

Best Practices for Deploying Amazon WorkSpaces

Network Access, Directory Services, and Security

July 2016



© 2016, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Notices

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

Contents

| | |
|---|----|
| Abstract | 4 |
| Introduction | 4 |
| WorkSpaces Requirements | 5 |
| Network Considerations | 6 |
| VPC Design | 7 |
| Traffic Flow | 8 |
| Example of a Typical Configuration | 12 |
| AWS Directory Service | 17 |
| AD DS Deployment Scenarios | 17 |
| Design Considerations | 27 |
| Multi-Factor Authentication (MFA) | 32 |
| Security | 34 |
| Encryption in Transit | 34 |
| Network Interfaces | 36 |
| WorkSpaces Security Group | 36 |
| Encrypted WorkSpaces | 38 |
| Monitoring or Logging Using Amazon CloudWatch | 41 |
| Amazon CloudWatch Metrics for WorkSpaces | 41 |
| Troubleshooting | 43 |
| AD Connector Cannot Connect to Active Directory | 43 |
| How to Check Latency to Closest AWS Region | 44 |
| Conclusion | 44 |
| Contributors | 44 |
| Further Reading | 45 |

Abstract

This whitepaper outlines a set of best practices for the deployment of Amazon WorkSpaces. The paper covers network considerations, directory services and user authentication, security, and monitoring and logging.

The document is broken into four categories to enable quicker access to relevant information. This document is intended for a network engineer, directory engineer, or security engineer.

Introduction

Amazon WorkSpaces is a managed desktop computing service in the cloud. Amazon WorkSpaces removes the burden of procuring or deploying hardware or installing complex software, and delivers a desktop experience with either a few clicks on the AWS Management Console, using the AWS command line interface (CLI), or by using the APIs. With Amazon WorkSpaces, you can launch a desktop within minutes, and connect to and access your desktop software from on-premises or an external network securely, reliably, and quickly. You can:

- Leverage your existing on-premises Microsoft Active Directory (AD) by using [AWS Directory Service: AD Connector](#).
- Extend your directory to the AWS Cloud.
- Build a managed directory with AWS Directory Service: Microsoft AD or Simple AD, to manage your users and WorkSpaces.

In addition, you can leverage your on-premises or cloud-hosted RADIUS server with AD Connector to provide multi-factor authentication (MFA) to your WorkSpaces.

You can automate provisioning of Amazon WorkSpaces by using the CLI or API, which enables you to integrate Amazon WorkSpaces into your existing provisioning workflows.

For security, in addition to the integrated network encryption that the WorkSpaces service provides, you can also enable encryption at rest for your WorkSpaces (see [Encrypted WorkSpaces](#) in the security section).

You can deploy applications to your WorkSpaces by using your existing on-premises tools, such as Microsoft System Center Configuration Manager (SCCM), or by leveraging the [Amazon WorkSpaces Application Manager](#) (Amazon WAM).

The following sections provide details about Amazon WorkSpaces, explain how the service works, describe what you need to launch the service, and let you know what options and features are available for you to use.

WorkSpaces Requirements

The Amazon WorkSpaces service requires three components to deploy successfully:

- **WorkSpaces client application.** An Amazon WorkSpaces-supported client device. Find a full list here: [Supported Platforms and Devices](#).

You can also use Personal Computer over Internet Protocol (PCoIP) zero clients to connect to WorkSpaces. For a list of available devices, see [PCoIP Zero Clients for Amazon WorkSpaces](#).

- **A directory service to authenticate users and provide access to their WorkSpace.** Amazon WorkSpaces currently works with AWS Directory Service and Active Directory. You can use your on-premises Active Directory server with AWS Directory Service to support your existing enterprise user credentials with WorkSpaces.
- **Amazon Virtual Private Cloud (Amazon VPC) in which to run your Amazon WorkSpaces.** You'll need a minimum of two subnets for a WorkSpaces deployment because each AWS Directory Service construct requires two subnets in a Multi-AZ deployment.

Network Considerations

Each WorkSpace is associated with a specific Amazon VPC and AWS Directory Service construct you used to create it. All AWS Directory Service constructs (Simple AD, AD Connector, and Microsoft AD) require two subnets to operate, each in different Availability Zones. Subnets are permanently affiliated with a Directory Service construct and can't be modified after an AWS Directory Service is created. Therefore, it's imperative that you determine the right subnet sizes before you create the Directory Services construct. Carefully consider the following before you create the subnets:

- How many WorkSpaces will you need over time? What is the expected growth?
- What types of users will you need to accommodate?
- How many Active Directory Domains will you connect?
- Where do your Enterprise User Accounts reside?

Amazon recommends defining user groups, or personas, based on the type of access and the user authentication you require as part of your planning process. These answers are helpful when you need to limit access to certain applications or resources. Defined user personas can help you segment and restrict access using AWS Directory Service, network access control lists, routing tables, and VPC security groups. Each AWS Directory Service construct uses two subnets and applies the same settings to all WorkSpaces that launch from that construct. For example, you can use a security group that applies to all WorkSpaces attached to an AD Connector to specify whether MFA authentication is required, or whether the end user can have local administrator access on their WorkSpace.

Note Each AD Connector connects to one Microsoft Active Directory organizational unit (OU). You must construct your Directory Service to take your user personas into consideration so that you can take advantage of this capability.

This section describes best practices for sizing your VPC and subnets, traffic flow, and implications for directory services design.

VPC Design

Here are a few things to consider when designing the VPC, subnets, security groups, routing policies, and network ACLs for your Amazon WorkSpaces so that you can build your WorkSpaces environment for scale, security, and ease of management:

- **VPC.** We recommend using a separate VPC specifically for your WorkSpaces deployment. With a separate VPC, you can specify the necessary governance and security guardrails for your WorkSpaces by creating traffic separation.
- **Directory Services.** Each AWS Directory Service construct requires a pair of subnets that provide for a highly available directory service split between Amazon AZs.
- **Subnet size.** WorkSpaces deployments are tied to a directory construct and reside in the same VPC subnets as your chosen AWS Directory Service. A few considerations:
 - Subnet sizes are permanent and cannot change; you should leave ample room for future growth.
 - You can specify a default security group for your chosen AWS Directory Service; the security group applies to all WorkSpaces that are associated with the specific AWS Directory Service construct.
 - You can have multiple AWS Directory Services use the same subnet.

Consider future plans when you design your VPC. For example, you might want to add management components such as an antivirus server, a patch management server, or an Active Directory or RADIUS MFA server. It's worth planning for additional available IP addresses in your VPC design to accommodate such requirements.

For in-depth guidance and considerations for VPC design and subnet sizing, see the **re:Invent** presentation [How Amazon.com is Moving to Amazon WorkSpaces](#).

Network Interfaces

Each WorkSpace has two elastic network interfaces (ENIs), a management network interface (eth0), and a primary network interface (eth1). AWS uses the management network interface to manage the WorkSpace; it's the interface on which your client connection terminates. AWS leverages a private IP address range for this interface. For network routing to work properly, you can't use this private address space on any network that can communicate with your WorkSpaces VPC.

For a list of the private IP ranges that we use on a per region basis, see [Amazon WorkSpaces Details](#).

Note Amazon WorkSpaces and their associated management network interfaces don't reside in your VPC, and you can't view the management network interface or the Amazon Elastic Compute Cloud (Amazon EC2) instance ID in your AWS Management Console (see Figure 4, Figure 5, and Figure 6). However, you can view and modify the security group settings of your primary network interface (eth1) in the AWS Management Console. Also, the primary network interface of each WorkSpace does count toward your ENI Amazon EC2 resource limits. For large deployments of WorkSpaces, you would need to open a support ticket via the AWS Management Console to increase your ENI limits.

Traffic Flow

You can break down Amazon WorkSpaces traffic into two main components:

- The traffic between the client device and the Amazon WorkSpace service
- The traffic between the Amazon WorkSpace service and customer network traffic

In the next section, we discuss both of these components.

Client Device to WorkSpace

The device running the Amazon WorkSpaces client, regardless of its location (on-premises or remote), will use the same two ports for connectivity to the WorkSpaces service. The client uses https over port 443 for all authentication and session-related information, and it uses port 4172 (PCoIP port) with both TCP and UDP for pixel streaming to a given WorkSpace and for network health checks. Traffic on both ports is encrypted. Port 443 traffic is used for authentication and session information and leverages TLS for encrypting the traffic. Pixel streaming traffic leverages AES-256-bit encryption for communication between the client and etho of the WorkSpace, via the streaming gateway. More information can be found in the [Security](#) section, later in this document.

We publish per-region IP ranges of our PCoIP streaming gateways and network health check endpoints. You can limit outbound traffic on port 4172 from your corporate network to the AWS streaming gateway and network health check endpoints by allowing only outbound traffic on port 4172 to the specific AWS regions in which you're using Amazon WorkSpaces. For the IP ranges and network health check endpoints, see [Amazon WorkSpaces PCoIP Gateway IP Ranges](#).

