

RES SOFTWARE

Installing XenApp 4.5 Platinum Edition with RES Wisdom

Brian Casselman

6/10/2009

This document will guide you through creating the Resources and Modules to perform a RES Wisdom installation of XenApp Platinum 4.5 Edition on a Windows Server 2003 and join the new server to an existing farm hosted on SQL server 2005 (a Direct-Join installation).

Table of Contents

Prepare the Installation files.....	3
RES Wisdom Resources.....	4
RES Wisdom Resource Package	4
(3)RES Wisdom Resource Located on a Fileshare.....	8
(3)RES Wisdom Datastore resource.....	10
(2)RES Wisdom Modules.....	11
(3)Module – XenApp Platinum Prerequisites	13
Module – XenApp Platinum 4.5	21
(2)RES Wisdom Projects.....	29
(1)Summary.....	32

The goal of this document is to outline the steps for installing and configuring XenApp 4.5 Platinum edition on a Windows Server 2003 machine using RES Wisdom after a base image or OS install is performed. This document will guide you through creating the Resources and Modules to perform a RES Wisdom installation of XenApp Platinum 4.5 Edition on a Windows Server 2003 running Terminal Services and join the new server to an existing farm hosted on SQL server 2005 (a Direct-Join installation).

A quick note...This document only covers the XenApp components. Although RES Wisdom can perform tasks like, joining a Domain, rename the computer, installing applications, enabling Terminal Services, etc, these tasks are outside the scope of this document.

NOTE: This document is designed as a training guide. This is **NOT** a RES Software best practice guide. It is designed to show examples of the many ways to automate Tasks and build Resources. You should use whichever method works best for you in your particular situation and what matches up with the needs of your company.

This document assumes that you have familiarity with Microsoft Windows and a basic knowledge of RES Wisdom and Citrix XenApp server.

One thing to keep in mind when creating all Resources and Tasks in RES Wisdom...since you are building Modules and Projects to delegate to other administrators or share amongst your colleagues, take the time to be descriptive in your comments so that others, and yourself after a few months have passed, know exactly what the Resources, Tasks, Modules, or Projects are for.

Prepare the Installation files

In order to use RES Wisdom to install XenApp, there are a few files that need to be created and added to the RES Wisdom database for use on the XenApp server to be. These files are as follows:

- UnattendedInstall.exe – This is the executable that is used to perform an unattended installation of Citrix XenApp. This file is located on the XenApp server CD in the Support\Install directory.
- UnattendedTemplate.txt – This is an answer file to provide answers to the questions asked when you run Setup. This file is located on the XenApp server CD in the Support\Install directory.
- DSN file – A template DNS file sampled from an existing XenApp server already in the farm.
- SetPSEdition.wsf – This is a file obtained from the XenApp server CD in the Support\Install directory. It is used to set the Edition of the install to “Platinum” after the unattended install completes.

CAUTION: An unattended installation performed with the Platinum installation media can only set the server edition to Platinum. Conversely, an unattended

installation using the Enterprise/Advanced media can only set the server edition to Enterprise, Advanced, or Standard.

WARNING: Even though there is an option in the UnattendedTemplate.txt file to set the server edition to “Platinum”, experience has shown that it does not recognize this line item during the install and it sets the values to “Enterprise”. After the server reboots, you will receive an error message stating that the incorrect licenses are install on the license server. Further details can be found at <http://support.citrix.com/article/CTX117079>.

RES Wisdom Resources

Resources are files that are used in the execution of RES Wisdom Tasks. This section will cover building all the Resources needed for installing XenApp Platinum 4.5 Edition in a RES Wisdom environment. For example, if you create a Task - **Perform Unattended Installation** to install an application, you first need to store the executable so that RES Wisdom can access it. There are three types of Resources:

Resources consisting of a set of files can be stored as a:

- **RES Wisdom Resource Package** – A Resource containing multiple files that is stored directly in the datastore.

Resources consisting of a single file can be stored as a:

- **File share** - Requires credentials for access to the share each time the Task is run. Availability of the Resource also depends on availability of the share but does not consume space in the datastore. Ideal for large installation files.
- In the **RES Wisdom Datastore** - This Resource is stored directly in the datastore and can be used repeatedly without requiring additional credentials.

RES Wisdom Resource Package

The first Resource that needs to be created is a **RES Wisdom Resource Package**. As mentioned above, this type of resource contains multiple files.

Open the RES Wisdom Management Console, select the **Resources** node in the left pane, and click **Add**. Create a new Resource of type ‘**RES Wisdom Resource Package**’ that will contain the four files mentioned previously. In the example, the Resource is named “**XenAppPlatinum_UnattendInstallFiles**”. On the **Contents** tab select the **Change** button in the lower right corner and add the **UnattendedInstall.exe** file in the Support\Install directory from the XenApp server media. This can be

accomplished either by choosing the **Add** button, browsing to the file, and clicking **OK** or simply by drag-and-drop the file from Windows Explorer into the right pane of the **Add/Edit RES Wisdom Resource Package** dialog box (Figure 1).

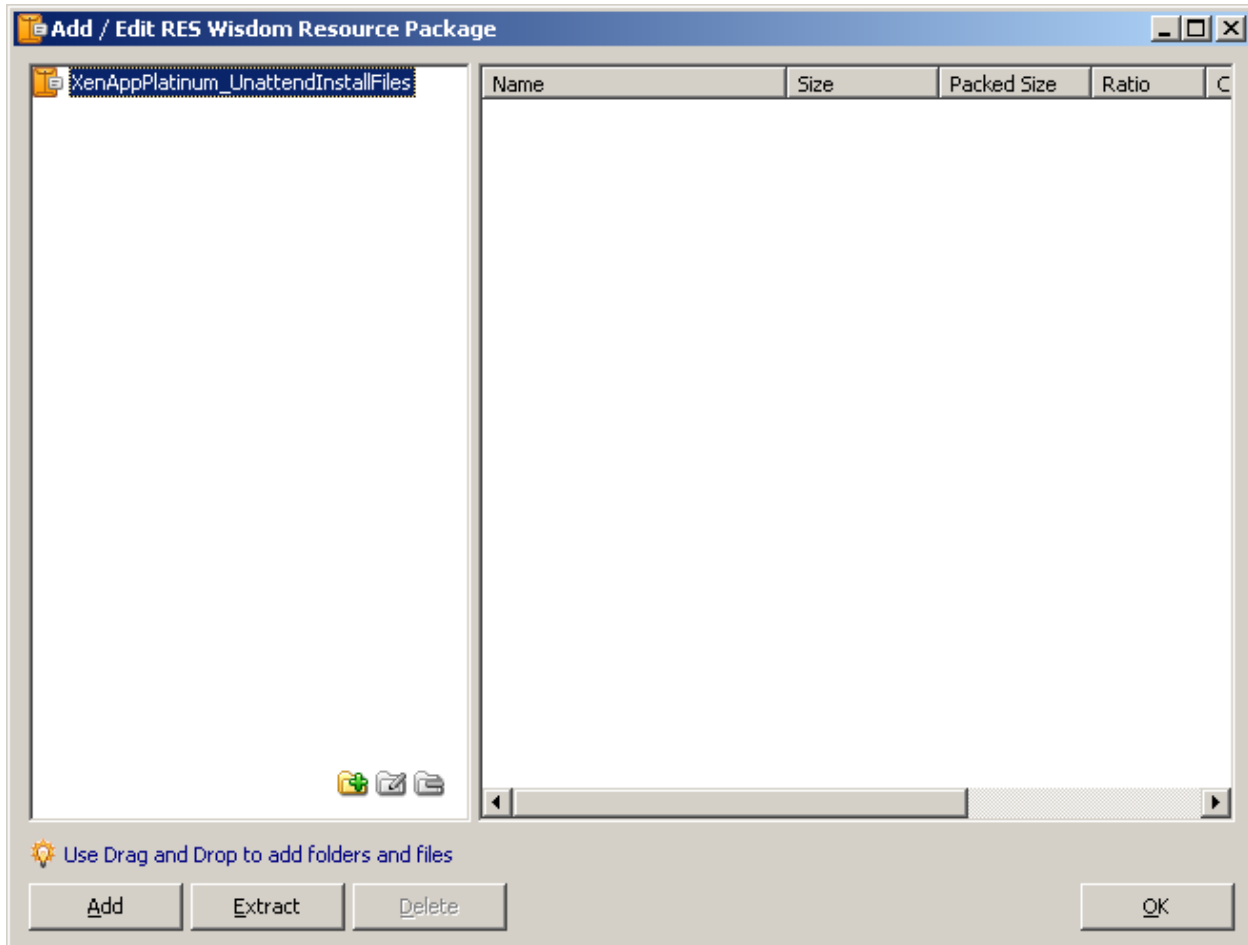


Figure 1: RES Wisdom Resource Package dialog

Follow the same procedure and add the **SetPSEdition.wsf** file in the Support directory of the install media. If you do not have the file, see **Appendix A**. Copy and paste it into a text editor and save as **SetPSEdition.wsf**.

Next, create the answer file for the unattended installation. Browse to the Support\Install directory on the XenApp server media and make a copy of the **UnattendedTemplate.txt** file for editing. In this example, the file is named "**Unattended_CPS45_RESWisdom.txt**". Changes to the file will be discussed following the text. In case you do not have the file, see **Appendix B**, copy and paste it into a text editor and save as **Unattended_CPS45_RESWisdom.txt**.

The following represents changes made to the file for this example. For clarity, comments have been removed so just the configurations remain. These are the settings for joining a new XenApp server to an existing farm running SQL 2005. Make the necessary changes to reflect the settings specific to the environment that RES Wisdom will be installing into.

NOTE: When editing script files, answer files, and command lines, use a text editor like Notepad to avoid problems with quotation marks and other punctuation that can cause problems during execution.

AS A REMINDER: Specifying passwords in the unattended answer file is not supported for security reasons. You must use the CTX_ODBC_PASSWORD command-line parameter to specify the password that can authenticate to this database. This will be covered in more detail in the Modules section.

```
[Citrix License Agreement]
AcceptLicense=Yes

[Data Store Configuration]
CreateFarm=No
DirectConnect=Yes
*Leave this blank to use the default zone name
ZoneName=

[Direct Connect Settings]
DSNFilePath="C:\Temp\TSFiles\MF20SQL.DSN"
UserName=CitrixDBA

[Shadowing Restrictions]
AllowShadowing=Yes
ProhibitRemoteControl=No
ProhibitNotificationOff=Yes
ProhibitLoggingOff=No

[Citrix XML Service]
ExtendIIS=No
* This setting applies only if ExtendIIS is No
DedicatedPortNumber=8081
* This setting applies only if ExtendIIS is Yes
*EnableVirtualScripts=Yes

[Options]
RebootOnFinish=Yes
LogLevel=cewup
LogFile=c:\msi.log
UILevel= REDUCED_UI
IgnoreMCM=No
RemoveWITurnkey=No

[LicenseServer]
LicenseServerChoice=UseFarmSettings
```

```
[MFRDP]
DisableRDPPromptForPassword=Yes

[IMAEncryption]
EncryptionEnable=0
KeyType=file
NewKeyPath=
KeyPath=
```

Pay particular attention to is the line:

```
DSNFilePath="C:\Temp\TSFiles\MF20SQL.DSN"
```

This will come into play later on when building the RES Wisdom Module for installing XenApp, so keep track of this setting. Customize this file and enable additional settings as required for your build but these are the minimums needed for a basic installation. Consult the organization's written policies and procedures for settings specific to each deployment.

Now, create a DSN file that will be copied to the server into the **%ProgramFiles%\Citrix\Independent Management Architecture** directory during the installation of XenApp. The easiest method for doing this is to browse to this directory on a pre-built member server in the existing farm and copy the **MF20.DSN** file. Before adding this file to the RES Wisdom Resource Package, edit the file and make the following changes to the bolded lines and substitute in the variables shown:

```
[ODBC]
DRIVER=SQL Server
UID=%CTX_ODBC_USER_NAME%
WSID=%ComputerName%
APP=Citrix IMA
SERVER=<SQL Server Name>
Description=Citrix Datastore
```

Save the file and upload it into the RES Wisdom Resource Package using the same procedure as before. Rename the file to make it easier to keep track of. In this example, the file is named **MF20SQL.dsn**. During the install of XenApp the variables will be replaced with the proper username and computer name based on the answer file created earlier. The **%CTX_ODBC_USER_NAME%** setting is replaced by the "UserName" field under the **[Direct Connect Settings]** section of the Unattended_CPS45_RESWisdom.txt file and the **%computername%** will be replaced by the system variable from Windows. Once populated with the proper variables, the file will be renamed to the default instance of **MF20.dsn** and copied to the **%ProgramFiles%\Citrix\Independent Management Architecture** directory.

Once completed the Resource should look like (Figure 2).

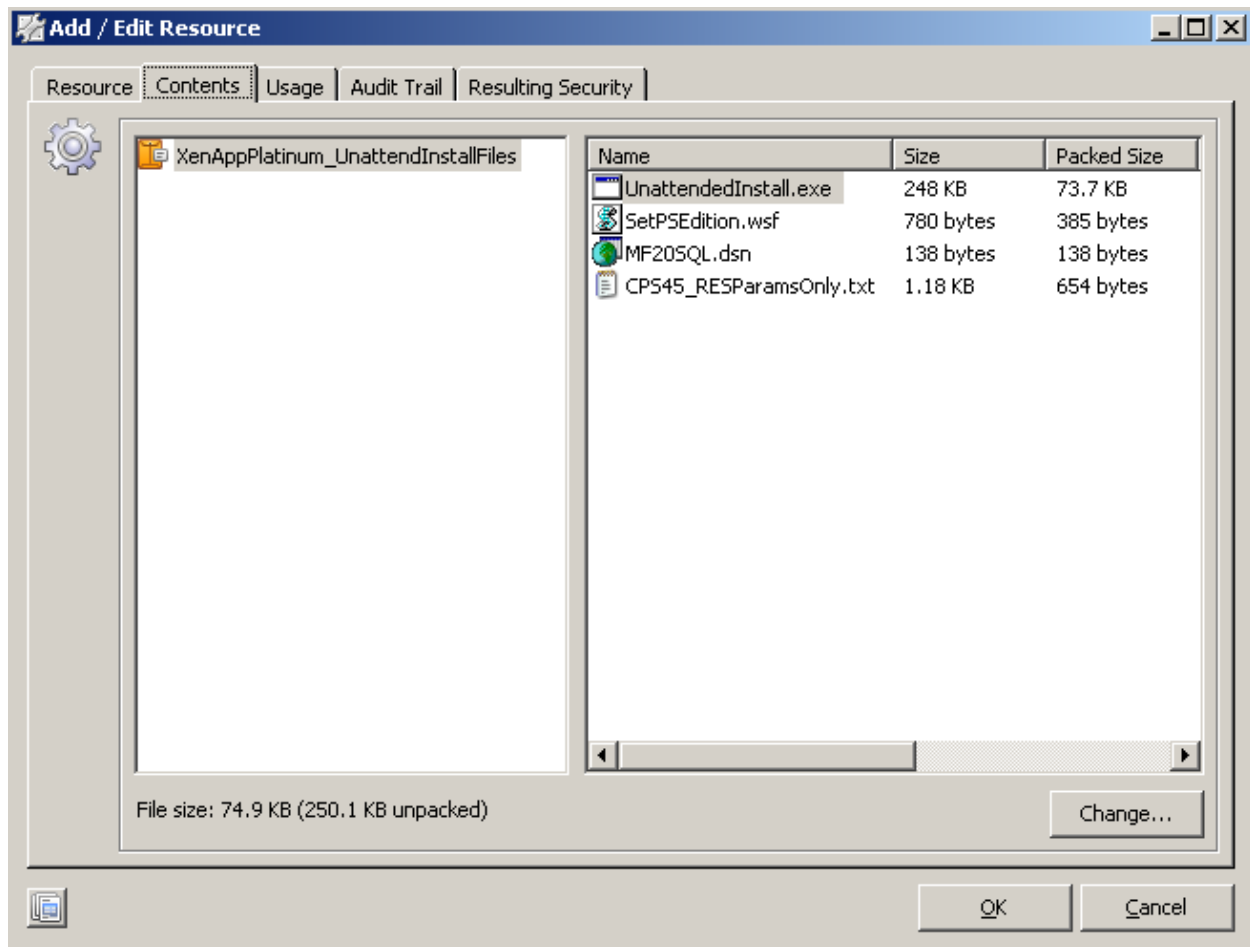


Figure 2: Completed RES Wisdom Resource Package for the Unattended Installation Files

This completes the creation of the RES Wisdom Resource Package for use in installing XenApp. In the next section we will move on to the next resource needed for the installation.

