

# HP OpenView Operations for UNIX White Paper

## Deploying OVO HTTPS Agents Using Radia



Abstract.....	3
About Radia and OVO .....	3
Radia .....	3
OVO .....	3
OVO HTTPS agent .....	3
Before you start .....	4
Compatibility.....	4
Deploying OVO HTTPS agent software using Radia.....	5
Creating a clone image of the OVO HTTPS agent software.....	5
Uploading the OVO agent software clone image into Radia Packager.....	6
Creating an application, service, and helper class in Radia System Explorer.....	10
Create an application for the OVO agent software clone image in Radia System Explorer.....	11
Create a service for the OVO agent software clone image in Radia System Explorer.....	13
Create a helper class for variable replacement during deployment.....	16
Link the helper instance to the service for the OVO agent software clone image.....	18
Deploying the OVO agent software clone image using Radia Management Portal.....	19
Modify the OVO HTTPS agent policy in Radia.....	19
Notify the target device.....	23
Verify the OVO HTTPS agent installation on the target device.....	25
Activate the target device as managed node in OVO .....	25
Deploying OVO HTTPS policies and instrumentation using Radia .....	26
Creating a clone image of OVO HTTPS policies and instrumentation files .....	26
Uploading OVO HTTPS policies and instrumentation files .....	27
Creating an application and a service in Radia .....	28
Create an application for the OVO policy and instrumentation clone image.....	28
Create a service for the OVO policy and instrumentation clone image.....	28
Deploying the OVO HTTPS policies and instrumentation files.....	30
Deploying OVO agent patches .....	31
Deploying parameterized threshold monitor policies .....	32
About parameterized threshold monitor policies .....	32
Parameterizing a threshold monitor policy.....	33

Configuring Radia to deploy parameterized threshold monitor policies.....	34
Edit the installation and de-installation helper files .....	34
Update Radia with the parameterized policy .....	35
Edit the helper class for variable replacement .....	36
Link the helper instance to the service.....	37
Deploy the parameterized threshold monitor policy .....	38
Example: parameterized threshold monitor policy .....	39
Glossary .....	41
For more information .....	43
Call to action .....	43

## Abstract

This white paper explains how to use Radia to install an OVO HTTPS agent for Windows. The first part of this white paper describes how to install the OVO HTTPS agent software for Windows. The second part explains how to deploy OVO HTTPS policies and instrumentation files to the managed nodes. The third part suggests how to maintain OVO managed nodes with OVO agent patches. And the fourth part reveals how you can use advanced scripting in threshold monitor policies to parameterize the threshold values.

In environments with both OVO and Radia installed, it can be advantageous to use Radia to install the OVO HTTPS agent on managed devices. Using Radia's distribution mechanisms is faster, less labor-intensive, and more reliable than OVO's agent installation methods when thousands of managed nodes must be updated.

To deploy an OVO HTTPS agent, you must first create clone images of the OVO agent software, and the policy and instrumentation files. To make a clone image as generic as possible, you can parameterize node-specific information, for example in the agent profile file and also in the policy files. In parameterized files, certain information is stored in variables for which you specify values before deployment. Radia replaces the variables with the specified values during deployment.

When the clone images are complete, you upload them into Radia, create Radia applications and services for the OVO packages and files, and provide helper classes which are needed to resolve the variables in the parameterized files during deployment. You then use Radia Management Portal to specify values for the variables and to deploy the packages. To complete the deployment, you activate the managed nodes in OVO so that they start sending messages.

## About Radia and OVO

### Radia

Radia, HP OpenView's Configuration Management solution, automates the management of software such as operating systems, applications, patches, content, and configuration settings to ensure that each computing device is maintained in the right configuration.

### OVO

HP OpenView Operations for UNIX (OVO) monitors, controls, and reports on the availability and performance of your heterogeneous, large-scale IT environment. It consolidates information for all IT components that support your business: network, systems, storage, databases, and applications.

### OVO HTTPS agent

The OVO agent software is the part of OVO that is installed on managed nodes to gather information, process this information, and generate appropriate responses. The OVO HTTPS agent is new with OVO for UNIX 8. It uses the HTTPS communication protocol and complements the already well-established OVO DCE agent.

## Before you start

This white paper describes advanced techniques that require a thorough understanding of both OVO and Radia. In addition, you must be familiar with the *Installing OVO Agents Using Clone Images* white paper and the *OVO HTTPS Agents Concepts and Configuration Guide*.

---

### Note

This white paper describes only a small installation scenario with one Windows managed node. The intention is to provide you with an informational framework for deploying OVO HTTPS agents with Radia. Users with advanced knowledge of Radia will be able to refine and expand the techniques described in this document, for example to deploy the OVO HTTPS agent software for other operating systems and platforms.

---

## Compatibility

This white paper is based on the following versions of OVO for UNIX and Radia:

Product	Version
HP OpenView Operations Management Server on HP-UX	8.23 and higher
HP OpenView Operations HTTPS Agent for Windows	8.16 and higher
Radia for Windows	4.0 and higher

# Deploying OVO HTTPS agent software using Radia

To deploy OVO HTTPS agents using Radia, you must complete the following high-level steps. Each step is described in detail in the following sections of this white paper:

1. Creating a clone image (page 5)
2. Uploading the OVO agent software clone image into Radia Packager (page 6)
3. Creating an application, service, and helper class in Radia System Explorer (page 10)
4. Deploying the OVO agent software clone image using Radia Management Portal (page 19)

## Creating a clone image of the OVO HTTPS agent software

Creating a clone image of an OVO HTTPS agent is described in detail in the white paper *Installing OVO Agents Using Clone Images*, which is available from the HP OpenView Product Manuals web site at [http://ovweb.external.hp.com/lpe/doc\\_serv/](http://ovweb.external.hp.com/lpe/doc_serv/). (Select **Operations for UNIX** version **8.x**. Then select the clone image white paper in the list of manuals and click **Open** or **Download**.)

When you create a clone image for deployment with Radia, add an agent profile file that has been created using the following instructions:

1. Use a text editor to create a text file named `OVO_agt_profile.txt`.

Note that the file name is not important but it may be useful if the name reflects the contents of the file. The `.txt` suffix lets you easily view the file on Windows systems.

2. Add the following lines to the file:

```
set sec.core.auth:MANAGER_ID=&(APPINFO.OVOM_ID)
set sec.core.auth:MANAGER=&(APPINFO.OVOMGR)
set sec.cm.client:CERTIFICATE_SERVER=&(APPINFO.OVOM_CER)
```

The variables (starting with the ampersand (&)) will be replaced by Radia during deployment.

3. Add the agent profile file to the clone image.

---

### Note

The instructions in the remainder of this document assume that the OVO agent software clone image will be installed using the automatic certificate installation method, which is the default installation method.

---

## Uploading the OVO agent software clone image into Radia Packager

To upload the OVO agent software clone image into Radia Packager:

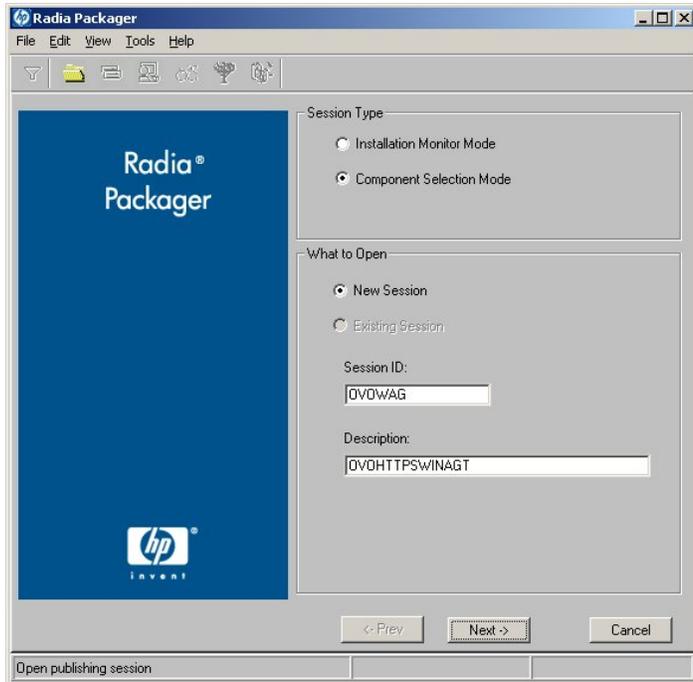
1. Make the OVO agent software clone image available on the system where Radia Administrator Workstation is installed, for example using FTP. For easier maintenance, create a clone directory like C:\ovo\_https\_agt\_win.
2. Create two helper files that will allow Radia to call the OVO agent installation and de-installation scripts on the client computers:
  - Installation helper file:
    - a. Use a text editor to create ovo\_agt\_install.bat.
    - b. Add the following lines:

```
cd <clone_image_directory_on_client_computer>
%SystemRoot%\system32\cscript.exe opc_inst.vbs -non_int
-configure <OVO_agent_profile>
```
    - c. Add the helper file to the clone image.
  - De-installation helper file:
    - a. Use a text editor to create ovo\_agt\_remove.bat.
    - b. Add the following lines:

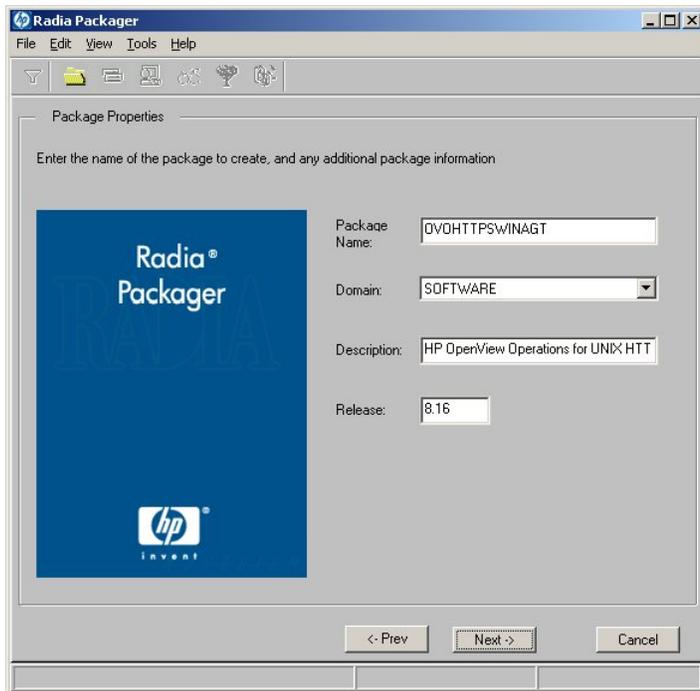
```
cd <clone_image_directory_on_client_computer>
%SystemRoot%\system32\cscript.exe opc_inst.vbs -non_int
-remove
```
    - c. Add the helper file to the clone image.
3. Start Radia Packager:

Click **Start > All Programs > Radia Administrator Workstation > Radia Packager**. The Open Publishing Session window displays.
4. In the Open Publishing Session window, do the following:
  - a. Click **Component Selection Mode**.
  - b. Click **New Session**.

