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.

© 2019 Amazon Web Services, Inc. or its affiliates. All rights reserved.
About this Guide

Amazon WorkSpaces is a managed, secure cloud desktop service. You can use Amazon WorkSpaces to provision either Windows or Linux desktops in just a few minutes and quickly scale to provide thousands of desktops to workers across the globe.

This guide helps cloud architects quickly deploy a secure and scalable *Proof-of-Concept* (POC) Amazon WorkSpaces environment that is aligned with the [AWS Well-Architected Framework](https://aws.amazon.com/en_us/about-aws/what-is-aws-well-architected-framework/).
Overview

When deploying an Amazon WorkSpaces environment, adhering to the principles outlined in the AWS Well-Architected Framework will help ensure a secure, scalable, high-performing, and cost-optimized solution.

This guide reviews the Framework and leads the architect through the process of deploying a well-architected Amazon WorkSpaces Proof-of-Concept (POC).

Before You Begin

The architect should have a basic understanding of core AWS technologies, including Amazon Virtual Private Cloud (VPC), Amazon Elastic Compute Cloud (Amazon EC2), Security Groups, Network Access Control Lists, subnetting, and routing.

Secure Proper Access and Privileges

The architect should ensure that they have access to the following:

- An AWS account and console access with sufficient privileges to manipulate the required resources involved in the POC. For instructions on setting permissions for WorkSpaces resources and operations, see the Control Access to Amazon WorkSpaces Resources chapter of the Amazon WorkSpaces Administrator Guide. AWS recommends applying the principles of Least Privilege when granting access to resources to reduce potential attack surface. This principle entails granting users no more than the minimum amount of privilege required to perform their role.

- Access and privileges to manipulate customer-premise resources that may be involved in the POC such as Microsoft Active Directory (for authentication and authorization), firewalls, routers, VPN devices, multi-factor authentication (MFA) devices, etc. Firewalls block unauthorized access to network resources while permitting network traffic to and from authorized sources.
Increase Service Limits

New AWS accounts are limited to creating only one non-graphics WorkSpace per Region and are not allowed to create any graphics WorkSpaces. Use the Limit Increase Request Form to request that your Amazon WorkSpaces service limit be increased to support your POC. You can also navigate to the form from the AWS console by selecting Support, then Support Center. Choose Create case, then Service limit increase. Under Limit type, choose WorkSpaces.

Identify the Supported Availability Zones

As of this writing, Amazon WorkSpaces is supported in 11 commercial regions around the world. (See the AWS Region Table for the current list of supported regions.) However, the service may not be supported in every Availability Zone (AZ) within those regions. When creating the subnets for your Amazon WorkSpaces POC, you must ensure that they are created in AZs that support the Amazon WorkSpaces service.

AZ mappings vary between AWS accounts. For example, us-east-1a in one AWS account may point to different physical AZs than us-east-1a in another AWS account. Work with your AWS Solutions Architect to obtain the Amazon WorkSpaces-supported AZs for the region in which you are intending to deploy, based on your AWS account number. You will be provided information similar to the following for your account:

Table 1 – Amazon WorkSpaces Supported Availability Zones (example)

<table>
<thead>
<tr>
<th>Account #</th>
<th>Region</th>
<th>Supported Availability Zones</th>
</tr>
</thead>
<tbody>
<tr>
<td>111122223333</td>
<td>us-east-1</td>
<td>us-east-1b, us-east-1c, us-east-1e</td>
</tr>
</tbody>
</table>

To learn more about Regions, Availability Zones, and other concepts related to the AWS Global Infrastructure, see this interactive tutorial.
AWS Well-Architected Framework Review

The AWS Well-Architected Framework has been developed to help cloud architects build secure, high-performing, resilient, and efficient infrastructure for their applications. Based on five pillars—operational excellence, security, reliability, performance efficiency, and cost optimization—the Framework provides a consistent approach for customers and partners to evaluate architectures, and implement designs that will scale over time.

Table 2 – The Pillars of the AWS Well-Architected Framework

<table>
<thead>
<tr>
<th>Pillar Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operational Excellence</td>
<td>The ability to run and monitor systems to deliver business value and to continually improve supporting processes and procedures.</td>
</tr>
<tr>
<td>Security</td>
<td>The ability to protect information, systems, and assets while delivering business value through risk assessments and mitigation strategies.</td>
</tr>
<tr>
<td>Reliability</td>
<td>The ability of a system to recover from infrastructure or service disruptions, dynamically acquire computing resources to meet demand, and mitigate disruptions such as misconfigurations or transient network issues.</td>
</tr>
<tr>
<td>Performance Efficiency</td>
<td>The ability to use computing resources efficiently to meet system requirements, and to maintain that efficiency as demand changes and technologies evolve.</td>
</tr>
<tr>
<td>Cost Optimization</td>
<td>The ability to run systems to deliver business value at the lowest price point.</td>
</tr>
</tbody>
</table>

For more information, see the [AWS Well-Architected Framework whitepaper](#).

A well-architected Amazon WorkSpaces deployment should have design elements related to all pillars of the framework. This POC provides a start but does not represent a complete implementation of the AWS Well-Architected Framework.
Architecture Overview

The following diagram shows the network flow for an Amazon WorkSpaces user connecting to the service via the public internet from outside the corporate firewall.

Figure 1 – Amazon WorkSpaces Network Flow: Connecting from Public internet
Building on the general architecture above, this guide will walk through setting up an environment similar to the following:

*Figure 2 – POC Architecture*
VPC Best Practices

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

You can easily customize the network configuration for your Amazon VPC. For example, you can create a public subnet for your web servers that has access to the internet, and place your backend systems, such as WorkSpaces, databases, or application servers, in a private-facing subnet without internet access. You can leverage multiple layers of security, including security groups and network access control lists, to help control access to Amazon EC2 instances and WorkSpaces in each subnet.

The VPC for this POC walkthrough has the following design elements:

<table>
<thead>
<tr>
<th>Design Element</th>
<th>Purpose</th>
<th>Framework Pillars</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>NAT Gateway</strong></td>
<td>Scalable service to enable outbound internet access by WorkSpaces in private subnets while preventing the internet from initiating a connection with them</td>
<td>Security, Performance, Reliability</td>
</tr>
<tr>
<td><strong>Internet Gateway</strong></td>
<td>Horizontally scaled, redundant, and highly available VPC component that allows communication between instances in your VPC and the internet. Public subnets have routes to the internet through this component.</td>
<td>Performance, Reliability</td>
</tr>
<tr>
<td><strong>2 Public Subnets</strong></td>
<td>Contains the NAT Gateways and other Internet-facing components</td>
<td>Performance</td>
</tr>
<tr>
<td><strong>2 Private Subnets</strong></td>
<td>Contains the elastic network interfaces (ENI) for the Directory Services instances (e.g., AD Connector), cloud-based remote domain controllers (optional) and WorkSpaces’ customer managed ENIs</td>
<td>Security, Reliability, Performance, Operational Excellence</td>
</tr>
<tr>
<td>Design Element</td>
<td>Purpose</td>
<td>Framework Pillars</td>
</tr>
<tr>
<td>----------------------</td>
<td>--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>---------------------------------------</td>
</tr>
<tr>
<td>Virtual Private</td>
<td>Facilitates secure Virtual Private Network (VPN) connections between your corporate network and AWS. Redundant connections provide improved availability.</td>
<td>Security, Reliability, Cost Optimization</td>
</tr>
<tr>
<td>Gateway &amp; Customer</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Gateways</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
VPC Setup Walkthrough

Configure VPC\(^5\)

**Step 1: Allocate an Elastic IP Address**

Allocate an Elastic IP address for your NAT gateway as follows. Note that if you are using an alternative method of providing internet access, you can skip this step.

