Setting Up Resources in VMware Workspace Portal

Workspace Portal 2.1

This document supports the version of each product listed and supports all subsequent versions until the document is replaced by a new edition. To check for more recent editions of this document, see http://www.vmware.com/support/pubs.

EN-001541-01
You can find the most up-to-date technical documentation on the VMware Web site at:
http://www.vmware.com/support/
The VMware Web site also provides the latest product updates.
If you have comments about this documentation, submit your feedback to:
docfeedback@vmware.com
## Contents

1 About the VMware Workspace Portal Resource Guide 5

2 General Information About Setting Up Resources in Workspace 7
   Accessing the Workspace Consoles 7

3 Providing Access to Web Applications 9
   Establish Secure Single Sign-On to Your Web Applications Through Workspace 9
   Adding Web Applications to Your Organization’s Catalog 12
   Entitle Users and Groups to Web Applications 17

4 Providing Access to View Desktop and Application Pools 19
   Integrating View 19
   Reducing Resource Usage and Increasing Performance of Workspace for Windows In Non-Persistent View Desktops 25

5 Providing Access to VMware ThinApp Packages 27
   Integrating VMware ThinApp Packages 28
   Entitle Users and Groups to ThinApp Packages 35
   Distributing and Managing ThinApp Packages with Workspace 37
   Using the Command-Line HorizonThinAppCtrl.exe Application 40
   Updating Managed ThinApp Packages After Deployment in Workspace 41
   Delete ThinApp Packages from Workspace 46
   Make Existing ThinApp Packages Compatible with Workspace 47
   Change the ThinApp Packages Share Folder 49

6 Configuring Workspace for Windows 51
   Command-Line Installer Options for Workspace for Windows 51
   Install the Workspace for Windows Application with Identical Settings to Multiple Windows Systems 57
   Add Desktop Client Installer Files to workspace-va Virtual Appliances 58

7 Providing Access to Citrix-Published Resources 61
   Integrating Workspace with Citrix-Published Resources 61
   Enabling Citrix PowerShell Remoting on Citrix Server Farm 64
   Preparing and Installing Integration Broker 66
   Deploying Integration Broker 67
   Synchronizing Workspace with Integration Broker 70
   View User and Group Entitlements to Citrix-Published Resources 72
   Editing Workspace Application Delivery Settings for Citrix-Published Resources 72
   Managing Categories for Citrix-Published Resources 74
8  Troubleshooting Workspace Resource Configuration  75
    Blank Screen Displays When Installing Update to Workspace for Windows  75
    ThinApp Packages Fail to Launch from the User Portal  76
    Users Accessing Citrix-Published Resources Receive an Encryption Error  79
    Citrix-Published Resources Are Not Available in Workspace  79
    When Users Launch a Citrix-Published Resource, the Browser Displays 500 Internal Server Error  81
    Memory Issue Prevents Proper Configuration of Integration Broker  82

Index  83
About the VMware Workspace Portal Resource Guide

The Setting Up Resources in VMware Workspace™ Portal Guide provides information and instructions to add and customize resources that you add to the Workspace catalog. Such resources include Web applications, Windows applications captured as ThinApp® packages, View desktops and application pools, DaaS® desktops, and Citrix-published resources.

Intended Audience

This information is intended for anyone who configures and administers the resources for Workspace. The information is written for experienced Windows or Linux system administrators who are familiar with virtual machine technology.
Accessing the Workspace Consoles

The Workspace consoles include an end-user console, the Workspace App Portal, and the following administrator consoles, Appliance Configurator, Connector Services Admin, and Workspace Admin Console. The administrator consoles are used to set up different features in Workspace. You can access all of the administrator consoles at https://WorkspaceFQDN.com:8443.

You can also access each console directly with the URL specific to that console.

**Table 2.1. Workspace URLs**

<table>
<thead>
<tr>
<th>Workspace URLs</th>
<th>What you can do here</th>
</tr>
</thead>
<tbody>
<tr>
<td>Appliance Configurator <a href="https://WorkspaceFQDN.com:8443/cfg/login">https://WorkspaceFQDN.com:8443/cfg/login</a></td>
<td>In the Appliance Configurator, you can manage the database, update certificates, enable Syslog, change the Workspace admin password, and manage other infrastructure functions. Log in as the Workspace administrator, using the username admin and the admin password you set during configuration.</td>
</tr>
<tr>
<td>Connector Services Admin <a href="https://WorkspaceFQDN.com:8443/hc/admin/about">https://WorkspaceFQDN.com:8443/hc/admin/about</a></td>
<td>In the Connector Services Admin, you can configure the directory, set up auth adapters, and administer other enterprise integrations such as virtual desktops and remote applications. Integrations include ThinApp, View and Application pools, DaaS Desktops, and Citrix-published resources. You can also check directory sync status and alerts. Log in as the Workspace administrator, using the username admin and the admin password you set during configuration.</td>
</tr>
</tbody>
</table>
Table 2-1. Workspace URLs (Continued)

<table>
<thead>
<tr>
<th>Workspace URLs</th>
<th>What you can do here</th>
</tr>
</thead>
<tbody>
<tr>
<td>Workspace Admin Console</td>
<td>In the Workspace Admin Console, you can set up the resource catalog and administer your users and groups, entitlements, and reports. Log in using this URL with your Active Directory user name and password. You go directly to the Workspace Admin Console dashboard.</td>
</tr>
<tr>
<td><a href="https://WorkspaceFQDN.com/SAAS/admin">https://WorkspaceFQDN.com/SAAS/admin</a></td>
<td></td>
</tr>
<tr>
<td>Workspace App Portal</td>
<td>From this URL, sign in with your Active Directory user name and password. You go to the Workspace App Portal, also referred to as the User Portal, which includes applications that are enabled for you as an end user. From here you can manage Workspace applications enabled for your use or go to the Workspace Admin Console from the Administration Console link in the drop-down menu by your name.</td>
</tr>
<tr>
<td><a href="https://WorkspaceFQDN/SAAS/apps/#/launcher">https://WorkspaceFQDN/SAAS/apps/#/launcher</a></td>
<td></td>
</tr>
</tbody>
</table>
Providing Access to Web Applications

You can entitle Workspace users to your organization's external Web applications.

To enable users to access a Web application through Workspace, verify that the following requirements are met:

- If you configure the Web application to use a federation protocol, use SAML 1.1, SAML 2.0, or WS-Federation 1.2. However, you have the option of configuring the Web application to not use a federation protocol at all.
- The users you plan to entitle to the Web application are registered users of that application.
- If the Web application is a multitenant application, Workspace points to your instance of the application.

This chapter includes the following topics:

- “Adding Web Applications to Your Organization's Catalog,” on page 12
- “Entitle Users and Groups to Web Applications,” on page 17

Establish Secure Single Sign-On to Your Web Applications Through Workspace

You can provide single sign-on through Workspace to your Web applications that are configured with either SAML or WS-Federation protocol for authentication.

Configuring Web Applications that Use SAML Protocol

Many of the applications in the cloud application catalog use Security Assertion Markup Language (SAML1 or SAML 2) to exchange authentication and authorization data to verify that users can access a Web application.

The configuration form for adding Web applications to your catalog is partially configured. You can complete some SAML configurations in the Workspace Admin Console, but you might also need to work with your Web application account representatives to complete other required setup.

Configuring Workspace for Single-Sign on to Microsoft Office 365 Applications

Office 365 SharePoint and Office 365 Outlook Web applications can be configured for single sign-on through Workspace. To use single sign-on to access these Office 365 applications, the Microsoft Office 365 domain must be changed from managed to federated, and Office 365 domain parameters settings changed to authenticate through Workspace.
To set up single-sign on between Office 365 and Workspace, you must perform the following.

- Make changes to the domain attribute mapping in Workspace
- Synchronize Active Directory to Workspace
- Update the settings for the Office 365 account to Workspace settings

**End User Authentication to Office 365 from Native Clients**

Office 365 does not support single sign on for desktop or mobile native clients, such as Outlook.

**Table 3-1. Supported Native Clients**

<table>
<thead>
<tr>
<th>Device Type</th>
<th>Native Client</th>
</tr>
</thead>
<tbody>
<tr>
<td>Windows computers</td>
<td>Microsoft Outlook Client</td>
</tr>
<tr>
<td>Apple Mac computers</td>
<td>Microsoft Outlook Client, Mac Outlook Web App</td>
</tr>
<tr>
<td>iPhone and iPad devices</td>
<td>iOS Email, Outlook Web App</td>
</tr>
<tr>
<td>Android devices</td>
<td>Android Email</td>
</tr>
</tbody>
</table>

You determine the credential verification method that Workspace uses to authenticate Workspace users attempting to access Office 365. In the Workspace Admin Console, click the Catalog tab, from the Any Application Type drop-down menu, select Web Applications, click the Office 365 Outlook application, and click Configuration. On the Application Configuration page, select a method from the Credential Verification drop-down menu.

<table>
<thead>
<tr>
<th>Credential Verification Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Active Directory Password</td>
<td>When you select the Active Directory Password credential verification method, Workspace users use their Active Directory password to access Office 365.</td>
</tr>
<tr>
<td>Per App Password</td>
<td>When you select the Per App Password credential verification method, Workspace users must configure a password in the Office 365 application. Users right-click the Office 365 application in the Workspace App Portal and click Set Password. They then configure their native client with this password. A forgotten password cannot be retrieved. If users forget their passwords, they go back to the Office 365 app and enter a new password. They also must change this password on the native client.</td>
</tr>
</tbody>
</table>

**Office 365 Requirements**

Work with your Microsoft service provider to make sure that your managed Office 365 environment is correctly set up before you configure Workspace for single sign-on. The Office 365 directory synchronization tool must have synchronized your Active Directory to the Office 365 account, and the Windows PowerShell must be installed on the Windows server.

**Mapping Attributes in Workspace**

To enable Workspace to interact with Office 365, you must map the following Workspace user attributes to the Active Directory user attributes.

**Procedure**

1. Log in to the Connector Services Admin.
2. Click User Attributes and verify that the Workspace userPrincipalName attribute is mapped to the Directory userPrincipalName attribute.
3. Click Add an attribute and add the Workspace attribute objectGUID and map it to the Directory attribute objectGUID.
4 Click Save.

5 To sync your changes to Active Directory immediately, select Directory Sync and click Edit Directory Sync Rules to run the sync wizard, otherwise your changes are synced to Active Directory at your next scheduled sync interval.

6 Exit the Connector Services Admin.

**What to do next**

Convert your Office 365 managed domain to a federated domain for single sign-on and update the settings for the Office 365 account to Workspace settings.

### Converting Office 365 to a Federated Domain for Single Sign-On and Changing Office 365 Parameters to Workspace

You must convert your Office 365 managed domain to a federated domain for single sign-on and update the settings for the Office 365 account to Workspace settings.

Prepare your Office 365 domain to use Workspace for authentication.

**Procedure**

1 Run the `Set-MsolDomainAuthentication` cmdlet to change the following variables to the Workspace settings.

<table>
<thead>
<tr>
<th>Table 3-2. Replacing the Cmdlet Variables</th>
</tr>
</thead>
<tbody>
<tr>
<td>Line of cmdlet</td>
</tr>
<tr>
<td>------------------------------------------</td>
</tr>
<tr>
<td>-DomainName</td>
</tr>
<tr>
<td>-IssuerUri</td>
</tr>
<tr>
<td>-FederationBrandName</td>
</tr>
<tr>
<td>-PassiveLogOnUri</td>
</tr>
<tr>
<td>-ActiveLogOnUri</td>
</tr>
<tr>
<td>-LogOffUri</td>
</tr>
</tbody>
</table>
### Table 3-2. Replacing the Cmdlet Variables (Continued)

<table>
<thead>
<tr>
<th>Line of cmdlet</th>
<th>cmdlet Variable or Variables</th>
<th>Replace with</th>
</tr>
</thead>
<tbody>
<tr>
<td>-MetadataExchangeUri</td>
<td>host and port</td>
<td>The URL that specifies the metadata exchange end point used for authentication, such as <a href="https://example123.mycompany.com/SAAS/auth/wsfed/services/mex">https://example123.mycompany.com/SAAS/auth/wsfed/services/mex</a>.</td>
</tr>
<tr>
<td>-SigningCertificate</td>
<td>SAML signing cert from Application Manager</td>
<td>The Workspace Manager signing certificate. On the Workspace Admin Console, go to Settings &gt; SAML Certificate, and copy the Signing Certificate and paste this as the value for SigningCertificate. Exclude &quot;-----BEGIN CERTIFICATE-----&quot; and &quot;-----END CERTIFICATE-----&quot; from the certificate content. Notice Make sure you do not include additional spaces or extra line returns when you paste the certificate or it will not work.</td>
</tr>
</tbody>
</table>

2 Verify the federation settings. Type **Get-MsolDomainFederationSettings -DomainName <YOUR DOMAIN>**

**Example: Example of Output From Powershell Cmdlet**

```
Set-MsolDomainAuthentication
-DomainName example.mycompanydomain_name.com
-Authentication Federated
-IssuerUri example
-FederationBrandName Mycompany, Inc.
-PassiveLogOnUri https://host:port/SAAS/API/1.0/POST/sso
-LogOffUri https://login.microsoftonline.com/logout.srf
-ActiveLogOnUri https://host:port/SAAS/auth/wsfed/active/logon
-MetadataExchangeUri https://host:port/SAAS/auth/wsfed/services/mex
-SigningCertificate
```

Adding Web Applications to Your Organization's Catalog

You can add your organization's Web applications to your catalog and make these applications accessible to your Workspace users and groups.

When you add an entry for a Web application to the catalog, you create an application record and configure the address of the Web application. Workspace uses the application record as a template to establish a secure connection with the Web application.

The following methods can be used to add application records of Web applications from the Catalog page.
Add a Web Application to Your Catalog from the Cloud Application Catalog

The cloud application catalog is populated with Web applications. These applications include some information in their application records. When you add a Web application to your catalog from the cloud application catalog, you must provide additional information to complete the application record.

When you add a Web application to the catalog, you are creating an entry that points indirectly to the Web application. The entry is defined by the application record, which is a form that includes a URL to the Web application.

Prerequisites

When you add a Web application to the catalog, you can apply a Web-application-specific access policy set to control user access to the application. If such a Web-application-specific access policy set does not already exist and you intend to apply one to this Web application, create the access policy set now. See Workspace Administrator’s Guide for information about managing Web-application specific policy sets.

Procedure

1. Log in to the Workspace Admin Console.
2. Click the Catalog tab.
3. Click + Add Application > Web Application ...from the cloud application catalog.
4. Click the icon of the Web application to add to your organization’s catalog.
   The application record is added to your catalog, and record’s Details page displays with the name and authentication profile already specified in the form.
5. (Optional) Customize the information on the Details page for your organization’s needs.
   Items on the page are populated with information specific to the Web application.
   For example, you might choose an icon that best represents this Web application to your Workspace users when they see the application listed in their Workspace client.
   You can edit some of the populated items, depending on the application.

<table>
<thead>
<tr>
<th>Form Item</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>If necessary, change the name of the application.</td>
</tr>
<tr>
<td>Description</td>
<td>Add a description of the application that the users can read.</td>
</tr>
</tbody>
</table>
Click Choose File to upload an icon for the application. Workspace supports PNG, JPG, and ICON file formats up to 4MB. Workspace resizes uploaded icons to 80px X 80px. To prevent distortion, upload icons where the height and width are equal to each other and as close as possible to the 80px X 80px resize dimensions.

To allow the application to appear in a category search of catalog resources, select the respective category from the drop-down menu. You must have created the category earlier.

6 Click **Save**.

7 Click **Configuration**, edit the application record’s configuration details, and click **Save**.

Some of the items on the form are prepopulated with information specific to the Web application. Some of the prepopulated items are editable, while others are not. The information requested varies from application to application.

For some applications, the form has an Application Parameters section. If the section exists for an application and a parameter in the section does not have a default value, provide a value to allow the application to launch. If a default value is provided, you can edit the value.

When you use Office 365 SharePoint or Office 365 Outlook applications, you must edit the Application Parameters section. For Office 365 SharePoint, enter your domain as `mycompany`. For Office 365 Outlook, enter your domain as `mycompany.com`.

8 Select the **Entitlements**, **Licensing**, and **Provisioning** tabs and customize the information as appropriate.

<table>
<thead>
<tr>
<th>Tab</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Entitlements</td>
<td>Entitle users and groups to the application. You can configure entitlements while initially configuring the application or anytime in the future.</td>
</tr>
<tr>
<td>Access Policies</td>
<td>Apply a Web-application-specific access policy set to control user access to the application.</td>
</tr>
<tr>
<td>Licensing</td>
<td>Configure license tracking. Add license information for the application to track license use in reports.</td>
</tr>
<tr>
<td>Provisioning</td>
<td>Select a provisioning adapter. Workspace ships with the provisioning adapters for Google Apps and Mozy. If you are configuring either of these Web applications, you can select the appropriate provisioning adapter. Provisioning provides automatic application user management from a single location. Provisioning adapters allow the Web application to retrieve specific information from Workspace as required. For example, to enable automatic user provisioning to Google Apps, user account information, such as user ID, first name, and last name must exist in the Google Apps database. An application might require other information, such as group-membership and authorization-role information.</td>
</tr>
</tbody>
</table>

What to do next

For details about adding user and group entitlements for Web applications, see “Entitle Users and Groups to Web Applications,” on page 17.
Add a Web Application to Your Catalog by Creating a New Application Record

You create an application record when the Web application to add to your catalog is not available in the cloud application catalog.

Prerequisites

When you add a Web application to the catalog, you can apply a Web-application-specific access policy set to control user access to the application. If such a Web-application-specific access policy set does not already exist and you intend to apply one to this Web application, create the access policy set now. See Workspace Administrator’s Guide for information about managing Web-application specific policy sets.

When you successfully complete the application record for a Web application, an entry is created in your catalog that points indirectly to the Web application, and the Web application and Workspace can use SAML to communicate with each other.

Procedure

1. Log in to the Workspace Admin Console.
2. Click the Catalog tab.
3. Click + Add Application > Web Application ...create a new one. The application record is added to your catalog, and the system displays the record’s Details page.
4. Complete the information on the Details page, and click Next.

<table>
<thead>
<tr>
<th>Form Item</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Application Name</td>
<td>Provide the name of the application.</td>
</tr>
<tr>
<td>Description</td>
<td>(Optional) Provide a description of the application.</td>
</tr>
<tr>
<td>Icon</td>
<td>(Optional) Click Choose File to upload an icon for the application.</td>
</tr>
<tr>
<td></td>
<td>Workspace supports PNG, JPG, and ICON file formats up to 4MB. Workspace resizes uploaded icons to 80px X 80px. To prevent distortion, upload icons where the height and width are equal to each other and as close as possible to the 80px X 80px resize dimensions.</td>
</tr>
</tbody>
</table>

Authentication Profile Specify the appropriate federation protocol, if any.