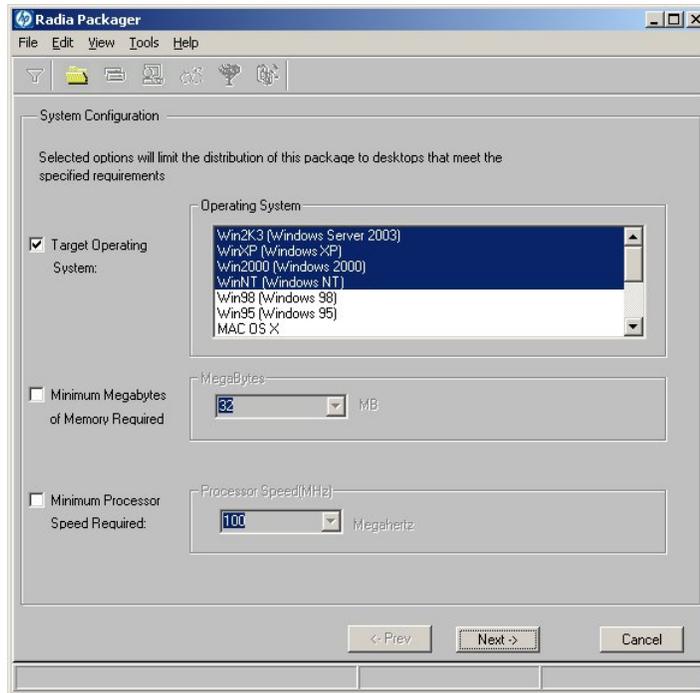
- c. Type a session ID and description into the corresponding fields.
- d. Click **Next**.



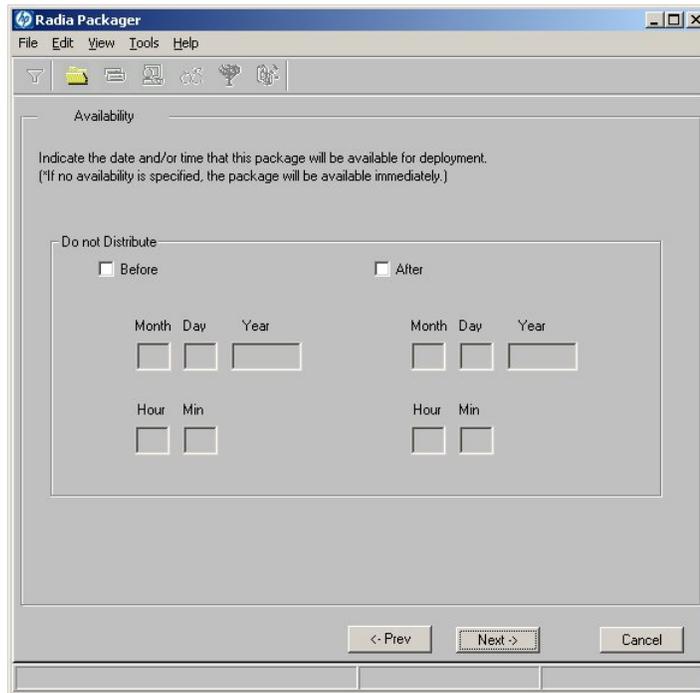
- 5. In the Package Properties window, specify the package name, domain, description, and release, then click **Next**.



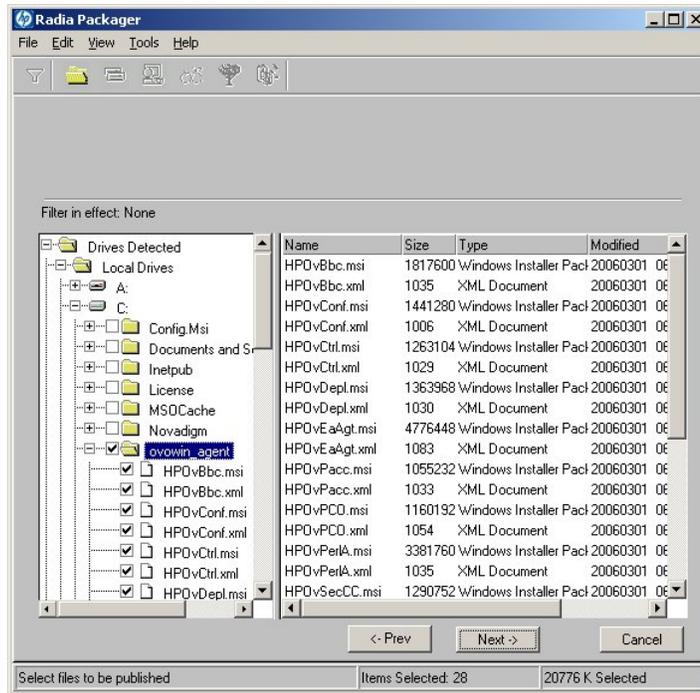
6. In the System Configuration window, select the target operating systems, then click **Next**.



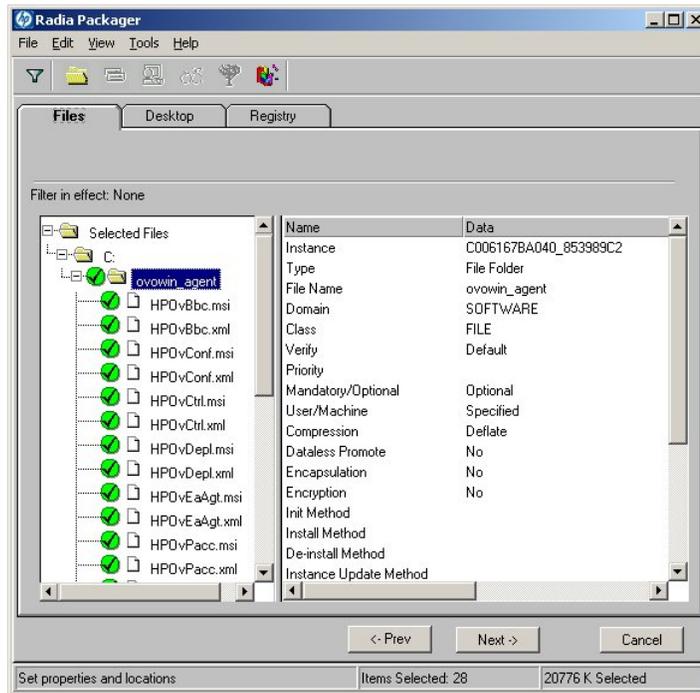
7. In the Availability window, click **Next**. (Do not specify anything in this window.)



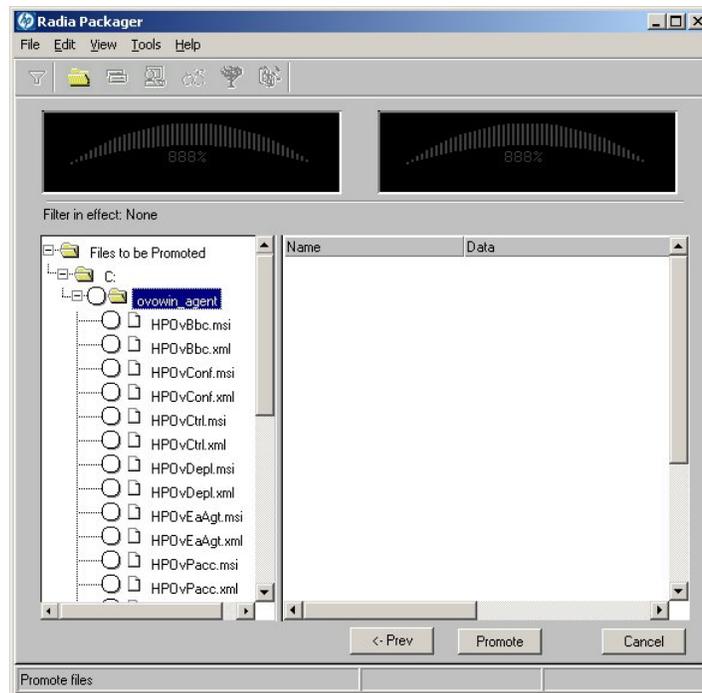
8. Select the files to be published, then click **Next**.



9. Verify that all required files are selected, then click **Next**.



10. Click **Promote** to add the files to the Radia Database.



11. Radia displays a message to inform you that the files were promoted to the database successfully. Click **Finish** to complete the session and close Radia Packager.

## Creating an application, service, and helper class in Radia System Explorer

Use Radia System Explorer to create an application package and a service for the agent software package. To resolve the variables of the OVO agent profile file, create a helper class and instance, and link the helper instance to the service.

The following list gives an overview of the high-level steps that must be completed. See the following sections for detailed information about each step:

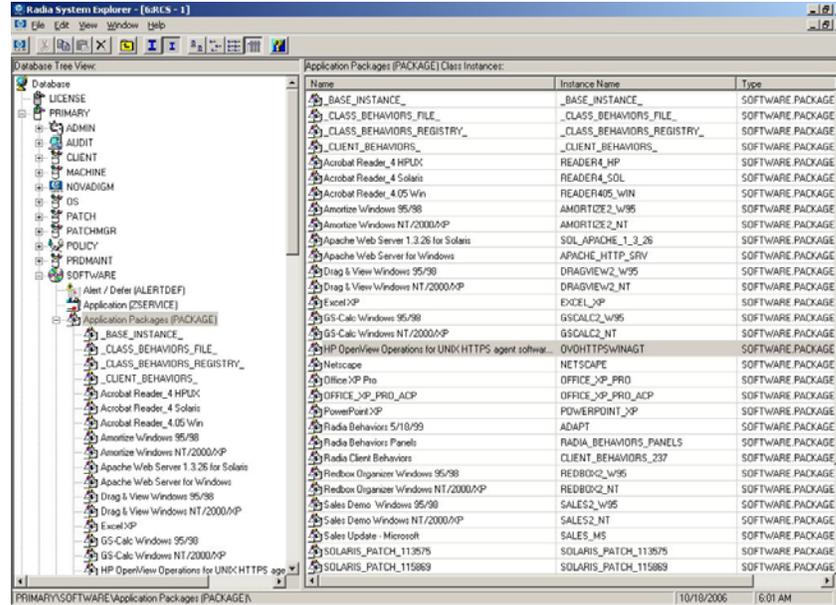
1. Create an application for the OVO agent software clone image in Radia System Explorer (page 10)
2. Create a service for the OVO agent software clone image in Radia System Explorer (page 13)
3. Create a helper class for variable replacement during deployment (page 16)
4. Link the helper instance to the service for the OVO agent software clone image (page 18)

## Create an application for the OVO agent software clone image in Radia System Explorer

To create an application in Radia System Explorer:

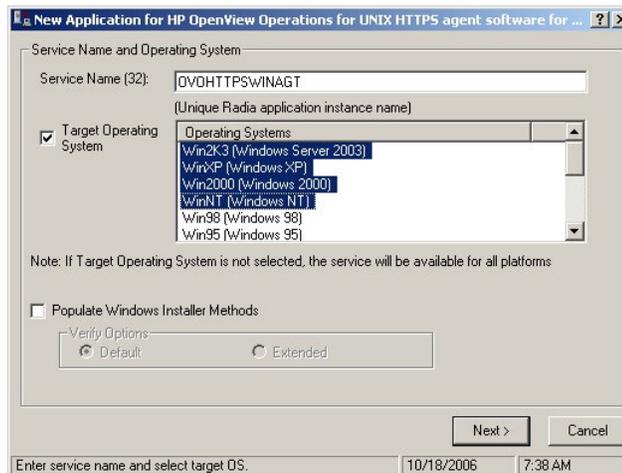
1. Start Radia System Explorer:

Click **Start > All Programs > Radia Administrator Workstation > Radia System Explorer**. Radia System Explorer opens.



2. In the Database Tree View, click **PRIMARY > SOFTWARE > Application Packages (PACKAGE)**, then right-click your OVO HTTPS agent package, and click **New Application Wizard...** in the shortcut menu. The New Application wizard opens and displays the Service Name and Operating System window.

3. Type the name of OVO HTTPS agent package into the **Service Name (32)** field. Then select the operating systems of the target devices and click **Next**.



