Best Practices for Deploying Amazon WorkSpaces

AWS Whitepaper

Best Practices for Deploying Amazon WorkSpaces: AWS Whitepaper

Table of Contents

Abstract 1
Abstract 1
Introduction 2
WorkSpaces Requirements 3
Network Considerations 4
VPC Design 5
Network Interfaces 5
Traffic Flow 6
Client Device to WorkSpace 6
Amazon WorkSpaces Service to VPC 8
Example of a Typical Configuration 8
AWS Directory Service 11
AD DS Deployment Scenarios 12
Scenario 1: Using AD Connector to Proxy Authentication to On-Premises Active Directory service 12
Scenario 2: Extending On-Premises AD DS into AWS (Replica) 14
Scenario 3: Standalone Isolated Deployment Using AWS Directory Service in the AWS Cloud 17
Scenario 4: AWS Microsoft AD and a Two-Way Transitive Trust to On-Premises 18
Scenario 5: AWS Microsoft AD using a Shared Services Virtual Private Cloud (VPC) 20
Scenario 6: AWS Microsoft AD, Shared Services VPC, and a One-Way Trust to On Premises 21
Design Considerations 23
VPC Design 23
VPC Design: DHCP and DNS 25
Active Directory: Sites and Services 25
Protocol 26
Multi-Factor Authentication (MFA) 27
MFA – Two-Factor Authentication 27
Disaster Recovery / Business Continuity 28
WorkSpaces Cross-Region Redirection 28
Smart Card Support 30
Root CA 30
In-session 31
Pre-session 31
Client Deployment 32
Endpoint Selection 33
Web Access client 35
Amazon WorkSpaces Tags 36
Tag Restrictions 36
Resources that you can tag 36
Using the cost allocation tag 36
Automating Amazon Workspaces Deployment 37
Common WorkSpaces Automation Methods 37
WorkSpaces Deployment Automation Best Practices 38
Amazon Workspaces Language Packs 39
Amazon WorkSpaces Profile Management 39
Folder Redirection 39
Best Practices 39
Things to Avoid 40
Other Considerations 40
Profile Settings 40
Group Policies 40
Amazon Workspaces Volumes 41
Best Practices 41
WorkSpaces Logging 41
Amazon WorkSpaces Client 3.x 41

Amazon Workspaces Migrate 43
Migration Process 43
Migration Procedure 43
Migration Limits 43
Cost 44
WorkSpaces Migration Best Practices 44
Well-Architected Framework 45
Security 47
Encryption in Transit 47
Registration and Updates 47
Authentication Stage 47
Authentication — Active Directory Connector (ADC) 47
Broker Stage 48
Streaming Stage 48
Network Interfaces 48
Management Network Interface 48
WorkSpaces Security Group 49
ENI Security Groups 49
Encrypted WorkSpaces 50
What is Encrypted? 50
When Does Encryption Occur? 50
How is a New WorkSpace Encrypted? 50
Access Control Options and Trusted Devices 51
IP Access Control Groups 51
Monitoring or Logging Using Amazon CloudWatch 52
Amazon CloudWatch Metrics for WorkSpaces 52
Amazon CloudWatch Events for WorkSpaces 53
Cost Optimization 54
Self-Service WorkSpace Management Capabilities 54
Amazon WorkSpaces Cost Optimizer 54
Opting Out with Tags 55
Troubleshooting 56
AD Connector Cannot Connect to Active Directory 56
Troubleshooting a WorkSpace Custom Image Creation Error 56
Troubleshooting a Windows WorkSpace Marked as Unhealthy 57
Collecting a WorkSpaces Support Log Bundle for Debugging 58
PcoIP Server-Side Log 58
WebAccess Server-Side Logs 59
Client-Side Logs 59
Automated Server Side Log Bundle Collection for Windows 60
How to Check Latency to Closest AWS Region 60
Conclusion 61
Contributors 62
Further Reading 63
Document Revisions 64
Notices 65

Best Practices for Deploying Amazon WorkSpaces

Publication date: **April 28, 2021** (*Document Revisions* (p. 64))

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.

This whitepaper is meant to enable quick access to relevant information. It is intended for network engineers, directory engineers, or security engineers.

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 Amazon Web Services (AWS) command line interface (CLI), or by using the application programming interface (API). With Amazon WorkSpaces, you can launch a Microsoft Windows or Amazon Linux desktop within minutes, which enables you to connect to and access your desktop software securely, reliably, and quickly from on-premises or from an external network. You can:

- Leverage your existing, on-premises Microsoft Active Directory (AD) by using AWS Directory Service : Active DirectoryConnector (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.
- Leverage your on-premises or cloud-hosted RADIUS server with AD Connector to provide multi-factor authentication (MFA) to your WorkSpaces.

You can automate the 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 Amazon WorkSpaces service provides, you can also enable encryption at rest for your WorkSpaces. See the Encrypted WorkSpaces (p. 50) section of this document).

You can deploy applications to your WorkSpaces by using your existing on-premises tools, such as Microsoft System Center Configuration Manager (SCCM), Puppet Enterprise, or Ansible.

The following sections provide details about Amazon WorkSpaces, explain how the service works, describe what you need to launch the service, and tells you 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. See Getting Started with Your Workspace.

  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 Microsoft AD. You can use your on-premises AD server with AWS Directory Service to support your existing enterprise user credentials with Amazon WorkSpaces.
- **Amazon Virtual Private Cloud (Amazon VPC) in which to run your Amazon WorkSpaces** — You'll need a minimum of two subnets for an Amazon WorkSpaces deployment because each AWS Directory Service construct requires two subnets in a Multi-AZ deployment.

Network Considerations

Each WorkSpace is associated with the specific Amazon VPC and AWS Directory Service construct that 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 (AZs). Subnets are permanently affiliated with a Directory Service construct and can't be modified after it is created. Because of this, 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 AD 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. Answers to these questions 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 is required, or whether an end-user can have local administrator access on their WorkSpace.

> **Note**
> Each AD Connector connects to your existing Enterprise Microsoft AD. To take advantage of this capability and specify an Organizational Unit (OU), you must construct your Directory Service to take your user personas into consideration.

VPC Design

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

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

- **VPC** — AWS recommends 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 provides 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 uses 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 AWS uses on a per-Region basis, see Amazon WorkSpaces Details.

> **Note**
> Amazon WorkSpaces and their associated management network interfaces do not reside in your VPC, and you cannot view the management network interface or the Amazon Elastic Compute Cloud (Amazon EC2) instance ID in your AWS Management Console (see Figures 4, 5, and 6). However, you can view and modify the security group settings of your primary network interface (`eth1`) in the console. The primary network interface of each WorkSpace does count toward your ENI Amazon EC2 resource quotas. For large deployments of Amazon WorkSpaces, you need to open a support ticket via the AWS Management Console to increase your ENI quotas.

Traffic Flow

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

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

The next section discusses both of these components.

Client Device to WorkSpace

Regardless of its location (on premises or remote), the device running the Amazon WorkSpaces client uses the same two ports for connectivity to the Amazon WorkSpaces service. The client uses port 443 (HTTPS port) for all authentication and session-related information, and port 4172 (PCoIP port), with both Transmission Control Protocol (TCP) and User Datagram Protocol (UDP), for pixel streaming to a given WorkSpace and network health checks. Traffic on both ports is encrypted. Port 443 traffic is used for authentication and session information and uses TLS for encrypting the traffic. Pixel streaming traffic uses AES-256-bit encryption for communication between the client and `eth0` of the WorkSpace, via the streaming gateway. More information can be found in the Security (p. 47) section of this document.

AWS publishes per-region IP ranges of its 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 can 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 choosing **Network** on the bottom-right side of the client. See Figure 1.

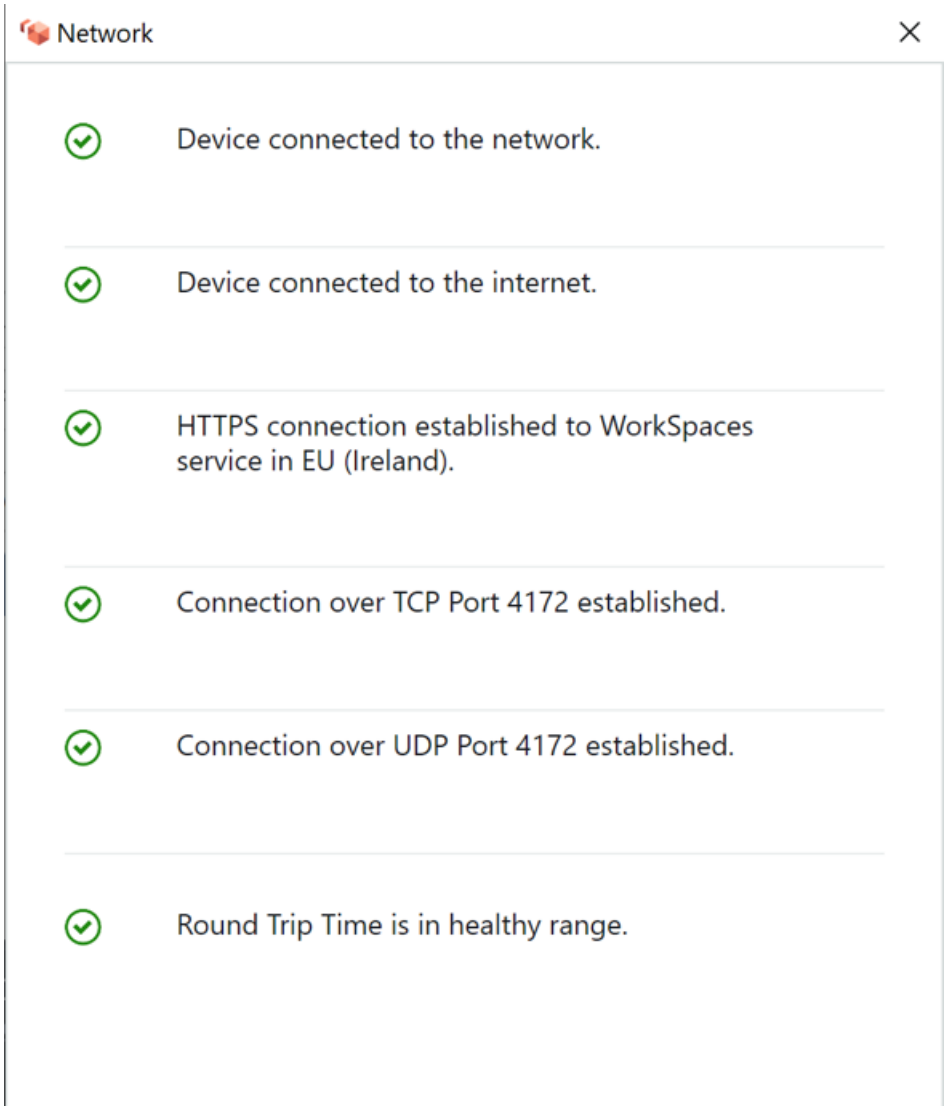

Figure 1 — WorkSpaces Client — network check

A user initiates a connection from their client to the Amazon WorkSpaces service by supplying their login information for the directory used by the Directory Service construct, typically their corporate directory. The login information is sent via HTTPS to the authentication gateways of the Amazon WorkSpaces service in the Region where the WorkSpace is located. The authentication gateway of the Amazon WorkSpaces service then forwards the traffic to the specific AWS Directory 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 (p. 12) in this document). 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 AD server. The AD Connector determines which Active Directory server to connect 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 is checked before the directory service authentication. Should the MFA validation fail, the user's login information is not forwarded to your AWS Directory Service.

After a user is authenticated, the streaming traffic starts by using 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 uses 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 the Monitoring or Logging Using Amazon CloudWatch (p. 52) section of this document.

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, AWS recommends that the round-trip time (RTT) between the WorkSpaces client and the AWS Region where your WorkSpaces are located is less than 100 milliseconds (ms). Typically this means your WorkSpaces client is located less than 2,000 miles from the Region in which the WorkSpace is being hosted. AWS provides a Connection Health Check webpage to help you determine the most optimal AWS Region to connect to 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 either a Windows or Linux desktop (your Amazon WorkSpace) that is connected to your virtual private cloud (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 in your VPC has access to any resource in the VPC, and to any network 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 route table of the subnet and default security group that your AWS Directory Service configures for each WorkSpace, as well any additional security groups that you assign to the ENI. You can add security groups to the ENI facing your VPC at any time by using the AWS Management Console or AWS CLI. (For more information on security groups, see Security Groups for Your WorkSpaces.) 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 the the section called "Scenario 1: Using AD Connector to Proxy Authentication to On-Premises Active Directory service" (p. 12) section of this whitepaper shows the traffic flow described.

Example of a Typical Configuration

Consider a scenario where you have two types of users and 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 without restrictions from their WorkSpace.

For contractors, you want to block local administrator access so that they can only use specific pre-installed applications. You want to apply restrictive network access controls using security groups for these WorkSpaces. You need to open ports 80 and 443 to specific internal websites only, and you want to entirely 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. You will need to create two AD Connectors, one for each user persona. Each AD Connector requires two subnets that have enough IP addresses available 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 Network Address Translation (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 the section called "Scenario 1: Using AD Connector to Proxy Authentication to On-Premises Active Directory service" (p. 12)) or part of the AWS Cloud implementation (see the section called "Scenario 2: Extending On-Premises AD DS into AWS (Replica) " (p. 14) and the section called "Scenario 3: Standalone Isolated Deployment Using AWS Directory Service in the AWS Cloud" (p. 17) in the AD DS Deployment Scenarios (p. 12)) section of this document.

Given that all WorkSpaces are granted some form of internet access, and given that they are hosted in a private subnet, you also must create public subnets that can access the internet through an internet gateway. You 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 zones. 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.

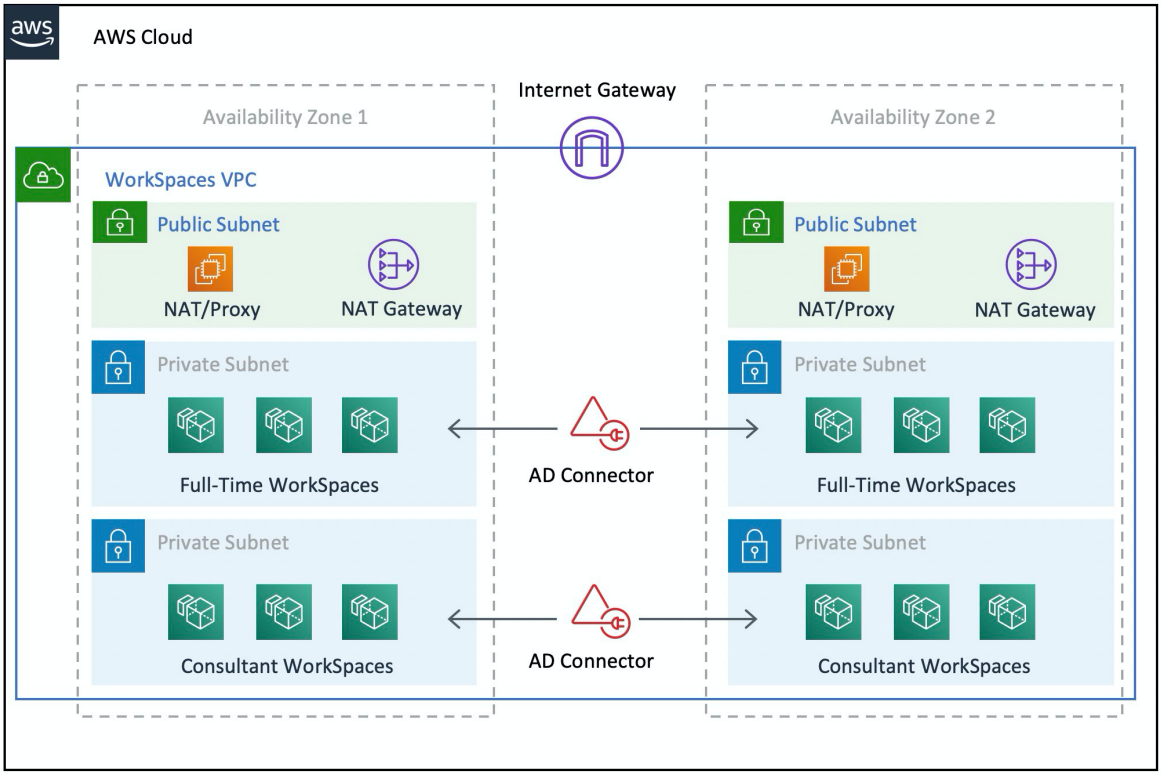

Figure 2 — High-level VPC design

The following information describes how to configure the two different WorkSpaces types:

To configure WorkSpaces for full-time employees:

1. In the Amazon WorkSpaces Management Console, choose the **Directories** option on the menu bar.
2. Choose the directory that hosts your full-time employees.
3. Choose **Local Administrator Setting**.

By enabling this option, any newly created WorkSpace will have local administrator privileges. To grant internet access, configure NAT for outbound internet access from your VPC. To enable MFA, you need to specify a RADIUS server, server IPs, ports, and a pre-shared key.

For full-time employees' WorkSpaces, inbound traffic to the WorkSpace can be limited to Remote Desktop Protocol (RDP) from the Helpdesk subnet by applying a default security group via the AD Connector settings.

