

# HP OpenView Server Management Using Radia

for the Windows and Linux operating systems

Software Version: 1.0

---

## Configuration Baseline Auditor Guide

Manufacturing Part Number: T3424-90087

June 2005



## Legal Notices

### Warranty

Hewlett-Packard makes no warranty of any kind with regard to this document, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. Hewlett-Packard shall not be held liable for errors contained herein or direct, indirect, special, incidental or consequential damages in connection with the furnishing, performance, or use of this material.

A copy of the specific warranty terms applicable to your Hewlett-Packard product can be obtained from your local Sales and Service Office.

### Restricted Rights Legend

Use, duplication, or disclosure by the U.S. Government is subject to restrictions as set forth in subparagraph (c)(1)(ii) of the Rights in Technical Data and Computer Software clause in DFARS 252.227-7013.

Hewlett-Packard Company  
United States of America

Rights for non-DOD U.S. Government Departments and Agencies are as set forth in FAR 52.227-19(c)(1,2).

### Copyright Notices

© Copyright 2005 Hewlett-Packard Development Company, L.P.

No part of this document may be copied, reproduced, or translated into another language without the prior written consent of Hewlett-Packard Company. The information contained in this material is subject to change without notice.

### Trademark Notices

Linux is a registered trademark of Linus Torvalds.

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

OpenLDAP is a registered trademark of the OpenLDAP Foundation.

### Acknowledgements

PREBOOT EXECUTION ENVIRONMENT (PXE) SERVER  
Copyright © 1996-1999 Intel Corporation.

TFTP SERVER

Copyright © 1983, 1993

The Regents of the University of California.

OpenLDAP

Copyright 1999-2001 The OpenLDAP Foundation, Redwood City, California, USA.

Portions Copyright © 1992-1996 Regents of the University of Michigan.

OpenSSL License

Copyright © 1998-2001 The OpenSSLProject.

Original SSLeay License

Copyright © 1995-1998 Eric Young (eay@cryptsoft.com)

DHTML Calendar

Copyright Mihai Bazon, 2002, 2003

## Support

Please visit the HP OpenView support web site at:

**<http://www.hp.com/managementsoftware/support>**

This web site provides contact information and details about the products, services, and support that HP OpenView offers.

HP OpenView online software support provides customer self-solve capabilities. It provides a fast and efficient way to access interactive technical support tools needed to manage your business. As a valuable support customer, you can benefit by using the support site to:

- Search for knowledge documents of interest
- Submit enhancement requests online
- Download software patches
- Submit and track progress on support cases
- Manage a support contract
- Look up HP support contacts
- Review information about available services
- Enter discussions with other software customers
- Research and register for software training

Most of the support areas require that you register as an HP Passport user and log in. Many also require a support contract.

To find more information about access levels, go to:

**[http://www.hp.com/managementsoftware/access\\_level](http://www.hp.com/managementsoftware/access_level)**

To register for an HP Passport ID, go to:

**<http://www.managementsoftware.hp.com/passport-registration.html>**

# Contents

<b>1</b>	<b>Introduction .....</b>	<b>7</b>
	How the Configuration Baseline Auditor Works .....	10
	Chapters Summary .....	12
<b>2</b>	<b>Configuring Your Environment.....</b>	<b>13</b>
	Importing the BASELINE domain .....	13
	Adding Drag and Drop Support for the Baseline Domain .....	14
	Specifying the Data Collection Directory .....	16
	Modifying Configuration Baseline Importer values after installation .....	37
	Changing your Configuration Baseline Auditor Database password .....	39
<b>3</b>	<b>Defining Subsystems .....</b>	<b>49</b>
	Example: Creating and Configuring a Subsystem .....	50
	Creating a Subsystem.....	50
<b>4</b>	<b>Viewing the Results .....</b>	<b>63</b>
	About the Configuration Baseline Auditor .....	64
	Using the Configuration Baseline Auditor Interface .....	67
	Generating Configuration Baseline Auditor Reports.....	74
<b>A</b>	<b>About the Baseline Domain.....</b>	<b>79</b>
	Baseline Scanner.....	80
	Client Methods (CMethod) .....	80
	Collection Options (BASECOLL) .....	80
	Device Filters (DEVFLTR) .....	81
	File Filters (FILEFLTR).....	81
	File Root (FILEROOT).....	84

Node Filters (NODEFLTR) .....	84
Registry Filters (REGFLTR) .....	84
Subsystem Components (PACKAGE).....	86
Subsystems (ZSERVICE) .....	87
<b>B Troubleshooting .....</b>	<b>89</b>
Logging .....	89
Increased Logging Details in Radia's method.log.....	90
<b>Glossary .....</b>	<b>93</b>
<b>Index.....</b>	<b>95</b>

# 1 Introduction

## What is the Configuration Baseline Auditor?

The Configuration Baseline Auditor (CBA) is a component of the Radia Server Management suite that allows you to:

- Define a logical group of resources (files and registry settings), known as subsystems, for which you define scan filters to determine what information will be captured.
  - ▶ The intent of the Configuration Baseline Auditor is not to capture all components of an entire system as a subsystem, but to define a logical group of resources, which is typically an application.
- Collect configuration information about the subsystems, and capture its state at a specific point in time.
- View configuration changes.

Typical uses of the CBA are to review changes to a subsystem over time, to review differences between subsystems that should be configured identically, or to review differences between a subsystem and a well-defined baseline configuration.

When defining a subsystem, you define collection options and filters used to scan the resources on the device. As you collect information about configuration changes on your device, a state is created within the subsystem. Then, you use the Configuration Baseline Auditor to compare different states within a subsystem.

Benefits of the Configuration Baseline Auditor are that you can:

- Collect detailed file and registry value information.
- Collect the subsystem state at regular intervals, such as daily or weekly.
- Transfer information about configuration changes only, which reduces data transfer and database import overhead.
- Easily define the resources that belong to specific subsystems.

This guide describes how to install the Configuration Baseline Auditor and its components, how to define subsystems and how to compare states in a subsystem.



To use the Configuration Baseline Auditor, you must own a Server LTU for HP OpenView using Radia.

## Audience

This document is for system administrators who are proficient in working with the Radia environment, including but not limited to, the System Explorer, Radia Client, and Radia Integration Server.

## Supported Platforms

Collection of information is supported on the following platforms:

- Windows 2000 and 2003 Server
- Linux Red Hat Enterprise 3 and 4
- SUSE Linux Versions 8 and 9

The Configuration Baseline Importer and Configuration Baseline Auditor are supported on Windows 2000 and above.

## Software Requirements

- HP OpenView Common Components
- Microsoft SQL Server 2000 SP3 or later
- Radia Infrastructure
  - Configuration Server to store subsystems and deployment policies
  - Radia Integration Server
  - Administrator Workstation





The System Explorer must be Version 4.x Build 36 or later in order to have all of the features that you will need to perform the tasks described in this document.

- Application Manager or Software Manager version 4.0 SP1 or above
- Radia Reporting Server 4.1.1 (optional)

# How the Configuration Baseline Auditor Works

Configuration Baseline Auditor processes can be divided into three categories.

## 1 Defining Subsystems

The Configuration Server contains a BASELINE domain used to define subsystems. It also contains a scanning script, known as the Configuration Baseline Scanner.