(3) RES Wisdom Resource Located on a Fileshare

RES Wisdom Resources located on a fileshare are perfect for large installation files or groups of files stored on the network. This will save space in the datastore and will help offload the database server during installation. For organizations that have a pre-existing central software repository, this can be a good resource type. Fileshare resources makes it easier to distribute Building Blocks for Projects or Modules since there will be no files to upload during creation.

Some things to keep in mind when using Fileshare Resources are that bandwidth control cannot be used during the upload process to the Dispatcher and the Agents. This resource type can be faster, but will require credentials for access to the share each time the Task is run, therefore it must exist in the same domain as the target machines in. Availability of the Resource also depends on availability of the share.

To create a Fileshare Resource:

Open the RES Wisdom Management Console, select the **Resources** node, and click **Add**. Create a new Resource of type **'Located on Fileshare'**. Browse to the share on the network where the MSI files for the Citrix installation are stored (Figure 3).

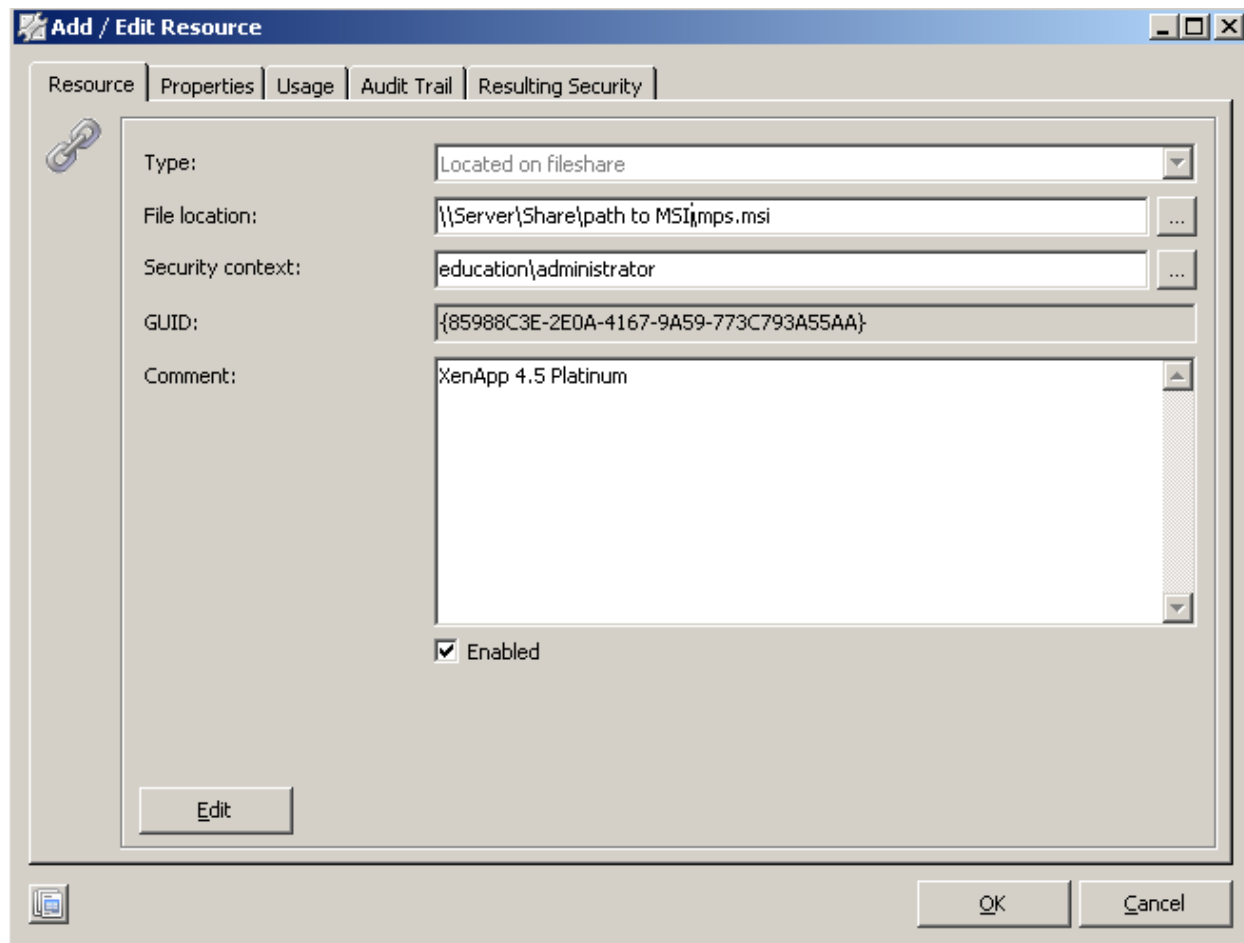


Figure 3: Completed Fileshare resource

NOTE: You can choose to store the Citrix installation files in the RES Wisdom database instead of a file share but all files stored in the datastore will contribute to increasing the size of the database.

Fill in the credentials for accessing the file share and your description and save the resource by clicking OK.

(3)RES Wisdom Datastore resource

The last Resource needed for the XenApp installation will be stored in the datastore directly. This will take a bit longer to queue up for the installs since it is first copied up to the Dispatcher's cache, but it does not require credentials for access and is readily available for repeated use.

When installing XenApp directly from the mps.msi, items like the Access Management Console, the prerequisites, and console snap-ins will not be installed. At the time of this writing, other methods of installing XenApp have not been attempted. For now, a couple of additional Resources are needed to complete the installation of XenApp server: Prerequisites (JRE and .NET) and the Access Management Console.

The installation files for these applications are located on the CD or media share and are named **CtxInstall.exe** (located in Administration\Access Management Console folder), **jre-1_5_0_09-windows-i586-p.exe** (located in Support\JRE1.5 folder), and **dotnetfx.exe** (located in Support\DotNet20 folder).

Open the RES Wisdom Management Console, select the **Resources** node, and click **Add**. Create a new Resource of type '**Stored in datastore**'. Browse to the location of the Java Runtime installer file, in this case, **jre-1_5_0_09-windows-i586-p.exe** (located in Support\JRE1.5 folder of the CD), and click **Open**. Enter a description for the Resource and click **OK**.

NOTE: The **GUID** for the resource will be generated once you are finished and click **OK**. The final product looks like Figure 4.

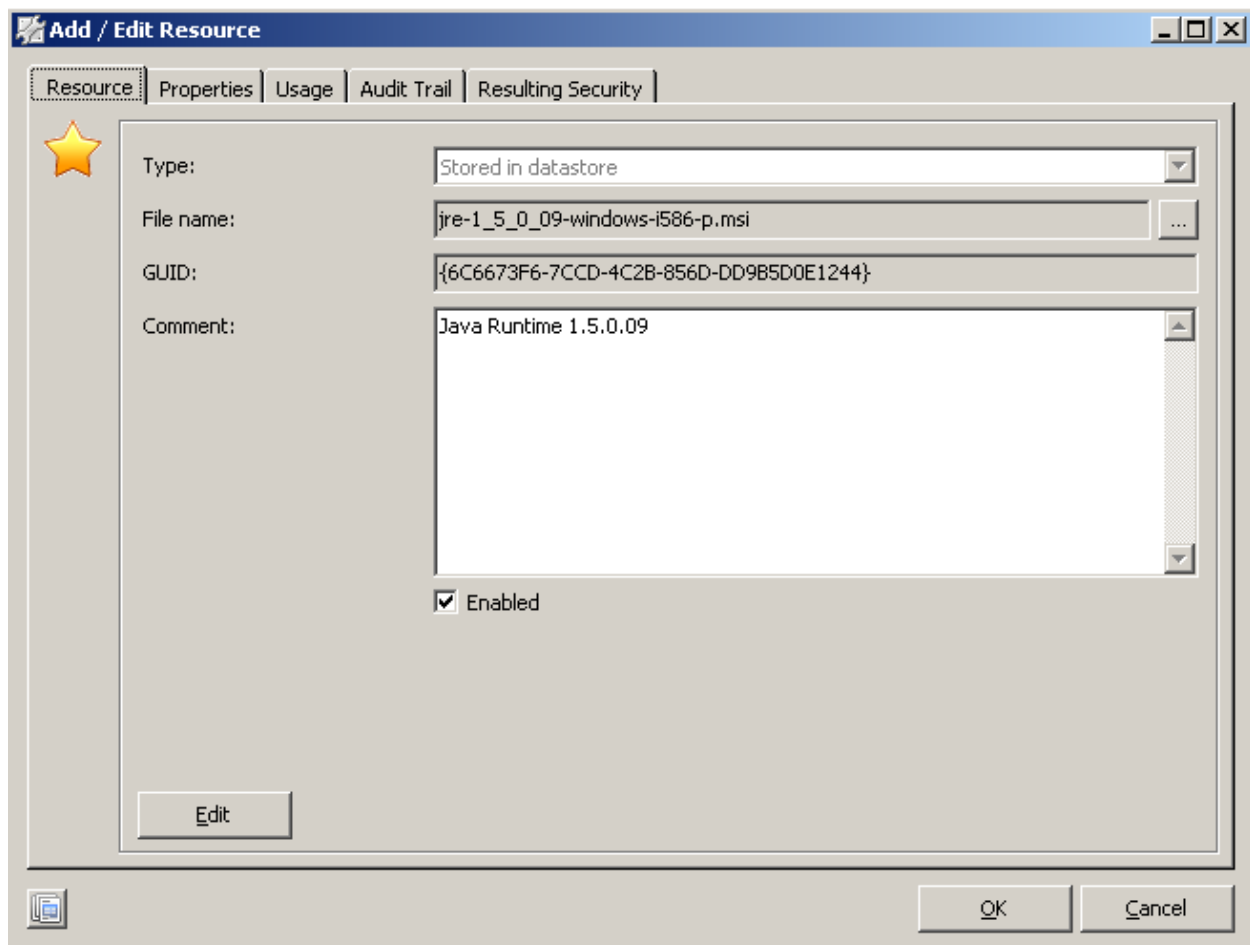


Figure 4: Java Runtime Resource

Repeat this process for the remaining files, **CtxInstall.exe** (located in Administration\Access Management Console folder), and **dotnetfx.exe** (located in Support\DotNet20 folder).

This completes the Resource creation for installing XenApp.

(2)RES Wisdom Modules

Once all the Resources are created, we need to do something with them. This is where **Modules** come in. Modules are containers for the **Tasks** that you can execute on Agents. Modules hold information about the way in which these Tasks should be executed, such as order of execution, parameters and conditions. You can add multiple Tasks to a single Module.

Tasks are actions that can be performed on an **Agent**. With a Task you can, for example, install software on Agents or manage users and groups in Active Directory, but also for example, query security risks on

Agents or query mailboxes on Exchange servers. Tasks leverage Resources to make things happen, in our example to install XenApp. See Table 1 for additional Tasks that are available.





 Computer Properties (Manage, Query)
 Environment Variables (Set, Delete, Query)
 Files (Perform Operations, Query)
 Microsoft Product Key (Set, Query)
 Printer Driver (Add, Remove, Query)
 Registry Settings (Apply, Query)
 Remote Terminal Server Logons (Change, Query)
 Service Properties (Manage, Query)
 Share (Add, Remove, Query)
 TCP/IP Properties (Query)
 Windows Shell (Set, Query)
 Active Directory Computer (Create, Manage, Delete, Query)
 Active Directory Group (Create, Manage, Delete, Query)
 Active Directory Object (Move, Query)
 Active Directory Organizational Unit (OU) (Create, Manage, Delete, Query)
 Active Directory User (Create, Manage, Delete, Query)
 Exchange Mailbox (Create, Manage, Move, Export, Delete, Query)
 RES PowerFuse Workspace Manager (Refresh)
 Resource (Download)
 SoftGrid Client (Invoke, Query)
 Unattended Installation (Perform)
 VMware ESX Virtual Machine (Create, Manage, Delete, Snapshot, Query)
 Windows Installer Package (Install, Repair, Patch, Remove)
 Command (Execute)
 E-mail (Send)
 Hosts (Discover)
 Message Box (Show)
 Parameters (Query)
 RES Wisdom Component (Deploy, Repair, Remove)
 RES Wisdom Dispatchers (Discover)
 Secure Shell (SSH) Commands (Execute)
 SQL Statement (Execute, Query)
 Wait (Perform)
 Windows PowerShell Script (Execute)
 Baseline Security (MBSA) (Query)
 Certificate (Import, Delete, Query)
 File Permissions (Set, Query)
 Local Group (Create, Manage, Delete, Query)
 Local User (Create, Manage, Delete, Query)
 Microsoft Update (Install)
 Registry Permissions (Set, Query)
 Share Permissions (Set, Query)
 Windows Firewall Settings (Manage, Query)
 Citrix Published Applications (Query)
 Computer (Reboot, Shutdown, Inventory, Uptime)
 Disk Fragmentation (Defragment, Analyze)
 Disk Space (Query)
 Event Logs (Query)
 Installed Programs (Query)

Table 1: Wisdom Tasks

Agents execute RES Wisdom **Jobs** on the devices where they are installed. Jobs are Projects or Modules that have been scheduled to run on Agents. Agents contact a **Dispatcher** at regular intervals to check whether a new Job is available. If so, it will download all necessary data from the Dispatcher, and then it will execute the Job.