**To allocate an Elastic IP address**

1. Open the Amazon VPC console at [https://console.aws.amazon.com/vpc/](https://console.aws.amazon.com/vpc/).
2. In the navigation pane, choose *Elastic IPs*.
3. Choose *Allocate new address*.
4. On the *Allocate new address* page, choose *Allocate* and make note of the Elastic IP address, then choose *Close*.

**Step 2: Create a VPC**

Create a VPC with two public subnets and two private subnets as follows.

To provide the layout in Figure 2, we want Public Subnet_1 and Private Subnet_1 to share the same Availability Zone and we want Public subnet_2 and Private Subnet_2 to share a different Availability Zone. Both Availability Zones selected must support the Amazon WorkSpaces service as discussed in the *Identify the Supported Availability Zones* section.

**To Set Up a VPC**

1. Open the Amazon VPC console at [https://console.aws.amazon.com/vpc/](https://console.aws.amazon.com/vpc/).
2. In the navigation pane, choose *VPC Dashboard*.
3. Choose *Launch VPC Wizard*.
4. Choose *VPC with Public and Private Subnets* and then choose *Select*.
5. Configure the VPC as follows:
   a. For *IPv4 CIDR block*, type the CIDR block for the VPC.
   b. For *VPC name*, type a name for the VPC.
6. Configure the public subnet as follows:
a. For **IPv4 CIDR block**, type the CIDR block for the subnet.

b. For **Availability Zone**, select an AZ supported by the Amazon WorkSpaces service.

c. For **Public subnet name**, type a name for the subnet (for example, `WorkSpaces Public Subnet_1`).

6. Configure the first private subnet as follows:
   a. For **Private subnet's IPv4 CIDR**, type the CIDR block for the subnet.
   b. For **Availability Zone**, select the same AZ chosen in step 5b above.
   c. For **Private subnet name**, type a name for the subnet (for example, `WorkSpaces Private Subnet_1`).

7. For **Elastic IP Allocation ID**, choose the Elastic IP address that you created. If you are using an alternative method of providing internet access, you can skip this step.

8. Choose **Create VPC**. (This action can take several minutes to complete.) After the VPC is created, choose **OK**.

**Step 3: Add a Second Public Subnet**

In the previous step, you created a VPC with one public subnet and one private subnet. Use the following procedure to add a second public subnet.

**To Add a Subnet**

1. In the navigation pane, choose **Subnets**

2. Choose **Create Subnet**.

3. For **Name tag**, type a name for the second public subnet (for example, `WorkSpaces Public Subnet_2`).

4. For **VPC**, select the VPC that you created.

5. For **Availability Zone**, select an AZ from the list of WorkSpaces supported AZ’s provided by your AWS Solutions Architect, but different from that of the first public subnet.

6. For **IPv4 CIDR block**, type the CIDR block for the subnet.

7. Choose **Yes, Create**.

8. Click **Close** to return to the VPC console view.
Now we must modify the route table of the subnet to make it a public subnet.

9. Choose **Route Tables** on the navigation pane. Two route tables should be visible for this VPC. (You can Filter by VPC in the upper left-hand corner of the VPC console to narrow the list of resources shown to one VPC).

10. Note which route table for the VPC is not the main route table. Verify on the **Routes** tab that there is a route for 0.0.0.0/0 with a destination to the Internet Gateway (igw-xxxxxx). If confirmed, modify the Name field as “Route to IGW”.

11. Verify that the other route table (Main = Yes) has a route for 0.0.0.0/0 to the NAT gateway previously created. If confirmed, set the name to “Route to NAT1”.

12. Select **Subnets** on the left menu.

13. Select the recently created second Public Subnet.

14. Select the **Route Table** tab and choose **Edit Route Table Associations**.

15. In the drop-down, select the route table that points to the Internet Gateway and click **Save**. Click **Close** to return to the VPC console view. The subnet is now a Public subnet as it has a route to internet via the IGW.

Next, we need to create a NAT gateway in the second public subnet.

**Step 4: Add a NAT Gateway to the Second Public Subnet**

1. Select **NAT Gateways** from the left menu

2. Choose **Create NAT Gateway**.

3. For subnet, choose the second public subnet created previously.

4. Select Create New EIP. Then choose the newly created EIP from the dropdown and click **Create NAT Gateway**.

5. Note the ID of the second NAT gateway and select **Close** to return to the VPC console view.

Create a new Route Table pointing internet traffic to this second NAT Gateway.

6. Select **NAT Gateways** from the navigation pane.

7. Two NAT Gateways should be visible if filtering by the VPC we are working with in this exercise. Taking note of either the NAT Gateway ID, the private IP address or the Creation time, Edit the names of the two NAT Gateways as NAT1 and NAT2.
8. Select **Route Tables** from the navigation pane.

9. Select **Create Route Table**.

10. Name the route table **Route to NAT2** and choose **Create**. Select Close to return to the VPC console view.

11. Select the new route table. Select the **Routes** tab and select **Edit Routes**.

12. For **Destination** enter 0.0.0.0/0 and **Target** the recently created second NAT Gateway, NAT2. Click **Save**.

**Step 5: Add a Second Private Subnet**

In the previous steps, you created a second public subnet and a NAT Gateway. Use the following procedure to add a second private subnet and route internet traffic through the second NAT Gateway.

**To Add a Subnet**

1. In the navigation pane, choose **Subnets**.

2. Choose **Create Subnet**.

3. For **Name tag**, type a name for the private subnet (for example, **WorkSpaces Private Subnet_2**).

4. For **VPC**, select the VPC that you created.

5. For **Availability Zone**, select an AZ from the list of WorkSpaces supported AZs provided by your AWS Solutions Architect, but different from the first private subnet.

6. For **IPv4 CIDR block**, type the CIDR block for the subnet.

7. Choose **Yes, Create**. Click **Close** to return to the VPC Console view.

Now we must associate this private subnet with the route table directing internet traffic through the second NAT instance (NAT2).

8. Select the second private subnet. Select **Route Table** tab and **Edit route table association**.

9. In the drop-down, choose the route table pointing to the second NAT Gateway (NAT2) and click **Save**.
Step 6: Verify the Route Tables

You can verify the route tables that you created.

To verify the route tables

1. In the navigation pane, choose **Subnets**, and select the first public subnet that you created.
2. On the **Route Table** tab, choose the ID of the route table (for example, Route to IGW).
3. On the **Routes** tab, verify that there is one route for local traffic and another route that sends all other traffic to the internet gateway for the VPC.
4. In the navigation pane, choose **Subnets**, and select the first private subnet that you created (for example, WorkSpaces Private Subnet 1).
5. On the **Route Table** tab, choose the ID of the route table.
6. On the **Routes** tab, verify that there is one route for local traffic and another route that sends all other traffic to the first NAT gateway (NAT1).
7. In the navigation pane, choose **Subnets**, and select the second private subnet that you created (for example, WorkSpaces Private Subnet 2).
8. On the **Routes** tab, verify that the route table is the route table directing internet traffic through the second NAT Gateway (for example, Route to NAT2). If the route table is different, choose **Edit** and select this route table.
Port Requirements

To connect to your WorkSpaces, the network that your Amazon WorkSpaces clients are connected to must have certain ports open to the IP address ranges for the various AWS services (grouped in subsets). These address ranges vary by AWS Region. These same ports must also be open on any firewall running on the client. These requirements are important to the Security pillar of the AWS Well-Architected Framework.