The Amazon WorkSpaces client has a built-in network status check. This utility shows users whether their network will support a connection by way of a status indicator on the bottom right of the application. A more detailed view of the network status can be accessed by selecting **Network** on the bottom-right side of the client, the result of which is shown in Figure 1.

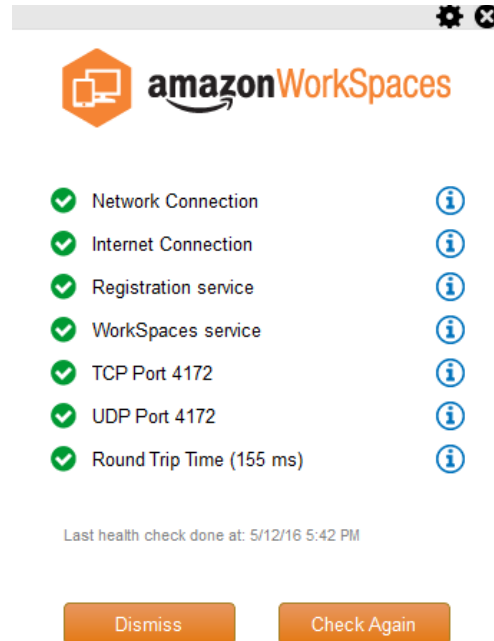


Figure 1: WorkSpaces client – network check

A user initiates a connection from his or her client to the WorkSpaces service by supplying his or her login information for the directory used by the Directory Service construct, typically your corporate directory. The login information is sent via https to the authentication gateways of the Amazon WorkSpaces service in the region in which the WorkSpace is located. The authentication gateway of the Amazon WorkSpaces service then forwards the traffic to the specific AWS Directory Service service construct associated with your WorkSpace. For example, when using the AD Connector, the AD Connector forwards the authentication request directly to your Active Directory service, which could be on-premises or in an AWS VPC (see AD DS Deployment Scenarios). The AD Connector does not store any authentication information and acts as a stateless proxy. As a result, it's imperative that the AD Connector has connectivity to an Active Directory server. The AD Connector determines the Active Directory server that it is connecting to by using the DNS servers that you define when you create the AD Connector.

If you're using an AD Connector and you have MFA enabled on the directory, the MFA token will be checked before the directory service authentication. Should the MFA validation fail, the user's login information will not be forwarded to your AWS Directory Service.

Once a user is authenticated, the streaming traffic starts by leveraging port 4172 (PCoIP port) through the AWS streaming gateway to the WorkSpace. Session-related information is still exchanged via https throughout the session. The streaming traffic leverages the first ENI on the WorkSpace (eth0 on the WorkSpace) that is not connected to your VPC. The network connection from the streaming gateway to the ENI is managed by AWS. In the event of a connection failure from the streaming gateways to the WorkSpaces streaming ENI, a CloudWatch event is generated (see [Monitoring or Logging Using Amazon CloudWatch](#) section of this whitepaper).

The amount of data that is sent between the Amazon WorkSpaces service and the client depends on the level of pixel activity. To ensure an optimal experience for users, we recommend that the round trip time (RTT) between the WorkSpaces client and the AWS Region where your WorkSpaces are located is less than 100 ms. Typically this means your WorkSpaces client is located less than two thousand miles from the Region in which the WorkSpace is being hosted. We provide a [Connection Health Check](#) webpage that you can refer to in order to determine the most optimal AWS region to connect to for the Amazon WorkSpaces service.

Amazon WorkSpaces Service to VPC

After a connection is authenticated from a client to a WorkSpace and streaming traffic is initiated, your WorkSpaces client will display a Windows desktop (your WorkSpace) that is connected to your VPC, and your network should show that you have established that connection. The WorkSpace's primary ENI, identified as eth1, will have an IP address assigned to it from the Dynamic Host Configuration Protocol (DHCP) service that is provided by your VPC, typically from the same subnets as your AWS Directory Service. The IP address stays with the WorkSpace for the duration of the life of the WorkSpace. The ENI that is in your VPC has access to any resource in the VPC and to any network that you have connected to your VPC (via a VPC peering, an AWS Direct Connect connection, or VPN connection).

ENI access to your network resources is determined by the default security group (see more on security groups [here](#)) that your AWS Directory Service configures for each WorkSpace and any additional security groups that you assign to the ENI. You can add security groups to the ENI facing your VPC at will by leveraging the AWS Management Console or CLI. In addition to security groups, you can use

your preferred host-based firewall on a given WorkSpace to limit network access to resources within the VPC.

Figure 4 in AD DS Deployment Scenarios, later in this whitepaper, shows the traffic flow described earlier.

Example of a Typical Configuration

Let's consider a scenario where you have two types of users and that your AWS Directory Service uses a centralized Active Directory for user authentication:

- **Workers who need full access from anywhere** (for example, full-time employees). These users will have full access to the Internet and the internal network, and they will pass through a firewall from the VPC to the on-premises network.
- **Workers who should have only restricted access from inside the corporate network** (for example, contractors and consultants). These users have restricted Internet access through a proxy server (to specific websites) in the VPC, and will have limited network access in the VPC and to the on-premises network.

You'd like to give full-time employees the ability to have local administrator access on their WorkSpace to install software and you would like to enforce two-factor authentication with MFA. You also want to allow full-time employees to access the Internet unabated from their WorkSpace.

For contractors, you want to block local admin access so that they can use only specific pre-installed applications. You want to apply very restrictive network access controls via security groups for these WorkSpaces. You need to open port 80 and 443 to specific internal websites only, and you would like to block their access to the Internet.

In this scenario, there are two completely different types of user personas with different requirements for network and desktop access. It's a best practice to manage and configure their WorkSpaces differently. To do so, you'll need to create two AD Connectors, one for each user persona. Each AD Connector requires two subnets that need enough IP addresses to meet your WorkSpaces usage growth estimates.

Note Each AWS VPC subnet consumes five IP addresses (the first four and the last IP address) for management purposes and each AD Connector consumes one IP address in each subnet in which it persists.

Further considerations for this scenario are as follows:

- AWS VPC subnets should be private subnets, so that traffic such as Internet access can be controlled through either a NAT Gateway, Proxy-NAT server in the cloud, or routed back through your on-premises traffic management system.
- A firewall is in place for all VPC traffic bound for the on-premises network.
- Microsoft Active Directory server and the MFA RADIUS servers are either on-premises (see Scenario 1: Using AD Connector to Proxy Authentication to On-Premises AD DS) or part of the AWS Cloud implementation (see Scenarios 2 and 3, AD DS Deployment Scenarios).

Given that all WorkSpaces will be granted some form of Internet access, and given that they will be hosted in a private subnet, you also need to create public subnets that can access the Internet through an Internet gateway. You will need a NAT gateway for the full-time employees allowing them to access the Internet, and a Proxy-NAT server for the consultants and contractors to limit their access to specific internal websites. To plan for failure, design for high availability, and limit cross-AZ traffic charges, you should have two NAT gateways and NAT or proxy servers in two different subnets in a Multi-AZ deployment. The two AZs that you select as public subnets will match the two AZs that you use for your WorkSpaces subnets in regions that have more than two AZs. You can route all traffic from each WorkSpaces AZ to the corresponding public subnet to limit cross-AZ traffic charges and provide easier management. Figure 2 shows the VPC configuration.