A **Dispatcher** is a computer that communicates information from the Datastore to Agents. Dispatchers contact the Datastore at regular intervals. If new Jobs are available for Agents to which it connects, the Dispatcher will download all necessary data from the Datastore and store it. The Agents will use this data to execute the Jobs.

(3)Module – XenApp Platinum Prerequisites

The first module that needs to be created installs the XenApp prerequisites. Open the RES Wisdom Management Console, select the **Modules** node in the left pane, and click **Add**. Give the module a name. In this example, the Module is named “**Software, Citrix, Deploy Citrix XenApp Server 4.5 Prerequisites**”.

TIP: When naming Modules and Projects, choose names that reflect the functions and container names where the Modules are stored in the folder list. For example, if the module is going to install XenApp on a Windows Server 2003 machine connecting directly to a SQL server, the name might be: **Software, Citrix, Deploy XenApp 4.5 Platinum - Join, Direct** (use commas to split out the functions). From this name, you can easily identify the module’s purpose. It is a **Software** installation of a **Citrix** product which is to **Deploy XenApp Platinum 4.5** and **Joins** it to an existing farm that uses a **Direct** connection to the database.

Next, choose the **Tasks** tab and click **Add** to bring up the Select Task dialog (Figure 5)

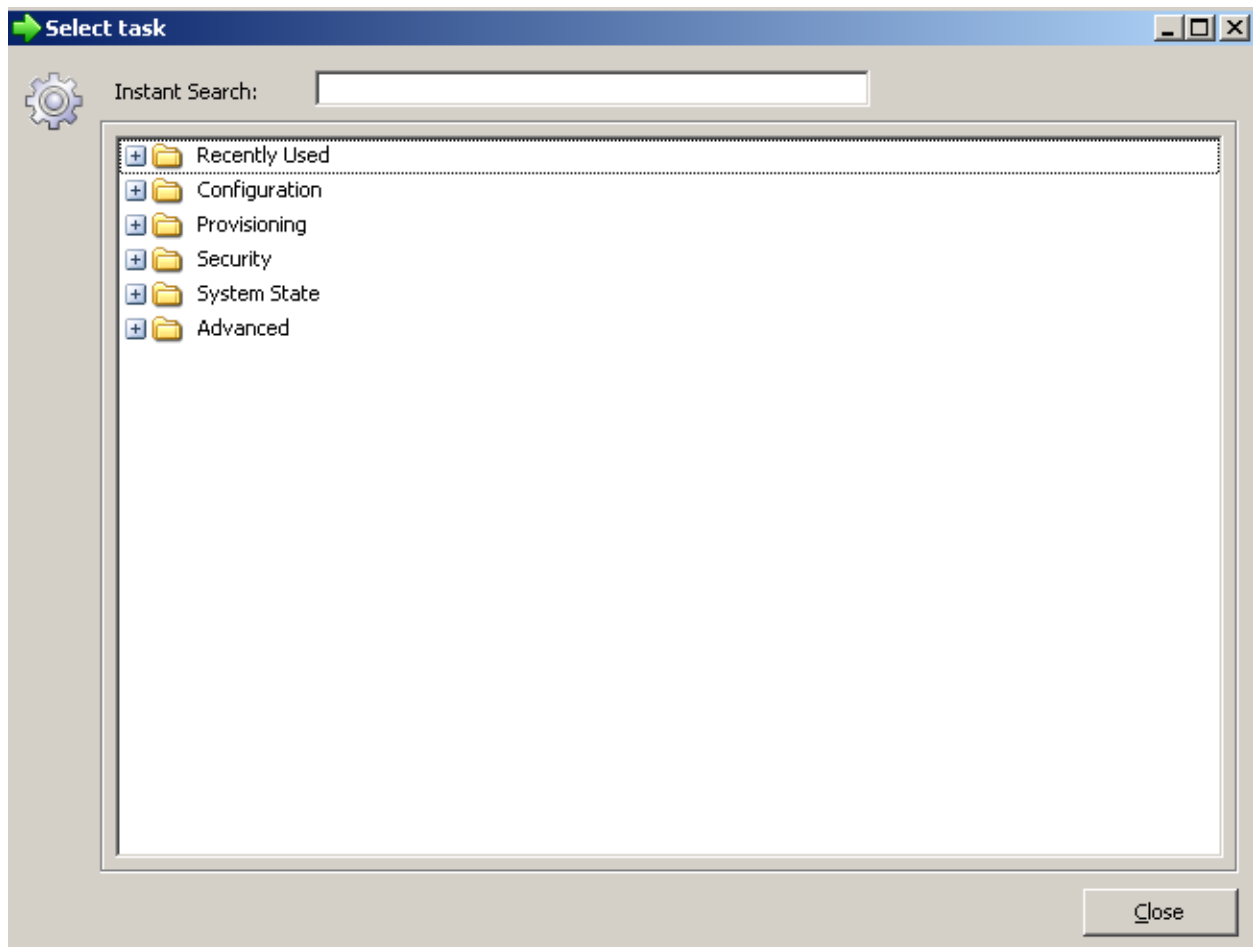


Figure 5: Select Task Dialog Box

This first Module installs the XenApp prerequisites, starting with the Java Runtime Engine (JRE). This file is of type .MSI so we will need to choose the task **'Windows Installer Package (Install)'**. This is located in the **Provisioning** section or you can start typing 'Windows' or 'Install' in the **Instant Search** field (Figure 6).

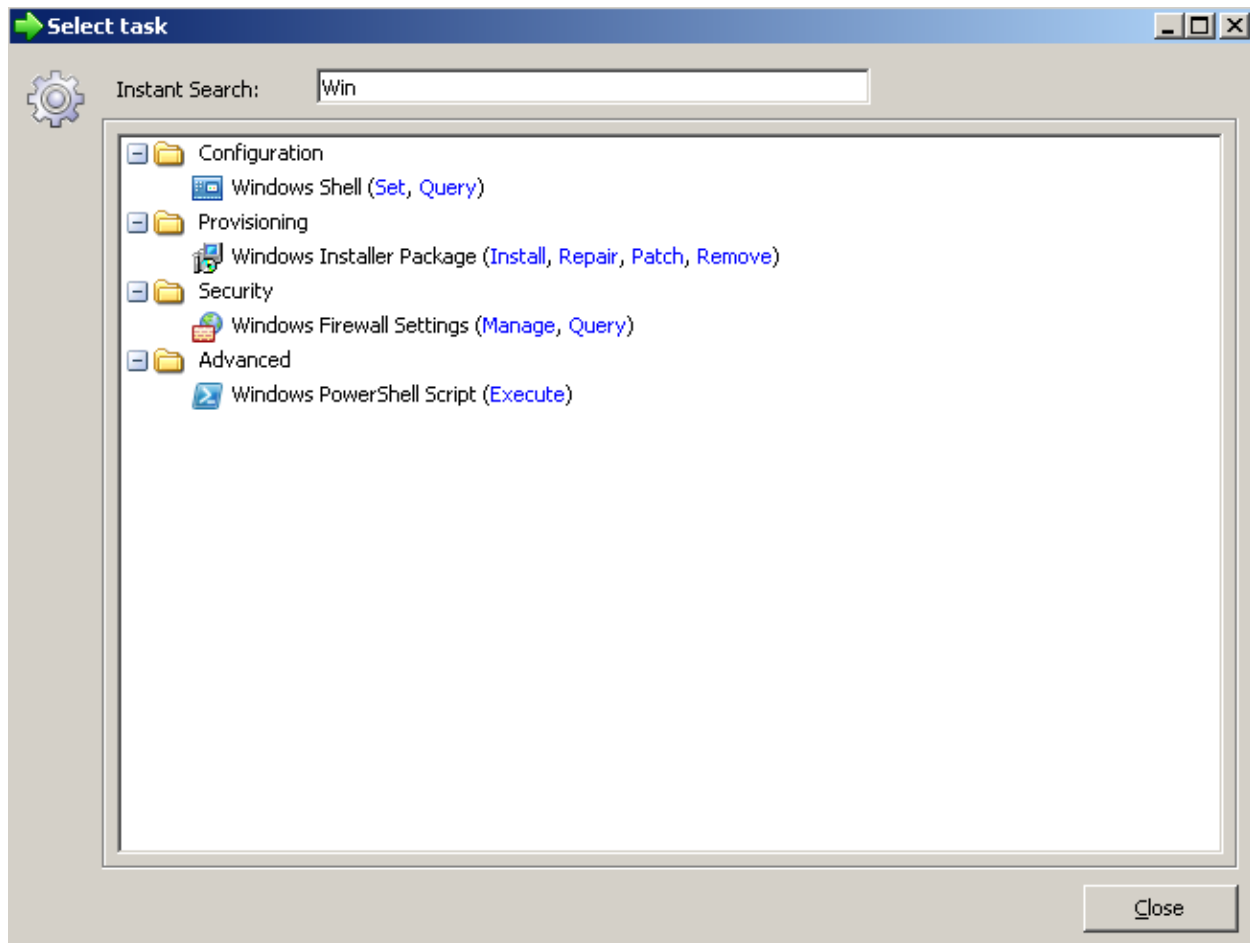
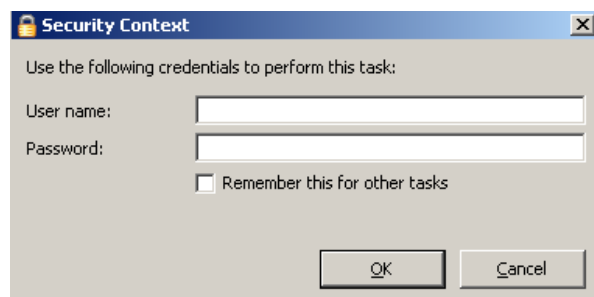


Figure 6: Search field

The Task **Windows Installer Package** can be used to install, repair, patch or remove .MSI packages on Agents. These Tasks automatically convert the options that you select into a command line, and execute this command line.

NOTE: All the available tasks are shown between the parentheses () as hyperlinks, simply hover over the task you want to use and click.

Fill in the **Security Context** field. The execution of a Task may require credentials, for example to access a file share or an Exchange mailbox. Such credentials go in the Task's Security Context field.



The user name should include the domain: **domain\user**. In Tasks that include a **Domain** field, this domain is filled out automatically as the default domain in the Security Context. This can be changed to another domain if necessary. It is recommended that you use **Parameters** for a Security Context. See the table below titled "Cool Feature – Parameters".

In the **Filename** field, browse to the Resource that contains the .MSI file for the JRE installer (Figure 7) and click **OK**.

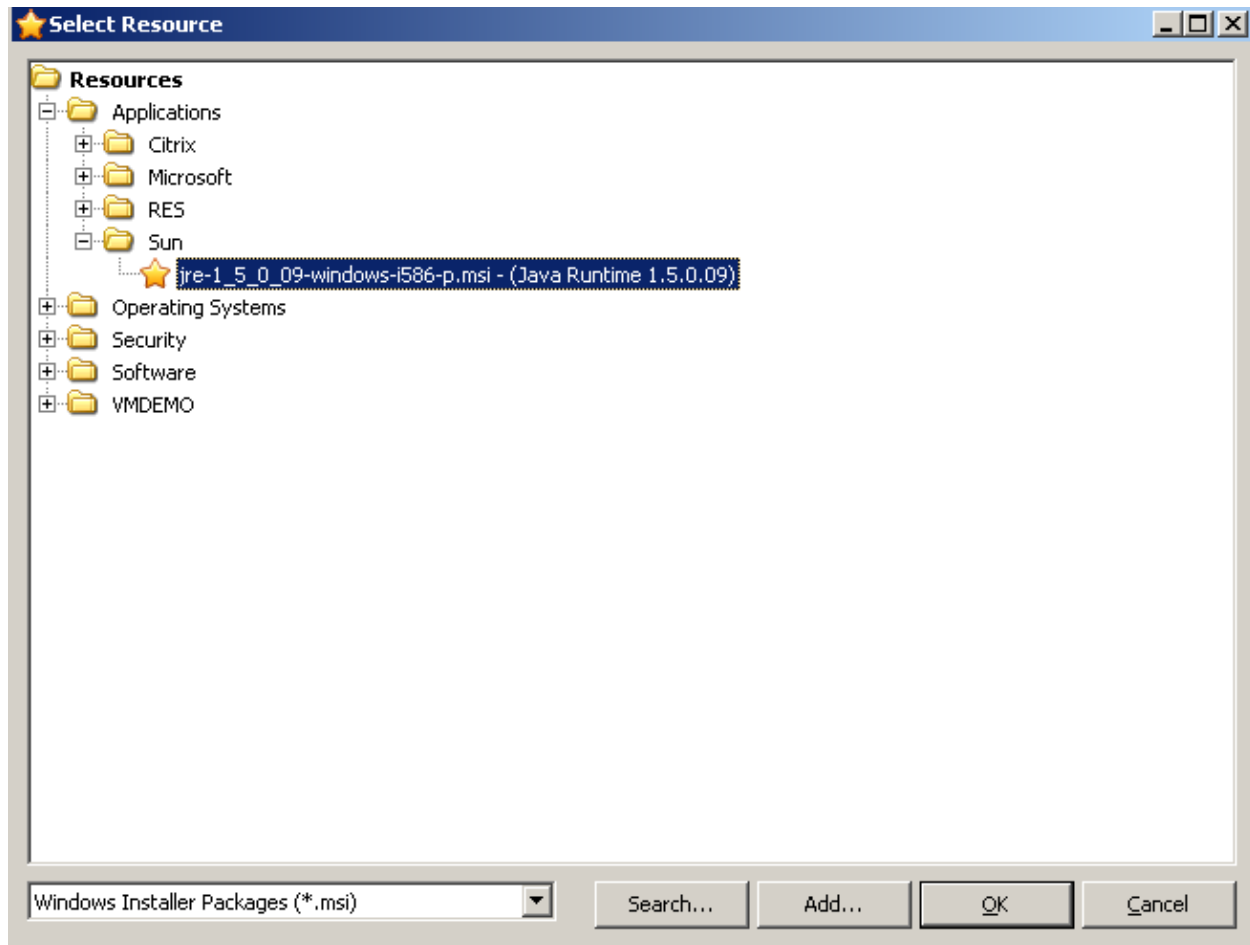


Figure 7: Resource Selection Dialog

Verify the file information is accurate. Next, Click on the **Condition** tab. A **Condition** determines whether a Task, Module or Run Book Job should be executed, skipped or failed. A condition contains one or more expressions that determine whether the condition can be satisfied and defines what action should be taken based on this. Even though Conditions are optional, it is a good idea to use them to prevent unwanted installations on the wrong agents.

