# Table of Contents

What Is Amazon WAM? .................................................................................................................. 1  
Overview ........................................................................................................................................ 1  
Related Services ............................................................................................................................. 2  
Pricing .............................................................................................................................................. 2  
Get Started ..................................................................................................................................... 3  
Prerequisites ................................................................................................................................... 3  
Step 1: Select a Subscription Plan ................................................................................................. 3  
Step 2: Add an Application to Your Catalog ............................................................................... 4  
Step 3: Assign an Application to a User ................................................................................. 4  
Step 4: Access the Applications Assigned to You ................................................................... 4  
Managing Applications .................................................................................................................. 5  
Building an Application Catalog ................................................................................................... 5  
Subscribing to an Application ......................................................................................................... 5  
Uploading an Application ............................................................................................................... 6  
Removing an Application .............................................................................................................. 7  
Assigning an Application ............................................................................................................... 7  
Setting Up the Amazon WAM Installer and Accessing Assigned Applications ....................... 8  
Updating an Application ................................................................................................................ 10  
Versions ....................................................................................................................................... 10  
Constraints ................................................................................................................................... 11  
Tags .............................................................................................................................................. 12  
Users and Groups ......................................................................................................................... 12  
Support Details ............................................................................................................................ 12  
Monitoring Application Usage ..................................................................................................... 12  
Applications ................................................................................................................................... 12  
Users ............................................................................................................................................ 13  
Removing All Application Assignments ..................................................................................... 13  
Packaging and Validating ............................................................................................................. 15  
Packaging an Application .............................................................................................................. 15  
Step 1: Launch a Packaging Instance ......................................................................................... 16  
Step 2: Create an Application Package ..................................................................................... 16  
Step 3: Capture the Application Installation ........................................................................... 17  
Step 4: Upload the Application Package ................................................................................... 19  
Packaging Specific Applications .................................................................................................. 20  
7-Zip ........................................................................................................................................... 20  
Chrome ....................................................................................................................................... 22  
Firefox ........................................................................................................................................ 24  
Validating a Package ..................................................................................................................... 26  
Step 1: Launch the Validation Instance ..................................................................................... 26  
Step 2: Validate the Application Package ................................................................................ 27  
Updating a Package ...................................................................................................................... 28  
Creating a New Application Version ........................................................................................... 28  
Working with Files and Folders ................................................................................................. 29  
Working With the Registry .......................................................................................................... 33  
Working with Environment Variables ......................................................................................... 35  
Working with Fonts ...................................................................................................................... 36  
Working with Startup Items ......................................................................................................... 38  
Working with Services and Drivers ............................................................................................. 40  
Adding Required Applications or Scripts ............................................................................... 42  
Sandboxing Your Package .......................................................................................................... 45  
Setting File Security ...................................................................................................................... 47  
Using Licenses ............................................................................................................................. 48  
Finding a Package ......................................................................................................................... 49  
Sharing a Package ......................................................................................................................... 49
What Is Amazon WorkSpaces Application Manager?

Amazon WorkSpaces Application Manager (Amazon WAM) offers a fast, flexible, and secure way for you to deploy and manage applications for Amazon WorkSpaces with Windows. 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 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.
Related Services

Amazon WAM is only available on Amazon WorkSpaces with Windows. For more information, see the Amazon WorkSpaces Application Manager FAQ.

Amazon WAM is not available in every AWS Region. For more information about the supported Regions, see the AWS Regional Services List.

Pricing

Amazon WAM is available in two versions: Amazon WAM Lite and Amazon WAM Standard. With Amazon WAM Lite, you can manage and deliver applications from the AWS Marketplace free of charge. You pay only for the applications that your users activate. With Amazon WAM Standard, you can build your application catalog with line-of-business applications, third-party applications for which you own licenses, and applications from the AWS Marketplace for Desktop Apps.

With Amazon WAM, you pay only for the applications that are used, on a per-user basis. For more information, see Pricing and Plans.
Get Started with Amazon WAM

In this tutorial, you'll learn how to deploy an application from the AWS Marketplace to a Windows WorkSpace using Amazon WorkSpaces Application Manager (Amazon WAM).

Note

- Amazon WAM is not available in every AWS Region. For more information about the supported Regions, see the AWS Regional Services List.
- Amazon WAM is not available for Linux WorkSpaces.

Prerequisites

Before you begin, do the following:

- Create an Amazon WorkSpaces directory. Amazon WAM supports AWS Directory Service for Microsoft Active Directory (also known as AWS Managed Microsoft AD), Simple AD, and AD Connector.

  Important
  If you have set up a trust relationship between AWS Managed Microsoft AD in the AWS Cloud and your existing on-premises Microsoft Active Directory, you can't assign applications to users in your on-premises Microsoft Active Directory.

- Create a user and WorkSpace for that user using the directory that you created.

For more information, see the following documentation in the Amazon WorkSpaces Administration Guide:

- Launch a WorkSpace Using AWS Managed Microsoft AD
- Launch a WorkSpace Using Simple AD
- Launch a WorkSpace Using AD Connector

Step 1: Select a Subscription Plan

If you've used the Amazon WAM console before, you have already selected a subscription plan. Otherwise, you must select a subscription plan. For more information, see Pricing and Plans.

To select a subscription plan

2. Select the region that contains your WorkSpaces.
3. On the navigation pane, choose Subscriptions.
4. If you are new to Amazon WAM, you see a welcome page. On the welcome page, choose Get started building your catalog. On the Select subscription plan page, choose WAM Lite or WAM Standard, and then choose Confirm. Remain on the Add applications to catalog from AWS Marketplace page and go to Step 2: Add an Application to Your Catalog (p. 4).
5. Otherwise, you see the Subscription plan page. Check the value of Current subscription plan. To change the plan from WAM Lite to WAM Standard, choose Upgrade. To change the plan from WAM Standard to WAM Lite, choose Downgrade. When you are finished, choose Applications from the navigation pane, and then choose Add application from AWS Marketplace. Remain on the Add applications to catalog from AWS Marketplace page and go to the next step.
Step 2: Add an Application to Your Catalog

You can build your catalog from the AWS Marketplace, your line-of-business applications, and applications that you have already licensed. Use the following procedure to add an application from the AWS Marketplace to your catalog.

To add an application to your application catalog

1. In the search AWS Marketplace field, type the name of an application (for example, *solitaire*) and then press Enter. Select an application from the results (for example, Big Solitaires 3).
2. On the application page, choose **Accept terms and subscribe**.
3. Choose **Return to application catalog**.
4. If needed, set the source to **Source: AWS Marketplace**. After several minutes, the application appears in the list of subscribed applications.

Step 3: Assign an Application to a User

Your users can only access the Amazon WAM applications that you assign to them. Use the following procedure to assign a subscribed application to a user.

To assign an application to a user

1. Locate the application and select its checkbox.
2. Choose **Actions**, **Assign application(s) to users**.
3. On the **Select users** page, do the following:
   a. For **Directory**, select the directory that you used for the WorkSpace.
   b. For **Type**, choose **Users**.
   c. In the search field, type the name of the user, or leave the field empty to list all users for the directory.
   d. Choose **Search**.
   e. For **Search results**, select the checkbox next to the user and then choose the right arrow (>) to add the user to the list.
   f. Choose **Next**.
4. Complete the **Configure options** page as needed, and then choose **Review**. Note that you can change the default options only if you are using the Amazon WAM Standard subscription plan.
5. Choose **Confirm and assign**. The selected application is assigned to the selected user.

Step 4: Access the Applications Assigned to You

Your users use the Amazon WorkSpaces Application Manager client application (Amazon WAM) on their WorkSpaces to access the applications that you assign to them. If it's not already available, you must set up the Amazon WAM installer on the WorkSpace desktop. After you set up the Amazon WAM installer, your users will be able to access the applications that are assigned to them. For more information, see **Setting Up the Amazon WAM Installer and Accessing Assigned Applications**.
Managing Your Amazon WAM Applications

You can use Amazon WorkSpaces Application Manager (Amazon WAM) to deploy applications to the WorkSpaces that you created for your users. First, you add applications to your application catalog. Then you assign applications to the users. After you assign applications to users and set up the Amazon WAM installer on their WorkSpaces, they can connect to their WorkSpaces and use the Amazon WAM client application to install and use the applications. You can also use Amazon WAM to monitor application usage and deploy updated versions of the applications you've assigned to the users.

Contents
• Building an Application Catalog (p. 5)
• Assigning an Amazon WAM Application to Users and Groups (p. 7)
• Setting Up the Amazon WAM Installer and Accessing Assigned Applications (p. 8)
• Updating an Application (p. 10)
• Monitoring Application Usage (p. 12)
• Removing All Application Assignments (p. 13)

Building an Application Catalog

After you add a Amazon WAM application to your application catalog, you can assign it to one or more of your WorkSpaces users. You can add applications to your catalog by subscribing to them from the AWS Marketplace. With Amazon WAM Standard, you can also upload your line-of-business applications to your application catalog or any third-party applications for which you have a license.

Note
Amazon WAM is not available in every AWS Region. For more information about the supported Regions, see the AWS Regional Services List.

Contents
• Subscribing to an Application (p. 5)
• Uploading an Application (p. 6)
• Removing an Application (p. 7)

Subscribing to an Application

You can subscribe to applications from the AWS Marketplace, which adds them to the application catalog.
To subscribe to an application from the AWS Marketplace

1. Open the WorkSpaces console at https://console.aws.amazon.com/workspaces/ and choose Applications in the navigation pane.
2. Choose Add application from AWS Marketplace.
3. Find an application and choose the link to open its detail page.
4. In the application detail page, select the version or versions of the application to add to your catalog and choose Accept terms and subscribe.
5. Choose Return to application catalog.
6. If needed, set the source to AWS Marketplace. After several minutes, the application appears in the list of subscribed applications.

You can now assign this application to users or groups. For more information, see Assigning an Amazon WAM Application to Users and Groups (p. 7).

Uploading an Application

You can upload your line-of-business applications or licensed third-party applications to your application catalog.

**Prerequisite**

Before you begin, package your application for upload. For more information, see Package an Application with Amazon WorkSpaces Application Manager Studio (p. 15).

**To upload an application package**

1. Open the WorkSpaces console at https://console.aws.amazon.com/workspaces/ and choose Applications in the navigation pane.
2. Choose Create new application.
3. On the Enter application details page, do the following:
   a. For Application name, enter the name of the application.
   b. For Short description, enter a description for the application search page.
   c. For Description, enter a description of the application.
   d. For Category, select an appropriate application category.
   e. For Provided by, enter the name of the application publisher.
   f. (Optional) For Vendor, enter the name of the application creator, if it is different than the application publisher.
   g. Choose Next.
4. On the Enter support details page, do the following:
   a. For Email contact, enter the email address that users should use to get support.
   b. For Support link, enter the URL that users should use to get support.
Removing an Application

You can remove any application from your application catalog when you are finished with it. For example, remove an application when your users stop using it, when you want to replace an old version of an application with a newer one, or when you want to replace one application with a different one.

Prerequisite

Before you can remove an application from the application catalog, you must remove all assignments. For more information, see Users and Groups (p. 12).

To remove an application from the AWS Marketplace

1. Open the WorkSpaces console at https://console.aws.amazon.com/workspaces/ and choose Applications in the navigation pane.
2. Set the source to AWS Marketplace.
3. Find the application and choose the link to open its detail page.
4. In the application detail page, choose Unsubscribe.

To remove an application that you uploaded

1. Open the WorkSpaces console at https://console.aws.amazon.com/workspaces/ and choose Applications in the navigation pane.
2. Set the source to Your own applications.
3. Find the application and select its check box.
4. Choose Actions, Delete application.

Assigning an Amazon WAM Application to Users and Groups

After you add an application to the application catalog, you can assign it to one or more users or groups.
When you assign an application to a group, only users who are direct members of the group receive the assigned applications. Users who are members of sub-groups of the assigned group do not receive the assigned applications.

To assign an application to a user or group

1. Open the WorkSpaces console at https://console.aws.amazon.com/workspaces/ and choose Applications in the navigation pane.
2. Set the source to Your own applications if the application is an uploaded application or AWS Marketplace if the application is a subscribed application.
3. Select up to five applications to assign, and then choose Actions, Assign application(s) to users.
4. On the Select users page, do the following:
   a. For Directory, select the AWS Directory Service directory that you used for the WorkSpaces.
   b. For Type, select the type of entity to search for in the directory (users or groups).
   c. (Optional) For Fields containing, type all or part of the user or group name. Leave this field empty to search for all users or groups in the directory.
   d. Choose Search.
   e. Select the checkboxes for the users or groups, and then choose the right arrow icon (>) to add them to the list
   f. Choose Next.
5. On the Configure assignment options page, do the following. Note that you can change the default options only if you are using Amazon WAM Standard.
   a. For Version, select the version of the application.
   b. For Installation Type, select the installation type. Note that required applications are installed on a user's WorkSpace automatically when the user starts the Amazon WorkSpaces Application Manager client application, while the user must install optional applications manually.
   c. For Auto update, indicate whether applications should be updated to new versions automatically.
   d. Choose Review.
6. On the Review page, review the assignments, make any changes needed, and choose Confirm and assign.

Setting Up the Amazon WAM Installer and Accessing Assigned Applications

Your users use the Amazon WorkSpaces Application Manager client application (Amazon WAM) on their WorkSpaces to access the applications that you assign to them. If it's not already available, you must set up the Amazon WAM installer on the WorkSpace desktop. After you set up the Amazon WAM installer, your users will be able to access the applications that are assigned to them.

Note
The Amazon WAM client application requires 4 GB of disk space on the WorkSpace for caching.

To set up the Amazon WAM installer and install and start assigned applications

1. Connect to your WorkSpace using your preferred client application or web browser. For more information, see Getting Started with Your WorkSpace in the Amazon WorkSpaces User Guide;
2. If it's already available, choose the Amazon WAM shortcut on the desktop of your WorkSpace to start the Amazon WorkSpaces Application Manager client application.
If there is no Amazon WAM shortcut on the desktop, reboot the WorkSpace to add the Amazon WorkSpaces Application Manager client application. If the Amazon WAM shortcut is still not there after rebooting the WorkSpace, do the following:

a. On the Windows Start menu, choose Windows System, then choose Command Prompt.
b. In the Command Prompt window, enter the following commands, one at a time, and press Enter after each command:

```
c:\cd "C:\Program Files\Amazon"
 C:\WorkSpacesApplicationManager.exe
```

If the WorkSpacesApplicationManager.exe file is not present on your WorkSpace, contact the AWS Support Center.
c. When asked whether to allow Amazon WorkSpaces Application Manager to make changes to your device, choose Yes.
d. After the WorkSpaces Application Manager installation program has finished running, choose Finish.
e. Close the Command Prompt window.