Whitelisting TCP port 443 (Authentication)

This port is used for client application updates, registration, and authentication. The desktop client applications support the use of a proxy server for port 443 (HTTPS) traffic. To enable the use of a proxy server, open the client application, choose Advanced Settings, select Use Proxy Server, specify the address and port of the proxy server, and choose Save.

This port must be open to the following IP address ranges:

- The AMAZON subset in the GLOBAL Region.
- The AMAZON subset in the Region that the WorkSpace is in.
- The AMAZON subset in the us-east-1 Region.
- The AMAZON subset in the us-west-2 Region.
- The S3 subset in the us-west-2 Region.

For details on obtaining the list of IP addresses indicated above, see this page for the location of the json file and PowerShell filtering options.

The following code snippet can be used with the AWS Tools for PowerShell to obtain the TCP port 443 outbound whitelist for WorkSpaces deployed in the us-east-1 Region:

```powershell
Get-AWSPublicIpAddressRange -servicekey AMAZON -Region GLOBAL | out-file .\outfile.txt
Get-AWSPublicIpAddressRange -servicekey AMAZON -Region us-east-1 | out-file .\outfile.txt -Append
Get-AWSPublicIpAddressRange -servicekey AMAZON -Region us-west-2 | out-file .\outfile.txt -Append
Get-AWSPublicIpAddressRange -servicekey S3 -Region us-west-2 | out-file .\outfile.txt -Append
```
The output is captured in the file `outfile.txt`

See [Port Requirements for Amazon WorkSpaces](https://aws.amazon.com/documentation/workspaces/port-requirements/) in the *Amazon WorkSpaces Administration Guide* for the complete list of port requirements.
Decision Point: Select a Directory Service to Use for the POC

There are three different AWS Directory Services that you can use with Amazon WorkSpaces:

1. **Simple AD**: A standalone directory in the cloud, where you create and manage user identities and manage access to WorkSpaces and other applications.

2. **AD Connector**: A proxy service that provides an easy way to connect WorkSpaces and other compatible AWS applications such as Amazon WorkDocs and Amazon EC2 for Windows Server instances, to your existing on-premises Microsoft Active Directory. With AD Connector, you can simply add one service account to your Active Directory.

3. **AWS Directory Service for Microsoft Active Directory**: Also known as *AWS Managed Microsoft AD*, AWS Directory Service for Microsoft Active Directory is powered by an actual Microsoft Windows Server Active Directory (AD), managed by AWS in the AWS Cloud. For integration with your corporate Active Directory and WorkSpaces, a two-way trust is required.

For more details about these alternatives, see the [AWS Directory Service Administration Guide](#). 6

To use Simple AD for standalone testing of Amazon WorkSpaces functions without integrating it with your corporate Active Directory or corporate network, skip the next two sections and continue with Launch a WorkSpace with Simple AD.
Establish VPN Connectivity Between Your Corporate Network and AWS

To establish a secure, low-cost VPN connection between your corporate network and AWS, follow the Getting Started instructions in the AWS Site-to-Site VPN User Guide or watch the following AWS Knowledge Center video tutorial.

**Note:** A Site-to-Site VPN connection has two tunnels to help ensure connectivity in case one of the Site-to-Site VPN connections becomes unavailable. To improve availability and protect against a loss of connectivity, you can set up a second Site-to-Site VPN connection to your VPC and virtual private gateway by using a second customer gateway. Click here for more details.

Deploy Cloud-Based Remote Domain Controllers

As a best practice, you can deploy cloud-based remote domain controllers (RDC) to speed-up authentication and session startup in large directories with complex organizational unit and group policy structures. This also reduces the dependency on the VPN connection, increasing overall reliability. As noted, this practice can positively affect the adherence to the Performance and Reliability pillars of the AWS Well-Architected Framework.

**Note:** Amazon WorkSpaces does not support read-only Domain Controllers because of the requirement to create Active Directory computer objects at launch.

Deploying two or more RDCs in different private subnets enhances Security and Reliability. For more information on working with Windows EC2 instances, see Getting Started with Amazon EC2 Instances in the Amazon Elastic Compute Cloud User Guide for Windows Instances.

You are now ready to deploy your choice of directory service and continue the POC implementation. Continue with the next section, Simple AD, or skip to the section for AD Connector or Trusted Domain configurations.
Launch a WorkSpace with Simple AD

Step 1: Create a Simple AD Directory

Create a Simple AD directory. AWS Directory Service creates two directory servers, one in each of the private subnets of your VPC. Initially, there are no users defined in the directory. You will add a user in the next step when you create the WorkSpace.

To Create a Simple AD Directory

1. Open the Amazon WorkSpaces console at https://console.aws.amazon.com/workspaces/.
2. In the navigation pane, choose Directories.
3. Choose Set up Directory, Select Simple AD and choose Next.
4. Configure the directory as follows:
   a. Keep Directory size as Small.
   b. For Organization name, type a unique organization name for your directory (for example, my-example-directory). This name must be at least four characters in length, consist of only alphanumeric characters and hyphens (-), and begin and end with a character other than a hyphen.
   c. For Directory DNS, type the fully qualified name for the directory (for example, example.com).
   d. (Optional) For NetBIOS name, type a short name for the directory (for example, example).
   e. For Administrator password and Confirm password, type a password for the directory administrator account. For more information about the password requirements, see How to Create a Microsoft AD Directory in the AWS Directory Service Administration Guide.
   f. (Optional) For Description, type a description for the directory.
   g. Choose Next.
   h. For VPC, select the VPC that you created.
   i. For Subnets, select the two private subnets you created previously.
   j. Choose Next Step.
5. Review your input and choose Create directory.

6. Choose Done. The initial status of the directory is Requested, and then Creating. When directory creation is complete, the status is Active.

Directory Creation

Amazon WorkSpaces completes the following tasks on your behalf:

- Creates an IAM role to allow the Amazon WorkSpaces service to create elastic network interfaces and list your Amazon WorkSpaces directories. This role has the name workspaces_DefaultRole.
- Sets up a Simple AD directory in the VPC that is used to store user and WorkSpace information. The directory has an administrator account with the user name Administrator and the specified password.
- Creates two security groups, one for directory controllers and another for WorkSpaces in the directory.

Step 2: Create a WorkSpace

Now you are ready to launch the WorkSpace.

To Create a WorkSpace for a User

1. Open the Amazon WorkSpaces console at https://console.aws.amazon.com/workspaces/.

2. In the navigation pane, choose WorkSpaces.

3. Choose Launch WorkSpaces.

4. On the Select a Directory page, do the following:
   a. For Directory, choose the directory that you created.
   b. (Important) Under Select Subnets, choose two private subnets created in different AZs that are supported by WorkSpaces. The WorkSpaces are created in the selected subnets.
   c. For Enable Self-Service Permissions, choose Yes. The Administrator can further refine these permissions by choosing to Update Details of the directory after launching the first WorkSpace. See the Self-Service WorkSpace Management section of this guide for more details.
d. For Enable Amazon WorkDocs, choose Yes.

Note
This option is available only if Amazon WorkDocs is available in the selected Region.

e. Choose Next Step. Amazon WorkSpaces registers your Simple AD directory.

5. On the Identify Users page, add a new user to your directory as follows:
   a. Complete Username, First Name, Last Name, and Email. Use an email address that you have access to.
   b. Choose Create Users.
   c. Choose Next Step.