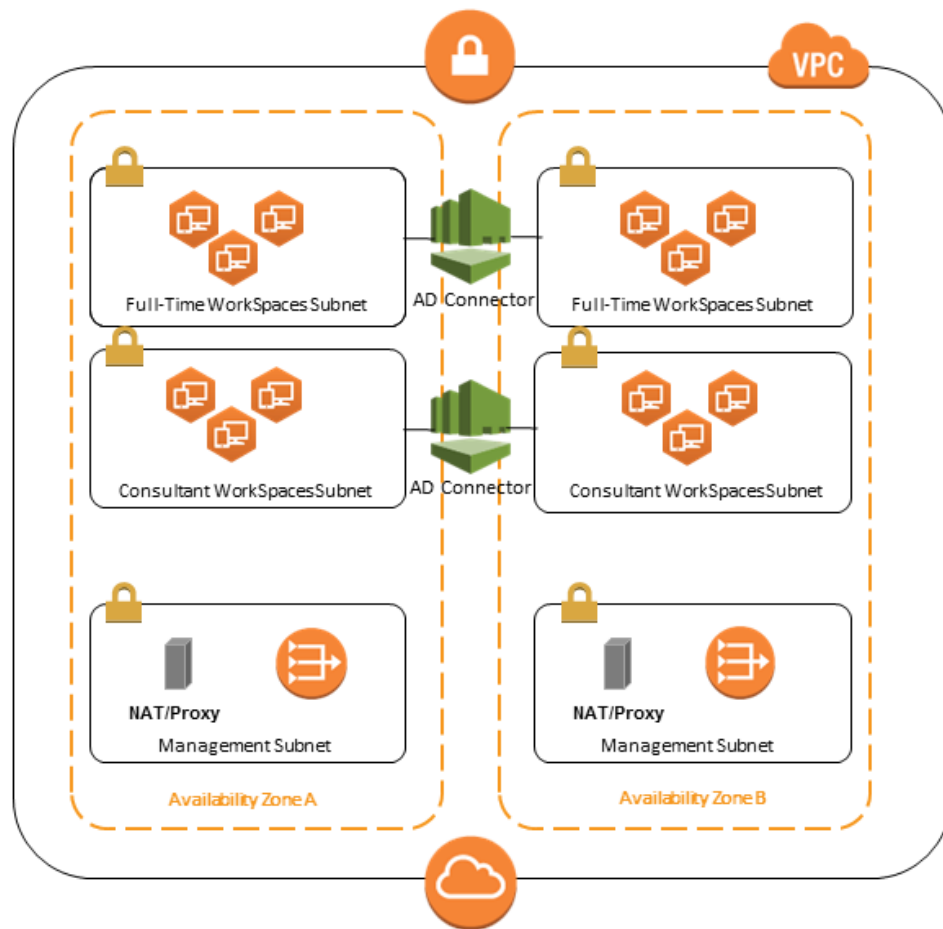


Figure 2: High-level VPC design

The following information describes how to configure the two different WorkSpaces types described earlier.

- Full-time employees:** In the Amazon WorkSpaces Management Console, select the **Directories** option on the menu bar, select the directory that hosts your full-time employees, and then select **Local Administrator Setting**. By enabling this option, any newly created WorkSpace will have local administrator privileges. To grant Internet access, you should configure Network Address Translation (NAT) for outbound Internet access from your VPC. To enable MFA, you need to specify a RADIUS server, sever IPs, ports, and preshared key.

For full-time employees' WorkSpaces, in-bound traffic to the WorkSpace

would be limited to Remote Desktop Protocol (RDP) from the Helpdesk subnet by applying a default security group via the AD Connector settings.

- **Contractors and consultants:** In the Amazon WorkSpaces Management Console, disable **Internet Access** and the **Local Administrator Setting**. Then add a security group under the **Security Group** settings section to enforce a security group for all new WorkSpaces created under that directory.

For consultants' WorkSpaces, limit outbound and inbound traffic to the WorkSpaces by applying a default Security group via the AD Connector settings to all WorkSpaces associated with the AD Connector. The security group would prevent outbound access from the WorkSpaces to anything other than HTTP and HTTPS traffic and inbound traffic to RDP from the Helpdesk subnet in the on-premises network.

Note The security group applies only to the ENI that is in the VPC (eth1 on the Workspace), and access to the Workspace from the WorkSpaces client is not restricted as a result of a security group. Figure 3 shows the final WorkSpaces VPC design described earlier.

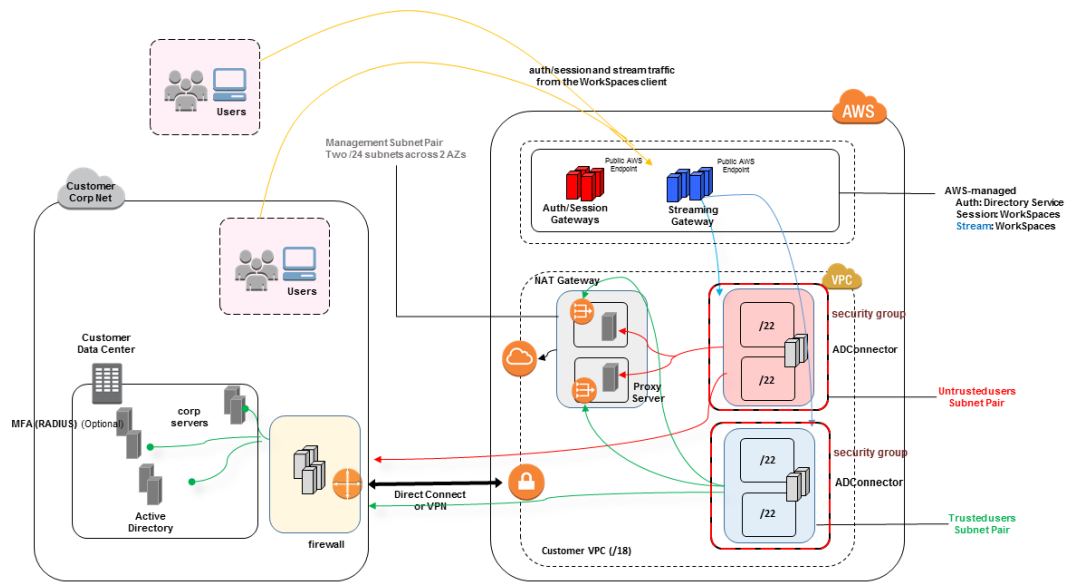


Figure 3: WorkSpaces design with user personas

AWS Directory Service

As mentioned in the Introduction, Amazon WorkSpaces is underpinned by AWS Directory Service. With AWS Directory Service you can create three types of directories. The first two live in the AWS Cloud:

- AWS Directory Service for Microsoft Active Directory (Enterprise Edition), or **Microsoft AD**, which is a managed Microsoft Active Directory, powered by Windows Server 2012 R2.
- **Simple AD**, a standalone, Microsoft Active Directory-compatible, managed directory service powered by Samba 4.

The third, **AD Connector**, is a directory gateway that allows you to proxy authentication requests and user or group lookups to your existing on-premises Microsoft Active Directory.

The following section describes communication flows for authentication between the Amazon WorkSpaces brokerage service and AWS Directory Service, best practices for implementing WorkSpaces with AWS Directory Service, and advanced concepts such as MFA. We also discuss infrastructure architecture concepts for Amazon WorkSpaces at scale, requirements on Amazon VPC, and AWS Directory Service, including integration with on-premises Microsoft Active Directory Domain Services (AD DS).

AD DS Deployment Scenarios

Underpinning Amazon WorkSpaces is the AWS Directory Service, and correct design and deployment of the directory service is critical. The following three scenarios build upon the *Microsoft Active Directory Domain Services [quick start guide](#)*, detailing the best practice deployment options for AD DS, specifically for integration with WorkSpaces. The *Design Considerations* section of this chapter goes into the specific requirements and best practices of using AD Connector for WorkSpaces, which is an integral part of the overall WorkSpaces design concept.

- **Scenario 1: Using AD Connector to proxy authentication to on-premises AD DS.** In this scenario network connectivity (VPN/Direct Connect (DX)) is in place to the customer, with all authentication proxied

via AWS Directory Service (AD Connector) to the customer on-premises AD DS.

- **Scenario 2: Extending on-premises AD DS into AWS (Replica).** This scenario is similar to scenario 1, but here a replica of the customer AD DS is deployed on AWS in combination with AD Connector, reducing latency of authentication/query requests to AD DS and the AD DS global catalog.
- **Scenario 3: Standalone isolated deployment using AWS Directory Service in the AWS Cloud.** This is an isolated scenario and doesn't include connectivity back to the customer for authentication. This approach uses AWS Directory Service (Microsoft AD) and AD Connector. Although this scenario doesn't rely on connectivity to the customer for authentication, it does make provision for application traffic where required over VPN or DX.

Scenario 1: Using AD Connector to Proxy Authentication to On-Premises AD DS

This scenario is for customers who don't want to extend their on-premises AD DS into AWS or where a new deployment of AD DS is not an option. Figure 4: AD Connector to on-premises Active Directory depicts at a high level each of the components and shows the user authentication flow.

