role/ExampleAssumedRole",
                "accountId": "123456789012",
                "userName": "ExampleAssumedRole"
            },
            "webIdFederationData": {},
            "attributes": {
                "creationDate": "2023-09-05T17:45:15Z",
                "mfaAuthenticated": "false"
            }
        }
    },
    "eventTime": "2023-09-05T17:45:27Z",
    "eventSource": "launchwizard.amazonaws.com",
    "eventName": "CreateDeployment",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "192.0.2.0",
    "userAgent": "aws-cli/2.2.17 Python/3.8.8 Darwin/21.6.0 exe/x86_64 prompt/off
 command/example",
    "errorCode": "InternalServerException",
    "requestParameters": {
        "workloadName": "SAP",
        "name": "Example",
        "specifications": "Example",
        "deploymentPatternName": "SapHanaSingle"
    },
    "responseElements": $null,
    "requestID": "86168559-75e9-11e4-8cf8-75d18EXAMPLE",
    "eventID": "832b82d5-d474-44e8-a51d-093ccEXAMPLE",
    "readOnly": false,
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "recipientAccountId": "123456789012",
    "eventCategory": "Management"
}
```

Example: DeleteDeployment

The following example shows a CloudTrail log entry that demonstrates the `DeleteDeployment` operation.

```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "AssumedRole",
        "principalId": "AIDAQRSTUVWXYZEXAMPLE:ExampleAssumedRoleSessionName",
        "arn": "arn:aws:sts::123456789012:assumed-role/ExampleAssumedRole/
ExampleRoleSessionName",
        "accountId": "123456789012",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "sessionContext": {
            "sessionIssuer": {
                "type": "Role",
                "principalId": "AIDAQRSTUVWXYZEXAMPLE",
                "arn": "arn:aws:iam::123456789012:role/ExampleAssumedRole",
                "accountId": "123456789012",
                "userName": "ExampleAssumedRole"
            },
            "webIdFederationData": {},
            "attributes": {
                "creationDate": "2023-09-05T17:45:15Z",
                "mfaAuthenticated": "false"
            }
        }
    },
    "eventTime": "2023-09-06T16:42:53Z",
    "eventSource": "launchwizard.amazonaws.com",
    "eventName": "DeleteDeployment",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "192.0.2.0",
    "userAgent": "aws-cli/2.2.17 Python/3.8.8 Darwin/21.6.0 exe/x86_64 prompt/off
 command/lw.delete-deployment",
    "errorCode": "ValidationException",
    "requestParameters": {
        "deploymentId": "DeploymentIdExample"
    },
    "responseElements": {
        "message": "Example Message."
    },
    "requestID": "86168559-75e9-11e4-8cf8-75d18EXAMPLE",
    "eventID": "832b82d5-d474-44e8-a51d-093ccEXAMPLE",
    "readOnly": false,
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "recipientAccountId": "123456789012",
    "eventCategory": "Management"
}
```

Example: GetDeployment

The following example shows a CloudTrail log entry that demonstrates the `GetDeployment` operation.

```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "AssumedRole",
        "principalId": "AIDAQRSTUVWXYZEXAMPLE:ExampleAssumedRoleSessionName",
        "arn": "arn:aws:sts::123456789012:assumed-role/ExampleAssumedRole/
ExampleRoleSessionName",
        "accountId": "123456789012",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "sessionContext": {
            "sessionIssuer": {
                "type": "Role",
                "principalId": "AIDAQRSTUVWXYZEXAMPLE",
                "arn": "arn:aws:iam::123456789012:role/ExampleAssumedRole",
                "accountId": "123456789012",
                "userName": "ExampleAssumedRole"
            },
            "webIdFederationData": {},
            "attributes": {
                "creationDate": "2023-09-05T17:45:15Z",
                "mfaAuthenticated": "false"
            }
        }
    },
    "eventTime": "2023-09-06T03:39:02Z",
    "eventSource": "launchwizard.amazonaws.com",
    "eventName": "GetDeployment",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "192.0.2.0",
    "userAgent": "aws-cli/2.2.17 Python/3.8.8 Darwin/21.6.0 exe/x86_64 prompt/off
 command/example",
    "requestParameters": {
        "deploymentId": "DeploymentIdExample"
    },
    "responseElements": null,
    "requestID": "86168559-75e9-11e4-8cf8-75d18EXAMPLE",
    "eventID": "832b82d5-d474-44e8-a51d-093ccEXAMPLE",
    "readOnly": true,
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "recipientAccountId": "123456789012",
    "eventCategory": "Management"
}
```

Example: GetWorkload

The following example shows a CloudTrail log entry that demonstrates the `GetWorkload` operation.