6. On the Select Bundle page, select a bundle and then choose Next Step.

7. On the WorkSpaces Configuration page, choose a running mode and then choose Next Step.

8. On the Review & Launch WorkSpaces page, choose Launch WorkSpaces. The initial status of the WorkSpace is PENDING. When the launch is complete, the status is AVAILABLE and an invitation is sent to the email address that you specified for the user.

Step 3: Connect to the WorkSpace
After you receive the invitation email, you can connect to your WorkSpace using the client of your choice. After you sign in, the client displays the WorkSpace desktop.

Note: When you are connected to your WorkSpace from a Windows or macOS client, you can toggle the full screen display by using following command shortcuts:

Windows client: Ctrl+Alt+Enter
macOS client: Control+Option+Return
To Connect to the WorkSpace

1. Open the link in the invitation email. When prompted, type a password and activate the user. Remember this password as you will need it to sign in to your WorkSpace.

   **Note:** Passwords are case-sensitive and must be between 8 and 64 characters in length, inclusive. Passwords must contain at least one character from four of the following categories: lowercase letters (a-z), uppercase letters (A-Z), numbers (0-9), and special characters (~!@#$%^&*-_+\>[]\(){}\[]\:";'<>,.?/).

2. When prompted, download one of the client applications or launch Web Access.
   If you aren't prompted and you haven't installed a client application already, open [http://clients.amazonworkspaces.com/](http://clients.amazonworkspaces.com/) and follow the directions.

3. Start the client, enter the registration code from the invitation email, and choose Register.

4. When prompted to sign in, type the username and password for the user, and then choose Sign In.

5. (Optional) When prompted to save your credentials, choose Yes.
Launch a WorkSpace with AD Connector

Step 1: Create an AD Connector

Create AD Connector Service Account

You must have credentials for a service account in the existing directory that has been delegated the following privileges:

- Read users and groups - Required
- Join computers to the domain - Required
- Create computer objects - Required only when using Seamless Domain Join and Amazon WorkSpaces

For more information, see AD Connector Prerequisites and Delegate privileges to your service account.

To create an AD connector

1. Open the Amazon WorkSpaces Console at https://console.aws.amazon.com/workspaces/
2. In the navigation pane, choose Directories.
3. Choose Set up Directory, Create AD Connector.
4. Choose Small and enter a Description (optional).
5. Choose VPC and subnets. Choose your VPC. You may either select from the available subnets or enter No Preference. The DNS servers you specify must be accessible from each subnet.
6. For Organization name, type a unique organization name for your directory (for example, my-example-directory). This name must be at least four characters in length, consist of only alphanumeric characters and hyphens (-), and begin or end with a character other than a hyphen.
7. For Directory DNS name, type the fully-qualified name of your on-premises directory (for example, example.com).
8. (Optional) For Directory NetBIOS name, type the short name of your on-premises directory (for example, example).
9. **For DNS IP addresses**, type the IP address of at least one DNS server in your on-premises directory.

10. **For Service account username**, type the user name of a user in your on-premises directory. The user must have permissions to read users and groups, create computer objects, and join computers to the domain.

11. **For Service account password and Confirm password**, type the password for the on-premises user account.

12. Choose **Next**.

13. Review your entries and choose **Create directory**. It takes several minutes for your directory to be connected. The initial status of the directory is **Requested** and then **Creating**. When directory creation is complete, the status is **Active**.

### Step 2: Create a WorkSpace

Now you are ready to launch WorkSpaces for one or more users in your on-premises directory.

**To Launch a WorkSpace for an Existing User**

1. Open the Amazon WorkSpaces console at [https://console.aws.amazon.com/workspaces/](https://console.aws.amazon.com/workspaces/).

2. In the navigation pane, choose **WorkSpaces**.

3. Choose **Launch WorkSpaces**.

4. For **Directory**, choose the directory that you created.

5. (Important) Under **Select Subnets**, choose two private subnets created in different Availability Zones supported by WorkSpaces. WorkSpaces will be created in the subnets chosen here.

6. (Optional) If this is the first time you have launched a WorkSpace in this directory, and Amazon WorkDocs is supported in the Region, you can enable or disable Amazon WorkDocs for all users in the directory. For more information, see [Amazon WorkDocs Sync Client Help](https://console.aws.amazon.com/workspaces/) in the *Amazon WorkDocs Administration Guide*. You can also enable self-service permissions (which can be further customized later). Leave both **Enable WorkDocs** and **Enable Self-Service Permissions** as **Yes**. See the Self-Service WorkSpace Management section of this guide for more details.
7. Choose **Next**. Amazon WorkSpaces registers your AD Connector.

8. Select one or more existing users from your on-premises directory. Do not add new users to an on-premises directory through the Amazon WorkSpaces console.

9. To find users to select, you can type all or part of the user's name and choose **Search** or choose **Show All Users**. You cannot select a user without an email address.

10. After you select the users, choose **Add Selected** and then choose **Next Step**.

11. Under **Select Bundle**, choose the default WorkSpace bundle to be used for the WorkSpaces. Under **Assign WorkSpace Bundles**, you can choose a different the bundle for an individual WorkSpace if needed. When you have finished, choose **Next Step**.

12. Choose a running mode for your WorkSpaces and then choose **Next Step**. For more information, see **Manage the WorkSpace Running Mode**.

13. Choose **Launch WorkSpaces**. The initial status of the WorkSpace is **PENDING**. When the launch is complete, the status is **AVAILABLE**.

14. Send invitations to the email address for each user. For more information, see **Send an Invitation Email**.

**Step 3: Connect to the WorkSpace**

You can connect to your WorkSpace using the client of your choice. After you sign in, the client displays the WorkSpace desktop.

**Note:** When you are connected to your WorkSpace from a Windows or macOS client, you can toggle the full screen display by using the following command shortcuts:

Windows client: Ctrl+Alt+Enter

macOS client: Control+Option+Return

**To Connect to the WorkSpace**

1. Open the link in the invitation email.

2. When prompted, download one of the client applications or launch Web Access.
3. If you aren't prompted and you haven't installed a client application already, open http://clients.amazonworkspaces.com/ and follow the directions.

4. Start the client, enter the registration code from the invitation email, and choose Register.

5. When prompted to sign in, type the username and password for the user, and then choose Sign In.

6. (Optional) When prompted to save your credentials, choose Yes.
Launch a WorkSpace Using a Trusted Domain

Step 1: Establish a Trust Relationship

To Set Up the Trust Relationship

1. Set up AWS Managed Microsoft AD in your virtual private cloud (VPC). For more information, see Create Your AWS Managed Microsoft AD directory in the AWS Directory Service Administration Guide.

2. Create a trust relationship between your AWS Managed Microsoft AD and your on-premises domain. Ensure that the trust is configured as a two-way trust. For more information, see Tutorial: Create a Trust Relationship Between Your AWS Managed Microsoft AD and Your On-Premises Domain in the AWS Directory Service Administration Guide.

A two-way trust is required so that on-premises credentials can be used to manage and authenticate with WorkSpaces, and so that WorkSpaces can be provisioned to on-premises users and groups.

Step 2: Create a WorkSpace

After you establish a trust relationship between your AWS Managed Microsoft AD and your on-premises Microsoft Active Directory domain, you can provision WorkSpaces for users in the on-premises domain.

Note that you must ensure that GPO settings are replicated across domains before you can apply them to Amazon WorkSpaces.