For this module, create a condition to determine if the JRE has already been installed. There are many different ways to accomplish this. For this example, we will use the uninstall information located in the registry. Conditions can be based on the items listed in Figure 8. Use whichever method work best for the individual task.

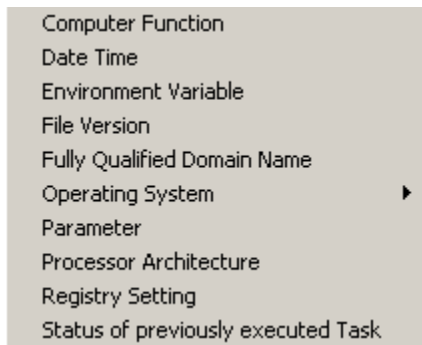


Figure 8: Rules for Conditions

The JRE **Registry Setting** information is located in **HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Uninstall\{3248FOA8-6813-11D6-A77B-00B0D0150090}\DisplayName** where the key value is equal to “**J2SE Runtime Environment 5.0 Update 9**”. From the pull-down menu, choose “**All the expressions need to be true for the condition to be true**”. If the condition is TRUE, then we set the function to “**Skip this task**”. If the condition is FALSE, then we set the function to “**Execute this task**”. Click **OK** to complete this task.

Next, Click **Add**, to create the task for installing the .NET 2.0 resource. This file is of type .EXE and we will be passing unattended install information so we will need to choose the task ‘**Unattended Installation (Perform)**’. This is located in the **Provisioning** section or you can start typing ‘Unattended’ or ‘Perform’ in the **Instant Search** field.

In the **Perform Unattended Installation** dialog box click on the browse button next to the **Filename** field and choose the **dotnetfx.exe** Resource (Figure 9).

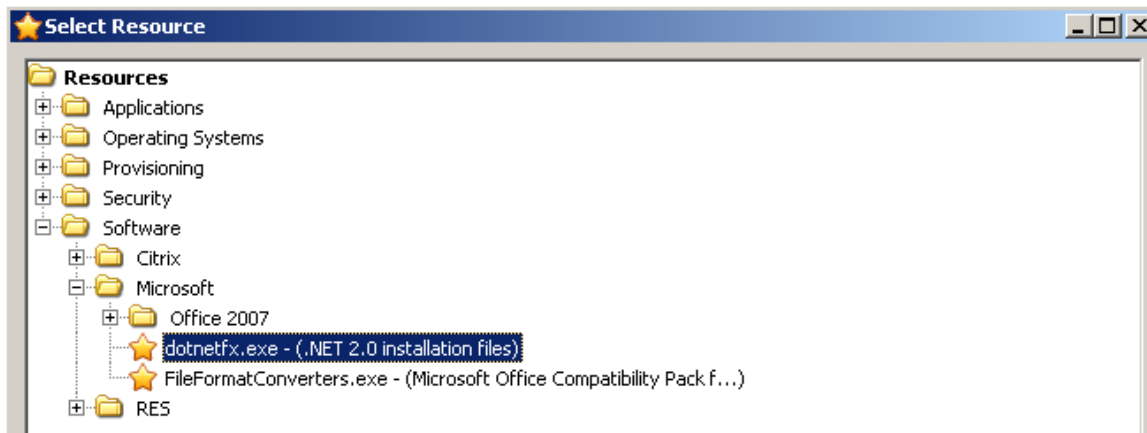


Figure 9:

Since this install is unattended, we need to pass some command line parameters to this .EXE file.

Parameters:	<input install.exe="" q"="" type="text" value="/q:a /c:"/>	
Security context (optional):	<input type="text" value="\$[Account]"/>	...
Optional grab log file:	<input type="text"/>	...

More command line arguments can be located on the Internet using your favorite search engine or known websites like appdeploy.com.

Enter your Security Context - Since within the context of this module a Parameter for Security Context has already been created, we can right-click and choose the pre-configured parameter (Figure 10), in this example - "Account".

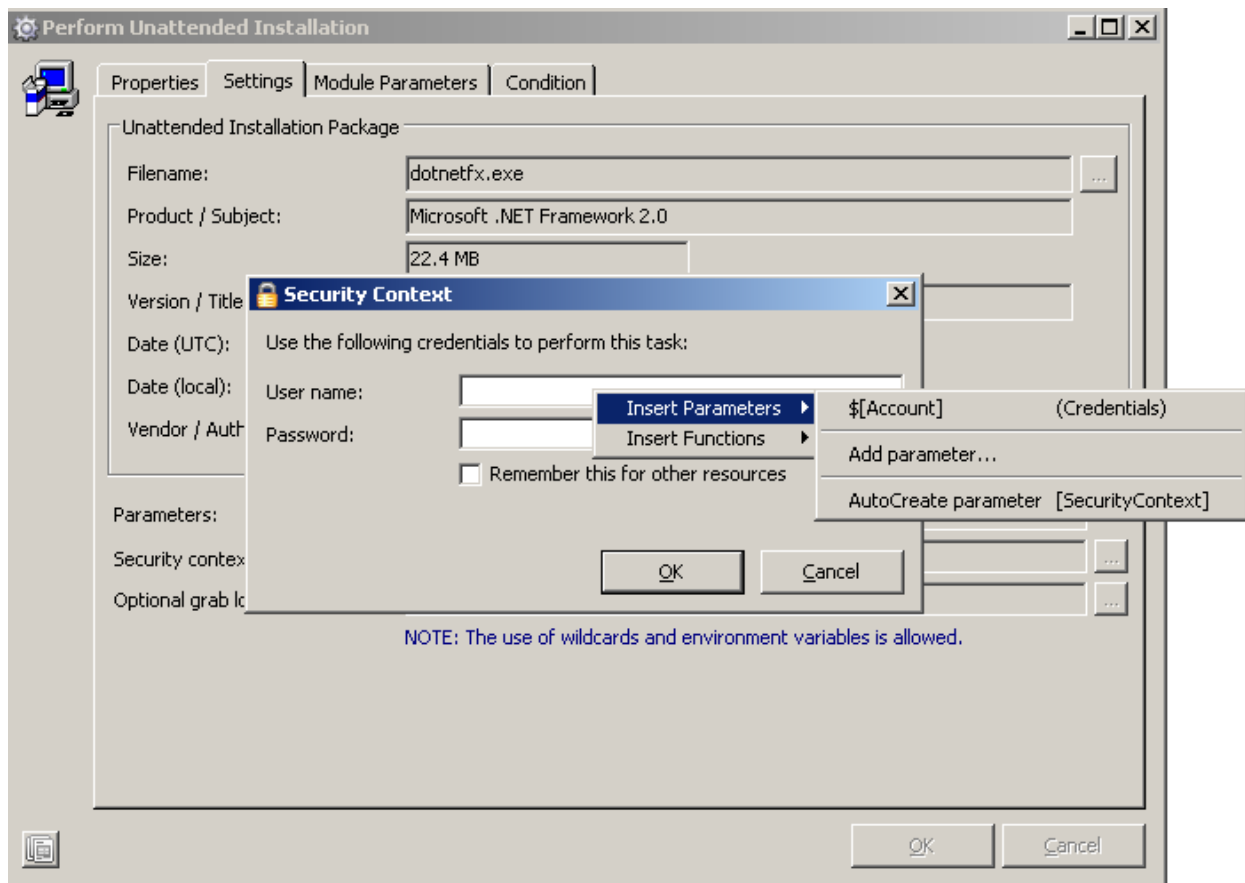


Figure 10: Security Context Selection

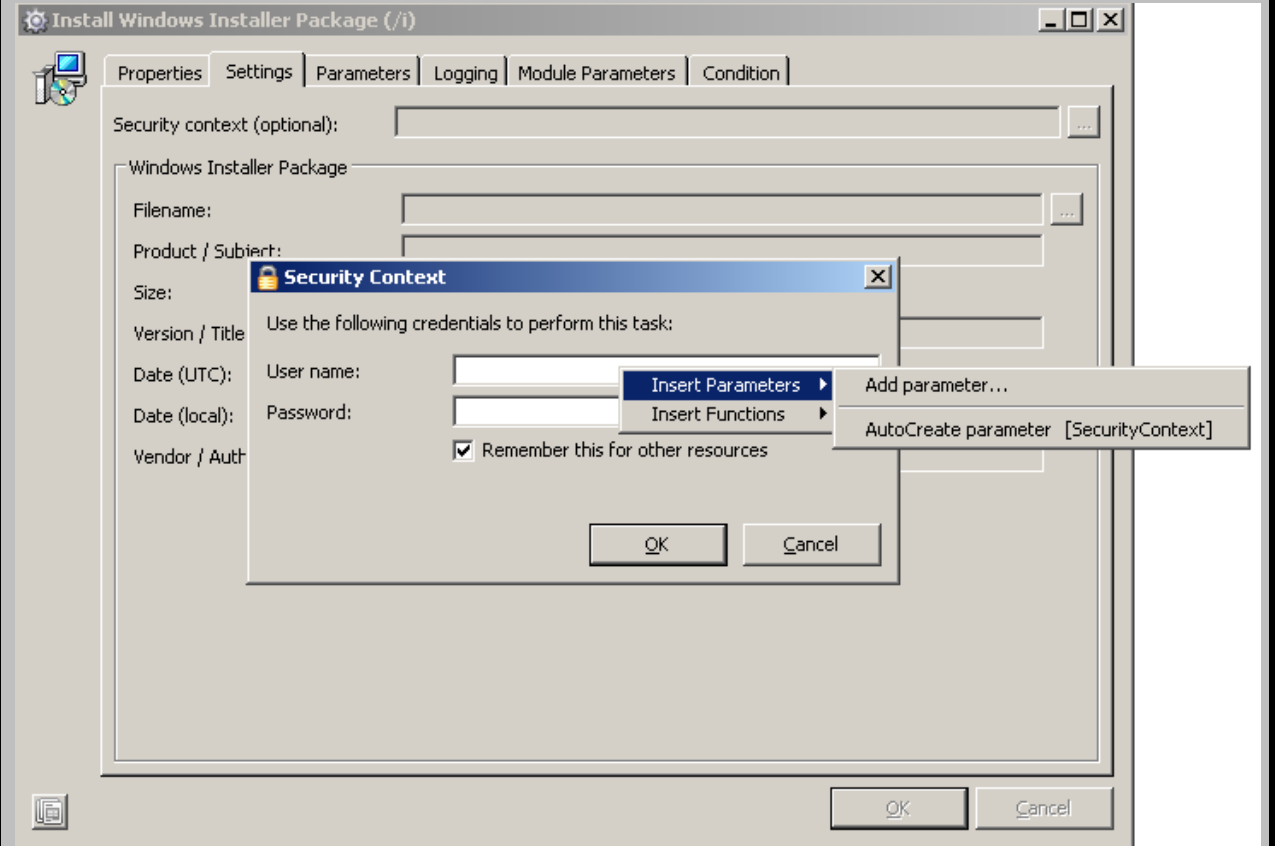
Next, click on the **Condition** tab and create a condition just like the one in the previous Task. This time the task will install .NET 2.0. The registry setting for this file is located in **HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Uninstall\{7131646D-CD3C-**

40F4-97B9-CD9E4E6262EF}\DisplayName with a key value equal to “Microsoft .NET Framework 2.0”. Click **OK** and complete this module.

COOL FEATURE – PARAMETERS

When defining the Security Context for Tasks, use **Parameters**. Visit <http://www.ressoftware.com/pm-downloads.aspx?PageID=107> for more information on Parameters. **Parameters** are placeholders in a Task for the values in various fields, such as text, a file path, credentials, etc. By using a Parameter for the Domain Credentials, you can input the Domain Administrator username and password into the parameter prior to running the job. This will allow the Job to be delegated to be executed another IT person or outside consultant without giving them the Domain Administrator credentials.

To add a parameter for the Security Context, click the browse box next to the field. In the **User Name** field, right-click and choose ‘**AutoCreate parameter**’. (Figure A)



The screenshot shows the 'Install Windows Installer Package (/i)' dialog box with the 'Parameters' tab selected. The 'Security context (optional):' field has a browse button. Below it, the 'Windows Installer Package' section contains fields for 'Filename:', 'Product / Subject:', 'Size:', 'Version / Title', 'Date (UTC):', 'Date (local):', and 'Vendor / Auth'. The 'User name:' field in the 'Security Context' dialog is right-clicked, opening a context menu with options: 'Add parameter...', 'AutoCreate parameter [SecurityContext]', 'Insert Parameters', and 'Insert Functions'. The 'Remember this for other resources' checkbox is checked. 'OK' and 'Cancel' buttons are visible at the bottom of the dialog.

Figure A: Autocreate Parameters

You can also choose your own parameter name, in this example I choose “Account”. Fill in the necessary fields and click **OK**. Notice the passwords are masked to prevent other users from obtaining the information. (Figure B)

The screenshot shows the 'Edit Module Parameter' dialog box with the following fields and values:

Field	Value
Parameter name:	Account
Description:	Account and password with domain administrator privileges. Use domain\username convention.
Hint:	Please provide the necessary input
Type:	Credentials
User name:	education\administrator
Password:	*****
Confirm password:	*****

Figure B: Edit Module Parameter Dialog Box

Module – XenApp Platinum 4.5

Now that the prerequisites are completed we can begin creating the set of tasks that installs XenApp Platinum 4.5. Open the RES Wisdom Management Console, select the **Modules** node in the left pane, and click **Add**. Give the module a name. In this example, the module is named “**Software, Citrix, Deploy XenApp 4.5 Platinum - Join, Direct**”.

We will first copy down to the target machine some of the Resources created earlier. This creates a consistent and controlled path for our install files that will be the same on any machine we install

XenApp on. On the **Tasks** tab, click **Add**. Choose the task '**Resource (Download)**'. This is located in the **Provisioning** section or you can start typing 'Resource' or 'Download' in the **Instant Search** field. Use the Task **Resource (Download)** to download Resources to Agents, for example to pre-cache large files or to send a specific version of a DLL file to Agents on which it is needed.

In the **Download Resources** dialog box, click **Add** and select the Resource Package that contains the set of files created earlier. In this example choose the resource **XenAppPlatinum_UnattendInstallFiles** (Figure 11). Click **OK**. At the bottom of the window, enable the checkbox next to **Specify destination folder** and provide a folder location on the target machine where the files should be copied. In this example it is C:\TempTSFiles (remember the path in the **Unattended_CPS45_RESWisdom.txt** file?)

Another thing to consider is the use of Parameters. Just like the Security Context, Parameters can be used for almost any field or input when executing tasks. Since we will be using this path a few times during the install, we will make the temp location into a Parameter. Right-click in the **Destination** field and select **Insert parameter/Add parameter**. Give the parameter a name, in this example it is named **WindowsTempLocation** (it is a general recommendation not to use spaces in parameter names). Add a description, a hint if you like, select the type as **Text**, and finally, add the value as **C:\TempTSFiles**. Now we can reuse this parameter anywhere in the module that asks for a path to the temp folder. Click **OK** to complete the task.