To configure WorkSpaces for contractors and consultants:

1. In the Amazon WorkSpaces Management Console, disable the **Internet Access** and the **Local Administrator** setting.
2. 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 prevents 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.

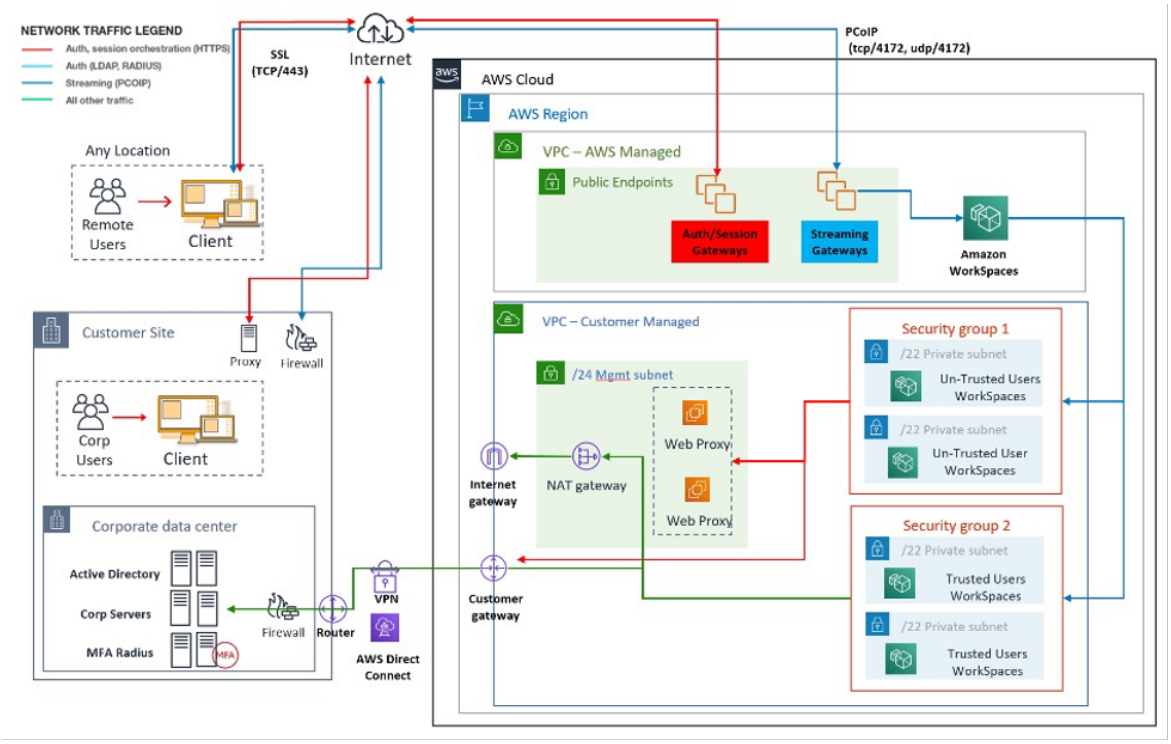

Figure 3 — WorkSpaces design with user personas

AWS Directory Service

As mentioned in the introduction, AWS Directory Service is a core component of Amazon WorkSpaces. With AWS Directory Service, you can create three types of directories with Amazon WorkSpaces:

- **AWS Managed Microsoft AD**, which is a managed Microsoft Active Directory, powered by Windows Server 2012 R2. AWS Managed Microsoft AD is available in Standard or Enterprise Edition.
- **Simple AD** is standalone, Microsoft Active Directory-compatible, managed directory service powered by Samba 4.
- **AD Connector** is a directory proxy for redirecting 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. It also discusses 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

Backing Amazon WorkSpaces is the AWS Directory Service, and the proper design and deployment of the directory service is critical. The following three scenarios build upon the Active Directory Domain Services on AWS Quick Start guide, and describe the best practice deployment options for AD DS when used with Amazon WorkSpaces. The Design Considerations (p. 23) section of this document details 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 Active Directory service** — In this scenario, network connectivity (VPN/Direct Connect) 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 Direct Connect.
- **AWS Microsoft AD and a Two-Way Transitive Trust to On-Premises** — This scenario includes the AWS Managed Microsoft Active Directory Service (MAD) with a two-way transitive trust to the on-premises Microsoft AD forest.
- **Scenario 5: AWS Microsoft AD using a Shared Services Virtual Private Cloud (VPC)** — This scenario uses AWS Managed Microsoft AD in a Shared Services VPC to be used as an Identity Domain for multiple AWS Services (Amazon EC2, Amazon WorkSpaces, and so on.) while using the AD Connector to proxy Lightweight Directory Access Protocol (LDAP) user authentication requests to the AD domain controllers.
- **Scenario 6: AWS Microsoft AD, Shared Services VPC, and a One-Way Trust to On Premises** — This scenario is similar to Scenario 5, but it includes disparate identity and resource domains using a one-way trust to on-premises.

Scenario 1: Using AD Connector to Proxy Authentication to On-Premises Active Directory service

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

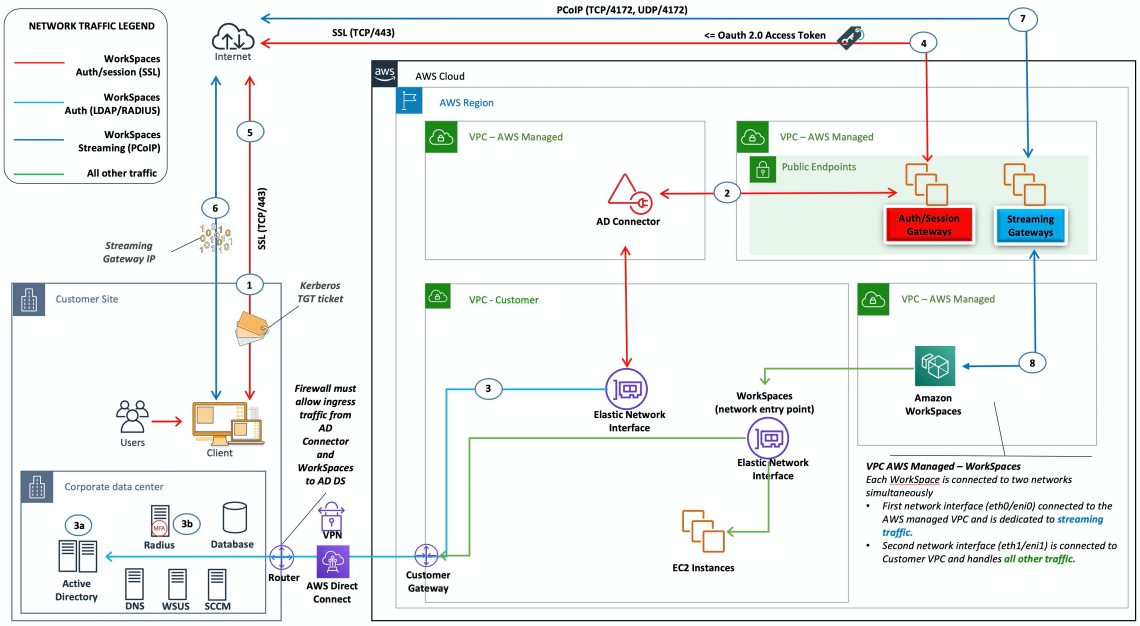

Figure 4 — AD Connector to on-premises Active Directory

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 (p. 47) section of this document.

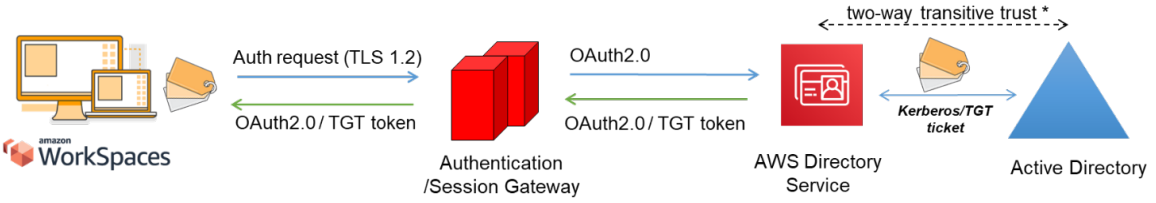

Figure 5 — User authentication via the Authentication Gateway

Scenario 1 shows a hybrid architecture where the customer might already have resources in AWS, as well as resources in an on-premises data center that could be accessed via Amazon 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 constructs.

AWS

- **Amazon VPC** — Creation of an Amazon VPC with at least two private subnets across two AZs.
- **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 in the same private subnets as the AD Connector (see the Design Considerations (p. 23) section of this document).

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, Macs, iPads, Android tablets, zero clients, and Chromebooks) used to access the Amazon WorkSpaces service. See this list of client applications for supported devices and web browsers.

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

- **Reliance on connectivity** — If connectivity to the data center is lost, users cannot log in to their respective WorkSpaces, and existing connections will remain active for the Kerberos/Ticket-Granting Ticket (TGT) lifetime.
- **Latency** — If latency exists via the connection (this is more the case with VPN than Direct Connect), 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 Direct Connect link, and so it depends on the connection type. This is either Data Transfer Out from Amazon EC2 to internet or Data Transfer Out (Direct Connect).

> **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.

In general, the WorkSpaces experience is highly dependent on item 5 shown in Figure 4. For this scenario, the WorkSpaces authentication experience is highly dependent on the network link between the customer AD and the WorkSpaces VPC. The customer should ensure the link is highly available.

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

This scenario is similar to scenario 1. However, in this scenario, 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.

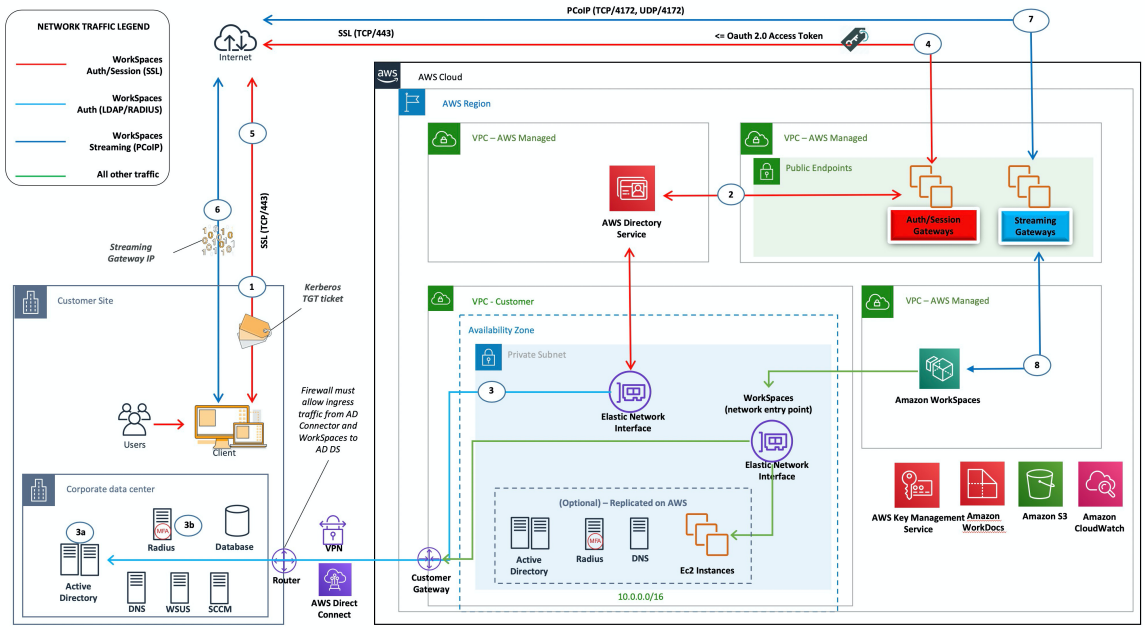

Figure 6 — Extend customer Active Directory Domain to the cloud

As in scenario 1, AD Connector is used for all user or MFA authentication, which in turn is proxied to the customer AD DS (see Figure 5). In this scenario, the customer AD DS is deployed across AZs on Amazon EC2 instances that are promoted to be domain controllers in the customer's on-premises AD 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 on AWS, see the Design Considerations (p. 23) section of this document.

After 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 Direct Connect.

This architecture uses the following components or constructs.

AWS

- **Amazon VPC** — Creation of an Amazon VPC with at least four private subnets across two AZs (two for the customer AD DS, two for AD Connector or Amazon WorkSpaces).
- **DHCP Options Set** — Creation of an Amazon VPC DHCP options set. This enables the customer to define a specified domain name and DNSs (AD DS local). For more information, see DHCP Options Sets.
- **Amazon virtual private gateway** — Enable communication with a customer-owned 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 on Amazon EC2 instances in dedicated private VPC subnets.
- **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 . For more information, see the Design Considerations (p. 23) section of this document.

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, Macs, iPads, Android tablets, zero clients, and Chromebooks) used to access the Amazon WorkSpaces service. See this list of client applications for supported devices and web browsers. This solution doesn't have the same caveats as scenario 1. Amazon 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, all authentication is local and low latency. See the

  For Scenario 2: Extending On-Premises AD DS into AWS (Replica) (p. 14) 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 Amazon 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.

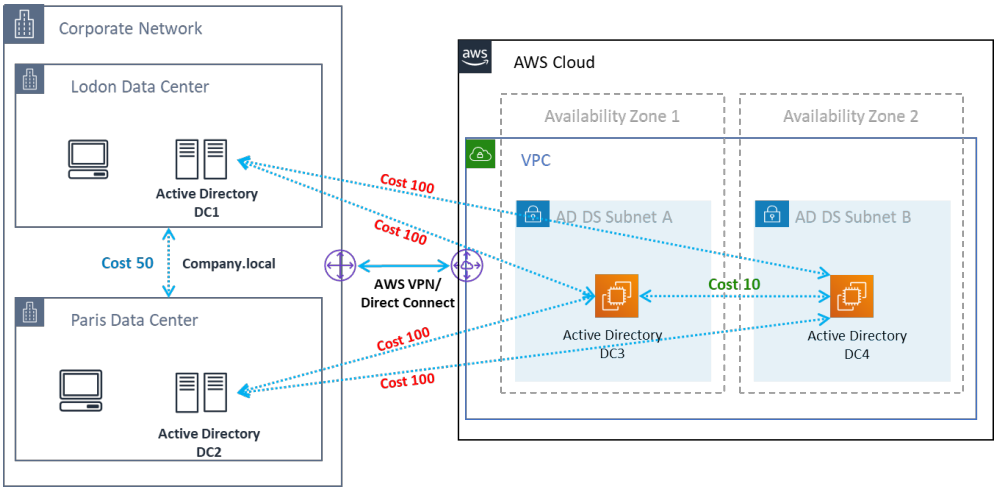

Figure 13 — Active Directory sites and services: client affinity

  Best practice
  Define high cost for site links between on-premises AD DS and the AWS Cloud. Figure 13 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.
  (p. 25) section of this document.
- **Traffic costs** — In this scenario, authentication is local, with only AD DS replication having to traverse the VPN or Direct Connect 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 is also the case when a customer wants 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, customers can rely on AWS Directory Service for tasks such as building a highly available directory topology, monitoring domain controllers, and configuring backups and snapshots.

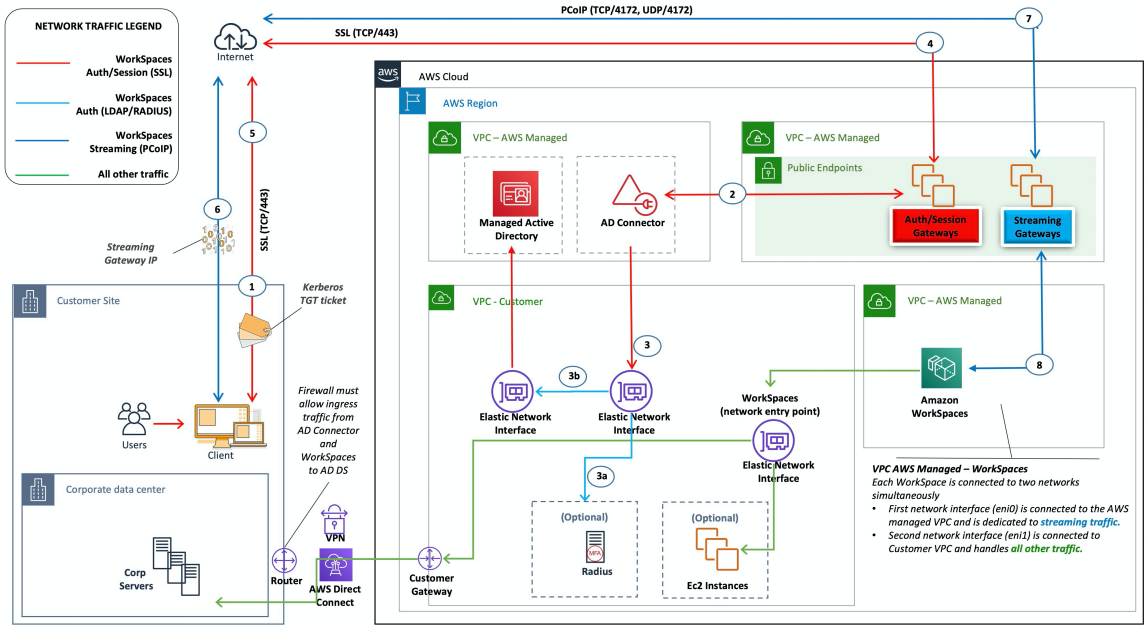

Figure 7 — Cloud only — AWS Directory Services (Microsoft AD)

As in scenario 2, the AD DS (Microsoft AD) is deployed into dedicated subnets that span two AZs, 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 functions within the Amazon VPC, which is a standard best practice. For more information, see the Design Considerations (p. 23) section of this document.