You should now see the Amazon WAM shortcut on your desktop.

3. Choose the Amazon WAM shortcut to access your applications. Do one of the following:

   • To access the applications that have been assigned to you and are already installed, choose MY APPS.
   • To find applications that have been assigned to you but have not been installed, choose DISCOVER. To install an application, choose the triangle (install).

   **Note**
   Required applications are installed on a user's WorkSpace automatically when the user starts the Amazon WAM client application and appear in Amazon WAM under MY APPS. Optional applications must be installed manually and appear in Amazon WAM under DISCOVER. For more information about specifying whether applications are required or optional, see Assigning an Amazon WAM Application to Users and Groups.

4. You can start your Amazon WAM applications by using the Amazon WAM client application or the Windows Start menu. For more information about working with the Amazon WAM client application, see Using the Amazon WorkSpaces Application Manager Client Application in the Amazon WAM User Guide.

   **Note**
   You cannot access applications through the Windows Start menu until after the Amazon WAM application has been installed on the WorkSpace. If the Amazon WAM application is not running, you won't see your installed Amazon WAM applications on the Windows Start menu. To start the Amazon WAM application, choose the Amazon WAM shortcut on the desktop of your WorkSpace.
Updating an Application

From the application detail page, you can perform the following tasks:

- View and edit the application versions, and create a new version
- View and edit the constraints for the application
- View and edit the tags for the application
- View and edit the assignments for the application
- View the support details for the application

Amazon WAM Lite subscribers have limited features. Amazon WAM Standard subscribers can assign users to the applications they have uploaded and configure the assignment options, such as controlling versions, managing the update settings, or configuring the installation type.

To open the detail page for an application

1. Open the WorkSpaces console at https://console.aws.amazon.com/workspaces/ and choose Applications in the navigation pane.
2. To view the details for one of your own applications, set Source to Your own applications. To view the details for a subscribed application, change Source to AWS Marketplace.
3. Open the application detail page by choosing the name of the application.
4. If the application is one of your own applications, you can edit the information about the application, such as the name, vendor, provider, and description. To edit these, choose the pencil icon next to the application name, enter the new information, and choose Save.

Topics

- Versions (p. 10)
- Constraints (p. 11)
- Tags (p. 12)
- Users and Groups (p. 12)
- Support Details (p. 12)

Versions

You can view the version information for all of the applications in your catalog, regardless of the source. You can also update the versions for your own applications. To view the versions for an application, in the application detail page, expand the Versions section.

If the application is one of your own applications, you can perform the following tasks:

- Create a New Version (p. 11)
- Edit an Existing Version (p. 11)

Amazon WAM Lite subscribers can assign only the most recent version of an application and can’t assign any previous versions of either an uploaded application or an application installed from the AWS Marketplace.
Create a New Version

You can create a new version of an application. Before you can create a new version of the application, you must have uploaded a new version of the application package as explained in Creating a New Application Version (p. 28).

To create a new version of an application

1. In the Versions section of the application details page, choose Create new version.
2. In the Select application package page, enter the following fields and choose Submit. The new application version is created.

   - **Package details**
     - **Package build**
       - Select the package build for the new version.
   - **Version information**
     - **Version title**
       - Enter a short name for the new version.
     - **Description**
       - Enter the description of the new version.

Edit an Existing Version

You can modify the title and description of an existing version.

To edit an existing version of an application

1. In the Versions section of the application details page, select the version to edit, and choose Edit version.
2. Modify the version title and description as required and choose Submit. It takes several minutes for the updated version information to be available.

Constraints

You can view and modify the constraints for an application from the application details page.

To view the constraints for an application

1. In the application detail page, expand the Constraints section. The application constraints are displayed.
2. Here, you can perform the following tasks:
   - To add a constraint, choose Add constraints.
   - To create a new version of the application, choose Create new version, and perform the procedure in Assigning an Amazon WAM Application to Users and Groups (p. 7).
   - To edit an existing version of the application, select the version to edit and choose Edit version. You can modify the version title and description.
Tags

Amazon WAM Standard customers can view, add, and remove tags from their applications.

To view the tags for an application
1. In the application detail page, expand the Tags section. The application tags are displayed.
2. Here, you can perform the following tasks:
   • To add a tag, enter the tag name and value, and choose Add tag.
   • To remove a tag, select the tag and choose Delete tag.

Users and Groups

You can view and modify the users and groups to which your applications are assigned. Amazon WAM Lite customers can't assign users and groups to their uploaded applications. Amazon WAM Standard customers can assign users and groups to their uploaded applications.

To view the users and groups to which an application is assigned
1. In the application detail page, expand the Users and Groups section. The users to which the application is assigned are displayed.
2. Here, you can perform the following tasks:
   • To assign the application to another user or group, choose Add user or group, and perform the procedure in Assigning an Amazon WAM Application to Users and Groups (p. 7).
   • To update the assignment options for the application, choose Edit assignment options. On the Configure assignment options (p. 8) page, you can update the assignment options.
   • To remove an application assignment, select the users and groups to remove the assignment from, choose Remove user or group, verify that you want to perform the action, and choose Continue. The application is no longer assigned to the selected users and groups.

Support Details

To view the support information for an application, in the application detail page, expand the Support details section. The support information for the application is displayed.

Monitoring Application Usage

Amazon WAM allows you to track your application usage by viewing usage reports in the Amazon WAM console.

Topics
• Applications (p. 12)
• Users (p. 13)

Applications

To view application usage information, perform the following steps.
To view application usage information

1. Open the WorkSpaces console at https://console.aws.amazon.com/workspaces/ and choose **Usage** in the navigation pane.
2. In the **Usage** page, choose the **Application** tab. All of your applications are displayed in the list.

If there are more search results than can be displayed in one page, you can scroll through the pages by choosing the left arrow (<) or right arrow (>) in the list header.

You can view additional information about a specific application by choosing the expander button next to the product.

**Users**

To view usage information by user, perform the following steps.

**To view usage information by user**

1. Open the WorkSpaces console at https://console.aws.amazon.com/workspaces/ and choose **Usage** in the navigation pane.
2. In the **Usage** page, choose the **Users** tab.
3. Under **Directory**, choose your directory. All of your users that have assigned applications are displayed in the list.

   **Note**
   Users are not displayed in the list until they have connected to their WorkSpace after the Amazon WorkSpaces Application Manager client application has been installed.

   If there are more search results than can be displayed in one page, you can scroll through the pages by choosing the left arrow (<) or right arrow (>) in the list header.

   You can view the applications that are assigned to a specific user by choosing the expander button next to the username.

**Removing All Application Assignments**

The Amazon WAM service automatically registers itself with an AWS Directory Service directory. This allows Amazon WAM to access the directory and prevents the directory from being deleted while it is in use by Amazon WAM. Before you can delete the directory that your WorkSpaces are assigned to, you must remove all Amazon WAM application assignments in the directory and deregister Amazon WAM from the directory. To make this task easier, the Amazon WAM console provides the ability to remove all application assignments and deregister itself from a directory in a single operation.

   **Note**
   Even if you do not have any application assignments, you must perform the following procedure to deregister Amazon WAM from a directory.

**To remove all application assignments and deregister Amazon WAM from a directory**

1. Open the WorkSpaces console at https://console.aws.amazon.com/workspaces/ and choose **Usage** in the navigation pane.
2. In the **Usage** page, choose the **Users** tab.
3. Under **Directory**, choose the directory. All of the directory users that have assigned applications are displayed in the list.
Note
Users are not displayed in the list until they have connected to their WorkSpace after the Amazon WorkSpaces Application Manager client application has been installed.

4. Choose **Remove all assignments**. The process to remove all application assignments and deregister the directory is initiated immediately. The entire process takes 10 to 15 minutes to complete. After all applications assignments have been removed, the service is deregistered from the directory, which is removed from the list of directories in the **Users** tab.

If you are unable to delete the directory after following this procedure, other applications or services might be registered to the directory. For more information, see [Delete a Directory](#) in the *Amazon WorkSpaces Administration Guide*.
Packaging and Validating Your Applications

You can deploy and manage applications for Amazon WorkSpaces using Amazon WorkSpaces Application Manager (Amazon WAM).

Contents
- Package an Application with Amazon WorkSpaces Application Manager Studio (p. 15)
- Packaging Specific Applications Using Amazon WAM (p. 20)
- Validating an Application Package with Amazon WAM (p. 26)
- Updating a Package (p. 28)
- Finding a Package with Amazon WAM (p. 49)
- Sharing a Package with Amazon WAM (p. 49)
- Deleting a Package with Amazon WAM (p. 50)
- Understanding Color Coding (p. 50)

Package an Application with Amazon WorkSpaces Application Manager Studio

Creating application packages requires an EC2 instance that is configured for packaging Amazon WAM applications.

You will be charged for the AWS resources that you use when packaging your application, such as the packaging instance, Amazon S3 storage, and bandwidth.

Prerequisites
- Create a VPC with a public subnet in a region that supports Amazon WAM.
- Create a security group for this VPC that allows access to the instance over port 3389 (RDP) from your computer. For more information about security groups, see Security groups for your VPC and Creating a security group in the Amazon VPC User Guide.
- Create a key pair to use to access your packaging instance.
- Create the AmazonWamAppPackaging IAM role. For more information, see Create the Application Packaging Role (p. 53).
- If you haven't done so already, select a subscription plan. For more information, see Select a Subscription Plan (p. 3).

Tasks
- Step 1: Launch a Packaging Instance (p. 16)
- Step 2: Create an Application Package (p. 16)
Step 1: Launch a Packaging Instance

Amazon WAM provides the Amazon WAM Studio to use when launching a packaging instance.

Before you launch a packaging instance, make sure you've met the prerequisites for packaging an application. For more information, see Package an Application (p. 15).

To launch a packaging instance

The following are the specific settings to use when launching a packaging instance. If you need additional information about launching EC2 instances, see Launching Your Instance from an AMI in the Amazon EC2 User Guide for Windows Instances.

**Note**

Amazon WAM doesn't support Instance Metadata Service Version 2 (IMDSv2) for use with packaging or validation instances.

1. Open the Amazon EC2 console at https://console.aws.amazon.com/ec2/.
2. In the navigation pane, choose AMIs.
3. Select the AWS Region where your VPC is located.
4. In the first filter, choose Private images.
5. Select the Amazon WAM Studio AMI and choose Launch.

   This AMI is shared with your account within several minutes after you select a subscription plan.

6. For the instance type, we recommend that you select an m5.xlarge instance type or larger for best performance. Amazon EC2 instance charges apply. The appropriate instance type for your situation depends on the compute requirements and size of your application. For most applications, we recommend that you select an xlarge size instance of the latest M series. For a list of the available instance types, see Amazon EC2 Instance Types.

7. On the Configure Instance Details page, do the following:
   - For Network, select your VPC.
   - For Subnet, select the public subnet in your VPC.
   - For Auto-assign Public IP, select Enable.
   - For IAM role, select your AmazonWamAppPackaging role. For more information, see Create the Application Packaging Role (p. 53).

8. (Optional) On the Add Tags page, choose click to add a Name tag and provide a name for this instance (for example, WAM packaging instance).

9. On the Configure Security Group page, select an existing security group that allows access to the instance over port 3389 (RDP) from your network.

10. Choose Launch.

11. After the launch succeeds, the packaging instance is ready for you to create a package. For more information, see Step 2: Create an Application Package (p. 16).

Step 2: Create an Application Package

After you launch a packaging instance, you'll connect to it and use it to create an application package.
To create an application package

1. Connect to the packaging instance. For more information, see Connecting to Your Windows Instance in the Amazon EC2 User Guide for Windows Instances.
2. Download the Windows installer for the application to be packaged.
3. Choose the Amazon WAM Studio shortcut on the desktop of your packaging instance.
4. In the Amazon WorkSpaces Application Manager Studio toolbar, choose New.
5. For 1. Application Package Information, do the following:
   a. For Package title, type the name of the application.
   b. For Description, type a description.
7. In the Capture Application Installation dialog box, in the Installer path field, choose Browse to navigate to the installer executable file. If the application doesn't have an installer, enter explorer.exe.

   Note
   If you encounter problems running the installer on Windows 10 but it worked with earlier versions of Windows, choose Compatibility Settings to select the appropriate compatibility mode for the installer.
8. For 3. Optional Settings, choose Settings. You can keep the default settings or make any changes that you need. For more information, see the following:
   - Adding Required Applications or Scripts (p. 42)
   - Sandboxing Your Package (p. 45)
   - Setting File Security (p. 47)
9. For 4. Package Build Information, do the following:
   a. For Package build label, type a label for the package. This is displayed in the Amazon WAM console as the version label.
   b. Choose Launch Settings and specify the command line and compatibility settings for the application. Note that this is the command used to launch the application, not the application installer. After you upload a package, you cannot modify the command line. If your package doesn't have a primary application, use explorer.exe.
   c. For Working folder, specify the working folder for the applications.
10. On the Amazon WorkSpaces Application Manager Studio toolbar, choose Save to save the application package project.
11. After saving the application package, capture the application installation. For more information, see Step 3: Capture the Application Installation (p. 17).

Step 3: Capture the Application Installation

When creating an application package, you install the application on the packaging instance. During the installation, the Amazon WorkSpaces Application Manager Studio application captures all changes to the file system, registry, environment, and other systems during the installation.

Before capturing the application installation, save the application project. This is required before you can capture the application installation because some applications require that the system be restarted to complete.

The installation process is different for applications that have an installer and applications that do not have an installer. Applications that do not have an installer are delivered as standalone executables or in an archive, such as a ZIP file.
Applications With an Installer

To capture the installation of an application that has an installer

1. Choose Install.
2. For Installer path, enter the path of the application installer, or choose Browse to navigate to it.
3. For Command line parameters, enter any command line parameters required for the installation.
4. For Compatibility Settings, select any Windows compatibility settings that are required by the installer, such as being run as administrator.

    Note
    If you encounter problems running the installer on Windows 10 but it worked with earlier versions of Windows, choose Compatibility Settings to select the appropriate compatibility mode for the installer.

5. Choose Launch and install the application as needed. All changes to the files system, registry, and environment are captured during the installation. For more information about what is captured and how to modify that information, see Captured Information (p. 19).

6. When the installation is complete, the Capture dialog box appears with the message “The Installer is finished.” Choose OK.

7. The Progress dialog box appears. After the capture process is finished, you might see some warnings in the Progress dialog box, but these are usually safe to ignore. Make sure to review the warnings to determine if you need to take action. When you’ve finished reviewing any warnings, choose Close.

8. After capturing the application installation, upload the application package. For more information, see Step 4: Upload the Application Package (p. 19).