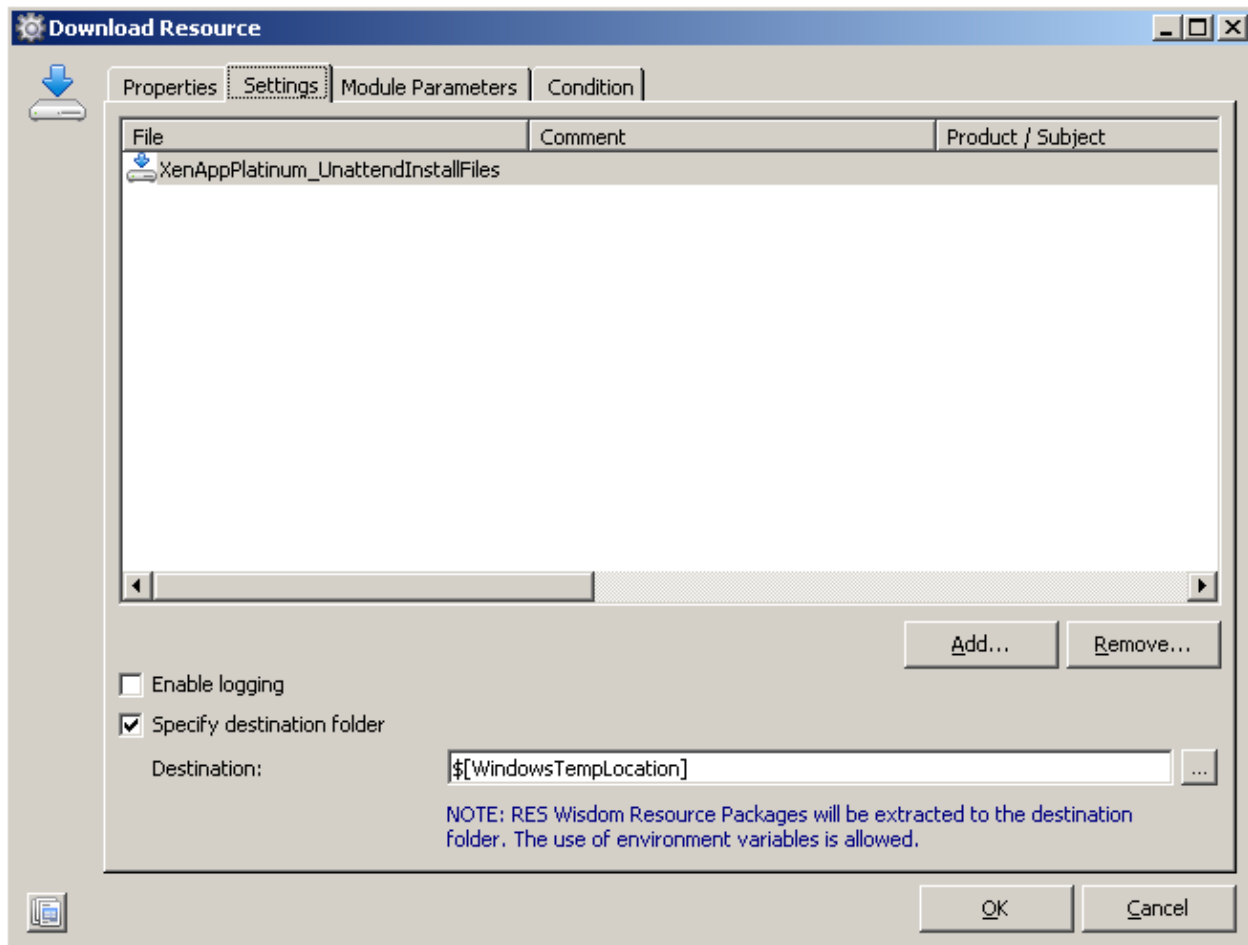


Figure 11: Unattended Installation files and Temp Path Parameter

By leveraging parameters in this way, if later on you decide to change the path for the temp location, you only need to change the value in the parameter instead of all the locations within the tasks.

Now that the files are copied down to the server, we can create the task that executes the command-line to launch the unattended installer for XenApp Platinum. On the **Tasks** tab, click **Add** and choose the task '**Command (Execute)**'. This is located in the **Advanced** section or you can start typing 'Command' or 'Execute' in the **Instant Search** field.

In the command-line field, enter the execution command. Since this file is of type .EXE, we can use the format: *<path to executable> <path to install files> <path to unattended.txt> <command-line arguments>* (make sure to type spaces between each item). This is the point where the command-line argument **CTX_ODBC_PASSWORD** will be used. So the command line will look something like...

```
"C:\TempTSFiles\UnattendedInstall.exe" "\\Server\Software Share\<path to msi>\mps.msi"
"C:\TempTSFiles\CPS45_RESParamsOnly.txt" CTX_ODBC_PASSWORD="Password"
```

There are many other command-line arguments that can be used to fine-tune the installation. Consult the XenApp Administrator's Guide for more information on this.

In this example, we can refine the command-line a bit further by introducing our Parameter created earlier for the temp path like this...

```
"${WindowsTempLocation}\UnattendedInstall.exe" "\\Server\Software Share\

```

Lastly, we can go even further and substitute a **Link** to the Resource from the files stored in the datastore. In the command-line field, simply right-click the place where the path to the install files would normally go and select **Insert Resource Link** (Figure 12) and browse to the **mps.msi** file Resource (Figure 13).

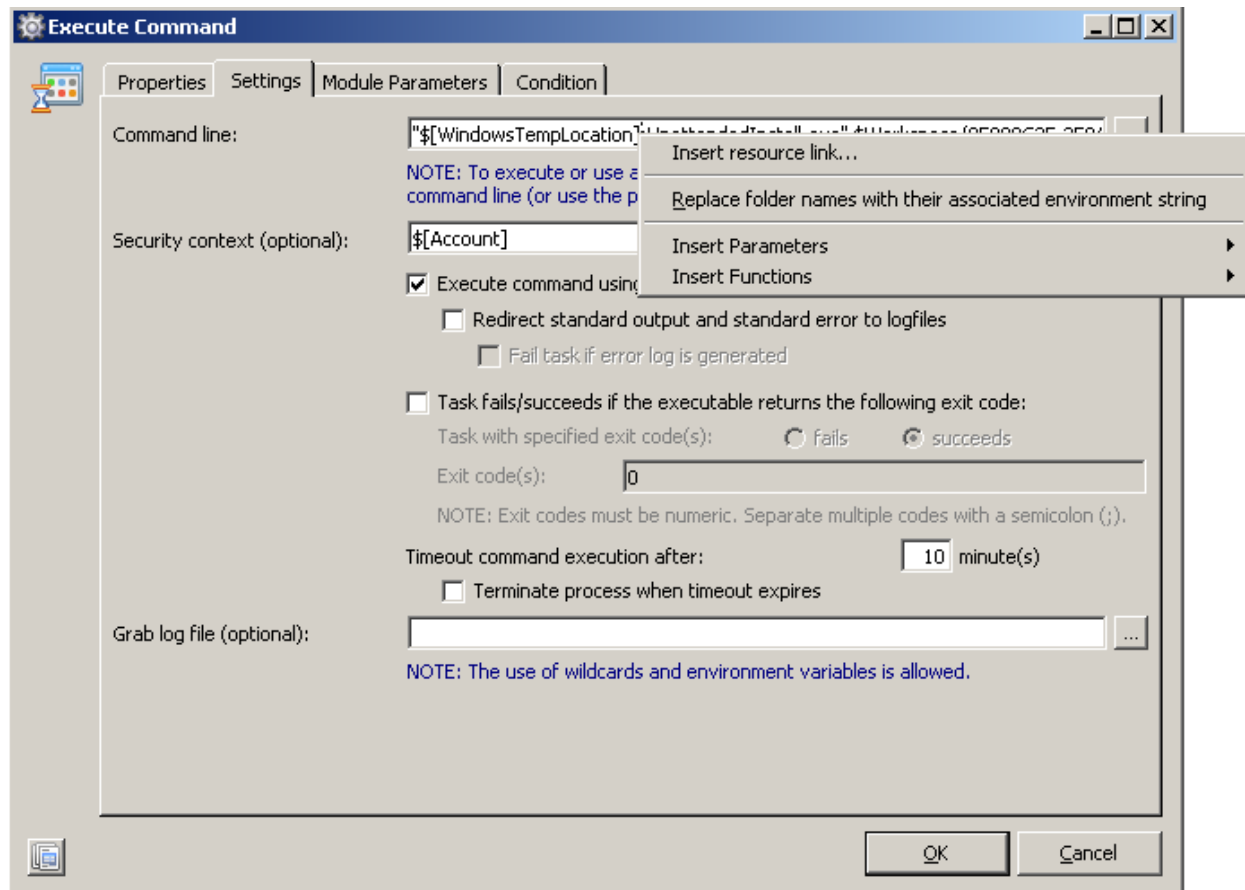


Figure 12: Using a Resource Link

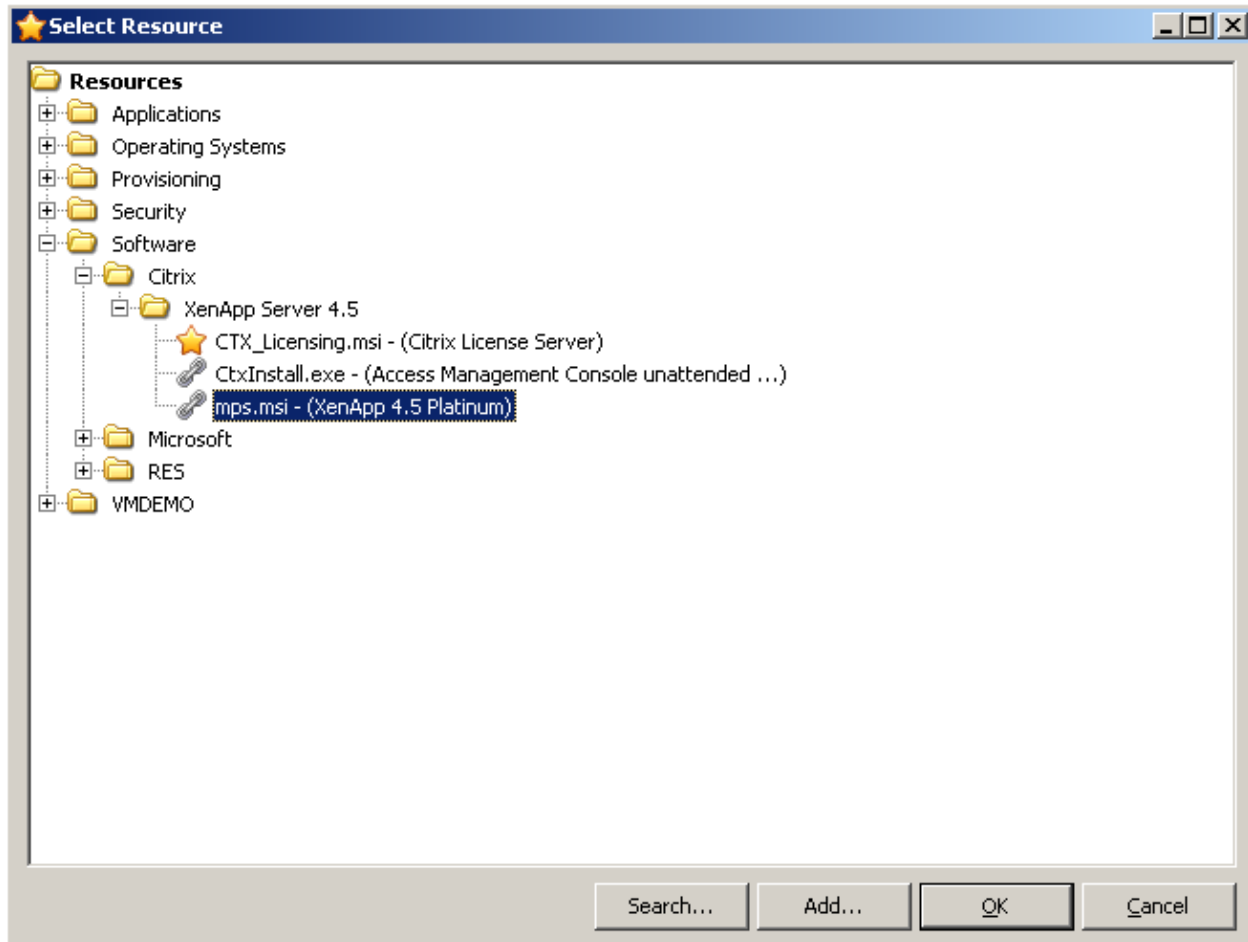


Figure 13: Resource Link Selection

Click **OK** to insert the Resource Link (also referred to as `$Workspace{GUID}`) and the command-line will now look like this...

```
"$ [WindowsTempLocation] \UnattendedInstall.exe" $Workspace{85988C3E-2E0A-4167-9A59-773C793A55AA}
"$ [WindowsTempLocation] \CPS45_RESParamsOnly.txt" CTX_ODBC_PASSWORD="Password1"
```

You can use the `$Workspace{GUID}` to reference any Resource in the datastore. This comes in handy if the path to the media changes, you will not have to update all your tasks, just the Resource. A unique `$Workspace{GUID}` is generated for each resource that is created in the Wisdom database.

Enter the Security Context (don't forget to use a parameter for this) and enable the checkbox for **Execute command using the Windows command interpreter**. The Task **Execute Command** can be an advanced Task. If you enter a complex command line, RES Wisdom cannot interpret its validity. Therefore, always consider selecting **Execute command using the Windows command interpreter** and **Redirect standard output and standard error to log files**

Now that the XenApp install is completed, a reboot is required – just like you were installing it manually. Guess what? There happens to be a Task for just such an event. On the **Tasks** tab, click **Add** and choose the task **'Computer (Reboot)'**. This is located in the **System State** section or you can start typing 'Computer' or 'Reboot' in the **Instant Search** field. Create a reboot task that looks like Figure 14 and click **OK**.