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. The scenario also works well for proof of concepts, lab, and production environments because of its isolation mode.

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 constructs:

AWS

- **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 DS Microsoft AD , two for AD Connector or WorkSpaces.
- **DHCP options set** — Creation of an Amazon VPC DHCP options set. This allows a customer to define a specified domain name and DNS(s) (Microsoft AD). For more information, see DHCP Options Sets.
- **Optional: Amazon virtual private gateway** — Enable communication with a customer-owned 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. For more information, see the Active Directory: Sites and Services (p. 25) section of this document.

Customer

- **Optional: Network Connectivit** — corporate VPN or AWS Direct Connect endpoints.
- **End user devices** — Corporate or BYOL end-user devices (such as Windows, Macs, iPads, Android tablets, zero clients, and Chromebooks) used to access the Amazon WorkSpaces service. See this list of client applications for supported devices and web browsers.

Like scenario 2, this scenario 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.

Scenario 4: AWS Microsoft AD and a Two-Way Transitive Trust to On-Premises

This scenario, shown in Figure 8, has AWS Managed AD deployed in the AWS Cloud, which has a two-way transitive trust to the customer on-premises Active Directory. User accounts and WorkSpaces are created in the Managed AD, with the Active Directory trust enabling resources to be accessed in the on-premises environment.

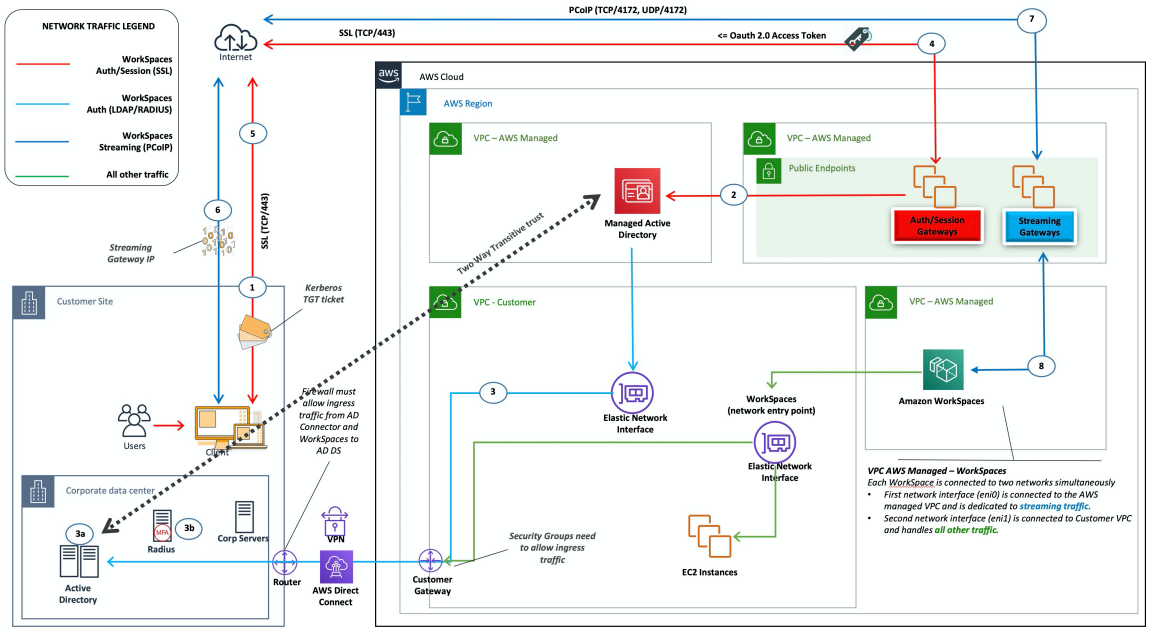

Figure 8 —AWS Microsoft AD and a two-way transitive trust to an on-premises location

As in scenario 3, the AD DS (Microsoft AD) is deployed into dedicated subnets that span two AZs, making AD DS highly available in the AWS Cloud.

This scenario works well for customers who want to have a fully managed AWS Directory Service, including deployment, patching, high availability, and monitoring of their AWS Cloud. This scenario also allows WorkSpaces users to access AD-joined resources on their existing networks. This scenario requires a domain trust to be in place. Security groups and firewall rules need to allow communication between the two active directories.

In addition to the placement of AWS Directory Service, Figure 8 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:

AWS

- **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).
- **DHCP options set** — Creation of an Amazon VPC DHCP options set. This allows a customer to define a specified domain name and DNS(s) (Microsoft AD). For more information, see DHCP Options Sets.
- **Optional: Amazon virtual private gateway** — Enable communication with a customer-owned 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.
- **Amazon WorkSpaces** — WorkSpaces are deployed into the same private subnets as the AD Connector. For more information, see the Active Directory: sites and services section of this document.

Customer

- **Network Connectivity** — Corporate VPN or AWS Direct Connect endpoints.
- **End user devices** — Corporate or BYOL end-user devices (such as Windows, Macs, iPads, Android tablets, zero clients, and Chromebooks) used to access the Amazon WorkSpaces service. See this list of client applications for supported devices and web browsers.

  This solution requires connectivity to the customer on-premises data center to allow the trust process to operate. If WorkSpaces users are using resources on the on-premises network, then latency and outbound data transfer costs need to be considered.

Scenario 5: AWS Microsoft AD using a Shared Services Virtual Private Cloud (VPC)

This scenario, shown in Figure 9, has an AWS Managed AD deployed in the AWS Cloud, providing authentication services for workloads that are either already hosted in AWS or are planned to be as part of a broader migration. The best practice recommendation is to have Amazon WorkSpaces in a dedicated VPC. Customers should also create a specific AD OU to organize the WorkSpaces computer objects.

To deploy WorkSpaces with a shared services VPC hosting Managed Active Directory, deploy an Active Directory Connector with an ADC service account created in the Managed AD. The service account requires permissions to create computer objects in the WorkSpaces designated OU in the shared services Managed AD.

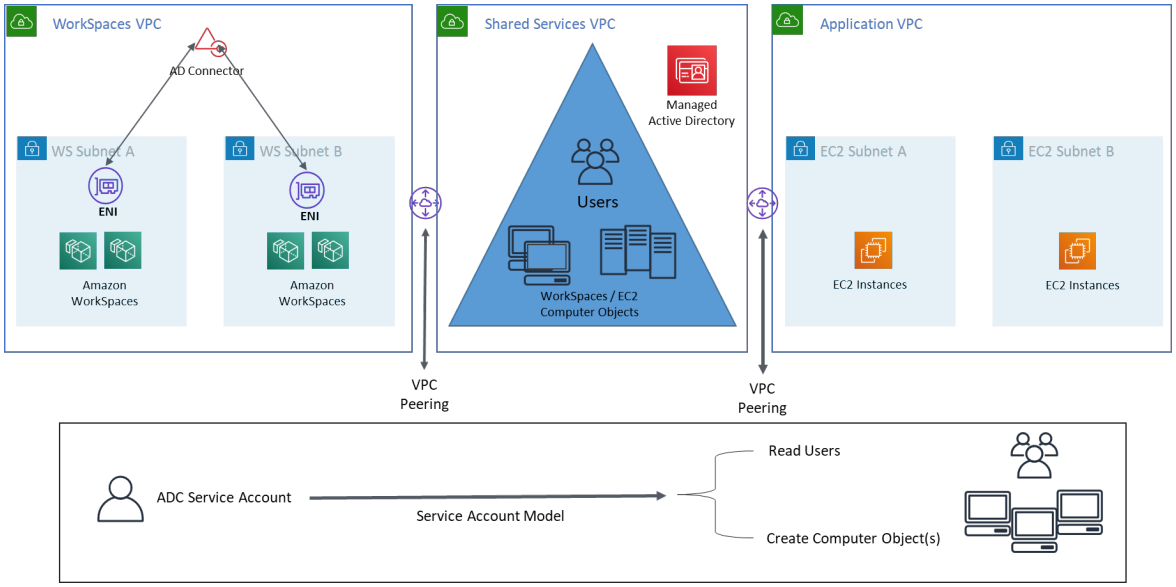

Figure 9 — AWS Microsoft AD using a Shared Services Virtual Private Cloud (VPC)

This architecture uses the following components or constructs:

AWS

- **Amazon VPC** — Creation of an Amazon VPC with at least two private subnets across two Availability Zones (two for AD Connector and WorkSpaces).
- **DHCP options set** — Creation of an Amazon VPC DHCP options set. This allows a customer to define a specified domain name and DNS(s) (Microsoft AD). For more information, see DHCP Options Sets.
- **Optional: Amazon virtual private gateway** — Enable communication with a customer-owned 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), AD Connector .
- **AWS Transit Gateway/VPC Peering** — enable connectivity between Workspaces VPC and the Shared Services VPC.
- **Amazon EC2** — Customer "Optional" RADIUS Servers for MFA.
- **Amazon WorkSpaces** — WorkSpaces are deployed into the same private subnets as the AD Connector . For more information, see the Active Directory: Sites and Services (p. 25) section of this document.

Customer

- **Network Connectivity** — Corporate VPN or AWS Direct Connect endpoints.
- **End user devices** — Corporate or BYOL end-user devices (such as Windows, Macs, iPads, Android tablets, zero clients, and Chromebooks) used to access the Amazon WorkSpaces service. See this list of client applications for supported devices and web browsers.

Scenario 6: AWS Microsoft AD, Shared Services VPC, and a One-Way Trust to On Premises

This scenario, as shown in Figure 10, uses an existing on-premises Active Directory for user accounts, and introduces a separate Managed Active Directory in AWS Cloud to host the computer objects associated with the WorkSpaces. This scenario allows the computer objects and Active Directory group policies to be managed independently from the corporate Active Directory.

This scenario allows the computer objects and Active Directory group policies to be managed independently from the corporate Active Directory. This scenario is useful when a third party wants to manage WorkSpaces on a customer's behalf, as it allows the third party to define and control the WorkSpaces and policies associated with them, without a need to grant the third-party access to the customer AD.

In this scenario, a specific AD OU is created to organize the WorkSpaces computer objects in the Shared Services AD.

> **Note**
> Amazon Linux WorkSpaces require a two-way trust to be in place for them to be created.

To deploy Windows WorkSpaces with the computer objects created in the Shared Services VPC hosting Managed Active Directory using user accounts from the customer identity domain, deploy an Active Directory Connector (ADC) referencing the corporate AD. Use an ADC service account created in the corporate AD (identity domain) that has delegated permissions to create computer objects in the Organizational Unit (OU) that was configured for the Windows WorkSpaces in the Shared Services Managed AD, and that has read permissions to the corporate Active Directory (identity domain).

To ensure the Domain Locator function is able to authenticate WorkSpaces users in the desired AD Site for the identity domain, name both domain's AD Sites for the Amazon WorkSpaces Subnets identically as per Microsoft's documentation. It is a best practice to have both identity domain and Shared Services domain AD Domain Controllers in the same AWS Region as Amazon WorkSpaces.

For detailed instructions to configure this scenario, review the implementation guide to set up a one-way trust for Amazon WorkSpaces with AWS Directory Services.

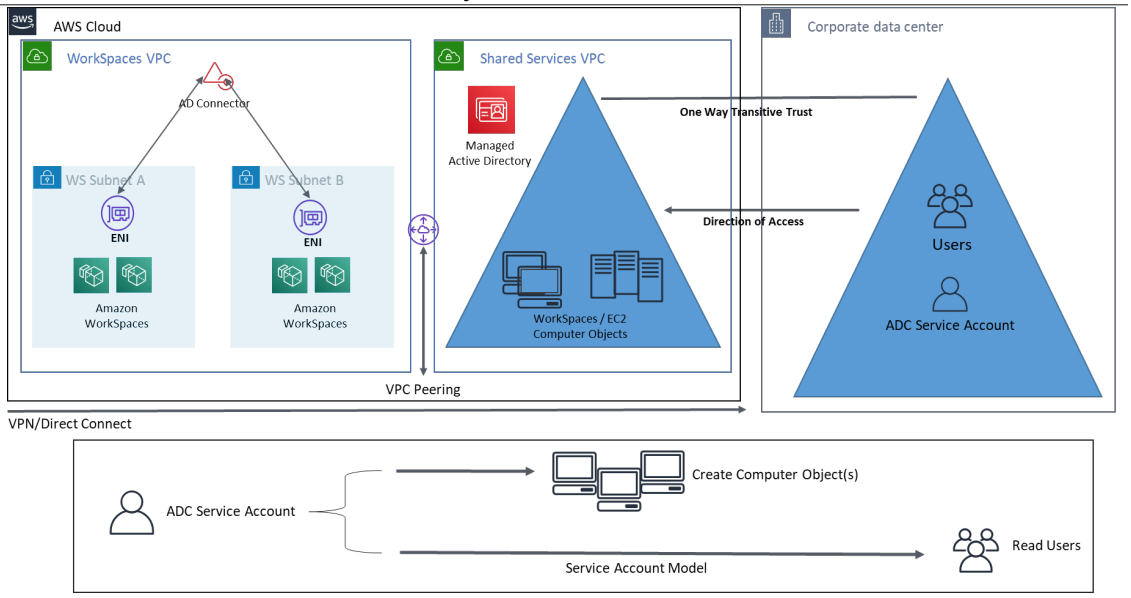

Figure 10 — AWS Microsoft, Shared Services VPC and a one-way trust to AD on-premises

This architecture uses the following components or constructs:

AWS

- **Amazon VPC** — Creation of an Amazon VPC with at least two private subnets across two AZs (two for AD Connector and WorkSpaces).
- **DHCP options set** — Creation of an Amazon VPC DHCP options set. This allows a customer to define a specified domain name and DNS (Microsoft AD). For more information, see DHCP Options Sets.
- **Optional: Amazon virtual private gateway** — Enable communication with a customer-owned 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), AD Connector.
- **Transit Gateway/VPC Peering** — Enable connectivity between Workspaces VPC and the Shared Services VPC.
- **Amazon EC2** — Customer "Optional" RADIUS Servers for MFA.
- **Amazon WorkSpaces** — WorkSpaces are deployed into the same private subnets as the AD Connector. For more information, see the Active Directory: Sites and Services (p. 25) section of this document.

Customer

- **Network Connectivity** — Corporate VPN or AWS Direct Connect endpoints.
- **End user devices** — Corporate or BYOL end-user devices (such as Windows, Macs, iPads, Android tablets, zero clients, and Chromebooks) used to access the Amazon WorkSpaces service. See this list of client applications for supported devices and web browsers.

Design Considerations

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

VPC Design

As discussed in the Network Considerations (p. 4) section of this document and documented earlier for scenarios 2 and 3, customers 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 11 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.

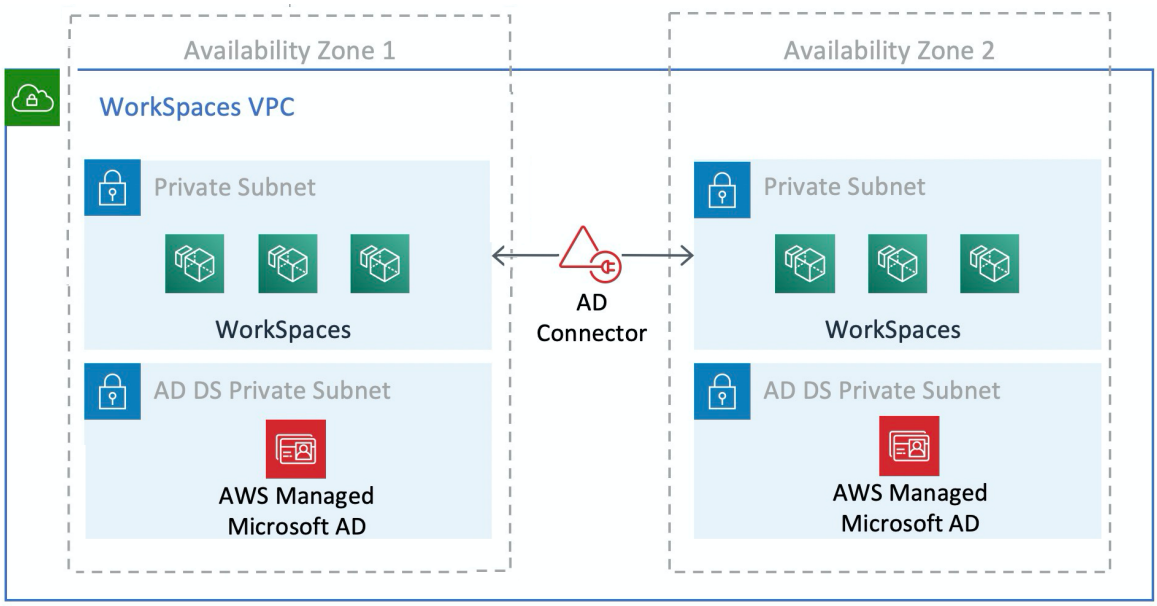

Figure 11 — AD DS network segregation

Figure 12 shows a design similar to scenario 1; however, in this scenario the on-premises portion resides in a dedicated Amazon VPC.