4. Select **Application Manager** as application target type and click **Next**.

The screenshot shows a dialog box titled "New Application for HP OpenView Operations for UNIX HTTPS agent software for ...". The "Application Target Type" section has two options: "Application Manager" (selected with a checked checkbox) and "Software Manager" (unchecked). Under "Application Manager", there are several sub-features listed: "Just-In-Time" (Transparent real time automated management), "Fixed Scheduling" (Routine, reliable, scheduled update delivery), "Mandatory Services" (Automatic application installation), "Central Notification" (Immediate delivery of application updates), and "Versioning" (Rollback/forward of new versions). Under "Software Manager", there are: "User Catalog" (User application management control), "Adaptability" (Automatically adapt to situational specific conditions), "Personalization" (Establish and change application preferences), and "Updates" (User controls when updates are applied). A "Note" at the bottom states: "If an application's features require products not licensed on the target machine, either the application may not be installed or may be installed with limited settings." At the bottom of the dialog are buttons for "< Previous", "Next >", and "Cancel". The status bar at the very bottom shows "Choose the application target type", the date "10/18/2006", and the time "7:45 AM".

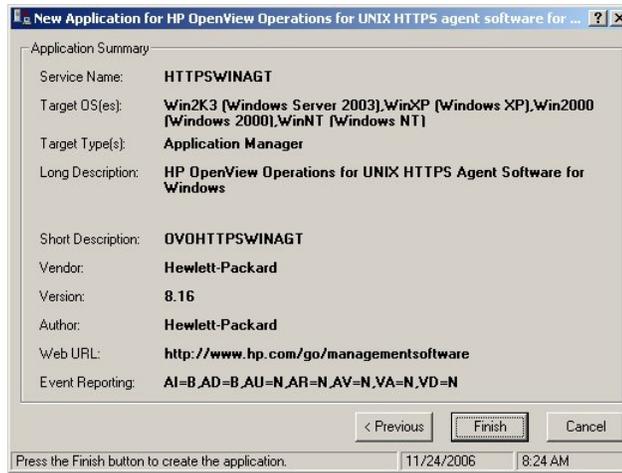
5. Type the properties of the application into the Application Properties window and click **Next**.

The screenshot shows the "Application Properties" tab of the same dialog box. It contains several text input fields: "Service Name" (HTTPSWINAGT), "Long Description" (HP OpenView Operations for UNIX HTTPS Agent Software for Window), "Short Description" (OVOHTTPSWINAGT), "Vendor" (Hewlett-Packard), "Version" (8.16), "Author" (Hewlett-Packard), and "Web URL" (http://www.hp.com/go/managementsoftware). At the bottom are buttons for "< Previous", "Next >", and "Cancel". The status bar at the bottom shows "Enter the application properties", the date "11/24/2006", and the time "8:22 AM".

6. Select the events that the Radia configuration client will report and click **Next**.

The screenshot shows the "Application Level Event Reporting" tab. It features a section titled "The Client Should Report the Following Application Level Events:" followed by a list of events. Each event has a checkbox on the left and three radio buttons on the right for "Success", "Failure", and "Both". The events are: "Application Installation" (checked), "Application Deinstallation" (checked), "Application Update" (unchecked), "Application Repair" (unchecked), "Application Verify" (unchecked), "Version Activation" (unchecked), and "Version Deactivation" (unchecked). For "Application Installation" and "Application Deinstallation", the "Both" radio button is selected. At the bottom are buttons for "Use Base" and "Save as Default". Below these are buttons for "< Previous", "Next >", and "Cancel". The status bar at the bottom shows "Select the events the client should report on.", the date "10/18/2006", and the time "7:49 AM".

- Review the application summary and click **Finish**.

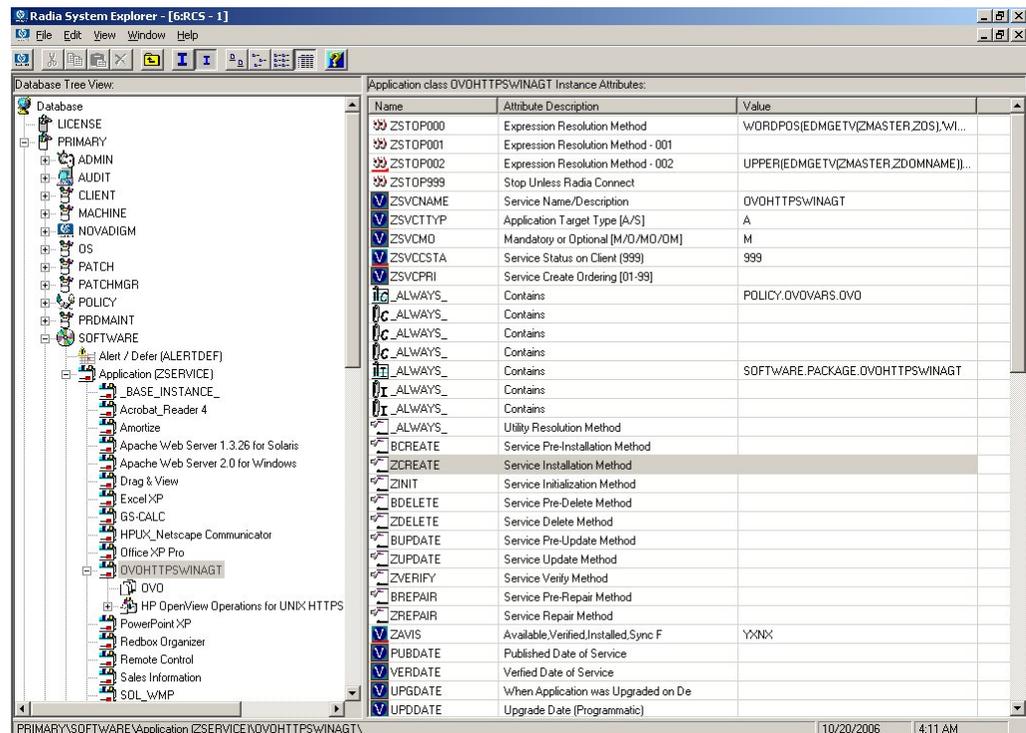


Radia System Explorer confirms the fact that your application has been successfully added.

### Create a service for the OVO agent software clone image in Radia System Explorer

To create a service in Radia System Explorer:

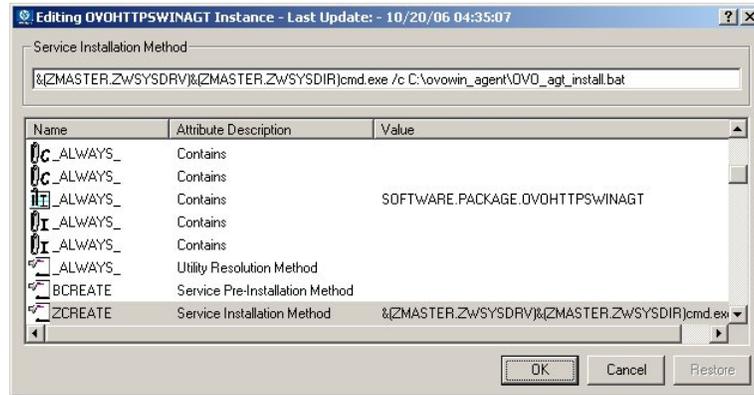
- In the Database Tree View, click **PRIMARY > SOFTWARE > Application (ZSERVICE)**, then double-click your OVO HTTPS agent software package to expand it.



2. Double-click the **ZCREATE** attribute and type the name and path of your OVO agent installation helper file into the **Service Installation Method** field:

```
& ( ZMASTER . ZWSYSDRV ) & ( ZMASTER . ZWSYSDIR ) cmd . exe /c  
<OVO_install_helper_file>
```

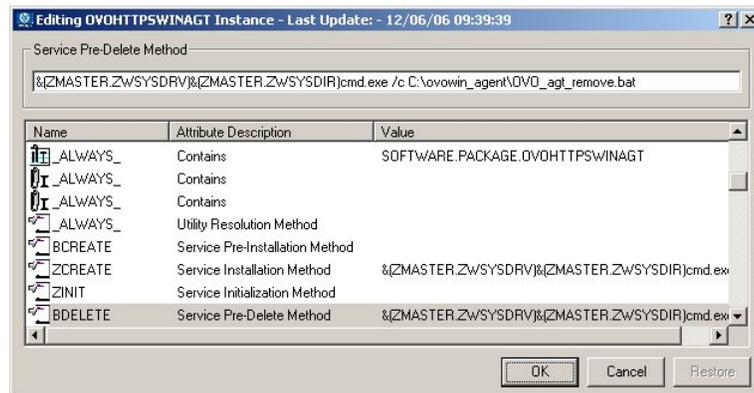
This is the `ovo_agt_install.bat` file you created before uploading the clone image into Radia Packager (see page 6).



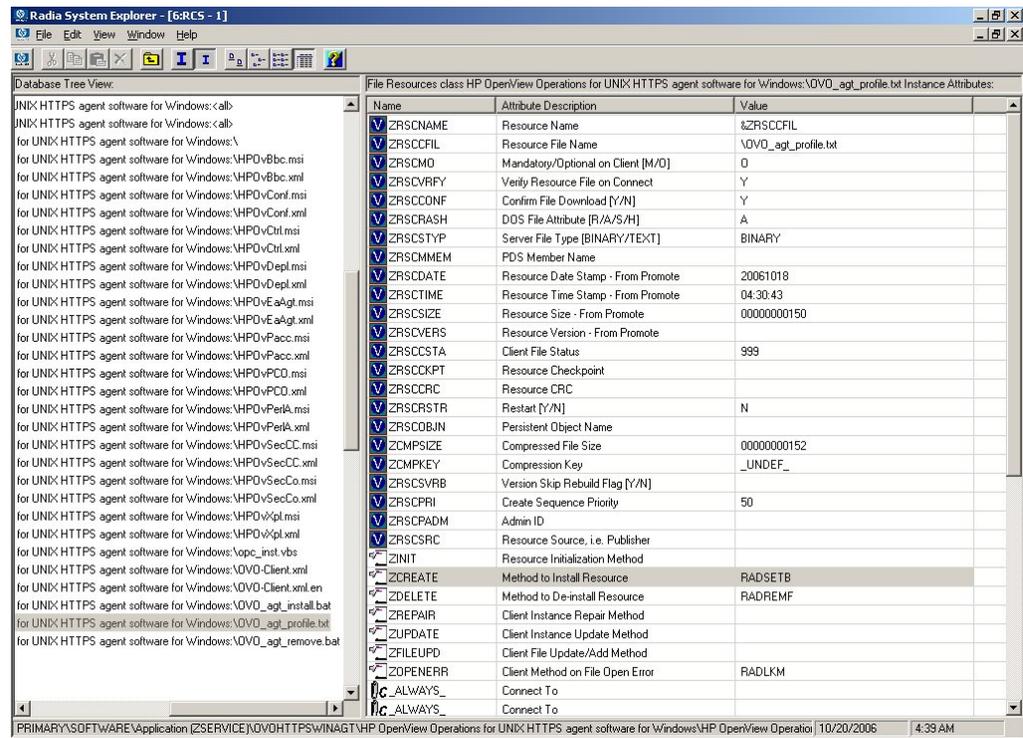
3. Double-click the **BDELETE** attribute and type the name and path of your OVO agent de-installation helper file into the **Service Pre-Delete Method** field:

```
& ( ZMASTER . ZWSYSDRV ) & ( ZMASTER . ZWSYSDIR ) cmd . exe /c  
<OVO_remove_helper_file>
```

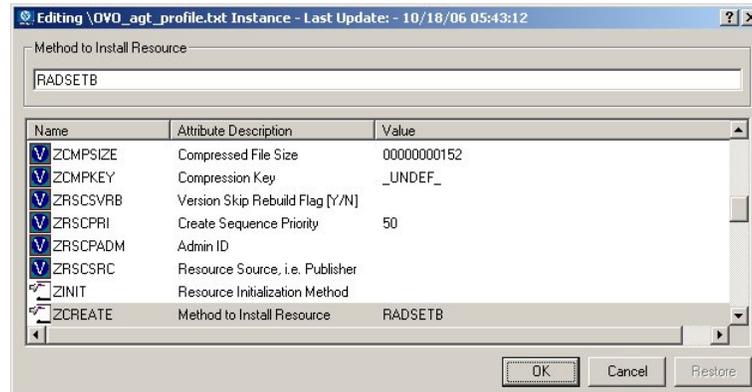
This is the `ovo_agt_remove.bat` file you created before uploading the clone image into Radia Packager (see page 6).