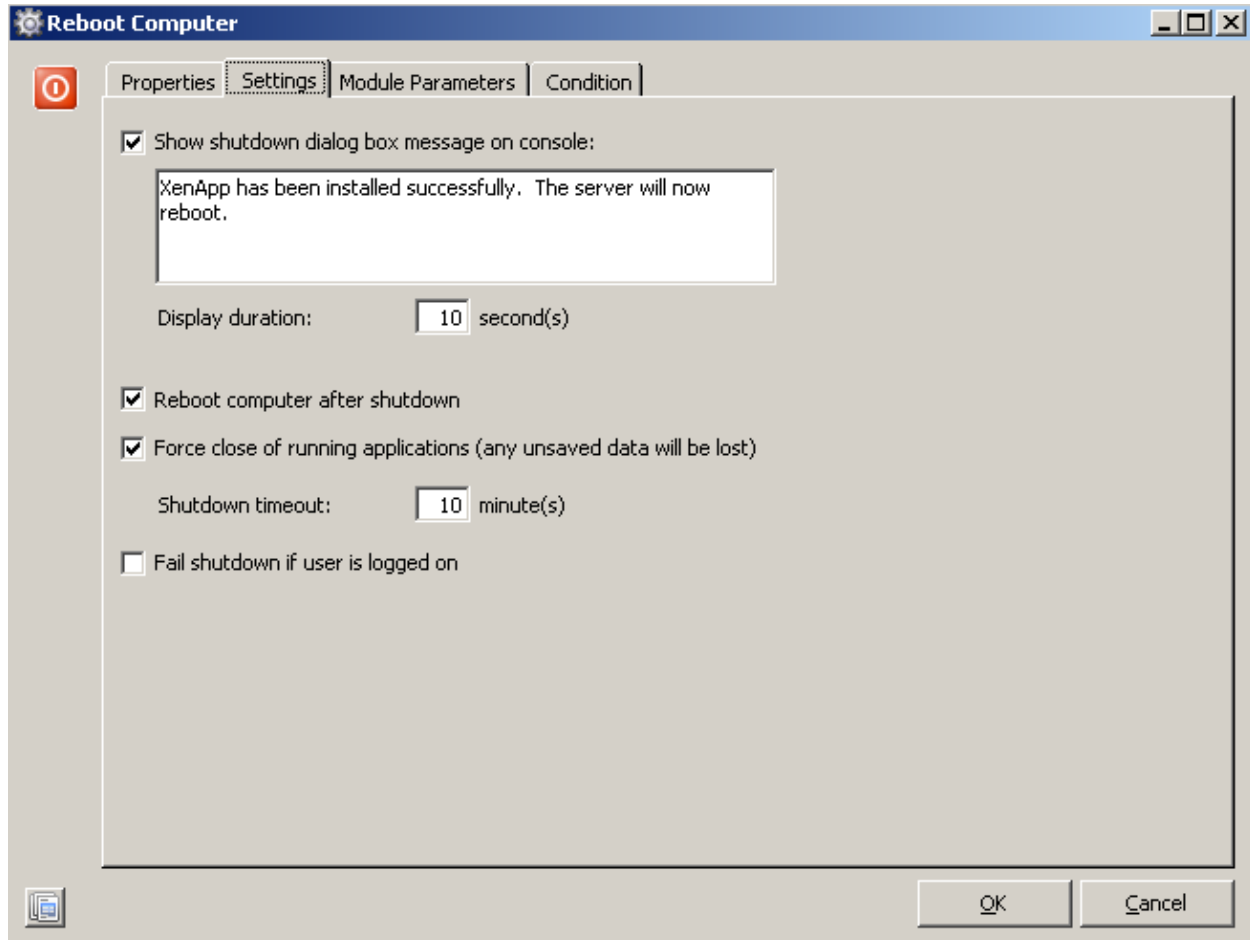


Figure 14: Reboot Task

TIP: The console message is optional. However, when scheduling a large module that has many reboots, it will keep you informed as to the stage you are at during the execution of the Module or Project.

Once the reboot completes, the next step is to set the XenApp Edition to Platinum.

REMEMBER: Even though there is an option in the UnattendedTemplate.txt file to set the server edition to “Platinum”, experience has shown that it does not recognize this line item during the install and it sets the values to “Enterprise”.

To set the edition, we need to execute the **CScript** file that was initially copied to the temp folder. On the **Tasks** tab, click **Add** and choose the task ‘**Command (Execute)**’. This is located in the **Advanced** section or you can start typing ‘Command’ or ‘Execute’ in the **Instant Search** field. In the command-line field type:

```
| cscript $[WindowsTempLocation]\SetPSEdition.wsf %ComputerName% PLT
```

Add your Security Context, enable the checkbox for **Execute command using the Windows command interpreter**, and click **OK** to complete the task.

Once that task is complete we can begin installing the **Access Management Console**. On the **Tasks** tab, click **Add** and choose the task ‘**Command (Execute)**’. This is located in the **Advanced** section or you can start typing ‘Command’ or ‘Execute’ in the **Instant Search** field. The command-line will be a Resource Link to the **CtxInstall.msi** with a **/SILENT** switch at the end.

```
| $Workspace{7177819D-F9C4-4ED6-A930-40874BE8FB08} /SILENT
```

Add your Security Context, enable the checkbox for **Execute command using the Windows command interpreter**, and click **OK** to complete the task.

The next step will be a cleanup Task. We do not want to leave the files on the server so create a task to **perform** an action on these **files**. On the **Tasks** tab, click **Add** and choose the task ‘**Files (Perform Operations)**’. This is located in the **Configuration** section or you can start typing ‘Files’ or ‘Perform’ in the **Instant Search** field. Click **Add** in the Perform File Operations dialog box, and for the action type choose **Delete – File/Folder**. Add your security context parameter and in the source path add the parameter for the temp file location path (Figure 15). Are you starting to love parameters?

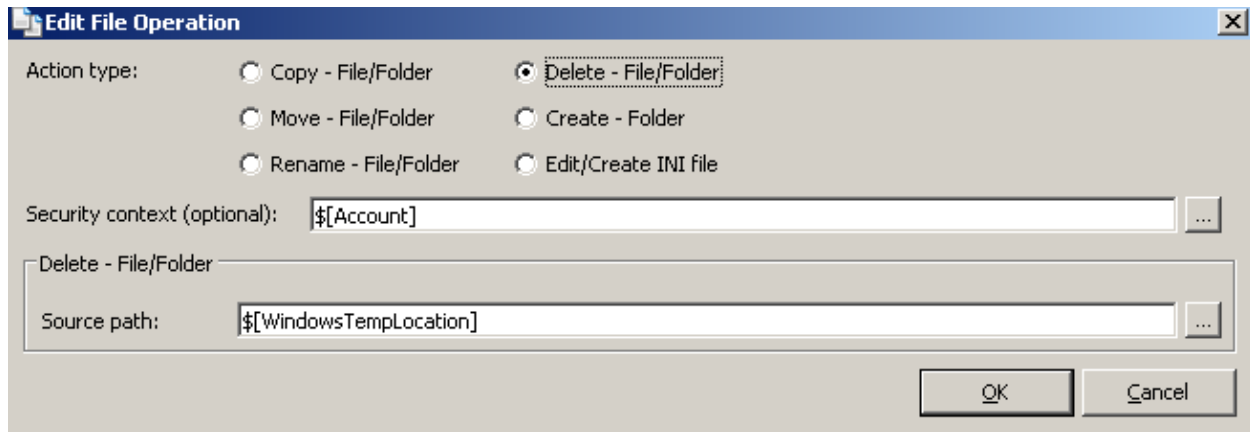


Figure 15: Delete Files Using Parameters

At last, the final step!!! Create another reboot task to finalize the XenApp installation. Your Module will look like Figure 16.

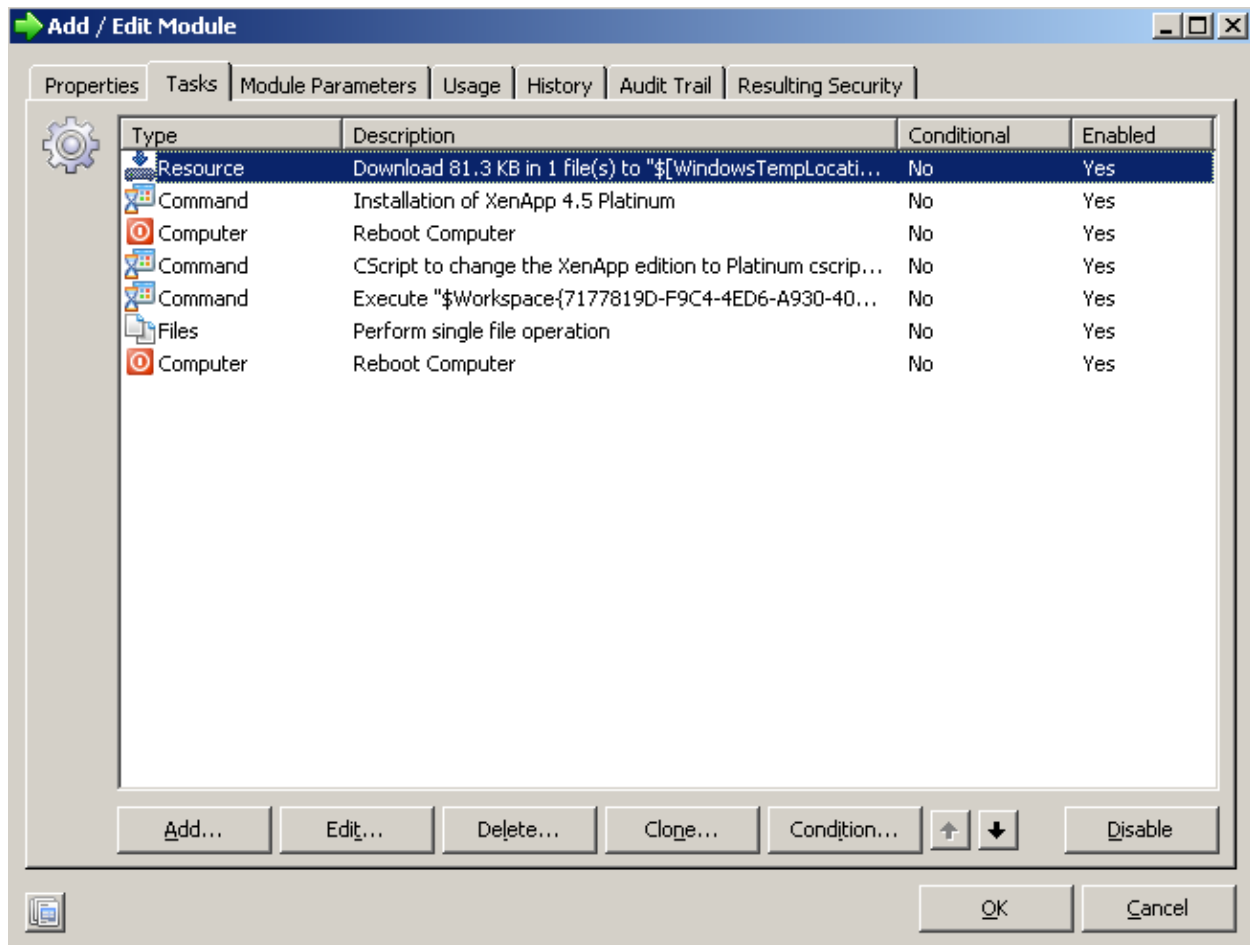


Figure 16: Completed XenApp installation Module

CONGRATULATIONS!

(2)RES Wisdom Projects

If you find yourself repeatedly scheduling the same sequence of Modules, combine them into a **Project**. By combining several Modules into one Project, those Modules can be scheduled as one Job instead of separately as a series of Jobs.

To illustrate, add the two Modules created in this document to a Project. Open the RES Wisdom Management Console, select the **Projects** node in the left pane, and click **Add**. Give the Project a name (keep the naming conventions that have been discussed in previous sections). In this example the Project name is **Domain, Servers, Provision a Citrix XenApp Platinum 4.5 edition server**. On the Modules tab, click **Add** to select the Modules created previously (Figure 17) and click **OK**.

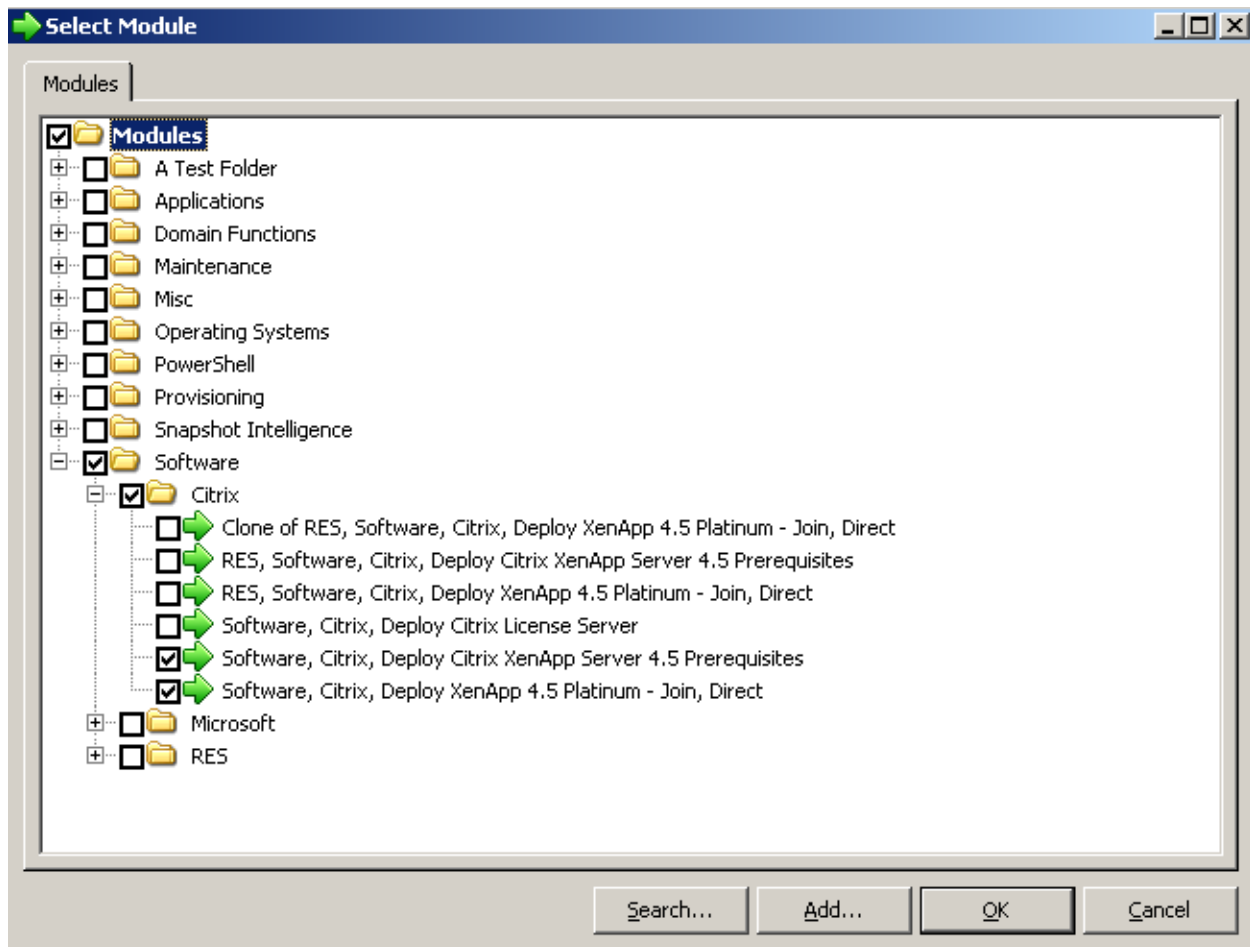


Figure 17: Add Module Selection Screen

You should now see both Modules listed. Select the **Current Resulting Tasks** tab to view all the tasks associated with these Modules (Figure 18)