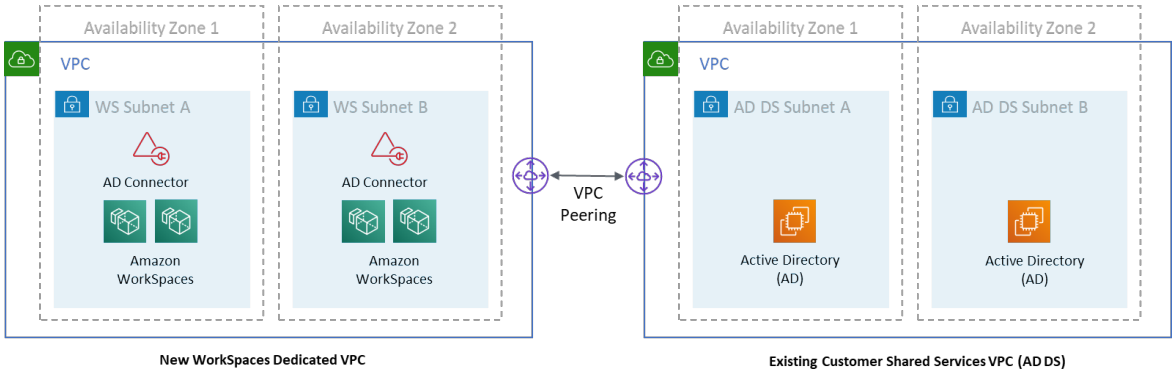

Figure 12 — Dedicated WorkSpaces VPC

> **Note**
> For customers who have an existing AWS deployment where AD DS is being used, AWS recommends that customers locate their WorkSpaces in a dedicated VPC and 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 and from the AWS Cloud, as shown in the following table.

Table 1 — Bidirectional AD DS communications to and from the AWS Cloud

Protocol	Port	Use	Destination
TCP	53, 88, 135, 139, 389, 445, 464, 636	Auth (primary)	Active Directory (private data center or Amazon EC2)*
TCP	49152 – 65535	RPC High Ports	Active Directory (private data center or Amazon EC2)**
TCP	3268-3269	Trusts	Active Directory (private data center or Amazon EC2)*
TCP	9389	Remote Microsoft Windows PowerShell (optional)	Active Directory (private data center or Amazon EC2)*
UDP	53, 88, 123, 137, 138, 389, 445, 464	Auth (primary)	Active Directory (private data center or Amazon EC2)*
UDP	1812	Auth (MFA) (optional)	RADIUS (private data center or Amazon 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 Add 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 customer instances via DHCP. Correct functionality of Windows services within a customer VPC depends on this DHCP scope option. In each of the scenarios defined earlier, customers create and assign their own scope that defines the domain name and name servers. This ensures that domain-joined Windows instances or WorkSpaces are configured to use the AD DNS.

The following table is an example of a custom set of DHCP scope options that must be created for Amazon WorkSpaces and AWS Directory Services to function correctly.

Table 2 — Custom set of DHCP scope options

Parameter	**Value**
Name tag	Creates a tag with key = **name** and **value** set to a specific string Example: example.com
Domain name	example.com
Domain name servers	DNS server address, separated by commas Example: 192.0.2.10, 192.0.2.21
NTP servers	Leave this field blank
NetBIOS name servers	Enter the same comma separated IPs as per domain name servers Example: 192.0.2.10, 192.0.2.21
NetBIOS node type	2

For details about creating a custom DHCP option set and associating it with an 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 scenarios 2 or 3, this would be the locally deployed directory service (AD DS on Amazon EC2 or AWS Directory Services: Microsoft AD). We recommend each domain controller that resides in the AWS Cloud be a global catalog and Directory-Integrated DNS server.

Active Directory: Sites and Services

For Scenario 2: Extending On-Premises AD DS into AWS (Replica) (p. 14) 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 Amazon 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.

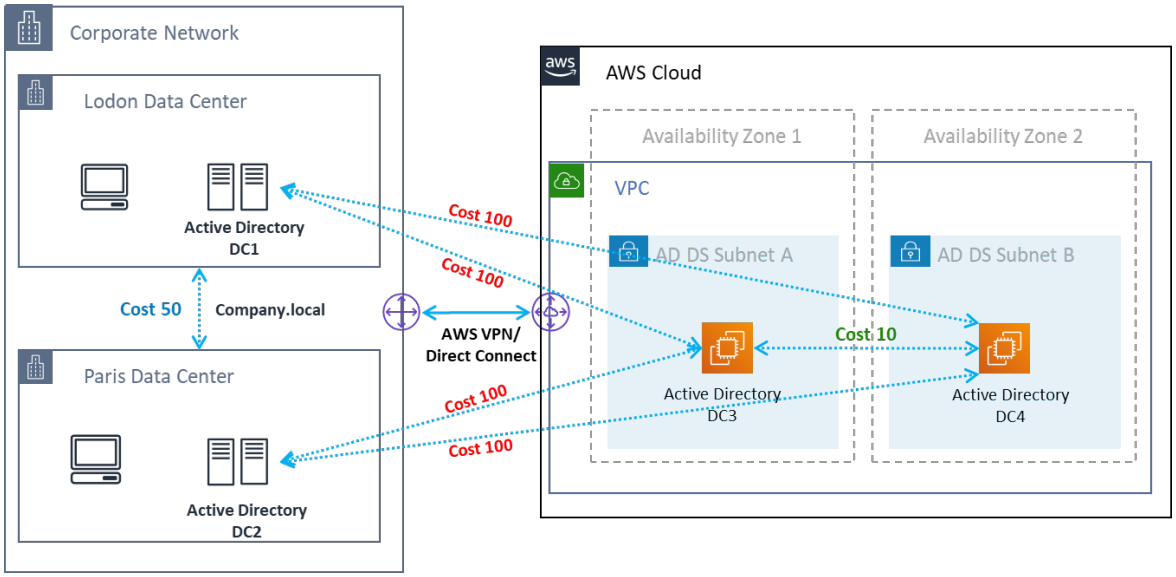

Figure 13 — Active Directory sites and services: client affinity

> **Best practice**
> Define high cost for site links between on-premises AD DS and the AWS Cloud. Figure 13 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.

Protocol

Amazon WorkSpaces Streaming Protocol (WSP) is a cloud-native streaming protocol that enables a consistent user experience across global distances and unreliable networks. WSP decouples the protocol from the WorkSpaces by offloading metric analysis, encoding, codec usage and selection. WSP uses port TCP/UDP 4195. When deciding whether or not use the WSP protocol, there are several key questions that should be answered prior to deployment. Please refer to the decision matrix below:

Table 3: decision matrix

Question	WSP	PCoIP
Will the identified WorkSpaces users need bi-directional audio / video?	X	
Will zero clients be used as the remote endpoint (local device)?		X
Will Windows or macOS be used for remote endpoint?	X	X
Will Ubuntu 18.04 be used for remote endpoint?		X

Question	WSP	PCoIP
Will the users access Amazon WorkSpaces via web access?		X
Is pre-session or in-session smart card support (PIC / CAC) needed?	X	
Will WorkSpaces be used in China (Ningxia) Region?		X
Will smart card pre-authentication or in-session support be required?	X	
Will smart card pre-authentication or in-session support be required?	X	

The aforementioned questions are critical to determine the protocol that should be used. For additional information, see Protocols for Amazon WorkSpaces. The protocol used can also be changed at a later time using the Amazon WorkSpaces Migrate feature. For more information on the use of this feature, see Migrate a WorkSpace.

When deploying WorkSpaces using WSP, the WSP Gateways should be added to an allow list to ensure connectivity to the service. Additionally, users connecting to a WorkSpaces using WSP, the round-trip time (RTT) should be under 250ms for best performance. Connections with an RTT between 250ms and 400ms will be degraded. If the user's connection is consistently degraded, we recommend deploying an Amazon WorkSpaces in a service-supported region closest to the end-user, if possible.

Multi-Factor Authentication (MFA)

Implementing MFA requires the Amazon 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 (p. 12), describes 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. AFter MFA is enabled, users are required to provide their **Username**, **Password**, and **MFA Code** to the WorkSpaces client for authentication to their respective WorkSpaces desktops.

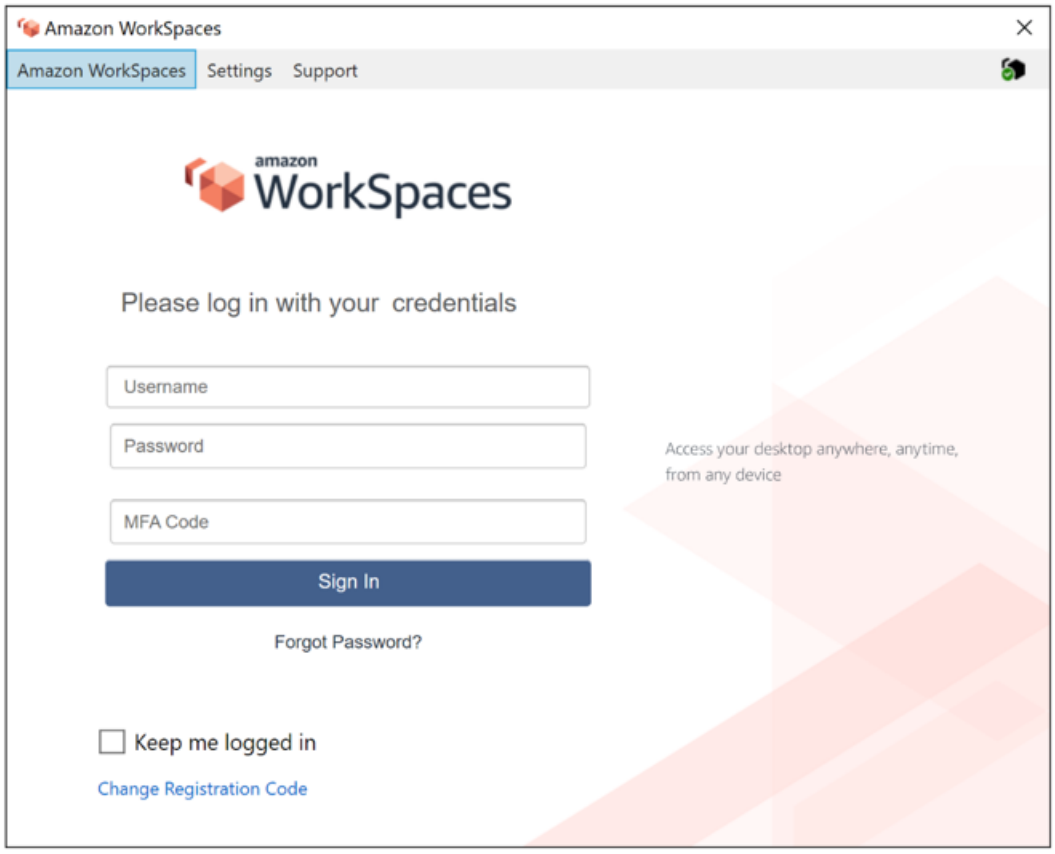

Figure 14 — WorkSpaces client with MFA enabled

Hard rule
Implementing MFA authentication requires customers to use AD Connector. AD Connector doesn't support selective "per user" MFA, as this is a global per AD Connector setting. If selective "per user" MFA is required, users must be separated by an 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 a VPC (see the AD DS Deployment Scenarios (p. 12) section of this document). If you are deploying a new RADIUS solution, several implementations exist, such as FreeRADIUS, and cloud services, such as Duo Security.

For a list of prerequisites to implement MFA with Amazon WorkSpaces, see Launch a WorkSpace Using AD Connector in the Amazon WorkSpaces Administration Guide. The process for configuring AD Connector for MFA is described in Managing an AD Connector Directory: Multi-factor Authentication in the Amazon WorkSpaces Administration Guide.

Disaster Recovery / Business Continuity

WorkSpaces Cross-Region Redirection

Amazon WorkSpaces is a regional service that provides remote desktop access to customers. Depending on business continuity and disaster recovery requirements (BC/DR), some customers require seamless failover to another region where the Amazon WorkSpaces service is available. This BC/DR requirement can be accomplished using the Amazon WorkSpaces Cross-Region redirection option. It allows customers to use a fully qualified domain name (FQDN) as their Amazon WorkSpaces registration code.

When your end users log in to WorkSpaces, you can redirect them across Amazon WorkSpaces Regions based on your DNS policies for the FQDN. This option can be used with public or private DNS zones. Cross-region failure can be manual or automated. The automated failover can be done by using DNS health checks to determine if a primary site is still available before failing over to the second region. If you don't have DNS health checks, you can create a TXT record within your managed DNS service.

An important consideration is to determine at what point a failure to a failover region should occur. The criteria for this decision should be based on your company policy, but should include the Recovery Time Objective (RTO) and the Recovery Point Objective (RPO). A Well-Architected Amazon WorkSpaces architecture design should include the potential for service failure. The time tolerance for normal business operation recovery will also factor into the decision.

Additionally, with cross-region redirection, user data replication to the new region should be considered. There are many options available for user data replication such as Amazon WorkDocs, Windows FSx (DFS Share), or 3rd party utilities to synchronize data volumes between regions. For more information, see Cross-Region Redirection for Amazon WorkSpaces.

WorkSpaces Interface VPC Endpoint (AWS PrivateLink) – API Calls

Amazon WorkSpaces public APIs are supported on AWS PrivateLink. AWS PrivateLink increases the security of data shared with cloud-based applications by reducing the exposure of data to the public internet. WorkSpaces API traffic can be secured inside a VPC by using an interface endpoint, which is an elastic network interface with a private IP address from the IP address range of your subnet that serves as an entry point for traffic destined to a supported service. This enables you to privately access WorkSpaces API services by using private IP addresses.

Using PrivateLink with WorkSpaces Public APIs also enables you to securely expose REST APIs to resources only within your VPC or to those connected to your data centers via AWS Direct Connect.

> **Note**
> You can restrict access to selected Amazon VPCs and VPC Endpoints, and enable cross account access using resource-specific policies.

Ensure that the security group that is associated with the endpoint network interface allows communication between the endpoint network interface and the resources in your VPC that communicate with the service. If the security group restricts inbound HTTPS traffic (port 443) from resources in the VPC, you might not be able to send traffic through the endpoint network interface. An interface endpoint supports TCP traffic only.

- Endpoints support IPv4 traffic only.
- When you create an endpoint, you can attach an endpoint policy to it that controls access to the service to which you are connecting.
- You have a quota on the number of endpoints you can create per VPC.
- Endpoints are supported within the same Region only. You cannot create an endpoint between a VPC and a service in a different Region.

Create Notification to receive alerts on interface endpoint events — You can create a notification to receive alerts for specific events that occur on your interface endpoint. To create a notification, you must associate an Amazon SNS topic with the notification. You can subscribe to the SNS topic to receive an email notification when an endpoint event occurs.

Create a VPC Endpoint Policy for Amazon WorkSpaces — You can create a policy for Amazon VPC endpoints for Amazon WorkSpaces to specify the following:

- The principal that can perform actions.
- The actions that can be performed.

- The resources on which actions can be performed.

Connect Your Private Network to Your VPC — To call the Amazon WorkSpaces API through your VPC, you have to connect from an instance that is inside the VPC, or connect your private network to your VPC by using an Amazon Virtual Private Network (VPN) or AWS Direct Connect. For information about Amazon VPN, see VPN connections in the Amazon Virtual Private Cloud User Guide. For information about AWS Direct Connect, see Creating a connection in the AWS Direct Connect User Guide.

For more information about using Amazon WorkSpaces API through a VPC interface endpoint, see Infrastructure Security in Amazon WorkSpaces.

Automating Amazon Workspaces

With Amazon WorkSpaces, you can launch a Microsoft Windows or Amazon Linux desktop within minutes, and connect to and access your desktop software from on-premises or an external network securely, reliably, and quickly. You can automate the provisioning of Amazon WorkSpaces to enable you to integrate Amazon WorkSpaces into your existing provisioning workflows.

Smart Card Support

Smart card support is available for both Microsoft Windows and Amazon Linux WorkSpaces support Common Access Card (CAC) and Personal Identity Verification (PIV) smart cards through WorkSpaces Streaming Protocol (WSP). Smart card support on WSP WorkSpaces offers an increased security posture for authenticating users on approved connecting endpoints with specific hardware in the form of smart card readers. It is important to first become familiar with the scope of support available for smart cards, and determining how smart cards would function in existing and future WorkSpaces deployments.

It is a best practice to determine which type of smart card support is required, pre-session authentication or in-session authentication, with the former only available at the time of this writing in AWS GovCloud (US-West), and the latter generally available with some considerations, such as:

- Does your organization possess smart card infrastructure integrated with your Windows Active Directory?
- Is your Online Certificate Status Protocol (OCSP) responder public-internet accessible?
- Are user certificates issued with User Principal Name (UPN) in the Subject Alternative Name (SAN) field?
- More considerations are detailed throughout the smart card section of this document.