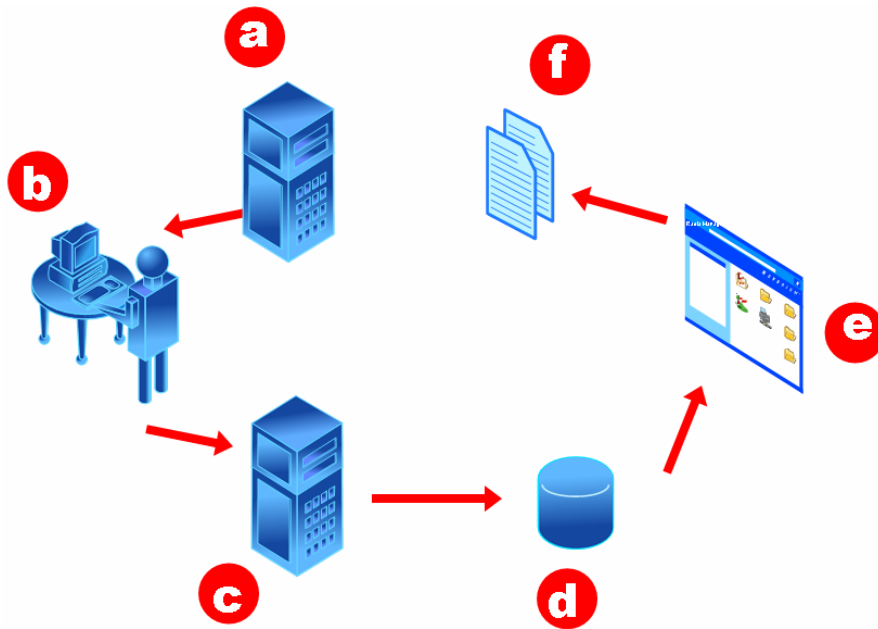
## 2 Collecting and Importing Data

A Radia client connect or timer initiates the Configuration Baseline Scanner. The scanner scans the device and creates a set of files (compressed XML files) that are sent to the Radia Integration Server. The Radia Integration Server stores the XML files from the device in a directory that you specify during the installation. The Configuration Baseline Importer component reads the XML files and imports them into a Microsoft SQL Server database.

## 3 Reporting

Use the Configuration Baseline Auditor or the Radia Reporting Server to view the results of the collection. These reports can be exported into multiple formats.

Figure 1 on page 11 is a diagram of the process flow.



**Figure 1: Process flow**

- a The Configuration Server contains the BASELINE domain. Use this domain to define subsystems, configure your filters, and specify collection options.
- b A Radia client connect or timer initiates the Configuration Baseline Scanner which is a script stored in the BASELINE domain.
- c When the scan is complete, XML files containing the collected subsystem information are created and stored in a pre-determined directory (typically the HP OpenView data directory).
- d The Configuration Baseline Importer retrieves the XML files from the pre-determined directory and imports them into the Microsoft SQL Server database.
- e Use the Configuration Baseline Auditor or Radia Reporting Server to view and compare subsystem states.
- f You can also use the Configuration Baseline Auditor or Radia Reporting Server to export reports of the comparisons into various formats.

# Chapters Summary

## [Chapter 1, Introduction](#)

This chapter describes the Configuration Baseline Auditor and how it works. It also provides information about the supported platforms and hardware and software requirements.

## [Chapter 2, Configuring Your Environment](#)

This chapter provides installation and configuration instructions.

## [Chapter 3, Defining Subsystems](#)

This chapter provides information about how to use the System Explorer to define subsystems.

## [Chapter 4, Viewing the Results](#)

This chapter describes how to use the Configuration Baseline Auditor or Radia Reporting Server to compare states within subsystems

## [Appendix A, About the Baseline Domain](#)

This appendix provides detailed information about the classes and instances in the Baseline Domain.

## [Appendix B, Troubleshooting](#)

This appendix describes some of the initial steps to take to determine where a problem has occurred.

# Related Documents

System Explorer Guide for the HP OpenView Administrator Workstation Using Radia (System Explorer Guide)

Installation and Configuration Guide for the HP OpenView Reporting Server Using Radia (Reporting Server Guide)

## 2 Configuring Your Environment

Before you can use the Configuration Baseline Auditor, you must prepare your environment. To configure the components to use the Configuration Baseline Auditor you will need to complete the following tasks:

- 1 Install and configure the Radia Environment.
  - Radia Integration Server
  - Configuration Server and Database
  - Radia Administrator Tools
  - Radia client
  - Import the BASELINE domain to your existing Radia Database.
  - Add Drag and Drop Support for the Baseline Domain
  - Make configuration changes to your existing Radia Integration Server.
- 2 Set up a Microsoft SQL Server Database
- 3 Install the Configuration Baseline Importer.
- 4 Install the Configuration Baseline Auditor.
- 5 Configure the Radia Reporting Server (optional)

The following sections describe the steps needed to configure your environment.

### Task 1 [Configure the Radia Environment](#)

#### Importing the BASELINE domain

You must import the BASELINE domain to the Radia Database. This domain contains:

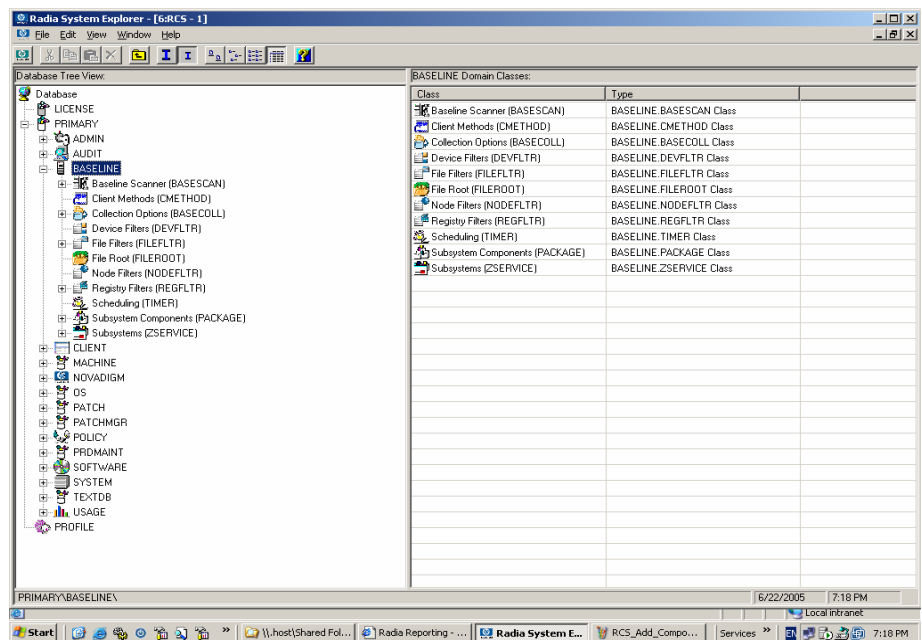
- Definitions for subsystems which have filters based on file system and/or Windows registry.
- Collection options attached to the subsystem.
- A timer for scheduling client connections.

## To update the Radia Database

- 1 Stop the **Configuration Server** service.
- 2 From the **Server Management CD**, go to **\Configuration Baseline\Radia Configuration Server Database** and copy the files to your Configuration Server's bin folder.
- 3 If importing to a Windows hosted Configuration Server, run `import_baseline.cmd`. If importing to a UNIX-hosted Configuration Server, run `import_baseline.sh`.