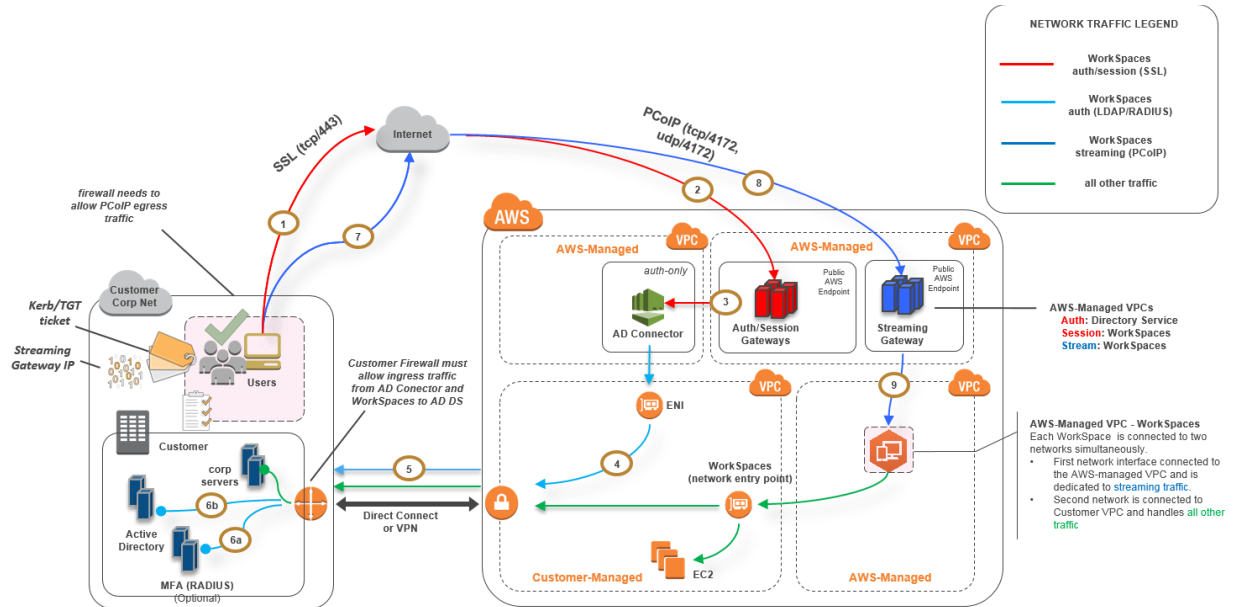


Figure 4: AD Connector to on-premises Active Directory

In this scenario AWS Directory Service (AD Connector) is used for all user or MFA authentication that is proxied through the AD Connector to the customer on-premises AD DS (Figure 5). For details on the protocols or encryption used for the authentication process, see the [Security](#) section of this whitepaper.

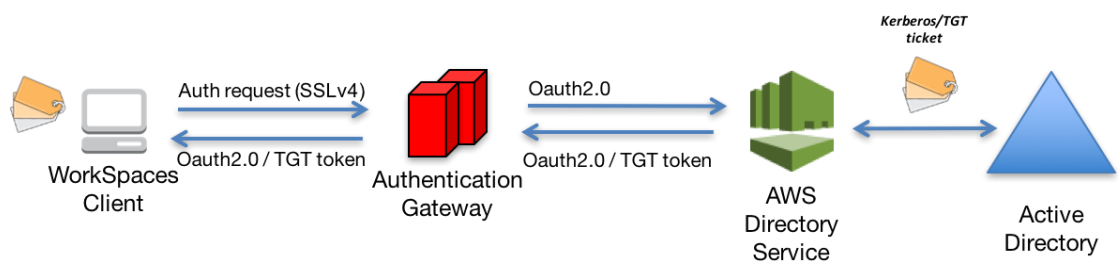


Figure 5: User Authentication via the Authentication Gateway

Scenario 1 shows a hybrid architecture where the customer may already have resources in AWS, as well as resources in an on-premises datacenter that could be accessed via WorkSpaces. The customer can leverage their existing on-premises AD DS and RADIUS servers for user and MFA authentication.

This architecture uses the following components or construct.

Amazon Web Services:

- **Amazon VPC:** Creation of an Amazon VPC with at least two private subnets across two Availability Zones.
- **DHCP Options Set:** Creation of an Amazon VPC DHCP Options Set. This allows customer-specified domain name and domain name servers (DNS) (on-premises services) to be defined. (For more information, see [DHCP Options Sets](#).)
- **Amazon virtual private gateway:** Enable communication with your own network over an IPsec VPN tunnel or an AWS Direct Connect connection.
- **AWS Directory Service:** AD Connector is deployed into a pair of Amazon VPC private subnets.
- **Amazon WorkSpaces:** WorkSpaces are deployed into the same private subnets as the AD Connector (see Design Considerations, AD Connector).

Customer:

- **Network connectivity:** corporate VPN or Direct Connect endpoints.
- **AD DS:** corporate AD DS.
- **MFA (optional):** corporate RADIUS server.
- **End user devices:** Corporate or BYOL end user devices (such as Windows, Mac, iPad or Android tablets, zero clients, Chromebook), used to access the Amazon WorkSpaces service (see [Supported Platforms and Devices](#)).

Although this solution is great for customers who don't want to deploy AD DS into the cloud, it does come with its pitfalls.

- **Reliance on connectivity:** If connectivity to the data center is lost, no user will be able to log in to their respective WorkSpaces, and existing connections will remain active for the Kerberos/TGT lifetime.
- **Latency:** If latency exists via the connection (this is more the case with VPN than DX), then WorkSpaces authentication and any AD DS-related activity, such as Group Policy (GPO) enforcement, will take more time.

- **Traffic costs:** All authentication must traverse the VPN or DX link, and so it depends on the connection type. This is either Data Transfer OUT From Amazon EC2 To Internet or Data Transfer Out (DX).

Note AD Connector is a proxy service. It doesn't store or cache user credentials. Instead, all authentication, lookup, and management requests are handled by your Active Directory. An account with delegation privileges is required in your directory service with rights to read all user information and join a computer to the domain.

For details about how to configure a user in your directory for AD Connector, see [Delegating Connect Privileges](#).

In general, the WorkSpaces experience is highly dependent on item 5 shown in Figure 4.

Scenario 2: Extending On-Premises AD DS into AWS (Replica)

This scenario is similar to scenario 1, however, in scenario 2 a replica of the customer AD DS is deployed on AWS in combination with AD Connector. This reduces latency of authentication or query requests to AD DS. Figure 6 shows a high-level view of each of the components and the user authentication flow.

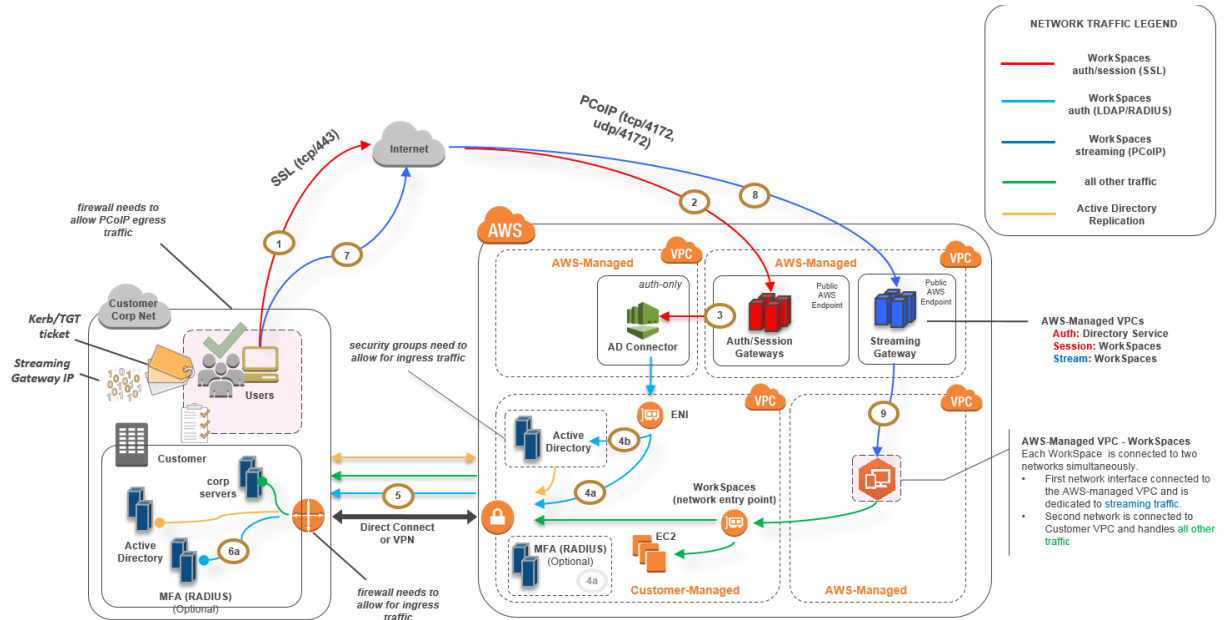


Figure 6: Extend customer Active Directory Domain to the cloud

As in scenario 1, AD Connector is used for all user or MFA authentication, which in turn is proxied to the customer AD DS (Figure 5). In scenario 2, the customer AD DS is deployed across Availability Zones on Amazon EC2 instances that are promoted to be domain controllers in the customer’s on-premises Active Directory forest, running in the AWS Cloud. Each domain controller is deployed into VPC private subnets to make AD DS highly available in the AWS Cloud. For best practices for deploying AD DS in the AWS Cloud, see Design Considerations later in this whitepaper.

Once WorkSpaces instances are deployed, they have access to the cloud-based domain controllers for secure, low-latency directory services and DNS. All network traffic, including AD DS communication, authentication requests, and Active Directory replication is secured either within the private subnets or across the customer VPN tunnel or DX.

This architecture uses the following components or construct.