To Launch WorkSpaces for Users in a Trusted On Premises Domain

1. Open the Amazon WorkSpaces console at https://console.aws.amazon.com/workspaces/.

2. In the navigation pane, choose WorkSpaces.

3. Choose Launch WorkSpaces.

4. On the Select a Directory page, choose the directory that you just registered and then choose Next Step.

5. On the Identify Users page, do the following:
   a. For Select trust from forest, select the trust relationship that you created.
b. Select the users from the on-premises domain and then choose Add Selected.

c. Choose Next Step.

6. Select the bundle to be used for the WorkSpaces and then choose Next Step.

7. Choose the running mode, choose the encryption settings, and configure any tags. When you are finished, choose Next Step.

8. Choose Launch WorkSpaces. Note that it can take up to 20 minutes for the WorkSpaces to become available, and up to 40 minutes if encryption is enabled. The initial status of the WorkSpace is PENDING. When the launch is complete, the status is AVAILABLE.

9. Send invitations to the email address for each user. For more information, see Send an Invitation Email.

Step 3: Connect to the WorkSpace

After you receive the invitation email, you can connect to your WorkSpace. Users can enter their user names as username, corp\username, or corp.example.com\username).

**Note:** When you are connected to your WorkSpace from a Windows or macOS client, you can toggle the full screen display by using the following command shortcuts:

Windows client: Ctrl+Alt+Enter
macOS client: Control+Option+Return

To Connect to the WorkSpace

1. Open the link in the invitation email. When prompted, type a password and activate the user. Remember this password as you will need it to sign in to your WorkSpace.

**Note:** Passwords are case-sensitive and must be between 8 and 64 characters in length, inclusive. Passwords must contain at least one character from three of the following categories: lowercase letters (a-z), uppercase letters (A-Z), numbers (0-9), and special characters (~!@#$%^&*_+-=\|\}{\[]:\;",<>.,/?).
2. When prompted, download one of the client applications or launch Web Access.
   If you aren't prompted and you haven't installed a client application already, open http://clients.amazonworkspaces.com/ and follow the directions.

3. Start the client, enter the registration code from the invitation email, and choose Register.

4. When prompted to sign in, type the username and password for the user, and then choose Sign In.

5. (Optional) When prompted to save your credentials, choose Yes.
Security

Cloud security at AWS is the highest priority. Amazon WorkSpaces natively applies several security features designed to reduce the risk of credentials and session data being compromised. For example, referring back to Figure 1, Amazon WorkSpaces use Secure Sockets Layer/Transport Layer Security (SSL/TLS) certificates and Kerberos for secure authentication (steps 1 through 6), while the streaming session is encrypted with Advanced Encryption Standard (AES) 256-bit encryption.

The Amazon WorkSpaces service automatically creates a security group for the directory service controllers and another security group for the WorkSpaces to allow granular control of network traffic. Network Access Control Lists should also be used to filter network traffic at the subnet level.

This POC has deployed Amazon WorkSpaces within private subnets to reduce the risks from internet-borne attacks.

Amazon WorkSpaces also allows the administrator to further filter client access by the following means:

- By client operating system – Windows, macOS, Android, iOS, ChromeOS, zero clients, web access
- Allow only trusted devices with root certificates installed (Windows & macOS)
- IP Access Control Groups – filter access to WorkSpaces by source CIDR ranges
- Multi-Factor Authentication (MFA) – Amazon WorkSpaces integrate with RADIUS-based MFA solutions. AWS has published walkthroughs for integrating with MFA solutions from Microsoft, Okta, and others.

These security features not only help protect the WorkSpaces instances and customer data, but also ensure secure access by authorized individuals consistent with the AWS Well-Architected Framework.
Advanced Features

Bring Your Own license (BYOL)

If your licensing agreement with Microsoft allows it, you can use your Windows 7, Windows 10 Enterprise, or Windows 10 Pro desktop images for your WorkSpaces. To do this, you must bring your own Windows License (BYOL) and provide a Windows 7 or Windows 10 image that meets the following requirements. To stay compliant with Microsoft licensing terms, run your Amazon WorkSpaces on hardware that is dedicated to you in the AWS Cloud. By bringing your own license, you can provide a consistent experience for your users. When using BYOL, AWS discounts the associated WorkSpaces cost by $4 per month. This can be a meaningful way to Cost Optimize the solution. For more information, see Amazon WorkSpaces Pricing.

**Important:** Image creation is not supported on Windows 10 systems that have been upgraded from one version of Windows 10 to a later version of Windows 10.

To get started, open the Amazon WorkSpaces console and choose Account Settings to enable your account for BYOL.

Requirements

Before you begin, verify the following:

- Your Microsoft licensing agreement allows Windows to be run in a virtual hosted environment.
- You will use a minimum of 200 Amazon WorkSpaces. This is a requirement for running your Amazon WorkSpaces on dedicated hardware. Running your Amazon WorkSpaces on dedicated hardware is required to comply with Microsoft licensing requirements.

If you plan to use GPU-enabled (Graphics and Graphics Pro) bundles, verify that you will run a minimum of 4 AlwaysOn or 20 AutoStop GPU-enabled WorkSpaces in a Region per month on dedicated hardware.
• Amazon WorkSpaces can use a management interface in the /16 IP address range. The management interface is connected to a secure Amazon WorkSpaces management network used for interactive streaming. This allows Amazon WorkSpaces to manage your WorkSpaces. For more information, see Network Interfaces. At least one of the following IP address ranges must be available for this purpose:
  o 10.0.0.0/8
  o 100.64.0.0/10
  o 172.16.0.0/12
  o 192.168.0.0/16
  o 198.18.0.0/15

• You must have a virtual machine (VM) that runs a supported 64-bit version of Windows 7 or Windows 10. For a list of supported versions, see the next section in this topic, Windows Versions That Are Supported for BYOL. The VM must also meet these requirements:
  o The Windows operating system must be activated against your key management servers.
  o The Windows operating system must have English (United States) as the primary language.
  o No software other than what is included with Windows 7 or Windows 10 can be installed in the VM. You can add additional software, such as an antivirus solution, later when you create a custom image.
  o If the VM is running Windows 10, the user profile must be placed in C:UsersDefault.
  o We recommend that you create a WorkSpaces_BYOL account with local administrator access before you share the image. The password for this account may be required later.

• Your VM must run PowerShell version 4 or later.

Windows Versions That Are Supported for BYOL

Your VM must run one of the following Windows versions:

• Windows 7 Service Pack 1
- Windows 10 Version 1607 (Anniversary Update)
- Windows 10 Version 1703 (Creators Update)
- Windows 10 Version 1709 (Fall Creators Update)
- Windows 10 Version 1803 (April 2018 Update)
- Windows 10 Version 1809 (October 2018 Update)

**Note:** Graphics and GraphicsPro bundles currently do not support Windows 10 Version 1809 (October 2018 Update) with BYOL.

**Step 1: Enable BYOL for Your Account by Using the Amazon WorkSpaces Console**

To enable BYOL for your account, you must specify a management network interface. This interface is connected to a secure Amazon WorkSpaces management network. It is used for interactive streaming of the WorkSpace desktop to Amazon WorkSpaces clients, and to allow Amazon WorkSpaces to manage the WorkSpace.

**Note:** The steps in this procedure for enabling BYOL for your account need to be performed only once per account, per Region.

**To Enable BYOL for your Account Using the Amazon WorkSpaces Console**

1. Open the Amazon WorkSpaces console at [https://console.aws.amazon.com/workspaces/](https://console.aws.amazon.com/workspaces/).
2. In the navigation pane, choose Account Settings. If your account is not currently eligible for BYOL, a message provides guidance for next steps.
3. Under Bring Your Own License (BYOL), in the Management network interface IP address range area, choose an IP address range, and then choose Display available CIDR blocks.