Smart card support is enabled through Group Policy. It is a best practice to add the Amazon WorkSpaces Group Policy administrative template for WSP to the Central Store of your Active Directory domain used by Amazon WorkSpaces directories. When applying this policy to an existing Amazon WorkSpaces deployment, all WorkSpaces will require the Group Policy update and a reboot for the change to take effect for all users as it is a computer-based policy.

Root CA

The nature of the portability of Amazon WorkSpaces client and user necessitates the requirement to remotely deliver third-party root CA certificate to the trusted root certificate store of each device users use to connect to their Amazon WorkSpaces. AD Domain Controllers and user devices with smart cards must trust the root CAs. Review the guidelines provided by Microsoft for enabling third-party CAs for more information on the exact requirements

In AD Domain-joined environments, these devices meet this requirement through Group Policy distributing root CA certificates. In scenarios where Amazon WorkSpacesclient is used from non-domain-

joined devices, an alternate delivery method for the third-party root CAs must be determined. Microsoft offers support for distributing root CAs through Intune as well.

In-session

In-session authentication simplifies and secures application authentication after Amazon WorkSpaces user sessions have already started. As mentioned previously, the default behavior for Amazon WorkSpaces disables smart cards and must be enabled through Group Policy. Therefore, less configuration is required to enable in-session authentication, as no changes need to be made to AD connectors and directories, as opposed to pre-session authentication.

Most common applications requiring in-session authentication support are through web browsers such as Mozilla Firefox and Google Chrome. While Mozilla Firefox requires limited configuration for in-session smart card support, Google Chrome functions as expected with no configuration change on Microsoft Windows WSP WorkSpaces. Amazon Linux WSP WorkSpaces requires additional configuration for in-session smart card support for both Mozilla Firefox and Google Chrome.

It is a best practice to ensure the root CAs are loaded in the user's Personal certificate store before troubleshooting, as the Amazon WorkSpaces client may not have permissions to the local computers. Use OpenSC to identify smart card devices when troubleshooting any suspected in-session authentication issues with smart cards.

While an Online Certificate Status Protocol (OCSP) Responder is required for pre-session authentication, it is generally recommended for in-session authentication. OCSP Responder improves the security posture of application authentication through a certificate revocation check.

Pre-session

Pre-session authentication with smart cards is fundamentally different than standard authentication, requiring the user to authenticate through a combination of both inserting the smart card and entering a PIN code. With this authentication type, the duration of users' sessions is bounded by the lifetime of the Kerberos ticket. See Access Amazon WorkSpaces with Common Access Cards for a full installation guide.

The following is an overview of pre-session authentication.

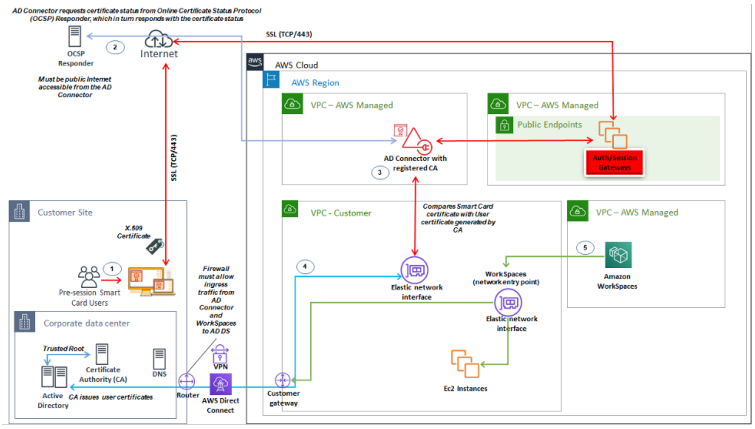

Figure 15 — Overvivew of pre-session authentication

1. The user opens Amazon WorkSpaces Client, inserts smart card, and enters their PIN. The PIN is used by Amazon WorkSpaces client to decrypt the X.509 certificate, which is then proxied to the AD Connector through the Authentication Gateway.
2. AD Connector validates the X.509 certificate against the publicly accessible OCSP Responder URL specified in the directory settings to ensure the certificate has not been revoked.

3. If the certificate is valid, the Amazon WorkSpaces client continues the authentication process by prompting the user to enter their PIN a second time to decrypt the X.509 certificate and proxy to the AD Connector, where it is then matched with the AD Connector's root and intermediary certificates for validation.
4. Once the validation of the certificate is successfully matched, Active Directory is used by the AD Connector to authenticate the user and a Kerberos ticket is created.
5. The Kerberos ticket is passed to the user's Amazon WorkSpace to authenticate and begin the WSP session.

> **Note**
> The OCSP responder must be publicly accessible, as connection is performed through the AWS-managed network and not the customer-managed network. There is no routing to private networks in this step.

Entering the username is not required, as the user certificates presented to AD Connector includes the `userPrincipalName` (UPN) of the user in the `subjectAltName` (SAN) field of the certificate. It is a best practice to automate all users that require pre-session authentication with smart cards have their AD user objects updated to authenticate with anticipated UPN in the certificate using PowerShell, rather than perform this individually in Microsoft Management Consoles.

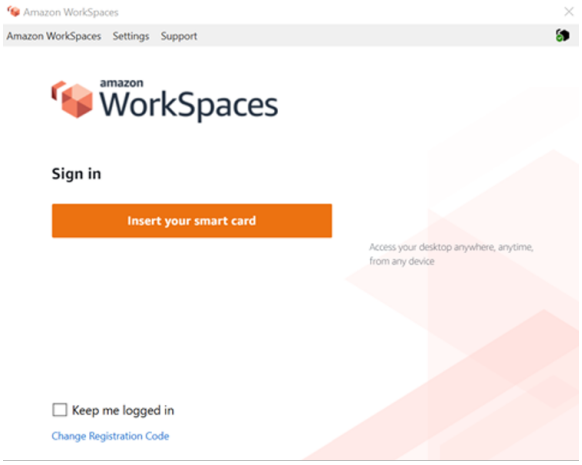

Figure 16 — Amazon Workspaces smart card sign-in screen

While not a technical requirement for pre-session smart card support, enabling Federal Information Processing Standards (FIPS) endpoint encryption at the Amazon WorkSpaces Directory is commonly required for authorization and compliance requirements in addition to smart cards.

Client Deployment

The Amazon WorkSpaces Client (version 3.X+) uses standardized configuration files which can be leveraged by administrators to preconfigure their user's WorkSpaces client. The path for the two main configuration files can be found at:

Table 4 — Configuration file paths

OS	Configuration File Path
Windows	`C:\Users\USERNAME\AppData\Local\Amazon Web Services\Amazon WorkSpaces`

OS	Configuration File Path
macOS	`/Users/USERNAME/Library/Application Support/Amazon Web Services/Amazon WorkSpaces`
Linux (Ubuntu 18.04)	`/home/ubuntu/.local/share/Amazon Web Services/Amazon WorkSpaces/`

Within these paths, you will find the two configuration files. The first configuration file is `UserSettings.json`, which will set things like current registration, proxy configuration, logging level, and the ability to save the registration list. The second configuration file is `RegistrationList.json`. This file will contain all of the WorkSpaces directory information for the client to use to map to the correct WorkSpaces directory. Preconfiguring the `RegistrationList.json` file populates all of the registration codes within the client for the user.

> **Note**
> If your users are running WorkSpaces client version 2.5.11, `proxy.cfg` will be used for client proxy settings and `client_settings.ini` will set log level as well as the ability to save the registration list. The default proxy setting will use what is set within the OS.

Since these files are standardized, administrators can download the WorkSpaces Client, set all of the applicable settings, and then push out the same configuration files to all of the end users. For the settings to take effect, the client must be started after the new configurations are set. If you change the configuration while the client is running, none of the changes will be set within the client.

The last setting that can be set for WorkSpaces users is Windows client auto update. This is not controlled via configuration files but the Windows registry instead. When a new version of the client comes out, you can create a registry key to skip that version. This can bet set by creating a string registry entry named `SkipThisVersion` with a value of the full version number in the following path:

```
Computer\HKEY_CURRENT_USER\Software\Amazon Web Services. LLC\Amazon WorkSpaces
\WinSparkle
```

This option is also available for macOS; however, the configuration is within a PLIST file which requires special software to edit. If you would still like to perform this action, it can be done by adding a `SUSkippedVersion` entry within the `com.amazon.workspaces` domain located at:

```
/Users/USERNAME/Library/Preferences
```

Endpoint Selection

Choosing an endpoint for your Workspaces

Amazon WorkSpaces provides support for multiple endpoint devices, from Windows desktops, to iPads, and Chromebooks. You can download the available WorkSpaces clients from the Amazon Workspaces website. Choosing the right endpoint for your users is an important decision. Here are some considerations to assist you in choosing an endpoint device:

- **Windows** — To utilize the Windows WorkSpaces client, the client must run the Windows 7, Windows 8, or Windows 10 desktop Operating System. Users can install the client for just their user profile without administrative privileges on the local machine. System administrators can deploy the client to managed endpoints with Group Policy, Microsoft Endpoint Manager Configuration Manager (MEMCM), or other application deployment tools used in an environment. The Windows client supports up to two 4K resolution monitors or four WUXGA (1920 x 1200) resolution monitors. Ensure that the IP addresses and ports listed here have been explicitly configured to ensure the client can connect to the WorkSpaces service. You can also deploy the client with endpoint configuration tools.

- **macOS** — To deploy the latest WorkSpaces client, macOS devices must run macOS 10.12 (Sierra) or later. You can deploy an older version of the WorkSpaces client to connect to PCoIP WorkSpaces if the endpoint is running OSX 10.8.1 or later. The macOS client supports up to two 4K resolution monitors or four WUXGA (1920 x 1200) resolution monitors. Additionally, ensure that the IP addresses and ports listed in IP Address and Port Requirements for Amazon WorkSpaces have been explicitly configured to ensure the client can connect to the service.

- **Linux** — The Amazon WorkSpaces Linux client requires 64-bit Ubuntu 18.04 (AMD64) to run. If your Linux endpoints do not run this OS version, the Linux client is not supported. Before you deploy Linux clients or provide users with their registration code, ensure that you enable Linux client access at the WorkSpaces directory level, as this is disabled by default and users will not be able to connect from Linux clients until it is enabled. The Linux client supports up to two 4K resolution monitors or four WUXGA (1920 x 1200) resolution monitors. Additionally, ensure that the IP addresses and ports listed in IP Address and Port Requirements for Amazon WorkSpaces have been explicitly configured to ensure the client can connect to the service.

- **iPad** — The Amazon WorkSpaces iPad client application supports PCoIP WorkSpaces. The iPads that are supported are the iPad2 or later with iOS 8.0 or later, iPad Retina with iOS 8.0 and later, iPad Mini with iOS 8.0 and later, and the iPad Pro with iOS 9.0 and later. Ensure the device the users will connect from meets those criteria. The iPad client application supports many different gestures. (See a full list of the supported gestures.) The WorkSpaces iPad client application also supports the Swiftpoint GT, ProPoint, and PadPoint mice. The Swiftpoint TRACPOINT, PenPoint and GoPoint mice are not supported. Additionally, ensure that the IP addresses and ports listed in IP Address and Port Requirements for Amazon WorkSpaces have been explicitly configured to ensure the client can connect to the service.

- **Android / Chromebook** — When looking to deploy an Android device or Chromebook as the endpoint for your end users, there are a few considerations that must be taken into account. Ensure the WorkSpaces the users will be connecting to are PCoIP WorkSpaces, as this client only supports PCoIP WorkSpaces. This client only supports a single display. If users require multi-monitor support, utilize a different endpoint.

  If you want to deploy a Chromebook, ensure that the model you deploy supports installing Android applications. Full feature support is supported only on the Android client, and not the legacy Chromebook client. This typically is only a consideration for Chromebooks made prior to 2019.

  Android support is provided for both tablets and phones as long as the Android is running OS 4.4 and later. However, it is recommended that the Android device runs OS 9 or above to utilize the latest WorkSpace Android client. Additionally, ensure that the IP addresses and ports listed in IP Address and Port Requirements for Amazon WorkSpaces have been explicitly configured to ensure the client can connect to the service.

- **Web Access** — Users can access their Windows WorkSpaces from any location using a web browser. This is ideal for users who must use a locked-down device or restrictive network. Instead of using a traditional remote access solution and installing the appropriate client application, users can visit the website to access their work resources. Users can utilize the WorkSpaces Web Access to connect to non-graphics-based Windows PCoIP WorkSpaces running Windows 10 or Windows Server 2016 with Desktop Experience. Users must connect using Chrome 53 or later, or Firefox 49 or later. The minimum supported screen resolution is 960 x 720 with a maximum supported resolution of 2560 x 1600. Multiple monitors are not supported. For the best user experience, when possible, we recommend users use an OS version of the client.

- **PCoIP Zero Client** — You can deploy PCoIP zero clients to end users that have or will have PCoIP WorkSpaces assigned to them. The Tera2 zero client must have a firmware version of 6.0.0 or later

to connect directly to the WorkSpace. To use multi-factor authentication with WorkSpaces, the Tera2 zero client device must run firmware version 6.0.0 or later. Support and troubleshooting of the zero-client hardware should be done with the manufacturer. AWS Premium Support can assist with troubleshooting the WorkSpace itself.

- **IGEL OS** — You can utilize IGEL OS on endpoint devices to connect to PCoIP based WorkSpaces as long as the firmware version is 11.04.280 or above. The supported features match that of the existing Linux client today. Before you deploy IGEL OS clients or provide users with their registration code, ensure you enable Linux client access at the WorkSpaces directory level as this is disabled by default and users will not be able to connect from IGEL OS clients until it is enabled. The lGEL OS client supports up to two 4K resolution monitors or four WUXGA (1920x1200) resolution monitors. Additionally, ensure that the IP addresses and ports listed in IP Address and Port Requirements for Amazon WorkSpaces have been explicitly configured to ensure the client can connect to the service.

Web Access Client

Designed for locked-down devices, the Web Access client delivers access to Amazon WorkSpaces without the need for deploying client software. The Web Access client is recommended only in settings where the Amazon WorkSpaces are Windows Operating System (OS) and are used for limited user workflows, such as a kiosk environment. Most use cases benefit from the feature set available from the full Amazon WorkSpaces client. The Web Access client is only recommended in specific use cases where devices and network restrictions require an alternative connection method.

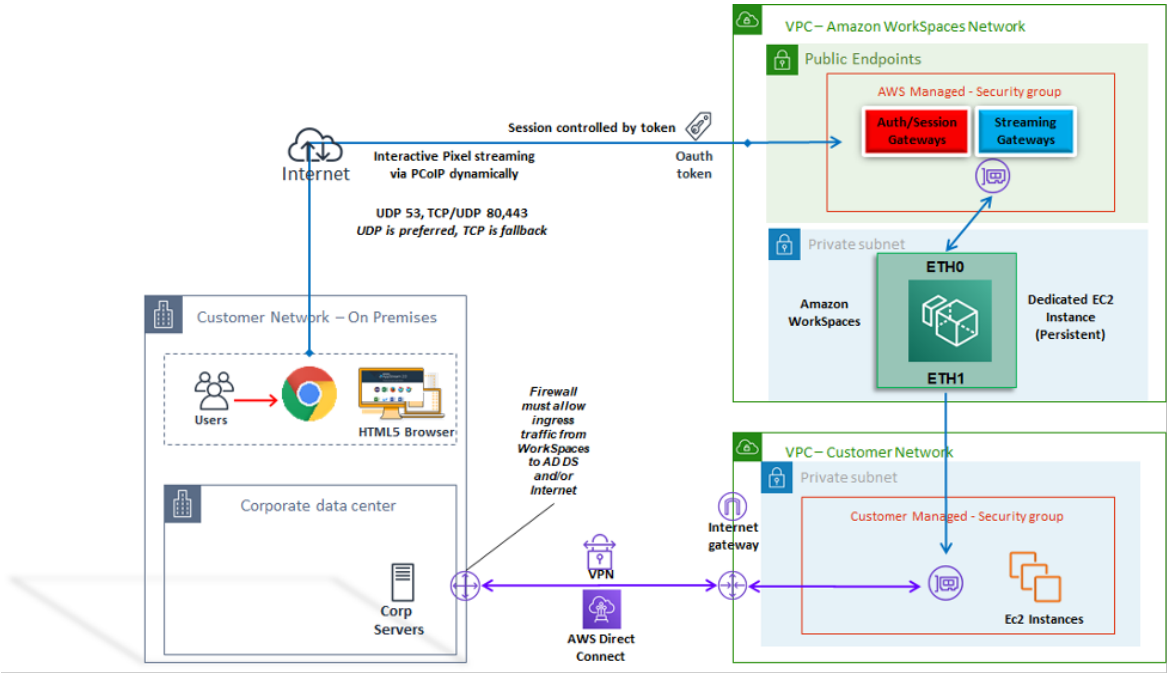

Figure 17 — Web access client architecture

As shown in the diagram, The Web Access client has different network requirements to stream PCoIP to users. DNS and HTTP/HTTPS are required for authentication and registration with the WorkSpaces gateways. Streaming traffic is not allocated to a fixed port as it is with the full Amazon WorkSpaces client; instead, it is dynamic allocated. UDP is preferable for streaming traffic; however, the web browser will fall back to TCP when UDP is restricted. In environments where TCP/UDP port 4172 is blocked and cannot be unblocked due to organizational restrictions, the Web Access client provides an alternative connection method for users.