A return code of 0 indicates that there were no errors reported during the import, and the updates are applied to the Radia Database. You may see a return code of 4 as well, which you can ignore.

- 4 Start the **Configuration Server** service.
- 5 Use the System Explorer to view the new **BASELINE** domain.

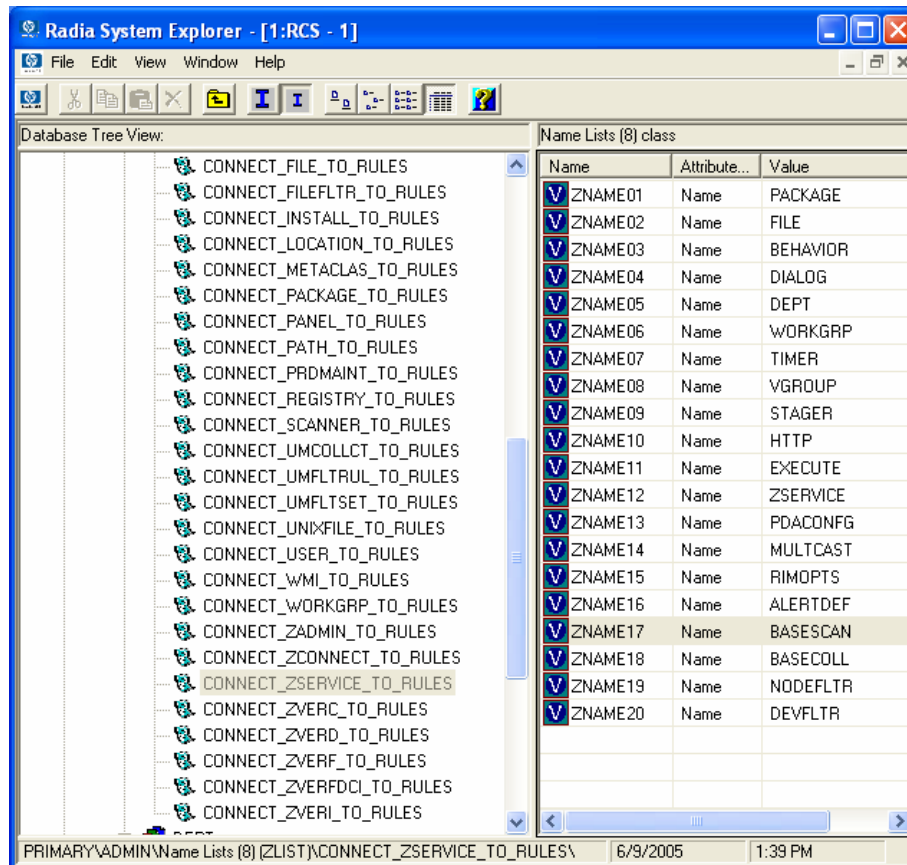


## Adding Drag and Drop Support for the Baseline Domain

The System Explorer must be modified to support drag and drop rules for the new BASELINE domain classes.

To add drag and drop support

- 1 Open the System Explorer.
- 2 Go to **PRIMARY.ADMIN.ZLIST**.
- 3 Add a new **ZLIST** instance called **CONNECT\_FILEFLTR\_TO\_RULES** and set **ZNAME01** to **FILEROOT**.
- 4 Go to the **ZLIST** instance called **CONNECT\_ZSERVICE\_TO\_RULES** and add the following values to the **ZNAME** attributes:
  - **BASESCAN**
  - **BASECOLL**
  - **NODEFLTR**
  - **DEVFLTR**



## Specifying the Data Collection Directory

You must specify the directory that will store the zipped XML files sent from the Radia client.

- ▶ This directory *must* match the directory where the Configuration Baseline Importer will look for data to import into the SQL Server Database. This directory is set during the installation of the Configuration Baseline Importer. See *Installing the Configuration Baseline Importer* on page 24 for details.

To specify the data collection directory to store the XML files

- 1 Stop the Radia Integration Server service.
- 2 From the Server Management CD, open `Configuration Baseline\Importer\putcfg.txt`.
- 3 Copy the text in this file to the clipboard.
- 4 Open `SystemDrive:\Novadigm\IntegrationServer\etc\put.cfg`.
- 5 Paste the text that you copied before the Usage Manager section. It should read as follows.

```
# -----  
# - Configuration Baseline Auditor Destination Directories  
# -----  
  
if {[info exists env(OvDataDir)]} {  
    file mkdir [set dir [file join $env(OvDataDir)  
datafiles/cba/import]]  
    Put_AddRoot /CBA_Import $dir  
} else {  
    syslog warning "Prerequisite installation of the HP  
OpenView Cross Platform Library has not been performed.  
Configuration baseline auditing is unable to receive baseline  
related data files from client systems."  
}
```

- The first statement checks to see if the HP OpenView common components have been installed. The default location is `ProgramFilesFolder\HP OpenView\Data`.



- If the HP OpenView common components have been installed, the second statement creates a directory in ProgramFilesFolder\HP OpenView\Data called datafiles\cba\import.
- The third statement specifies where to save data “put” to the CBA\_Import http target location as specified in the collection options (BASELINE.BASECOLL class defined in the Radia Database).
- The final statement adds an information message to the Radia Integration Server’s log if the HP OpenView Foundation Components (specifically, the Cross Platform Library) are not installed or the environment variable is not found.

If you want to store the files in a different location, you can modify the directories specified above. However, if you modify the values in put.cfg you must change the default value for the directory where the Configuration Baseline Importer will read the files from as described in Installing the Configuration Baseline Importer on page 24.

Below is an example of using an absolute path.

```
# -----
# - Configuration Baseline Auditor Destination Directories
# -----
if {[info exists env(ovDataDir)]} {
    file mkdir [set dir "c:/program files/HP OpenView/data
/datafiles/cba/import"]
    Put_AddRoot /Cba_Import $dir
}
```

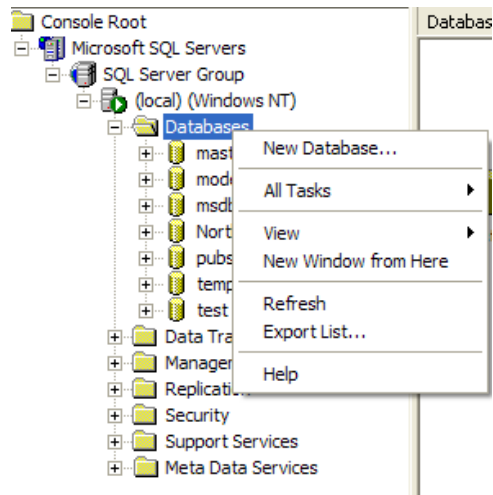
- 6 Save and close the file.
- 7 Start the Radia Integration Server service.