```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "AssumedRole",
        "principalId": "AIDAQRSTUVWXYZEXAMPLE:ExampleAssumedRoleSessionName",
        "arn": "arn:aws:sts::123456789012:assumed-role/ExampleAssumedRole/
ExampleRoleSessionName",
        "accountId": "123456789012",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "sessionContext": {
            "sessionIssuer": {
                "type": "Role",
                "principalId": "AIDAQRSTUVWXYZEXAMPLE",
                "arn": "arn:aws:iam::123456789012:role/ExampleAssumedRole",
                "accountId": "123456789012",
                "userName": "ExampleAssumedRole"
            },
            "webIdFederationData": {},
            "attributes": {
                "creationDate": "2023-09-05T17:45:15Z",
                "mfaAuthenticated": "false"
            }
        }
    },
    "eventTime": "2023-09-06T03:59:32Z",
    "eventSource": "launchwizard.amazonaws.com",
    "eventName": "GetWorkload",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "192.0.2.0",
    "userAgent": "aws-cli/2.2.17 Python/3.8.8 Darwin/21.6.0 exe/x86_64 prompt/off
 command/example",
    "requestParameters": {
        "workloadName": "SAP"
    },
    "responseElements": null,
    "requestID": "86168559-75e9-11e4-8cf8-75d18EXAMPLE",
    "eventID": "832b82d5-d474-44e8-a51d-093ccEXAMPLE",
    "readOnly": true,
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "recipientAccountId": "123456789012",
    "eventCategory": "Management"
}
```

Example: ListDeploymentEvents

The following example shows a CloudTrail log entry that demonstrates the `ListDeploymentEvents` operation.

```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "AssumedRole",
        "principalId": "AIDAQRSTUVWXYZEXAMPLE:ExampleAssumedRoleSessionName",
        "arn": "arn:aws:sts::123456789012:assumed-role/ExampleAssumedRole/
ExampleRoleSessionName",
        "accountId": "123456789012",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "sessionContext": {
            "sessionIssuer": {
                "type": "Role",
                "principalId": "AIDAQRSTUVWXYZEXAMPLE",
                "arn": "arn:aws:iam::123456789012:role/ExampleAssumedRole",
                "accountId": "123456789012",
                "userName": "ExampleAssumedRole"
            },
            "webIdFederationData": {},
            "attributes": {
                "creationDate": "2023-09-05T17:45:15Z",
                "mfaAuthenticated": "false"
            }
        }
    },
    "eventTime": "2023-09-06T03:38:02Z",
    "eventSource": "launchwizard.amazonaws.com",
    "eventName": "ListDeploymentEvents",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "192.0.2.0",
    "userAgent": "aws-cli/2.2.17 Python/3.8.8 Darwin/21.6.0 exe/x86_64 prompt/off
 command/example",
    "requestParameters": {
        "deploymentId": "DeploymentIdExample"
    },
    "responseElements": null,
    "requestID": "86168559-75e9-11e4-8cf8-75d18EXAMPLE",
    "eventID": "832b82d5-d474-44e8-a51d-093ccEXAMPLE",
    "readOnly": true,
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "recipientAccountId": "123456789012",
    "eventCategory": "Management"
}
```

Example: ListDeployments

The following example shows a CloudTrail log entry that demonstrates the `ListDeployments` operation.

```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "AssumedRole",
        "principalId": "AIDAQRSTUVWXYZEXAMPLE:ExampleAssumedRoleSessionName",
        "arn": "arn:aws:sts::123456789012:assumed-role/ExampleAssumedRole/
ExampleRoleSessionName",
        "accountId": "123456789012",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "sessionContext": {
            "sessionIssuer": {
                "type": "Role",
                "principalId": "AIDAQRSTUVWXYZEXAMPLE",
                "arn": "arn:aws:iam::123456789012:role/ExampleAssumedRole",
                "accountId": "123456789012",
                "userName": "ExampleAssumedRole"
            },
            "webIdFederationData": {},
            "attributes": {
                "creationDate": "2023-09-05T17:45:15Z",
                "mfaAuthenticated": "false"
            }
        }
    },
    "eventTime": "2023-09-06T03:38:02Z",
    "eventSource": "launchwizard.amazonaws.com",
    "eventName": "ListDeployments",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "192.0.2.0",
    "userAgent": "aws-cli/2.2.17 Python/3.8.8 Darwin/21.6.0 exe/x86_64 prompt/off
 command/example",
    "requestParameters": {
        "maxResults": 100
    },
    "responseElements": null,
    "requestID": "86168559-75e9-11e4-8cf8-75d18EXAMPLE",
    "eventID": "832b82d5-d474-44e8-a51d-093ccEXAMPLE",
    "readOnly": true,
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "recipientAccountId": "123456789012",
    "eventCategory": "Management"
}
```

Example: ListWorkloadDeploymentPattern

The following example shows a CloudTrail log entry that demonstrates the `ListWorkloadDeploymentPattern` operation.