After clicking Next, the Configuration page appears.

5. Edit the application record’s configuration details as necessary, and click Save.

Some of the items on the form are prepopulated.

When the SAML 2.0 POST Profile is selected on the Details page, the Configuration page includes the Configure Via section. Use the options in the Configure Via section to specify how the application metadata is retrieved. You can select retrieval by auto-discovery URL, meta-data XML, or manual configuration.

<table>
<thead>
<tr>
<th>Option</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>Auto-discovery (meta-data) URL</td>
<td>If the XML metadata is accessible on the Internet, provide the URL.</td>
</tr>
<tr>
<td>Meta-data XML</td>
<td>If the XML metadata is not accessible on the Internet, but is available to you, paste the XML in the text box.</td>
</tr>
<tr>
<td>Manual configuration</td>
<td>If the XML metadata is not available to you, complete the XML manual configuration items.</td>
</tr>
</tbody>
</table>
Select the Entitlements, Licensing, and Provisioning tabs and customize the information as appropriate.

<table>
<thead>
<tr>
<th>Tab</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Entitlements</td>
<td>Entitle users and groups to the application. You can configure entitlements while initially configuring the application or anytime in the future.</td>
</tr>
<tr>
<td>Access Policies</td>
<td>Apply a Web-application-specific access policy set to control user access to the application.</td>
</tr>
<tr>
<td>Licensing</td>
<td>Configure license tracking. Add license information for the application to track license usage in reports.</td>
</tr>
<tr>
<td>Provisioning</td>
<td>Select a provisioning adapter. Workspace ships with the provisioning adapters for the Google Apps and Mozy Web applications. If you are configuring either of these applications, you can select the appropriate provisioning adapter. Provisioning provides automatic application user-management from a single location. Provisioning adapters allow the Web application to retrieve specific information from Workspace as required. For example, to enable automatic user provisioning to Google Apps, user account information, such as user ID, first name, and last name must exist in the Google Apps database. Other information, such as group-membership and authorization-role information might be required by an application.</td>
</tr>
</tbody>
</table>

**What to do next**

See “Entitle Users and Groups to Web Applications,” on page 17 for details about adding user and group entitlements for Web applications.

**Add a Web Application to Your Catalog by Importing a ZIP or JAR File**

You can import to your catalog a Web application that was previously configured in another Workspace instance, for example when moving from a staging system to a production system.

This process involves exporting the application bundle of a Web application from a Workspace instance and importing the bundle to another Workspace instance. Because you import the Web application from a Workspace deployment, the application might not require further configuration, especially if you thoroughly tested the configuration values in the original deployment. To further configure the Web application after importing it, see “Add a Web Application to Your Catalog from the Cloud Application Catalog,” on page 13 or “Add a Web Application to Your Catalog by Creating a New Application Record,” on page 15.

**Procedure**

1. Log in to the Workspace Admin Console of the Workspace instance from which to export a Web application.
2. Click the Catalog tab.
3. Click Any Application Type > Web Applications.
4. Click the icon of the Web application to export.
5. Click Export this Application.
6. Click Export.
7. Save the zipped application bundle to your local system.
8. Log in to the Workspace Admin Console of the Workspace instance to which to import the Web application.
9. Click the Catalog tab.
10. Click + Add Application > Web Application ...import a zip or jar file.
11 Browse to the location on your local system where you saved the compressed application bundle as a ZIP file, select the file, and click Submit.

12 Edit the information on the Details, Configuration, Entitlements, Access Policies, Licensing, and Provisioning pages as necessary.

What to do next
For details about adding user and group entitlements for Web applications, see “Entitle Users and Groups to Web Applications,” on page 17.

Entitle Users and Groups to Web Applications
You can entitle users and groups to Web applications.

You can only entitle Workspace users, users who are imported from your directory server, to Web applications. When you entitle a user to a Web application, the user sees the application and can launch it from their Workspace App Portal. If you remove the entitlement, the user cannot see or launch the application.

In many cases, the most effective way to entitle users to Web applications is to add a Web application entitlement to a group of users. However, in certain situations entitling individual users to a Web application is more appropriate.

Procedure
1 Log in to the Workspace Admin Console.
2. Entitle users to a Web application.

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
</table>
| Access a Web application and entitle users or groups to it. | a. Click the Catalog tab.  
b. Click Any Application Type > Web Applications.  
c. Click the Web application to which to entitle users and groups.  
The information page for the Web application appears with the Entitlements tab selected by default. Group entitlements are listed in one table, user entitlements are listed in another table.  
d. Click Add group entitlement or Add user entitlement.  
e. Type the names of the groups or users.  
    You can search for users or groups by starting to type a search string and allowing the autocomplete feature to list the options, or you can click browse to view the entire list.  
f. Use the drop-down menu to select how to activate each selected Web application.  
    - Automatic displays the application by default in an entitled user’s list of Web applications the next time that user logs in using their Workspace client.  
    - User-Activated requires that an entitled user must add the Web application to their list of Web applications using their Workspace client before the user can use the application.  
g. Click Save.  |
| Access a user or group and add Web application entitlements to that user or group. | a. Click the Users & Groups tab.  
b. Click the Users or Groups tab.  
c. Click the name of a user or group.  
d. Click Add Entitlement.  
e. Select the check boxes next to the Web applications to which you want to entitle the user or group.  
f. Use the drop-down menu to select how to activate each selected Web application.  
    - Automatic displays the application by default in an entitled user’s list of Web applications the next time that user logs in using their Workspace client.  
    - User-Activated requires that an entitled user must add the Web application to their list of Web applications using their Workspace client before the user can use the application.  
g. Click Save.  |

The selected user or group is now entitled to use the Web application.
Providing Access to View Desktop and Application Pools

By integrating your organization’s View™ Connection Server instance with your Workspace system, you give your VMware Workspace Portal users the ability to use the Workspace App Portal to access their entitled View desktop and applications pools. Additionally, when the View module is enabled, you can use the Workspace admin console to see the associations between Workspace users and groups and their entitled View pools.

**Note** You use the View Connection Server instance and its associated View Administrator management Web interface to entitle users and groups to View desktop and application pools. See the View documentation.

This chapter includes the following topics:

- “Integrating View,” on page 19
- “Reducing Resource Usage and Increasing Performance of Workspace for Windows In Non-Persistent View Desktops,” on page 25

**Integrating View**

To use View with Workspace, you must join the Active Directory domain and sync with the View Connection Server.

You create and configure View pools in View, not in Workspace.

**Prerequisites**

- Verify that View is installed. For information about specific View versions that are supported by Workspace, see the VMware Product Interoperability Matrixes at http://www.vmware.com/resources/compatibility/sim/interop_matrix.php.
- Deploy and Configure View. You deploy View connection server on the default port 443 or on a custom port.
- Deploy and configure View pools and desktops with entitlements set for Active Directory users and groups.
- Ensure that you create View pools in the root folder of View. If you create View pools in a folder other than the root folder, Workspace cannot query those View pools and entitlements.
- Complete the information on the Join Domain page of the Connector Services Admin to join Workspace to the Active Directory domain.
- Enable the UPN attribute on Workspace on the User Attributes page.
- Configure SAML authenticator on the View Connection Server. You must always use the Workspace FQDN on the Authenticator configuration page.
- Verify that you have a DNS entry and an IP address that can be resolved during reverse lookup for each View Connection Server in your View setup. Workspace requires reverse lookup for View Connection Servers, View Security server, and load balancer. If reverse lookup is not properly configured, the Workspace integration with View fails.

- Sync Active Directory users and groups who are entitled to View pools in View Connection server to Workspace. To sync users and groups, in the Connector Services Admin click the Directory Sync tab and edit the directory sync rules.

Configuring View for Workspace includes some of the following tasks.

### Join an Active Directory Domain

To use View with Workspace, Workspace must join the Active Directory domain and sync with the View Connection Server.

During the setup process, you will be prompted to enter information for Workspace to join the Active Directory domain.

**Prerequisites**

- Verify that you have an Active Directory domain name, username, and password of an account in that Active Directory that has the rights to join the domain.
- Verify that the attribute userPrinciplaName in the Workspace Map User Attributes page is enabled.
- Verify that users and groups with View Pool entitlements assigned are synced using Directory sync.
- If applicable, establish a connection to multi-domains or trusted multi-forest domains in Active Directory. See *Installing and Configuring Workspace*.

**Procedure**

1. Log in to the Connector Services Admin.
2. In the Advanced tab, select **Join Domain**.
3. Type the information for the Active Directory domain and click **Join Domain**. Do not use non-ASCII characters when you enter your domain information.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
</table>
| **Active Directory** | Type the fully qualified domain name of the Active Directory. An example is **HS.TRDOT.COM**.  
**Note** The active directory FQDN must be in the same domain as the View Connection Server. Otherwise, your deployment fails. |
| **AD Username**  | Type the username of an account in the Active Directory that has permissions to join systems to that Active Directory domain. |
| **AD Password**  | Type the password associated with the **AD Username**. This password is not stored by Workspace. |

4. To configure View integration in a multi-domain environment, verify that Workspace and the View servers are joined to the same domain.

**What to do next**

Sync View with Workspace to propagate changes you make in View.
Add View Pods to Workspace and Sync Resources

You can add multiple View pod instances from the same Active Directory instance to Workspace on the View Pools page in the Connector Services Admin. Next you configure client access URLs for the different pods in the Workspace Admin Console, Settings > Network Ranges page.

You integrate Workspace with View by configuring the View Pools page in the Connector Services Admin. You can return to the page at anytime to modify the View configuration, such as to add or remove View pods.

Prerequisites
Verify that your Workspace system is integrated with your View system.

Procedure
1. Log in to the Connector Services Admin.
2. Click View Pools in the Navigation pane.
3. To permit access to the View Pools from Workspace, check the box Enable View Pools.
4. Click Add View Pod for each View pod you want to add.
5. Provide the configuration information specific to each View pod.

<table>
<thead>
<tr>
<th>Connection Server</th>
<th>Enter the name of the View Connection Server for this View pod.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Username</td>
<td>Enter the administrator username for this View pod.</td>
</tr>
<tr>
<td>Password</td>
<td>In the Password text box, enter the administrator password for this View pod.</td>
</tr>
<tr>
<td>Using Smart Card Authentication with Third-Party Identity Provider</td>
<td>If users use smart card authentication to sign in to this View pod instead of passwords, enable the check box.</td>
</tr>
</tbody>
</table>

6. (Optional) To automatically import newly added resource entitlements from View to Workspace, select the Perform Directory Sync check box.

If you do not select the Perform Directory Sync check box, you must separately perform a directory sync to import newly added resource entitlements.

7. From the Deployment Type drop-down menu, select the type of deployment Workspace uses to extend View resource entitlements to users.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>User-Activated</td>
<td>Workspace adds View resources to the App Center in the Workspace App Portal. To use the resource, users must move the resource from the App Center to their My Apps portal.</td>
</tr>
<tr>
<td>Automatic</td>
<td>Workspace adds the resource directly to users' My Apps portal for their immediate use.</td>
</tr>
</tbody>
</table>

8. Select how often you want this information to sync from the View Connection Server.

9. Click Save.

10. Click Sync Now.

Each time you change information in View, such as add an entitlement, add a user, and so on, a sync is required to propagate the changes to Workspace.
Log in to the Workspace Admin Console to finish the View Pod configuration.

a. Select **Settings > Network Ranges**.

b. Select an existing network range and click **Edit**.

c. In the View Pods section of the form, enter the View Pod client access URL host name and port number for the network range.

### Configure SAML Authentication

If you want to launch a View desktop from Workspace and use SSO, configure SAML authentication in the View server.

Do not perform this task if your organization uses smart card authentication to view resources using a third-party identity provider.

**Procedure**

1. Log in to the View Administrator Web interface as a user with the Administrator role assigned.
2. Configure SAML authentication for each replicated server in your View infrastructure.

---

**IMPORTANT**  View and Workspace must be in time sync. If View and Workspace are not in time sync, when you try to launch View desktop, an invalid SAML message occurs.

---

### What to do next

You must establish and maintain SSL Trust between Workspace and the View Connection Server.

### Establish or Update SSL Trust between Workspace and the View Connection Server

Initially, you must accept an SSL certificate on the View Connection server to establish trust between Workspace and the View Connection server. If you change an SSL certificate on the View Connection server after the integration, you must return to Workspace and reestablish that trust.

**Prerequisites**

- Verify that View has an SSL certificate installed. By default, View has a self-signed certificate.
- In View, change the certificate of the View Connection Server to a root-signed certificate. See the VMware View documentation for information about configuring a View Connection server instance or Security Server to use a new certificate.
- Configure SAML authentication on the View Connection server. You must always use the Workspace FQDN on the authenticator configuration page.

**NOTE**  If you use a third-party identity provider to access View desktops from Workspace, SAML authentication, on the View Connection server, must be set to **allowed**.

**Procedure**

1. Log in to the Connector Services Admin.
2. Open the View Pools page.
3. Click the **Update SSL Cert** link next to the Replicated Server Group.
4. Click **Accept** on the Certificate Information page.

If the Workspace certificate changes after the initial configuration, you must accept the SAML Authenticator from View again. If the View certificate changes, you must accept the SSL certificate in Workspace.
View the Connection Information for a View Desktop and Application Pools

You can view the information about the connection between Workspace and a View desktop or application pool.

Procedure

1. Log in to the Workspace Admin Console.
2. Click the Catalog tab.
3. Click Any Application Type > View Desktop Pool to view desktop pools. Click View Hosted Applications to view application pools.
4. Click the name of a View application.
5. Click the Details tab.
6. View the connection information, which consists of attributes retrieved from the View Connection Server instance.
   See the View documentation for details about these attributes.

View User and Group Entitlements to View Desktop and Application Pools

You can see the View pools to which your Workspace users and groups are entitled.

Prerequisites

- Verify that your Workspace system is integrated with your View system.
- Synchronize information and the respective entitlements from the View Connection Server instances to your Workspace system. You can force a sync on the View Pools page in the Connector Services Admin, by clicking Sync Now.

Procedure

1. Log in to the Workspace Admin Console.
2. View user and group entitlements to View desktop and application pools.

<table>
<thead>
<tr>
<th>Option</th>
<th>Action</th>
</tr>
</thead>
</table>
| List users and groups entitled to a specific View desktop pool. | a. Click the Catalog tab.  
   b. Click Any Application Type > View Desktop Pools or View Hosted Applications.  
   c. Click the icon for the View pool for which you want to list entitlements.  
   The Entitlements tab is selected by default. Group entitlements and user entitlements are listed in separate tables. |
| List of View desktop and application pool entitlements for a specific user or group. | a. Click the Users & Groups tab.  
   b. Click the Users tab or the Groups tab.  
   c. Click the name of an individual user or group.  
   The Entitlements tab is selected by default. Entitled View desktop and application pools, if any, are listed in the View Pools tables on the Entitlements page. |
Enable Multiple View Client URLs Access to Custom Network Ranges

If your company uses multiple client access URLs for different network ranges, the administrator must edit the default network range so the end user connects to the correct View Client Access URL and port number. If these settings are not updated, the View client will not launch.

Procedure
1. Log in to the Workspace Admin Console.
2. Click the Settings tab.
3. Click Network Ranges in the left navigation.
4. Click the Edit link by each network range.
5. Type in the information for View Client Access URL Host and Client Access URL Port using your company’s configuration.
6. Verify that each network range in your environment contains a View Client Access URL.

IMPORTANT If you miss a network range, end users who launch through that network range might have problems.

What to do next
If necessary, modify the View integration configuration.

Launch a View Pool

Users can launch a View pool from Workspace.

You can switch the display protocol between Open with View Client or Open with Browser by clicking Preferences from the drop-down.

Prerequisites
Install View Client. You must install View Client on the machine that launches Workspace.

NOTE For information about specific View Client versions, see the VMware Product Interoperability Matrixes at http://www.vmware.com/resources/compatibility/sim/interop_matrix.php.

Procedure
1. Log in to your Workspace instance.
2. Click the View Desktops icon.
3. Select your View pool.
4. Right-click the selected View pool and choose a protocol to launch the View desktop.

What to do next
If applicable, you can allow users to reset their View desktops in Workspace.

Allow Users to Reset Their View Desktops in Workspace

Depending on how you configure View and Workspace, users can use the Workspace App Portal to reset an unresponsive View desktop.

When you configure View to allow users to reset their desktops, the configuration applies to both View and Workspace.
Prerequisites

- Configure View to allow users to reset their desktops. See documentation for VMware Horizon with View, specifically the guide View Administration.

- To ensure that specific View Desktops are resettable by users, the client access URLs for the respective pods should have trusted certificates. If the URLs have root-signed or self-signed certificates, configure Workspace to trust those certificates. See Installing and Configuring Workspace for information about applying a Workspace root certificate.

Procedure

- (Optional) Verify that Workspace lists a given desktop as resettable by users.
  
  a. Select the Catalog tab.
  
  b. In the Any Application Type drop-down menu, click View Desktop Pools.
  
  c. Click the name of the desktop.
  
  d. Click Details.
  
  e. Confirm that the Reset allowed setting is set to true.

  If the setting is false, then View is not configured to allow users to reset the desktop.

What to do next

If a View desktop becomes unresponsive in the future, you or users can reset the desktop in the Workspace App Portal by right-clicking an unresponsive desktop in the My Apps portal and clicking Reset Desktop.

Reducing Resource Usage and Increasing Performance of Workspace for Windows In Non-Persistent View Desktops

To reduce resource usage and increase performance when using the Workspace App Portal in non-persistent desktops, also known as stateless desktops, you can configure the client with settings optimized for using it in a non-persistent View desktop.

Problem

When a non-persistent View desktop has the Workspace for Windows application installed in the View desktop, each time a user starts a session, an increased amount of resources are used, such as storage I/Os.

Cause

Non-persistent View desktops are inherently stateless. Such View desktops are also known as floating desktops, and new sessions can be created when the floating desktops are recomposed or the user is given a new desktop from the pool. Unless the Workspace for Windows application used in the non-persistent desktops is configured with settings that are optimized for this scenario, the users might experience degraded performance when ThinApp packages.

Solution

- Install the Workspace for Windows application in the template that is used for the non-persistent View desktops using the recommended command-line installer options.

<table>
<thead>
<tr>
<th>/v Installer Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ENABLE_AUTOUPTDATE = 0</td>
<td>Prevents the automatic update of the Workspace for Windows application to a newer version. Typically, your View administrator updates the application in the template.</td>
</tr>
<tr>
<td>INSTALL_MODE = RUN_FROM_SHARE</td>
<td>If you plan to have the users use ThinApp packages in these View desktops, use this option to have the ThinApp packages streamed from the server instead of downloaded to the Windows system.</td>
</tr>
</tbody>
</table>

This code sample is an example of installing the Workspace for Windows application with an optimal configuration for non-persistent View desktops where the users are expected to use ThinApp packages. The HORIZONURL option specifies the Workspace server for this installation.