Applications Without an Installer

To capture the installation of an application without an installer

1. Choose Install.
2. For Installer path, type explorer.exe.
3. Leave Command line parameters empty.
4. Select All processes.
6. Perform all steps needed to install the application, such as copying the standalone executable to the desired location, or extracting the files from a ZIP archive. All changes to the files system are captured during the installation. For more information about what is captured and how to modify that information, see Captured Information (p. 19).


8. The Capture dialog box appears with the message "The Installer is finished." Choose OK.

9. The Progress dialog box appears. After the capture process is finished, you might see some warnings in the Progress dialog box, but these are usually safe to ignore. Make sure to review the warnings to determine if you need to take action. When you’ve finished reviewing any warnings, choose Close.

10. After capturing the application installation, upload the application package. For more information, see Step 4: Upload the Application Package (p. 19).
Captured Information

The tabs at the bottom of the Amazon WorkSpaces Application Manager Studio contain all of the information that was captured during the installation of the application. Review each of these and modify them as needed for your package. Care must be taken when modifying these as making changes can make the application inoperable.

Installation Capture Information

Files

All changes to the file system are contained here. You can add, remove, or move files to customize your package.

Registry

All changes to the system registry are contained here. You can add, remove, or move registry entries to customize your package.

Environment

Any changes to the environment variables are contained here. You can add, remove, or modify these variables to customize your package.

Fonts

All changes to the system fonts are contained here. You can add or remove fonts to customize your package.

Startup

All changes to the system startup folders and registry keys are contained here. You can add, remove, or modify these to customize your package.

Services & Drivers

All services and drivers that were installed during the installation are contained here. You can add or remove these to customize your package.

Step 4: Upload the Application Package

After capturing the installation, you can upload the package.

To upload the application package

1. Choose **Upload** to upload the application package to Amazon WAM.
2. When the upload is complete, choose **Close**.
3. Close the application project.

**Note**

Do not modify a project after it has been uploaded. If you need to update a package, create a new version of the package. For more information, see Creating a New Application Version (p. 28).

The application package is now ready for validation. For more information, see Validating an Application Package with Amazon WAM (p. 26).
Packaging Specific Applications Using Amazon WAM

The following examples demonstrate how to package specific applications.

Topics
- Packaging 7-Zip Using Amazon WAM (p. 20)
- Packaging Google Chrome with Amazon WAM (p. 22)
- Packaging Mozilla Firefox Using Amazon WAM (p. 24)

Packaging 7-Zip Using Amazon WAM

Before you can create a package, you must first launch a packaging instance and a validating instance on Amazon EC2. For more information, see the following topics:
- Package an Application with Amazon WorkSpaces Application Manager Studio (p. 15)
- Packaging and Validating Your Applications (p. 15)

To create a package

To create a new package, use Windows Remote Desktop to connect to your packaging instance and then use Amazon WorkSpaces Application Manager Studio to create the package. For more information, see Connecting to Your Windows Instance Using RDP.

1. On the packaging instance, go to the 7-Zip download page and download the Windows 64-bit installer for 7-Zip.
2. Double-click the Amazon WAM Studio desktop shortcut to start the Amazon WorkSpaces Application Manager Studio.
3. On the Amazon WorkSpaces Application Manager Studio dashboard toolbar, choose New.
4. Under 1. Application settings, do the following:
   - For Title, enter the name of the application. This text appears in the Dashboard tab and in the Amazon WorkSpaces Application Manager Studio, to identify this application.
   - For Description, enter a short description for the application.
6. In the Capture Application Installation dialog box, do the following:
   - For Installer path, enter the path to the 7-Zip installer, or choose Browse to navigate to it.
   - For Capture file and registry activity for the following processes, select Installer process and sub-processes only.
   - Select Ignore changes under the Installer path specified above.
   - For After capturing is complete, do the following actions during the merge, select Replace short path names with long ones in the registry.
   - Choose Launch.

The installer for 7-Zip starts after you choose Launch. Follow the steps in the installer wizard to complete the installation.
- After 7-Zip is installed, the Capture dialog box appears with the message "The Installer is finished." Choose OK.
7. At the bottom of the Amazon WorkSpaces Application Manager Studio window, choose the Files tab, and then do the following:

- In the `<ProgramFilesX64>\7-Zip` directory, open the context (right-click) menu for `7-zip.dll` and choose Properties.
- In the File Properties dialog box, select Installed temporary (layer 2) and then choose OK.

8. To ensure that Windows Explorer does not hold a reference to `7-zip.dll` when the application is removed from the Workspace, create a configurable AppEvent that deregisters the file by using a batch file.

To create the batch file, do the following:

- On the packaging instance desktop, start Notepad.
- Enter the following and save the file to the desktop as `unreg_7zip.bat`.

```bash
@echo off
gsvr32 /s /u "c:\Program Files\7-Zip\7-zip.dll"
```

To create the configurable AppEvent, do the following:

- In Amazon WorkSpaces Application Manager Studio dashboard, under 3. Optional settings, choose Settings.
- In the Settings dialog box, choose Configurable AppEvents.
- In the Configurable AppEvent Setting dialog box, do the following:
  - Under Trigger, choose Before deactivation.
  - In File name under Handler command line, choose the `unreg_7zip.bat` file you created earlier.
  - Under Handler launching, select Run elevated (only supported on Windows Vista or later).
  - Under After launching, select Wait for the handler to exit. In time out after, type 5 and select minute(s).
  - Choose OK.
  - Choose OK.


10. On the Amazon WorkSpaces Application Manager Studio toolbar, choose Save to save the application package project.

11. Select the Project tab. In 4. Package Build, do the following:

   a. For Label, enter a label for the package. This label appears in the Amazon WAM console as the version label.
   b. For Command line, choose Launch Settings.
   c. In the Command line field of the Application Launch Settings dialog box, enter `C:\Program Files\7-Zip\7zFM.exe`.
   d. Select Set the working folder to the folder of the command line executable, if any, select Set the icon to the source file of the command line executable, if any, and choose OK.

12. In 5. Package upload, choose Upload. You might receive warnings about skipped files. These warnings are usually safe to ignore, but review the log to verify that the skipped files are unnecessary. When the upload is finished, choose Close.
13. Close the application project.

Note
Do not modify a project after it has been uploaded. If you need to update a package, create a new version of the package. For more information, see Creating a New Application Version (p. 28).

The application package is now ready for validation. For more information, see Validating an Application Package with Amazon WAM (p. 26).

Packaging Google Chrome with Amazon WAM

Before you can create a package, you must first launch a packaging instance and a validating instance on Amazon EC2. For more information, see the following topics:

- Package an Application with Amazon WorkSpaces Application Manager Studio (p. 15)
- Packaging and Validating Your Applications (p. 15)

To create a package for Google Chrome

To create a new package, use Windows Remote Desktop to connect to your packaging instance and then use Amazon WorkSpaces Application Manager Studio to create the package. For more information, see Connecting to Your Windows Instance Using RDP.

1. Double-click the Amazon WAM Studio desktop shortcut to start Amazon WorkSpaces Application Manager Studio.
2. In the Amazon WorkSpaces Application Manager Studio toolbar, choose New.
3. In 1. Application settings, do the following:
   - In the Title field, enter the name of the application. This text appears on the Dashboard tab and in the Amazon WorkSpaces Application Manager Studio to identify this application.
   - In the Description field, enter a short description for the application.
5. In the Capture Application Installation dialog box, do the following:
   - In the Installer path field, enter C:\Program Files (x86)\Internet Explorer \iexplore.exe.
   - In the Command line parameter field, enter "C:\Program Files\Google\Chrome \Application\chrome.exe".
   - For Capture file and registry activity for the following processes, select Installer process and sub-processes only.
   - Select Ignore changes under the Installer path specified above.
   - For After capturing is complete, do the following actions during the merge, select Replace short path names with long ones in the registry.
   - Choose Launch, go to Alternate (offline) Google Chrome installer (Windows), and follow the instructions to install Chrome offline on Windows for all accounts.
   - After installing Chrome, the Capture dialog box appears with the message "The Installer is finished." Choose OK.
   - The Progress dialog box appears. After the capture process is finished, you might see some warnings in the Progress dialog box, but these are usually safe to ignore. Make sure to review the warnings to determine if you need to take action. When you've finished reviewing any warnings, choose Close.
6. **Skip 3. Optional settings.**

7. At the bottom of the Amazon WorkSpaces Application Manager Studio window, choose the **Files** tab.
   
a. Find and expand the contents of `<ProgramFilesX64>`, then **Google**, and then **Chrome**. Under **Chrome**, there should be one subfolder named **Application**.
   
b. Open the context (right-click) menu for **Application**, and then choose **Delete**. When prompted, confirm the deletion.
   
c. Open the context (right-click) menu for **Chrome**.
      
      • Choose Add Subfolder.
      
      • In the Add Folder dialog box, choose Create from this existing folder.
      
      • Choose Browse, and navigate to `C:\Program Files\Google\Chrome\Application`.
      
      • Select Recursively include folders.
      
      • Select Include files within the folder(s).
      
      • Select Use existing folder path instead of current folder as addition target.
      
      • Choose OK.

8. To verify that your Chrome installation was successfully captured, expand the following folders. You should see files in all of these folders, except for `<ProgramFilesX64> > Google > Chrome > Application > 85.0.4183.102 > WidevineCdm > _platform_specific`.
   
   **Note**
   
   The version number for these folders differs depending on which version of Chrome that you've installed.

   • `<ProgramFilesX64> > Google > Chrome > Application > 85.0.4183.102`
   
   • `<ProgramFilesX64> > Google > Chrome > Application > 85.0.4183.102 > default_apps`
   
   • `<ProgramFilesX64> > Google > Chrome > Application > 85.0.4183.102 > Extensions`
   
   • `<ProgramFilesX64> > Google > Chrome > Application > 85.0.4183.102 > Installer`
   
   • `<ProgramFilesX64> > Google > Chrome > Application > 85.0.4183.102 > Locales`
   
   • `<ProgramFilesX64> > Google > Chrome > Application > 85.0.4183.102 > MEIPreload`
   
   • `<ProgramFilesX64> > Google > Chrome > Application > 85.0.4183.102 > swiftshader`
   
   • `<ProgramFilesX64> > Google > Chrome > Application > 85.0.4183.102 > VisualElements`
   
   • `<ProgramFilesX64> > Google > Chrome > Application > 85.0.4183.102 > WidevineCdm`
   
   • `<ProgramFilesX64> > Google > Chrome > Application > 85.0.4183.102 > WidevineCdm > _platform_specific`
   
   • `<ProgramFilesX64> > Google > Chrome > Application > 85.0.4183.102 > WidevineCdm > _platform_specific > win_x64`

9. Choose the **Registry** tab, and then create the following keys.

   **Important**
   
   The following registry keys disable automatic updates to Chrome. We recommend disabling automatic updates to applications installed through Amazon WAM because such updates might cause the application to stop working. Instead, you must manually update the version of the application. To create a version update, see **Updating an Application** (p. 10).

   • Open the context (right-click) menu for **HKEY_LOCAL_MACHINE\Software** and choose **Add Key**.
   
   • In the Add Registry Key dialog box, select Create from this existing registry key, and choose **Browse**.
   
   • In the Browse System Registry dialog box, choose **HKEY_LOCAL_MACHINE\Software\Policies** and choose **OK** twice.
   
   • Open the context (right-click) menu for **HKEY_LOCAL_MACHINE\Software\Policies**, and choose **Add Key**.
• In the Add Registry Key dialog box, select Create a new key with the following name, enter Google, and choose OK.
• Open the context (right-click) menu for HKEY_LOCAL_MACHINE\Software\Policies\Google, and choose Add Key.
• In the Add Registry Key dialog box, select Create a new key with the following name, enter Update, and choose OK.
• Open the context (right-click) menu for HKEY_LOCAL_MACHINE\Software\Policies\Google\Update, and choose Add Value.
• For Name in the Add Registry Value dialog box, enter AutoUpdateCheckPeriodMinutes. For Type, select REG_DWORD. For Data, enter 0, and choose OK.

10. Choose the Project tab, and then do the following in 4. Package build:

• For Label, enter a label for the package. This label appears in the Amazon WAM console as the version label.
• For Command line, choose Launch Settings.
• In the Command line field of the Application Launch Settings dialog box, choose Browse and navigate to C:\Program Files\Google\Chrome\Application\chrome.exe.
• Select Set the working folder to the folder of the command line executable, if any.
• Select Set the icon to the source file of the command line executable, if any.
• Choose OK.

11. Choose Save on the toolbar to save the application package project.
12. In 5. Package upload, choose Upload. You might receive warnings about skipped files. These warnings are usually safe to ignore, but review the log to verify that the skipped files are unnecessary. When the upload is finished, choose Close.
13. Close the application project.

Note
Do not modify a project after it has been uploaded. If you need to update a package, create a new version of the package. For more information, see Creating a New Application Version (p. 28).

The application package is now ready for validation. For more information, see Validating an Application Package with Amazon WAM (p. 26).

Packaging Mozilla Firefox Using Amazon WAM

Before you can create a package, you must first launch a packaging instance and a validating instance on Amazon EC2. For more information, see the following topics:

• Package an Application with Amazon WorkSpaces Application Manager Studio (p. 15)
• Packaging and Validating Your Applications (p. 15)

To create a package for Mozilla Firefox

To create a new package, use Windows Remote Desktop to connect to your packaging instance and then use Amazon WorkSpaces Application Manager Studio to create the package. For more information, see Connecting to Your Windows Instance Using RDP.

1. On your packaging instance, start Internet Explorer, and download the Windows 64-bit installer for Firefox Extended Support Release at Download Firefox Extended Support Release in your language.
2. Double-click the Amazon WAM Studio desktop shortcut to start Amazon WorkSpaces Application Manager Studio.
3. On the Amazon WorkSpaces Application Manager Studio Dashboard toolbar, choose **New**.

4. In **1. Application settings**, do the following:
   - For **Title**, enter the name of the application. This text appears on the **Dashboard** tab and in the Amazon WorkSpaces Application Manager Studio to identify this application.
   - For **Description**, enter a short description for the application.

5. In **2. Application installation**, select an operating system and choose **Install**. For Windows 10, choose **Windows 8 / 8.1 / 2012 / 2012 R2**.

6. In the **Capture Application Installation** dialog box, do the following:
   - For **Installer path**, enter the path to the Firefox installer, or choose **Browse** to navigate to it.
   - For **Command line parameter**, enter `-ms` to do a silent install.
   - For **Capture file and registry activity for the following processes**, select **Installer process and sub-processes only**.
   - Select **Ignore changes under the Installer path specified above**.
   - For **After capturing is complete, do the following actions during the merge**, select **Replace short path names with long ones in the registry**.
   - Choose **Launch**.