## Task 2 Setting up the Microsoft SQL Server Database

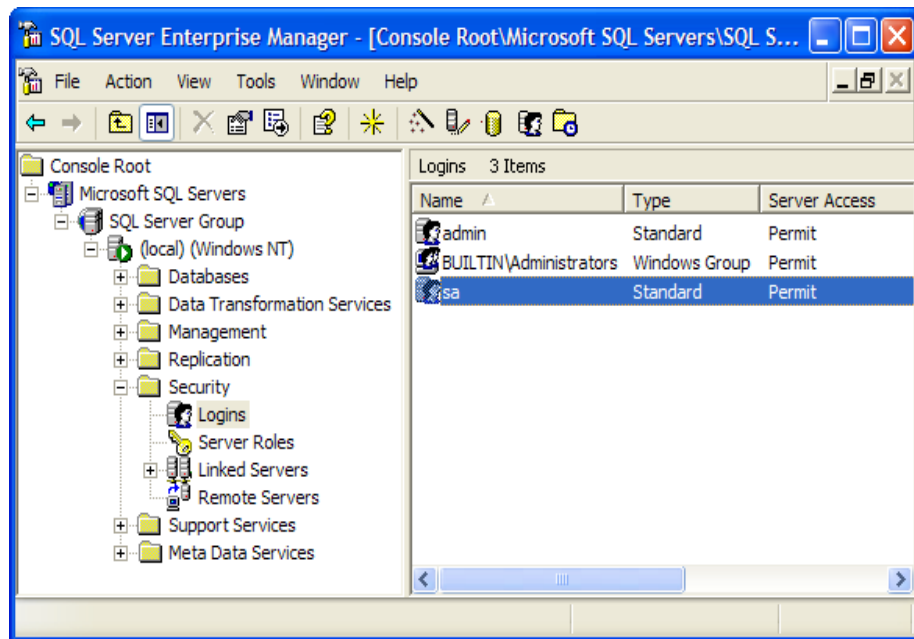
The Configuration Baseline Importer retrieves the XML files from the collection directory and imports them into the SQL Server database.

To set up your SQL Server database

- 1 Setup a SQL Server Database. To do this, go to **Start, Program Files, Microsoft SQL Server, Enterprise Manager**.
- 2 Expand the Console Root section until you see (local) (Windows NT).
- 3 Right-click **Databases** and select **New Database**.



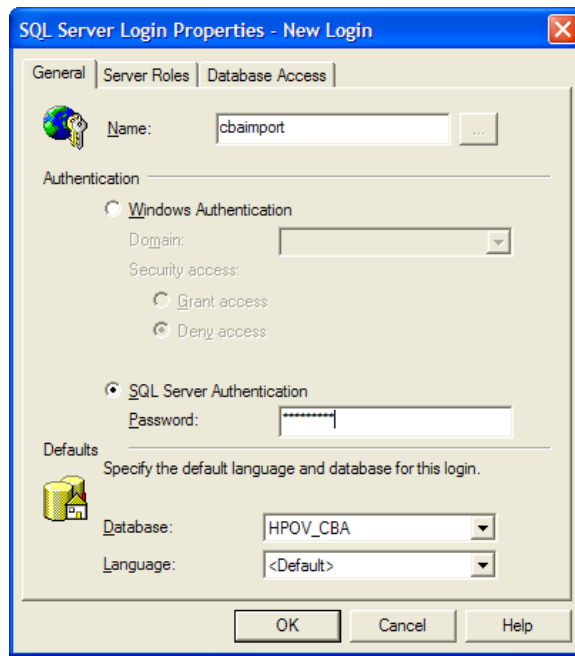
- 4 Type a name for the Database, such as **HPOV\_CBA**.
- 5 Click the **Data Files** tab and set the **Initial Size (MB)** to **100**.
- 6 In **File Growth** area, select **In Megabytes** and type **10** in the text box.
- 7 Click the **Transaction Log** tab and set the **Initial Size (MB)** to **20**.
- 8 In **File Growth** area, select **In Megabytes** and type **5** in the text box.
- 9 Click **OK**.
- 10 In the left pane, go to **Security** and expand it until you see **Logins**. Note that the Login “sa” is the default SQL server system administrator account.



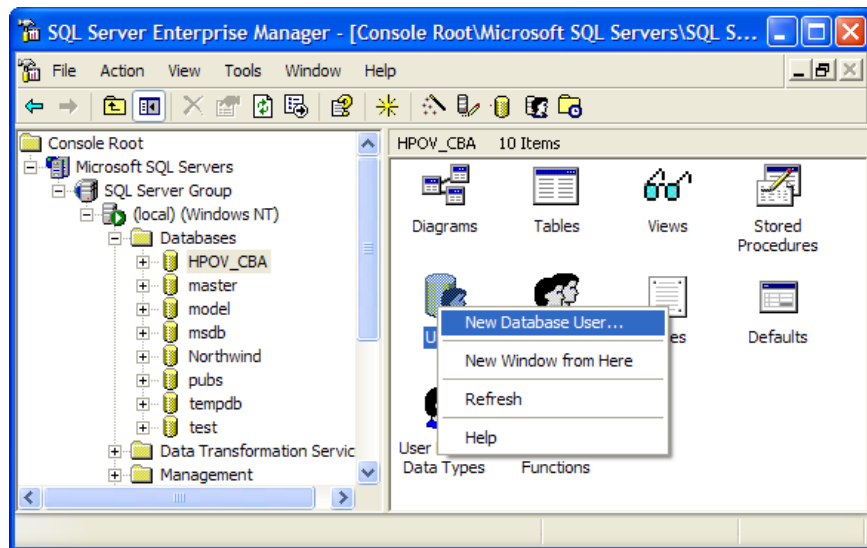
- 11 If you want to create a new login for the SQL Server, you can right-click **Login** and select **New Login**. If you would like to use Windows authentication, consult Microsoft's SQL Server documentation.
- 12 In the **Name** field, type a name such as **cbaimport**.
- 13 Select **SQL Server Authentication** and type a password.



Neither Windows Authentication, nor SQL Server Named Instances are supported in this version of Configuration Baseline Auditor. Choose SQL Server Authentication.

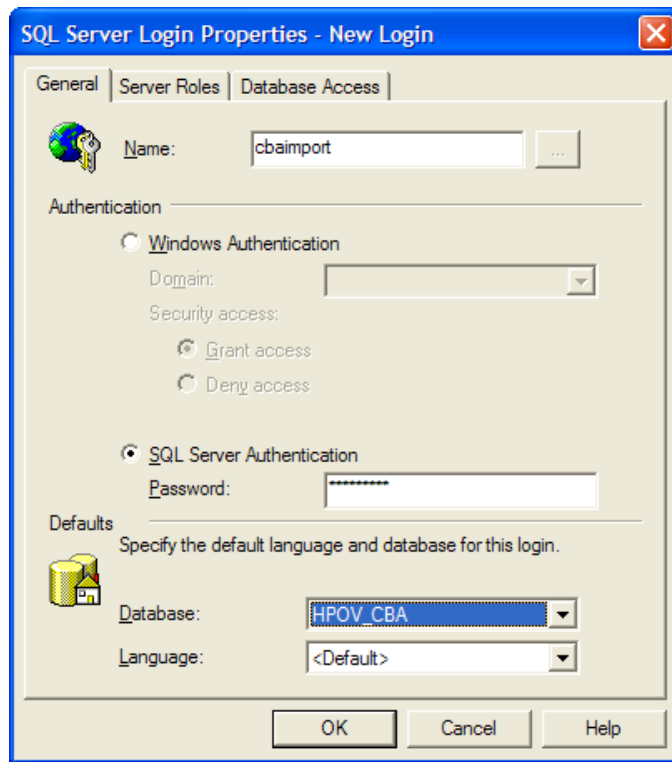


- 14 Click the **Database Access** tab. Select the database that you created and confirm that **public** and **db\_owner** are selected in the **Permit in Database Role** list box.
- 15 Click **OK**.
- 16 Type your password again and click **OK**.
- 17 In the left pane, under **Databases**, click the new database, **HPOV\_CBA**.
- 18 In the right pane, right-click **Users** and select **New Database User**.