`Workspace-n.n.n-nnnnnnn.exe /v HORIZONURL=https://server.company.com ENABLE_AUTOUPTDATE=0 INSTALL_MODE=RUN_FROM_SHARE`
Providing Access to VMware ThinApp Packages

With Workspace, you can centrally distribute and manage ThinApp packages. ThinApp packages are virtualized Windows applications, and are used on Windows systems. Entitled users who have the Workspace for Windows application installed on their Windows systems can launch and use their entitled ThinApp packages on those Windows systems.

In the ThinApp capture and build processes, you create a virtual application from a Windows application. That virtualized Windows application can run on a Windows system without that system having the original Windows application installed. The ThinApp package is the set of virtual application files generated by running the ThinApp capture and build processes on a Windows application. The package includes the primary data container file and entry point files to access the Windows application.

Not every ThinApp package is compatible with Workspace. When you capture a Windows application, the default settings in the ThinApp capture-and-build process create a package that Workspace cannot distribute and manage. You create a ThinApp package that Workspace can distribute and manage by setting the appropriate parameters during the capture and build processes. See the VMware ThinApp documentation for detailed information on ThinApp features and the appropriate parameters to use to create a package compatible with Workspace.

After you integrate your Workspace system with your ThinApp repository, you can see in your catalog those ThinApp packages from the repository that Workspace can distribute and manage. After you see the ThinApp packages in your Workspace catalog, you can entitle users and groups to those ThinApp packages, and optionally configure license tracking information for each package.

This chapter includes the following topics:

- “Integrating VMware ThinApp Packages,” on page 28
- “Entitle Users and Groups to ThinApp Packages,” on page 35
- “Distributing and Managing ThinApp Packages with Workspace,” on page 37
- “Using the Command-Line HorizonThinAppCtrl.exe Application,” on page 40
- “Updating Managed ThinApp Packages After Deployment in Workspace,” on page 41
- “Delete ThinApp Packages from Workspace,” on page 46
- “Make Existing ThinApp Packages Compatible with Workspace,” on page 47
- “Change the ThinApp Packages Share Folder,” on page 49
Integrating VMware ThinApp Packages

To use Workspace to distribute and manage applications packaged with VMware ThinApp, you must have a ThinApp repository that contains the ThinApp packages, point your Workspace system to that repository, and sync the packages. After the sync process is finished, the ThinApp packages are available in your Workspace catalog and you can entitle them to your Workspace users and groups.

ThinApp provides application virtualization by decoupling an application from the underlying operating system and its libraries and framework and bundling the application into a single executable file called an application package. To be managed by Workspace, these packages must be enabled with the appropriate options. For example, in the ThinApp Setup Capture wizard, you select the Manage with Workspace check box. For more information about ThinApp features and how to enable your applications for management by Workspace, see the VMware ThinApp documentation.

Typically, you perform the steps to connect your Workspace system to the repository and sync the packages as part of the overall setup and configuration of your Workspace environment. The ThinApp repository must be a network share that is accessible to your Workspace using a Uniform Naming Convention (UNC) path. Workspace synchronizes with this network share regularly to obtain the ThinApp package metadata that the Workspace system needs to be able to distribute and manage the packages. See “Workspace Requirements for ThinApp Packages and the Network Share Repository,” on page 28.

The network share can be a Common Internet File System (CIFS) or a Distributed File System (DFS) share. The DFS share can be a single Server Message Block (SMB) file share or multiple SMB file shares organized as a distributed file system. CIFS and DFS shares running on NetApp storage systems are supported.

Workspace Requirements for ThinApp Packages and the Network Share Repository

When you capture and store ThinApp applications to distribute from Workspace, you must meet certain requirements.

Requirements on the ThinApp Packages

To create or repackage ThinApp packages that Workspace can manage, you must use a version of ThinApp that Workspace supports. For information about specific ThinApp versions that are supported by Workspace, see the VMware Product Interoperability Matrixes at http://www.vmware.com/resources/compatibility/sim/interop_matrix.php.

You must have ThinApp packages that Workspace can manage. In the ThinApp capture-and-build process, you can create packages that Workspace can manage or ones that it cannot manage. For example, when you use the ThinApp Setup Capture wizard to capture an application, you can make a package that Workspace can manage by selecting the Manage with Workspace check box. See the VMware ThinApp documentation for detailed information on ThinApp features and the appropriate parameters to use to create a package compatible with Workspace.

For existing ThinApp packages, you can use the relink -h command to enable the packages for Workspace. For information about how to convert existing ThinApp packages to packages that Workspace can manage, see the VMware Workspace Portal Administrator’s Guide.

You must store the ThinApp packages on a network share that meets the requirements for the combination of network share type, repository access, and desired ThinApp package deployment mode for your organization’s needs.
Requirements on the Network Share Repository

The ThinApp packages must reside on a network share, also known as the ThinApp package repository. The network share must be accessible using a Uniform Naming Convention (UNC) path from each system running the Workspace client used to access the ThinApp packages. For example, a network share named appshare on a host named server is accessible using the UNC path \server\appshare. The fully qualified hostname of the network share folder must be resolvable from your Workspace.

The network share can be a Common Internet File System (CIFS) or a Distributed File System (DFS) share. The DFS share can be a single Server Message Block (SMB) file share or multiple SMB file shares organized as a distributed file system. CIFS and DFS shares running on NetApp storage systems are supported.

The network share must meet the criteria appropriate for the type of access you configure Workspace to use for accessing the ThinApp package repository: domain-based access or account-based access. The type of access determines the allowable combinations for the following items:

- Whether you use a CIFS network share or a DFS network share for the ThinApp package repository.
- Whether you must join Workspace and the network share's host to the same Active Directory domain.
- Whether the user's Windows system must join the Active Directory domain to use the ThinApp packages.
- The ThinApp package installation mode that the installed Windows client program is set to use for obtaining and running the virtualized applications on the Windows system on which the client is installed. The package installation mode that is used on the user's Windows system is set during the installation process when the Windows client is installed on that Windows system. This package installation mode determines the mode of ThinApp deployment used by that Windows system, download mode or streaming mode.
<table>
<thead>
<tr>
<th>Access Type</th>
<th>Network Share Type</th>
<th>Requirements on Workspace</th>
<th>Requirements for the User’s Windows System</th>
</tr>
</thead>
</table>
| Domain-based access | You can use a CIFS share for your ThinApp package repository when you use domain-based access. | You must join Workspace to the Active Directory domain so that Workspace can join the Windows network share and access the packages. For more information about how to configure your Workspace to join the domain, see information about configuring Kerberos in *Installing and Configuring Workspace*. **NOTE** Windows authentication is not required. The network share must support authentication and file permissions that are based on computer accounts. Workspace accesses the network share with the computer account of Workspace in the domain. The network share's folder and file permissions must be configured such that the combination of permissions allows read access for the computer account of Workspace in the domain. | The user’s Windows system must join the Active Directory domain before that user can use their entitled ThinApp packages. The following systems must all be joined to the same domain:  
  - The user’s Windows system  
  - Your Workspace system  
  - The host of the network share drive with the ThinApp packages  
When you use domain-based access, the following installation modes for the ThinApp packages are allowed.  
  - COPY_TO_LOCAL. With this installation mode, packages are downloaded to the client Windows system. This installation mode corresponds to using the ThinApp download mode for the virtualized application. The account that is used to log in to the client Windows system is the user account that is used to copy the packages from the network share to the client Windows system, and that account must have permissions to read the packages and copy the files from that network share. After the package is downloaded to the client Windows system and the user launches the package, the virtualized application runs locally on the client Windows system.  
  - RUN_FROM_SHARE. With this installation mode, packages are not downloaded to the client Windows system. A user launches the packages using shortcuts on the local desktop and the virtualized applications run from the network share using ThinApp streaming mode. The account that is used to log in to the client Windows system is the user account that is used to run the packages from the network share, and that account must have permissions to read and execute files from that network share.  
  **NOTE** RUN_FROM_SHARE is best suited for Windows systems that will always have connectivity to the ThinApp packages’ network share. Windows systems that best fit that description are Workspace desktops, because they are always connected to their domain. Floating, or stateless, Workspace desktops best use RUN_FROM_SHARE to avoid the resource usage inherent in downloading the packages to the Windows system.  
By default, the COPY_TO_LOCAL installation mode is set as the default installation mode when you install the Workspace for Windows client application on a Windows system by running the graphical version of the client’s installer program. To set a different installation mode as the default installation mode for the packages, you must run the command-line version of the client’s installer program. See the *VMware Workspace Portal Administrator’s Guide*. |
| Account-based access | You can use either a CIFS share or a DFS share for your ThinApp package repository | You must configure your Workspace system to use a share user account and password to access the packages | The user’s Windows system does not have to join the Active Directory domain before that user can use their entitled ThinApp packages. Windows authentication is not required. |

**NOTE**
In addition, the ThinApp packages repository must meet the following criteria according to the described situation.

- When your settings involve systems joining the Active Directory domain, make sure that a disjoint namespace does not prevent domain member computers from accessing the network share that hosts the ThinApp packages. A disjoint namespace occurs when an Active Directory domain name is different from the DNS namespace that machines in that domain use.

- The network share’s file and sharing permissions must be configured to provide read access and the ability to run applications to those users that you want to run the ThinApp applications using the COPY_TO_LOCAL or RUN_FROM_SHARE option.

For example, for the Active Directory user accounts of those users that you want to run the ThinApp applications in streaming mode, setting the Shared Folder permission to Read and the NTFS permission to Read & Execute provides read access and the ability to run the applications to those users.
The NTFS permission setting of **Read & Execute** is required to run a ThinApp application using the ThinApp streaming mode, which corresponds to the Workspace for Windows client’s RUN_FROM_SHARE installation mode. If your organization requires the NTFS permission set to **Read**, your users can use the ThinApp download mode for the virtualized application. ThinApp download mode corresponds to installing the Windows client with either the COPY_TO_LOCAL installation mode or HTTP_DOWNLOAD installation mode. With either of those installation modes, the applications are downloaded to the Windows systems and launched locally.

Both CIFS and DFS network shares must have the ThinApp packages organized in individual subdirectories in a directory under the namespace, not subdirectories in the namespace itself, such as `\server\appshare\thinapp1`, `\server\appshare\thinapp2`, and so on. See “Create a Network Share for ThinApp Packages That Workspace Manages,” on page 32.

**Create a Network Share for ThinApp Packages That Workspace Manages**

If you want to enable the VMware® ThinApp® management capabilities of Workspace and allow users to access ThinApp packages from the user application catalog, you must create a network share and store the ThinApp packages in that network share folder.

Workspace obtains the metadata it needs about the ThinApp packages from the network file share.

**Prerequisites**

- Verify that the ThinApp packages meet Workspace requirements.
- Verify that you have the appropriate access and permissions to create a network file share in your IT environment that meets Workspace requirements for ThinApp packages.

**Procedure**

1. Create a network share that meets the Workspace requirements for ThinApp packages.
2. In the network share, create a network share subfolder for each ThinApp package.
   
   Typically, you name the subfolder to match the name of the ThinApp application, or indicate what application is in the folder. For example, if the network share is named `appshare` on a host named `server`, and the application is called `abceditor`, the subfolder for the ThinApp package is `\server\appshare\abceditor`.

   **Note** Do not use non-ASCII characters when you create your network share subfolder names for ThinApp packages to distribute by using Workspace. Non-ASCII characters are not supported.

3. For each ThinApp package, copy its files, such as its EXE and DAT files, to the subfolder that is named for that package’s virtualized application.
   
   After copying the files, you have a set of subfolders and files that are similar to these files:

   - `\server\appshare\abceditor\abceditor.exe`
   - `\server\appshare\abceditor\abceditor.dat`

**What to do next**

Configure your Workspace system’s access to the ThinApp packages.
Configuring Workspace Access to ThinApp Packages

To configure Workspace to provide users access to ThinApp packages, you must enable your Workspace system to locate the stored ThinApp packages and sync the packages with your system.

Prerequisites

- Create a network share with the appropriate configuration and store the ThinApp packages in the appropriate location in that network share. See “Create a Network Share for ThinApp Packages That Workspace Manages,” on page 32.

- Verify that you have the UNC path to the network share folder where the ThinApp packages are located.

- Verify that you have an Active Directory domain name and the username and password of an account in that Active Directory that has the rights to join the domain. Even if you are using account-based access, the Connector Services Admin requires the completion of the Join Domain page before you can use the ThinApp Packages page in the Connector Services Admin.

To enable domain-based access, you must also join Workspace to the same Active Directory domain to which the ThinApp package repository is joined. Verify that you have the Active Directory domain name for the domain that the network share uses and the username and password of an account in that Active Directory that has the rights to join the domain. The Active Directory account is used to join Workspace to the domain.

- When enabling account-based access, verify that you have a username and password that has permission to read the network share. See “Workspace Requirements for ThinApp Packages and the Network Share Repository,” on page 28.

**NOTE** Unless you want to restrict use of the ThinApp packages to domain-joined Windows systems for all runtime situations, you should enable account-based access in addition to domain-based access. This combination provides the most flexibility for supporting runtime situations where users need to use their entitled ThinApp packages without joining their Windows systems to the domain.
### Procedure

1. Join Workspace to an Active Directory domain.
   
   a. Log in to the Connector Services Admin.
   
   b. Select the **Join Domain** tab.
   
   c. On the Join Domain page, type the information for the Active Directory domain and click **Join Domain**.

   **IMPORTANT** Do not use non-ASCII characters when you enter the Active Directory (AD) domain name, AD username, or AD password. Non-ASCII characters are not supported in these entry fields in the Connector Services Admin.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>AD Domain</strong></td>
<td>Type the fully qualified domain name of the Active Directory. An example is <strong>HS.TRDOT.COM</strong>.</td>
</tr>
<tr>
<td><strong>AD Username</strong></td>
<td>Type the username of an account in the Active Directory that has permissions to join systems to that Active Directory domain.</td>
</tr>
<tr>
<td><strong>AD Password</strong></td>
<td>Type the password associated with the <strong>AD Username</strong>. This password is not stored by Workspace.</td>
</tr>
</tbody>
</table>

**IMPORTANT** Each time you import the Workspace configuration you must rejoin Workspace to the domain.

The Join Domain page refreshes and displays a message that you are currently joined to the domain.
2. Enable your Workspace system to access the stored ThinApp packages.
   a. Select the Packaged Apps - ThinApp tab.
   b. Select the Enable packaged applications check box.
   c. Complete the information and click Save.

   **IMPORTANT** Do not use non-ASCII characters for entries in the fields on this page. Non-ASCII characters are not supported in these entry fields in the Connector Services Admin.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Path</td>
<td>Type the path to the shared folder where the ThinApp packages' folders are located, in the UNC path format <code>\server\share\subfolder</code>. For example: <code>\DirectoryHost\ThinAppFileShare</code>. For <code>DirectoryHost</code>, provide the hostname, not the IP address. For both CIFS and DFS network shares, this path must be a directory under the namespace, and not the namespace itself.</td>
</tr>
<tr>
<td>Choose Frequency</td>
<td>Select the interval at which you want Workspace to synchronize the information about the ThinApp packages located at the network share location with Workspace. For a weekly interval, set the day and time of day at which the syncing occurs. For a daily interval, set the time.</td>
</tr>
<tr>
<td>Enable account based access</td>
<td>Select this option if you want to use account based access. <strong>Note</strong>: If your ThinApp package repository is a DFS network share, you must select this option. If you want the ability for users to use their entitled ThinApp packages on non-domain-joined Windows systems, you must enable account based access.</td>
</tr>
<tr>
<td>Share User</td>
<td>Type the username for a user account that has read access to the network share. This information is used when Enable account based access is selected.</td>
</tr>
<tr>
<td>Share Password</td>
<td>Type the password associated with the Share User user account.</td>
</tr>
</tbody>
</table>

A message appears stating that the values are saved, and a summary of the last sync status is displayed.

3. Sync the ThinApp packages with your Workspace system by clicking Sync Now.

   The time it takes to complete the sync process depends on the number of ThinApp packages.

   When the sync process is done, a list of the ThinApp packages that were synced appears.

   Your Workspace system is configured so that you can entitle groups and users to ThinApp packages, and those users can run their entitled ThinApp packages using the Workspace client installed on their Windows systems.

   **What to do next**

   Entitle groups and users to ThinApp packages. See the *VMware Workspace Portal Administrator’s Guide*.

### Entitle Users and Groups to ThinApp Packages

You can entitle users and groups to Windows applications that are captured as ThinApp packages.

You can only entitle Workspace users, users who are imported from your directory server, to ThinApp packages. When you entitle a user to a ThinApp package, the user sees the application and can start it from the Workspace for Windows application on their system. If you remove the entitlement, the user cannot see or start the application.

Often, the most effective way to entitle users to ThinApp packages is to add a ThinApp package entitlement to a group of users. In certain situations entitling individual users to a ThinApp package is more appropriate.
Prerequisites

Configure Workspace to sync ThinApp packages to your Workspace catalog. When the ThinApp packages are synced to your catalog, you can entitle them to your users and groups.

Use the Connector Services Admin to sync ThinApp packages to your catalog. You cannot add ThinApp packages directly to your catalog from the Workspace Admin Console.

Procedure

1. Log in to the Workspace Admin Console.
2. Entitle users to a ThinApp package.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
</table>
| Access a ThinApp package and entitle users or groups to it. | a Click the Catalog tab.  
 | b Click Any Application Type > ThinApp Packages.  
 | c Click the ThinApp package to entitle users and groups to.  
 | The Entitlements tab is selected by default. Group entitlements are listed in one table, user entitlements are listed in another table.  
 | d Click Add group entitlement or Add user entitlement.  
 | e Type the names of the groups or users.  
 | You can search for users or groups by starting to type a search string and allowing the autocomplete feature to list the options. You can click browse to view the entire list.  
 | f From the drop-down menu, select the activation method for the ThinApp package.  
 | **Automatic** Users have immediate access to the ThinApp package the next time they log in to the Workspace client.  
 | **User-Activated** Users must activate the ThinApp package in Workspace for Windows before they can use the application.  
 | g Click Save. |

Access a user or group and add ThinApp package entitlements to that user or group. | a Click the Users & Groups tab.  
 | b Click the Users or Groups tab.  
 | c Click the name of an individual user or group.  
 | d Click Add entitlement.  
 | e Click the check boxes next to the ThinApp packages to entitle the user or group to.  
 | f From the drop-down menu, select the activation method for the ThinApp package.  
 | **Automatic** Users have immediate access to the ThinApp package the next time they log in to the Workspace client.  
 | **User-Activated** Users must activate the ThinApp package in Workspace for Windows before they can use the application.  
 | g Click Save. |

The selected users or groups are now entitled to use the ThinApp package.