   The installer for Firefox starts after you choose **Launch**.
   - After Firefox is installed, the **Capture** dialog box appears with the message "The Installer is finished." Choose **OK**.
   - The **Progress** dialog box appears. After the capture process is finished, you might see some warnings in the **Progress** dialog box, but these are usually safe to ignore. Make sure to review the warnings to determine if you need to take action. When you've finished reviewing any warnings, choose **Close**.

7. On the desktop of the packaging instance, start Notepad, and then do the following to create two files: a configuration file that disables the auto-update feature and sets the homepage, and a script file that uses the configuration file.

   **Note**
   The first line in the configuration and script files must be two forward slashes (`//`).

   - Copy the following code into Notepad and save the file as `C:\Program Files\Mozilla Firefox\mozilla.cfg`.

     ```
     // Disable automatic updates
     pref("app.update.enabled", false);
     pref("app.update.auto", false);
     pref("app.update.mode", 0);
     pref("app.update.service.enabled", false);

     // Disable the default browser check
     pref("browser.shell.checkDefaultBrowser", false);
     pref("browser.startup.homepage_override.mstone", "ignore");

     // Set the homepage
     pref("browser.startup.homepage","https://www.amazon.com");
     ```

   **Important**
   - This configuration file disables automatic updates to Firefox. We recommend disabling automatic updates to applications installed through Amazon WAM because such updates might cause the application to stop working. Instead, you must manually update the version of the application. To create a version update, see **Updating an Application (p. 10)**.
• This configuration file shows how to set the home page to amazon.com. You might want to specify another location.

• To create a script file that uses the configuration file, copy the following code into Notepad, and save the file as C:\Program Files\Mozilla Firefox\defaults\pref\local-settings.js.

```javascript
//
pref("general.config.obscure_value", 0);
pref("general.config.filename", "mozilla.cfg");
```

8. At the bottom of the Amazon WorkSpaces Application Manager Studio window, choose the Files tab.

• Open the context (right-click) menu for <ProgramFilesX64>\Mozilla Firefox and then choose Add File. In Source file name on the Add File dialog box, enter C:\Program Files\Mozilla Firefox\mozilla.cfg, and then choose OK.

• Open the context (right-click) menu for <ProgramFilesX64>\Mozilla Firefox\defaults\pref and then choose Add File. In Source file name on the Add File dialog box, enter C:\Program Files\Mozilla Firefox\defaults\pref\local-settings.js, and then choose OK.

9. At the bottom of the Amazon WorkSpaces Application Manager Studio window, choose the Project tab, and then do the following in 4. Package build:

• For Label, enter a label for the package. This label appears in the Amazon WAM console as the version label.

• For Command line, choose Launch Settings. In the Application Launch Settings dialog box, enter C:\Program Files\Mozilla Firefox\firefox.exe.

• Select Set the working folder to the folder of the command line executable, if any.

• Select Set the icon to the source file of the command line executable, if any.

• Choose OK.

10. On the Amazon WorkSpaces Application Manager Studio toolbar, choose Save to save the application package project.

11. In 5. Package upload, choose Upload. You might receive warnings about skipped files. These warnings are usually safe to ignore, but review the log to verify that the skipped files are unnecessary. When the upload is finished, choose Close.

12. Close the application project.

**Note**

Do not modify a project after it has been uploaded. If you need to update a package, create a new version of the package. For more information, see Creating a New Application Version (p. 28).

The application package is now ready for validation. For more information, see Validating an Application Package with Amazon WAM (p. 26).

---

**Validating an Application Package with Amazon WAM**

An application package is validated on a special EC2 instance. The first time you access the Amazon WorkSpaces Application Manager console, an Amazon EC2 AMI is shared with you. Use this AMI to launch the validation instance.

**Tasks**

• Step 1: Launch the Validation Instance (p. 27)
Step 1: Launch the Validation Instance

We recommend that you launch the validation instance into its own VPC to make launching and configuration easier. You can, and should, use the same VPC that you used for the packaging instance. For more information, see Step 1: Launch a Packaging Instance (p. 16).

Before launching the validation instance, you must create the AmazonWamAppPackaging IAM role. For more information, see Create the Application Packaging Role (p. 53).

To launch a validation instance

The following are the specific settings to use when launching a validation instance. If you need additional information about launching EC2 instances, see Launching Your Instance from an AMI topic in the Amazon EC2 User Guide for Windows Instances.

Note
Amazon WAM doesn't support Instance Metadata Service Version 2 (IMDSv2) for use with packaging or validation instances.

1. Open the Amazon EC2 console at https://console.aws.amazon.com/ec2/.
2. Select the AWS Region where your VPC is located.
3. In the navigation pane, choose AMIs.
4. In the first filter, choose Private images.
5. Select the Amazon WAM Admin Player AMI and choose Launch.
6. For the instance type, we recommend that you select an m5.xlarge instance type or larger for best performance. Amazon EC2 instance charges apply. The appropriate instance type for your situation depends on the compute requirements and size of your application. For most applications, we recommend that you select an xlarge size instance of the latest M series. For a list of the available instance types, see Amazon EC2 Instance Types.
7. On the Configure Instance Details page, do the following:
   - For Network, select your VPC (the one that you used for WAM packaging).
   - For Subnet, select the public subnet in the VPC.
   - For Auto-assign Public IP, select Enable.
   - For IAM role, select the AmazonWamAppPackaging role, which you created in Create the Application Packaging Role (p. 53).
8. (Optional) On the Add Tags page, choose click to add a Name tag and provide a name for this instance (for example, WAM validation instance).
9. On the Configure Security Group page, select a security group that allows access to the instance over port 3389 (RDP) from your network. You can use the same security group that you created for your packaging instance.
10. Choose Launch.
11. You can use the same key pair that you created for your packaging instance.
12. After the launch succeeds, the validation instance is ready for you to validate a package. For more information, see Step 2: Validate the Application Package (p. 27).

Step 2: Validate the Application Package

After the validation instance is launched, use it to validate your pending application packages. AWS recommends that you launch an entirely new validation instance for each application package that is created.
To validate an application package

1. Using a Remote Desktop client, connect to the validation instance. For more information, see Connecting to Your Windows Instance in the Amazon EC2 User Guide for Windows Instances. The remaining instructions are performed on the validation instance.
2. Choose the Amazon WAM Admin Player shortcut on the desktop of your validation instance.
3. Select Pending Apps in the navigation bar. Any applications that need to be validated are displayed.
4. Select the application package to be validated. If the package is a new package, click Install. If the package is a new version of an existing package that is already installed on the instance, click Upgrade. The application is installed or upgraded on the validation instance.
5. Launch the application as your users will, and verify that it works correctly.

If the application package does not install or run as expected, you can update the package by creating a package update and validating the update. For more information, see Creating an Application Update (p. 29).
6. Click Approve to mark the package as testing completed.

Using the application package, you can now create a new application, or create a new version of an existing application. For more information about creating a new application, see Uploading an Application (p. 6). For more information about creating a new version of an existing application, see Create a New Version (p. 11).

Updating a Package

After you create an application package, you can modify the contents of the package according to the needs of your users. For example, you may want to include additional files that your users need, change registry entries to work in your environment, or set environment variables that are unique to your users.

To update the contents of your application package, you must open the package in update mode. After you update the package, you have to validate the package before your users can download the new version.

Tasks
- Creating a New Application Version (p. 28)
- Working with Files and Folders (p. 29)
- Working With the Registry (p. 33)
- Working with Environment Variables (p. 35)
- Working with Fonts (p. 36)
- Working with Startup Items (p. 38)
- Working with Services and Drivers (p. 40)
- Adding Required Applications or Scripts (p. 42)
- Sandboxing Your Package (p. 45)
- Setting File Security (p. 47)
- Using Licenses in Your Package (p. 48)

Creating a New Application Version

In addition to creating new application packages, you can also create new versions of existing packages. This is particularly useful for distributing patches and updates for your applications. AWS recommends that you launch an entirely new packaging instance for each application package version that is created.
After the packaging instance is launched, use it to create a new application package version. If an application patch is being created, you can reuse a packaging instance that only has the application being patched installed on it.

Creating an Application Update

An application update (or patch) is an update to an existing application that requires that the application already be installed. AWS recommends using this approach to apply security updates, application updates, and minor version upgrades.

To create an application update
1. Using a Remote Desktop client, connect to your packaging instance. The remaining instructions are performed on the packaging instance.
2. Download the update installer or new application version to the instance.
3. Launch the Amazon WorkSpaces Application Manager Studio by opening the desktop shortcut.
4. In the Amazon WorkSpaces Application Manager Studio Dashboard, select the application to create an update for.
5. Choose Update on the Amazon WorkSpaces Application Manager Studio toolbar. If the application is already installed in the instance, you will see a message box that the files on the instance will be overwritten by the files in the package. Choose Yes. The message box displays a list of files that were overwritten.

   **Important**
   If you double-click the package or choose Open, the package opens in read-only mode. You cannot update the package in this mode.

6. Capture the upgrade installation as described in Step 3: Capture the Application Installation (p. 17).
7. If you need to modify any of the optional settings, choose Settings and modify any packager settings that are required. In most cases, the default settings are used.
8. In the Package build label field of the Package Build Information section, enter the label for the new version. In the Version notes (optional) field, enter a description for the new version.

   **Note**
   You cannot change the command line for a package that has been uploaded. If you need to modify the command line, you need to create an entirely new package.

9. Upload the new package version as described in Step 4: Upload the Application Package (p. 19).

The new version of the application package is now ready for validation. The new version cannot be assigned to users until the validation is completed. For more information, see Validating an Application Package with Amazon WAM (p. 26).

Working with Files and Folders

You can change the files and folders in your application package to suit the needs of your users. For example, you may want all of your users to have a folder that contains specific files. Or you may want to reduce the size of your package by removing unneeded folders and files.

You can change the folders and files in your package by choosing the Files tab in the project dashboard.
Your package contains two kinds of folders:

Template folders map to the actual system folder paths in the package instance. Template folders are in blue and cannot be removed or added to the package.

Root folders map to the same path shown in the navigation pane. These folders are in yellow and were added during package creation or manually. You can add root folders to any part of the folder tree.

**To add a root folder or subfolder**

1. In the Files tab, open the context (right-click) menu in the left navigation bar and choose Add Root Folder. To add a subfolder, open the context (right-click) menu for a folder in the left navigation bar and choose Add Subfolder.

2. In the Add Folder dialog box, do one of the following:

   - To create a new root folder, choose Create new root folder with the following name and enter a path for it.
   - To create a new root folder from an existing folder in the package instance, choose Create from existing source and then choose Browse to select the folder to add to the package. However, the files and subfolders in the root folder are not added to the package. To add all of the subfolders in the root folder, choose Recursively include subdirectories. To add the files in the folder and subfolders, choose Include files within the folder(s). To use a different name for the folder,
choose **Override existing folder name with the following name** and type the new name for the folder.

3. Choose **OK**.

**To add a file to a folder**

1. If the file is not in the package instance, add the file to the instance.
2. In the **Files** tab, open the context (right-click) menu for the folder where you want to add the file and choose **Add File**.
3. In the **Add File** dialog box, choose **Browse** and then choose the file to add to the package.
4. To change the name of the file, choose **Override source file name with the following name** and enter the new name of the file.
5. Choose **OK**.

Folders and files added after the package was created appear in magenta.

**To rename a root folder or file**

1. In the **Files** tab, open the context (right-click) menu for the root folder or file that you want to rename and choose **Rename**.
2. Type the new name of the root folder or file and press **Enter**.

**To edit the properties of a root folder or file**

1. In the **Files** tab, open the context (right-click) menu for the root folder or file that you want to rename and choose **Properties**.
2. In the dialog box, change the properties of the folder and choose **OK**.

**To set a folder as the working folder**

The working folder contains the application executable or some related files. Some applications only work if the working folder is set to a folder that does not contain the application executable file. For these types of applications, you have to set the working folder to another folder.
1. In the **Files** tab, open the context (right-click) menu for the root folder that you want to be the working folder and choose **Set As Working Folder**.

2. In the dialog box, change the properties of the folder and choose **OK**.

**To remove a folder or file from the package**

- In the **Files** tab, open the context (right-click) menu for the folder or file that you want to remove and then choose **Delete**.

**To exclude a folder or file from the package**

Folders, subfolders, and files that are excluded in the package are not installed in WorkSpaces, but are still part of the package. This gives you the flexibility of including certain folders and files for one package deployment and not for another. Excluding folders and files also enables you to test an package without a folder or file before permanently deleting it from the project.

- In the **Files** tab, open the context (right-click) menu for the root folder or file that you want to exclude from the package and then choose **Exclude**.

Excluded folders and files appear in gray italicized text.

**To change an unmerged folder into a merged folder**

If your package has a root folder that is already on the Amazon WAM client application, both package and client application folders are merged into a single folder. If you are assigning several packages that use the same folder, the contents of each folder are added to a single merged folder.

Merged folders have a plus sign (+).
• In the Files tab, open the context (right-click) menu for the unmerged folder, and then choose Merged Folder.

To show a folder or file in Explorer

• In the Files tab, open the context (right-click) menu for the folder or file that you want to show in Explorer and then choose Show in Explorer.

Working With the Registry

You can modify the registry in your package by choosing the Registry tab in the project dashboard. You can create a new subkey or add an existing subkey under any key, including root keys (or hives). However, you cannot create new root keys.
To add a subkey

1. In the Registry tab, open the context (right-click) menu for the root or subkey where you want to add a subkey and then choose Add Key.

2. In the Add Registry Key dialog box, create a new key by entering a new name or using an existing key to create a new key.

   If you create a new key from an existing key, you can choose one or more of the following options:

   - **Recursively include subkeys**
     
     Add all subkeys under the existing key. Clear this choice if you only want to add the selected key.

   - **Include values within the key(s)**
     
     Add the values of the keys. Clear this choice to add only the key.

   - **Use existing registry key instead of current key as addition target**
     
     Adds the complete path from the top level of the new key to save you from having to manually re-create the entire path, starting with the top level.

3. Choose OK.

To rename a registry key

1. In the Registry tab, open the context (right-click) menu for the root or subkey for which you want to change the name, and then choose Rename.

2. Change the name of the key and press Enter.

To change the disposition of a registry key

1. In the Registry tab, open the context (right-click) menu for the root or subkey for which you want to change the name, and then choose Disposition. Choose Apply setting to subkeys recursively to apply the same disposition to the subkeys.

2. Choose OK.

To add a value to the key

1. In the Registry tab, open the context (right-click) menu for the root or subkey for which you want to change the value, and then choose Add Value.