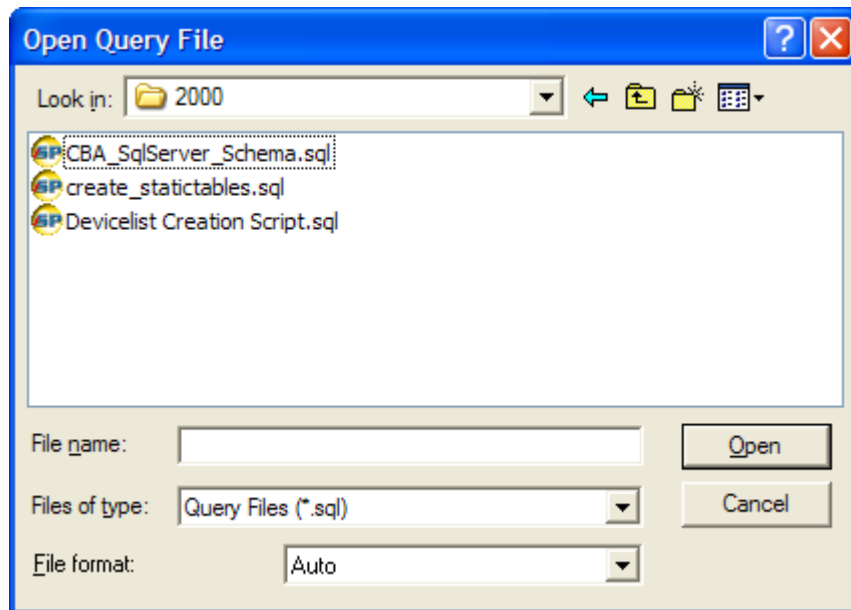


19 In the **Login Name** field, type a name such as **cbaimport**.

20 In the **Permit in Database Role** area, select **db\_owner**.



- 21 Click **OK**.
- 22 From the **Tools** menu, select **SQL Query Analyzer**.
- 23 From the **File** menu, select **Open**.
  - ▶ Make sure the **HPOV\_CBA** database is selected.
- 24 From the **Server Management CD**, go to Configuration Baseline\SQL Databases\SQL Server\2000 and select **CBA\_SqlServer\_Schema.sql**.



- 25 Click **Open**.
- 26 From the **Query** menu, select **Execute**. The status bar will tell you when the Query Batch is completed.
- 27 From the **Server Management CD**, go to Configuration Baseline\SQL Databases\SQL Server\2000 and select **create\_stautables.sql**.
- 28 Click **Open**.
- 29 From the **Query** menu, select **Execute**. The status bar will tell you when the Query Batch is completed.
- 30 From the **Server Management CD**, go to Configuration Baseline\SQL Databases\SQL Server\2000 and select **Device Creation Script.sql** which is used to enable the Reporting Server.
- 31 Click **Open**.
- 32 From the **Query** menu, select **Execute**. The status bar will tell you when the Query Batch is completed.
- 33 Close the **Analyzer** and **SQL Enterprise Manager**.

### Task 3 Installing the Configuration Baseline Importer

The Configuration Baseline Importer component reads the XML files and imports them into a SQL database.

To install the Configuration Baseline Importer

- 1 From the **Server Management CD**, go to `\Configuration Baseline\Importer` and double-click `setup.cmd`.

First, the HP OpenView common components will be installed. These components are a pre-requisite for the Configuration Baseline Importer and provide secure communication and control, configuration, and installation of other components. They also install a local java runtime that is necessary for the HP OpenView environment.

The **Welcome to the Installer for HP OpenView Cross Platform Component** window opens.

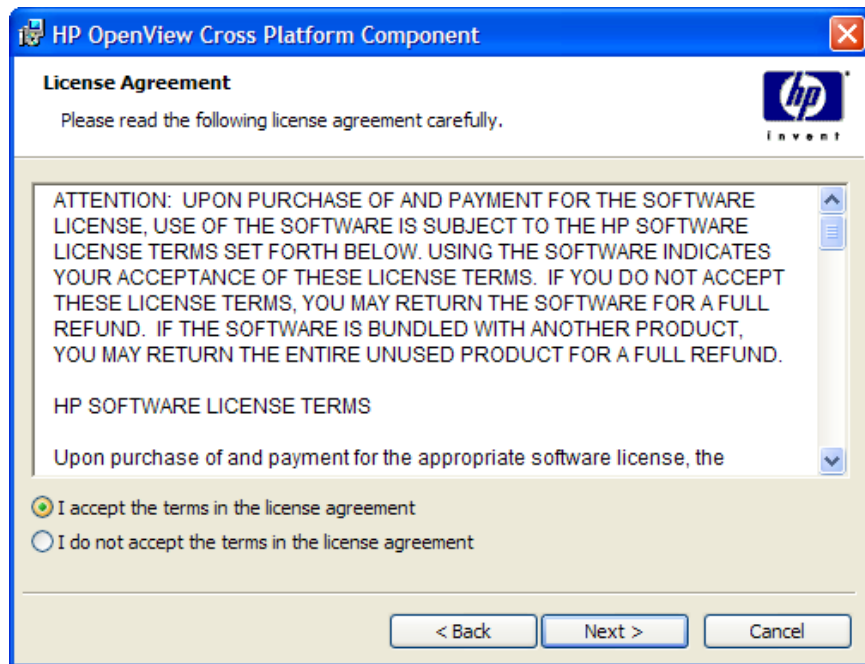


The interface for the HP OpenView Cross Platform Component window opens only if you have not previously installed the HP OpenView common components on the machine. If the components are already installed, you will see several progress bars and when they are done, skip to step 6.

- 2 Click **Next**.

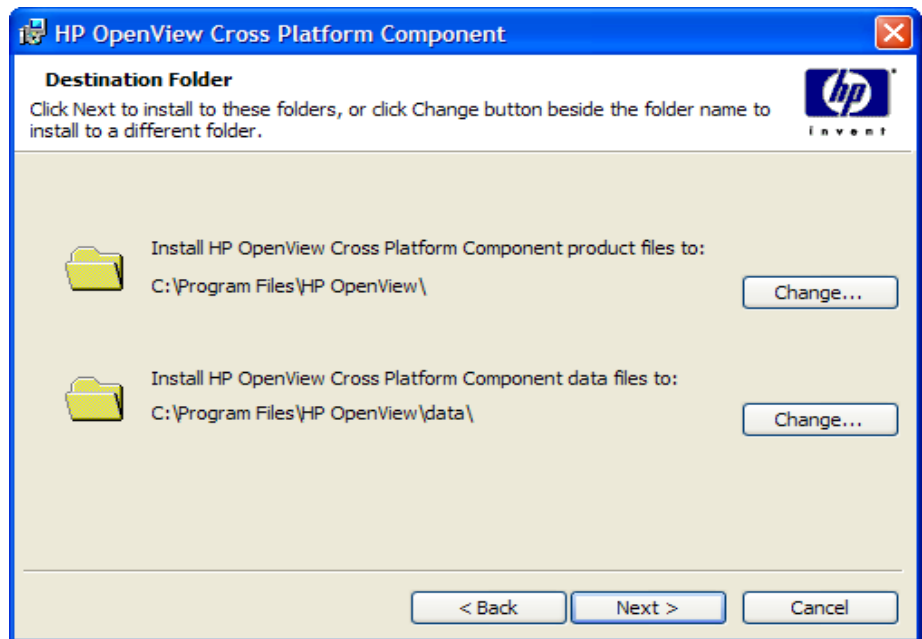
The **End User Licensing Agreement** window opens.