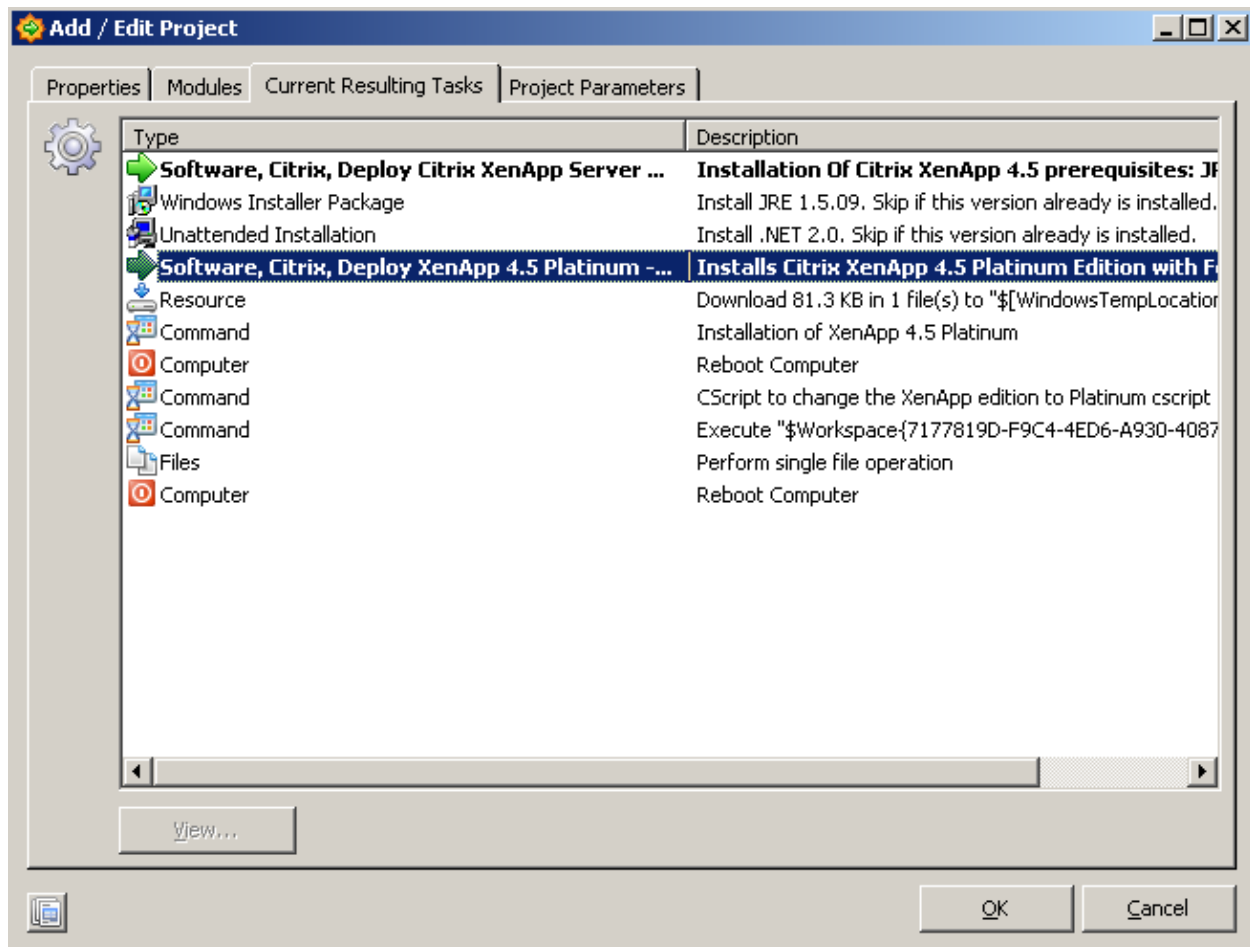


Figure 18: Current Resulting Tasks

Click **OK** to complete the Project. The only thing left to do now is schedule the Project (or Module) and execute it. To schedule a Project or Module, open the RES Wisdom Management Console, select the **Projects** node (or **Modules** node) in the left pane, right-click the Project or Module you wish to run and select **Schedule Job** from the context menu. The **Add/Edit Job** dialog appears (Figure 19).

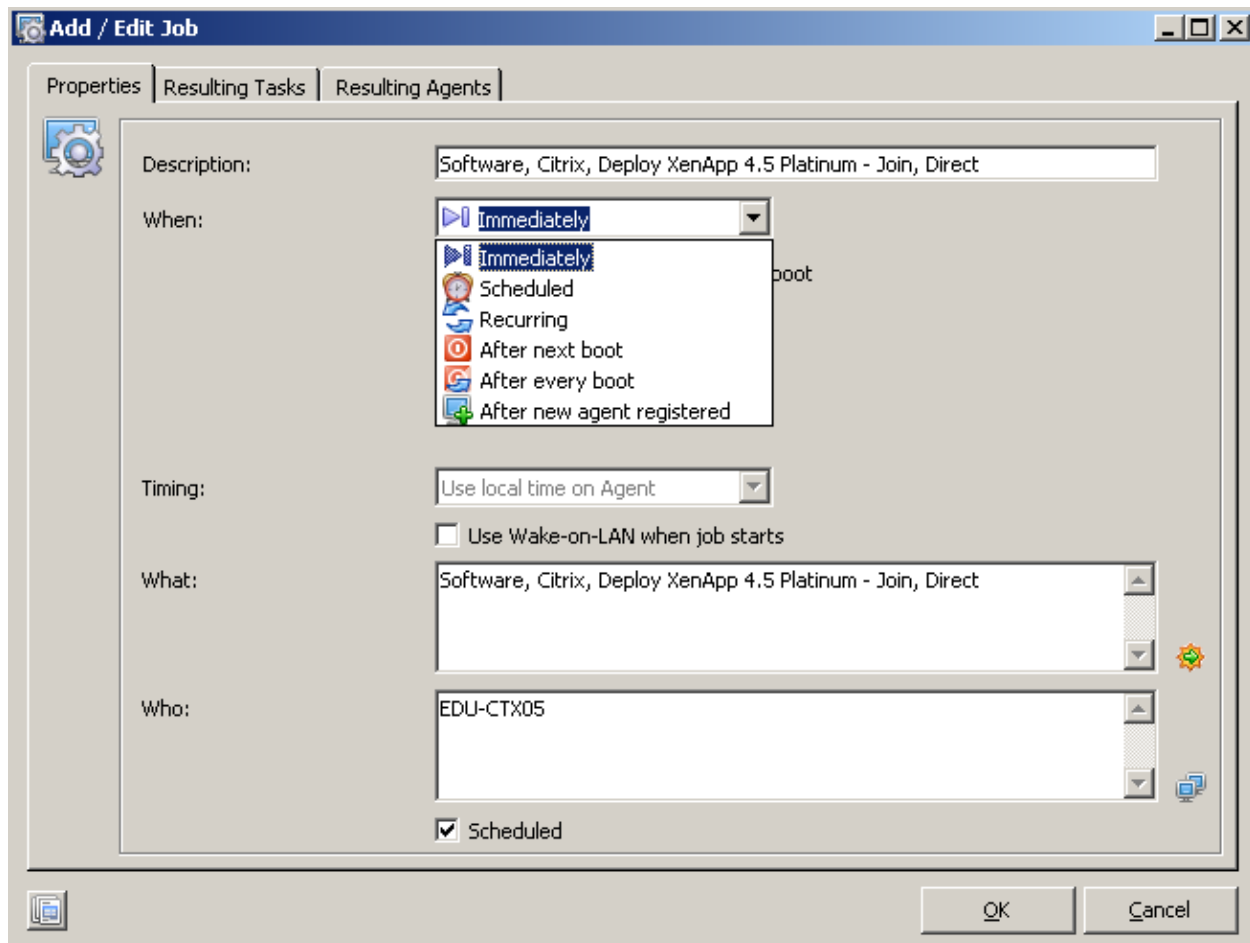


Figure 19: Job Scheduling

Here you can specify the Agent that the Job should run on (click in the Who field and select the agent(s) the job will execute on) and choose when it should run (from the **When** pull-down menu). For this example, we would choose **Immediately**. Click **OK**. The next time the Dispatchers contact the Datastore it check to see if new Jobs are available for Agents to which it connects. The Dispatcher will download all necessary data from the Datastore and store it. The Agents will use this data to execute the Jobs.

(1)Summary

This is just one way to accomplish this task. There are many ways to tackle any given task. Feel free to add and tweak each step as it fits your situation. You can add additional conditions as needed to really customize the Job.

As a final note, these are just two of the Modules from a larger **Project** that RES Wisdom uses to create a Citrix XenApp server.

Those of you with RES Wisdom experience can add these two Modules to an already existing project that builds a server. Here is a screen shot of the Project that I use to Provision my Citrix servers (Figure 20).

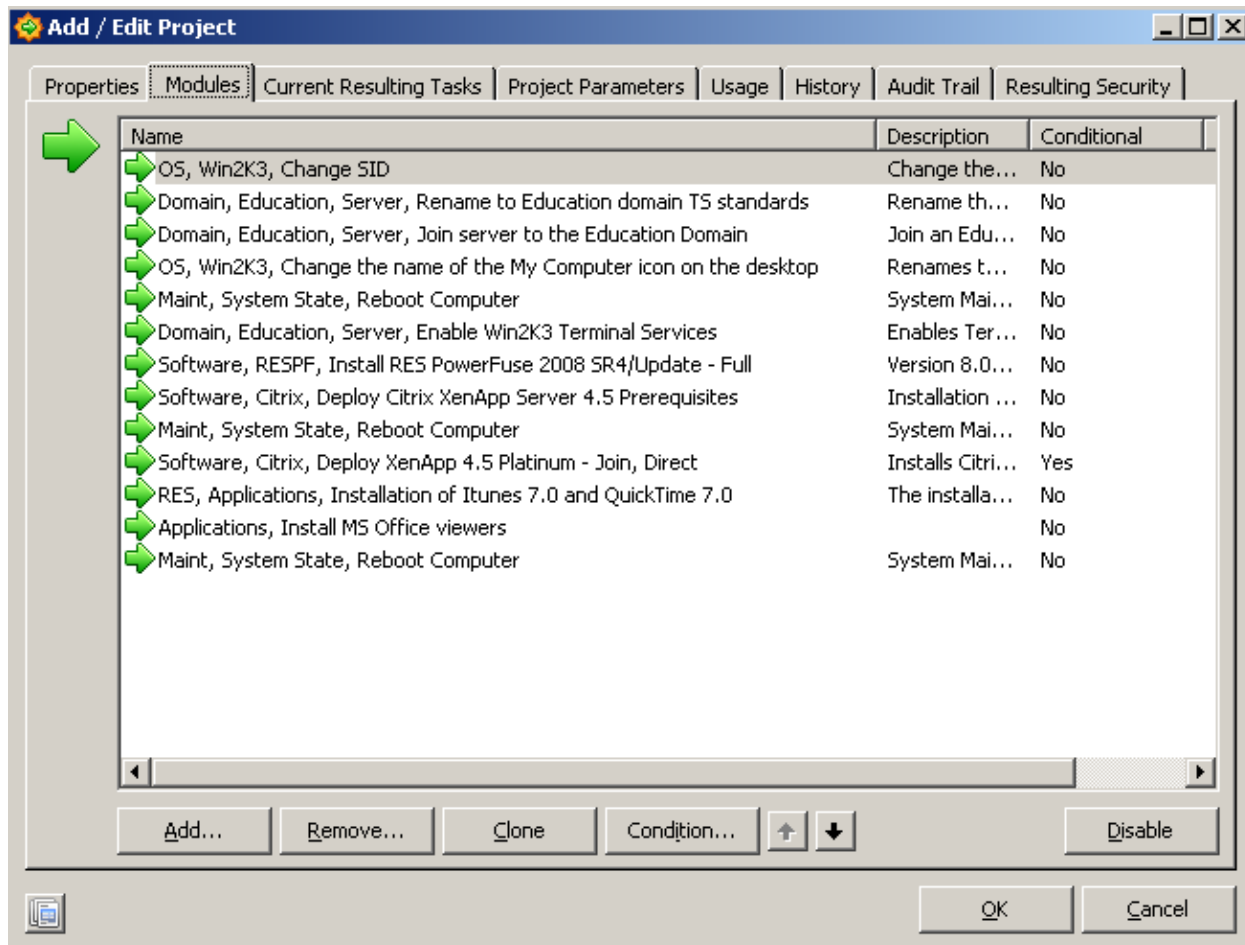


Figure 20: Project to Create a XenApp server

As you can see you can be as simple or as detailed as you like, the choice is yours. One of the benefits of RES Wisdom is that it is not an “all-or-nothing” product. You can simply choose the Tasks that are applicable to you at your level of integration and build from there.

Happy building!

Brian Casselman

Systems Engineer – RES Software

RCP, MCSE, CCIA

<http://www.ressoftware.com>

Appendix A - SetPSEdition Code Example

```
<package>
  <job id="plt">
    <reference object="MetaFrameCOM.MetaFrameFarm"/>
    <script language="VBScript">

      Dim s, w
      if WScript.Arguments.Count < 2 Then
        WScript.Echo "Script requires two arguments <servername> <edition>"
        WScript.Quit
      End if

      On Error Resume Next

      Set s = CreateObject("MetaFrameCOM.MetaFrameServer")
      s.Initialize MetaFrameWinSrvObject, WScript.Arguments(0)

      Set w = s.WinServerObject6
      w.MPSEdition = WScript.Arguments(1)

      if Err.Number <> 0 Then
        WScript.Echo "Error setting " & WScript.Arguments(0) & "'s edition to " & WScript.Arguments
(1)
      Else
        WScript.Echo WScript.Arguments(0) & "'s edition successfully set to " & WScript.Arguments(1
)
      End if

    </script>
  </job>
</package>
```

Appendix B – Citrix Unattended.txt file Example

```

*****
* This is a sample answer file for unattended installations of
* Citrix Presentation Server 4.5 for Windows.
*
* To run an unattended installation of Presentation
* Server, make a copy of this file and customize it for
* your needs.
*
* If you do not use an answer file or if you use an answer
* file but do not specify answers to some questions, default
* answers are used for those questions. The default answers
* in the installation are identical to the default answers
* listed in this sample answer file.
*
* Important: For security reasons, account passwords are not stored
* in this file and must be provided via the command-line.
* For information about the parameters used for providing
* credential information, see the "Setup Property Names and
* Values" section in the "Advanced Installation Methods"
* appendix of the Citrix Presentation Server Administrator's Guide.
*
* When upgrading, Setup ignores entries other than those from
* the following sections:
*     Citrix License Agreement
*     Options
*     LicenseServer
*     MFRDP
*
* For additional information about the Citrix Streaming Client, see
* the Citrix Application Streaming Guide.
*
* For additional information about installing Presentation Server, see the
* Citrix Presentation Server Administrator's Guide and the System
* Requirements section of the Installation Checklist. For
* additional information about licensing, see the Getting Started
* with Citrix Licensing Guide.
*****
*****
* Citrix License Agreement
*
* This section specifies your acceptance of the Citrix End-User
* License Agreement. You must set this value to "Yes" to
* indicate your acceptance of the Citrix End-User License
* Agreement.
*
* The unattended installation will fail if you use any value other than
* "Yes".
*****
[Citrix License Agreement]
AcceptLicense=No
*****
* Data Store Configuration
*
* This section specifies whether you are creating or joining a
* farm and how to connect to the data store.