2. In the Add Registry Value dialog box, do the following:
   
   - For Name, enter the name of the registry value.
   - For Type, select the data type of the value.
   - For Data, enter the registry value. Choose Display as hexadecimal to display the hexadecimal value.

   **Note**

   If the registry value appears to contain an environment variable, such as %ServerName%, Amazon Studio attempts to resolve the environment variable. If the environment variable does not exist, then Amazon Studio escapes the % characters. %ServerName% becomes %ServerName%%. To use an environment variable as the value, define the environment variable in the Environment tab.

3. Choose OK.
Working with Environment Variables

You can set the environment variables used by your package by choosing the Environment tab in the project dashboard. You can see the environment variables in the package, add new environment variables, or change existing environment variables.

The Environment tab shows two types of environment variables: user environment variables (specific for each user and set in the registry under HKEY_CURRENT_USER \ Environment) and system environment variables (global for everyone and set in the registry under HKEY_LOCAL_MACHINE \ SYSTEM\CurrentControlSet\Control\Session Manager\Environment). These environment variables are set in the application or on the operating system depending on the Environment Variables setting in Virtualization in the Settings dialog box.
• Environment variables shared on the entire operating system are added to the HKEY_CURRENT_USER \Environment registry key when you create the package. These environment variables are available to all applications for that operating system.

• Environment variables visible only to the application are set in the process environment when the application starts in the operating system. Child processes of the application may inherit these environment variables but other processes on the system cannot access these environment variables.

All virtual environment variables override any existing variables of the same name for either method. The only exception to this rule is for the %PATH% or %PATHEXT% variables. These variable are automatically appended to existing system values. For example:

PATH = %PATH%;c:\newfolder

Note
Some legacy applications may check the registry for environment variables under a specific hive and not use the variables in the process environment. For such applications, add the environment variables directly using the Registry tab in the correct locations and set the key disposition to Virtual-isolated (layer 4).

To add an environment variable

1. In the Environment tab, open the context (right-click) menu in the pane, and choose Add Environment Variable.

2. In the Add Environment Variable dialog box, enter the name and value of the environment variable, and choose OK.

To edit an environment variable

1. In the Environment tab, open the context (right-click) menu for the environment variable that you want to edit, and choose Edit.

2. In the Edit Environment Variable dialog box, edit the name and value of the environment variable, and choose OK.

To delete an environment variable

• In the Environment tab, open the context (right-click) menu for the environment variable that you want to delete, choose Delete, and choose OK.

Working with Fonts

You can configure the fonts used in your package by choosing the Fonts tab in the project dashboard.
The fonts in a package are either available only to the applications in the package or made available to other applications on the operating system, depending on the Font setting under the Virtualization in the Settings dialog box.

**Note**
To add more fonts to a project, you need to know the file name or names for the font, and not just the font name. In many cases, there is more than one file for each font. For example, Arial consists of four files, one for the basic font, and others for italic, bold, and bold italic.

**To import a system font**

1. In the Fonts tab, open the context (right-click) menu in the pane and choose Import Font.
2. In the Import Font dialog box, select a font, and choose OK.

   Repeat this step to import other system fonts.

**To add a font from a font file**

1. Copy the font file to the package instance.
2. In the Fonts tab, open the context (right-click) menu in the pane, and choose Import Font.
3. In the Add Font dialog box, enter a name for the font, and choose Browse to find the font file.
4. Choose OK

   **Note**
   If the virtualization setting for fonts is set to register, but the disposition for the corresponding font file is set to Virtual-isolated (layer 4), the disposition for the font appears in red. This font is not visible to other applications on the operating system, despite the virtualization setting.

**To edit a font**

1. In the Fonts tab, open the context (right-click) menu for the font that you want to edit, and choose Edit.
2. In the Edit Font dialog box, change the values, and choose OK.

**To change the properties of a font**

You can edit the prefetch, disposition, and security settings from Fonts tab, instead of finding and editing the font file in Files tab.

1. In the Fonts tab, open the context (right-click) menu for the font that you want to edit and choose Font File Properties.
2. In the **Font File Properties** dialog box, change the property values, and choose **OK**.

**To delete a font**

- In the **Fonts** tab, open the context (right-click) menu for the font that you want to delete, choose **Delete**, and choose **OK**.

**To exclude a font**

Fonts that are excluded are not used in WorkSpaces, but are still part of the project. This gives you the flexibility of including specific fonts in one package and not in another package. Excluding fonts variables also enables you to test an package without font before removing the font files from the package.

1. In the **Fonts** tab, open the context (right-click) menu for the font that you want to exclude, and then choose **Exclude**.
2. Choose **OK**.

Excluded fonts appear in italicized text.

---

**Working with Startup Items**

Startup items are programs that typically launch at the operating system startup, which are installed along with the application. These often include the system tray icons that permit direct access to the functions of the application, even when the application is not running.

You can configure the fonts used in your package by choosing the **Startup** tab in the project dashboard.

The operating system launches startup items as either as a special registry key called run keys or as a Start item. You can set your launch preference in **Virtualization** of the **Settings** dialog box.

The path to the executable in a run key is displayed in the **Startup** tab.

**HKCU Run** (**HKCU** is short for **HKEY_CURRENT_USER**)  
Applies to the current user.

**HKCU RunOnce**  
Applies to the current user but only runs once.
HKLM Run (HKLM is short for HKEY_LOCAL_MACHINE)

Applies to all WorkSpaces.

To add a startup item, you need following information:

- If you want the startup item to launch as a registry key or as a Start menu item
- Name and path of the registry key (if necessary)
- Path and name of the executable file

To add a registry key startup item

1. In the Startup tab, open the context (right-click) menu for either the HKCU or HKLM folder in the left navigation pane and choose Add Startup Value.
2. In the Add Startup Value dialog box, enter the name and path to the executable file of the startup item, and choose OK.

To add a Start menu startup item

1. In the Startup tab, open the context (right-click) menu for either the <UserStartup> or <CommonStartup> folder in the left navigation pane and choose Add Startup File.
2. In the Add Startup File dialog box, enter the name and path to the executable file of the startup item. To use a different name for the executable file, choose Override source file name with the following name and enter the new name.
3. Choose OK.

To edit a startup item

1. In the Startup tab, open the context (right-click) menu for the font that you want to edit, and choose Edit.
2. In the dialog box, change the values and choose OK.

Note
If the disposition of the registry key for a startup item is Installed-permanent (layer 1) or Installed-temporary (layer 2), but the corresponding executable file disposition is set to Virtual-integrated (layer 3) or Virtual-isolated (layer 4), the item is flagged with the disposition in red, because the registry key is physically present on the client, while the file to which it points is not.

To change the properties of a startup item

You can edit the prefetch, disposition, and security settings from Startup tab, instead of finding and editing the font file in Files tab.

1. In the Startup tab, open the context (right-click) menu for the font that you want to edit, and choose Edit Command Line File Properties.
2. In the Command Line File Properties dialog box, change the property values, and choose OK.

Note
If a startup item points to a shortcut (.lnk file) which has a Installed-permanent (layer 1) or Installed-temporary (layer 2) disposition, but the target executable file is set to Virtual-integrated (layer 3) or Virtual-isolated (layer 4), the item is flagged with the disposition in red,
because the shortcut is physically present in the operating system, while the file to which it points is not.

To delete a startup item

- In the **Startup** tab, open the context (right-click) menu for the font that you want to delete, choose **Delete**, and choose **OK**.

To exclude a startup item

Startup items that are excluded are not used in WorkSpaces, but are still part of the package. This gives you the flexibility of including specific startup items in one package deployment and not in another package. Excluding startup items also enables you to test a package without a startup item before removing the startup from the package.

1. In the **Startup** tab, open the context (right-click) menu for the font that you want to exclude, and then choose **Exclude**.
2. Choose **OK**.

Excluded fonts appear in italicized text.

**Working with Services and Drivers**

You can view the services and drivers installed when the package was created or add new services and drivers by choosing the **Services & Drivers** tab in the project dashboard.

Driver and services are shared components within the operating system, which means that while the files and registry keys can be virtual, the running process affect the entire operating system. Certain types of services or drivers must be ran at system startup and these, as well as any dependencies, may need to be made to Installed-temporary (layer 2). When you remove a service or driver, you may have to use a configurable AppEvent trigger to perform a system reboot.

To add a service or driver

1. In the **Services & Drivers** tab, open the context (right-click) menu in the pane, and choose **Import Service or Driver**.
2. In the **Import Service or Driver** dialog box, select a service or driver from the list and choose **OK**.
3. Choose **Yes** to add the image for the service or driver to your package.
To change the disposition of the registry key for a service or driver

1. In the Services & Drivers tab, open the context (right-click) menu of the service or driver for which you want to change the disposition, and choose Service Key Disposition.
2. In the Service Key Disposition dialog box, select the disposition for the registry key, and choose OK. If the message box appears, choose Yes to change the ancestor keys to Virtual-integrated (layer 3).

To edit the image file properties for a service or driver

If the you added the image file for service or driver through the Files tab, you can edit the prefetch, disposition, and security settings from within the Services tab, rather than having to locate and edit the file from the Files tab.

Note
Any service or driver in a package with Virtual-integrated (layer 3) disposition are guaranteed to have all of their dependencies (e.g. .dll, .ini, .manifest, .cfg, etc. files) available to the service or driver. If you set the service or driver to Installed-permanent (layer 1) or Installed-temporary (layer 2), set the dependencies to the Installed-permanent (layer 1) or Installed-temporary (layer 2) disposition. This is important if the services are meant to automatically start at Windows logon. You can use Dependency Walker as a tool to identify DLL files that an executable needs to run successfully.

1. In the Services & Drivers tab, open the context (right-click) menu for the service or driver for which you want to change the image file properties and choose Edit Image File Properties.
2. In the Image File Properties dialog box, select your settings, and choose OK.

To edit the service or driver properties

1. In the Services & Drivers tab, open the context (right-click) menu for the service or driver for which you want to change the image file properties, and choose Edit.
2. In the dialog box, change the settings and choose OK.

To change the failure action of a service or driver

- In the Services & Drivers tab, open the context (right-click) menu for the service or driver for which you want to change the failure action, and choose Turn Failure Actions OFF if restart is enabled or Turn Failure Actions ON if restart is already disabled. The option is grayed out if the failure action cannot be changed for the selected service or driver. If the failure actions are enabled for a service or driver, the name of that service is red.

To delete a service or driver

- In the Services & Drivers tab, open the context (right-click) menu for the service or driver that you want to delete, choose Delete, and choose OK.

To exclude a service or driver

Services or drivers that are excluded are not used in operating system, but are still part of the package. This gives you the flexibility of including specific services or drivers in one package deployment and not in another package. Excluding services or drivers also enables you to test an package without a service or driver before removing the service or driver from the package.

1. In the Services & Drivers tab, open the context (right-click) menu for the service or driver that you want to exclude, and then choose Exclude.
2. Choose **OK**.

Excluded services or drivers appear in italicized text.

## Adding Required Applications or Scripts

Some applications may require additional applications or scripts to run in order for the packaged application to run properly. These additional applications or scripts may need to be launched before, during, or after the packaged application launches. For example, you may need these additional applications or scripts to do the following:

- Configure plug-ins for office suites and browser applications
- Configure virtual printers
- Start a background application before the packaged application

To launch these additional applications or scripts, you can use a configurable AppEvent trigger. A configurable AppEvent trigger is an action you set in Amazon WorkSpaces Application Manager Studio that runs a handler to launch these additional applications or scripts. The handler can be an application written in any programming language, such as C++, or a script in a scripting language, such as VBScript.

Applications used in a configurable AppEvent trigger can be any size while scripts have to be less than 10 Mb in size. Applications or scripts added to the package cannot be directly modified. If the application or script is modified in the package, Amazon Studio or Amazon Admin Player disables the application or script.

### Sample Applications and Scripts

The packaging instance contains the following sample applications and scripts in **C:\Program Files \Amazon\WAM Admin Studio\Samples**.

<table>
<thead>
<tr>
<th>Filename</th>
<th>Description</th>
<th>Error codes</th>
</tr>
</thead>
<tbody>
<tr>
<td>AddService.exe</td>
<td>Adds and starts a service</td>
<td>-1</td>
</tr>
<tr>
<td>DotNetCheck.vbs</td>
<td>Checks for the Microsoft .NET Framework. Specify the major and minor version number to check in <strong>Parameter</strong>.</td>
<td>1</td>
</tr>
<tr>
<td>DirectXCheck.vbs</td>
<td>Checks for a specific version of Microsoft DirectX. Specify the major and minor version number to check in <strong>Parameter</strong>.</td>
<td>1</td>
</tr>
<tr>
<td>OpenFirewall.vbs</td>
<td>Opens the firewall for the specified program or ports.</td>
<td>-1</td>
</tr>
<tr>
<td>OSMemCheck.vbs</td>
<td>Check for sufficient memory.</td>
<td>-1</td>
</tr>
<tr>
<td>QuicktimeCheck.vbs</td>
<td>Checks for specific version of Apple QuickTime. Specify the major and minor version number to check in <strong>Parameter</strong>.</td>
<td>1</td>
</tr>
<tr>
<td>ReaderCheck.vbs</td>
<td>Checks for a specific version of Adobe Reader. Specify the major</td>
<td>1</td>
</tr>
<tr>
<td>Filename</td>
<td>Description</td>
<td>Error codes</td>
</tr>
<tr>
<td>---------------------</td>
<td>-------------------------------------------</td>
<td>-------------</td>
</tr>
<tr>
<td>RemoveService.exe</td>
<td>Stops and removes a service</td>
<td>-1</td>
</tr>
</tbody>
</table>

**Handler Errors**

If the handler returns a failure code, Amazon Studio logs the failure code but does not display any messages for error codes. The handler must display a message for error codes. If the failure occurs during the activation trigger, the application is not added to the Amazon Admin Player. If the failure occurs during the exit trigger, the handler continues.

**Note**

During a Windows logoff or shutdown, triggers are skipped because Windows does not allow a new process to be created during this time.

**Adding a Configurable AppEvent Trigger**

You can add a configurable AppEvent trigger through Amazon Studio.