- In the Database Tree View, double-click your OVO HTTPS agent package to display a list of all files that are included in the package. Then double-click the line containing the `OVO_agt_profile.txt` file to display a list of its instance attributes.



- Double-click the **ZCREATE** attribute and type **RADSETB** into the **Method to Install Resource** field. This ensures that Radia will be able to replace the variables in the `OVO_agt_profile.txt` file during deployment. Click **OK**.



## Create a helper class for variable replacement during deployment

To create a helper class for variable replacement:

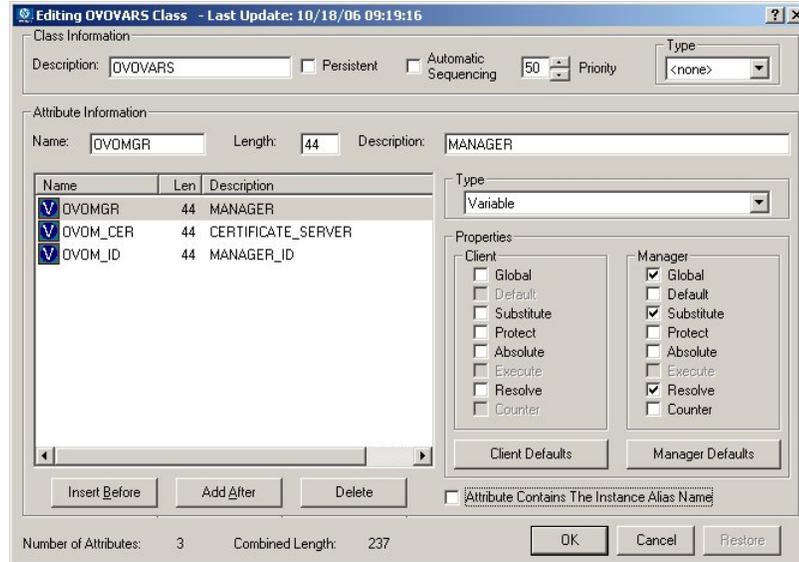
1. In the Database Tree View, click **PRIMARY > POLICY**. Then right-click **POLICY** and click **New Class** in the shortcut menu. The Create Class window opens.
2. Type a name for your new policy class and click **OK**.



3. An editor for the new class opens. Specify the following information in this editor:
  - a. Select **<none>** as class type.
  - b. Add the variables listed in the `OVO_agt_profile.txt` file as attributes to the class:

Name	Length	Description
OVOM_ID	44	MANAGER_ID
OVOMGR	44	MANAGER
OVOM_CER	44	CERTIFICATE_SERVER

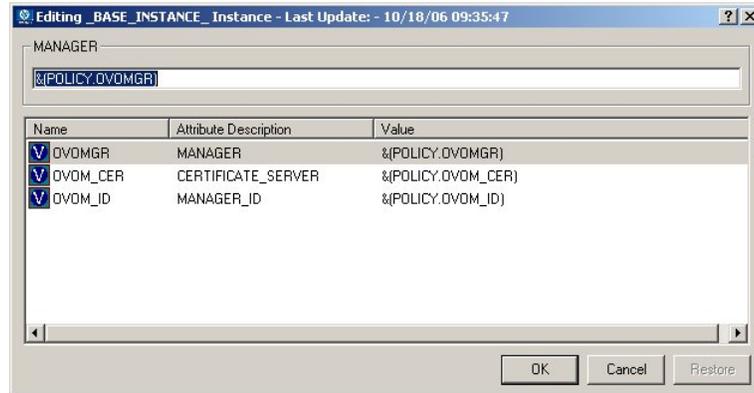
- c. For each variable, select **Resolve** in the manager properties section of the window
- d. Click **OK** to save your class and close the window.
- e. When prompted, confirm that you want to save your class.



4. Double-click your new class in the Database Tree View to expand it. Then right-click **\_BASE\_INSTANCE** and click **Edit Instance...** in the shortcut menu. An editor for this instance opens.

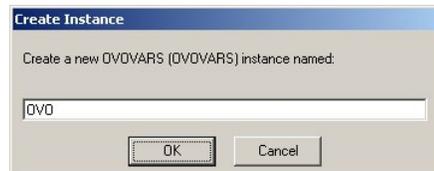
5. Specify a value for each attribute of the base instance, then click **OK**:

Name	Attribute Description	Value
OVOM_ID	MANAGER_ID	&(POLICY.OVOM_ID)
OVOMGR	MANAGER	&(POLICY.OVOMGR)
OVOM_CER	CERTIFICATE_SERVER	&(POLICY.OVOM_CER)



6. Right-click your class and click **New Instance** in the shortcut menu. The Create Instance window opens.

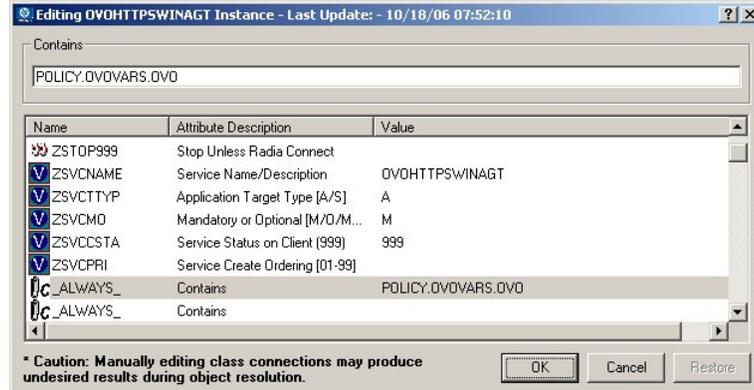
7. Type a name for your new helper instance and click **OK**.



## Link the helper instance to the service for the OVO agent software clone image

To link the helper instance to the service:

1. In the Database Tree View, click **PRIMARY > SOFTWARE > Application Packages (ZSERVICE)**, then double-click your OVO HTTPS agent package to expand it.
2. Double-click the attribute **\_ALWAYS\_** and type `POLICY.<class_name>.<instance_name>` into the **Contains** field, then click **OK**.



# Deploying the OVO agent software clone image using Radia Management Portal

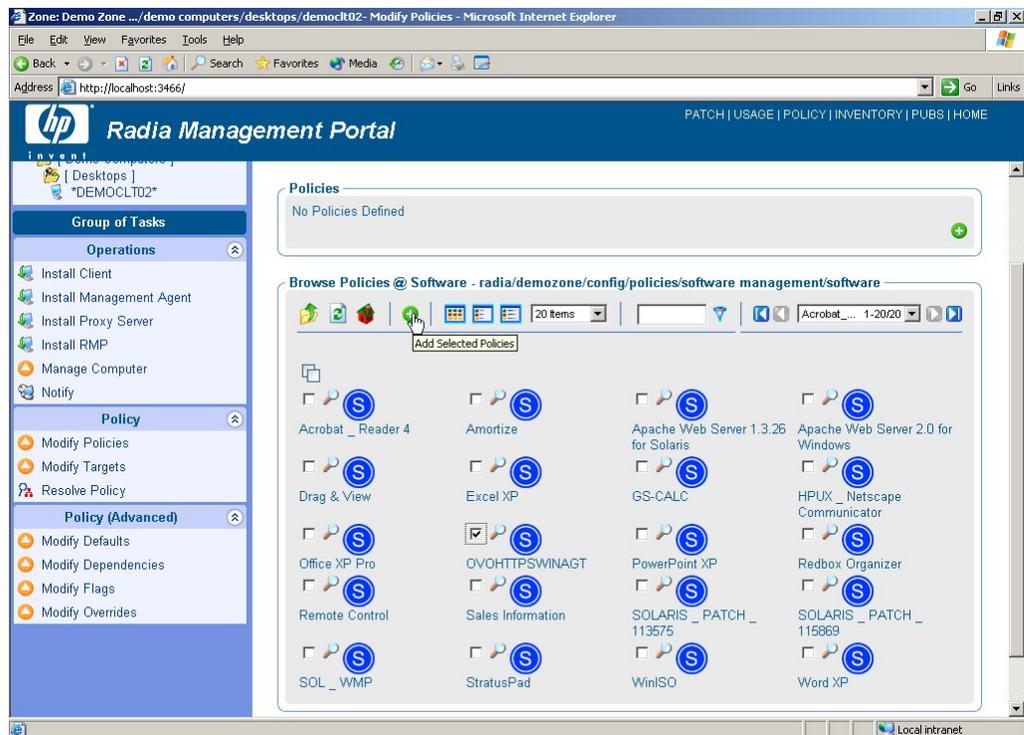
This procedure assumes that the target devices have already been set up in Radia and have the Radia Client installed. To deploy the OVO HTTPS agent to a target device, perform the following high-level steps. Each step is described in more detail in the following sections:

1. Modify the OVO HTTPS agent policy (page 19)
2. Notify the target device (page 23)
3. Verify the OVO HTTPS agent installation on the target device (page 25)
4. Activate the target device as managed node in OVO (page 25)

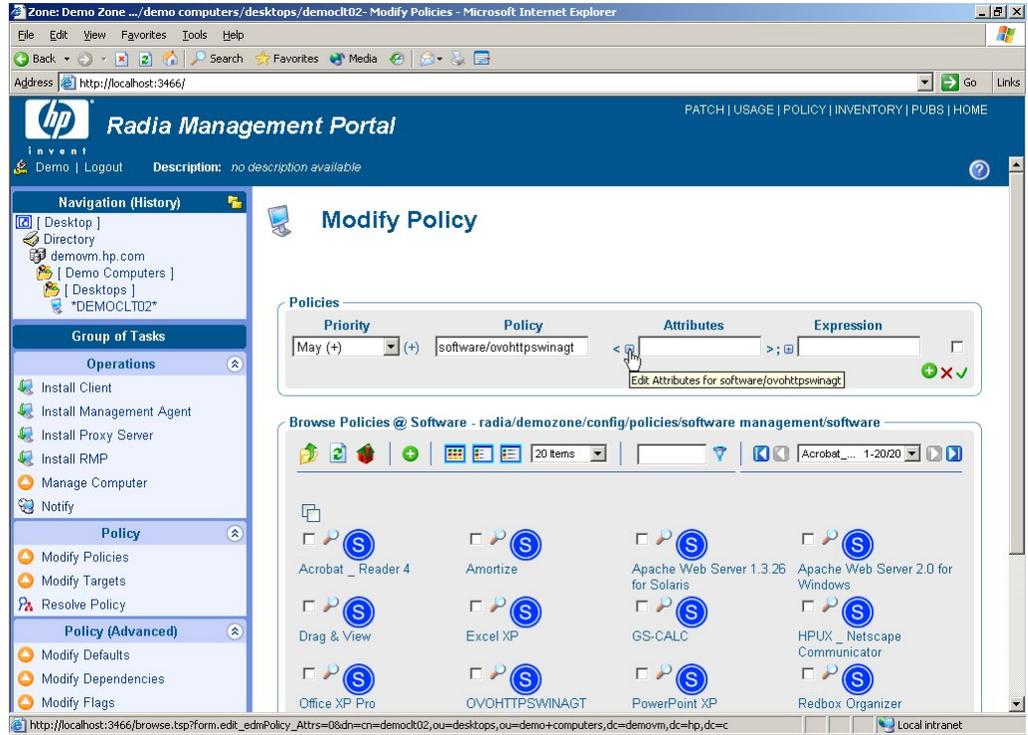
## Modify the OVO HTTPS agent policy in Radia

To modify the OVO HTTPS agent policy:

1. In a Web browser, start Radia Management Portal at <http://localhost:3466>.
2. Navigate to the target device on which you want to install the OVO HTTPS agent application package, then click **Modify Policies**.
3. Navigate to the software policies and select the OVO HTTPS agent policy. Then click  (Add).