What to do next

Verify that the Workspace for Windows application is installed on users' Windows systems.
Distributing and Managing ThinApp Packages with Workspace

Before your Workspace users can run their ThinApp packages that are registered to them using Workspace, those users must have the Workspace for Windows application installed and running on their Windows systems.

ThinApp packages are virtualized Windows applications. The ThinApp packages are distributed to Windows systems, and a user logged into the Windows system can launch and run those ThinApp packages that are registered on that Windows system. Workspace can distribute and manage ThinApp packages that are compatible with Workspace.

To successfully launch and run one of these virtualized applications in the user’s logged-in Windows session, the following elements are required:

- The virtualized application’s ThinApp package is registered for that user’s use by the Workspace server.
- A particular DLL is available on that Windows system.
- The HorizonThinAppClient.exe process is running.

When a compatible ThinApp package is created, it is configured to load a particular DLL when the logged-in user launches the virtualized application in their logged-in Windows session. At that time, the virtualized application attempts to load the DLL. When the DLL is loaded, it attempts to verify with the locally installed Workspace ThinApp client whether that ThinApp package is registered on that Windows desktop for that user. The locally installed ThinApp client determines whether that application is registered for that user without communicating with the Workspace server. If the application is registered on that Windows desktop for that user, the ThinApp client checks to see when it last synced with the Workspace server. If the ThinApp client confirms that the time from the last sync is within the offline grace period configured for the installed client, the client allows the application to run.

Because that DLL is available on the Windows system only if the Workspace for Windows application is installed, and because the HorizonThinAppClient.exe process is running if the Workspace for Windows application is running on that system, the Workspace for Windows application must be installed on the Windows system to run ThinApp packages that are distributed and managed by your Workspace server.

Deploying the Workspace for Windows Application To Use ThinApp Packages

The Workspace for Windows application can be installed by either double-clicking its installer EXE file, running the executable file using the command-line options, or running a script that uses the command-line options. Local administrator privileges are required to install the application. For information about installing the Workspace for Windows application by double-clicking its installer EXE file, see the Workspace User Guide.

The configuration of the installed application determines how a ThinApp package that is distributed by Workspace is deployed to that Windows system. By default, when the Workspace for Windows application is installed by double-clicking its installer EXE file, the client is configured to deploy ThinApp packages using the COPY_TO_LOCAL deployment mode, with the AUTO_TRY_HTTP option enabled. Those default installer options result in what is called a download deployment mode. With the COPY_TO_LOCAL and AUTO_TRY_HTTP default settings, the client application first tries to download the ThinApp packages by copying them to the Windows system endpoint, and if the first attempt fails, the client application tries to download the ThinApp packages using HTTP. If Workspace is configured for account-based access to your ThinApp repository, the client application can download the ThinApp packages using HTTP. After the ThinApp packages are downloaded to the local Windows system, the user runs the virtualized applications on the local system.
To avoid having the virtualized applications downloaded to the local Windows system and using space on the Windows system, you can have users run the ThinApp packages from the network share by using what is called a streaming deployment mode. To have your users run the ThinApp packages using streaming mode, you must install the Workspace for Windows application on the Windows systems using a command-line installation process. The installer has command-line options that you can use to set the runtime deployment mode for the ThinApp packages. To set the runtime deployment mode to stream the ThinApp packages, use the RUN_FROM_SHARE installer option.

One method for installing the Workspace for Windows application to multiple Windows systems is to use a script to install the application silently to the Windows systems. You can install the client silently to multiple Windows systems at the same time.

NOTE A silent installation does not display messages or windows during the install process.

You set a value in the script to indicate whether the clients installed by that script deploy ThinApp packages using the ThinApp streaming mode, or RUN_FROM_SHARE option, or one of the ThinApp download modes, such as the COPY_TO_LOCAL or HTTP_DOWNLOAD option. See “Install the Workspace for Windows Application with Identical Settings to Multiple Windows Systems,” on page 57.

**Determining the Appropriate Deployment Mode for ThinApp Packages on Windows Endpoints**

The configuration of the Workspace for Windows application on the Windows endpoint determines whether a ThinApp package that is distributed using Workspace is deployed using ThinApp streaming mode, RUN_FROM_SHARE, or one of the ThinApp download modes, such as the COPY_TO_LOCAL or HTTP_DOWNLOAD. When you create the script to silently install Workspace for Windows to Windows endpoints, such as desktop and laptop computers, you set the options that set the ThinApp package deployment mode. Choose the deployment mode that best fits the network environment for the selected endpoints, considering details such as network latency.

With streaming mode, when the Workspace for Windows application synchronizes with Workspace, the client downloads application shortcuts for the ThinApp packages' virtualized Windows applications to the Windows desktop, and when the user launches the ThinApp packages, the virtualized Windows applications run from the file share on which the ThinApp packages reside. Therefore, streaming mode is appropriate for systems that will always be connected to the network share, such as View desktops. With download mode, at the first use or update of a ThinApp package, the user must wait for the ThinApp package to download to the Windows system first, and shortcuts to be created. After the initial download, the user launches and runs the virtualized Windows application on the local Windows system.

IMPORTANT For non-persistent View desktops, also known as floating or stateless View desktops, you are expected to set the client to use ThinApp streaming mode by using the command-line installer option /v INSTALL_MODE=RUN_FROM_SHARE when installing the client. The RUN_FROM_SHARE option provides the most optimal runtime experience for using ThinApp packages in floating View desktops. See “Command-Line Installer Options for Workspace for Windows,” on page 51.
### Table 5-1. ThinApp Deployment Mode for the Virtualized Applications Captured as ThinApp Packages

<table>
<thead>
<tr>
<th>Mode</th>
<th>Description</th>
</tr>
</thead>
</table>
| ThinApp streaming mode    | In ThinApp streaming mode, the virtualized applications are streamed each time they are started. This method avoids using disk space in the desktop that would be used when copying the virtualized applications to the desktop. The desktop must be connected to the ThinApp packages' network share for the applications to run. The following environments might provide the consistency and stability required:  
  - View desktops, either stateless or persistent, with excellent connectivity to the file share on which the ThinApp packages reside.  
  - Users with Windows desktops that are not View desktops, that are shared by multiple users. This situation avoids the accumulation on disk of downloaded user-specific applications and also provides quick access to applications without causing a delay for downloads specific to a user.  
  The account that the user uses to log in to the Windows system is used to obtain the ThinApp packages from the network share. That account must have the appropriate permissions on the network share to read and execute files on the network share. |
| ThinApp download mode     | In ThinApp download mode, applications are downloaded to the Windows endpoint. The user runs the virtualized application locally on the endpoint. You might prefer ThinApp download mode for the following situations:  
  - Persistent View desktops  
  - LAN-connected desktops that are periodically offline  
  - A LAN with poor network latency  
  Workspace provides two flavors of the ThinApp download mode: COPY_TO_LOCAL and HTTP_DOWNLOAD. If the client is configured for COPY_TO_LOCAL, the Windows endpoint must be joined to the same domain as the file share unless the AUTO_TRY_HTTP option is enabled and Workspace is configured for account-based access to the ThinApp packages' network share. When the AUTO_TRY_HTTP option is enabled and Workspace is configured for account-based access, if the Windows endpoint is not joined to the same domain and the first attempt to download the ThinApp packages fails, the Workspace for Windows client application will automatically try to download the ThinApp packages using the HTTP protocol as for the HTTP_DOWNLOAD mode. With HTTP_DOWNLOAD, the Windows endpoint does not have to be joined to the same domain as the file share. However, the copy and sync times when using HTTP_DOWNLOAD are significantly longer than when using COPY_TO_LOCAL.  
  **IMPORTANT** If Workspace is not enabled for account-based access, downloading using the HTTP protocol does not work, even if AUTOTRY_HTTP is enabled or the client is configured with the HTTP_DOWNLOAD option.  
  When using COPY_TO_LOCAL, the account that the user uses to log in to the Windows system is used to obtain the ThinApp packages from the network share. That account must have the appropriate permissions on the network share to read and copy files from the network share. When using HTTP_DOWNLOAD, the share user account that is entered in the Connector Services Admin when you configure your Workspace system's access to the ThinApp packages' network share is the account that is used to download the ThinApp packages. That share user account needs to have read permission on the ThinApp packages' network share to copy the files from the network share. |

The ThinApp packages' network share must meet the appropriate requirements for the deployment mode that you set for the Windows endpoints. See *Installing and Configuring Workspace*.

### Offline Grace Period and ThinApp Packages

The offline grace period is the period of time for which a virtualized application is allowed to launch and run on a Windows system without syncing with the Workspace server.

ThinApp packages are virtualized Windows applications, and Workspace can distribute these applications to Windows systems. When Workspace distributes a ThinApp package to the Windows system for the first time for the user logged in to that system, the package's virtualized applications are registered on that Windows system for that user's use. The appropriate shortcuts are added to the Windows desktop, and the user can launch the virtualized applications using the shortcuts as for standard Windows applications installed to that system.
When a user launches one of the virtualized applications that was deployed to the Windows system by Workspace, the ThinApp package requests permission to run from the ThinApp agent running on the system. The ThinApp agent verifies the following conditions.

- Verifies whether the application is registered on this Windows desktop for the logged-in user.
- Verifies whether the Windows system has synced with the Workspace server within the configured offline grace period.

If both of those conditions are true, the ThinApp agent allows the virtualized application to run.

The frequency of how often the Workspace for Windows application syncs with the Workspace server is set by the POLLINGINTERVAL installer option. By default, the frequency is every 5 minutes. The offline grace period is set to 30 days by default. If a Windows system has had network connectivity to connect to the Workspace server at any time within a 30-day timespan, the application can sync with the server and virtualized applications can run.

However, if the Windows system has no network connectivity to connect to the Workspace server, the application cannot sync with the server. Virtualized applications registered on that Windows system can run on the disconnected system up to the time set by the offline grace period.

**Set the Offline Grace Period for ThinApp Packages**

Setting the offline grace period for the ThinApp agent on a Windows system requires modifying the registry. The default offline grace period is 30 days, and is a system-wide setting.

**Prerequisites**

- Verify that Workspace for Windows is installed on the Windows system.
- Verify that you have local administrative permissions to modify the Windows system's registry.

**Procedure**

1. Open the Windows system's registry in the Registry Editor.
2. Locate the OfflineGracePeriod registry key.

<table>
<thead>
<tr>
<th>Windows System</th>
<th>Registry Key</th>
</tr>
</thead>
<tbody>
<tr>
<td>Windows 32-bit</td>
<td>HKEY_LOCAL_MACHINE\SOFTWARE\VMware, Inc. \Horizon ThinApp\OfflineGracePeriod</td>
</tr>
<tr>
<td>Windows 64-bit</td>
<td>HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\VMware, Inc. \Horizon ThinApp\OfflineGracePeriod</td>
</tr>
</tbody>
</table>

The OfflineGracePeriod key has type REG_DWORD. The default settings is 720 (30 days).

3. Update the OfflineGracePeriod value to a timespan in hours.
4. Exit the Registry Editor.

**Using the Command-Line HorizonThinAppCtrl.exe Application**

The Workspace for Windows application includes a command-line application, HorizonThinAppCtrl.exe, that you can use to perform operations related to using ThinApp packages on the user's Windows system.

The installation process for Workspace for Windows installs HorizonThinAppCtrl.exe in the HorizonThinApp folder in the Windows directory location where the Workspace for Windows application is installed.

To use the HorizonThinAppCtrl.exe application to perform one of its supported commands, provide the command as the first argument, followed by the command's available options, as appropriate.

HorizonThinAppCtrl.exe command options
<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>HorizonThinAppCtrl.exe recheck</td>
<td>This command immediately does an entitlement check of the ThinApp packages that are associated with the user account that is logged into the Workspace for Windows application. Any newly entitled or updated ThinApp packages are synced.</td>
</tr>
<tr>
<td>HorizonThinAppCtrl.exe set InstallMode=install_mode</td>
<td>This command changes the ThinApp deployment mode used for ThinApp packages on this Windows system. Because this command changes the registry keys associated with the ThinApp deployment mode, only administrators with the appropriate registry permissions are able to change the install mode using this command. Available values for install_mode are:</td>
</tr>
<tr>
<td></td>
<td>- CopyToLocal</td>
</tr>
<tr>
<td></td>
<td>- RunFromShare</td>
</tr>
<tr>
<td></td>
<td>- HttpDownload</td>
</tr>
<tr>
<td>HorizonThinAppCtrl.exe authorize guid=ThinApp_GUID path=package_path</td>
<td>This command verifies whether a ThinApp package can be launched. This command does not actually launch the ThinApp package. Provide the ThinApp package's GUID and the path to the package's executable file. If ThinApp download mode is used for the packages on the Windows client system, the path is relative to the local cache root folder, which is the same as the path relative to the repository root. An example is HorizonThinAppCtrl.exe authorize guid=436E1D7D-552C-4F70-8197-DB1B95D38934 path=&quot;FileZilla Client 3.3.2/FileZilla.exe&quot; You can see the ThinApp package's GUID, application path, and executable file name on its resources page in the admin console.</td>
</tr>
<tr>
<td>HorizonThinAppCtrl.exe quit</td>
<td>This command tells the Workspace for Windows application to exit cleanly.</td>
</tr>
<tr>
<td>HorizonThinAppCtrl.exe launch app=package_path url=launch_url</td>
<td>This command is used to manually launch a ThinApp package, where package_path is the path to the package's executable file, and launch_url is the Workspace protocol URL for that package, in the form horizon://package_path. An example is HorizonThinAppCtrl.exe launch app=&quot;FileZilla Client 3.3.2/FileZilla.exe&quot; url=&quot;horizon://FileZilla Client 3.3.2/FileZilla.exe&quot; This command is not typically used by end users, who can launch their entitled ThinApp packages from their My Apps area in the user portal. This command is typically used for debugging.</td>
</tr>
</tbody>
</table>

### Updating Managed ThinApp Packages After Deployment in Workspace

After adding a ThinApp package to your organization’s catalog and entitling your Workspace users to that ThinApp package, your organization might want to update that package and have the users use a newer, or rebuilt, version of the ThinApp package, without having to unentitle the users from the current package and then entitling them to the newer package.

An updated ThinApp package might be made available because a newer version of the Windows application for that package is released, or because the packager of the application has changed the values of parameters used by the package.

ThinApp 4.7.2 and newer versions provide an update mechanism for ThinApp packages used in a Workspace system. This ThinApp update mechanism is different from other update mechanisms for ThinApp packages used outside of a Workspace environment. The updated ThinApp package must have been updated with this mechanism for you to be able to deploy the updated package in Workspace and have users automatically see the newer version.
For ThinApp packages that are managed in a Workspace system, two Package.ini parameters are used by Workspace to determine that a package is an updated version of another package.

**AppID**

The unique identifier for the ThinApp package in Workspace. All entry points (executables) for the package's application are assigned the same AppID. After a ThinApp package is synced to your organization's Workspace catalog, the package's AppID is displayed in the GUID column in the ThinApp package's resource page. This value consists of alphanumeric characters in a pattern of character sets, each set separated by dashes, such as in the following example:

```
XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX
```

Workspace considers any ThinApp package with the same AppID to be versions of the same application.

**VersionID**

The version number of the ThinApp package. Workspace uses the VersionID to keep track of different versions of the managed ThinApp package. You increment the VersionID value by one (1) to mark that ThinApp package as an update of another package, retaining the same AppID.

You place the updated package in a new folder in the network share folder configured for the managed ThinApp packages. See *Installing and Configuring Workspace*. When Workspace performs the scheduled sync with the network share folder and it encounters an application that has the same AppID as another application, it compares the VersionID values. The ThinApp package with the highest VersionID is used as the most recent update. Workspace automatically incorporates the previous user entitlements to the ThinApp package with the highest VersionID, and shortcuts on the users' systems are synced to point to the updated package.

**IMPORTANT** The standard ThinApp InventoryName parameter is important to successful updates of managed ThinApp packages. Both the previous and updated ThinApp packages must have the same value for the InventoryName parameter. If the person creating the ThinApp package changes the InventoryName in a package, and then creates an updated package, you must make sure the InventoryName values match for the updates to work properly in your Workspace system.

See the *ThinApp Package.ini Parameters Reference Guide* for details about the various parameters that are used in a ThinApp package's Package.ini file.

### Update a Managed ThinApp Package

Updating a ThinApp package that is already managed by Workspace and in your organization's catalog involves multiple steps. The updated ThinApp package might be provided to you by another group in your organization. To ensure that your Workspace system can automatically use the updated package in place of the existing one for the entitled users, you must ensure the updated package was created using the same AppID as the current package, has a VersionID value that is higher than the existing package's VersionID value, and is enabled for management by Workspace.

**Prerequisites**

Verify that you have access to the location where your managed ThinApp packages reside and can create subfolders at that location.

**What to do next**

Your Workspace catalog displays the new version of the updated ThinApp package after the next sync of your Workspace system with your ThinApp package location. If you want to see the new version reflected in the ThinApp package's resources page, you can manually sync using the Packaged Apps - ThinApp page of the Connector Services Admin.
**Obtain the AppID and VersionID values of a Managed ThinApp Package**

To ensure that Workspace automatically uses the updated ThinApp package in place of the current one, the updated ThinApp package must be created using the AppID of the currently managed ThinApp package and a higher VersionID value than the current version.

When the Setup Capture process is used to create an updated ThinApp package, the AppID value is automatically retrieved by the Setup Capture program from the existing ThinApp package’s executables, and the VersionID value is automatically incremented. However, the person who is creating the updated ThinApp package might use a different method for creating the updated package. When the Setup Capture process is not used to create the updated ThinApp package, the person creating the package must obtain the AppID and VersionID values for the ThinApp package that is currently managed by your Workspace system. The AppID and VersionID values are displayed on pages in the ThinApp package’s resource page in the Workspace Admin Console.

**Procedure**

1. Click the **Catalog** tab.
2. Click **Any Application Type > ThinApp Packages**.
3. Click the ThinApp package to open its resource page.
4. Click **Details**.
5. Make note of the value listed in the **Version** field on the Details page.
6. Click **ThinApp Package** to display the ThinApp Package page.
7. Make note of the **AppID** value listed in the **GUID** column.
   - The value listed in the GUID column is the value that Workspace uses to identify this ThinApp package.

**What to do next**

The person who is creating the updated ThinApp package should complete the steps in “Create the Updated ThinApp Package,” on page 43.