Amazon WorkSpaces searches for and displays available IP address ranges as IPv4 CIDR blocks, within the range that you specify. If you require a specific IP address range, you can edit the search range.

**Note:** Specify an IP address range that is outside your internal range.
4. Choose the CIDR block that you want from the list of results, and then choose **Enable BYOL**.

**Note:** This process may take several hours. While Amazon WorkSpaces is enabling your account for BYOL, proceed to the next step.

### Step 2: Run the BYOL Checker PowerShell Script on a Windows VM

After you enable BYOL for your account, you must confirm that your VM meets the requirements for BYOL. To do so, perform these steps to download and run the Amazon WorkSpaces BYOL Checker PowerShell script. The script performs a series of tests on the VM that you plan to use to create your image.

**Important:** The VM must pass all tests before you can use it for BYOL.

### To Download the BYOL Checker Script

Before you download and run the BYOL Checker script, verify that the latest Windows security updates are installed on your VM. While this script runs, it disables the Windows Update service.

1. Navigate to the following URL to download the BYOL Checker script .zip file: [https://d2zdcak60k1ljz.cloudfront.net/BYOLChecker.zip](https://d2zdcak60k1ljz.cloudfront.net/BYOLChecker.zip).
2. When prompted, right-click the Download link and choose Save target as.
3. In the Save as dialog box, navigate to the location where you want to save the script file, and choose Save. For example, you can create a folder in Drive C for this purpose, such as C:\BYOL.
4. When a message notifies you that the BYOL Checker script .zip file has finished downloading, choose OK to close the message.
5. Windows Explorer opens in the location where you saved the .zip file. Right-click the file, and choose Extract All.
6. In the **Select a Destination and Extract Files** dialog box, navigate to the location where you want to extract the script (for example, C:\BYOL\), and choose **Extract**.
7. If you created a folder for the BYOL script, you can copy the extracted files to the root of the folder.
8. Delete the original .zip files so that only the extracted files remain. Close any open applications and windows (for example, close Windows Explorer and your browser, if they are open).

9. If the VM is running on EC2, enable user data execution before you create an AMI from the instance. If the VM is running on on-premises software, this step is not required.

Perform these steps to run the BYOL Checker script.

To Run the BYOL Checker Script

1. From the Windows desktop, open Windows PowerShell. Choose the Windows Start button, right-click Windows PowerShell, and choose Run as administrator. If you are prompted by User Account Control to choose whether you want PowerShell to make changes to your device, choose Yes.

2. At the PowerShell command prompt, type the commands required to change to the directory where the BYOL Checker script is located. For example, if the script is located in the BYOL directory, type the following commands. After each command, press Enter:
   
   ```
   cd/
   cd byol
   ```

3. Type the following command to update the PowerShell execution policy on the computer. Doing so allows the BYOL Checker script to run:

   ```powershell
   Set-ExecutionPolicy Unrestricted
   ```

4. When prompted to confirm whether to change the PowerShell execution policy, type A to specify Yes for all.

5. Type the following command to run the BYOL Checker script:

   ```powershell
   \BYOLChecker.ps1
   ```

6. If a security notification appears, press the R key to run once.

7. In the Amazon WorkSpaces Image Validation dialog box, choose Begin Tests.

8. After each test is completed, you can view the status of the test. For any test with a status of Failed, choose Info to display information about how to resolve the issue that caused the failure. If any tests display a status of Warning, choose the Fix all Warnings button.
9. If applicable, resolve any issues that cause test failures and warnings, and repeat steps 7 and 8 until the VM passes all tests. All failures and warnings must be resolved before you export the VM.

10. The BYOL script checker generates two log files. These two files are located in the directory that contains the BYOL Checker script files and are named as follows:
   BYOLPrevalidationlogYYYYMMDDT
   ImageInfo.text

Do not delete these files. If an issue occurs, they may be helpful in troubleshooting.

Step 3: Export the VM from Your Virtualization Environment

To create an image for BYOL, you must first export the VM from your virtualization environment. The VM must be on a single volume that is at least 10 GB and smaller than 80 GB. For more information, see the documentation for your virtualization environment and Export Your VM from its Virtualization Environment in the VM Import/Export User Guide.

Important: Before you export the VM from your virtualization environment, verify that the VM meets the requirements for running Sysprep. Test run Sysprep on the image to make sure it works, but do not run Sysprep on the image that you upload for BYOL. Amazon WorkSpaces runs Sysprep as part of the BYOL image creation process. For information about Sysprep, see Sysprep (System Preparation) Overview in the Microsoft documentation.

Step 4: Import the VM as an Image into Amazon EC2

After you export your VM, review the VM Import/Export Requirements for importing Windows operating systems from a VM. You can use the AWS Command Line Interface (AWS CLI) import-image command or the ImportImage API operation to import your VM into Amazon EC2 as an Amazon Machine Image (AMI). For more information, see Importing a VM as an Image in the VM Import/Export User Guide.
Step 5: Create a BYOL Image by Using the Amazon WorkSpaces Console

Perform these steps to create an Amazon WorkSpaces BYOL image.

**Note**

To perform this procedure, verify that you have permissions to:

- Call Amazon WorkSpaces `ImportWorkspacesImage`.
- Call EC2 `DescribeImages` on the EC2 image that you want to use to create the BYOL image.
- Call EC2 `ModifyImageAttribute` on the EC2 image that you want to use to create the BYOL image.

For more information, see Changing Permissions for an IAM User in the IAM User Guide.

To Create an Image from the Windows VM

1. Open the Amazon WorkSpaces console at [https://console.aws.amazon.com/workspaces/](https://console.aws.amazon.com/workspaces/).
2. In the navigation pane, choose Images.
4. In the Create BYOL Image dialog box, do the following:
   - For **AMI ID**, click the EC2 Console link, and choose the EC2 image that you imported as described in the previous section (Step 4: Import the VM as an Image into EC2). The image name must begin with `ami-` and be followed by the identifier for the AMI (for example, `ami-5731123e`).
   - For **BYOL image name**, type a unique name for the image.
   - For **Image description**, type a description to help you quickly identify the image.
5. Choose Create.

While your image is being created, the image status in the image registry of the console appears as Pending. If the image validation does not succeed, the console displays an error code. When the image creation is complete, the status changes to Available.
Step 6: Create a Custom Bundle From the BYOL Image

After your BYOL image is created, you can use the image to create a custom bundle. For information, see Create a Custom WorkSpaces Bundle.

Step 7: Register a Directory for Dedicated WorkSpaces

To use BYOL images for WorkSpaces, you must register a directory for this purpose. To do so, perform these steps.

To Register a Directory for Dedicated WorkSpaces

1. Open the Amazon WorkSpaces console at https://console.aws.amazon.com/workspaces/.
2. In the navigation pane, choose Directories.
3. Select the directory and choose Actions, Register.
4. In the Register directory dialog box, for Enable Dedicated WorkSpaces, choose Yes.
5. Choose Register.

If you have already registered AWS Directory Service for Microsoft Active Directory or an AD Connector directory for WorkSpaces that does not run on dedicated hardware, you can set up a new Microsoft Active Directory or AD Connector directory for this purpose. You can also deregister the directory and then reregister it as a directory for dedicated WorkSpaces. To do so, perform these steps.