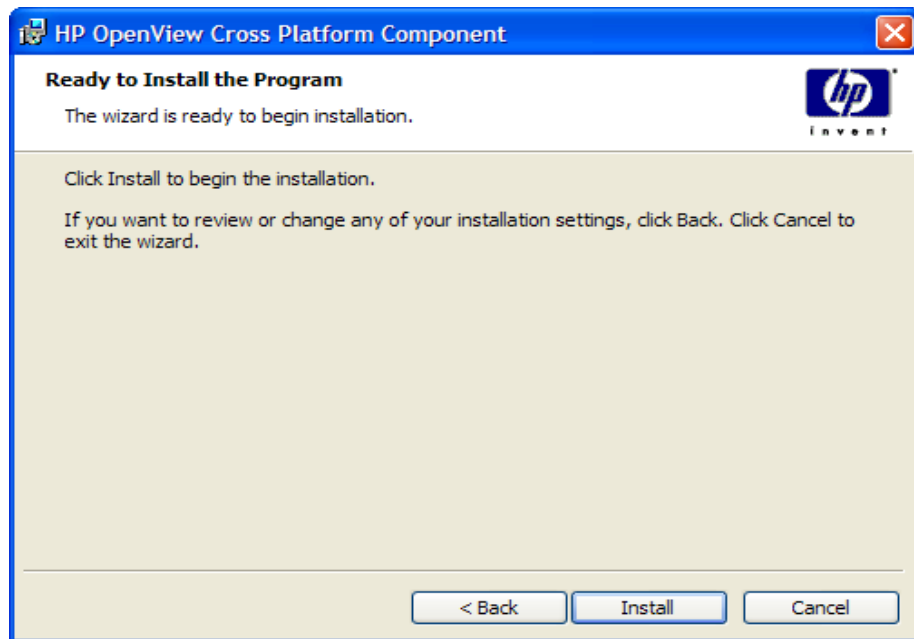
- 3 Select whether you accept the terms of the agreement or not and then click **Next**. If you do not accept the terms, the installation will be terminated.

The Destination folder window opens.



- 4 Accept the defaults or click **Change** to change the installation directories. Then, click **Next**.

The Ready to Install the Program window opens.



- 5 Click **Install**.
- 6 Click **Finish** when the HP OpenView Cross Platform component installation is complete. You will see several progress bars appear while the rest of the components are installed.

When they are done, the Welcome to the Configuration Baseline Importer Setup window opens.

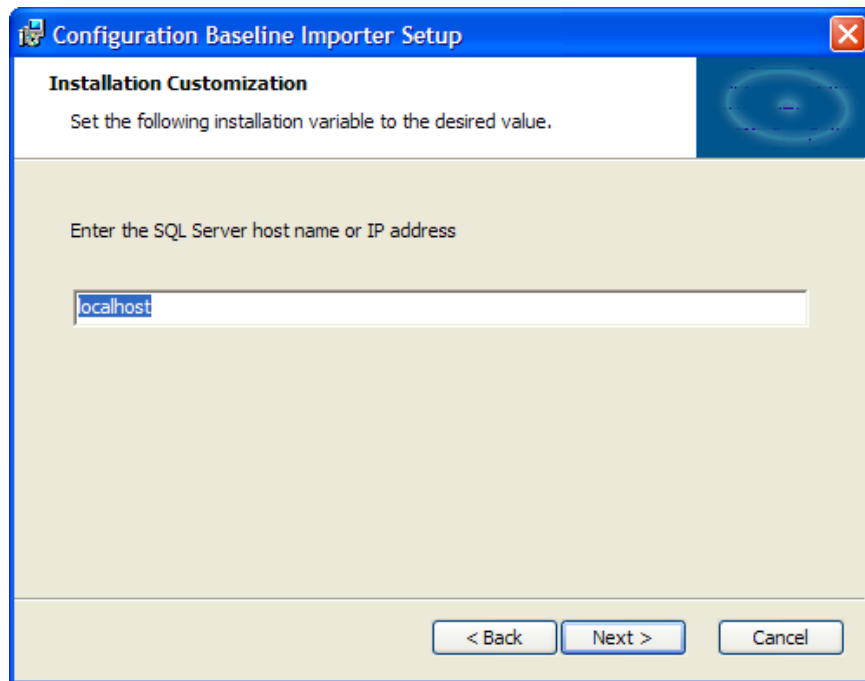


7 Click **Next**.

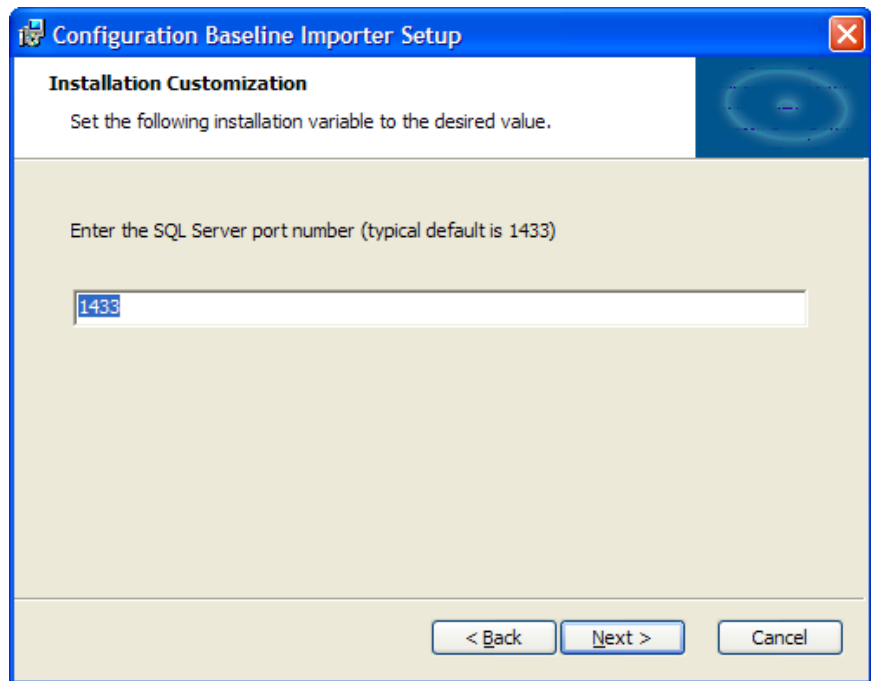
The End-User Licensing Agreement window opens.

8 Click **Next**.

The SQL Server host name or IP address window opens.