**Create the Updated ThinApp Package**

The AppID and VersionID values of the currently managed ThinApp package are used for creating the updated package. The updated package uses the same AppID value and a higher VersionID value.

Sometimes the updated ThinApp package is provided to you by another team in your organization. The person who creates the updated ThinApp package can use one of the described methods.

**Prerequisites**

Verify that you have the AppID and VersionID values of the current ThinApp package by completing the steps in “Obtain the AppID and VersionID values of a Managed ThinApp Package,” on page 43.

Verify that you have a version of the ThinApp program that is compatible with your version of Workspace. For information about specific ThinApp versions, see the VMware Product Interoperability Matrixes at http://www.vmware.com/resources/compatibility/sim/interop_matrix.php.
Procedure

◆ Using a version of the ThinApp program that is supported by Workspace, create the updated ThinApp package using one of the available methods.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
</table>
| Recapture using Setup Capture.             | Use this method when the project folder for the existing ThinApp package managed by Workspace is unavailable. To create an updated package with Setup Capture, you need only the following items:  
  - The application executables from the existing ThinApp package  
  - The application installer  
  - Setup Capture and the ThinApp program at a version supported by Workspace.  
  During the capture process, select to manage the package with Workspace and that the package is an update of an existing base ThinApp package. Browse to the folder that contains the executables for the currently managed ThinApp package. Point to the folder, and not to specific executables.  
  With this method, you do not need to obtain the AppID or VersionID values in advance of creating the updated package. After you designate the package as an update and point to the prior version in Setup Capture, the capture process reads the AppID of the prior package and reuses it for the updated package. The process also provides an incremented VersionID for the updated package, and assigns the same InventoryName. |
| Update the Package.ini file manually and then rebuild the package. | Use this method when you do not have the application installer for the recapture process, or when you need to update the package to a newer ThinApp version and want to update more than what the relink command would handle. Because rebuilding a package incorporates changes to the file system and registry which come in a new version of ThinApp, a rebuild would pick up those changes, such as when a new ThinApp version provides a new Package.ini parameter that you want to set.  
  To mark the new package as an update, edit the following Workspace parameters in the [Build Options] section of the Package.ini file:  
  - Set the AppID parameter to match the AppID value of the currently managed ThinApp application. You cannot reuse a value of genid for AppID, because then a new AppID value will be generated for the updated package and your Workspace system will not recognize the new package as an update to the existing one.  
  - Increment the value of the VersionID parameter to a higher integer than the currently managed ThinApp package. If there is no VersionID parameter set for the currently managed package, its value is 1 by default, and you would add a line for the VersionID parameter to Package.ini and set it to a value of 2 (VersionID = 2).  
  - Make sure the InventoryName parameter value matches the InventoryName value of the currently managed package. The InventoryName values for the current package and the updated package must be identical. |
| Use the relink --h command with the AppID and VersionID options. | Use this method in one of the following situations:  
  - You do not have the project folder for the application.  
  - You have already captured, built, and tested the package in a test environment that was not a Workspace system, and the only remaining steps are to enable the updated package for Workspace and place it in the network share used by the Workspace system.  
  - You are updating the package only to update the ThinApp runtime for the package to incorporate bug fixes available in that new ThinApp version. |
For example, if you have changed the project directory, including the Package.ini file, for a virtual application, rebuilt the package, and tested the package, the test environment might not have been Workspace. The final stage of updating the application is to enable it for Workspace. At that point, the easiest route is to use the relink -h command, instead of recapturing or rebuilding.

NOTE The ThinApp runtime is always updated when you run the relink -h command on a ThinApp package.

You can run the relink command from the ThinApp Program Files directory to get help on the command’s syntax.

When the existing ThinApp package is already enabled for use by Workspace, you can run the following command to reuse the package’s existing AppID and increment the VersionID:

relink -h -VersionID + executable-folder/*.*

Where executable-folder is a folder containing the executables of the ThinApp package you want to update.

IMPORTANT When you use the relink command, you cannot point it directly to the folder of package executables on the network share used for the ThinApp packages in the Workspace environment. The command converts the old executables to BAK files when it updates the ThinApp runtime, and it writes those BAK files, as well as the new files, to the folder. Because the network share typically does not allow writing to it, you must point relink to a copy of the folder of executables.

Other use cases for the relink command, including enabling a ThinApp package for use in a Workspace environment, are covered in the VMware knowledge base article at http://kb.vmware.com/kb/2021928.

You have a set of files (EXE files, and optionally DAT files) for the updated ThinApp package.

What to do next

Copy the files to a new subfolder on the network share, by completing the steps in “Copy an Updated ThinApp Package to the Network Share,” on page 45.

Copy an Updated ThinApp Package to the Network Share

After you create the updated ThinApp package, you copy the appropriate files to a new subfolder at the same level as the existing subfolder on the network share.

Prerequisites

Verify that you have the files for the updated ThinApp package, as a result of completing the steps in “Create the Updated ThinApp Package,” on page 43 and incrementing the VersionID value.

Verify that you have access to the network share and can make subfolders and copy files to it.

Procedure

1. In the network share folder, create a new subfolder for the updated ThinApp package.

   Retain the existing subfolder for the ThinApp package that you are updating, and do not alter its contents.

   After the next scheduled sync, Workspace ignores the older package, when it recognizes the new package has the same AppID value and a higher VersionID value.
Typically, you name the subfolder to match the name of the ThinApp application, or indicate what application is in the folder. For example, if the network share is named appshare on a host named server, and the application is called abceditor, the subfolder for the ThinApp package is `\server\appshare\abceditor`.

**Note** Do not use non-ASCII characters when you create your network share subfolder names for ThinApp packages to distribute by using Workspace. Non-ASCII characters are not supported.

1. Copy the EXE and DAT files for the updated ThinApp package into that new subfolder.
2. (Optional) If you do not want to wait for the next scheduled sync time, you can manually sync your Workspace system with the network share using the Packaged Apps - ThinApp page of the Connector Services Admin.

When Workspace performs the scheduled sync with the network share folder and it encounters an application that has the same AppID as another application, it compares the VersionID values. The ThinApp package with the highest VersionID is used as the most recent update. Workspace automatically incorporates the previous user entitlements to the ThinApp package with the highest VersionID, and shortcuts on the users' systems are synced to point to the updated package.

### Delete ThinApp Packages from Workspace

You can permanently remove a ThinApp package from Workspace.

When you delete a ThinApp package from Workspace, you permanently remove it. You can no longer entitle users to the ThinApp package unless you add it back to Workspace.

**Procedure**

1. Delete the ThinApp package subfolder from the network file share that is the ThinApp package repository connected to your Workspace system.
2. Delete the application from Workspace.
   a. Log in to the Workspace Admin Console.
   b. Click the Catalog tab.
   c. Click **Any Application Type > ThinApp Packages**.
   d. Search for the ThinApp package to delete.
   e. Click the ThinApp package name to display its resource page.
   f. Click **Delete**, read the message, and if you agree, click **Yes**.

The ThinApp package does not exist in your Workspace catalog.
Make Existing ThinApp Packages Compatible with Workspace

You can convert a ThinApp package from one that is not compatible with Workspace to one that Workspace can distribute and manage. You can use one of the following methods: use the ThinApp 4.7.2 relink command, rebuild the package from its ThinApp project files after editing the project’s Package.ini file to add the necessary Workspace parameters, or recapture the Windows application with the appropriate Workspace settings selected in the ThinApp Setup Capture program.

**Note** A ThinApp package that is compatible with Workspace can only be used for a Workspace deployment. Only Workspace users who have the Workspace Client for Windows installed can launch and run these enabled packages. At runtime, the ThinApp package loads a specifically named DLL, and uses that DLL to verify the user’s entitlement with their Workspace system. Because the DLL is installed with the Workspace Client for Windows, such ThinApp packages can only be run on Windows systems on which that client is installed.

**Prerequisites**

Verify that you have access to the necessary items for your chosen method.

- If you are using the relink command, verify that you have the executable files for the ThinApp package that you are converting and the ThinApp 4.7.2 relink.exe application.
- If you are updating the ThinApp project’s Package.ini file and rebuilding the package, verify that you have the project files needed by the ThinApp 4.7.2 program to rebuild the package.
- If you are recapturing the Windows application, verify that you have the ThinApp 4.7.2 Setup Capture program and the application installer and other items that the program needs to recapture the application. See the ThinApp User’s Guide for details.

Verify that you have access to the ThinApp network share used by your Workspace system, and can make subfolders and copy files to it.
Procedure

- Using a version of the ThinApp program that is supported by Workspace, create a compatible ThinApp package using one of the available methods.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Use the relink -h command.</td>
<td>Using the relink -h command is the easiest method. You must use the relink.exe program from ThinApp 4.7.2 or later. Use this method in one of the following situations:</td>
</tr>
<tr>
<td></td>
<td>- You cannot use the rebuild method because you do not have the project folder.</td>
</tr>
<tr>
<td></td>
<td>- Using Setup Capture to recapture the application would take too long.</td>
</tr>
<tr>
<td></td>
<td>- You do not have the application installer that is required for recapturing with Setup Capture.</td>
</tr>
<tr>
<td></td>
<td><strong>Note</strong>: The ThinApp runtime is always updated when you run the relink -h command on a ThinApp package.</td>
</tr>
<tr>
<td></td>
<td>You can run the relink command from the ThinApp Program Files directory to get help on the command's syntax.</td>
</tr>
<tr>
<td></td>
<td>To create a compatible package, use the basic syntax of the command: <code>relink -h executable-folder/*.*</code>.</td>
</tr>
<tr>
<td></td>
<td>Where <code>executable-folder</code> is a folder containing the executables of the ThinApp package you want to update.</td>
</tr>
<tr>
<td></td>
<td><strong>Important</strong>: When you use the relink command, you cannot point it directly to the folder of package executables on the network share used for the ThinApp packages in the Workspace environment. The command converts the old executables to BAK files when it updates the ThinApp runtime, and it writes those BAK files, as well as the new files, to the folder. Because the network share typically does not allow writing to it, you must point relink to a copy of the folder of executables. Other use cases for the relink command are covered in the VMware knowledge base article at <a href="http://kb.vmware.com/kb/2021928">http://kb.vmware.com/kb/2021928</a>.</td>
</tr>
<tr>
<td>Update the Package.ini file manually with the necessary Workspace parameters, and then rebuild the package.</td>
<td>Use this method when you do not have the application installer for the recapture process, when you want to avoid doing the up-front setup that recapturing the application requires, or when you want to incorporate functionality from a newer ThinApp version more than what the relink command would provide. Because rebuilding a package incorporates changes to the file system and registry which come in a new version of ThinApp, a rebuild would pick up those changes, such as when a new ThinApp version provides a new Package.ini parameter that you want to set. In the [Build Options] section of the Package.ini file, add the following parameters:</td>
</tr>
<tr>
<td></td>
<td>;-------- Workspace Parameters ----------</td>
</tr>
<tr>
<td></td>
<td>AppID=genid</td>
</tr>
<tr>
<td></td>
<td>NotificationDLLs=HorizonPlugin.dll</td>
</tr>
<tr>
<td></td>
<td>HorizonPlugin.dll is the DLL that the ThinApp runtime calls to verify the Workspace user's entitlement to use the virtualized application.</td>
</tr>
<tr>
<td></td>
<td>You can optionally include the HorizonOrgURL parameter and set it to your Workspace fully qualified domain name (WorkspaceFQDN). See Installing and Configuring Workspace.</td>
</tr>
<tr>
<td>Recapture using Setup Capture, and select the necessary Workspace settings.</td>
<td>Use this method when you would prefer to recapture the application rather than use one of the other methods. To create a compatible package using ThinApp Setup Capture, select the appropriate settings in the wizard to manage the package with Workspace during the capture process. See the ThinApp User's Guide for details on the capture process.</td>
</tr>
</tbody>
</table>

You have a set of files (EXE files, and optionally DAT files) for a ThinApp package that Workspace can distribute and manage.
What to do next

For steps to add ThinApp packages to the network share, see “Create a Network Share for ThinApp Packages That Workspace Manages,” on page 32.

Change the ThinApp Packages Share Folder

After you configure Workspace access to your ThinApp packages, your IT environment might change such that your ThinApp packages are in a new location. When this situation occurs, use the Packaged Apps - ThinApp page of the Connector Services Admin to update the path to the new location.

Prerequisites

Verify that the new network share location adheres to the network share requirements as described in “Workspace Requirements for ThinApp Packages and the Network Share Repository,” on page 28.

Procedure

1. Log in to the Connector Services Admin.
2. Select the Packaged Apps - ThinApp tab.
3. Change the value in the Path text box to the new shared folder where the ThinApp packages are located in the UNC path format.
4. (Optional) If the previous network share was a CIFS share and the new share is a DFS share, select the Enable account based access check box and enter the name and password of a user who has read access to that network share.
5. Click Save.
Configuring Workspace for Windows

Before your Workspace users can run their ThinApp packages that are registered to them using Workspace, those users must have the Workspace for Windows application installed and running on their Windows systems.

The Workspace for Windows application can be installed by either double-clicking its installer EXE file, running the executable file using the command-line options, or running a script that uses the command-line options. Local administrator privileges are required to install the application.

The configuration of the Workspace for Windows application on the Windows endpoint determines whether a ThinApp package that is distributed using Workspace is deployed using ThinApp streaming mode, RUN_FROM_SHARE, or one of the ThinApp download modes, COPY_TO_LOCAL or HTTP_DOWNLOAD. When you create the script to silently install Workspace for Windows to Windows endpoints, such as desktop and laptop computers, you set the options that set the ThinApp package deployment mode. Choose the deployment mode that best fits the network environment for the selected endpoints, considering details such as network latency.

Note: If any browser windows are open during installation of the Workspace for Windows application, problems might occur with launching ThinApp packages from the user portal. Either close all browser windows before installing the application, or immediately after installing the application, restart your browsers. See “ThinApp Packages Fail to Launch from the User Portal,” on page 76.

This chapter includes the following topics:

- “Command-Line Installer Options for Workspace for Windows,” on page 51
- “Install the Workspace for Windows Application with Identical Settings to Multiple Windows Systems,” on page 57
- “Add Desktop Client Installer Files to workspace-va Virtual Appliances,” on page 58

Command-Line Installer Options for Workspace for Windows

You can set various options for the Workspace for Windows client application when you run its installer program using the command line or a deployment script. Because this client application provides features related to the use of the file-sharing and ThinApp package capabilities, you typically use these installer options to set runtime options associated with those features.

Available Command-Line Options for the Workspace for Windows Installer

After you download the EXE file for the client application’s installer to a Windows system, you can see a list of the installation options by running the following command,

`Workspace-n.n.n--nnnnnn /?`
where n.n.n-nnnnnnn represents the file’s version and build number. A dialog box appears that lists the available installation options that you can pass to the installer when installing the client application using the command line or a deployment script.

### Table 6-1. Installer Command-Line Options

<table>
<thead>
<tr>
<th>Installer Option</th>
<th>Value</th>
<th>Description</th>
<th>Long Form Alternative</th>
</tr>
</thead>
<tbody>
<tr>
<td>/c</td>
<td>path to configuration file/filename</td>
<td>Provides a configuration file to use in the installation process.</td>
<td>/cfgfile</td>
</tr>
<tr>
<td>/T</td>
<td>path to temporary directory</td>
<td>Provides a directory to use as a temporary extraction location during the installation process.</td>
<td>/Temp</td>
</tr>
<tr>
<td>/S</td>
<td>path to search for files</td>
<td>Provides a set of paths for the installer to search for files, in addition to the installer package file.</td>
<td>/SearchPaths</td>
</tr>
<tr>
<td>/P</td>
<td>path to package file/filename</td>
<td>Provides a package file to use in the installation process.</td>
<td>/Package</td>
</tr>
<tr>
<td>/s</td>
<td>No values</td>
<td>Runs the installation silently. A silent installation does not display messages or windows during deployment. You typically use this option when using a deployment script to run the installer program in an unattended install process, and you want to suppress the display of interactive messages and windows. You can use this option with the /x or /uninst option to silently uninstall the application.</td>
<td>/silent</td>
</tr>
<tr>
<td>/nsr</td>
<td>No values</td>
<td>Suppresses an automatic reboot after a successful silent installation process.</td>
<td>/noSilentReboot</td>
</tr>
<tr>
<td>/f2</td>
<td>path to log file</td>
<td>Provides the location of the installation log file.</td>
<td>/log</td>
</tr>
<tr>
<td>/d</td>
<td>No values</td>
<td>Write debugging information to the installation log file.</td>
<td>/debug</td>
</tr>
<tr>
<td>/V</td>
<td>No values</td>
<td>Sets verbose logging.</td>
<td>/verbose</td>
</tr>
<tr>
<td>/L</td>
<td>English language name or localized language name or three-letter language abbreviation or language identifier</td>
<td>Runs the installer in the named language, for example, Workspace-n.n.n-nnnnnnn /lang French where n.n.n-nnnnnnn is the file version and build number for your downloaded installer program.</td>
<td>/lang</td>
</tr>
<tr>
<td>/v</td>
<td>key-value pairs</td>
<td>Provides a set of arguments to use in the installation process as a key-value pair, provided in the format key=value. These arguments are passed to the MSI file and configure runtime for the ThinApp packages capabilities provided by the Windows application.</td>
<td>/msi_args</td>
</tr>
<tr>
<td>/x</td>
<td>No values</td>
<td>Uninstalls the application.</td>
<td>/uninst</td>
</tr>
<tr>
<td>/clean</td>
<td>No values</td>
<td>Cleans out the installation registration information.</td>
<td>No long form alternative</td>
</tr>
</tbody>
</table>

### Key-Value Pairs for the /v Option

The following table describes the available key-value pairs used for the /v installer option.
### Table 6-2. Keys for the /v Installer Command-Line Option

<table>
<thead>
<tr>
<th>Keys for the /v Option</th>
<th>Values</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>HORIZONURL</td>
<td>URL</td>
<td>Provides the URL to your Workspace system, where HTTPS is the required protocol, to allow the Windows application to communicate with your Workspace system, for example HORIZONURL=<a href="https://WorkspaceFQDN">https://WorkspaceFQDN</a>. <strong>Note</strong> The value must include the full URL, including the protocol portion, such as https://.</td>
</tr>
</tbody>
</table>
| PROMPTFORAUTH          | 0 or 1 | Presents users with a login window if Kerberos authentication fails. Set the value of this variable to 1 to enable the client application to respond to a Kerberos failure by prompting users to log in from a browser window.  
  