Note: You can only perform this procedure if no WorkSpaces are associated with the directory.

To Deregister a Directory and Reregister it for Dedicated WorkSpaces

1. Open the Amazon WorkSpaces console at https://console.aws.amazon.com/workspaces/.
2. Terminate existing WorkSpaces.
3. In the navigation pane, choose Directories.
4. Select the directory and choose Actions, Deregister.
5. When prompted for confirmation, choose Deregister.
6. Select the directory again and choose **Actions, Register**.

7. In the Register directory dialog box, for Enable Dedicated WorkSpaces, choose Yes.

8. Choose **Register**.

**Step 8: Launch Your BYOL WorkSpaces**

After you register a directory for dedicated WorkSpaces, you can launch your BYOL WorkSpaces in that directory. For information about how to launch WorkSpaces, see [Launch a Virtual Desktop Using Amazon WorkSpaces](#).
Application Delivery – Custom Bundles

Create a Custom WorkSpaces Bundle

After you’ve launched a Windows or Amazon Linux WorkSpace and customized it, you can create an image from the WorkSpace and then create a custom bundle from the image. You can specify this bundle when you launch new WorkSpaces to ensure that they have the same configuration and software as the WorkSpace you used to create the bundle. This capability enhances Operational Excellence by allowing you to tailor the WorkSpaces environment to your business requirements, and do so automatically at scale.

**Important:** If you plan to create an image from a Windows 10 WorkSpace, use an image that hasn’t been upgraded. Image creation is not supported on Windows 10 systems that have been upgraded from one version of Windows 10 to a later version of Windows 10.

Requirements to Create Windows Custom Images

- All applications to be included in the image must be installed on the C:\ drive, or the user profile in D:\Users\username. They must also be compatible with Microsoft Sysprep.

  **Note:** If you are preparing a bring your own Microsoft Windows License (BYOL) Windows 10 image, place the user profile in C:\Users\Default.

- The user profile must exist and its total size (files and data) must be less than 10 GB.
- The C:\ drive must have enough available space for the contents of the user profile, plus an additional 2 GB.
- All application services running on the WorkSpace must use a local system account instead of domain user credentials. For example, you cannot have a Microsoft SQL Server Express installation running with a domain user’s credentials.
- The following components are required in an image; otherwise, the WorkSpaces you launch from the image will not function correctly:
Requirements to Create Amazon Linux Custom Images

- All applications to be included in the image must be installed outside of the /home directory (or user volume).
- The root volume (/) should be less than 97% full.
- The following components are required in an image; otherwise, the WorkSpaces you launch from the image will not function correctly:
  - Cloud-init
  - Teradici PCoIP agents and drivers
  - Skylight agent

Best Practices

Before you create an image from a WorkSpace, do the following:

- Install all operating system and application updates on the WorkSpace.
- Delete cached data from the WorkSpace that shouldn’t be included in the bundle (for example, browser history, cached files, and browser cookies).
- Delete configuration settings from the WorkSpace that shouldn’t be included in the bundle (for example, email profiles).
- Switch to Dynamic IP Address settings using DHCP.
To Create a Custom Bundle

1. If you are still connected to the WorkSpace, disconnect.

2. Open the Amazon WorkSpaces console at [https://console.aws.amazon.com/workspaces/](https://console.aws.amazon.com/workspaces/).

3. In the navigation pane, choose WorkSpaces.

4. Select the WorkSpace and choose Actions, Create Image.

5. A message displays prompting you to restart your WorkSpace before continuing, to update your WorkSpaces software to the latest version necessary.

   Restart your WorkSpace if needed by closing the message and following the steps in Restart a WorkSpace. When you’re done, repeat the previous step, and choose Next when this message appears.

6. Type an image name and a description that will help you identify the image, and then choose Create Image. While the image is being created, the status of the WorkSpace is Suspended and the WorkSpace is unavailable.

7. In the navigation pane, choose Images. The image is complete when the status changes to Available.

8. Select the image and choose Actions, Create Bundle.

9. Type a bundle name and a description, and then do the following:
   
   o For Bundle Type, choose the hardware from which your WorkSpace is launched.
   
   o For Root Volume Size, leave the default value or type a new value that is equal or greater than the current size. Then, type a value for User Volume Size.

   The available sizes for the root volume (for Microsoft Windows, the C: drive, for Linux, /) and the user volume (for Windows, the D: drive; for Linux, /home) are as follows:

   o Root: 80 GB, User: 10 GB, 50 GB, or 100 GB
   
   o Root: 175 GB, User: 100 GB

   Alternatively, you can expand the root and user volumes up to 1000 GB each.

10. Choose Create Bundle.
Image Creation for Windows WorkSpaces

When you create an image from a Windows WorkSpace, the entire contents of the C:\ drive are included. The entire contents of the user profile in D:\Users\username are included except for the following:

- Contacts
- Downloads
- Music
- Pictures
- Saved games
- Videos
- Podcasts
- Virtual machines
- .virtual box
- Tracing
- appdata\local\temp
- appdata\roaming\apple computer\mobilesync\n- appdata\roaming\apple computer\logs\n- appdata\roaming\apple computer\itunes\iphone software updates\n- appdata\roaming\macromedia\flash player\macromedia.com\support\flashplayer\sys\n- appdata\roaming\macromedia\flash player\#sharedobjects\n- appdata\roaming\adobe\flash player\assetcache\n- appdata\roaming\microsoft\windows\recent\n- appdata\roaming\microsoft\office\recent\n- appdata\roaming\microsoft office\live meeting
- appdata\roaming\microsoft shared\livemeeting shared\n- appdata\roaming\mozilla\firefox\crash reports\n
• appdata\roaming\mcafee\common framework\n• appdata\local\microsoft\feeds cache
• appdata\local\microsoft\windows\temporary internet files\n• appdata\local\microsoft\windows\history\n• appdata\local\microsoft\internet explorer\domstore\n• appdata\local\microsoft\internet explorer\imagestore\n• appdata\local\microsoft\internet explorer\iconcache\n• appdata\local\microsoft\internet explorer\domstore\n• appdata\local\microsoft\internet explorer\imagestore\n• appdata\local\microsoft\internet explorer\recovery\n• appdata\local\mozilla\firefox\profiles\n
Image Creation for Amazon Linux WorkSpaces

When you create an image from an Amazon Linux WorkSpace, the entire contents of the user volume (/home) are removed. The contents of the root volume (/) are included, except the following folders and keys, which are removed:

• /tmp
• /var/spool/mail
• /var/tmp
• /var/lib/dhcp
• /var/lib/cloud
• /var/cache
• /var/backups
• /etc/sudoers.d
• /etc/udev/rules.d/70-persistent-net.rules
• /etc/network/interfaces.d/50-cloud-init.cfg
• /etc/security/access.conf
• /var/log/amazon/ssm
• /var/log/pcoip-agent
• /var/log/skylight
• /var/lock/.skylight.domain-join.lock
• /var/lib/skylight/domain-join-status
• /var/lib/skylight/configuration-data
• /var/lib/skylight/config-data.json
• /home

The following keys are shredded during custom image creation:

• /etc/ssh/ssh_host_*_key
• /etc/ssh/ssh_host_*_key.pub
• /var/lib/skylight/tls.*
• /var/lib/skylight/private.key
• /var/lib/skylight/public.key
Application Delivery – Amazon WorkSpaces

Application Manager

Another way to enhance Operational Excellence is to deploy Amazon WorkSpaces Application Manager (Amazon WAM) with WorkSpaces. Amazon WAM offers a fast, flexible, and secure way for you to deploy and manage applications for WorkSpaces. Amazon WAM accelerates software deployment, updates, patching, and retirement by packaging Microsoft Windows desktop applications into virtual containers that run as though they are installed natively.