Amazon Web Services:

- **Amazon VPC:** Creation of an Amazon VPC with at least four private subnets across two Availability Zones (two for the customer AD DS, two for AD Connector or WorkSpaces).
- **DHCP Options Set:** Creation of an Amazon VPC DHCP Options Set. This allows you to define a customer-specified domain name and DNSs (AD DS local). For more information, see [DHCP Options Sets](#).
- **Amazon virtual private gateway:** Enable communication with your own network over an IPsec VPN tunnel or AWS Direct Connect connection.
- **Amazon EC2:**
 - Customer corporate AD DS domain controllers deployed on Amazon EC2 instances in dedicated private VPC subnets.
 - Customer “optional” RADIUS servers for MFA.
- **AWS Directory Services:** AD Connector is deployed into a pair of Amazon VPC private subnets.
- **Amazon WorkSpaces:** WorkSpaces are deployed into the same private subnets as the AD Connector (see Design Considerations, AD Connector).

Customer:

- **Network connectivity:** Corporate VPN or AWS Direct Connect endpoints.
- **AD DS:** Corporate AD DS (required for replication).
- **MFA “optional”:** Corporate RADIUS server.
- **End user devices:** Corporate or BYOL end user devices (such as Windows, Mac, iPad or Android tablets, zero clients, Chromebook), used to access the Amazon WorkSpaces service (see [Supported Platforms and Devices](#)).

Unlike scenario 1, this solution doesn't have the same pitfalls. Therefore, WorkSpaces and AWS Directory Service have no reliance on the connectivity in place.

- **Reliance on connectivity:** If connectivity to the customer data center is lost, end users can continue to work because authentication and “optional” MFA are processed locally.
- **Latency:** With the exception of replication traffic (see *Design Considerations: AD DS Sites and Services*), all authentication is local and low latency.
- **Traffic costs:** In this scenario, authentication is local, with only AD DS replication having to traverse the VPN or DX link, reducing data transfer.

In general the WorkSpaces experience is enhanced and isn't highly dependent on item 5 as shown in Figure 6. This becomes even more the case when you want to scale WorkSpaces to thousands of desktops, especially in relation to AD DS global catalog queries, as this traffic remains local to the WorkSpaces environment.

Scenario 3: Standalone Isolated Deployment Using AWS Directory Service in the AWS Cloud

This scenario, shown in Figure 7, has AD DS deployed in the AWS Cloud in a standalone isolated environment. AWS Directory Service is used exclusively in this scenario. Instead of fully managing AD DS yourself, you rely on AWS Directory Service for tasks such as building a highly available directory topology, monitoring domain controllers, and configuring backups and snapshots.

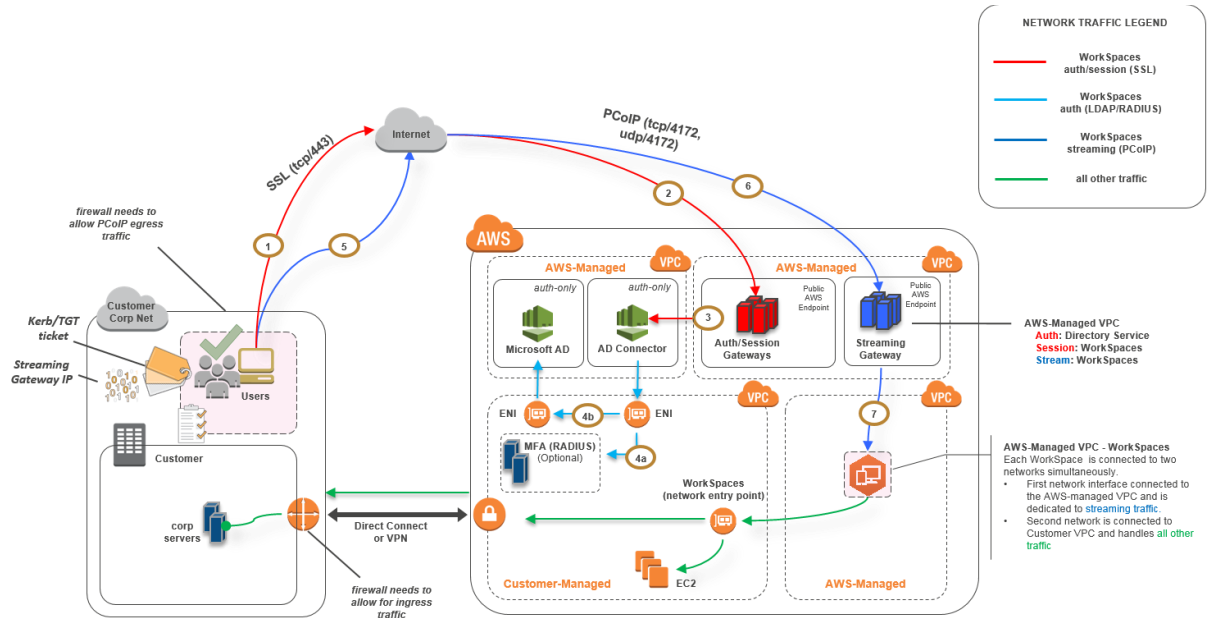


Figure 7: Cloud only - AWS Directory Services (Microsoft AD)

As in scenario 2, the AD DS (Microsoft AD) is deployed into dedicated subnets that span two Availability Zones, making AD DS highly available in the AWS Cloud. In addition to Microsoft AD, AD Connector (in all three scenarios) is deployed for WorkSpaces authentication or MFA. This ensures separation of roles or function within the Amazon VPC, which is a standard best practice (see *Design Considerations: Partitioned Network* section).

Scenario 3 is a standard all-in configuration that works well for customers who want to have AWS manage the deployment, patching, high availability, and monitoring of the AWS Directory Service. Because of its isolation mode, in addition to production, the scenario also works well for proof of concepts and lab environments.

In addition to the placement of AWS Directory Service, Figure 7 shows the flow of traffic from a user to a workspace and how the workspace interacts with the AD server and MFA server.

This architecture uses the following components or construct.

Amazon Web Services:

- **Amazon VPC:** Creation of an Amazon VPC with at least four private subnets across two Availability Zones (two for AD DS [Microsoft AD](#), two for AD Connector or WorkSpaces). “*Separation of roles.*”
- **DHCP options set:** Creation of an Amazon VPC DHCP options set. This allows you to define customer-specified domain name and DNSs (Microsoft AD). For more information, see [DHCP Options Sets](#).
- **Optional: Amazon virtual private gateway:** Enable communication with your own network over an IPsec VPN tunnel (VPN) or AWS Direct Connect connection. Use for accessing on-premises back-end systems.
- **AWS Directory Service:** Microsoft AD deployed into a dedicated pair of VPC subnets (AD DS Managed Service).
- **Amazon EC2:** Customer “Optional” RADIUS Servers for MFA.
- **AWS Directory Services:** AD Connector is deployed into a pair of Amazon VPC private subnets.
- **Amazon WorkSpaces:** WorkSpaces are deployed into the same private subnets as the AD Connector (see Design Considerations, AD Connector).

Customer:

- **Optional: Network Connectivity:** corporate VPN or AWS Direct Connect endpoints.
- **End user devices:** Corporate or BYOL end-user devices (such as Windows, Mac, iPad or Android tablets, zero clients, Chromebook), used to access the Amazon WorkSpaces service (see [Supported Platforms and Devices](#)).

Like scenario 2, this solution doesn’t have issues with reliance on connectivity to the customer on-premises data center, latency, or data out transfer costs (except where Internet access is enabled for WorkSpaces within the VPC) because by design, this is an isolated or cloud-only scenario.

Design Considerations

A functional AD DS deployment in the AWS Cloud requires a good understanding of both Active Directory concepts and specific AWS services. In this section, we discuss key design considerations when deploying AD DS for WorkSpaces, VPC best practices for AWS Directory Service, DHCP and DNS requirements, AD Connector specifics, and Active Directory sites and services.

VPC Design

As we discuss in the [Network Considerations](#) section of this document and documented earlier for scenarios 2 and 3, you should deploy AD DS in the AWS Cloud into a dedicated pair of private subnets, across two Availability Zones, and separated from AD Connector or WorkSpaces subnets. This construct provides highly available, low latency access to AD DS services for WorkSpaces, while maintaining standard best practices of separation of roles or functions within the Amazon VPC.

Figure 8 shows the separation of AD DS and AD Connector into dedicated private subnets (scenario 3). In this example all services reside in the same Amazon VPC.