```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "AssumedRole",
        "principalId": "AIDAQRSTUVWXYZEXAMPLE:ExampleAssumedRoleSessionName",
        "arn": "arn:aws:sts::123456789012:assumed-role/ExampleAssumedRole/
ExampleRoleSessionName",
        "accountId": "123456789012",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "sessionContext": {
            "sessionIssuer": {
                "type": "Role",
                "principalId": "AIDAQRSTUVWXYZEXAMPLE",
                "arn": "arn:aws:iam::123456789012:role/ExampleAssumedRole",
                "accountId": "123456789012",
                "userName": "ExampleAssumedRole"
            },
            "webIdFederationData": {},
            "attributes": {
                "creationDate": "2023-09-05T17:45:15Z",
                "mfaAuthenticated": "false"
            }
        }
    },
    "eventTime": "2023-09-06T03:59:32Z",
    "eventSource": "launchwizard.amazonaws.com",
    "eventName": "ListWorkloadDeploymentPatterns",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "192.0.2.0",
    "userAgent": "aws-cli/2.2.17 Python/3.8.8 Darwin/21.6.0 exe/x86_64 prompt/off
 command/example",
    "requestParameters": {
        "workloadName": "SAP",
        "maxResults": 10
    },
    "responseElements": null,
    "requestID": "86168559-75e9-11e4-8cf8-75d18EXAMPLE",
    "eventID": "832b82d5-d474-44e8-a51d-093ccEXAMPLE",
    "readOnly": true,
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "recipientAccountId": "123456789012",
    "eventCategory": "Management"
}
```

Example: ListWorkloads

The following example shows a CloudTrail log entry that demonstrates the `ListWorkloads` operation.