Amazon WAM is fully integrated with the AWS Management Console, and allows you to build an application catalog from your line-of-business applications, third-party applications that you own the license for, and applications purchased through the AWS Marketplace.

Overview

You can deploy subscriptions to your Amazon WorkSpaces users from the AWS Marketplace, your line-of-business applications, or applications where you already own the licenses. The following illustration shows the process to deploy applications.

Your users only have access to the Amazon WAM applications that you assign to them. The following is the process to assign an application in your application catalog to a user.
See [Managing Your Amazon WAM Applications](#) in the *Amazon WorkSpaces Application Manager Administration Guide* for a complete walkthrough of setting up and administrating your WAM environment with Amazon WorkSpaces.
API and Command Line Interface Tools

Advanced users can manipulate the Amazon WorkSpaces environment programmatically via a number of tools provided by AWS. These tools can help Architects build custom scripts and applets to improve Operational Excellence:

- **AWS Command Line Interface**
- **AWS Tools for PowerShell**
- AWS SDKs for various programming languages:
  - AWS SDK for Java
  - AWS SDK for .NET
  - AWS SDK for JavaScript
  - AWS SDK for Ruby
  - AWS SDK for Python (Boto)
  - AWS SDK for PHP
  - AWS SDK for Go
  - AWS Mobile SDK for iOS
  - AWS Mobile SDK for Android

**Command Line Interface Example**

The following example creates a WorkSpace for user `jimsmith` in the specified directory, from the specified bundle.

**Command:**

```bash
aws workspaces create-workspaces --cli-input-json file://create-workspaces.json
```

**Input:**

```json
This is the contents of the create-workspaces.json file:
{
```
See the WorkSpaces chapter of the AWS CLI Reference for the complete list of commands and options.

Amazon WorkSpaces Cost Optimizer

To help customers monitor their WorkSpace usage and enhance the Cost Optimization of the environment, AWS offers the Amazon WorkSpaces Cost Optimizer, a solution that analyzes all of your Amazon WorkSpaces usage data and automatically converts the WorkSpace to the most cost-effective billing option (hourly or monthly) depending on a user's individual usage. This solution is easy to deploy and uses AWS CloudFormation to automatically provision and configure the necessary AWS services to convert individual WorkSpaces.

See the WorkSpaces Cost Optimizer Administrator's Guide for more information.

Self-Service WorkSpace Management

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

This capability ties into the Operational Excellence and Cost Optimization pillars of the AWS Well-Architected Framework:

- Cache their credentials on their client. This lets them reconnect to their WorkSpace without re-entering their credentials.
- Restart their WorkSpace.
- Increase the volume size of the C: drive and D: drive on their WorkSpace.
- Change the compute type (bundle) for their WorkSpace.
• Switch the running mode of their WorkSpace.
• Rebuild their WorkSpace.

To enable one or more of these capabilities for your users, perform the following steps.

To Enable Self-Service Management Capabilities for your Users

1. Open the Amazon WorkSpaces console at https://console.aws.amazon.com/workspaces/.
2. In the navigation pane, choose Directories.
3. Select your directory, and choose Actions, Update Details.
4. Expand User Self-Service Permissions. Enable or disable the following options as required to determine the WorkSpace management tasks that users can perform from their client:
   - **Remember me** — Users can choose whether to cache their credentials on their client. The credentials are cached in RAM only. When users choose to cache their credentials, they can reconnect to their WorkSpaces without re-entering their credentials.
   - **Restart WorkSpace from client** — Users can restart their WorkSpace. Restarting disconnects users from their WorkSpace, shuts it down, and restarts it. The user data, operating system, and system settings are not affected.
   - **Increase volume size** — Users can expand the root and user volumes on their WorkSpace to a specified size without contacting IT support. Users can increase the size of the root volume (C: drive) up to 175 GB, and the size of the user volume (D: drive) up to 100 GB. WorkSpace root and user volumes come in set groups that can’t be changed. The available groups are [Root(GB), User(GB)]: [80, 10], [80, 50], [80, 100], [175 to 1000, 100 to 1000].

For a newly created WorkSpace, users must wait 6 hours before they can increase the size of these drives. After that, they can do so only once in a 6-hour period. When a volume size increase is in progress, users can perform most tasks on their WorkSpace. The tasks that they can’t perform are: changing their WorkSpace compute type, switching their WorkSpace running mode, restarting their WorkSpace, or rebuilding their WorkSpace.
Note: If users increase the volume size on their WorkSpace, this will increase the billing rate for their WorkSpace.

- **Change compute type** — Users can switch their WorkSpace between compute types (bundles). For a newly created WorkSpace, users must wait 6 hours before they can switch to a different bundle. After that, they can switch to a larger bundle only once in a 6-hour period, or to a smaller bundle once in a 30-day period. When a WorkSpace compute type change is in progress, users are disconnected from their WorkSpace, and they can't use or change the WorkSpace. This process may take up to an hour.

Note: If users change their WorkSpace compute type, this may change the billing rate for their WorkSpace.

- **Switch running mode** — Users can switch their WorkSpace between the **Always on** and **AutoStop** running modes. For more information, see [Manage the WorkSpace Running Mode](#).

Note: If users switch the running mode of their WorkSpace, this will change the billing rate for their WorkSpace.

- **Rebuild WorkSpace from client** — Users can rebuild the operating system of a WorkSpace to its original state. When a WorkSpace is rebuilt, the user volume (D: drive) is recreated from the latest backup. Because backups are completed every 12 hours, users' data may be up to 12 hours old. For a newly created WorkSpace, users must wait 12 hours before they can rebuild their WorkSpace. When a WorkSpace rebuild is in progress, users are disconnected from their WorkSpace, and they can't use or make changes to their WorkSpace. This process may take up to an hour.

5. Choose **Update** or **Update and Exit**.
Conclusion

This document provides a best practices approach to creating a proof-of-concept Amazon WorkSpaces environment and implements a design that will scale for a subsequent production rollout. By adhering to the five pillars of the AWS Well-Architected Framework (operational excellence, security, reliability, performance efficiency, and cost optimization) from the beginning, customers are able to minimize risks and maximize their benefits from the technology to drive better business outcomes.

Contributors

Contributors to this document include:

- Doug Martin, Sr. EUC Solutions Architect (Author)
- Rakesh Saini, Sr. Product Manager – WorkSpaces (Reviewer)
- Hassan TA, Sr. Product Manager – Technical, WorkSpaces (Reviewer)
- Joao Paulo “JP” Santana, Solutions Architect (Technical Reviewer)

Document Revisions

<table>
<thead>
<tr>
<th>Date</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>July 2019</td>
<td>Changed title and made minor text changes.</td>
</tr>
<tr>
<td>June 2019</td>
<td>First publication</td>
</tr>
</tbody>
</table>
Notes

2. [https://docs.aws.amazon.com/workspaces/latest/adminguide/workspaces-access-control.html](https://docs.aws.amazon.com/workspaces/latest/adminguide/workspaces-access-control.html)
4. [https://www.infrastructure.aws/](https://www.infrastructure.aws/)
6. [https://docs.aws.amazon.com/directoryservice/latest/admin-guide/what_is.html#choosing_an_option](https://docs.aws.amazon.com/directoryservice/latest/admin-guide/what_is.html#choosing_an_option)