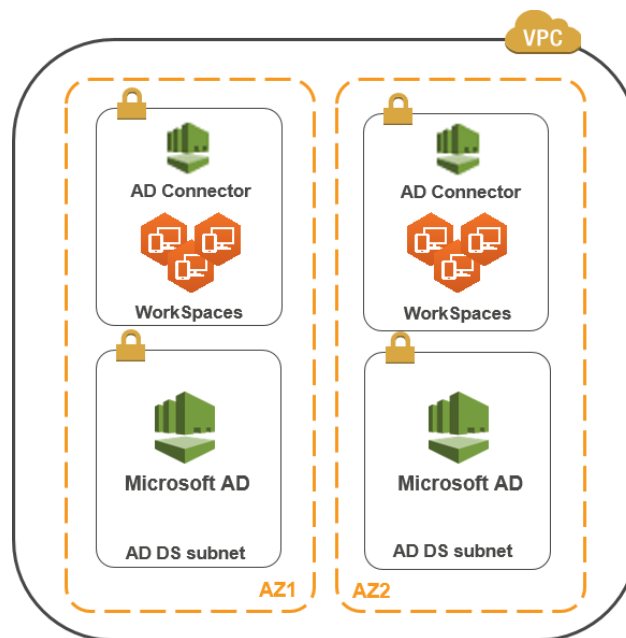


Figure 8: AD DS network segregation

Figure 9 shows a design similar to scenario 1, however, in this scenario the on-premises portion resides in a dedicated Amazon VPC.

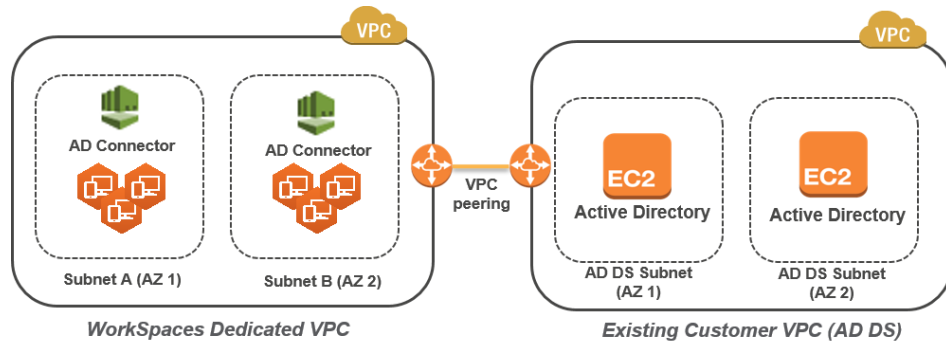


Figure 9: Dedicated WorkSpaces VPC

Note For customers who have an existing AWS deployment where AD DS is being used, we recommend that you locate your WorkSpaces in a dedicated VPC and that you use VPC peering for AD DS communications.

In addition to the creation of dedicated private subnets for AD DS, 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).

Note Best practice is to restrict the required security group rules to the WorkSpaces private subnets and, in the case of scenario 2, allow for bidirectional AD DS communications on-premises to/from the AWS Cloud, as shown in the following table.

| Protocol | Port | Use | Destination |
|----------|--------------------------------------|--|---|
| tcp | 53, 88, 135, 139, 389, 445, 464, 636 | Auth (primary) | Active Directory (private data center or EC2)* |
| tcp | 49152 – 65535 | RPC High Ports | Active Directory (private data center or EC2)** |
| tcp | 3268-3269 | Trusts | Active Directory (private data center or EC2)* |
| tcp | 9389 | Remote Microsoft Windows PowerShell (optional) | Active Directory (private data center or EC2)* |
| udp | 53, 88, 123, 137, 138, 389, 445, 464 | Auth (primary) | Active Directory (private data center or EC2)* |
| udp | 1812 | Auth (MFA) (optional) | RADIUS (private data center or EC2)* |

* See [Active Directory and Active Directory Domain Services Port Requirements](#)

**See [Service overview and network port requirements for Windows](#)

For step-by-step guidance for implementing rules, see [Adding Rules to a Security Group](#) in the *Amazon Elastic Compute Cloud User Guide*.

VPC Design: DHCP and DNS

With an Amazon VPC, DHCP services are provided by default for your instances. By default, every VPC provides an internal DNS server that is accessible via the Classless Inter-Domain Routing (CIDR) +2 address space and is assigned to all instances via a default 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. Correct functionality of Windows services within your VPC depends on this DHCP scope option and you need to set it correctly. In each of the scenarios defined earlier, you would create and assign your own scope that defines your domain name and name servers. This ensures that domain-joined Windows instances or WorkSpaces are configured to use the Active Directory DNS. The following table is an example of a custom set of DHCP scope options that must be created for WorkSpaces and AWS Directory Services to function correctly.

| Parameter | Value |
|-----------------------------|--|
| Name tag | Creates a tag with key = name and value set to a specific string Example: exampleco.com |
| Domain name | exampleco.com |
| Domain name servers | DNS server address, separated by commas Example: 10.0.0.10, 10.0.1.10 |
| NTP servers | Leave this field blank |
| NetBIOS name servers | Enter the same comma separated IPs as per domain name servers Example: 10.0.0.10, 10.0.1.10 |
| NetBIOS node type | 2 |

For details on creating a custom DHCP option set and associating it with your Amazon VPC, see [Working with DHCP Options Sets](#) in the *Amazon Virtual Private Cloud User Guide*.

In scenario 1, the DHCP scope would be the on-premises DNS or AD DS. However, in scenario 2 or 3, this would be the locally deployed directory service (AD DS on Amazon EC2 or AWS Directory Services: Microsoft AD). We

recommended that you to make each domain controller that resides in the AWS Cloud a global catalog and Directory Integrated DNS server.

Active Directory: Sites and Services

For [scenario 2](#), sites and services are critical components for the correct function of AD DS. Site topology controls Active Directory replication between domain controllers within the same site and across site boundaries. In scenario 2, at least two sites are present, on-premises and the AWS WorkSpaces in the cloud. Defining the correct site topology ensures client affinity, meaning that clients (in this case, WorkSpaces) use their preferred local domain controller.

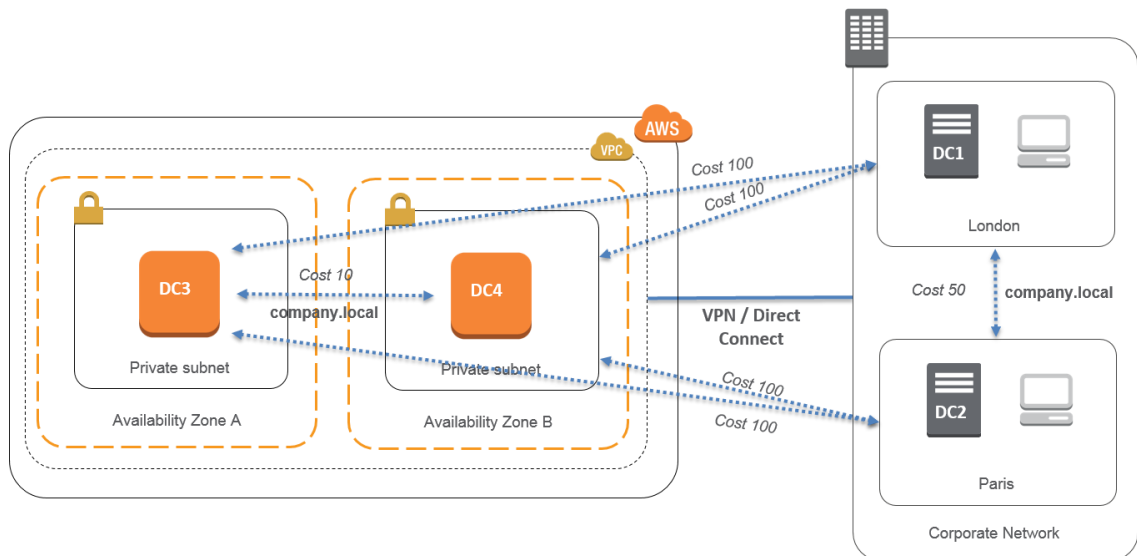


Figure 10: Active Directory sites and services: client affinity

Best practice Define high cost for site links between your on-premises AD DS and the AWS Cloud. Figure 10 is an example of what costs to assign to the site links (cost 100) to ensure site-independent client affinity.

These associations help ensure that traffic—such as AD DS replication, and client authentication—uses the most efficient path to a domain controller. In the case of scenarios 2 and 3, this helps ensure lower latency and cross-link traffic.

Multi-Factor Authentication (MFA)

Implementing MFA requires the WorkSpaces infrastructure to use AD Connector as its AWS Directory Service and have a RADIUS server. Although this document doesn't discuss the deployment of a RADIUS server, the previous section, AD DS Deployment Scenarios details the placement of RADIUS within each scenario.

MFA – Two-Factor Authentication

Amazon WorkSpaces supports MFA through AWS Directory Service: AD Connector and a *customer owned* RADIUS server. Once enabled, users are required to provide **Username**, **Password**, and **MFA Code** to the WorkSpaces client for authentication to their respective WorkSpaces desktops.