**To add a configurable AppEvent**

1. In Amazon Studio, choose **Update** to get the files in your package.
2. Under **3. Optional settings**, choose **Settings**.
3. In the navigation pane, choose **Configurable AppEvents** and then choose **Add**.
4. In the ** Configurable AppEvent Settings** dialog box, do the following:
   - For **Trigger**, select when to launch the handler.
<table>
<thead>
<tr>
<th>Trigger</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>After registration</td>
<td>Occurs one time when the application is first added to the Amazon WorkSpaces Application Manager client application.</td>
</tr>
<tr>
<td>After activation</td>
<td>Occurs one time immediately after the application package has been activated (installation is triggered).</td>
</tr>
<tr>
<td>After virtualization</td>
<td>Occurs immediately after virtualization has completed (installation is complete).</td>
</tr>
<tr>
<td>After launch</td>
<td>Occurs immediately after the application launches.</td>
</tr>
<tr>
<td>After exit</td>
<td>Occurs after the application exits.</td>
</tr>
<tr>
<td>Before deactivation</td>
<td>Occurs one time immediately before an application package deactivates and the application is removed from the Amazon WorkSpaces Application Manager client application.</td>
</tr>
<tr>
<td>Before devirtualization</td>
<td>Occurs before devirtualization starts and the application is removed from the WorkSpaces.</td>
</tr>
</tbody>
</table>

- For **Frequency**, choose when to launch the trigger.

<table>
<thead>
<tr>
<th>Trigger</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fire Once</td>
<td>The action only launches one time.</td>
</tr>
<tr>
<td>Fire Always</td>
<td>The action launches when the trigger occurs. If the trigger occurs only one time, then this action launches only one time.</td>
</tr>
</tbody>
</table>

- For **Handler command line**, enter the path and required parameters for the handler. If the application or script is on WorkSpaces, choose **Use local application**.

  **Warning**
  When using the **Extra** parameter, be sure to not include the `<`, `>`, `,`, `&`, `^`, `#`, `%`, `{`, `}`, `
  ~`, `\`, `'`, or space characters unless they are encoded by the portal and decoded by the handler. All characters appear as part of the value in the extra environment variable and do not need to be escaped.

- For **Handler launching**, choose one of the following as required:

  **Note**
  This setting can affect critical files and should be used cautiously.

  - **Run elevated** (only supported on Windows Vista or later)
    - If the handler requires administrator privileges.
  - **Run as LocalSystem** (only if allowed by group policy)
    - If the handler should be launched as a system process. The application or script in the handler has same access to files as the user.
• For After launching, choose **Wait for handler** to exit to require the handler to finish before
the configurable AppEvent trigger in the Amazon WorkSpaces Application Manager client
application goes to the next state. If this option is cleared, then the handler launches and the
trigger continues with the normal operation in parallel. Set the **Time out after** in minutes or
seconds for the trigger in the client to wait for the handler to exit. In **Success codes**, set a specific
return value to indicate success or failure for the trigger. By default, 0 is a success code and all
other values are failures. You can also set a different success code.

• Choose **OK** to add the configurable AppEvent trigger.

### Sandboxing Your Package

Sandboxing means to separate changes made to the file system or registry from the Amazon
WorkSpaces Application Manager client application. Amazon WAM Admin Player performs partial
sandboxing for files, which means that only content modified or added to folders or subfolders of root
folders in the package are sandboxed. For example, if a package contains only the installation root folder
and the **Program Files** folder, and if the application creates a new file under the **Windows** folder, then
this new file is directly written to the system. Amazon Admin Player sandboxes all default registry hives
(that is, **HKCR**, **HKCU**, **HKLM**, **HKU**), but any other hive must be part of the package to be sandboxed.

For the paths in the package that are sandboxed, the content is separated from the system, meaning that
they are not written to the local system but stored separately and only made available to the application.
Again using the example above, if the package creates a new file under the **Program Files** folder, then
the file is sandboxed and not written to the system. Only existing files under the **Program Files** folder
that are not contained in the package would be modified by the application if attempted. The same rule
applies for existing registry keys not contained within the package.

As a rule, do not sandbox any documents or data saved by the user. The following table lists the folders
are the only exception for sandboxed folders.

#### Sandbox exclusion folders

<table>
<thead>
<tr>
<th>Sandbox exclusion folder</th>
<th>Path</th>
</tr>
</thead>
<tbody>
<tr>
<td>System Root</td>
<td><code>C:\</code></td>
</tr>
<tr>
<td>Desktop</td>
<td><code>C:\Users\&lt;username&gt;\Desktop</code></td>
</tr>
<tr>
<td>Documents</td>
<td><code>C:\Users\&lt;username&gt;\Documents</code></td>
</tr>
<tr>
<td>Downloads</td>
<td><code>C:\Users\&lt;username&gt;\Downloads</code></td>
</tr>
<tr>
<td>Music</td>
<td><code>C:\Users\&lt;username&gt;\Music</code></td>
</tr>
<tr>
<td>Pictures</td>
<td><code>C:\Users\&lt;username&gt;\Pictures</code></td>
</tr>
<tr>
<td>Videos</td>
<td><code>C:\Users\&lt;username&gt;\Videos</code></td>
</tr>
<tr>
<td>Links</td>
<td><code>C:\Users\&lt;username&gt;\Links</code></td>
</tr>
<tr>
<td>Favorites</td>
<td><code>C:\Users\&lt;username&gt;\Favorites</code></td>
</tr>
<tr>
<td>Contacts</td>
<td><code>C:\Users\&lt;username&gt;\Contacts</code></td>
</tr>
<tr>
<td>Saved Games</td>
<td><code>C:\Users\&lt;username&gt;\Saved Games</code></td>
</tr>
<tr>
<td>Searches</td>
<td><code>C:\Users\&lt;username&gt;\Searches</code></td>
</tr>
<tr>
<td>Temp</td>
<td><code>C:\Users\&lt;username&gt;\AppData\Local\temp</code></td>
</tr>
</tbody>
</table>
Sandboxing Your Package

If your application writes to these folders, the data is directly written to the system, even if the Users folder is part of the application as a root folder, as would be expected by users.

To configure sandboxing in your package

1. In Amazon Studio, choose Update to get the files in your package.
3. In the navigation pane, choose Sandboxing.
4. To exclude a folder, choose Folder Exclusions. In the Sandbox Folder Exclusions dialog box, choose Add and then type the path to the folder that you want to exclude. Choose OK.
   
   Tip
   You can drag-and-drop the folder to exclude to the Folder Exclusions dialog box from Explorer.
5. To exclude a registry key, choose Registry Key Exclusions. In the Sandbox Registry Key Exclusions dialog box, choose Add and enter the registry key to exclude from the package.
6. For Disposition for new registry key roots, choose Virtual-integrated (layer 3) for resources that the application and local system can see but are not physically installed on the local system, or Virtual-isolated (layer 4) for resources that only the application can see and are not physically installed on the local system.
7. For Application settings, choose Preserve application settings when application is removed to save the application-specific settings modified by the user when the application assignment is removed from the user, or Allow Windows to road application settings to write the application-specific settings to the standard Windows roaming profiles location.
8. Choose OK.
Setting File Security

The standard security settings for the files in your application may need to be bypassed by specific application processes. For example, an application patch (such as Windows Updates) may be required to modify a file with a read-only attribute. In this case, a security override can be used to allow this to happen while still protecting the application. Another example would be to not allow anti-virus or indexing software from fetching all the application pages. This undesirable side effect can also be prevented by denying those specific processes from accessing the virtual application.

**Note**

Using a different security setting only affect the file security settings but do not change the visibility of the disposition layer. If the file disposition is set to **Virtual-isolated (layer 4)**, then setting a system process to override all the security settings still does not allow that process to see this file.

To change the security of a file

1. In Amazon Studio, choose **Update** to get the files in your package.
2. Under **Optional settings**, choose **Settings**.
3. In the navigation pane, choose **Security** and **Add**.
4. In the **Security Override Process** dialog box, do the following:
   - For **Process name**, type the exact name of the process for the security override controls. Do not use wildcards.
   - For **Recognition method**, choose one of the following methods to use to determine which file version to override the process. You may have several versions of the same file in the system. For any option that requires a file path, enter the path and file name in the **Process file** field.

   **Process name**
   
   Use only the process name. This is the weakest verification method.

   **Process path**
   
   Use the path and filename of the process.

   **MD5 Hash**
   
   Use an MD5 algorithm to compute and store a fingerprint of the executable file to verify that the process file is authentic. This method is the most secure verification method.

   **CRC Checksum**
   
   Use a CRC algorithm to compute and store a checksum value to verify that the process file is authentic.
• For **Choose how virtual assets should be accessed (disposition override)**, choose one of the following:

  Allow or deny access based on the dispositions specified for the virtual assets

  Use the individual disposition setting for files, folders, registry keys, and values.

  **Always allow access**

  Treat all files, folders, registry keys, and values as having Virtual-integrated (layer 3) disposition, even if they have Virtual-isolated (layer 4) disposition.

  **Always deny access**

  Treat all files, folders, registry keys, and values as having Virtual-isolated (layer 4) disposition, even if they have Virtual-integrated (layer 3) disposition.

• For **Security settings override**, choose **Allow** or **Deny** for the following options and choose **Apply**.

  **Note**

  **Deny** takes precedence over **Allow**. Choosing both settings is the same as choosing **Deny**.

  **Read or copy content of files**

  Read or copy the application files.

  **Write or modify content of files**

  Write or modify the contents of the application files.

  **Make files visible through folder listings**

  Show the files in the folder listing.

**Using Licenses in Your Package**

How you add an application license depends on the type of license.

**Options**

- Using a Single License on All Instances (p. 48)
- Using an ISV License Service (p. 48)
- Using a License per Instance (p. 49)

**Using a Single License on All Instances**

For an application that uses a single master license for all users, enter the license at the appropriate screen when you package the application. The license is captured in the package and used in all instances.

**Using an ISV License Service**

For an application using an ISV license service hosted at a website, enter the service URL at the appropriate screen when you package the application. The information provided by the service to activate the application is captured in the package and used in all instances.

You can also automatically connect to the license service by adding an environment variable with the license service location to each WorkSpaces client and then use a Configurable AppEvent to run a script to read the environment variable to get to the license service to activate the application.
To use a Configurable AppEvent to connect to an ISV license service

This procedure assumes that your WorkSpaces clients have an environment variable with license service location and you have a script that use the license service to activate the application.

Note
You cannot add a Configurable AppEvent to a new version of a package.

1. Copy your script to the package instance.
2. Under 3. Optional settings in Amazon WorkSpaces Application Manager Studio, choose Settings.
3. In the Settings dialog box, choose Configurable AppEvents and Add.
4. In the Configurable AppEvent Settings dialog box, do the following:
   - For Trigger, choose After launch.
   - For Frequency, choose Fire the first time only.
   - For Handler, in the File name field, enter the name of the script to activate the application.
   - Choose OK twice to close the dialog boxes.

Using a License per Instance

For an application using one license per instance, enter the license information at the appropriate screen when the application installs on the computer.

Finding a Package with Amazon WAM

You can see a list of packages in the application catalog through the WorkSpaces console.

To find a package

2. Choose Packages, which shows a list of all of the packages in the application catalog.

Sharing a Package with Amazon WAM

You can create an application package and then share it with other AWS accounts. Accounts with access to your shared package can create a new application package that is based on your shared application.

Your packages are shared by version. To share a new version of a package, add the AWS account number to the permission list of the new package.

To share a package

You need the AWS account numbers for the accounts with which you want to share the package.

2. Choose Packages, which shows a list of all of the packages in the application catalog.
3. Select your package and choose Modify Permissions.
4. For Version, select the version of the package to share.
5. For **AWS Account Number**, type the AWS account number of the account with which to share the package and choose **Add Permission**.

6. Choose **Save**.

   After you add it to the permission list, the other account can upload the application using the shared package to their application catalog. For more information, see **Uploading an Application** (p. 6).

**To remove an account from the permissions list**

1. Open the WorkSpaces console at **https://console.aws.amazon.com/workspaces/**.
2. Choose **Packages**, which shows a list of all of the packages in the application catalog.
3. Select your package and choose **Modify Permissions**.
4. Next to the account number to delete, choose **Delete** and **Save**.

---

**Deleting a Package with Amazon WAM**

You can delete a package through the Amazon WorkSpaces Application Manager Studio. When you delete a package, you delete all versions of the packages.

You cannot delete a package if it has been shared with other AWS accounts within the region or has apps created and provisioned from it. To delete this kind of package, unshare the package and delete all apps associated with it.

**To delete a package**

1. Connect to your packaging instance using a remote desktop client and launch Amazon WorkSpaces Application Manager Studio. The remaining instructions are performed on the package instance.
2. Select a package from the **Dashboard**, and then choose from one of the following options:
   - Open the context menu (right-click) and choose **Delete**.
   - Choose **Delete** in the toolbar.
   - Open the package to view the package details, and then choose **Delete** in the toolbar.

---

**Understanding Color Coding**

You can determine the status of an item in the dashboard by looking at the color. If the item has a conflict, hover over the item to see a message.

The following table shows the meaning of each color.

<table>
<thead>
<tr>
<th>Item codes</th>
<th>Item type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Black</td>
<td>All items</td>
<td>Normal.</td>
</tr>
<tr>
<td>Grey</td>
<td>Template folders</td>
<td>The folder does not contain any files.</td>
</tr>
<tr>
<td>Grey Italic</td>
<td>Files</td>
<td>The item is excluded from the package.</td>
</tr>
</tbody>
</table>
# Understanding Color Coding

<table>
<thead>
<tr>
<th>Color / attribute</th>
<th>Item type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Blue</td>
<td>Files</td>
<td>The item was added after the package was created. This color does not apply to items that were manually added to the package.</td>
</tr>
<tr>
<td></td>
<td>Folders and subfolders</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Registry keys and values</td>
<td></td>
</tr>
<tr>
<td>Magenta</td>
<td>Files</td>
<td>The item was changed after the package was created. This color does not apply to items that were manually added to the package.</td>
</tr>
<tr>
<td></td>
<td>Folders and subfolders</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Registry keys and values</td>
<td></td>
</tr>
<tr>
<td>Red</td>
<td>Files</td>
<td>The item was skipped during package creation.</td>
</tr>
<tr>
<td></td>
<td>Folders and subfolders</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Registry keys and values</td>
<td></td>
</tr>
<tr>
<td>Red</td>
<td>All items</td>
<td>The item is missing from the package or the item has a conflict. For more information, see the following table on conflict codes.</td>
</tr>
</tbody>
</table>

**Bold text in the tabs represent a conflict in your package. The following sections describe the conflicts in the tabs.**

### Files tab

<table>
<thead>
<tr>
<th>Field</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>Source File</td>
<td>The file for the item is not in the package. Add the missing file to the package.</td>
</tr>
</tbody>
</table>

### Fonts tab

<table>
<thead>
<tr>
<th>Field</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Action</strong> and <strong>File Disposition</strong></td>
<td>The fonts are set to be registered, but the font will not be visible to the operating system because the Fonts key is set to Virtual-isolated (layer 4).</td>
</tr>
<tr>
<td><strong>File Path</strong></td>
<td>The file for the item is not in the package. Add the missing file to the package.</td>
</tr>
</tbody>
</table>
Startup tab