  `Workspace-n.n..n-nnnnnnn`  
  `/v PROMPTFORAUTH=1`  
  
  Setting the value of this variable to 0 is the same as not including the variable in the command. The result is that upon a Kerberos failure the client application does not automatically open the login in a browser window. However, the user receives an indication in the Windows system tray that the client is not authenticated to the server yet and can start the process manually. |
<table>
<thead>
<tr>
<th>Keys for the ( /v ) Option</th>
<th>Values</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>INSTALL_MODE</td>
<td>COPY_TO_LOCAL</td>
<td>Sets the deployment mode for how the Workspace for Windows application obtains ThinApp packages at runtime. ThinApp packages are virtualized Windows applications. The ThinApp packages reside on a network share that is integrated with your Workspace system.</td>
</tr>
<tr>
<td></td>
<td>HTTP_DOWNLOAD</td>
<td>With COPY_TO_LOCAL, the user's entitled packages are downloaded to the client Windows system using a file copy. When the user launches a ThinApp package, the virtualized application runs locally on that system. Before the user's first download and use of an entitled ThinApp package and to continue synchronizing the packages to the client Windows system, the client Windows system must join the same Active Directory domain to which the ThinApp packages' network share is joined. The user account used to log in to the Windows system is the account that is used to obtain the ThinApp packages from the network share. That account must have the appropriate permissions on the network share to read and copy files from the network share.</td>
</tr>
<tr>
<td></td>
<td>RUN_FROM_SHARE</td>
<td>With HTTP_DOWNLOAD, the user's entitled packages are downloaded to the client Windows system using the HTTP protocol. When the user launches a ThinApp package, the virtualized application runs locally on that system. The Workspace client uses the user's Workspace system account to authenticate to your Workspace Server to obtain the list of the user's entitled packages to download. The share user account provided in the Connector Services Admin for enabling account-based access to the ThinApp packages' network share is the account used by Workspace to access the ThinApp packages from the repository. That share user account for Workspace needs read permission on the network share. The account that the user used to log in to the client Windows system and the user's Workspace system account do not need to have any permissions on the network share. The client Windows system does not have to join the same domain to which the ThinApp packages' network share is joined. This download method is typically slower than using the other modes. The benefit to this mode is that the client Windows system does not have to join the Active Directory domain to obtain and run the virtualized application. <strong>IMPORTANT</strong> For the HTTP_DOWNLOAD option to work, the ThinApp packages integration in Workspace must be configured for account-based access. See &quot;Installing and Configuring Workspace.&quot;</td>
</tr>
<tr>
<td></td>
<td></td>
<td>With RUN_FROM_SHARE, the virtualized application is streamed to the client Windows system from the network share when the user launches the ThinApp package. The RUN_FROM_SHARE option is best suited for Windows systems that will always have connectivity to the network share where the ThinApp packages reside, because the ThinApp packages are not present on the Windows system and the virtualized applications only run if the Windows system can connect to the network share. The client Windows system must join the same Active Directory domain to which the ThinApp...</td>
</tr>
</tbody>
</table>
Table 6-2. Keys for the /v Installer Command-Line Option (Continued)

<table>
<thead>
<tr>
<th>Keys for the /v Option</th>
<th>Values</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>packages' network share is joined. The user account used to log in to the Windows system is the account that is used to obtain the ThinApp packages from the network share. That account must have the appropriate permissions on the network share to read and execute files on the network share. For all of the modes, the network share must have the appropriate file and sharing permissions configured. See Installing and Configuring Workspace.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>IMPORTANT When installing Workspace for Windows in floating View desktops, you best use the RUN_FROM_SHARE option to avoid copying the ThinApp packages into those stateless View desktop systems.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>When the client application is installed with one of these configurations, the user account that logs into the Windows system must have the appropriate file and sharing permissions on the network share to be able to obtain the ThinApp packages:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- The RUN_FROM_SHARE option</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- The COPY_TO_LOCAL option, without also having the AUTO_TRY_HTTP option enabled and account-based access configured in Workspace</td>
</tr>
<tr>
<td></td>
<td></td>
<td>NOTE The INSTALL_MODE key replaces the deprecated DOWNLOAD key used in previous releases of the Workspace for Windows application.</td>
</tr>
<tr>
<td>POLLINGINTERVAL</td>
<td>frequency</td>
<td>Enables you to set the frequency, measured in seconds, of synchronizations between the installed client application and your Workspace system to check for new ThinApp packages or entitlements. If unspecified, the default value of 300 seconds (5 minutes) applies.</td>
</tr>
<tr>
<td>ENABLE_AUTOUPTDATE</td>
<td>0 or 1</td>
<td>Enables you to disable the automatic update check and download activity. If enabled, the installed Workspace for Windows application automatically checks if a newer application is available for downloading in your Workspace system. If there is a newer version available, the Workspace for Windows application automatically downloads and updates itself to the newer version. This option is enabled by default. Set the value of this variable to 0 to disable automatic update. If unspecified, the default value of 1 applies.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>NOTE An updated Workspace for Windows application is not installed if the logged-in user account does not have administrator privileges.</td>
</tr>
<tr>
<td>ENABLE_THINAPP</td>
<td>0 or 1</td>
<td>Enables you to disable syncing of ThinApp packages by the client application. Set the value of this variable to 0 to disable the syncing of ThinApp packages. If unspecified, the default value of 1 applies.</td>
</tr>
</tbody>
</table>
Table 6-2. Keys for the /v Installer Command-Line Option (Continued)

<table>
<thead>
<tr>
<th>Keys for the /v Option</th>
<th>Values</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SHARED_CACHE</td>
<td>0 or 1</td>
<td>Determines whether the ThinApp package cache is located in a common folder in the Windows system to which the client application is being installed. Set the value of this variable to 1 to specify that all user accounts on the Windows system share a common cache location. By default, the common folder is C:\Program Data\VMware\Horizon ThinApp. If unspecified, the default value of 0 applies, and each Windows user account gets its own cache, and its default location is %LOCALAPPDATA%\VMware\Horizon ThinApp\PackageCache. <strong>Note</strong> If you specify a shared cache, the Workspace for Windows application does not automatically delete ThinApp packages from this shared cache. Because SHARED_CACHE=1 indicates that all user accounts on the Windows system share the same location, the packages must remain in the shared location so that entitled users can use them, even when you unentitle one user. When you unentitle a user from a ThinApp package, the Workspace for Windows application unregisters that package for that user. Other entitled users on that Windows system can continue to use the ThinApp package. You can delete the common cache manually to reclaim the space if no user accounts on that Windows system are entitled to use the ThinApp packages. Each ThinApp package has its own folder under the cache location.</td>
</tr>
</tbody>
</table>

| AUTO_TRY_HTTP          | 0 or 1 | When the Workspace for Windows application is installed with the COPY_TO_LOCAL option and account-based access is configured for Workspace, the AUTO_TRY_HTTP option determines whether the client should automatically try downloading the user's entitled ThinApp packages using the HTTP protocol, as for the HTTP_DOWNLOAD option, if the first attempt to download fails. This option is enabled by default. Set the value of this option to 0 to disable automatically trying the HTTP protocol for the download. If unspecified, the default value of 1 applies. **Important** For the AUTO_TRY_HTTP option to work, the ThinApp packages integration in Workspace must be configured for account-based access. See “Workspace Requirements for ThinApp Packages and the Network Share Repository,” on page 28. |

**Example: Example of Using the Workspace for Windows Command-Line Installer Options**

If your Workspace instance has a URL of https://WorkspaceFQDN, and Workspace is configured for account-based access to your ThinApp packages' network share, and you want to silently install the Workspace for Windows application to multiple desktops of that Workspace instance with these options:

- The ThinApp install option set to HTTP_DOWNLOAD, because you expect these Windows systems will not be likely to join the domain. Workspace is appropriately configured for account-based access to the ThinApp packages' network share.
- The clients check for new packages and entitlements with the Workspace system every 60 seconds.

You would create a script that invokes the following command:

```
VMware-Horizon-Workspace-n.n.n-nnnnnn.exe /s
/v HORIZONURL=https://WorkspaceFQDN INSTALL_MODE=HTTP_DOWNLOAD POLLINGINTERVAL=60
```

where you replace the n.n.n-nnnnnn portion of the file name to match the name of your downloaded Workspace for Windows installer.
Deprecated Command-Line Options

The following options were used in previous Workspace releases and are no longer used in the current release.

- The /z SSLBYPASS option is ignored if provided on the command line.
- The /z HORIZONSERVER option is superseded by the /v HORIZONURL option.
- The /v DOWNLOAD option is superseded by the /v INSTALL_MODE option.

Install the Workspace for Windows Application with Identical Settings to Multiple Windows Systems

To deploy the Workspace for Windows application to multiple Windows systems and have the same configuration settings applied to all of those systems, you can implement a script that installs the Workspace for Windows application using the command-line installation options.

**IMPORTANT** Error messages do not appear on screen when you deploy Workspace for Windows silently. To check for errors during a silent installation, monitor the %TEMP% folder, checking for new vminst.XXXXXX.log files. The error messages for a failed silent installation appear in these files.

Typically, this deployment scenario is used for Windows systems that are View desktops. For a description of settings to use for non-persistent, also known as floating or stateless, View desktops, see “Reducing Resource Usage and Increasing Performance of Workspace for Windows In Non-Persistent View Desktops,” on page 25.

Prerequisites

- Verify that the Windows systems are running Windows operating systems that are supported for the version of the Workspace for Windows application you are installing. See the Workspace User Guide or the release notes that apply for that Workspace for Windows application.
- Verify that the Windows systems have installed browsers that are supported by the Windows application you are installing.
- If you want the ability to run a command to familiarize yourself with the available options before you create the deployment script, verify that you have a Windows system on which you can run that command. The command to list the options is only available on a Windows system. See “Command-Line Installer Options for Workspace for Windows,” on page 51.

Procedure

1. Obtain the Workspace for Windows installer’s executable file and locate that executable file on the system from which you want to silently run the installer.

One method for obtaining the executable file is to download it using the your Workspace system’s download page. If you have set up your Workspace system to provide the Windows application installer from the download page, you can download the executable file by opening the download page’s URL in a browser. The download page’s URL is https://WorkspaceFQDN/download, where WorkspaceFQDN is the fully qualified domain name for your Workspace system.
Using the installer's command-line options, create a deployment script that fits the needs of your organization.

Examples of scripts you can use are Active Directory group policy scripts, login scripts, VB scripts, batch files, SCCM, and so on.

For example, if your Workspace instance has a URL of https://WorkspaceFQDN, and you want to silently install the Windows client to Windows systems that you expect will be used off the domain, with the ThinApp deployment mode set to download mode and have the Workspace for Windows application sync with the server every 60 seconds, you would create a script that invokes the following command:

```
Workspace-n.n.n-nnnnnnn.exe /s
/v HORIZONURL=https://WorkspaceFQDN
INSTALL_MODE=HTTP_DOWNLOAD
POLLINGINTERVAL=60
```

where you replace the n.n.n-nnnnnnn portion of the file name to match that of your downloaded file.

Run the deployment script against the Windows systems.

If the silent installation is successful, the Workspace for Windows application is deployed to the Windows systems. Users logged in to those Windows systems can access their entitled assets from those systems.

**Notice**  A user's entitled ThinApp package is streamed or downloaded and cached to the user's Windows system after the polling interval elapses. As a result, users might see the ThinApp package displayed when they log in to the Workspace browser-based user portal. The ThinApp package does not start until the client syncs the application on the next polling interval.

**What to do next**
Verify that Workspace for Windows is properly installed on the Windows systems by trying some of the typical user tasks.

### Add Desktop Client Installer Files to workspace-va Virtual Appliances

When new versions of the Desktop clients are released, you copy and install a zip file that includes the Desktop client file for Windows computers from the VMware Downloads page to each workspace-va virtual appliance that is configured. You run the `check-client-updates.pl` command on each workspace-va virtual appliance to deploy the installer files and restart the Tomcat service on each workspace-va virtual appliance.

**Prerequisites**
- Users must have administrator privileges on their computers to install and automatically update the Desktop client. If users do not have administrator privileges, you can use software distribution tools to distribute and update the Workspace desktop applications to your users.
- Schedule adding these installer files to the workspace-va virtual appliances to run during a maintenance window since the workspace-va virtual appliance is restarted and this might interrupt user access.

**Procedure**

1. Download the Workspace Desktop client zip file from the VMware Web site, [https://my.vmware.com/web/vmware/downloads](https://my.vmware.com/web/vmware/downloads), to a computer that can access the workspace-va virtual appliance.

2. Copy the zip file to a temporary location on the workspace-va virtual appliance.
   ```
   scp clients-n.n.n-nnnnnnn.zip root@workspace-va.com:/tmp/
   ```

3. Log in to the workspace-va virtual appliance as the root user.
4 Unzip and install the new clients to the Downloads directory.

/usr/local/horizon/scripts/check-client-updates.pl --install --clientfile /tmp/clients.n.n.n-nnnnnn.zip

This script automatically unzips the file and copies the Desktop clients’ installer file for the Windows computers to the /opt/vmware/horizon/workspace/webapps/ROOT/client directory. It automatically updates to the /opt/vmware/horizon/workspace/webapps/ROOT/client/cds directory, and updates the URL parameter value for the downloads link.

5 Restart the Tomcat service on the workspace-va virtual appliance.

6 Repeat these steps on each workspace-va virtual appliance in your environment.

Users can download the Desktop clients from their Workspace accounts or via the download link, https://WorkspaceFQDN/download. Each user’s Desktop clients are automatically updated.
Providing Access to Citrix-Published Resources

You can provide Workspace users access to Citrix-published resources.

**NOTE**
Citrix published resources includes Citrix-published applications and Citrix-published desktops within the Citrix XenApp farm and does not include Citrix XenDesktop.

When you integrate a Citrix deployment with Workspace, Workspace users can use Citrix Receiver on their systems and devices to access their entitled Citrix-published resources.

**NOTE**
After you integrate Workspace with your Citrix deployment, you use the Citrix deployment to manage Citrix-published applications and Citrix-published desktops and to entitle users to those resources. You can then use the Workspace Admin Console to view these Citrix resources and their entitlements.

Workspace provides default global application delivery settings for Citrix-published resources. For example, you can edit the settings that control resource streaming and resource security. You can configure the delivery settings globally, for all the Citrix resources in the Workspace catalog, or for individual Citrix resources.

This chapter includes the following topics:

- “Integrating Workspace with Citrix-Published Resources,” on page 61
- “Enabling Citrix PowerShell Remoting on Citrix Server Farm,” on page 64
- “Preparing and Installing Integration Broker,” on page 66
- “Deploying Integration Broker,” on page 67
- “Synchronizing Workspace with Integration Broker,” on page 70
- “View User and Group Entitlements to Citrix-Published Resources,” on page 72
- “Editing Workspace Application Delivery Settings for Citrix-Published Resources,” on page 72
- “Managing Categories for Citrix-Published Resources,” on page 74

**Integrating Workspace with Citrix-Published Resources**

Workspace supports Citrix-published resources. You can use Workspace to seamlessly integrate with existing Citrix deployments. Workspace also supports applications, such as View, ThinApp, SAAS, and so on.

You can see more details about Workspace and Citrix farm synchronization in the Workspace and Citrix Farm Synchronization diagram. Workspace does not overwrite the settings in Citrix-published resources. Instead, it copies the information from the Citrix farm and reuses it in Workspace.
Workspace provides support for the following functions:

- Synchronize Citrix-published applications, Citrix-published desktops, or both from a Citrix farm to Workspace.
- Synchronize entitlements from a Citrix farm to Workspace entitlement store.
- Launch Citrix-published resources using SSO.

Workspace uses the Integration Broker to deliver Citrix-published resources to the end user, including the following functions:

- Workspace pushes Citrix farm information and publishes resource information from the Citrix farm to the catalog based on configured synchronization.
- A Workspace tenant administrator can set the generic user settings template and the ICA launch template for all the resources in an organization. This template is saved as an organization artifact in the Workspace data store.
- Workspace tenant administrator can set the ICA launch template by resource in the Workspace catalog. This template is saved as part of the resource definition in the Workspace catalog.

Workspace synchronizes the Citrix-published resources and entitlements from the Citrix farm to the Workspace entitlement store. Synchronization occurs based on the frequency set in the schedule. The Citrix farm is the single source of truth for all supported operations in Workspace.

Workspace uses the Citrix Receiver to launch Citrix-published resources. The end user must install the Citrix Receiver on their device. The Citrix Receiver delivers the Citrix-published resources to the end user.
Workspace also provides multi-device support. End users can launch a Citrix-published resource, such as the Textpad application, from Workspace on any device, such as a laptop, domain-joined desktop, or non-domain-joined desktop.

The following table describes the administrator’s task on the Citrix farm and the corresponding operation that results after a synchronization with Workspace.

**Table 7-1. Workspace and Citrix Farm Synchronization**

<table>
<thead>
<tr>
<th>Action in the Citrix farm</th>
<th>Result after sync with Workspace</th>
</tr>
</thead>
<tbody>
<tr>
<td>Publish a new resource, application or desktop, to the Citrix farm.</td>
<td>Workspace creates the Citrix-published resource in the Workspace catalog.</td>
</tr>
<tr>
<td>Edit a resource in the Citrix farm.</td>
<td>Workspace updates the Citrix-published resource in the Workspace catalog.</td>
</tr>
<tr>
<td>Delete a resource in the Citrix farm.</td>
<td>Workspace deletes the entitlements associated with the Citrix-published resource and then deletes the Citrix-published resource from the catalog.</td>
</tr>
<tr>
<td>Add an end user entitlement for a resource in the Citrix farm.</td>
<td>Workspace creates an entitlement in the Workspace entitlement store to associate with the Citrix-published resource and domain identities.</td>
</tr>
<tr>
<td>Remove an end user entitlement for a resource in the Citrix farm.</td>
<td>Workspace creates an entitlement in the Workspace entitlement store to associate with the Citrix-published resource and domain identities.</td>
</tr>
</tbody>
</table>

Workspace does not require additional setup after an Workspace upgrade or a Citrix product upgrade to maintain the integration between Workspace and Citrix-published resource support. To reinstall Citrix Receiver, see the Citrix documentation.

Workspace uses its own Integration Broker component and a Citrix SDK to handle SSO from Workspace to Citrix-published resources.
Enabling Citrix PowerShell Remoting on Citrix Server Farm

You must enable remote invocations between the Integration Broker and the Citrix farm. Citrix PowerShell Remoting requires a secure HTTPS channel to make remote calls. To enable this function, you must install a valid certificate from a certification authority.

Prerequisites
Verify that the instructions you use to set up Citrix PowerShell Remoting match the version of the Citrix server farm you use.

Set Up Citrix PowerShell Remoting on Citrix Server Farm 6.5

You must enable Citrix PowerShell remoting on every machine where you want to receive connections. Citrix PowerShell remoting enables connections between Integration Broker and the Citrix server farm.