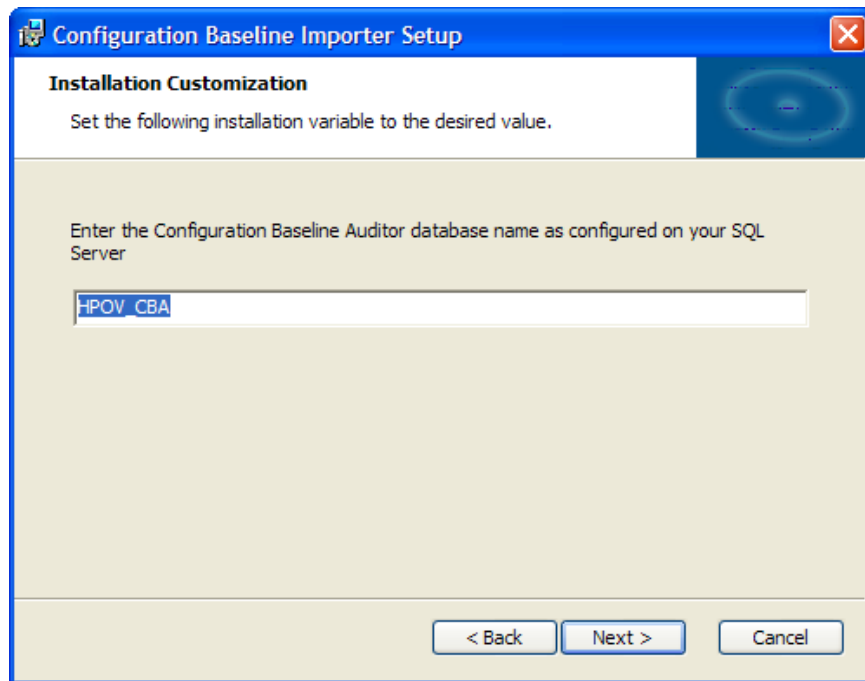


- 9 Type the hostname or IP address for your SQL Server and then click **Next**. The Enter the SQL Server port number window opens.



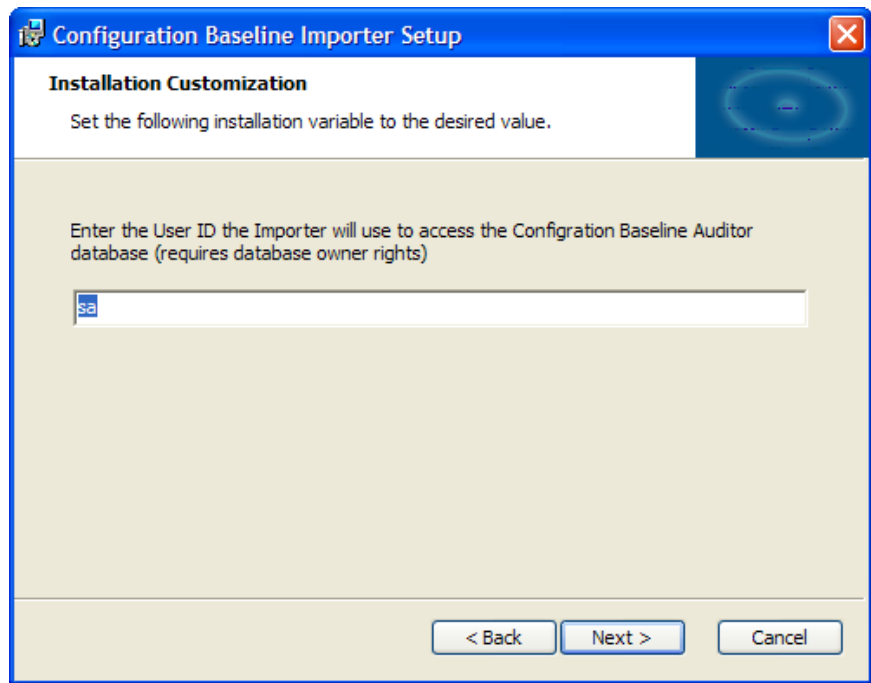
- 10 Type the port number for your SQL Server (typical default is 1433) and then click **Next**.

The Configuration Baseline Auditor Database name window opens.



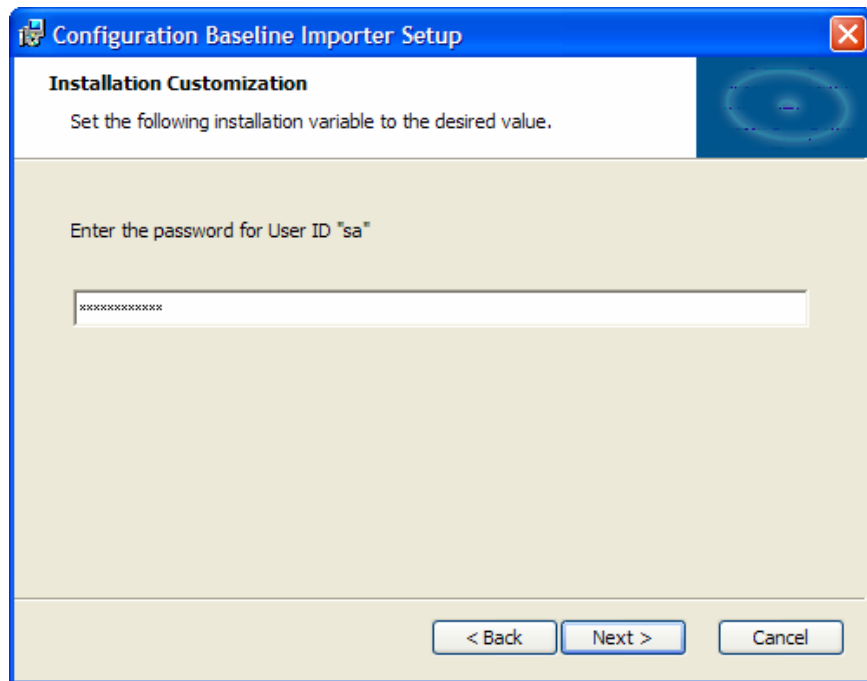
- 11 Type the name of your Configuration Baseline Auditor Database that you created on your SQL Server and then click **Next**.

The User ID window opens.

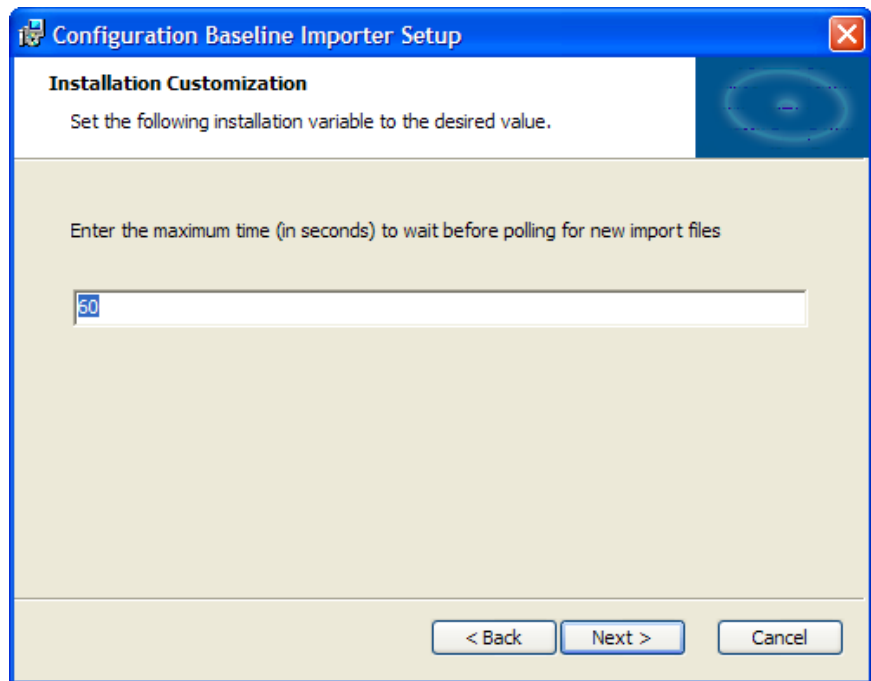


- 12 Type the **User ID** that you set up in your Configuration Baseline Auditor database that you created on your SQL Server and then click **Next**.  
Note that the User ID **sa** is the default SQL server system administrator account.  
The Password window opens.