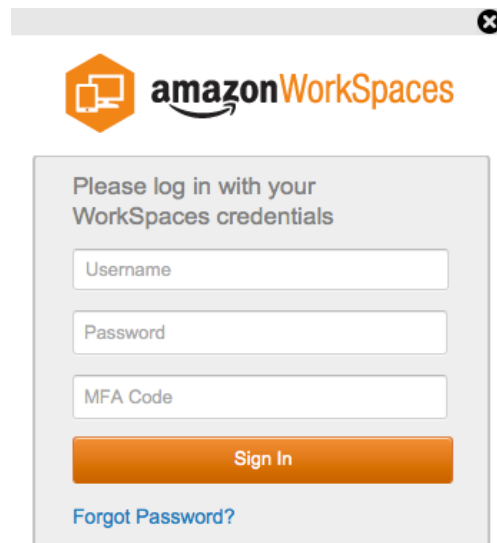
The image shows a screenshot of the Amazon WorkSpaces login window. At the top, there is the Amazon WorkSpaces logo. Below the logo, the text reads "Please log in with your WorkSpaces credentials". There are three input fields: "Username", "Password", and "MFA Code". Below these fields is an orange "Sign In" button. At the bottom left of the login box, there is a blue link that says "Forgot Password?". The entire login box is set against a light gray background.

Figure 11: WorkSpaces client with MFA enabled

Hard rule Implementing MFA authentication requires you to use AD Connector. AD Connector doesn't support selective "per user" MFA, as this is a global per AD Connector setting. If you require selective "per user" MFA, you must separate users by AD Connector.

WorkSpaces MFA requires one or more RADIUS servers. Typically, these are existing solutions, for example RSA, or the servers can be deployed within your VPC (see AD DS Deployment Scenarios). If you're deploying a new RADIUS solution, several implementations exist in the industry today, such as [FreeRADIUS](#), and cloud services such as [Duo Security](#).

For a list of prerequisites to implement MFA with Amazon WorkSpaces, see the *Amazon WorkSpaces Administration Guide*, [Preparing Your Network for an AD Connector Directory](#). The process for configuring your AD Connector for MFA is described in Managing an AD Connector Directory: [Multi-factor Authentication](#), in the *Amazon WorkSpaces Administration Guide*.

Security

This section explains how you secure the data by using encryption when you're using Amazon WorkSpaces services. We describe encryption in transit and at rest, as well as the utilization of security groups to protect network access to the WorkSpaces. You can find additional information about authentication (including MFA support) in the AWS Directory Service section.

Encryption in Transit

Amazon WorkSpaces uses cryptography to protect confidentiality at different stages of communication (in transit) and also to protect data at rest (encrypted WorkSpaces). The processes in each stage of the encryption used by Amazon WorkSpaces in transit is described in the following sections. For information about the encryption at rest, see the [Encrypted WorkSpaces](#) section later in this whitepaper.

Registration and Updates

The desktop client application communicates with Amazon for updates and registration using https.

Authentication Stage

The desktop client initiates authentication by sending credentials to the Authentication Gateway. The communication between the desktop client and Authentication Gateway uses https. At the end of this stage, if the authentication succeeds, Authentication Gateway returns an OAuth 2.0 token to the desktop client, through the same https connection.

Note The desktop client application supports the use of a proxy server for port 443 (HTTPS) traffic, for updates, registration, and authentication.

After receiving the credentials from the client, the Authentication Gateway sends an authentication request to AWS Directory Service. The communication from

Authentication Gateway to AWS Directory Service takes place over HTTPS, so no user credentials are transmitted in clear text.

Authentication - AD Connector

AD Connector uses Kerberos to establish authenticated communication with on-premises AD, so it can bind to LDAP and execute subsequent LDAP queries. At this moment, the AWS Directory Service does not support LDAP with TLS (LDAPS). However, no user credentials are transmitted in clear text at any time. For increased security, it is possible to connect your WorkSpaces VPC with your on-premises network (where your AD resides) using a VPN connection. When using an AWS hardware VPN connection, you will set up encryption in transit by using standard IPSEC (IKE and IPSEC SAs) with AES-128 or AES-256 symmetric encryption keys, SHA-1 or SHA-256 for integrity hash, and DH groups (2,14-18, 22, 23 and 24 for phase 1; 1,2,5, 14-18, 22, 23 and 24 for phase 2) using PFS.

Broker Stage

After receiving the OAuth 2.0 token (from Authentication Gateway, if the authentication succeeded), the desktop client will query Amazon WorkSpaces services (Broker Connection Manager) using HTTPS. The desktop client authenticates itself by sending the OAuth 2.0 token and, as a result, the client will receive the endpoint information of the WorkSpaces streaming gateway.

Streaming Stage

The desktop client requests to open a PCoIP session with the streaming gateway (using the OAuth 2.0 token). This session is aes256 encrypted and uses the PCoIP port for communication control (that is, 4172/tcp).

Using the OAuth2.0 token, the streaming gateway requests the user-specific WorkSpaces information from the WorkSpaces service, over https.

The streaming gateway also receives the TGT from the client (which is encrypted using the client user's password) and, by using Kerberos TGT pass-through, the gateway initiates a Windows login on the WorkSpace, using the user's retrieved Kerberos TGT.

The WorkSpace then initiates an authentication request to the configured AWS Directory Service, using standard Kerberos authentication.

After the WorkSpace is successfully logged in, the PCoIP streaming starts. The connection is initiated by the client on port tcp 4172 with the return traffic on port udp 4172. Additionally, the initial connection between the streaming gateway and your WorkSpaces desktop over the management interface is via UDP 55002. (See the Amazon Workspaces documentation, [Amazon WorkSpaces Details](#). The initial outbound UDP port is 55002.) The streaming connection, using ports 4172 (tcp and udp), is encrypted by using AES 128- and 256-bit ciphers, but default to 128-bit. You can actively change this to 256-bit via PCoIP-specific Active Directory GPO ([pcoip.adm](#)).

Network Interfaces

Each Amazon WorkSpace has two network interfaces, called the [primary network interface and management network interface](#).

The primary network interface provides connectivity to resources inside your VPC, such as access to AWS Directory Service, Internet, and your corporate network. It is possible to attach security groups to this primary network interface (as you would do to any ENI). Conceptually, we differentiate the security groups attached to this ENI based on the scope of the deployment: WorkSpaces security group and ENI security groups.

Management Network Interface

You can't control the management network interface via security groups, however, you can leverage a host-based firewall on your WorkSpace to block ports or control access. We don't recommend applying restrictions on the management network interface. If you decide to add host-based firewall rules to manage this interface, you need to keep a few ports open so that the WorkSpaces service can manage the health and accessibility to the WorkSpace as defined in the [Amazon WorkSpaces Administration Guide](#).

WorkSpaces Security Group

A default security group is created per AWS Directory Service and is automatically attached to all WorkSpaces that belong to that specific directory.

As with any other security group, it's possible to modify the rules of a WorkSpaces security group. The results take effect immediately after the changes are applied.

It is also possible to change the default WorkSpaces security group attached to an AWS Directory Service by changing the WorkSpaces [security group](#) association.

Note A newly associated security group will be attached only to WorkSpaces created or rebuilt after the modification.

ENI Security Groups

Because the primary network interface is a regular ENI, you can manage its configuration using the different AWS management tools (see [Elastic Network Interfaces \(ENI\)](#)). In particular, look for the WorkSpace IP (in the WorkSpaces page in the Amazon WorkSpaces console), and then use that IP address as a filter to find the corresponding ENI (in the Network Interfaces section of the Amazon EC2 console).

Once you find the ENI, you can directly manage security groups from there. When manually assigning security groups to the primary network interface, consider the port requirements of Amazon WorkSpaces, as explained in [Amazon WorkSpaces Details](#).

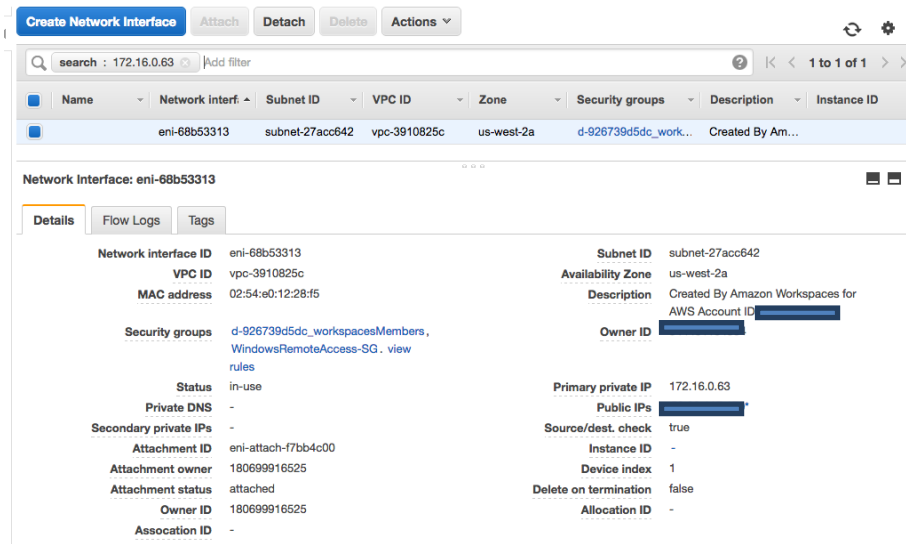


Figure 12: Managing security group associations

Encrypted WorkSpaces

Each Amazon WorkSpace is provisioned with a root volume (C: drive) and a user volume (D: drive). The encrypted WorkSpaces feature enables you to encrypt either volume or both volumes.