<table>
<thead>
<tr>
<th>Field</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Name and Command Disposition</strong></td>
<td>The Registry Run (or RunOnce) key asset cannot be started by the system because the key is set to Installed-permanent (layer 1) or Installed-temporary (layer 2), but its target command line is set to Virtual-integrated (layer 3) or Virtual-isolated (layer 4) so it will not be visible to the system.</td>
</tr>
</tbody>
</table>
| **Command Disposition** for entries under HKCU Run or HKCU RunOnce | One of these two conditions exists:  
  - The shortcut file does not exist on the system.  
  - The startup shortcut cannot be started by the operating system because it is set to Installed-permanent (layer 1) or Installed-temporary (layer 2), but its target command line is set to Virtual-integrated (layer 3) or Virtual-isolated (layer 4) so it will not be visible to the operating system. |

Services & Drivers tab

<table>
<thead>
<tr>
<th>Field</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Action</strong></td>
<td>This action cannot be performed on a service whose type is not specified.</td>
</tr>
<tr>
<td><strong>Error Control</strong></td>
<td>The error control type is not specified.</td>
</tr>
<tr>
<td><strong>Interactive</strong></td>
<td>The interactive setting is not specified.</td>
</tr>
<tr>
<td><strong>Image File</strong></td>
<td>The driver is set to start on boot or system start, but its service key or image file is set to Virtual-integrated (layer 3) or Virtual-isolated (layer 4) so it will not be visible to the system at that time.</td>
</tr>
<tr>
<td><strong>Image File and Key Disposition</strong></td>
<td>The image file layer is greater than the service key layer, which can yield inconsistent behavior except if the service key is set to Virtual-integrated (layer 3) or Virtual-isolated (layer 4).</td>
</tr>
<tr>
<td><strong>Name</strong></td>
<td>The service is set to restart or reboot automatically upon failure. This will cause the service to always restart when the application is shut down, which will put it back into the Running state. To fix this issue, set the failure action of the service to Turn Failure Actions OFF.</td>
</tr>
<tr>
<td><strong>Start</strong></td>
<td>The start type is not specified.</td>
</tr>
<tr>
<td><strong>Type</strong></td>
<td>The service type is not specified.</td>
</tr>
</tbody>
</table>
Controlling Access to Amazon WAM Resources

Amazon WAM must have permission to perform certain actions on your behalf. You can grant this access using IAM roles.

By default, IAM users don't have permission to access Amazon WAM resources. To allow an IAM user to perform actions on Amazon WAM resources, you must create a policy that grants the user permission to access Amazon WAM.

For more information about configuring WorkSpaces to meet your security and compliance objectives, see Security in WorkSpaces.

Contents
- Create the Application Packaging Role (p. 53)
- Create the AWS Marketplace Access Role (p. 53)
- (Optional) Grant an IAM User Access to Amazon WAM (p. 55)

Create the Application Packaging Role

This IAM role allows the Amazon WAM packaging instance to access your application package catalog. If you have not already done so, create the AmazonWamAppPackaging role using the following steps.

To create an IAM role to access your Amazon WAM application catalog
1. Open the IAM console at https://console.aws.amazon.com/iam/.
2. In the navigation pane, choose Roles and then choose Create role.
3. On the Select type of trusted entity page, select EC2, and then choose Next: Permissions.
4. On the Attach permissions policies page, select the check box for the AmazonWorkSpacesApplicationManagerAdminAccess policy and then choose Next: Tags.
5. (Optional) Add tags to the role, and then choose Next: Review.
6. On the Review page, type AmazonWamAppPackaging as the name of the role, and then choose Create role.

Important
You must specify AmazonWamAppPackaging as the name of the role or packaging and validation applications can't access your packages.

Create the AWS Marketplace Access Role

This IAM role allows Amazon WAM to access the AWS Marketplace on your behalf. The first time you log in to the Amazon WAM console, you are prompted to create a role with the name AmazonWamMarketplace_Default_Role. You must allow this role to be created.
The following is the IAM policy for the `AmazonWamMarketplace_Default_Role` role:

```json
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "catalog-admin:Describe*",
        "catalog-admin:Get*",
        "catalog-admin:Search*",
        "catalog-admin:List*",
        "catalog-admin:CreateListing",
        "catalog-admin:UpdateListing",
        "catalog-admin:DeleteListing",
        "catalog-user:SimulateView*",
        "catalog-user:SimulateGet*",
        "catalog-user:SimulateBrowse*"
      ],
      "Resource": "*"
    }
  ]
}
```

This role trusts the `wam.amazonaws.com` service to assume it. The following is the trust policy document:

```json
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "",
      "Effect": "Allow",
      "Principal": {
        "Service": "wam.amazonaws.com"
      },
      "Action": "sts:AssumeRole"
    }
  ]
}
```

If you're using AWS Organizations to manage policies for your AWS account, include the following policy within the `AmazonWamMarketplace_Default_Role` role:

```json
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "Stmt1541731687000",
      "Effect": "Allow",
      "Action": [
        "ec2:AcceptReservedInstancesExchangeQuote",
        "wam:*",
        "catalog-admin:*",
        "catalog-user:*"
      ],
      "Resource": [
        "*
      ]
    }
  ]
}
```
(Optional) Grant an IAM User Access to Amazon WAM

The following IAM policy allows an IAM user or group of users to administer Amazon WAM.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "catalog-admin:*",
        "ds:*",
        "iam:ListAttachedRolePolicies",
        "iam:ListRoles",
        "iam:CreateRole",
        "iam:PutRolePolicy"
      ],
      "Resource": "*"
    }
  ]
}
```

For more information about embedding a policy in a user or group, see Working with Inline Policies in the IAM User Guide.
Troubleshooting Amazon WAM Issues

The following are possible issues that you might have using Amazon WAM.

Issues
- Finding the Log Files (p. 56)
- Application Packaging Issues (p. 56)
- Amazon WAM Application Issues (p. 59)
- Device Driver and Service Issues (p. 61)
- Missing Shortcut, File, and Folder Issues (p. 64)
- General Issues (p. 65)
- User Account Issues (p. 66)

Finding the Log Files

If you are running into packaging issues, check the log files.

The Amazon Studio log file contains every error that occurred when you packaged your application.

The Amazon Admin Player log file shows the files and registry keys captured and filtered to the package and any errors.

To see the Amazon WorkSpaces Application Manager Studio log file
1. On the packaging instance, open the C:\Program Files\Amazon\WAM Studio folder.
2. Open the WamStudio.log file.

To see the Amazon Admin Player log file
1. In the Amazon WAM Admin Player, choose File and Options.
2. In the Options dialog box, on the Log tab, choose View Log.

Application Packaging Issues

The following are possible application packaging issues.

Issues
- How can I hide the Add/Remove Programs entry for my application? (p. 57)
- I cannot change the settings in my package (p. 57)
- I cannot print from my application (p. 57)
How can I hide the Add/Remove Programs entry for my application?

Many applications add an entry in the Add or Remove Programs or Programs and Features list in the Windows Control Panel. This entry can be confusing because the application is virtual and cannot be uninstalled. You can remove this entry from the list.

To hide the application entry

1. In the project in Amazon WorkSpaces Application Manager Studio, choose the Registry tab.
2. Open the context (right-click) menu of the following registry key and choose Delete.

   HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Uninstall

I cannot change the settings in my package

You opened your package in read-only mode by double-clicking the package in the dashboard. To change the settings in your package, open the package in update mode by choosing Update on the Amazon Studio toolbar. If the application in the package is already installed on the packaging instance, a message box offers a choice to overwrite the existing files. Choose Yes to overwrite the files on the package instance with the files in the package. The overwritten files are displayed in the Progress dialog box.

I cannot print from my application

Some application install new printer devices, which can be included in a package. On the Files tab of Amazon WorkSpaces Application Manager Studio, the spool folder is a subdirectory found under the SystemDir system folder. Applications that come with printer drivers may not work properly if this folder is removed from the Files tab. On the Registry tab, the following registry key contains references to the printer drivers in the spool folder:

   HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Print\Environments\Windows NT x86\Drivers

If a new printer does not appear in the application, use the following procedure.

To restart the print spooler

1. In Amazon WorkSpaces Application Manager Studio, choose the Project tab on the top of the project pane.
3. In the Settings dialog box, choose Virtualization.
4. Under Special requirements, choose Restart Print Spooler and OK.
I cannot retrieve my application list

If you receive an error from the Amazon WorkSpaces Application Manager Studio stating that your application list could not be retrieved, verify that you have created the proper IAM role as explained in Create the Application Packaging Role (p. 53), and that you applied that role to the packaging instance.

My application does not run because of a licensing issue

Many applications employ some form of copy protection to prevent theft and enforce proper licensing. Amazon WorkSpaces Application Manager Studio does not remove copy protection. If you need to enable multiple users to use an application, package the application so that it uses a serial number for high volume licensing and multiple systems can activate with a single key (for example, Microsoft products use volume license keys). Otherwise, the application prompts users for a license; not all systems will be able to activate and make the application fully functional.

Test the package on other systems to ensure that there is no form of license enforcement that you are not already aware of. If the package does not work or prompts for licensing, retest the package on the packaging instance. If the application binds to hardware, it should work on the same system. For information about supported licensing models, contact the software ISV.

My application fails to start

Sometimes an application can fail to start when the command line is incorrect. You might have to specify a different executable at the command line or you might need to add parameters. Check the original application shortcut, typically found on the Start menu, and use the same command line and parameters as what appears in 4. Package build of the Amazon WorkSpaces Application Manager Studio.

Another problem may be that some applications modify the PATH environment variable of the local machine during installation or after starting for the first time. There may be some system-specific entries included in the PATH variable (for example: PATH=%PATH%;C:\ApplicationPath1;C:\ApplicationPath2). Ensure that the package does not contain system-specific entries by choosing the Environment tab, Path, and then checking for the any application-specific path environment variables in the Value field.

My application is missing dependencies

Many applications require the presence of a specific runtime library or a minimum version of the .NET Framework. Amazon WorkSpaces Application Manager Studio comes with the latest builds of the following C++ runtime libraries:

- Visual C++ 2008 x86 and x64
- Visual C++ 2005 x86 and x64

If your application requires other dependencies, install the dependencies before you install the application. After you capture the dependency installation, set the disposition of the dependency files and registry keys to Virtual-isolated (layer 4) before you upload your package.

To set the disposition of files and registry keys

1. In Amazon WorkSpaces Application Manager Studio, choose the Files tab on the top of the project pane.
2. Open the context menu (right-click) for the dependency file or folder and choose Properties.
3. In the Properties dialog box, choose Virtual-isolated (layer 4) and OK.
4. On the Registry tab, open the context menu (right-click) for the registry hive or key and choose Disposition.
5. In the Registry Key Disposition dialog box, choose Virtual-isolated (layer 4) and OK.

**My file associations do not work**

Windows manages file extension associations to applications under the HKEY_CLASSES_ROOT hive of the Windows registry. The file type must be set in the registry and the application executable must not be hidden. Make sure that the file extensions you need exist in the Registry tab of the project. If the extension is missing, manually add the extensions.

You can also package a new version of the application where you choose All processes under Capture file and registry activity for the following processes in the Capture Application Installation dialog box, run the application, and configure the default file associations. Check that the application executable file is not hidden and the disposition is set to Virtual-integrated (layer 3).

**To create a new package version**

To create a new package version, you need the installation files.

1. In Amazon WorkSpaces Application Manager Studio, choose the Dashboard tab on the top of the project pane, select your project in the list, and then choose Update in the toolbar and Close.
3. In the Capture Application Installation dialog box, do the following:
   - For the Installer path field, enter the path and file name to your installer.
   - For the Command line parameter field, enter any parameters required to install your application.
   - Under Capture file and registry activity for the following processes, choose All processes.
   - Choose Ignore changes under the Installer path specified above.
   - Choose Launch, install your application, and then choose OK and Close.
4. On the Files tab, open the context (right-click) menu for your executable file and choose Properties.
5. In File Properties, under Disposition, choose Virtual-integrated (layer 3) and OK.
6. On the Project tab, do the following:
   - In 4. Package build, enter text to describe this new version in the Label field. This text appears as a version of the package.
   - In 5. Package upload, choose Upload.

**Amazon WAM Application Issues**

The following are possible issues that you might have with Amazon WAM applications.

**Issues**

- I assigned applications to a user, but I don't see the user in the usage report (p. 60)
- I assigned applications to a user, but the user has no apps (p. 60)
- I assigned applications to a user, but the apps aren't appearing in the Amazon WorkSpaces Application Manager client application (p. 60)
- I assigned applications to a user, but the user sees the message "No apps yet. To get apps, contact your admin" in the Amazon WorkSpaces Application Manager client application (p. 60)
I assigned applications to a user, but I don't see the user in the usage report

You won't see the user in the usage report until after the user launches the Amazon WorkSpaces Application Manager client application for the first time. For more information, see Users (p. 13).

I assigned applications to a user, but the user has no apps

Verify that the Amazon WAM shortcut is on the WorkSpace client desktop. If the shortcut isn't on the client desktop, see Setting Up the Amazon WAM Installer and Accessing Assigned Applications.

I assigned applications to a user, but the apps aren't appearing in the Amazon WorkSpaces Application Manager client application

Applications that have the Installation type specified as Required should appear by default in the MY APPS section of the Amazon WAM client application. Applications that have the Installation type specified as Optional don't appear in the MY APPS section until the user chooses DISCOVER in the Amazon WAM client application.

I assigned applications to a user, but the user sees the message "No apps yet. To get apps, contact your admin" in the Amazon WorkSpaces Application Manager client application

Applications that have the Installation type specified as Required should appear by default in the MY APPS section of the Amazon WAM client application. Applications that have the Installation type specified as Optional don't appear in the MY APPS section until the user chooses DISCOVER in the Amazon WAM client application.

If the applications still aren't appearing after the user has chosen DISCOVER, the certificate authority (CA) might be missing, out of date, or not enabled. The WorkSpace must have at least one of the following certificate authorities (CA) installed and enabled:

- Amazon Root CA 1
- Starfield Services Root Certificate Authority - G2
- Starfield Class 2 Certificate Authority

Amazon WorkSpaces enforces the use of digital certificates issued by Amazon Trust Services, as of May 2018. Amazon Trust Services is already a trusted Root CA on the operating systems supported by Amazon WorkSpaces. However, if the Root CA list for your operating system is not up-to-date, you might encounter issues with Amazon WorkSpaces.
To add Amazon Trust Services to the local Root CA list in your WorkSpace

2. Download the Starfield certificate in DER format (2b071c59a0a0ae76b0eabd2bad23bad4580b69c3601b630c2eaf0613afa83f92).
3. Open the Microsoft Management Console. (From a Command Prompt window, run mmc.)
5. On the Certificates snap-in page, select Computer account and choose Next. Keep the default, Local computer. Choose Finish. Choose OK.
6. Expand Certificates (Local Computer) and select Trusted Root Certification Authorities. Choose Action, All Tasks, Import.
7. Follow the wizard to import the certificate that you downloaded.

I need to manage User Access Control on Windows

For apps that trigger User Access Control (UAC) input, the behavior with Amazon WAM is similar to natively installed apps. Amazon WAM does not introduce any changes. You can manage UAC through the same policies that you regularly use for managing user permissions.

For apps that use a script as part of the application installation (for example, KMS activation), the script can be set to run as local system administrator. If the user on a WorkSpace is not a local administrator, Amazon WAM requires the following two registry settings to be set to allow the script to execute within the package:

- HKLM\SOFTWARE\Amazon\StreamingCore\Settings\AppEvent\AllowLocalSystem,DWORD,"1"
- HKLM\SOFTWARE\Amazon\StreamingCore\Settings\Driver\PhysicalLayerMode,DWORD,"1"

I need to perform HTTPS Proxy Inspection on my WorkSpaces

All Amazon WAM endpoints are accessed over port 443 (HTTPS). If you are routing WorkSpace outbound traffic using a corporate proxy and performing HTTPS content inspection, we recommend that you exclude the Amazon WAM endpoints. Excluding the endpoints prevents validation or latency issues.

For Amazon WAM to function correctly, the following endpoints must be accessible over HTTPS:

- *.cloudfront.net
- wam-idb.region-code.amazonaws.com
- wam-ps.region-code.amazonaws.com
- s3.amazonaws.com
- *.s3.amazonaws.com
- s3-external-1.amazonaws.com
- *.s3-external-1.amazonaws.com
- *.s3-region-code.amazonaws.com
- sqs.region-code.amazonaws.com

Device Driver and Service Issues

The following are possible device driver and service issues.
How do I package drivers for Plug and Play devices?

Plug and play drivers for devices such as printers are installed in two phases:

- Amazon WorkSpaces Application Manager Studio uses the installer to copy the driver files to the hard drive.
- Windows associates the files with the correct device.

Starting with Vista, Windows uses a protected repository for trusted device drivers called the driver store. To set up a device drive, use a Configurable AppEvent that uses the `pnputil.exe` utility. This utility adds and removes drivers from the driver store. The syntax for adding drivers using the utility is:

```
pnputil.exe -i -a "C:\<folder>\driver.inf"
```

**Warning**

Do not attempt to set files in the driver store to Installed-permanent (layer 1) or Installed-temporary (layer 2) because it causes virtualization to fail.

**To exclude device driver files**

1. In Amazon WorkSpaces Application Manager Studio, choose the **Files** tab on the top of the project pane.
2. Open the context menu (right-click) for the `<WindowsDir>\inf\` folder and choose **Exclude**.
3. To exclude all virtual files used by the device driver, do the following:
   - On the instance desktop, do the following to open Device Manager:
     - Choose **Start**.
     - type `devmgmt.msc` and then press Enter.
     - Double-click the device driver in the list to open the **Properties** dialog box.
     - In **Properties**, choose the **Driver** tab and **Driver Details**, make note of the files listed in the **Driver File Details** dialog box, and then choose **OK**.
     - In Amazon WorkSpaces Application Manager Studio, choose the **Files** tab on the top of the project pane.
     - Open the context menu (right-click) for each file and choose **Exclude**.
4. To exclude files for TWAIN devices, do the following:
   - In the **Dashboard** tab of Amazon WorkSpaces Application Manager Studio, select your project and choose **Update** on the toolbar.
   - Choose the **Files** tab.
   - For each driver file under `<WindowsDir>\SSDriver\`, open the context menu (right-click) and choose **Properties**.
   - In the **Properties** dialog box, choose **Installed-temporary (layer 2)** under **Disposition** and **OK**.
   - In Amazon WorkSpaces Application Manager Studio, select your project on the top of the project pane, and then under **3. Optional settings**, choose **Settings**.
   - In **Settings**, choose **Configurable AppEvents** and **Add**.
7. In the **Configurable AppEvent Settings** dialog box, choose **After activation** under **Trigger**.

8. • In **File name** in **Handler command line**, type `%SystemRoot%\system32\pnputil.exe`. Under **Handler Launching**, choose **Run elevated (only supported on Windows Vista or later)** and **OK**.

**Note**
Device drivers remain on the system even after the application has been removed by Amazon WAM Admin Player. This is expected behavior.

---

**I cannot stop a service with a running status**

Windows Task Manager tracks all executable files and services with Virtual-integrated (layer 3) or Virtual-isolated (layer 4) disposition. When the executable file or service is in use, Windows Task Manager shows this file with a **Status** value of **Running**.

To avoid usage tracking, set the disposition of these files (such as ServiceImagePath) and services to Installed-temporary (layer 2) disposition.

In addition, some services may be configured to restart automatically after an interruption.

**To configure an interrupted service to restart automatically**

1. In Amazon WorkSpaces Application Manager Studio, choose the **Registry** tab on the top of the project pane.
2. Remove the Failure action key from the following hive for the service:

```
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services
```

**My Windows services or drivers do not start**

If a service needs to start using a unique user account (this normally defaults to LocalSystem), the application needs a Configurable AppEvent to create this account.

**Warning**
Some systems services have security descriptors that are strictly enforced. When adding a new service, it is important to match the security descriptor of the installed version. Use `sc sdshow <service_name>` at the command prompt to display the security descriptor set for the service.
Marking an application to require a system restart after adding a service typically corrects the security descriptors.

If the application has boot or system drivers, mark all of the files and registry keys for those services or drivers as Installed-temporary (layer 2).

**To set the disposition of the files and registry keys**

1. In Amazon WorkSpaces Application Manager Studio, choose the **Services & Drivers** tab.
2. Open the context menu (right-click) for the service or driver and choose **Service Key Disposition**.
3. In the dialog box, choose **Installed-temporary (layer 2)**, and **OK**.
4. Open the context menu (right-click) for the service or driver and choose **Image File Properties**.
5. In the dialog box, choose **Installed-temporary (layer 2)**, and **OK**.

**Warning**
Set both the files and registry keys to Installed-temporary (layer 2) disposition. You may have problems if the files and registry keys do not have the same disposition setting.
6. On the Registry tab, open the context menu (right-click) for each legacy registry key for your service or driver and choose Exclude.

7. Open the context menu (right-click) for each Enum registry key for your service or driver and choose Exclude.

- HKEY_LOCAL_MACHINE\System\CurrentControlSet\Enum\Root
- HKEY_LOCAL_MACHINE\System\CurrentControlSet\Services\{NAME}\Enum

8. On the Project tab, choose to allow or avoid a system restart. In 3. Optional settings, choose Settings, and then do one of the following in the Settings dialog box:

To require a system restart
1. Choose Virtualization.
2. Under Special requirements, select Require system reboot and OK.

To avoid a system restart
Use a Configurable AppEvent to avoid a system restart.
1. Choose Configurable AppEvents and Add.
2. In the Configurable AppEvent Settings dialog box, choose After Launch under Trigger.
3. In File name under Handler command line, type sc.exe start <driver_name> and OK.

For more information about how services work, go to How to create a Windows service in the Microsoft Knowledge Base.

Note
A system driver cannot be stopped and removed from a system without a reboot. Amazon WorkSpaces Application Manager Studio removes these files from the system but the driver is still loaded until a reboot occurs.

Missing Shortcut, File, and Folder Issues

The following are possible issues with missing shortcuts, files, and folders.

Issues
- My shortcut is missing (p. 64)
- My shortcut is a blank window (p. 65)
- The text is in the wrong font (p. 65)
- My folders and files are not in the application in my WorkSpace (p. 65)

My shortcut is missing

If the shortcut files are missing after the application has been installed, verify the following:

- For the shortcut to appear on the Start menu, the shortcut should be located in the CommonPrograms or UserPrograms folder.
- For the shortcut to appear on the Windows desktop, the shortcut file should be located in the CommonDesktop or UserDesktop folder.
- The Disposition setting for the shortcut files and folders should be set to Virtual-integrated (layer 3). This allows that shortcut to be accessible on the client computer's operating system.
• The security setting on the shortcut files should not have the **Hide from folder listing** option selected. If this option is selected, the shortcut files will not be visible on the client computer.

**My shortcut is a blank window**

If the desktop icon for the application is a blank Windows icon, the executable file for the application may be set to **Prevent Copying** under **Security settings** in the **File Properties** dialog box. This is normal behavior with this setting and the application functions normally.

**The text is in the wrong font**

Most applications come with their own program fonts. Fonts do not appear if an application is packaged with **Inject fonts for use only by the application** selected under **Virtualization** in the **Settings** dialog box. Private fonts are only used in the application if it is launched from the Amazon WorkSpaces Application Manager Studio.

**To make private fonts available to all applications**

1. In Amazon WorkSpaces Application Manager Studio, choose the **Projects** tab on the top of the project pane.
2. Under **3. Optional settings**, choose **Settings**.
3. In the **Settings** dialog box, choose **Virtualization**. Under **Fonts**, choose **Register fonts for system-wide use** and **OK**.

In addition to the **Fonts** tab, registered fonts can be found in the following locations:

- In the **fonts** directory in **SystemFonts** on the **Files** tab.
- In the following registry key on the **Registry** tab:

  `HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Fonts`

**My folders and files are not in the application in my WorkSpace**

If you package an application to an install folder that has the same name as a folder used in a WorkSpace, the folder in the WorkSpace may disappear. This can also affect files under common folders marked with Virtual-integrated (layer 3) disposition for integrated, but not merged.

**To fix an existing application package with missing folders and files**

1. Use a unique name for the install folder.
2. In Amazon WorkSpaces Application Manager Studio, choose the **Files** tab on the top of the project pane.
3. Open the context (right-click) menu and then choose **Properties**.
4. In the **File Properties** dialog box, choose **Virtual-isolated (layer 4)** under **Disposition** and **OK**.

**General Issues**

The following are general issues that you might have when packaging an application.
My users see a warning message about an alternate data stream (ADS)

Users may receive warning messages about alternate data streams (ADS) when their application starts in WorkSpaces. In most cases, these messages are not a serious problem and can be ignored. In rare cases, the application may require an ADS to use a particular file. In this case, the package should exclude the file from the sandbox. Set the disposition of the file to Installed-permanent (layer 1) or Installed-temporary (layer 2).

To exclude a file from the sandbox and set the disposition

1. In Amazon WorkSpaces Application Manager Studio, choose the Files tab on the top of the project pane.
2. Open the context menu (right-click) for the file and choose Exclude.
3. Open the context menu (right-click) for the file and choose Properties.
4. In the Properties dialog box, choose Installed-permanent (layer 1) or Installed-temporary (layer 2) under Disposition and OK.

My folder, which should have files from other applications, contains only files from my application

Your folder is a root folder whose contents are only visible to the application in the package. To see the files from other applications in the unmerged folder, update the package, and configure the folder to be a merged folder. A merged folder makes files from other application visible to all applications. To change an unmerged folder into a merged folder, open the context (right-click) menu for the unmerged folder in the Files tab, and then choose Merged Folder.

User Account Issues

The following are possible issues user account issues that you might have when packaging an application.

Issues

• My roaming user accounts and folder redirection does not work (p. 66)
• How do I create unique user accounts? (p. 67)

My roaming user accounts and folder redirection does not work

Amazon WorkSpaces Application Manager Studio allows the HKEY_CURRENT_USER registry settings and the %appdata% folder contents for roaming profile support. Any limitations with Windows roaming profiles still apply.
If you are using folder redirection to a network share through a mapped drive or a UNC path, set the disposition of the folder and the files to the recommended Installed-permanent (layer 1) or Installed-temporary (layer 2). The application will not start if the files or folders are set to Virtual-integrated (layer 3) or Virtual-isolated (layer 4).

For example, if the %appdata% folder is being redirected to \filesystem\user1\appdata, the template folder ?roamingappdata? must only contain files and folders with Installed-permanent (layer 1) or Installed-temporary (layer 2) disposition. Redirection to another local disk supports files set to the Virtual-Integrated (layer 3) or Virtual-isolated (layer 4) disposition.

**Warning**

With folder redirection, the network folder must exist prior to activating the application. For example, if %appdata% is mapped to n:\{username}\appdata, the appdata folder must already exist on the server.

**To enable roaming profile support**

1. In Amazon WorkSpaces Application Manager Studio, choose the **Project** tab on the top of the project pane.
2. Under **3. Optional settings**, choose **Settings**.
3. In the **Settings** dialog box, choose **Sandboxing, Allow Windows to roam application settings** under **Application settings**, and **OK**.

**How do I create unique user accounts?**

Some applications create a special user account when you install them. Amazon WorkSpaces Application Manager Studio cannot capture user accounts because they are not transferable between operating systems. You need to create a Configurable AppEvent to replicate the user accounts and password on the host computer programmatically when Amazon WorkSpaces Application Manager Studio installs the application.

**Note**

User accounts may appear as a security identifier (SID) in the registry. For example, S-1-5-18 is the local system account. For more information, go to [Well-known security identifiers in Windows operating systems](https://support.microsoft.com/en-us/help/13985) in the Microsoft Knowledge Base.
Amazon WorkSpaces Application Manager Quotas

The following table lists the quotas (also referred to as limits) for Amazon WorkSpaces Application Manager (Amazon WAM). The Service Quotas console provides information about Amazon WAM quotas. You can use the Service Quotas console to view the default service quotas or to request quota increases for adjustable quotas.

<table>
<thead>
<tr>
<th>Resource</th>
<th>Default</th>
<th>Description</th>
<th>Adjustable</th>
</tr>
</thead>
<tbody>
<tr>
<td>Application assignments per user</td>
<td>50</td>
<td>The maximum number of application assignments per user, in this account in the current Region.</td>
<td>Yes</td>
</tr>
<tr>
<td>User/WorkSpace or group assignments per application</td>
<td>200</td>
<td>The maximum number of user/WorkSpace or group assignments per application, in this account in the current Region.</td>
<td>Yes</td>
</tr>
<tr>
<td>Total package size without storage fees</td>
<td>100 GB</td>
<td>The maximum total size (in GB) of all your packages without storage fees, in this account in the current Region. There is no quota for the number of applications you can package, but storage fees will be applied if your packages exceed 100 GB.</td>
<td>No</td>
</tr>
<tr>
<td>Application size</td>
<td>5 GB</td>
<td>The maximum application size (in GB) that can be packaged in this account in the current Region. Applications that are larger than 5 GB cannot be packaged using Amazon WAM.</td>
<td>No</td>
</tr>
</tbody>
</table>