Procedure
1. Install an SSL server certificate on each Citrix server farm host.
   For information about how to install an SSL server certificate, see the Microsoft documentation.
2. Click Properties and verify that Server Authentication is enabled for the certificate.
3. Enable Citrix PowerShell Remoting.
   a. Open Citrix PowerShell Module > Program Files.
   b. Type the set-executionpolicy remotesigned command.
Type the Import-Module command.

Import-Module C:\Program Files\Citrix\PowerShell
Modules\Citrix.XenApp.Commands.Remoting\Enable-XAPSRemoting.ps1

If the Citrix PowerShell modules are not installed in the default location, replace the default path with the path that is used in your environment.

d Type the Enable-XAPSRemoting command.

**Set Up Citrix PowerShell Remoting on Citrix Server Farm 5.0 or 6.0**

You must enable Citrix PowerShell remoting on every machine where you want to receive connections. Citrix PowerShell remoting enables connections between Integration Broker and the Citrix server farm.

**Prerequisites**

- If you do not have Winrm installed, download and install Winrm from the Microsoft Web site.

**Procedure**

1 Install an SSL server certificate on each Citrix server farm host.
2 Click Properties and verify that Server Authentication is enabled for the certificate.
3 Open the PowerShell console in the administrator mode.
4 Enable Citrix PowerShell Remoting.
   a Type the Get-Service winrm command to verify that Winrm is installed on the server.
   b Type the Enable-PSRemoting command.
      This command enables PowerShell Remoting on the server.
   c Install the Citrix PowerShell SDK 5.0 or 6.0 depending on the Citrix server version.
   d Enable winrm HTTPS listener from the command prompt.
      1 Create a certificate on the server.
      2 Record the certificate’s thumb print.
      3 Verify that the certificate’s thumb print is configured.
      winrm quickconfig -transport:https
   e Create the listener.
      winrm create winrm/config/Listener?Address=*+Transport=HTTPS @{Hostname="host FQDN";CertificateThumbprint="certificate thumbprint"}
f Verify that the listener was created.

```christian
winrm e winrm/config/listener
```

This server is ready to use.

g After the listener is created, go to the Integration Broker server to verify that PowerShell remoting is installed correctly.

```christian
winrm identify -r:https://XENAPP_HOSTNAME:5986 -u:USERNAME
```

Output:

```
IdentifyResponse
ProductVendor=Microsoft Corporation
ProductVersion=OS: 6.0.6002 SP: 2.0 Stack: 2.0
```

Preparing and Installing Integration Broker

Before you install Integration Broker, you must prepare your Workspace environment. You must verify that your software is installed and configured correctly.

**IMPORTANT**  Workspace 2.1 does not install or require installation of any VMware software on Citrix servers.

Prepare Integration Broker Server for Windows Server 2008 or Windows Server 2012

Before you install Integration Broker, you must prepare your Workspace environment and verify that your software is installed and configured correctly.

### Prerequisites

- Verify that Windows Server 2008 R2 or Windows Server 2012 are installed with the latest updates. To check for updates, select **Control Panel > Windows Update**.
- Install .NET Framework 3.5. When you install .NET, it installs version 3.5 as a feature. Verify that you include WCF activation.
- From the Add Role Service pane, install IIS 7 with 6.0 Management Compatibility Mode. You must also install the Management Tools if this is your only IIS 7 instance.
- Configure an application pool. You can use the default application pool or create an application pool that is dedicated to Integration Broker.


### Procedure

1. Configure the Integration Broker using the default application pool.
   a. Click the default application pool.
   b. Verify these requirements.
      - .NET framework version 2.0
      - Set 32-bit applications to true.
2 Configure Identity to use the same account as the Citrix-published resources administrator.
   Integration Broker uses this account to authenticate.
   a Right-click the application pool.
   b Click **Identity** in the Advanced Settings dialog.
   c Click **Custom Account** and click **Set**.
   d Type the credentials for the Citrix-published resources administrator username and password, *Domain Name* username and *Domain Name* password.

   **NOTE** Depending on your operating system, you might need to download either the 32-bit or 64-bit version of Microsoft Visual J#.

4 Depending on your version of Citrix Server Farm, download and install version 6.0 or 6.5 of Citrix PowerShell SDK from the Citrix Web site.
   a Set the execution policy for Citrix PowerShell Remoting.
   b If the Citrix XenApp Farm’s execution policy is configured to use remote sign on, you must add your root certificate to the Trusted Root Certification Authorities store. See the Microsoft Web site about adding root certificates to the store.
   c If the Citrix XenApp Farm’s execution policy is configured to unrestricted, you do not need to add root CAs to the Trusted Root Certification Authorities store.

5 Before you run this command, verify that PowerShell SDK is successfully installed.
   a Launch PowerShell SDK as administrator.
   b Verify PowerShell remoting.
      This sample command string applies to Citrix Server Farm 6.5.
      \texttt{Get-XAApplication -ComputerName CITRIX SERVER NAME}
      This sample command string applies to Citrix Server Farm 6.0.
      \texttt{Invoke-Command -ComputerName XENAPP_HOST_NAME -ScriptBlock \{ Add-PSSnapin Citrix* ;Get-XAApplication \} -Credential DOMAIN\USERNAME}
   c Verify that the list includes all the applications hosted by Citrix.

What to do next

If the \texttt{Invoke-Command} Command fails, see “Memory Issue Prevents Proper Configuration of Integration Broker,” on page 82.

Next, deploy and configure Integration Broker.

**Deploying Integration Broker**

To deploy Integration Broker, you must run the Integration Broker installation and set up a secure channel between Integration Broker and the Citrix server farm.

**ATTENTION** It is recommended that you install only one instance of Integration Broker per Windows Server instance.
Install Integration Broker

Workspace uses the Workspace Integration Broker component and the Citrix SDK to handle single sign-on between Workspace, Citrix server farms, and Citrix-published resources.

**Prerequisites**

Install Citrix PowerShell remoting. See the Citrix documentation for more information.

**Procedure**

1. Log in as a Windows administrator.
2. Open the [IB.msi](#) file to run the Integration Broker installation.
3. Type the Web location where you want to install the Integration Broker.
4. (Optional) If you created a separate pool for the Integration Broker, select your application pool.

   **CAUTION** Do not change the Virtual Directory name.

5. Click **Next** to finish installing Integration Broker.

Set Up Integration Broker for HTTP and HTTPS Bindings

Citrix PowerShell Remoting requires a secure HTTPS channel to make remote calls. Without a secure HTTPS channel, you cannot enable remote invocations between Integration Broker and Citrix server farm.

A certificate is required to use the IIS Server. You can purchase or generate a certificate from a third-party root CA. OpenSSL is installed by default. OpenSSL is an open source implementation of the SSL and TLS protocols.

**Prerequisites**

Mark the certificate key exportable. See the Microsoft documentation for more information on certificate keys.

**Procedure**

1. Add HTTPS binding to the Integration Broker Web site.
   a. Click **Start > Run**.
   b. Open inetmgr and right-click on the default Web site.
   c. Click **Edit Bindings**.
   d. Add the HTTPS binding using the newly created certificate in the drop-down menu.
2. Download the Citrix Web Interface SDK 5.4 (WISDK zip file) from the Citrix Web site.
3. After the installation is finished, unzip the [wisdk.zip](#) file.
4. Copy the contents from the [WIS_4_0_SDK/zipfiles/sdkdemo/wisdk](#) directory to the default bin directory at `c:\inetpub\wwwroot\IB\bin`.
5. Restart IIS.
6. Verify that the HTTP binding produces the expected output by typing `http://hostname/IB/API/RestServiceImpl.svc/ibhealthcheck` in the address bar of a browser.

   The expected output displays.
   ```
   All ok
   ```
Verify that the HTTPS binding produces the expected output by typing https://hostname /IB/API/RestServiceImpl.svc/ibhealthcheck in the address bar of a browser.

The expected output displays.

All ok

Create a Self-signed Certificate Example

These instructions provide a sample for how to set a self-signed certificate using OpenSSL for Integration Broker.

Procedure

1 Create a self-signed certificate for the IIS server.
2 Create the ibcerts folder to use as the working directory.
3 Create a configuration file using the vi openssl_ext.conf command.
   a Copy and paste the following OpenSSL commands into the configuration file.

   ```
   # openssl x509 extfile params
   extensions = extend
   [req] # openssl req params
   prompt = no
   distinguished_name = dn-param
   [dn-param] # DN fields
   C = US
   ST = CA
   O = VMware (Dummy Cert)
   OU = Horizon Workspace (Dummy Cert)
   CN = hostname (Virtual machine hostname where the Integration Broker is installed.)
   emailAddress = EMAIL PROTECTED
   [extend] # openssl extensions
   subjectKeyIdentifier = hash
   authorityKeyIdentifier = keyid:always
   keyUsage = digitalSignature,keyEncipherment
   extendedKeyUsage=serverAuth,clientAuth
   [policy] # certificate policy extension data
   
   Note Type the CN value before you save the file.
   ```
   
   b Run this command to generate a private key.

   ```
   openssl genrsa -des3 -out server.key 1024
   ```

c Type the passphrase for server.key, for example, vmware.

d Rename the server.key file to server.key.orig.

   ```
   mv server.key server.key.orig
   ```

e Remove the password associated with the key.

   ```
   openssl rsa -in server.key.orig -out server.key
   ```

4 Create a CSR (certificate signing request) with the generate key. The server.csr is stored in your working directory.

   ```
   openssl req -new -key server.key -out server.csr -config ./openssl_ext.conf
   ```
5 **Sign the CSR.**

```bash
openssl x509 -req -days 365 -in server.csr -signkey server.key -out server.crt -extfile openssl_ext.conf
```

The expected output displays.

```
Signature ok subject=/C=US/ST=CA/O=VMware (Dummy Cert)/OU=Horizon Workspace (Dummy Cert)/CN=w2-hwdog-xa.vmware.com/emailAddress=EMAIL PROTECTED
```

6 **Create P12 format.**

```bash
openssl pkcs12 -export -in server.crt -inkey server.key -out server.p12
```

a Press `Enter` at the prompt for an export password.

**IMPORTANT** Do not enter a password.

The expected output is `server.p12` file.

b Move the `server.p12` file to the Windows machine where Integration Broker is installed.

c From the Command Prompt, type `mmc`.

d Click **File > Add or Remove Snap-ins**.

e In the Snap-in window, click **Certificates** and click **Add**.

f Select the **Computer account** radio button.

7 **Import the certificate into the root and personal store certificates.**

a Choose **All Files** in the dialog.

b Select the `server.p12` file.

c Click the **Exportable** check box.

d Leave the password blank.

e Accept the defaults for the subsequent steps.

8 Copy the certificate into the Trusted Root CAs in the same `mmc` console.

9 **Verify that the content of the certificate includes these elements.**

- Private key
- CN in the subject attribute that matches the Integration Broker Host Name
- Extended key usage attribute with both client and server authentication enabled

### Synchronizing Workspace with Integration Broker

When you enable Citrix-published resource support, you establish communication and schedule the synchronization frequency between Workspace and the Citrix server farm.

**Prerequisites**

- Configure the network, SSL, and vCenter extension. See *Installing and Configuring Workspace* for information about configuring Workspace settings.
- Review Citrix documentation for Citrix server 5.0, 6.0, and 6.5 at the Citrix Web site.
- Set up dedicated Sync and SSO Integration Brokers to distribute the load in a large-scale enterprise deployment.
Procedure

1. Log in to the Connector Services Admin.
2. Click Published Apps - Citrix in the left navigation pane.
3. Select the Enable Citrix-based Applications check box.
4. Type the Sync Integration Broker host name and port number.
5. Select the Use SSL check box.

**NOTE**: Do not select Use SSL for both the Sync Integration Broker and the SSO Integration Broker.

6. Click Use same as Sync Integration Broker button if you do not need to distribute heavy traffic.
7. (Optional) Set up the SSO Integration Broker.
   a. Type the SSO Integration Broker host name and port number.
   b. Select the Use SSL check box.
   c. Add a server farm.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Version</strong></td>
<td>5.0, 6.0, or 6.5</td>
</tr>
<tr>
<td><strong>Server name</strong></td>
<td>Server name assigned in your environment.</td>
</tr>
<tr>
<td><strong>Servers (failover order)</strong></td>
<td>Organize the servers in failover order. Workspace respects this order during SSO and under failover conditions.</td>
</tr>
<tr>
<td><strong>Transport type</strong></td>
<td>HTTP, HTTPS, and SSL RELAY</td>
</tr>
<tr>
<td><strong>Port numbers</strong></td>
<td>HTTP Port, SSL Relay Port</td>
</tr>
<tr>
<td><strong>Deployment Type</strong></td>
<td>Select the type of deployment Workspace uses to make the Citrix-published resource available to users.</td>
</tr>
<tr>
<td></td>
<td>• User-Activated - Workspace adds Citrix resources to the App Center in the Workspace App Portal. To use the resource, users must move the resource from the App Center to their My Apps portal.</td>
</tr>
<tr>
<td></td>
<td>• Automatic - Workspace adds the resource directly to users’ My Apps portal for their immediate use.</td>
</tr>
</tbody>
</table>

8. Click Sync now.

At times, when you synchronize Integration Broker with SSL, the synchronization can be slow depending on factors in your environment, such as network speed and traffic.

**NOTE**: The anonymous user group feature in the Citrix product is not supported with Workspace.

The Sync now operation forces a synchronization between Workspace and Citrix-published resources, which includes Citrix-published applications and Citrix-published desktops.

After the synchronization is complete, Citrix-published resources and corresponding entitlements are synchronized with Workspace.

**What to do next**

End users can now add Citrix-published resources to the Workspace launcher and launch them.
View User and Group Entitlements to Citrix-Published Resources

You can see the Citrix-published applications and desktops to which your Workspace users and groups are entitled.

**IMPORTANT** You cannot use Workspace to make changes to your Citrix deployment. If a Citrix administrator makes any changes, such as entitling new users to a Citrix-published resource, or adding a new server farm, you must force a sync to propagate the changes to Workspace. Use the Workspace Connector Services Admin to force a sync.

**Prerequisites**

Verify that Workspace is integrated with your Citrix deployment. See Chapter 7, “Providing Access to Citrix-Published Resources,” on page 61.

Synchronize information, including entitlements, from your Citrix deployment to Workspace. You can force a sync with the following steps:

1. Log in to the Connector Services Admin.
2. Click **Published Apps - Citrix** and click **Sync Now**.

**Procedure**

1. Log in to the Workspace Admin Console.
2. View user and group entitlements to Citrix-published resources.

   Citrix-published resources include Citrix-published applications and Citrix-published desktops.

<table>
<thead>
<tr>
<th>Option</th>
<th>Action</th>
</tr>
</thead>
</table>
   | View the list of users and groups entitled to a specific Citrix-published resource. | a Click the Catalog tab.  
   | | b Click **Any Application Type > Citrix Published Applications**.  
   | | c Click the name of the Citrix-published resource for which you want to list entitlements.  
   | | The Entitlements tab is selected by default. Group entitlements and user entitlements are listed in separate tables. |
   | | View the list of Citrix-published resource entitlements for a specific user or group. | a Click the Users & Groups tab.  
   | | b Click the Users tab or the Groups tab.  
   | | c Click the name of an individual user or group.  
   | | The Entitlements tab is selected by default. Entitled Citrix-published resources are listed in the Citrix Published Applications table on the Entitlements page. |

**Editing Workspace Application Delivery Settings for Citrix-Published Resources**

You can use the Workspace Admin Console to edit the delivery settings of Citrix-published applications and desktops.

You can edit the delivery settings globally for all of the Citrix-published applications and Citrix-published desktops available from your Workspace deployment, or individually for specific Citrix-published resources.

You configure the delivery settings by editing Independent Computing Architecture (ICA) properties. ICA is a Citrix proprietary protocol. A wide range of ICA properties are available, controlling areas such as security, display, and compression. For more information about configuring ICA properties, see Citrix documentation.
Workspace includes default global settings that define how the configured Citrix deployment delivers Citrix-published resources to users. You can use the admin console to configure the default Workspace settings and to add new settings.

You can also use the admin console to specify delivery settings for individual resources. Settings for individual resources take precedence over global settings. When you provide ICA properties for the delivery of a specific resource, list all the properties necessary for the Citrix deployment to deliver the resource in the manner you expect. When delivery settings exist in Workspace for an individual resource, Workspace applies only those settings and ignores all global resource delivery settings.

**Edit the Workspace Resource Delivery Settings Globally for All Citrix-Published Resources**

You can use the Workspace Admin Console to edit the global settings for the Citrix-published applications and desktops in your Workspace deployment.

The ICA properties text boxes for the global resource delivery settings are populated with default settings until you edit them.

**Procedure**

1. Log in to the Workspace Admin Console as an administrator.
2. Select **Settings > Citrix Published Applications**.
3. Edit the ICA properties according to Citrix guidelines.
   - The ICA Client Properties and the ICA Launch Properties text boxes work together. They must both have content or they must both be empty.
4. Click **Save**.

Unless individual resources have their own resource delivery settings, your Citrix deployment applies the global ICA properties when it delivers Citrix-published resources available through Workspace to users.

**Edit the Workspace Application Delivery Settings for a Single Citrix-Published Resource**

You can use the Workspace Admin Console to edit the settings for individual Citrix-published applications and desktops in your Workspace deployment.

The ICA properties text boxes for individual applications are empty until you manually add properties. When you edit the application delivery settings, the ICA properties, of an individual Citrix-published resource, those settings take precedence over the global settings. You can configure the global ICA property settings from the Citrix Published Applications page by selecting **Settings > Citrix Published Application**.

**Procedure**

1. Log in to the Workspace Admin Console as an administrator.
2. Click the **Catalog** tab.
3. Click **Any Application Type > Citrix Published Applications**.
4. Click the name of the Citrix-published resources to edit.
5. Click **Configuration**.
6. View the information about the resource as carried forward from your Citrix deployment.
   - The page provides several details about the resource, such as the resource name, resource ID, server name, and so on. Also, the page displays information about the resources enablement. If the **Enabled** check box is not selected, the resource is disabled in your Citrix deployment.
7 If the **Enabled** check box is not selected and you want to hide the resource from users, select the **Hide When Disabled** check box.

8 In the ICA properties text boxes, add properties or edit existing properties according to Citrix guidelines.

   The ICA Client Properties and the ICA Launch Properties text boxes work together. They must both have content or they must both be empty.

9 Click **Save**.

Your Citrix deployment applies the ICA properties listed on the Workspace Configuration page of a resource when it delivers the Citrix-published resource to users.

**Managing Categories for Citrix-Published Resources**

You can use the Workspace Admin Console and your Citrix deployment to manage Citrix-published resource categories.

In your Citrix deployment, you give a Citrix-published application or desktop a category name by editing the **Client application folder** text box in the resource's properties. When you integrate your Citrix deployment with Workspace, existing category names for Citrix-published applications and desktops are carried over to Workspace.