4. Click  (Edit Attributes) to the left of the **Attributes** field to open the Attribute Editor.

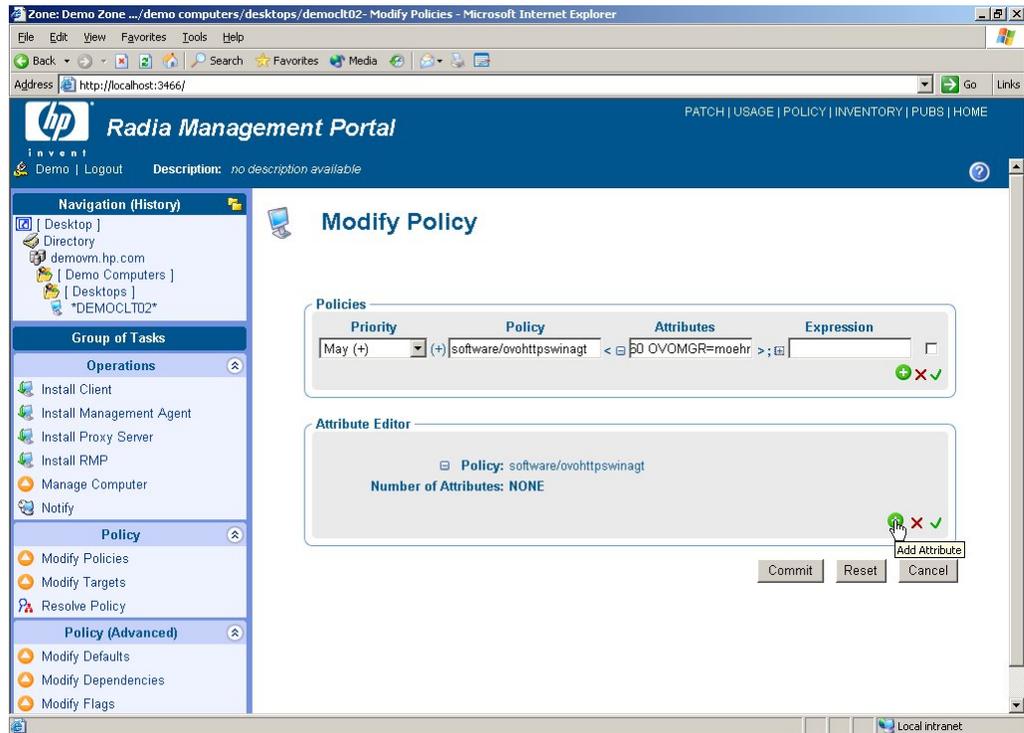


5. In the **Attributes** field, type the variables as name-value pairs, separated by spaces:

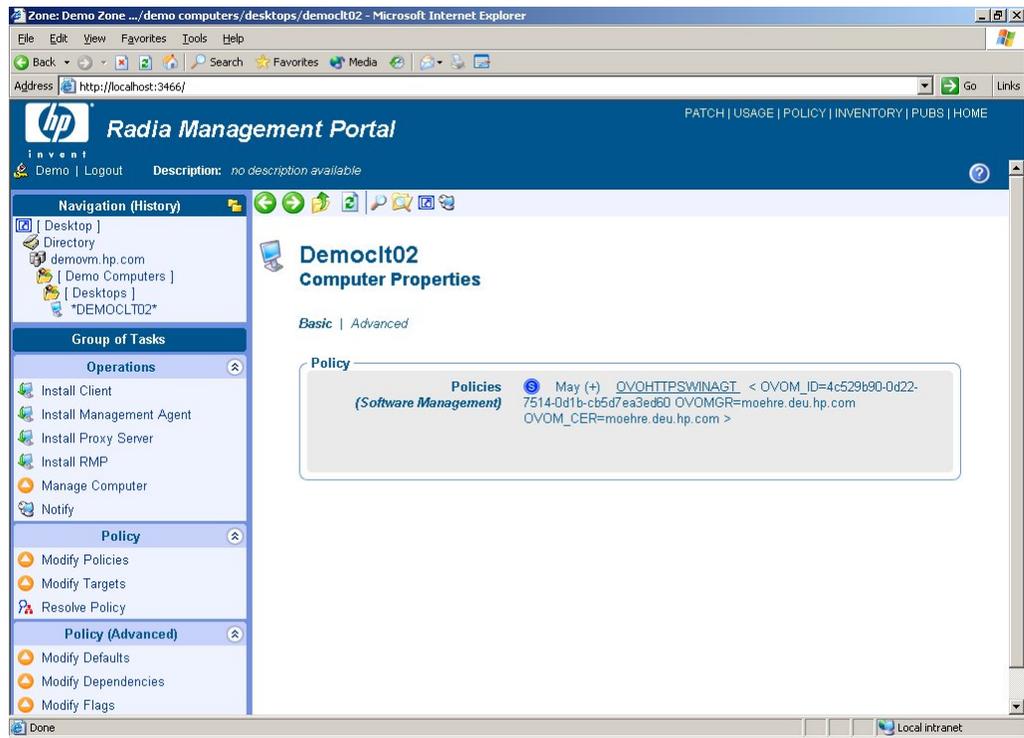
**Name** = **Value**  
OVOM\_ID = OvCoreId of the primary OVO management server  
OVOMGR = Long hostname of the primary OVO management server  
OVOM\_CER = Long hostname of the system where the certificate authority is located

For example: OVOM\_ID=4c529b90-0d22-7514-0d1b-cb5d7ea3ed60  
OVOMGR=moehre.deu.hp.com OVOM\_CER=moehre.deu.hp.com

Then click  (Add) to add the attributes to the policy.



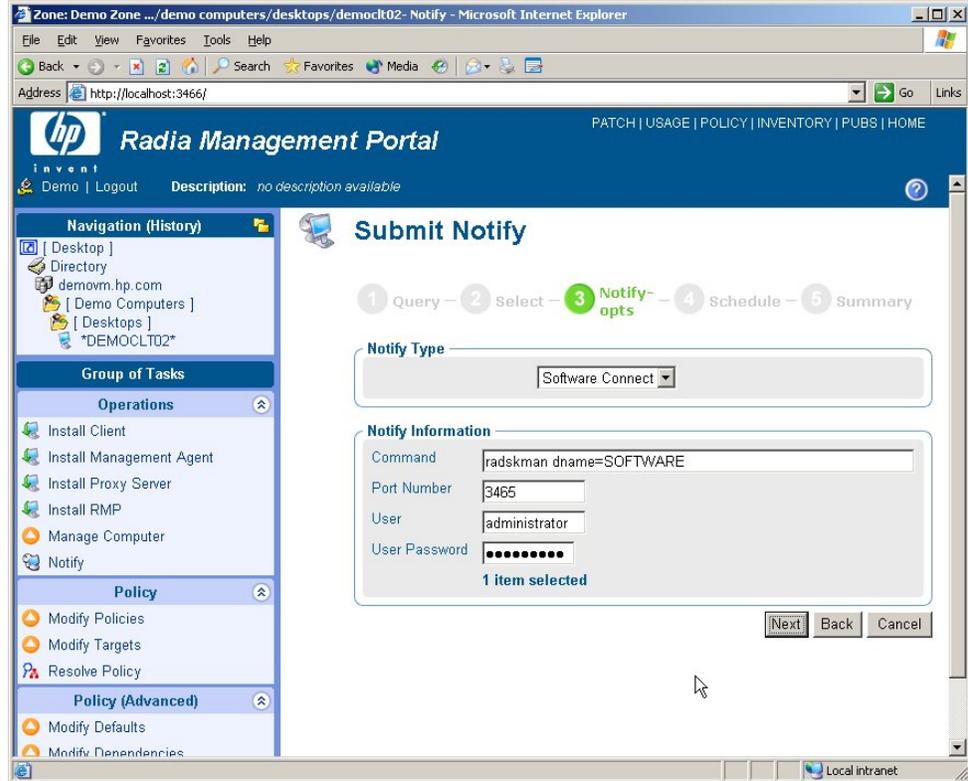
6. Click **Commit** to store the attribute values in the database.  
The properties of the target device are displayed. Verify that the correct attributes and values are associated with the policy for the OVO HTTPS agent package.



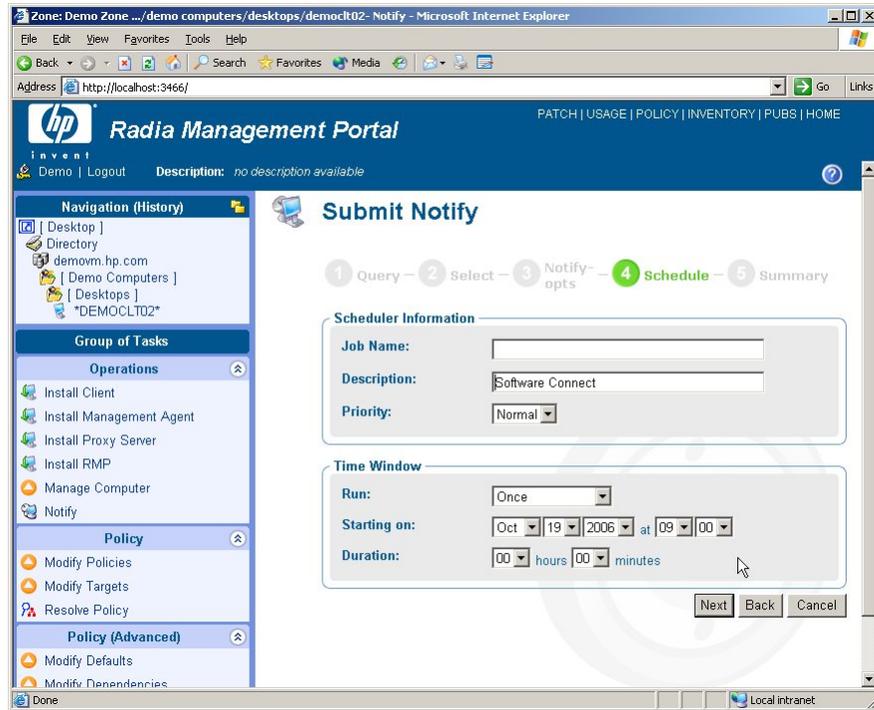
## Notify the target device

To notify the target device:

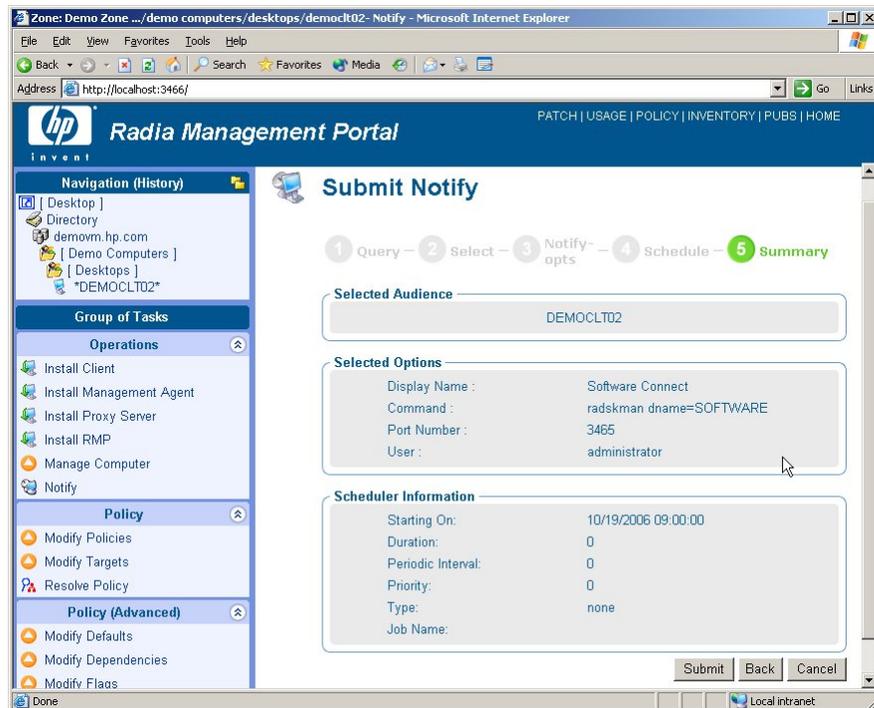
1. Click **Notify** to notify the target device that the OVO HTTPS agent package is ready for deployment.
2. Select Software Connect as notify type and type the user name and password to be used for logging into the target device. Then click **Next**.