```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "AssumedRole",
        "principalId": "AIDAQRSTUVWXYZEXAMPLE:ExampleAssumedRoleSessionName",
        "arn": "arn:aws:sts::123456789012:assumed-role/ExampleAssumedRole/
ExampleRoleSessionName",
        "accountId": "123456789012",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "sessionContext": {
            "sessionIssuer": {
                "type": "Role",
                "principalId": "AIDAQRSTUVWXYZEXAMPLE",
                "arn": "arn:aws:iam::123456789012:role/ExampleAssumedRole",
                "accountId": "123456789012",
                "userName": "ExampleAssumedRole"
            },
            "webIdFederationData": {},
            "attributes": {
                "creationDate": "2023-09-05T17:45:15Z",
                "mfaAuthenticated": "false"
            }
        }
    },
    "eventTime": "2023-09-06T03:59:32Z",
    "eventSource": "launchwizard.amazonaws.com",
    "eventName": "ListWorkloads",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "192.0.2.0",
    "userAgent": "aws-cli/2.2.17 Python/3.8.8 Darwin/21.6.0 exe/x86_64 prompt/off
 command/example",
    "requestParameters": null,
    "responseElements": null,
    "requestID": "86168559-75e9-11e4-8cf8-75d18EXAMPLE",
    "eventID": "832b82d5-d474-44e8-a51d-093ccEXAMPLE",
    "readOnly": true,
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "recipientAccountId": "123456789012",
    "eventCategory": "Management"
}
```

AWS Launch Wizard documentation history

The following table describes the documentation for this release of AWS Launch Wizard.

Change	Description	Date
Launch Wizard supports deploying additional workloads with API operations	AWS Launch Wizard now supports deploying additional workload deployments with the CreateDeployment API operation. For more information, see the **Get started** section for your desired workload.	June 12, 2024
Launch Wizard for SAP supports SAP Web Dispatcher	AWS Launch Wizard for SAP now supports SAP Web Dispatcher as an optional component for NetWeaver stack on HANA deployments.	April 30, 2024
Launch Wizard for SAP supports new application software	AWS Launch Wizard for SAP now supports S/4 HANA 2023 and S/4 HANA Foundations 2023.	March 11, 2024
Launch Wizard for SAP available in additional Regions	AWS Launch Wizard for SAP is now available in the Asia Pacific (Melbourne), Europe (Spain), and Europe (Zurich) Regions.	January 26, 2024
AWS Launch Wizard for SAP deployments with SAP ASE database	You can now deploy AWS Launch Wizard for SAP systems with SAP ASE database.	December 22, 2023

AWS Launch Wizard APIs are available	AWS Launch Wizard APIs are now available for creating SAP deployments. You can also list details about existing deployments using new Launch Wizard API operation s. For more information, see Deploying an SAP application (AWS CLI).	November 8, 2023
AWS managed policy updates - Deprecated an existing policy and added a new policy	AWS Launch Wizard deprecated an existing AWS managed policy and added a new AWS managed policy.	September 1, 2023
AWS managed policy updates - Update to an existing policy	AWS Launch Wizard updated an existing AWS managed policy.	February 15, 2023
AWS managed policy updates - Update to an existing policy	AWS Launch Wizard updated an existing AWS managed policy.	January 6, 2023
Proxy server example	You can deploy an SAP application with AWS Launch Wizard using a proxy server such as, Squid.	August 12, 2022
AWS Launch Wizard for SAP support for cloning deployments	You can now clone your SAP deployments created after April 21, 2022.	April 21, 2022
AWS Launch Wizard for Remote Desktop Gateway	You can set up a new Remote Desktop Gateway infrastru cture to an existing AWS infrastructure using AWS Launch Wizard for Remote Desktop Gateway.	December 8, 2021

AWS Launch Wizard for SAP integration with AWS Service Catalog	You can create AWS Service Catalog products from successful deployments with AWS Launch Wizard.	August 30, 2021
AWS Launch Wizard for SAP support for no rollback on failure	When you select "No rollback on failure" for your AWS Launch Wizard deployments, if a deployment fails, Launch Wizard does not delete the AWS resources that were created for the deployment.	March 5, 2021
AWS Launch Wizard for Active Directory support for no rollback on failure	When you select "No rollback on failure" for your AWS Launch Wizard deployments, if a deployment fails, Launch Wizard does not delete the AWS resources that were created for the deployment.	March 5, 2021
AWS Launch Wizard SQL support for no rollback on failure	When you select "No rollback on failure" for your AWS Launch Wizard deployments, if a deployment fails, Launch Wizard does not delete the AWS resources that were created for the deployment.	March 5, 2021
AWS Launch Wizard for SAP support for custom IP address specification	You can specify a private IP address for each Amazon EC2 instance in your SAP deployment.	February 26, 2021

SUSE/RHEL high availability for SAP applications	You can configure SUSE/RHEL high availability for SAP applications as part of your deployment with AWS Launch Wizard.	February 1, 2021
AWS Launch Wizard for SAP support for SAP application installation	You can install supported SAP applications using customer-provided SAP software.	December 16, 2020
AWS Launch Wizard for SQL integration with AWS Systems Manager Application Manager.	You can manage resources created by Launch Wizard for SQL from the Systems Manager Application Manager console.	December 15, 2020
AWS Launch Wizard for Active Directory	You can set up a new Active Directory infrastructure or add domain controllers to an existing AWS infrastructure using AWS Launch Wizard for Active Directory.	December 15, 2020
AWS Launch Wizard for SAP support for custom pre-deployment and post-deployment scripts	You can run custom pre- and post-deployment configuration scripts using AWS Launch Wizard for SAP.	November 17, 2020
AWS Launch Wizard support for SQL Server application single-node deployments.	You can deploy your SQL Server application on a single instance.	October 28, 2020
AWS Launch Wizard for SAP support for application single-node deployments	You can deploy your SAP application on a single instance.	October 15, 2020

Route 53/DNS association support	You can provide your DNS domain name or Route53 hosted zone to enable DNS association for your deployed EC2 instances.	June 18, 2020
AWS Launch Wizard for SQL Server integration with CloudWatch Application Insights	You can set up monitorin g for your application with CloudWatch Application Insights.	June 18, 2020
SQL Server witness node support.	You can add a witness node to your SQL Server Always On configuration.	May 11, 2020
Proxy server support	You can route outbound internet traffic for deployed EC2 instances through a proxy server.	May 11, 2020
Initial release	Initial release of AWS Launch Wizard for SAP User Guide.	April 8, 2020
Initial release	Initial release of the AWS Launch Wizard for SQL Server User Guide.	November 14, 2019