By default, the Web Access client is disabled at the Directory level. To enable users to access their Amazon WorkSpaces through a web browser, either use the AWS Management Console to update the Directory settings, or programmatically using the WorkspaceAccessProperties API to modify `DeviceTypeWeb` to `Allow`. Additionally, the administrator must ensure Group Policy settings do not conflict with logon requirements.

Amazon Workspaces Tags

Tags enable you to associate metadata with AWS resources. Tags can be used with Amazon WorkSpaces to registered directories, bundles, IP Access Control Groups, or images. Tags assist with cost allocation to internal cost centers. Before using tags with Amazon WorkSpaces, review the *Tagging Best Practices* whitepaper.

Tag Restrictions

- Maximum number of tags per resource—50
- Maximum key length—127 Unicode characters
- Maximum value length—255 Unicode characters
- Tag keys and values are case-sensitive. Allowed characters are letters, spaces, and numbers representable in UTF-8, plus the following special characters: + - = . _ : / @. Do not use leading or trailing spaces.
- Do not use the "`aws:`" or "`aws:workspaces:`" prefixes in your tag names or values because they are reserved for AWS use. You can't edit or delete tag names or values with these prefixes.

Resources that You can Tag

You can add tags to the following resources when you create them:

- WorkSpaces
- Imported images
- IP access control groups

You can add tags to existing resources of the following types:

- WorkSpaces
- Registered directories
- Custom bundles
- Images
- IP access control groups

Using the Cost Allocation Tag

To view your WorkSpaces resource tags in the Cost Explorer, activate the tags that you have applied to your WorkSpaces resources by following the instructions in Activating User-Defined Cost Allocation Tags in the AWS Billing and Cost Management User Guide.

Although tags appear 24 hours after activation, it can take four to five days for values associated with those tags to appear in the Cost Explorer. To appear and provide cost data in Cost Explorer, WorkSpaces

resources that have been tagged must incur charges during that time. Cost Explorer shows only cost data from the time when the tags were activated forward. No historical data is available at this time.

Managing Tags

To update the tags for an existing resource using the AWS CLI, use the create-tags and delete-tags commands. For bulk updates and to automate the task on a large number of WorkSpaces resource, Amazon WorkSpaces adds support for AWS Resource Groups Tag Editor. AWS Resource Groups Tag Editor enables you to add, edit, or delete AWS tags from your WorkSpaces along with your other AWS resources.

Automating Amazon Workspaces Deployment

With Amazon WorkSpaces, you can launch a Microsoft Windows or Amazon Linux desktop within minutes, and connect to and access your desktop software from on-premises or an external network securely, reliably, and quickly. You can automate the provisioning of Amazon WorkSpaces to enable you to integrate Amazon WorkSpaces into your existing provisioning workflows.

Common WorkSpaces Automation Methods

Customers can use a number of tools to allow for rapid Amazon WorkSpaces deployment. The tools can be used to allow simplify management of WorkSpaces, reduce costs and enable an agile environment that can scale and move fast.

AWS CLI and API

There are Amazon WorkSpaces API operations you can use to interact with the service securely, and at scale. All public APIs are available with the AWS CLI SDK and Tools for PowerShell, while private APIs such as image creation are available only through the AWS Console. When considering operational management and business self-service for Amazon WorkSpaces consider that WorkSpaces APIs *do* require technical expertise and security permissions to use.

API calls can be made using the AWS SDK. AWS Tools for Windows PowerShell and AWS Tools for PowerShell Core are PowerShell modules built on functionality exposed by the AWS SDK for .NET. These modules enable you to script operations on AWS resources from the PowerShell command line, and integrate with existing tools and services. For example, API calls can enable you to automatically manage the WorkSpaces lifecycle by integrating with AD to provision and decommission WorkSpaces based on a user's AD group membership.

AWS CloudFormation

AWS CloudFormation enables you to model your entire infrastructure in a text file. This template becomes the single source of truth for your infrastructure. This helps you to standardize infrastructure components used across your organization, enabling configuration compliance and faster troubleshooting.

AWS CloudFormation provisions your resources in a safe, repeatable manner, enabling you to build and rebuild your infrastructure and applications. You can use CloudFormation to commission and decommission environments, which is useful when you have a number of accounts that you want to build and decommission in a repeatable fashion.

When considering operational management and business self-service for Amazon WorkSpaces, consider that AWS CloudFormation *does* require technical expertise and security permissions to use.

Self-Service WorkSpaces Portal

Customers can use build on WorkSpaces API commands and other AWS Services to create a WorkSpaces self-service portal. This helps customers streamline the process to deploy and reclaim WorkSpaces at scale. Using a WorkSpaces portal, you can enable your workforce to provision their own WorkSpaces with an integrated approval workflow that does not require IT intervention for each request. This reduces IT operational costs, while helping end-users get started with WorkSpaces faster. The additional built-in approval workflow simplifies the desktop approval process for businesses. A dedicated portal can offer an automated tool for provisioning Windows or Linux cloud desktops, and enable users to rebuild, restart, or migrate their WorkSpace, as well as provide a facility for password resets.

There are guided examples of creating Self Service WorkSpaces Portals referenced in the Further Reading (p. 63) section of this document. AWS Partners provide preconfigured WorkSpaces management portals via the AWS Marketplace.

Integration with Enterprise IT Service Management

As enterprises adopt Amazon WorkSpaces as their virtual desktop solution at scale, there is a need to implement, or integrate with, IT Service Management (ITSM) systems. ITSM integration allows for self-service offerings for provisioning and operations. The AWS Service Catalog enables you to manage commonly deployed AWS services and provisioned software products centrally. This service helps your organization achieve consistent governance and compliance requirements, while enabling users to deploy only the approved AWS services they need. The AWS Service Catalog can be used to enable a self-service lifecycle-management offering for Amazon WorkSpaces from within IT Service Management tools such as ServiceNow.

WorkSpaces Deployment Automation Best Practices

You should consider Well Architected principles of selecting and designing WorkSpaces deployment automation:

- **Design for Automation** — Design to deliver the least possible manual intervention in the process to enable repeatability and scale.
- **Design for Cost Optimization** — By automatically creating and reclaiming WorkSpaces, you can reduce the administration effort needed to provide resources and remove idle or unused resources from generating unnecessary cost.
- **Design for Efficiency** — Minimize the resources needed to create and terminate WorkSpaces. Where possible, provide Tier 0 self-service capabilities for the business to improve efficiency.
- **Design for Flexibility** — Create a consistent deployment mechanism that can handle multiple scenarios, and can scale with the same mechanism (customized using tagged use case and profile identifiers).
- **Design for Productivity** — Design your WorkSpaces operations to allow for the correct authorization and validation to add or remove resources.
- **Design for Scalability** — The pay-as-you go model that Amazon WorkSpaces uses can drive cost savings by creating resources as needed, and removing them when they are no longer necessary.
- **Design for Security** — Design your WorkSpaces operations to allow for the correct authorization and validation to add or remove resources.
- **Design for Supportability** — Design your WorkSpaces operations to allow for non-invasive support and recovery mechanisms and processes.

Amazon Workspaces Language Packs

Amazon WorkSpaces bundles that provide the Windows 10 desktop experience supports English (US), French (Canadian), Korean, and Japanese. However, you can include additional language packs for Spanish, Italian, Portuguese, and many more language options. For more information, see How do I create a new Windows WorkSpace image with a client language other than English?.

Amazon WorkSpaces Profile Management

Amazon WorkSpaces separates the user profile from the base Operating System (OS) by redirecting all profile writes to a separate Amazon Elastic Block Store (Amazon EBS) volume. In Microsoft Windows, the user profile is stored in `D:\Users\username`.

In Amazon Linux, the user profile is stored in `/home`. The EBS volume is snapshotted automatically every 12 hours. The snapshot is automatically stored in a nanaged Amazon Simple Storage Service (Amazon S3) bucket, to be used in the event that an Amazon WorkSpace is rebuilt or restored.

For most organizations, having automatic snapshots every 12 hours is superior to the existing desktop deployment of no backups for user profiles. However, customers can require more granular control over user profiles; for example, migration from desktop to WorkSpaces, to a new OS/AWS Region, support for DR, and so on. There are alternative methods for profile management available for Amazon WorkSpaces.

Folder Redirection

While folder redirection is a common design consideration in Virtual Desktop Infrastructure (VDI) architectures, it is not a best practice, or even a common requirement in Amazon WorkSpaces designs. The reason for this is Amazon WorkSpaces is a persistent Desktop as a Service (DaaS) solution, with application and user data persisting out of the box.

There are specific scenarios where Folder Redirection for User Shell Folders (for example, `D:\Users\username\Desktop` redirected to `\\Server\RedirectionShare$\username\Desktop`) are required, such as immediate recovery point objective (RPO) for user profile data in disaster recovery (DR) environments.

Best Practices

The following best practices are listed for a robust folder redirection:

- Host the Windows File Servers in the same AWS Region and AZ that the Amazon WorkSpaces are launched in.
- Ensure AD Security Group Inbound Rules include the Windows File Server Security Group or private IP addresses; otherwise ensure that the on-premises firewall allows those same TCP and UDP port-based traffic.
- Ensure Windows File Server Security Group Inbound Rules include TCP 445 (SMB) for all Amazon WorkSpaces Security Groups.
- Create an AD Security Group for Amazon WorkSpaces users to authorize users access to the Windows File Share.
- Use DFS Namespace (DFS-N) and DFS Replication (DFS-R) to ensure your Windows File Share is agile, not tied to anyone one specific Windows File Server, and all user data is automatically replicated between Windows File Servers.
- Append '$' to the end of the share name to hide the share hosting user data from view when browsing the network shares in Windows Explorer.

- Create the file share following Microsoft's guidance for redirected folders: Deploy Folder Redirection with Offline Files. Follow the guidance for Security Permissions and GPO configuration closely.
- If your Amazon WorkSpaces deployment is Bring Your Own License (BYOL), you must also specify disabling Offline Files following Microsoft's guidance: Disable Offline Files on Individual Redirected Folders.
- Install and run Data Deduplication (commonly referred to as 'dedupe') if your Windows File Server is Windows Server 2016 or newer to reduce storage consumption and optimize cost. See Install and enable Data Deduplication and Running Data Deduplication.
- Back up your Windows File Server file shares using existing organizational backup solutions.

Things to Avoid

- Do not use Windows File Servers that are accessible only across a wide area network (WAN) connection, as the SMB protocol is not designed for that use.
- Do not use the same Windows File Share that is used for Home Directories to mitigate the chances of users accidentally deleting their User Shell folders.
- While enabling Volume Shadow Copy Service (VSS) is recommended for ease of file restores, this alone does not remove the requirement to back up the Windows File Server file shares.

Other Considerations

- Amazon FSx for Windows File Server offers a managed service for Windows file shares, and simplify the operational overhead of folder redirection, including automatic backups.
- Utilize AWS Storage Gateway for SMB File Share to back up your file shares if there is no existing organizational backup solution.

Profile Settings

Group Policies

A common best practice in enterprise Microsoft Windows deployments is to define user environment settings through Group Policy Object (GPO) and Group Policy Preferences (GPP) settings. Settings such as shortcuts, drive mappings, registry keys, and printers are defined through the Group Policy Management Console. The benefits to defining the user environment through GPOs include, but are not limited to:

- Centralized configuration management
- User profile defined by AD Security Group Membership or OU placement
- Protection against deletion of settings
- Automate profile creation and personalization at first logon
- Ease of future updating

> **Note**
> Follow Microsoft's Best Practices for optimizing Group Policy performance.

Interactive Logon Banners Group Policies must not be used as they are not supported on Amazon WorkSpaces. Banners are presented on the Amazon WorkSpaces Client through AWS support requests. Additionally, removable devices must not be blocked through group policy, as they are required for Amazon WorkSpaces.

GPOs can be used to manage Windows WorkSpaces. For more information, see Manage Your Windows WorkSpaces.

Amazon Workspaces Volumes

Each Amazon WorkSpaces instance contains two volumes: an *operating system* volume and a *user* volume.

- Amazon Windows WorkSpaces — The `C:\` drive is used for the Operating System (OS) and the `D:\` drive is user volume. The user profile is located on the user volume (AppData, Documents, Pictures, Downloads, and so on).
- Amazon Linux WorkSpaces — With an Amazon Linux WorkSpace, the system volume (`/dev/xvda1`) mounts as the root folder. The user volume is for user data and applications; `/dev/xvdf1` mounts as `/home`.

For operating system volumes, you can select a starting size for this drive of 80 GB or 175 GB. For user volumes, you can select a starting size of 10 GB, 50 GB, or 100 GB. Both volumes can be increased up to 2TB in size as needed; however, to increase the user volume beyond 100 GB, the OS volume must be 175 GB. Volume changes can be performed only once every six hours per volume. For additional information on modifying the WorkSpaces volume size, see the Modify a WorkSpace section of the Administration Guide.

WorkSpaces Volumes Best Practices

When planning an Amazon WorkSpaces deployment, we recommend factoring the minimum requirements for OS installation, in-place upgrades, and additional core applications that will be added to the image on the OS volume. For the user volume, we recommend starting with a smaller disk allocation, and incrementally increasing the user volume size as needed. Minimizing the size of the disk volumes reduces the cost of running the WorkSpace.

> **Note**
> While a volume size can be increased, it cannot be decreased.

Amazon WorkSpaces Logging

In an Amazon WorkSpaces environment, there are many log sources that can be captured to troubleshoot issues and monitor the overall WorkSpaces performance.

Amazon WorkSpaces Client 3.x

On each Amazon WorkSpaces client, the client logs are located in the following directories:

- Windows —`%LOCALAPPDATA%\Amazon Web Services\Amazon WorkSpaces\logs`
- macOS — `~/Library/"Application Support"/"Amazon Web Services"/"Amazon WorkSpaces"/logs`
- Linux (Ubuntu 18.04 or later) — `/opt/workspacesclient/workspacesclient`

There are many instances where diagnostic or debugging details may be needed for a WorkSpaces session from the client side. Advanced client logs can be enabled as well by adding an "`-l3`" to the workspaces executable file. For example:

```
"C:\Program Files (x86)\Amazon Web Services, Inc\Amazon WorkSpaces"
```

```
workspaces.exe -l3
```

Amazon WorkSpaces Service

The Amazon WorkSpaces service is integrated with Amazon CloudWatch Metrics, CloudWatch Events, and AWS CloudTrail. This integration allows of the performance data and API calls to be logged into the central AWS service.

When managing an Amazon WorkSpaces environment, it is important to constantly monitor certain CloudWatch metrics to determine the overall environment health status.

Metrics

While there are other CloudWatch metrics available for Amazon WorkSpaces (see Monitor Your WorkSpaces Using CloudWatch Metrics), the three following metrics will assist in maintaining the WorkSpace instance availability:

- **Unhealthy** — The number of WorkSpaces that returned an unhealthy status.
- **SessionLaunchTime** — The amount of time it takes to initiate a WorkSpaces session.
- **InSessionLatency** — The round trip time between the WorkSpaces client and the WorkSpace.

For more information on WorkSpaces logging options, see Logging Amazon WorkSpaces API Calls by Using CloudTrail. The additional CloudWatch Events will assist with capturing the client-side IP of the user session, when the user connected to the WorkSpaces session, and the what endpoint was used during the connection. All of these details assist with isolating or pinpointing user reported issues during troubleshooting sessions.

> **Note**
> Some CloudWatch Metrics are available only with AWS Managed AD.

Amazon Workspaces Migrate

Amazon WorkSpaces migrate feature enables you to bring your user volume data to a new bundle. You can use this feature to:

- Migrate your WorkSpaces from the Windows 7 Experience to the Windows 10 Desktop Experience.
- Migrate from a PCoIP WorkSpace to a WorkSpaces Streaming Protocol (WSP) WorkSpace.
- Migrate WorkSpaces from one public, or custom, bundle to another. For example, you can migrate from GPU-enabled (Graphics and GraphicsPro) bundles to non-GPU-enabled bundles, and vice versa.

Migration Process

With WorkSpaces Migrate, you can specify the target WorkSpaces bundle. The migration process recreates the WorkSpace using a new root volume from the target bundle image, and the user volume from the latest original user volume snapshot. A new user profile is generated during migrate for better compatibility. The data in your old user profile that cannot be moved to the new profile is stored in a `.notMigrated` folder.

During migration, the data on the user volume (drive `D`) is preserved, but all the data on the root volume (`C:\` drive) is lost. This means that none of the installed applications, settings, and changes to the registry are preserved. The old user profile folder is renamed with the `.NotMigrated` suffix, and a new user profile is created.

The migration process takes up to one hour per WorkSpace. In addition, if the migrate workflow fails to complete the process, the service will automatically roll back the WorkSpace to its original state before migration, minimizing any data loss risk.

Any tags assigned to the original WorkSpace are carried over during migration. The running mode of the WorkSpace is preserved. The migrated WorkSpace has a new WorkSpace ID, computer name, and IP address.

Migration Procedure

You can migrate WorkSpaces through the Amazon WorkSpaces console, the AWS CLI using the migrate-workspace command, or the Amazon WorkSpaces API. All migration requests gets queued, and the service will automatically throttle the total number of migration requests if there are too many.