3. Do not modify the schedule information. Click **Next**.



4. Review the summary of the notify task, then click **Submit** to start deploying the OVO HTTPS agent package.



Radia starts the deployment process and will keep you informed about the progress and results.

## Verify the OVO HTTPS agent installation on the target device

To verify the OVO HTTPS agent installation:

5. Verify that the variables now have the correct values in the deployed agent profile file: change to the clone directory and view the agent profile file in a text editor.
6. Verify that the certificate server has been set correctly, type: `ovconfget sec.cm.client`  
The output should be similar to the following: `CERTIFICATE_SERVER=moehre.deu.hp.com`

## Activate the target device as managed node in OVO

To activate the target device as managed node in OVO:

See the *Installing OVO Agents Using Clone Images* white paper for more information about activating the managed nodes on the management server. The activation process involves the following high-level steps:

1. Add the target devices as managed nodes to the node bank.
2. Add the managed nodes to the required node groups.
3. Grant the certificate requests of the managed nodes.
4. Mark the managed node as installed in the OVO database.
5. Start heartbeat monitoring on the managed nodes.

# Deploying OVO HTTPS policies and instrumentation using Radia

To deploy policies and instrumentation for OVO HTTPS agents using Radia, you must complete the following high-level steps. Each step is described in more detail in the following sections:

1. Creating a clone image of OVO HTTPS policies and instrumentation (page 26)
2. Uploading OVO HTTPS policies and instrumentation (page 27)
3. Creating an application and a service (page 28)
4. Deploying the OVO HTTPS policies and instrumentation files (page 30)

## Creating a clone image of OVO HTTPS policies and instrumentation files

Downloading policies and instrumentation files is described in detail in the white paper *Installing OVO Agents Using Clone Images*. Use the command-line tools `opctmpldwn` and `opcinstrumdwn` to download the required configuration for the managed node for which you created the agent software clone image. It is recommended that you create a separate clone image for the downloaded policies and instrumentation files for easier maintenance in Radia.

The downloaded policies are signed with the certificate of the OVO management server that is responsible for the managed node, and can therefore only be installed on managed nodes that have the same management server. However, if multiple OVO management servers share the same certificate server, you can install the policies on all managed nodes that report to these servers.

## Uploading OVO HTTPS policies and instrumentation files

Uploading OVO HTTPS policies and instrumentation files into Radia Packager is similar to uploading the OVO HTTPS agent software clone image. Instead of creating agent software installation and de-installation helper files, you must create helper files for policies and instrumentation files.

1. Make the policies and instrumentation files available on the system where Radia Administrator Workstation is installed, for example using FTP. For easier maintenance, create a clone directory with the following structure:  
C:\ovo\_https\_policies\_instr\_win\*<policies>*  
C:\ovo\_https\_policies\_instr\_win\instrumentation\*<instrumentation>*
2. Create two helper files that will allow Radia to install and de-install the policies and instrumentation files on the client computers:
  - Installation helper file:
    - a. Use a text editor to create `ovo_pol_install.bat`.  
Add the following lines:

```
rem install instrumentation
copy c:\ovo_https_policies_instr_win\instrumentation
"C:\Program Files\HP OpenView\data\bin\instrumentation"

rem install policies
ovpolicy -install -dir C:\ovo_https_policies_instr_win
-add-category My_OVO_RADIA_group
```
    - b. Add the helper file to the clone image directory.
  - De-installation helper file:
    - a. Use a text editor to create `ovo_pol_remove.bat`.
    - b. Add the following lines:

```
rem remove policy
ovpolicy -remove -category My_OVO_RADIA_group
```

Do not add any commands to remove the instrumentation files from the managed nodes because the files may be shared with other HP OpenView applications. The OVO agent software de-installation automatically removes all instrumentation files.
    - c. Add the helper file to the clone image.
3. Use Radia Packager to upload the policy and instrumentation clone image into Radia. If you need assistance with this task, see [Uploading the OVO agent software clone image into Radia Packager](#) on page 6 for a similar task.

## Creating an application and a service in Radia

Use Radia System Explorer to create an application package and a service for the policy and instrumentation package.

The following list gives an overview of the high-level steps that must be completed. See the following sections for detailed information about each step:

1. Create an application for the OVO policy and instrumentation clone image (page 28)
2. Create a service for the OVO policy and instrumentation clone image (page 28)

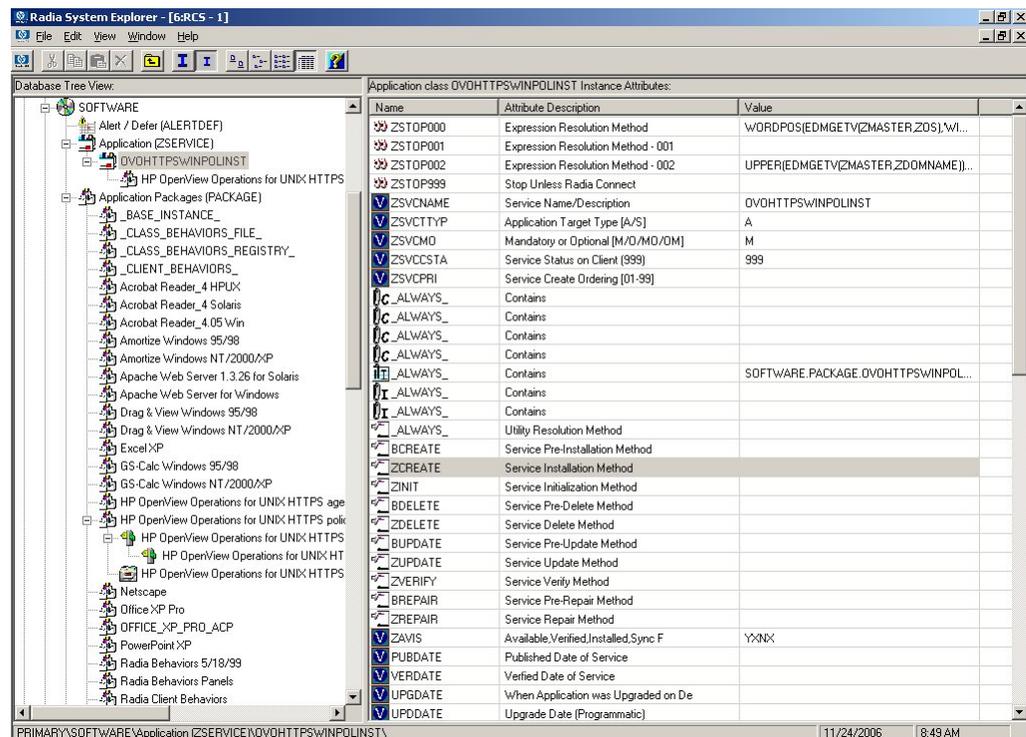
### Create an application for the OVO policy and instrumentation clone image

Create an application for the OVO policy and instrumentation clone image in Radia System Explorer. If you need assistance with this task, see Create an application for the OVO agent software clone image in Radia System Explorer on page 11 for more information.

### Create a service for the OVO policy and instrumentation clone image

To create a service in Radia System Explorer:

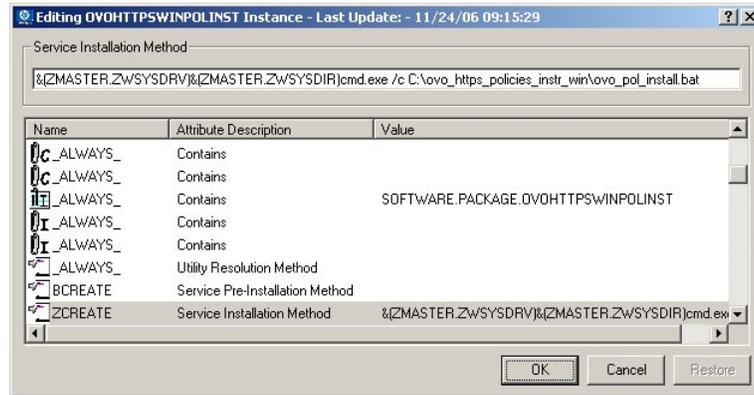
1. Start Radia System Explorer:  
Click **Start > All Programs > Radia Administrator Workstation > Radia System Explorer**. Radia System Explorer opens
2. In the Database Tree View, click **PRIMARY > SOFTWARE > Application (ZSERVICE)**, then double-click your OVO HTTPS policy and instrumentation package to expand it.



3. Double-click the **ZCREATE** attribute and type the name and path of your OVO policy installation helper file into the **Service Installation Method** field:

```
& (ZMASTER.ZWSYSDRV)&(ZMASTER.ZWSYSDIR)cmd.exe /c  
<OVO_install_helper_file>
```

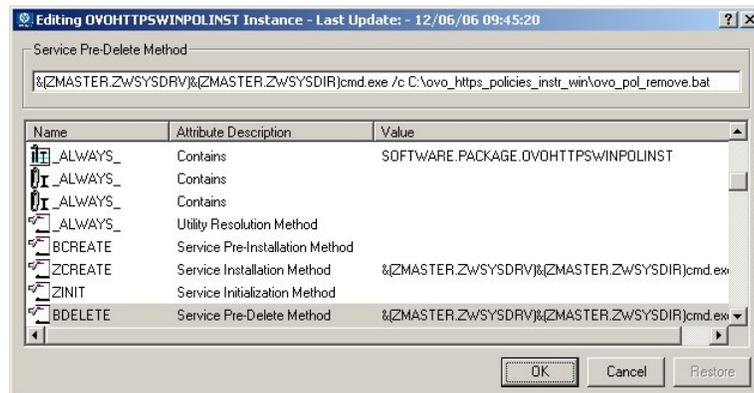
This is the `ovo_pol_install.bat` file you created before uploading the clone image into Radia Packager (see page 27).



4. Double-click the **BDELETE** attribute and type the name and path of your OVO policy de-installation helper file into the **Service Pre-Delete Method** field:

```
& (ZMASTER.ZWSYSDRV)&(ZMASTER.ZWSYSDIR)cmd.exe /c  
<OVO_remove_helper_file>
```

This is the `ovo_pol_remove.bat` file you created before uploading the clone image into Radia Packager (see page 27).