```

```
*
* Multiple options are available as follows:
*
* -> To create a farm using an Access database:
*   > Set CreateFarm to Yes
*   - Set LocalDBType to Access
*   - Set DirectConnect to No
*   - Complete the Farm Settings section
*
* -> To create a farm using a SQL Server Express database:
*   - Set CreateFarm to Yes
*   - Set LocalDBType to SQL
*   - Set DirectConnect to No
*   - Complete the Farm Settings section
*
* -> To create a farm using an Oracle, Microsoft SQL Server, or
*   IBM DB2 database:
*   - Set CreateFarm to Yes
*   - Set DirectConnect to Yes
*   - Complete the Direct Connect Settings section
*   - Complete the Farm Settings section
*
* -> To join a farm using an Oracle, SQL Server, or DB2
*   database:
*   - Set CreateFarm to No
*   - Set DirectConnect to Yes
*   - Complete the Direct Connect Settings section
*
* -> To join a farm by connecting to an Access or SQL Server Express
*   database stored on another server (Indirect Connection):
*   - Set CreateFarm to No
*   - Set DirectConnect to No
*   - Complete the Indirect Connect Settings section
*
* Specify the name of the Zone where this server will reside.
* If no Zone name is specified, a default Zone name is applied
* to the first server in a farm. If no Zone name is specified
* for servers joining a farm, the servers are added to the Zone
* where the farm resides.
*
* If you are using an Oracle, SQL Server, or DB2 database,
* manually create a .DSN file within the ODBC Data Source
* Administrator "File DSN" option before running the unattended
* installation process.
*
* IMPORTANT: Read the Citrix Presentation Server Administrator's
* Guide for information about data store configuration, supported
* databases, setting up a DSN file, moving servers within
* farms, and renaming Zones.
*****
[Data Store Configuration]
CreateFarm=Yes
LocalDBType=Access
DirectConnect=No
; Leave this blank to use the default zone name
ZoneName=
*****
* Direct Connect Settings
*
```

```

* This section specifies settings for an Oracle, SQL Server, or
* DB2 database.
*
* This section is used only if the value for DirectConnect is
* Yes in the Data Store Configuration section.
*
* In this section you must specify:
*
* 1. The path to the DSN file for this database.
*   - If you are creating a farm, see the Citrix Presentation
*     Server Administrator's Guide for details on how to
*     create a DSN file to use here.
*   - If you are joining a farm, you must specify the path
*     to the DSN file created on the server where you
*     created the farm or a copy of it. For details, see the
*     Citrix Presentation Server Administrator's Guide.
*
* Note: Specifying passwords in the unattended answer file is
* not supported for security reasons. You must use the
* CTX_ODBC_PASSWORD command-line parameter to specify
* the password that can authenticate to this database.
*
* If Windows NT authentication is being used, please include
* the domain in the user name in the form domain\username.
* This command is described in the Citrix Presentation Server
* Administrator's Guide.
* Note: Fully Qualified Domain Names are not supported.
*****
[Direct Connect Settings]
DSNFilePath=
UserName=

* To specify the password, use the CTX_ODBC_PASSWORD command-line
* parameter.

* Note: The section below was the section for MSDE in Citrix Presentation Server 4.0.
* It now uses different parameters.
*****
* SQL Server Express Settings
*
* This section specifies which SQL Server Express settings to use when
* connecting to the database.
*
* Use this section if you set CreateFarm to Yes and
* LocalDBType to SQL
*
* In this section you must specify:
*
* 1. The name of the Sequel Server Express instance to use when
*    connecting to the data store. The default is CITRIX_METAFRAME.
*****
[SQL Server Express Settings]
InstanceName=CITRIX_METAFRAME
*****
* Indirect Connect Settings
*
* This section specifies settings for connecting indirectly to
* an Access database on another server.

```

```

*
* This section is used only if CreateFarm is No AND
* DirectConnect is No.
*
* In this section you must specify:
*
* 1. The name of the server you want to indirectly connect to
*    and the port number to use.
*
* 2. The username and domain of the farm administrator account
*    that can access the first server in the farm (or a server
*    already running Presentation Server).
*
* Note: Specifying passwords in the unattended answer file is
* not supported for security reasons. You must use the
* CTX_INDIRECT_JOIN_PASSWORD command-line parameter to specify
* the password for the account for indirect join below.
* This parameter is described in the Citrix Presentation Server
* Administrator's Guide.
*
* Note: When using the indirect connect settings, you must specify
* the domain name and the user name since both fields are required.
* Fully Qualified Domain Names are not supported.
*****
[Indirect Connect Settings]
IndirectServerName=
IndirectServerPort=2512
UserName=
DomainName=
*To specify the password, use the CTX_INDIRECT_JOIN_PASSWORD command-line
*parameter.
*****
* Farm Settings
*
* This section specifies the settings for creating a farm.
*
* This section is used only if CreateFarm is Yes in the Data
* Store Configuration section.
*
* In this section you must specify:
*
* 1. The name of the farm you are creating.
*
* 2. A Windows NT user (user name and domain) who will be the
*    administrator of this farm. This user can later designate
*    other users as administrators of the farm using the
*    Presentation Server Console.
*****
[Farm Settings]
FarmName=Farm
FarmAdministratorUsername=Administrator
FarmAdministratorDomain=
*****
* Shadowing Restrictions
*
* This section specifies whether or not shadowing is enabled. If
* shadowing is enabled, this section specifies shadowing
* restrictions.
*****

```

```

[Shadowing Restrictions]
AllowShadowing=Yes
ProhibitRemoteControl=No
ProhibitNotificationOff=No
ProhibitLoggingOff=No
*****
* Citrix XML Service
*
* This section allows you to specify how you want to add XML
* support.
*
* The default port is 80 for the XML service. You can share the
* default port 80 with IIS (if installed) or you can dedicate a
* port for the XML Service.
*
* If ExtendIIS is set to No, the specified port is used for the
* Citrix XML Service, and EnableVirtualScripts is ignored.
* If ExtendIIS is set to No and IIS is installed (or any
* service already using port 80), Citrix XML Service takes the
* secondary default port as 8080.
*
* If ExtendIIS is set to Yes, DedicatedPortNumber is ignored
* and EnableVirtualScripts is used to decide if virtual scripts
* directory will be created ONLY if VirtualScriptsDirectory
* does not yet exist on the system. If it exists,
* EnableVirtualScripts has no effect on the installation.
*
* If you do set ExtendIIS to Yes and IIS is not available, the
* dedicated port number is used.
*
* If you do set ExtendIIS to Yes and IIS exists and
* VirtualScriptsDirectory does not yet exist on the system but
* EnableVirtualScripts is set to No, installation fails.
*
* If you do not use the default port 80, all Citrix ICA Clients
* using TCP/IP with HTTP server location and NFuse-enabled Web
* servers must be configured to make requests to the specified
* port number.
*
* If ExtendIIS is set to Yes and you are installing on Windows
* Server 2003 x64 Edition, the unattended (command line) setting
* will be disregarded and ExtendIIS set to No.
*****
[Citrix XML Service]
ExtendIIS=No
; This setting applies only if ExtendIIS is No
DedicatedPortNumber=80
; This setting applies only if ExtendIIS is Yes
EnableVirtualScripts=Yes
*****
* Options
*
* This section contains additional options for unattended
* installation.
*
* - RebootOnFinish -
*
* RebootOnFinish specifies whether Setup shuts down and restarts
* the server after the installation completes.

```

```
*
* If you set RebootOnFinish to No, manually restart the server
* before using Citrix Presentation Server.
*
* - LogLevel -
*
* LogLevel specifies the granularity of Windows Installer logging in
* the log file.
*
* To disable Windows Installer logging, set LogLevel to No.
*
* To configure Windows Installer logging, set LogLevel to one of the
* following values:
*
*      *      everything (the same as using all options listed
*              below for voicewarmup)
*      v      verbose
*      o      out of disk space
*      i      informational messages
*      c      command line parameters
*      e      error messages
*      w      warning messages
*      a      action execution messages
*      r      informational messages for the currently
*              running action
*      m      out of memory errors
*      u      user request messages
*      p      property values
*      +      append to an existing log file
*      !      flush each line to the log
*      x      extra debugging information - for Windows 2003
*              only;
*              using the x option when installing on Windows
*              2000 Server generates a usage error.
*
* Logfile specifies the path to the log file for the
* installation.
*
* - UILevel -
*
* UILevel specifies how much appears on the screen during
* the silent installation. Default is BASIC_UI_NO_MODAL.
* You can set UILevel according to one of the following:
*
* NO_UI
* BASIC_UI
* BASIC_UI_NO_MODAL
* BASIC_UI_MODAL
* REDUCED_UI
*
* NO_UI is equivalent to the msiexec q option. Setup
* displays no UI.
*
* BASIC_UI is equivalent to the msiexec qb option. Setup
* displays basic UI, but hides Cancel buttons.
*
* BASIC_UI_NO_MODAL is equivalent to the msiexec qb-
* option. Setup displays no modal dialog boxes.
*
```

```
* BASIC_UI_MODAL is equivalent to the msiexec qb+ option.
* Setup displays no UI except for a modal dialog box
* at the end of the installation.
*
* REDUCED_UI is equivalent to the msiexec qr option.
* Setup displays a reduced UI without a modal dialog box
* at the end of the installation.
*
* - IgnoreMCM -
*
* IgnoreMCM specifies whether Setup shows an error
* message if it detects an installation of Citrix
* Conferencing Manager prior to version 3.0.
*
* If IgnoreMCM is set to No, Setup generates an error
* and exits if it detects an incompatible version
* of Citrix Conferencing Manager.
*
* Set IgnoreMCM to Yes, if you do not want Setup to
* generate an error and exit if it detects an incompatible
* version of Citrix Conferencing Manager.
*
* Versions of Citrix Conferencing Manager prior to
* version 3.0 are not compatible with Presentation
* Server. Before upgrading Presentation Server,
* upgrade Conferencing Manager to version 3.0 or 4.0. For
* more information, see the Citrix Conferencing
* Manager Administrator's Guide.
*
* - RemoveWITurnkey -
*
* The preferred method for upgrading Citrix Web Interface is to
* upgrade to Access Suite Console 4.5 and Web Interface 4.5 before
* installing Presentation Server 4.5.
*
* RemoveWITurnkey determines whether or not Setup shows an error
* message during the upgrade after detecting that you installed
* Web Interface as part of the earlier installation.
*
* Supported values:
*
*      Yes      Remove Web Interface if it is detected on the
*                system.
*      No       Exit the installation if Setup detects the Web
*                Interface on the system (default).
*
* If RemoveWITurnkey is set to No and Setup detects an
* incompatible version of Web Interface installed on the
* system, Setup exits with an error message.
*
* If you want to ignore the error message, set RemoveWITurnkey
* to Yes. Please note that at the end of the upgrade, the
* Web Interface is no longer installed on the system.
*
* If you want to specify a different install
* location (INSTALLDIR) than the default one, you must
* set InstallLocation="Target Path" For example:
* InstallLocation="C:\Program Files\abc abc\"
*
```

```

* If you do not set InstallLocation, the product will
* be installed in the default location: \Program Files\Citrix.
*****
[Options]
RebootOnFinish=Yes
LogLevel=*v
LogFile=c:\msi.log
UILevel= BASIC_UI_NO_MODAL
IgnoreMCM=No
RemoveWITurnkey=No
*****
* Presentation Server
*
* ServerType specifies the edition of Presentation Server that you
* are installing.
*
* Supported values:
*   Platinum (for Platinum Edition)
*
*****
[PresentationServer]
ServerType=Platinum
*****
* Citrix License Server
*
* This section describes the Citrix License Server settings.
*
* If CreateFarm is set to Yes, supported values for
* LicenseServerChoice are:
*
* - "Point"
* - "DontKnow"
*
* If CreateFarm is set to No, supported values for LicenseServerChoice
* are:
*
* - "Point"
* - "UseFarmSettings"
* - "DontKnow"
*
* If you set LicenseServerChoice to "Point," you must set
* LicenseServerName to either the name of the license server or its
* IP address.
*
* If you want to set a different license server port from the
* default value, you must set both LicenseServerPortDefault=No and
* LicenseServerPort. For example:
*
* LicenseServerPortDefault=No
* LicenseServerPort=27009
*
*****
[LicenseServer]
LicenseServerChoice=Point
LicenseServerName=localhost
LicenseServerPortDefault=Yes
LicenseServerPort=27000
*****
*

```

```
* MFRDP (Remote Desktop Protocol)
*
* Set DisableRDPPromptForPassword to "Yes" if you enable
* Terminal Services to accept credentials directly from the
* Remote Desktop Windows Client.
*
* If you set DisableRDPPromptForPassword to "No" users are
* prompted for credentials each time they connect using
* Remote Desktop Protocol.
*
*****
[MFRDP]
DisableRDPPromptForPassword=Yes
*****
*
* IMA Encryption
*
* Setting EncryptionEnable to 1 instructs Setup to enable IMA
* Encryption for the farm during a Create Farm server installation.
*
* This option is only relevant for a Create Farm installation. If you
* need to enable encryption on a join, use the console. The default
* is 0.
*
* Set KeyType to "generate," "file," or "existing" to indicate how
* Setup will receive the IMA Encryption settings. The default is "file."
*
* Set NewKeyPath to the complete path where you want the generated key to
* be saved. This option is only available when using KeyType=generate.
*
* Set KeyPath to the location where an already-generated key will be loaded.
* This option is only available when using KeyType=file.
*
* The KeyPath, NewKeyPath, and KeyType attributes are useful only when they
* are used in conjunction. Setting one of the options and not the other will
* result in the encryption key not being loaded.
*
*****
[IMAEncryption]
EncryptionEnable=0
KeyType=file
NewKeyPath=
KeyPath=
```