Migration Limits

- You cannot migrate to a public or custom Windows 7 desktop experience bundle. You cannot migrate to BYOL Windows 7 bundles.
- You can migrate BYOL WorkSpaces *only* to other BYOL bundles.
- You cannot migrate a WorkSpace created from public or custom bundles to a BYOL bundle.
- Migrating Linux WorkSpaces is not currently supported.
- In AWS Regions that support more than one language, you can migrate WorkSpaces between language bundles.
- The source and target bundles must be different. (However, in Regions that support more than one language, you can migrate to the same Windows 10 bundle as long as the languages differ.) If you want to refresh your WorkSpace using the same bundle, rebuild the WorkSpace instead.

- You cannot migrate WorkSpaces across Regions.
- Note that WorkSpaces cannot be migrated when they are in `ADMIN_MAINTENANCE` mode.

Cost

During the month in which migration occurs, you are charged prorated amounts for both the new and the original WorkSpaces. For example, if you migrate WorkSpace A to WorkSpace B on May 10, you will be charged for WorkSpace A from May 1 to May 10, and you will be charged for WorkSpace B from May 11 to May 30.

WorkSpaces Migration Best Practices

Before you migrate a WorkSpace, do the following:

- Back up any important data on drive `C` to another location. All data on drive `C` is erased during migration.
- Make sure that the WorkSpace being migrated is at least 12 hours old, to ensure that a snapshot of the user volume has been created. On the **Migrate WorkSpaces** page in the Amazon WorkSpaces console, you can see the time of the last snapshot. Any data created after the last snapshot is lost during migration.
- To avoid potential data loss, make sure that your users log out of their WorkSpaces, and don't log back in until after the migration process is finished.
- Make sure that the WorkSpaces you want to migrate have a status of `AVAILABLE`, `STOPPED`, or `ERROR`.
- Make sure that you have enough IP addresses for the WorkSpaces you are migrating. During migration, new IP addresses will be allocated for the WorkSpaces.
- If you are using scripts to migrate WorkSpaces, migrate them in batches of no more than 25 WorkSpaces at a time.

Well-Architected Framework

AWS Well-Architected helps cloud architects build secure, high-performing, resilient, and efficient infrastructure for their applications and workloads. It describes the key concepts, design principles, and architectural best practices for designing and running workloads in the cloud. It is based on five key pillars:

- Operational Excellence (OE)
- Security
- Reliability
- Performance Efficiency
- Cost Optimization

When architecting an Amazon WorkSpaces environment, it is important to evaluate these key pillars to determine the maturity deployment level, and discover additional features that can be used with the Amazon WorkSpaces. While there is overall guidance for the AWS Well-Architect Framework, AWS provides some key questions that can be included in the planning phase of your WorkSpaces deployment to ensure each of the five pillars are considered.

General

- What is the business driver for this project?

Operational Excellence

- How do you segregate access control between users and different admin groups?

Security

1. What are the security and compliance requirements to be considered for the WorkSpaces to operate in?
2. Are there any restrictions on routing to external IP addresses?
3. Are the required WorkSpaces ports allowed through the corporate firewall?
4. Is or will multi-factor authentication be used with this deployment?
5. How do you many user identities and authorization requests today?

Reliability

- What is the data retention policy for desktops?
- What is the Recovery Point Objective (RPO) for end-user data?
- What is the Recovery Time Objective (RTO) for end-user data?

Cost Optimization

1. Have the WorkSpaces bundles been right sized for the user case and applications?
2. Will the users consume WorkSpaces more than 82 hours per month?

While these questions do not constitute an exhaustive list of items that should be considered, they provide some overarching guidance to assist you with a Well-Architected Amazon WorkSpaces deployment.

Security

This section explains how to secure data by using encryption when using Amazon WorkSpaces services. It describes encryption in transit and at rest, and the use of security groups to protect network access to the WorkSpaces. This section also provides information on how to control end device access to WorkSpaces by using Trusted Devices, and IP Access Control Groups.

Additional information on authentication (including MFA support) in the AWS Directory Service can be found in this 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 (p. 50) section of this document.

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, the 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 the authentication gateway to AWS Directory Service takes place over HTTPS, so no user credentials are transmitted in plain text.

Authentication — Active Directory Connector (ADC)

AD Connector uses Kerberos to establish authenticated communication with on-premises AD, so it can bind to LDAP and run subsequent LDAP queries. Client-side LDAPS support in ADC is also available to encrypt queries between Microsoft AD and AWS Applications. Before implementing client-side LDAPS functionality, review the prerequisites for client-side LDAPS.

The AWS Directory Service also supports LDAP with TLS. No user credentials are transmitted in plaintext at any time. For increased security, it is possible to connect a WorkSpaces VPC with the on-premises network (where AD resides) using a VPN connection. When using an AWS hardware VPN connection,

customers can set up encryption in transit by using standard IPSEC (Internet Key Exchange (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 perfect forward secrecy (PFS).

Broker Stage

After receiving the `OAuth 2.0` token (from the 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 AES-256 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 Amazon 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 a WorkSpaces desktop over the management interface is via UDP 55002. (See documentation for IP Address and Port Requirements for Amazon WorkSpaces. 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. Customers can actively change this to 256-bit, either using PCoIP-specific AD Group Policy settings for Windows WorkSpaces, or with the pcoip-agent.conf file for Amazon Linux WorkSpaces. To learn more about Group Policy administration for Amazon WorkSpaces, see IP Address and Port Requirements for Amazon WorkSpaces.

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 the customer VPC, such as access to AWS Directory Service, the internet, and the customer corporate network. It is possible to attach security groups to this primary network interface. 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

The management network interface cannot be controlled via security groups; however, customers can use a host-based firewall on WorkSpaces to block ports or control access.AWS doesn't recommend

applying restrictions on the management network interface. If a customer decides to add host-based firewall rules to manage this interface, a few ports should be open so the Amazon WorkSpaces service can manage the health and accessibility to the WorkSpace. See Network Interfaces 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 is possible to modify the rules of a WorkSpaces security group. The results take effect immediately after the changes are applied. However, do not delete this security group. If you delete this security group, your WorkSpaces won't function correctly, and you won't be able to recreate this group and add it back.

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, it can be managed by using the different AWS management tools. See Elastic Network Interfaces. Look for the WorkSpace IP address (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).

After the ENI is located, it can be directly managed by security groups. When manually assigning security groups to the primary network interface, consider the port requirements of Amazon WorkSpaces. See Network Interfaces in the Amazon WorkSpaces Administration Guide.

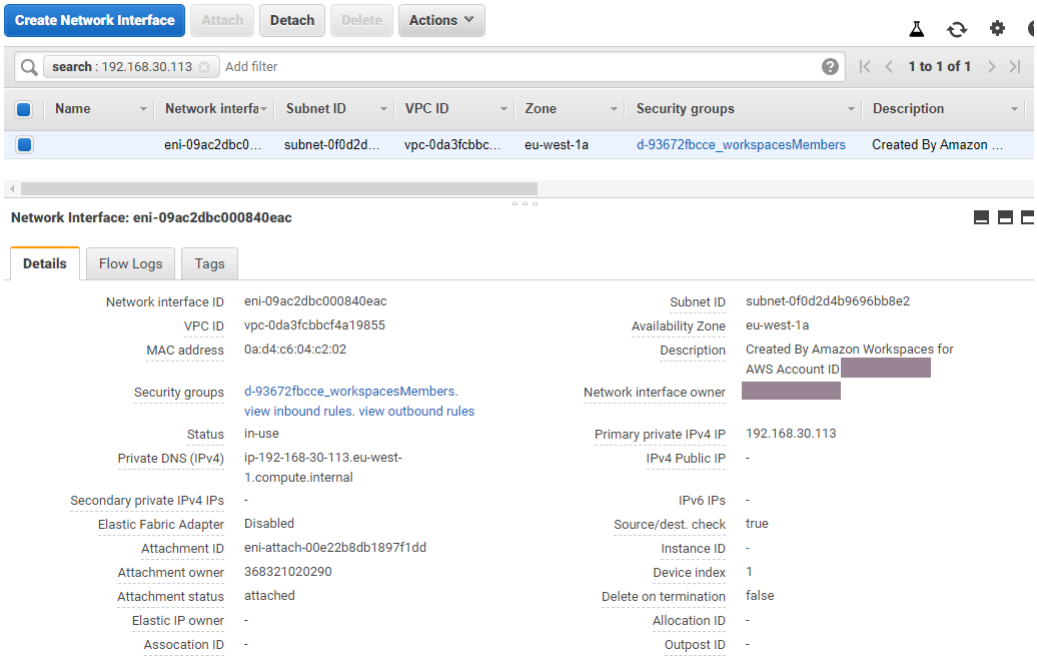

Figure 18 — Managing security group associations

Encrypted WorkSpaces

Each Amazon WorkSpace is provisioned with a root volume (`C:` drive for Windows WorkSpaces, `root` for Amazon Linux WorkSpaces) and a user volume (`D:` drive for Windows WorkSpaces, `/home` for Amazon Linux WorkSpaces). The encrypted WorkSpaces feature enables one or both volumes to be encrypted.

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?

Encryption for a WorkSpace should be specified when launching (creating) the WorkSpace. WorkSpaces volumes can be encrypted only at launch time: after launch, the volume encryption status cannot be changed. Figure 19 shows the Amazon WorkSpaces console page for choosing encryption during the launching of a new WorkSpace.

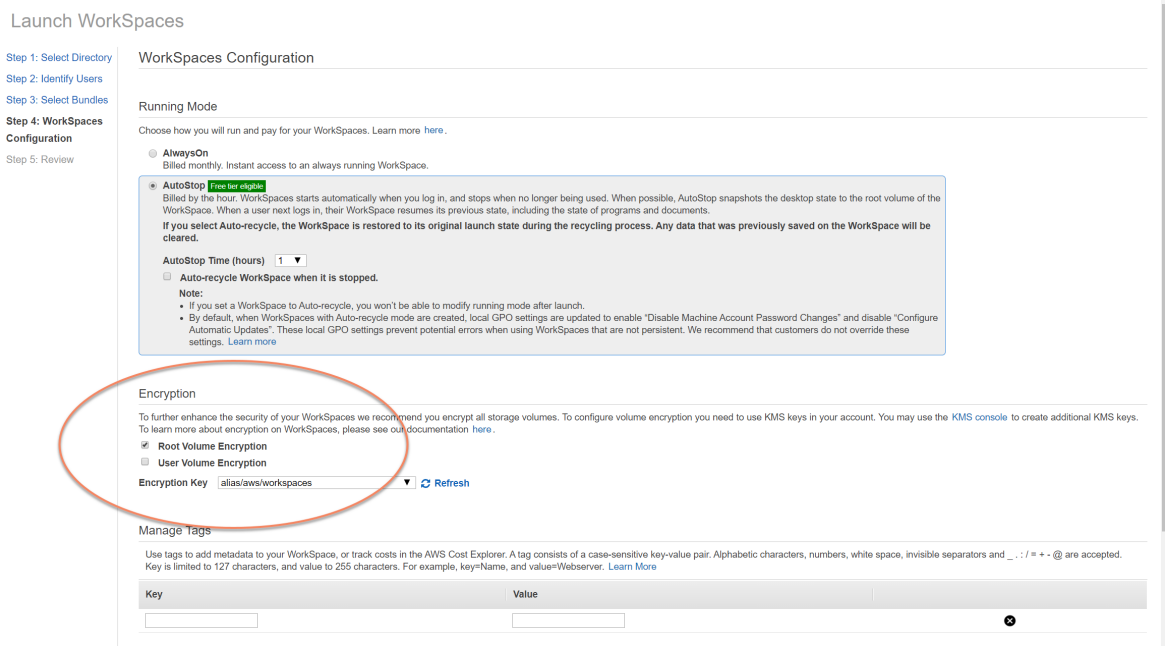

Figure 19 — Encrypting WorkSpace root volumes

How is a New WorkSpace Encrypted?

A customer can choose the Encrypted WorkSpaces option from either the Amazon WorkSpaces console or AWS CLI, or by using the Amazon WorkSpaces API when a customer launches a new WorkSpace.

To encrypt the volumes, Amazon WorkSpaces uses a CMK from AWS Key Management Service (AWS KMS). A default AWS KMS CMK is created the first time a WorkSpace is launched in a Region. (CMKs have a Region scope.)

A customer 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 each of the WorkSpace volumes. (In a strict sense, it is Amazon EBS that will encrypt the volumes). For current CMK limits, see AWS KMS Resource quotas.

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

For a detailed description of the WorkSpaces encryption process, see How Amazon WorkSpaces uses AWS KMS. Consider how the use of CMK will be monitored to ensure that a request for an encrypted WorkSpace is serviced correctly. For additional information about AWS KMS customer master keys and data keys, see the AWS KMS page.

Access Control Options and Trusted Devices

Amazon WorkSpaces provides customers options to manage which client devices can access WorkSpaces. Customers can limit WorkSpaces access to trusted devices only. Access to WorkSpaces can be allowed from macOS and Microsoft Windows PCs using digital certificates. It can also allow or block access for iOS, Android, Chrome OS, Linux, and zero clients, as well as the WorkSpaces Web Access client. With these capabilities, it can further improve the security posture.

Access control options are enabled for new deployments for users to access their WorkSpaces from clients on Windows, MacOS, iOS, Android, ChromeOS, and Zero Clients. Access using Web Access or a Linux WorkSpaces client is not enabled by default for a new Workspaces deployment, and will need to be enabled.

If there are limits on corporate data access from trusted devices (also known as managed devices), WorkSpaces access can be restricted to trusted devices with valid certificates. When this feature is enabled, Amazon WorkSpaces uses certificate-based authentication to determine whether a device is trusted. If the WorkSpaces client application can't verify that a device is trusted, it blocks attempts to log in or reconnect from the device.

For more information about controlling which devices can access WorkSpaces, see Restrict WorkSpaces Access to Trusted Devices.

> **Note**
> **Certificates for trusted devices only apply to the Amazon WorkSpaces Windows and macOS clients.** This feature does not apply to the Amazon WorkSpaces Web Access client, or any third-party clients, including but not limited to Teradici PCoIP software and mobile clients, Teradici PCoIP zero clients, RDP clients, and remote desktop applications.

IP Access Control Groups

Using IP address-based control groups, customers can define and manage groups of trusted IP addresses, and allow users to access their WorkSpaces only when they're connected to a trusted network. This feature helps customers gain greater control over their security posture.

IP access control groups can be added at the WorkSpaces directory level. There are two ways to get started using IP access control groups.

- **IP Access Controls page** — From the WorkSpaces management console, IP access control groups can be created on the IP Access Controls page. Rules can be added to these groups by entering the IP addresses or IP ranges from which WorkSpaces can be accessed. These groups can then be added to directories on the Update Details page.

- **Workspace APIs** — WorkSpaces APIs can be used to create, delete, and view groups; create or delete access rules; or to add and remove groups from directories.

For a detailed description of the using IP access control groups with the Amazon WorkSpaces encryption process, see IP Access Control Groups for Your WorkSpaces.

Monitoring or Logging Using Amazon CloudWatch

Monitoring network, servers, and logs is an integral part of any infrastructure. 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.

These metrics, like all CloudWatch metrics, can be viewed in the AWS Management Console (Figure 20), accessed via the CloudWatch APIs, and monitored by CloudWatch alarms and third-party tools.

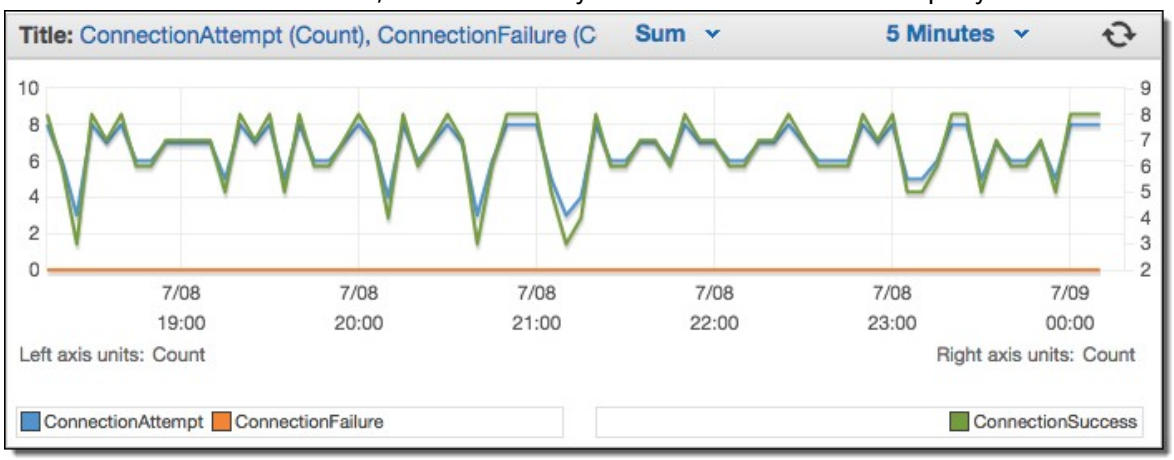

Figure 20 — CloudWatch metrics – ConnectionAttempt / ConnectionFailure

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

- **`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 21.