## Deploying the OVO HTTPS policies and instrumentation files

Deploying the OVO HTTPS policies and instrumentation files is similar to deploying the OVO HTTPS agent software package which is described on page 19. Refer to this section if you need help with the following task:

1. Modify the Radia policy for the OVO HTTPS policy and instrumentation package.
2. Commit your changes to the Radia database.
3. Notify the target devices and deploy the package.

## Deploying OVO agent patches

Deploying OVO agent patches to managed nodes is similar to deploying the initial agent software but requires fewer steps:

1. Install the OVO agent patches on the management server.
2. Update the clone image with the patched files.
3. Upload the updated clone image into Radia Packager.
4. Deploy the package.

---

### Tip

Use Radia's versioning mechanisms to keep track of the version number of the OVO agent software.

---

## Deploying parameterized threshold monitor policies

Thresholds in monitor policies may change frequently and may therefore need to be adjusted often or may vary for different managed nodes. This is why you may want to consider parameterizing threshold monitor policies, similar to how the agent profile file was parameterized in [Uploading the OVO agent software clone image into Radia Packager on page 6](#). Before deployment, you can then specify the values you want to use within the policies for a particular group of managed nodes. Radia will replace the variables with the values you entered and update the specified managed nodes.

To parameterize a threshold monitor policy, familiarize yourself with the following sections:

- [About parameterized threshold monitor policies \(page 32\)](#)
- [Parameterizing a threshold monitor policy \(page 33\)](#)
- [Configuring Radia to deploy parameterized threshold monitor policies \(page 34\)](#)
- [Example: parameterized threshold monitor policy \(page 39\)](#)

### About parameterized threshold monitor policies

To parameterize a threshold monitor policy, you must embed a script into the policy, which converts the hard-coded threshold values into parameters and instructs the policy to obtain the values through an `ovconfget eaagt` call.

Embedded scripts are as such not yet supported with OVO for UNIX but you can download an advanced scripting monitor policy from an OVO for Windows management server, adapt it for use with OVO for UNIX, insert the Perl script that parameterizes the threshold monitor values, then upload the policy into Radia, and finally configure Radia to resolve the variables during deployment.

The following section explain how to parameterize a threshold monitor policy for OVO for UNIX.

## Parameterizing a threshold monitor policy

To parameterize a threshold monitor policy:

1. Download an advanced scripting policy from an OVO for Windows management server:
  - a. Download the policy to a directory on the C Drive:  
`ovpmutil CFG POL DNL c:\<dir> /p "\<adv_script_policy_path>"`
  - b. Convert the downloaded structured storage file to an ASCII header and data file:  
`ovpmutil PCV /x "C:\<dir>\<downloaded_adv_script_policy>"`
2. Transfer the downloaded advanced scripting policy to the OVO for UNIX management server.
3. Update the policy header file:
  - a. Copy an existing policy header file from:  
`/var/opt/OV/datafiles/policies/monitor/<uuid>_header.xml`
  - b. Generate a new unique ID for the policy header file name using the HP-UX command-line tool `uuidgen`. See the `uuidgen(1)` man page for more information.
  - c. Rename the copied policy header file with the generated UUID.
  - d. Copy the generated UUID into the policy header:

```
...
<ids>
    <container_id>ed1f81cc-e206-71d8-0c7d-
0f887e0b0000</container_id>
</ids>
...
```
  - e. Sign the policy header file:  
`opctmpldwn -sign <policy_header_file>`
4. Embed the Perl script into the policy data file. See the Example: parameterized threshold monitor policy on page 39.
5. Test the installation of the policy on a managed node:
  - a. Deploy the policy:  
`ovpolicy -install -dir <dir_with_adv_script_policy> \`  
`-host <node>`
  - b. Check that the policy was deployed successfully:  
`ovpolicy -l -level 4 -host <node>`
6. Add the policy to a clone image.

## Configuring Radia to deploy parameterized threshold monitor policies

To configure Radia to be able to deploy parameterized threshold monitor policies, you must complete the following steps:

1. Edit the installation and de-installation helper files (page 34)
2. Update Radia with the parameterized policy (page 35)
3. Edit the helper class for variable replacement (page 36)
4. Link the helper instance to the service (page 37)

### Edit the installation and de-installation helper files

Add the following information to the helper files created in Uploading OVO HTTPS policies and instrumentation files on page 27:

- Installation helper file

```
rem set variable
ovconfchg -ns eaagt -set AD_READING_XPLCONF.CriticalThreshold
&(APPINFO.AD_CRIT)
ovconfchg -ns eaagt -set AD_READING_XPLCONF.MajorThreshold    80
ovconfchg -ns eaagt -set AD_READING_XPLCONF.WarningThreshold  70

rem install instrumentation
copy c:\ovo_https_policies_instr_win\instrumentation "C:\Program
Files\HP OpenView\data\bin\instrumentation"

rem install policy
ovpolicy -install -dir \ovo_https_policies_instr_win -add-category
My_OVO_RADIA_group
```

The variable `&(APPINFO.AD_CRIT)` will be replaced with a value you specify before deployment.

- De-installation helper file

```
rem remove policy
ovpolicy -remove -category My_OVO_RADIA_group

rem clean variables
ovconfchg -ns eaagt -clear AD_READING_XPLCONF.WarningThreshold
ovconfchg -ns eaagt -clear AD_READING_XPLCONF.MajorThreshold
ovconfchg -ns eaagt -clear AD_READING_XPLCONF.CriticalThreshold
```

Do not add any commands to remove the instrumentation files from the managed nodes because the files may be shared with other HP OpenView applications. The OVO agent software de-installation automatically removes all instrumentation files.

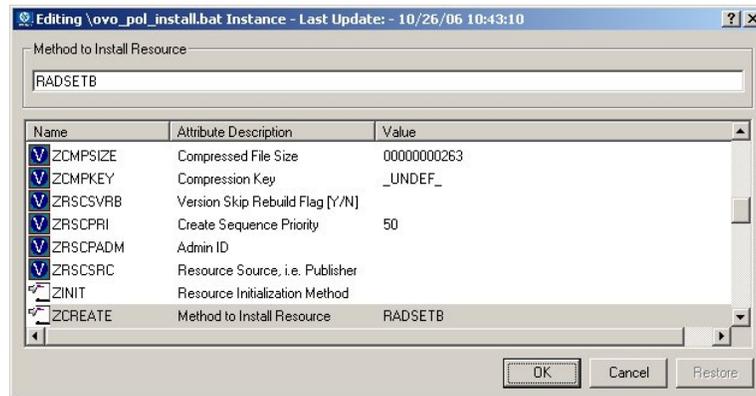
## Update Radia with the parameterized policy

1. Upload the clone image with the parameterized policy into Radia Packager, and create an application and service for it in Radia System Explorer as described in Deploying OVO HTTPS policies and instrumentation using Radia on page 26.
2. In Radia System Explorer, specify RADSETB for the ZCREATE attribute of the policy installation helper file.  
In the Database Tree View, double-click your parameterized policy package to display a list of all files that are included in the package. Then double-click the line containing the policy installation helper file (ovo\_pol\_install.bat) to display a list of its instance attributes.

The screenshot shows the Radia System Explorer interface. The Database Tree View on the left lists various policy and instrumentation files. The File Resources pane on the right displays the instance attributes for the selected file 'ovo\_pol\_install.bat'.

Name	Attribute Description	Value
ZRSCNAME	Resource Name	&ZRSCCFIL
ZRSCCFIL	Resource File Name	\ovo_pol_install.bat
ZRSCMO	Mandatory/Optional on Client [M/O]	O
ZRSCVRFY	Verify Resource File on Connect	Y
ZRSCCONF	Confirm File Download [Y/N]	Y
ZRSCRASH	DOS File Attribute [R/A/S/H]	A
ZRSCSTYP	Server File Type [BINARY/TEXT]	BINARY
ZRSCMEMM	PDS Member Name	
ZRSCDATE	Resource Date Stamp - From Promote	20061026
ZRSCTIME	Resource Time Stamp - From Promote	09:38:54
ZRSCSIZE	Resource Size - From Promote	00000000342
ZRSCVERS	Resource Version - From Promote	
ZRSCSTA	Client File Status	999
ZRSCCKPT	Resource Checkpoint	
ZRSCCRC	Resource CRC	
ZRSCRSTR	Restart [Y/N]	N
ZRSCOBJN	Persistent Object Name	
ZCMPSIZE	Compressed File Size	00000000263
ZCMPKEY	Compression Key	_UNDEF_
ZRSCSVRB	Version Skip Rebuild Flag [Y/N]	
ZRSCPRI	Create Sequence Priority	50
ZRSCPADM	Admin ID	
ZRSCSRC	Resource Source, i.e. Publisher	
ZINIT	Resource Initialization Method	
ZCREATE	Method to Install Resource	RADREMF
ZDELETE	Method to De-install Resource	
ZREPAIR	Client Instance Repair Method	
ZUPDATE	Client Instance Update Method	
ZFILEUPD	Client File Update/Add Method	
ZOPENERR	Client Method on File Open Error	RADLKM
ZC_ALWAYS_	Connect To	
ZC_ALWAYS_	Connect To	

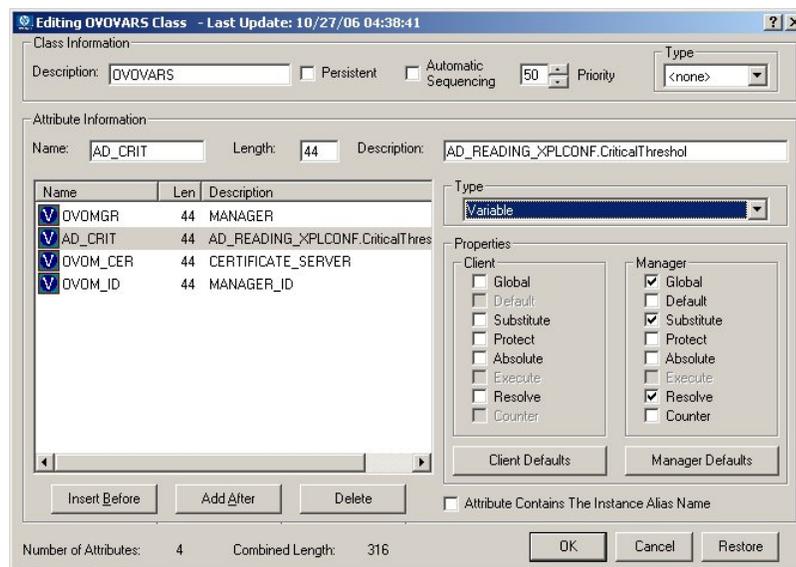
- Double-click the **ZCREATE** attribute and type **RADSETB** into the **Method to Install Resource** field. This ensures that Radia will be able to replace the variable in the `ovo_pol_install.bat` file during deployment. Click **OK**.



### Edit the helper class for variable replacement

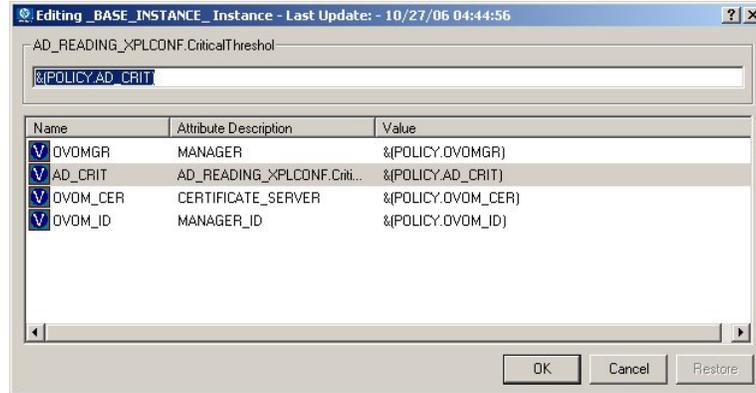
- In the Database Tree View, click **PRIMARY > POLICY**. Then right-click the policy you created for the OVO HTTPS agent software package in the section Edit the helper class for variable replacement on page 16. Click **Edit Class...** in the shortcut menu.
- An editor for the class opens:
  - Add the variable **AD\_CRIT** as follows:

Name	Length	Description
AD_CRIT	44	AD_READING_XPLCONF.CriticalThreshold
  - Select **Resolve** in the manager properties section of the window
  - Click **OK** to save your class and close the window.
  - When prompted, confirm that you want to save your class.