After the integration, you can continue to create categories in your Citrix deployment. If you enabled the **Sync categories from server farms** check box on the Published Apps - Citirx page, the new categories are carried over to Workspace during the next sync. See “Synchronizing Workspace with Integration Broker,” on page 70.

You can also create categories directly in Workspace. See the **Workspace Administrator’s Guide** for information about using resource categories.

In the admin console, you can create and view categories of all Citrix-published resources by clicking the **Catalog** tab and clicking **Any Application Type > Citrix Published Applications**. You can view and edit the categories of a specific Citrix-published resource by clicking the name of the resource and selecting **Details**.

When you create a category in Workspace, the category never appears in your Citrix deployment.

When you create a category in your Citrix deployment, the category appears in Workspace at the next sync. When you update a category name in your Citrix deployment, the updated category name appears in Workspace while the original category name remains. If you want to remove the original category name from Workspace, you must remove it manually.
Troubleshooting Workspace Resource Configuration

You can troubleshoot issues that you or users experience after you configure Workspace resources. Refer to the troubleshooting problem that you are experiencing.

This chapter includes the following topics:

- “Blank Screen Displays When Installing Update to Workspace for Windows,” on page 75
- “ThinApp Packages Fail to Launch from the User Portal,” on page 76
- “Users Accessing Citrix-Published Resources Receive an Encryption Error,” on page 79
- “Citrix-Published Resources Are Not Available in Workspace,” on page 79
- “When Users Launch a Citrix-Published Resource, the Browser Displays 500 Internal Server Error,” on page 81
- “Memory Issue Prevents Proper Configuration of Integration Broker,” on page 82

Blank Screen Displays When Installing Update to Workspace for Windows

When you have the Workspace for Windows application already installed, and then you download a more recent version, and start to install it, a blank screen displays.

**Problem**

When you already have the Workspace for Windows application installed, and you start the installer for a more recent version, sometimes a blank screen displays and you cannot complete the installation.

When you look at the log files for the installer, you see lines at the end of file similar to the following lines:

```
20130918112739:INFO  CBootstrapCmd::LuaUIShow: calling StartSequence('reboot_prompt')
20130918112739:INFO  CHtmlUI::StartSequence: About to Run UI Sequence reboot_prompt 00000120 01F386F8 1
20130918112739:INFO  CHtmlUI::StartSequence: Running UI Sequence reboot_prompt
20130918112739:ERROR** CHtmlDialog::CallJScript: Failed to obtain required required jscript ID; COM Error: -2147352570
```

**Cause**

This problem can occur if one or more registry key settings are incorrect. For descriptions of similar symptoms and other potential solution steps, see the VMware knowledge base article at http://kb.vmware.com/kb/1027986 and the Microsoft Knowledge Base article 831430 at http://support.microsoft.com/kb/831430.
Solution

1  Reboot the Windows system, and then try running the installer again.
   If rebooting the system does not resolve the problem, continue with the next step.

2  Re-register the jscript.dll and vbscript.dll files.
   a  Open a command prompt on the Windows system.
   b  Run the following commands to re-register the necessary files and details.
      If the system is a 32-bit version of Windows, run these commands:
         regsvr32 jscript.dll
         regsvr32 vbscript.dll
      If the system is a 64-bit version of Windows, run these commands:
         cd C:\windows\syswow64
         regsvr32 C:\windows\system32\jscript.dll
         regsvr32 C:\windows\system32\vbscript.dll

3  Restart Windows.

ThinApp Packages Fail to Launch from the User Portal

When a user tries to launch a ThinApp package from the user portal, a browser message might appear that prompts the user to download and install Workspace for Windows even when the application is already installed and running.

Problem

After installing Workspace for Windows, when the user opens the user portal in a browser on that Windows system, logs in, and tries to launch a ThinApp package, a message might appear stating that Workspace for Windows must be installed on the system, and prevents the ThinApp package from starting. This message might appear even when the Workspace process is running on the Windows system. The Workspace for Windows client might report that all files are up to date.

Cause

This problem can occur for multiple reasons.
The Workspace for Windows browser plugin is not properly installed or it is not activated in the browser window for the browser in which the user is trying to launch the ThinApp package.

Because installation of the Workspace for Windows application is required to run ThinApp packages on the Windows system, the user portal uses a browser plugin to verify whether the application is installed before launching the ThinApp package from the user portal. When the user clicks the icon for a ThinApp package in the user portal, the Workspace for Windows browser plugin checks to see if the application is installed before launching the package. If the browser plugin is not installed and active in the browser, the verification cannot happen, the message appears, and the package does not launch.

If there are browser windows open during the Workspace for Windows installation process, the browser plugin might not be properly installed for that browser. The browser plugin might become deactivated in the browser if the user disabled the plugin in the browser's add-ons or plug-ins page.

The custom protocol handler used to launch the ThinApp package from the browser has been disabled for the browser in which the user is trying to launch the ThinApp package.

On the My Apps page in the user portal, ThinApp packages are represented using a link with a horizon:// protocol. When the Workspace for Windows client is installed, the installer registers a protocol handler for that horizon:// protocol. The protocol handler is an executable named HorizonThinAppLauncher.exe, and is registered as a handler by the registry entry HKEY_CLASSES_ROOT\horizon\shell\open\command. When the user tries to launch a ThinApp package from its icon in My Apps, this HorizonThinAppLauncher.exe application is launched.

If the user has disabled the use of all protocol handlers in the browser, or disabled the use of the handler for the horizon:// protocol, ThinApp packages will not launch using their icons in the My Apps page. Some browsers present a warning when protocol handlers are launched and give the user the option to select to execute the protocol handler. One way in which the user might have disabled the use of the horizon:// protocol handler is when the user clicked one of the ThinApp package icons for the first time, when the browser warning dialog appeared to ask for permission to run the protocol handler, the user selected No or a similar choice to prevent the launch, and also selected Remember my selection or a similar choice that prevents the launch for all such links. Because permission to run the protocol handler was not given and is remembered, none of the ThinApp packages launch from the My Apps page.

Solution

1. Verify the user has logged in to the Workspace for Windows application with the user's Workspace user account.
   The user signs into the client using the Workspace icon in the Windows system tray.

2. If this problem appears shortly after the application is installed on the system, close all open browser windows, reopen the browser, log in to the user portal, and try launching the ThinApp package.
If the problem appears even after closing the open browser windows and reopening the browser, verify the browser plugin appears in the browser's list of plugins and is active.

<table>
<thead>
<tr>
<th>Browser</th>
<th>Description</th>
</tr>
</thead>
</table>
| Internet      | For Internet Explorer, a COM server is registered instead of a browser plugin or add-on. To test whether the COM server is installed, create a test HTML file with the following contents and open that file in Internet Explorer. The result tells whether the COM server is installed or not.

```html
<html>
<script type="text/vbscript">
On Error Resume Next
dim objName
objName = "HorizonAgentFinder.HorizonFinder"
dim obj
Set obj = CreateObject(objName)
document.write(objName & " is ")
if IsEmpty(obj) then	document.write("not installed")
else	document.write("installed")
end if
</script>
</html>
```

Firefox       | Open Firefox's Add-ons Manager by clicking Tools > Add-ons. On the Plugins page, verify the VMware Horizon Agent Finder browser plugin is listed and set it to always activate. |

Chrome       | Open Chrome's content settings by opening the Settings page and clicking Show advanced settings > Content settings. Click Disable individual plug-ins to display the list of plugins. Verify the VMware Horizon Agent Finder browser plugin is listed and set it to always activate. |

Safari for Windows | Open Safari's list of installed plugins by clicking Help > Installed Plug-ins. Verify the VMware Horizon Agent Finder browser plugin is listed. Verify that plugin is activated for Safari. |

Verify the registry entry \`HKEY_CLASSES_ROOT\horizon\shell\open\command\` exists and has a value that is a path that points to the location of the required protocol handler, named HorizonThinAppLauncher.exe, where Workspace for Windows was installed on the Windows system.

If the registry entry does not exist, or does not have a value that points to the location where Workspace for Windows was installed, uninstall Workspace for Windows and reinstall it.

If the registry entry exists and has a value that points to the location of the HorizonThinAppLauncher.exe executable, verify the executable exists at that location and has not been moved or deleted.

If the registry entry does not exist, or does not have a value that points to the location where Workspace for Windows was installed, uninstall Workspace for Windows and reinstall it.

If the registry entry exists and has a value that points to the location of the HorizonThinAppLauncher.exe executable, verify that the (Default) value for the registry entry \`HKEY_CLASSES_ROOT\horizon\` has a Data value of URL:horizon Protocol and that the URL Protocol value for the \`HKEY_CLASSES_ROOT\horizon\` entry exists.

If the Data value for the (Default) value of the \`HKEY_CLASSES_ROOT\horizon\` registry entry is not set to URL:horizon Protocol, update the Data value to set it to URL:horizon Protocol. If the URL Protocol value does not exist for the \`HKEY_CLASSES_ROOT\horizon\` entry, you can create it using a value name URL Protocol and no value data.
Determine if the user disabled the `horizon://` protocol for the browser, or if all protocol handlers are disabled in the browser, and if so, enable the protocol handler for the browser as appropriate for your organization’s needs.

In most situations, the browsers rely on the settings in the registry for information about the protocol handlers available for that Windows system. For some browsers, when the user clicks a link that is associated with a protocol handler, a dialog prompt appears that asks the user a question such as **Do you want to allow this website to open a program on your computer?** or **This link needs to be opened with an application** or a similar statement about needing to launch an external application to handle the link. Typically, the dialog provides the user with the option of not launching the external application and to remember that choice for all links of that type. The steps to re-enable the ability to launch the application associated with the protocol handler are usually different depending on the browser type. Consult the documentation for the user’s type of browser on how to enable protocol handlers for that browser type.

### Users Accessing Citrix-Published Resources Receive an Encryption Error

The XenApp ICA properties on the Workspace server must include the encryption property set to the same encryption level as configured on the XenApp servers in the farm, otherwise users cannot access their Citrix-published applications or desktops.

**Problem**

When a user connects to a Citrix-published resource from Workspace, the following error message is displayed.

**You do not have the proper encryption level to access this Session**

**Cause**

Workspace does not set encryption levels. If the encryption level on the XenApp server is set higher than the default setting used in the Citrix-Receiver, users see this error.

**You must set a higher encryption level in Workspace.**

**Solution**

1. Log in to the Workspace Admin Console.
2. Select **Settings > Citrix Published Applications**.
3. Edit the **ICA Client Properties** text box. To set the encryption level to 128, enter `EncryptionLevelSession=EncRC5-128`.
4. Edit the **ICA Launch Properties** text box. To set the encryption level to 128, enter `[EncRC5–128]

   ```
   DriverNameWin16=pdc128w.dll
   DriverNameWin32=pdc128n.dll
   ```

### Citrix-Published Resources Are Not Available in Workspace

A communication issue between Integration Broker and PowerShell SDK might prevent Citrix-Published Applications and Desktops from appearing in the Workspace Catalog.

**Problem**

After you integrate Citrix with Workspace, Citrix-published resources do not appear in the Workspace Catalog.
Cause

A configuration issue might exist in Integration Broker that prevents proper communication with PowerShell SDK.

Solution

You can specify URLs in a browser to troubleshoot where an Integration Broker configuration issue exists. This troubleshooting method can help you identify if the problem is a configuration issue in the following areas.

- The Citrix server farm
- Citrix-published resources
- Resource entitlements

If a Web page does not display the expected output, it displays an error and adds information to the Integration Broker logs. Review the Integration Broker logs to continue the troubleshooting process.

Procedure

1. Use a browser to check the Integration Broker configuration of the Citrix server farm.
   a. In a browser, enter a URL such as one of the following, replacing the place holders with the appropriate information.
      - Citrix Server Farm 6.5
      - Citrix Server Farm 5.5 or 6.0
   b. Review the content of the Web page and, if necessary, review the Integration Broker logs.

   If Integration Broker is properly configured, the page displays Citrix server farm information, such as the following.

   "[{"FarmName":"test data","ServerVersion":"6.0.6410","AdministratorType":"Full","SessionCount":"2","MachineName":"test data"}]"

   If the Web page does not display the server farm information, log information is sent to the Integration broker. To further troubleshoot the issue, review the logs on the Integration Broker host at %programdata%/VMware/HorizonIntegrationBroker.
2 Use a browser to check the Integration Broker configuration syncing Citrix-published resources in the server farm with Workspace.
   a In a browser, enter a URL such as one of the following, replacing the place holders with the appropriate information.
      - Citrix Server Farm 6.5
        https://IBhostname/IB/API/RestServiceImpl.svc/hznxenapp/admin/applications?
        computerName=XenAppServerHostname&xenappversion=Version65orLater
      - Citrix Server Farm 5.5 or 6.0
        https://IBhostname/IB/API/RestServiceImpl.svc/hznxenapp/admin/applications?
        computerName=XenAppServerHostname&xenappversion=Legacy
   b Review the content of the Web page and, if necessary, review the Integration Broker logs.
      If Integration Broker is properly configured, the page displays a list of all the resources in the Citrix server farm.
      If the Web page does not display a list of resources, log information is sent to the Integration broker. To further troubleshoot the issue, review the logs on the Integration Broker host at %programdata%/VMware/HorizonIntegrationBroker.

3 Use a browser to check the Integration Broker configuration of the entitlements for a Citrix-published resource.
   You can test a single Citrix-published resource.
   a In a browser, enter a URL such as one of the following, replacing the place holders with the appropriate information.
      Replace the ApplicationName place holder with the name of the application you are specifying.
      - Citrix Server Farm 6.5
        https://IBhostname/IB/API/RestServiceImpl.svc/hznxenapp/admin/entitlements?
        computerName=XenAppServerHostname&xenappversion=Version65orLater&appName=ApplicationName
      - Citrix Server Farm 5.5 or 6.0
        https://IBhostname/IB/API/RestServiceImpl.svc/hznxenapp/admin/entitlements?
        computerName=XenAppServerHostname&xenappversion=Legacy&appName=ApplicationName
   b Review the content of the Web page and, if necessary, review the Integration Broker logs.
      If Integration Broker is properly configured, the page displays a list of all the entitlements for the application you specified.
      If the Web page does not display a list of entitlements, log information is sent to the Integration broker. To further troubleshoot the issue, review the logs on the Integration Broker host at %programdata%/VMware/HorizonIntegrationBroker.

When Users Launch a Citrix-Published Resource, the Browser Displays 500 Internal Server Error

A mismatch between the configurations of the Citrix server farm and Workspace might cause the launch of Citrix-published resources to fail.

**Problem**

Launching a Citrix-published resource fails as the browser displays 500 Internal Server Error.
Cause

A 500 error occurs when the Citrix server farm information provided in the Connector Services Admin does not match the Citrix server configuration.

Solution

1. Note the settings of the transport type, port number, and SSL relay port number of each server farm integrated with your Workspace deployment.
2. Log in to the Connector Services Admin.
3. Click Published Apps - Citrix.
4. In the Server Farms section, change the Transport type, Port, and SSL Relay Port settings for each server farm to match the settings in your Citrix server configuration.

Memory Issue Prevents Proper Configuration of Integration Broker

When you integrate Workspace with Citrix server farm versions 6.0 and earlier, insufficient memory allotted to PowerShell SDK results in an error.

Problem

When you issue the Invoke-Command command to verify PowerShell remoting, an error related to insufficient memory appears. You are instructed to issue the Invoke-Command command during “Prepare Integration Broker Server for Windows Server 2008 or Windows Server 2012,” on page 66.

Cause

On the Windows system where PowerShell remoting is executed, the memory allotted to PowerShell SDK might be insufficient for the number of Citrix-published resources.

Solution

You can increase the memory allotted to the PowerShell SDK.

Procedure

1. When the error appears, issue the command to increase the allotted memory. For example,
   `winrm set winrm/config/winrs '@{MaxMemoryPerShellMB="1024"}'`
2. Reissue the Invoke-Command command and complete the task.
Index

Numerics
500 Internal Server Error 81

A
access policy sets, Web-application-specific 13, 15, 16
Active Directory, join 20
add desktop client applications to virtual appliances 58
administrator consoles 7
Appliance Configurator 7
application records
Web applications 15
Web applications from the cloud application catalog 13
applications, Web 9, 13, 15, 16
automatic client application update directory 58

C
catalog, Web applications 12
Citrix deployment 61
Citrix server 64, 81
Citrix XenDesktop 61
Citrix PowerShell Remoting 64, 65
Citrix Receiver 61
Citrix server farm 65, 81
Citrix-published applications 61, 72–74, 79
Citrix-published desktops 61, 72–74, 79
Citrix-published resources 61, 72–74, 79, 81
Citrix-published resource support 70
client downloads directory 58
cloud application catalog, Web applications 13
configure, SAML authentication 22
configure SAML protocol 9
configure Web applications 9
Connector Services Admin 7

D
desktop systems, configuring 51

E
enable 24
end-user console 7
entitlements
ThinApp packages 35
Web applications 17
example 69

global resource settings 73

H
HTTPS on IIS 68

I
ICA properties 72, 73, 79
Independent Computing Architecture 72
install 67, 68
installation options, Windows application 51
Integration Broker 66, 68, 70, 79, 82
Integration Broker Server 67

J
join, Active Directory Domain 20

M
multiple desktop systems, deploying the Windows application 57
multitenant Web applications 9

N
network share, ThinApp packages 32

O
Office 365
convert to Workspace authentication 11
example of Powershell cmdlet 11
mapping attributes to AD user attributes 10
powershell 11

P
PowerShell Remoting 64
PowerShell remoting 82
PowerShell SDK 79, 82

S
SAML 9, 13, 15
SAML authentication 22
self-signed certificate 69
single sign-on to Web applications 9
single sign-on, Office 365 11
SSL trust, update on View Connection Server 22
sync, View Connection Server 20

T
ThinApp, packages 28
ThinApp applications 28
ThinApp download mode 37
ThinApp packages
change share folder 49
configuring access to 33
deleting 46
deploying Workspace for Windows 57
enabling for management 47
entitling to users and groups 35
HorizonThinAppCtrl.exe 40
modifying the offline grace period 40
network share repository 32
offline grace period 39
overview 37
providing access 27
setting options when installing the Windows application 51
upgrading 41–43, 45
ThinApp streaming mode 37
troubleshooting
launching a ThinApp package 76
updating Workspace for Windows 75

U
URLs 7
users
View Hosted Applications 23
viewing entitlements to View desktop pools 23

V
View 19
View entitlements 23
View application pools 23
View Connection Server 23
View desktop pools View desktop pools, providing access 19
View desktop
allow reset 24
start 24
View desktop pools, See View desktop pools
View Hosted Application 23
View pods 21

W
Web applications
adding 9, 12