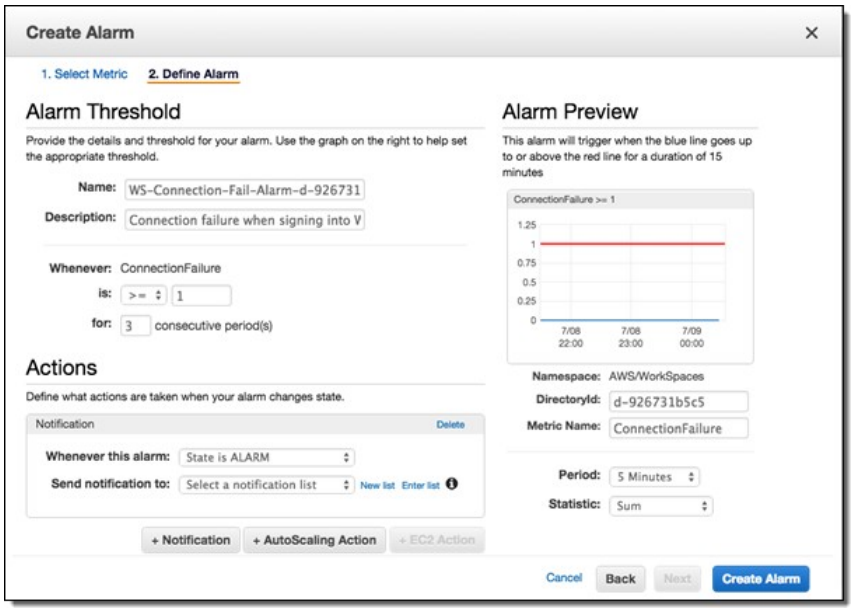

Figure 21 — Create CloudWatch alarm for WorkSpaces connection errors

Amazon CloudWatch Events for WorkSpaces

Events from Amazon CloudWatch Events can be used to view, search, download, archive, analyze, and respond to successful logins to WorkSpaces. The service can monitor client WAN IP addresses, Operating System, WorkSpaces ID, and Directory ID information for users' logins to WorkSpaces. For example, it can use events for the following purposes:

- Store or archive WorkSpaces login events as logs for future reference, analyze the logs to look for patterns, and take action based on those patterns.
- Use the WAN IP address to determine where users are logged in from, and then use policies to allow users access only to files or data from WorkSpaces that meet the access criteria found in the CloudWatch Event type of `WorkSpaces Access`.
- Use policy controls to block access to files and applications from unauthorized IP addresses.

For more information on how to use CloudWatch Events, see the Amazon CloudWatch Events User Guide. To learn more about CloudWatch Events for WorkSpaces, see Monitor your WorkSpaces using Cloudwatch Events.

Cost Optimization

Self-Service WorkSpace Management Capabilities

In Amazon WorkSpaces, self-service WorkSpace management capabilities can be enabled for users to provide them with more control over their experience. Allowing users self-service capability can also reduce your IT support staff workload for Amazon WorkSpaces. When self-service capabilities are enabled, they enable users to perform one or more of the following tasks directly from their Windows, macOS, or Linux client for Amazon WorkSpaces:

- Cache their credentials on their client. This lets them reconnect to their WorkSpace without re-entering their credentials.
- Restart their WorkSpace.
- Increase the size of the root and user volumes on their WorkSpace.
- Change the compute type (bundle) for their WorkSpace.
- Switch the running mode of their WorkSpace.
- Rebuild their WorkSpace.

There are no on-going cost implications for allowing users the **Restart** and **Rebuild** options for their WorkSpaces. Users should be aware that a **Rebuild** of their WorkSpace will cause their WorkSpace to be unavailable, for up to an hour, as the rebuild process takes place.

Options to increase the size of the volumes, change the compute type and switch the running mode can incur additional costs for WorkSpaces. A best practice is to enable self-service to reduce the workload for the support team. Self-service for additional cost items should be allowed within a workflow process that ensures that authorization for additional charges has been obtained. This could be through a dedicated self-service portal for WorkSpaces, or by integration with existing Information Technology Service Manage (ITSM) services, such as ServiceNow.

For more detailed information, see Enable Self-Service WorkSpace Management Capabilities for Your Users. For an example describing enabling a structured portal for user self-service, see Automate Amazon WorkSpaces with a Self-Service Portal.

Amazon WorkSpaces Cost Optimizer

The *running mode* of a WorkSpace determines its immediate availability and how it will be billed. Here are the current running WorkSpaces running mode:

- **`AlwaysOn`** — Use when paying a fixed monthly fee for unlimited usage of WorkSpaces. This mode is best for users who use their WorkSpace full time as their primary desktop.
- **`AutoStop`** — Use when paying for WorkSpaces by the hour. With this mode, WorkSpaces stop after a specified period of inactivity and the state of apps and data is saved. To set the automatic stop time, use **`AutoStop Time (hours)`**.

A best practice is to monitor usage and set the WorkSpaces' running mode to be the most cost effective. This can be done with the Amazon WorkSpaces Cost Optimizer. This solution deploys an Amazon CloudWatch event that invokes an AWS Lambda function every 24 hours.

This solution can convert individual WorkSpaces from an hourly billing model to a monthly billing model on any day after the threshold is met. If the solution converts a WorkSpace from hourly billing to monthly billing, the solution does not convert the WorkSpace back to hourly billing until the beginning of the next month, and only if usage was below the threshold. However, the billing model can be manually change at any time using the AWS Management Console. The solution's AWS CloudFormation template includes parameters that will run these conversions.

Opting Out with Tags

To prevent the solution from converting a WorkSpace between billing models, apply a resource tag to the WorkSpace using the tag key **`Skip_Convert`** and any tag value. This solution will log tagged WorkSpaces, but it will not convert the tagged WorkSpaces. Remove the tag at any time to resume automatic conversion for that WorkSpace.

For more details, see Amazon WorkSpaces Cost Optimizer.

Troubleshooting

Common administration and client issues, such as error messages like "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 the on-premises directory, the firewall for the on-premises network must have certain ports open to the CIDRs for both subnets in the VPC. See Scenario 1: Using AD Connector to Proxy Authentication to On-Premises Active Directory service (p.) in this document. To test if these conditions are met, perform the following steps:

To test 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 Microsoft Visual Studio project files are included to modify the test application, if desired.
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 the domain name is excluded, the functional levels won't be tested.
- *<server_IP_address>* — The IP address of a domain controller in the on-premises domain. The ports are tested against this IP address. If the IP address is excluded, the ports won't be tested.

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

Troubleshooting a WorkSpace Custom Image Creation Error

If a Windows or Amazon Linux WorkSpace has been launched and customized, a custom image can be created from that WorkSpace. A custom image contains the operating system, application software, and settings for the WorkSpace.

Review the requirements to create a Windows custom image or the requirements to create an Amazon Linux custom image. Image creation requires that all prerequisites are met before image creation can start.

To confirm that the Windows WorkSpace meets the requirements for image creation, AWS recommends running the Image Checker. The Image Checker performs a series of tests on the WorkSpace when an

image is created, and provides guidance on how to resolve any issues it finds. For detailed information read installing and configuring the image checker.

After the WorkSpace passes all tests, a **Validation Successful** message appears. You can now create a custom bundle. Otherwise, resolve any issues that cause test failures and warnings, and repeat the process of running the Image Checker until the WorkSpace passes all tests. All failures and warnings must be resolved before an image can be created.

For more information, follow the tips for resolving issues detected by the Image Checker.

Troubleshooting a Windows WorkSpace Marked as Unhealthy

The Amazon WorkSpaces service periodically checks the health of a WorkSpace by sending it a status request. The WorkSpace is marked as `Unhealthy` if a response isn't received from the WorkSpace in a timely manner. Common causes for this problem are:

- An application on the WorkSpace is blocking network connection between the Amazon WorkSpaces service and the WorkSpace.
- High CPU utilization on the WorkSpace.
- The computer name of the WorkSpace is changed.
- The agent or service that responds to the Amazon WorkSpaces service isn't in running state.

The following troubleshooting steps can return the WorkSpace to a healthy state:

- First, reboot the WorkSpace from the Amazon WorkSpaces console (signin required). If rebooting the WorkSpace doesn't resolve the issue, either use RDP, or connect to an Amazon Linux WorkSpace using SSH.

- If the WorkSpace is unreachable by a different protocol, rebuild the WorkSpace from the Amazon WorkSpaces console.
- If a WorkSpaces connection cannot be established, verify the following:

Verify CPU Utilization

- Open Task Manager to determine if the WorkSpace is experiencing high CPU utilization. If it is, try any of the following troubleshooting steps to resolve the issue:

1. Stop any service that is consuming a high amount of CPU.
2. Resize the WorkSpace to a compute type greater than what is currently used .
3. Reboot the WorkSpace .

> **Note**
> To diagnose high CPU utilization, and for guidance if the above steps don't resolve the high CPU utilization issue, see How do I diagnose high CPU utilization on my EC2 Windows instance when my CPU is not throttled? .

Verify the Computer Name of the WorkSpace

- If the computer name of the WorkSpacewas changed, change it back to the original name:

1. Open the Amazon WorkSpaces console (signin required), and then expand the `Unhealthy` WorkSpace to show details.
2. Copy the **Computer Name**.
3. Connect to the WorkSpace using RDP.

   Open a command prompt, and then enter hostname to view the current computer name.
   - If the name matches the Computer Name from step 2, skip to the next troubleshooting section.
   - If the names don't match, enter `sysdm.cpl` to open system properties, and then follow the remaining steps in this section.
4. Choose **Change**, and then paste the Computer Name from step 2.
5. Enter the domain user credentials if prompted.

Confirm that `SkyLightWorkspaceConfigService` is in Running State

From **Services**, verify if the WorkSpace service `SkyLightWorkspaceConfigService` is in running state. If it's not, start the service.

Verify Firewall Rules

- Confirm that the Windows Firewall and any third-party firewall that is running have rules to allow the following ports:
  - Inbound TCP on port 4172: Establish the streaming connection.
  - Inbound UDP on port 4172: Stream user input.
  - Inbound TCP on port 8200: Manage and configure the WorkSpace.
  - Outbound UDP on port 55002: PCoIP streaming.

If the firewall uses stateless filtering, then open ephemeral ports 49152-65535 to allow return communication.

If the firewall uses stateful filtering, then ephemeral port 55002 is already open.

Collecting a WorkSpaces Support Log Bundle for Debugging

When troubleshooting WorkSpaces issues, it will be necessary to gather the log bundle from the affected WorkSpace and the host where the WorkSpaces client is installed. There are two fundamental categories of logs:

- **Server-side logs** — The WorkSpace is the server in this scenario, so these are logs that live on the WorkSpace itself.
- **Client-side logs** — These will be on the device that the end user is using to connect to the WorkSpace.
  - Note that only Windows and macOS clients write logs locally.
  - Zero clients and iOS clients do not log.
  - Android logs are encrypted on the local storage and uploaded automatically to the WorkSpaces client engineering team. Only that team can review the logs for Android devices.

PcoIP Server-Side Log

All of the PCoIP components write their log files into one of two folders:

- Primary location: `C:\ProgramData\Teradici\PCoIPAgent\logs`
- Archive location: `C:\ProgramData\Teradici\logs`

Sometimes when working with AWS Support on a complex issue, it will be necessary to put the PCoIP Server agent into verbose logging mode. To enable this:

1. Open the following registry key: `HKEY_LOCAL_MACHINE\SOFTWARE\Policies\Teradici\PCoIP \pcoip_admin_defaults`.
2. In the `pcoip_admin_defaults` key create the following 32 bit DWORD: `pcoip.event_filter_mode` .
3. Set the value for `pcoip.event_filter_mode` to "`3`" (Dec or Hex).

For reference, these are the log thresholds which can be defined in this DWORD.

- 0 — (CRITICAL)
- 1 — (ERROR)
- 2 — (INFO)
- 3 — (Debug)

If the `pcoip_admin_default` DWORD doesn't exist, the log level is 2 by default. It is recommended to restore a value of 2 to the DWORD after it no longer need verbose logs, as they are much larger and will consume disk space unnecessarily.

WebAccess Server-Side Logs

The WorkSpaces web access client uses the STXHD service. The logs for WebAccess is stored at `C:\ProgramData\Amazon\Stxhd\Logs`.

Client-Side Logs

These logs come from the WorkSpaces client that the user connects with, so the logs are on the end user's computer. The log file locations for Windows and Mac are:

- **Windows** — `"%LOCALAPPDATA%\Amazon Web Services\Amazon WorkSpaces\Logs"`
- **macOS** — `~/Library/Logs/Amazon Web Services/`
- **Linux** — `~/.local/share/Amazon Web Services/Amazon WorkSpaces/logs`

To help troubleshoot issues that the users might experience, enable advanced logging can be used on any Amazon WorkSpaces client. Advanced logging is enabled for every subsequent client session until it is disabled.

1. Before connecting to the WorkSpace, the end user should enable advanced logging for their WorkSpaces client.
2. The end user should then connect as normal and use their WorkSpace, and attempt to reproduce the issue.
3. Advanced logging generates log files that contain diagnostic information and debugging-level details, including verbose performance data.

This setting persists until explicitly turned off. Once the user has been able to reproduce the issue with verbose logging on, this setting should be disabled, as it generates large log files.

Automated Server Side Log Bundle Collection for Windows

The `Get-WorkSpaceLogs.ps1` script is very helpful for quickly gathering a server-side log bundle for AWS Premium Support. The script can be requested from AWS Premium Support by requesting it in a support case.

1. Connect to the WorkSpace using the client or using Remote Desktop Protocol (RDP)
2. Start an administrative command prompt (run as administrator).
3. Launch the script from the command prompt with the following command:

```
powershell.exe -NoLogo -ExecutionPolicy RemoteSigned -NoProfile -File "C:\Program Files
\Amazon\WorkSpacesConfig\Scripts\Get-WorkSpaceLogs.ps1"
```

4. The script will create a log bundle on the user's desktop.

The script creates a zip file with the following folders:

- **`C`** — Contains the files from Program Files, Program Files (x86), ProgramData, and Windows related to Skylight, EC2Config, Teradici, Event viewer, and Windows logs (Panther and others).
- **`CliXML`** — Contains XML files that can be imported in PowerShell by using `Import-CliXML` for interactive filtering. See Import-Clixml.
- **`Config`** — Detailed logs for each check that is performed.
- **`ScriptLogs`** — Logs about the script execution (not relevant to the investigation, but useful to debug what the script does).
- **`tmp`** — Temporary folder (it should be empty).
- **`Traces`** — Packet capture done during the log collection.

How to Check Latency to Closest AWS Region

The Connection Health Check website quickly checks whether all of the required services that use Amazon WorkSpaces can be reached. It also does a performance check to each AWS Region where Amazon WorkSpaces is available, and lets users know which one will be the fastest.

Conclusion

There is 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 Windows or Linux 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 access from anywhere, using the device of their choice.

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

Contributors

Contributors to this document include:

- Naviero Magee, Sr. EUC Solutions Architect, Amazon Web Services
- Andrew Wood, Sr. EUC Solutions Architect, Amazon Web Services
- Dzung Nguyen, Sr. Consultant, Amazon Web Services
- Stephen Stetler, Sr. EUC Solutions Architect, Amazon Web Services
- Asriel Agronin, Sr. EUC Solutions Architect, Amazon Web Services
- Andrew Morgan, EUC Solutions Architect, Amazon Web Services
- Chris Ott, Sr. EUC Solutions Architect, Amazon Web Services

Further Reading

For additional information, see:

- Amazon WorkSpaces Administration Guide
- Amazon WorkSpaces Developer Guide
- Amazon WorkSpaces Clients
- Managing Amazon Linux 2 Amazon WorkSpaces with AWS OpsWorks for Puppet Enterprise
- Customizing the Amazon Linux WorkSpace
- How to improve LDAP Security in AWS Directory Service with client-side LDAPs
- Use Amazon CloudWatch Events with Amazon WorkSpaces and AWS Lambda for greater fleet visibility
- How Amazon WorkSpaces Use AWS KMS
- AWS CLI Command Reference – WorkSpaces
- Monitoring Amazon WorkSpaces Metrics
- MATE Desktop Environment
- Troubleshooting AWS Directory Service Administration Issues
- Troubleshooting Amazon WorkSpaces Administration Issues
- Troubleshooting Amazon WorkSpaces Client Issues
- Automate Amazon WorkSpaces with a Self-Service Portal

Document Revisions

To be notified about updates to this whitepaper, subscribe to the RSS feed.

update-history-change	update-history-description	update-history-date
Updated content (p. 64)	Updated content for WorkSpaces Streaming Protocol, smart card authentication, diagrams, client deployments, end device selection, and web access	April 28, 2021
Updated content (p. 64)	Updated content	December 1, 2020
Whitepaper updated (p. 64)	Updated content since first publication and added new diagrams.	May 1, 2020
Initial publication (p. 64)	First published.	July 1, 2016

Notices

Customers are responsible for making their own independent assessment of the information in this document. This document: (a) is for informational purposes only, (b) represents current AWS product offerings and practices, which are subject to change without notice, and (c) does not create any commitments or assurances from AWS and its affiliates, suppliers or licensors. AWS products or services are provided “as is” without warranties, representations, or conditions of any kind, whether express or implied. 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.