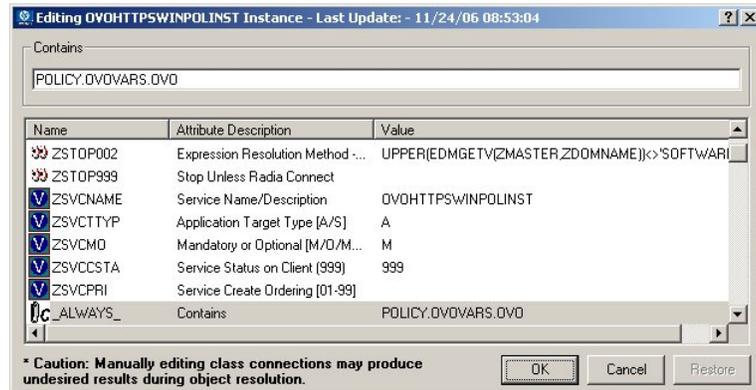
3. Double-click your class in the Database Tree View to expand it. Then right-click **\_BASE\_INSTANCE** and click **Edit Instance...** in the shortcut menu. An editor for this instance opens.
4. Specify a value for the AD\_CRIT attribute, then click **OK**:

Name	Attribute Description	Value
AD_CRIT	AD_READING_XPLCONF.CriticalThreshold	&(POLICY.AD_CRIT)



### Link the helper instance to the service

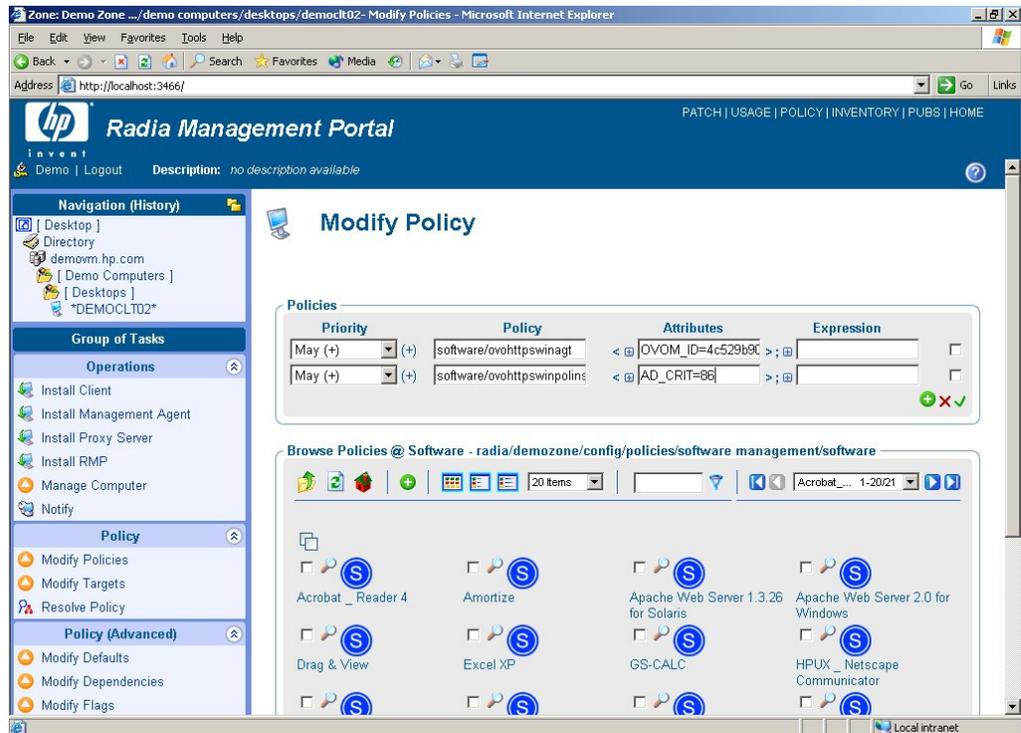
1. In the Database Tree View, click **PRIMARY > SOFTWARE > Application (ZSERVICE)**, then double-click the parameterized policy package to expand it.
2. Double-click the attribute **\_ALWAYS\_** and type `POLICY.<class_name>.<instance_name>` into the **Contains** field, then click **OK**.



## Deploy the parameterized threshold monitor policy

Use Radia Management Portal to deploy the parameterized threshold monitor policy to the target devices. Deploying the parameterized policy is similar to deploying the OVO HTTPS policies and instrumentation files which is described on page 30.

Before committing your changes to the Radia database, specify a value for the AD\_CRIT attribute, for example: AD\_CRIT=86.



After the package has been deployed, verify that the threshold monitor values are set correctly on the managed node:

1. On the managed node, type `ovpolicy -l -level 4`.  
This produces a list of all policies that are installed on the system.
2. On the managed node, type `ovconfget eaagt`.  
This produces a list of all configuration variables that are set in the `eaagt` name space.
3. Send an `opcmon` message to test the configured threshold and view the message in the OVO message browser.

## Example: parameterized threshold monitor policy

The following code shows a parameterized threshold monitor policy. The embedded script is a Perl script, marked in blue color.

```
SYNTAX_VERSION 10
```

```
ADVMONITOR "AD_READING_XPLCONF"
  DESCRIPTION "<initial policy version>"
  SCRIPTTYPE "Perl"
  INSTANCEMODE ONCE
  MAXTHRESHOLD
  ICASE
  EXTERNAL "SRC"
  DESCRIPTION ""
  MSGCONDITIONS
  DESCRIPTION "Critical message"
  CONDITION_ID "ealaff0b-dc59-498a-8a"
  CONDITION
  THRESHOLD
  SCRIPT "#PARAMETERS START
#PARAMETER AD_READING_XPLCONF.CriticalThreshold INT DEFAULT 95
NODEINFO
my $CriticalThreshold;
$CriticalThreshold = $OVOSystem-
>ParameterGetNum(\"AD_READING_XPLCONF.CriticalThreshold\",95),
b8"

#PARAMETER AD_READING_XPLCONF.MajorThreshold INT DEFAULT 85 VALUE 85
NODEINFO
my $MajorThreshold;
$MajorThreshold = $OVOSystem-
>ParameterGetNum(\"AD_READING_XPLCONF.MajorThreshold\",85);

#PARAMETER AD_READING_XPLCONF.WarningThreshold INT DEFAULT 75 VALUE 75
NODEINFO
my $WarningThreshold;
$WarningThreshold = $OVOSystem-
>ParameterGetNum(\"AD_READING_XPLCONF.WarningThreshold\",75);

#PARAMETERS END

$Session->Value('CriticalThreshold', $CriticalThreshold);
$Session->Value('MajorThreshold', $MajorThreshold);
$Session->Value('WarningThreshold', $WarningThreshold);

my $src;

$src = $Policy->Source(\"SRC\");
if ( $src->Value() > $CriticalThreshold )
{
  $Rule->Status(TRUE);
}
"

SETSTART
SEVERITY Critical
TEXT "Threshold <$SESSION(CriticalThreshold)> reached:
<$MSG_TEXT>"
```

Embedded Perl script.

Parameters are read from ovconfget eaagt.

Parameters are exported to the entire session so that they are available to all conditions.

```

        DESCRIPTION "Major message"
        CONDITION_ID "20d227e0-38b6-423c-a491-ad61317a699d"
        CONDITION
            THRESHOLD
            SCRIPT "
my $src;

$src = $Policy->Source(\"SRC\");
if ( $src->Value() > $Session->Value('MajorThreshold') )
{
    $Rule->Status(TRUE);
}
"
        SETSTART
        SEVERITY Major
        TEXT "Threshold <$SESSION(MajorThreshold)> reached:
<$MSG_TEXT>"

        DESCRIPTION "Warning message"
        CONDITION_ID "4bcfafc7-c205-41f2-8dec-a9f3986c48be"
        CONDITION
            THRESHOLD
            SCRIPT "
my $src;

$src = $Policy->Source(\"SRC\");
if ( $src->Value() > $Session->Value('WarningThreshold') )
{
    $Rule->Status(TRUE);
}
"
        SETSTART
        SEVERITY Warning
        TEXT "Threshold <$SESSION(WarningThreshold)> reached:
<$MSG_TEXT>"

```

## Glossary

- **Agent (OVO):**  
Program that receives requests from a manager program, and can gather information, perform processing, and generate responses.
- **Class (Radia):**  
A logical partition in the Radia database that groups similar instances. It is the third level in the hierarchical structure of the database.
- **Class instance (Radia):**  
A specific occurrence of a class. Each instance of a particular class inherits the attributes defined for that class.
- **Client computer (Radia):**  
A computer that has the Radia client installed and where the OVO agent software will be installed. It may also be referred to as a target device.
- **Clone directory (OVO):**  
Directory on the target device that contains the clone image.
- **Clone image (OVO):**  
A set of files that represent an identical copy of the software and configuration of a typical OVO agent.
- **Database (Radia):**  
An object-oriented database that stores all the information needed to manage assets on a device, including the software and/or data that Radia distributes, the policies determining which subscribers or users are entitled to which packages, and security and access rules for administrators. It has a hierarchical structure containing four levels: files, domains, classes, and instances.
- **Managed node (OVO):**  
Computer system or intelligent device (for example, a network printer or router) monitored or controlled by OVO. The OVO agent collects, filters, and processes information from each node, and sends it to the management server.
- **Management server (OVO):**  
Central computer system of the domain to which all managed nodes forward their OVO messages.
- **Notify (Radia):**  
A message sent to the managed device that tells the device to start a client connect.
- **Package (Radia):**  
(n) A unit of software or data that can be published to the Radia database.  
(v) The process of grouping data into a unit that can be published to the Radia database.
- **Policy (OVO):**  
A set of one or more specifications rules and other information that help automate network, system, service, and process management. Policies can be deployed to various targets (for example, managed systems, devices, network interfaces) providing consistent, automated administration across the network.
- **Policy (Radia):**  
A designation of the services to which a subscriber, a client computer, or a managed device is entitled.

- **Service (Radiq):**  
A service organizes a group of related packages (applications), methods, or behaviors into manageable units.
- **Target device (Radiq):**  
A workstation or server on which you want to install, replace, or update software.

## For more information

- HP OpenView Management Software  
<http://www.hp.com/go/managementsoftware>
- HP OpenView Product Manuals  
[http://ovweb.external.hp.com/lpe/doc\\_serv/](http://ovweb.external.hp.com/lpe/doc_serv/)

This paper was written by Hans-Peter Schmollinger, Hewlett-Packard Company.

## Call to action

To help us better understand and meet your needs for technical information about HP OpenView, submit a support case or enhancement request using HP Software Support Online at <http://support.openview.hp.com/support.jsp>.

© 2006 Hewlett-Packard Development Company, L.P. The information contained herein is subject to change without notice. The only warranties for HP products and services are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. HP shall not be liable for technical or editorial errors or omissions contained herein.

HP-UX Release 10.20 and later and HP-UX Release 11.00 and later (in both 32 and 64-bit configurations) on all HP 9000 computers are Open Group UNIX 95 branded products.

Microsoft® and Windows® are U.S. registered trademarks of Microsoft Corporation.

UNIX® is a registered trademark of The Open Group.

PDF only, 12/2006