What is Encrypted?

The data stored at rest, disk I/O to the volume, and snapshots created from encrypted volumes are all encrypted.

When Does Encryption Occur?

You should specify encryption for a WorkSpace when launching (creating) the WorkSpace. WorkSpaces volumes can be encrypted only at launch time: after launch, you cannot change the encryption status of a volume. Figure 13 shows the Amazon WorkSpaces console page for choosing encryption during the launching of a new WorkSpace.

Launch WorkSpaces

Step 1: Select Directory

Step 2: Identify Users

Step 3: Select Bundles

Step 4: WorkSpaces Configuration

Step 5: Review

Encryption

You can choose to optionally encrypt the storage volumes in your WorkSpaces. To configure volume encryption you need to use KMS keys in your account. You may use the [IAM console](#) to create additional KMS keys. To learn more about encryption on WorkSpaces, please see our [documentation here](#).

| Username | Root Volume (C: Drive) Encryption | User Volume (D: Drive) Encryption | Encryption Key |
|----------|-------------------------------------|-------------------------------------|----------------------|
| Admin | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | alias/aws/workspaces |

Figure 13: Encrypting WorkSpaces volumes

How Is a New WorkSpace Encrypted?

You can choose the Encrypted WorkSpaces option from either the Amazon WorkSpaces console or AWS CLI, or by using the Amazon WorkSpaces API at the moment you launch a new WorkSpace.

To encrypt the volumes, Amazon WorkSpaces uses a customer master key (CMK) from AWS Key Management Service (KMS). A default AWS KMS CMK is created the first time a WorkSpace is launched in a region (CMKs have a region scope). You can also create a customer managed CMK to use with encrypted WorkSpaces. The CMK is used to encrypt the data keys that are used by Amazon WorkSpaces service to encrypt the volumes (in a strict sense, it will be Amazon Elastic Block Store (Amazon EBS) service that will encrypt the volumes). Each CMK can be used to encrypt keys for up to 30 WorkSpaces.

Note Creating custom images from an encrypted WorkSpace is currently not supported. Also WorkSpaces launched with root volume encryption enabled can take up to an hour to get provisioned.

For a detailed description of the WorkSpaces encryption process, see [Overview of Amazon WorkSpaces Encryption Using AWS KMS](#). For additional information about AWS KMS customer master keys and data keys, see [AWS Key Management Service Concepts](#).

Monitoring or Logging Using Amazon CloudWatch

Monitoring is an integral part of any infrastructure, be that network, servers, or logs. Customers who deploy Amazon WorkSpaces need to monitor their deployments, specifically the overall health and connection status of individual WorkSpaces.

Amazon CloudWatch Metrics for WorkSpaces

CloudWatch metrics for WorkSpaces is designed to provide administrators with additional insight into the overall health and connection status of individual WorkSpaces. Metrics are available per WorkSpace or aggregated for all WorkSpaces in an organization within a given directory (*AD Connector, see Identity*).

These metrics, like all CloudWatch metrics, can be viewed in the AWS Management Console (Figure 13), accessed via the CloudWatch APIs, and monitored by CloudWatch alarms and third-party tools.

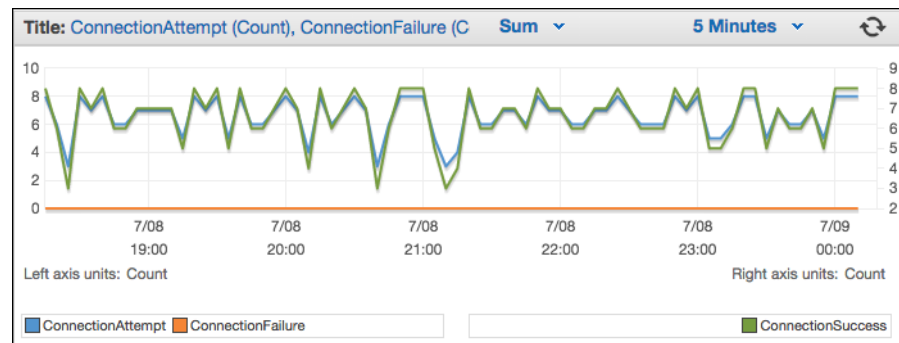


Figure 14: CloudWatch metrics – ConnectionAttempt/ConnectionFailure

By default, the following metrics are enabled and are available at no extra charge:

- **Available:** WorkSpaces that respond to a status check are counted in this metric.
- **Unhealthy:** WorkSpaces that don't respond to the same status check are counted in this metric.

- **ConnectionAttempt:** The number of connection attempts made to a WorkSpace.
- **ConnectionSuccess:** The number of successful connection attempts.
- **ConnectionFailure:** The number of unsuccessful connection attempts.
- **SessionLaunchTime:** The amount of time taken to initiate a session, as measured by the WorkSpaces client.
- **InSessionLatency:** The round-trip time between the WorkSpaces client and WorkSpaces, as measured and reported by the client.
- **SessionDisconnect:** The number of user-initiated and automatically closed sessions.

Additionally, alarms can be created, as shown in Figure 15.

Figure 15: Create CloudWatch alarm for WorkSpaces connection errors

Troubleshooting

Common administration and client issues, such as “I see the following error message: "Your device is not able to connect to the WorkSpaces Registration service" or “Can't connect to a WorkSpace with an interactive logon banner” can be found on the Client and Admin Troubleshooting pages in the *Amazon WorkSpaces Administration Guide*.

AD Connector Cannot Connect to Active Directory

For AD Connector to connect to your on-premises directory, the firewall for your on-premises network must have certain ports open to the CIDRs for both subnets in the VPC (see [AD Connector](#)). To test if these conditions are met, perform the following steps.

To verify the connection

1. Launch a Windows instance in the VPC and connect to it over RDP. The remaining steps are performed on the VPC instance.
2. Download and unzip the [DirectoryServicePortTest](#) test application. The source code and Visual Studio project files are included so you can modify the test application, if you choose.
3. From a Windows command prompt, run the DirectoryServicePortTest test application with the following options:

```
DirectoryServicePortTest.exe -d <domain_name> -ip <server_IP_address> -tcp  
"53,88,135,139,389,445,464,636,49152" -udp "53,88,123,137,138,389,445,464"  
<domain_name>
```

<domain_name>

The fully qualified domain name, used to test the forest and domain functional levels. If you exclude the domain name, the functional levels won't be tested.

<server_IP_address>

The IP address of a domain controller in your on-premises domain. The ports will be tested against this IP address. If you exclude the IP address, the ports won't be tested.

This will determine if the necessary ports are open from the VPC to your domain. The test app also verifies the minimum forest and domain functional levels.

How to Check Latency to Closest AWS Region

In October 2015, Amazon WorkSpaces launched the [Connection Health Check website](#). The website quickly checks whether you can get to all of the required services to use WorkSpaces. It also does a performance check to each AWS Region where WorkSpaces run, and lets users know which one will be fastest for them.

Conclusion

We're seeing a strategic shift in end-user computing as organizations strive to be more agile, better protect their data, and help their workers be more productive. Many of the benefits already realized with cloud computing also apply to end user computing. By moving their desktops to the AWS Cloud with Amazon WorkSpaces, organizations can quickly scale as they add workers, improve their security posture by keeping data off devices, and offer their workers a portable desktop with anywhere access from the device of their choice.

Amazon WorkSpaces is designed to be integrated into existing IT systems and processes, and this whitepaper describes the best practices for doing this. The result of following the guidelines in this whitepaper is a cost-effective cloud desktop deployment that scales with your business on the AWS global infrastructure.

Contributors

The following individuals contributed to this document:

- Justin Bradley, Solutions Architect, Amazon Web Services
- Mahdi Sajjadpour, Senior Consultant, AWS Professional Services

- Mauricio Munoz, Solutions Architect, Amazon Web Services

Further Reading

For additional help, please consult the following sources:

- [Troubleshooting AWS Directory Service Administration Issues](#)
- [Troubleshooting Amazon WorkSpaces Administration Issues](#)
- [Troubleshooting Amazon WorkSpaces Client Issues](#)
- [Amazon WorkSpaces Administration Guide](#)
- [Amazon WorkSpaces Developer Guide](#)
- [Supported Platforms and Devices](#)
- [How Amazon WorkSpaces Uses AWS KMS](#)
- [AWS CLI Command Reference – workspaces](#)
- [Monitoring Amazon WorkSpaces Metrics](#)