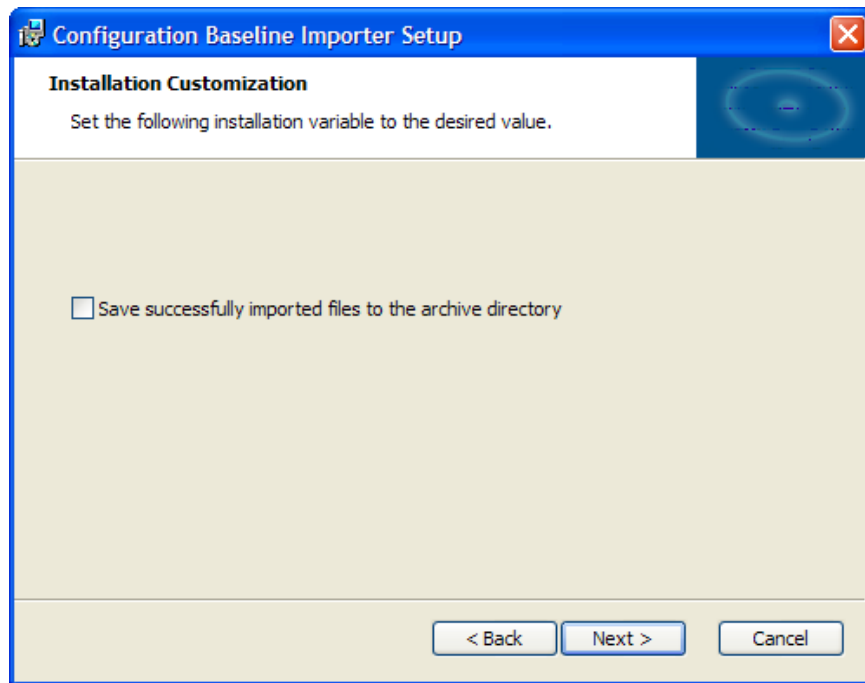


- 13 Type password that you set up in your Configuration Baseline Auditor database that you created on your SQL Server and then click **Next**.  
The Wait time window opens.



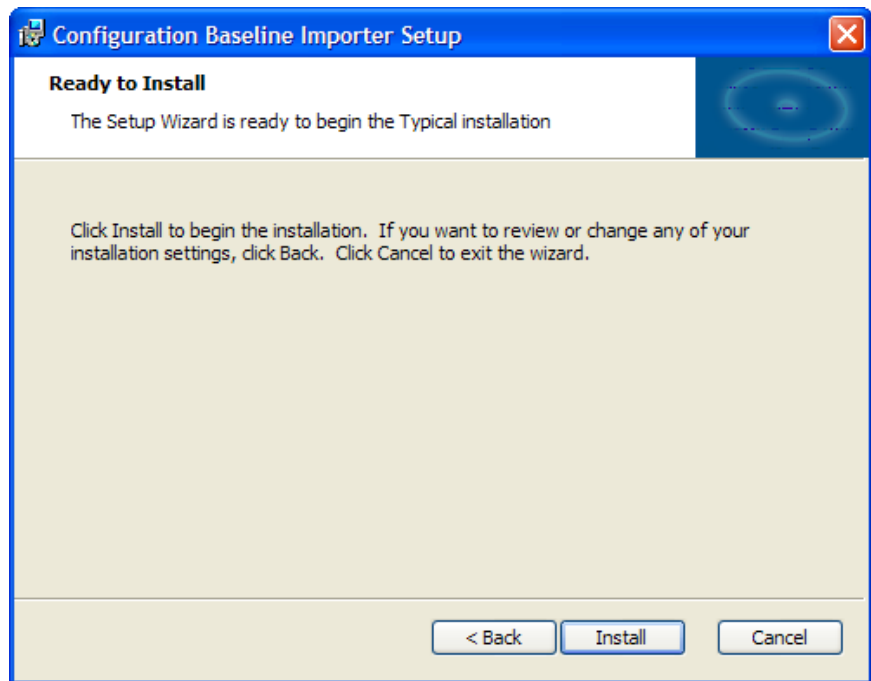
- 14 Type the maximum time, in seconds, to wait before polling the Radia Integration Server for new files to be imported to the SQL Server Database and then click **Next**.

The Archive option window opens.



- 15 Select whether you would like to have the compressed XML files that have been imported successfully stored in an archive directory. See Table 1 on page 37 for where these files are stored.

The Ready to Install window opens.

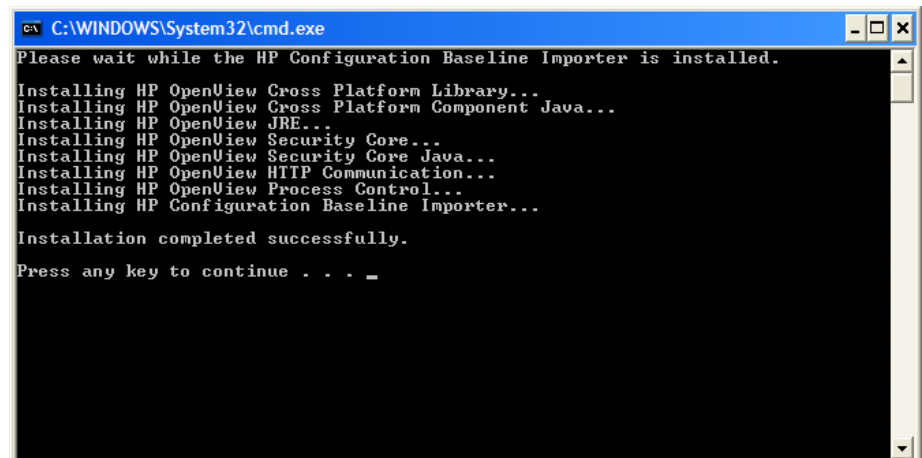


16 Click **Install**.

Several command windows will open and close.

17 Click **Finish** to complete the installation.

The final command window opens.



- 18 Press any key to continue. This completes the installation for the Configuration Baseline Importer.

## Modifying Configuration Baseline Importer values after installation

In some cases, you may need to modify the values that you set during the installation of the Configuration Baseline Importer.

▶ The defaults are listed in `path\cbainport.ini`. *Do not make changes directly to this file* If you do so, the changes will not be stored properly.

- To change Configuration Baseline Importer installation information, open a command prompt and type:

```
ovconfchg -ns cbainport -set keyname value
```

▶ If there are spaces in the value, type the value with quotes (“ ”). For example, if you want to change the value for `input_data_path`, the command line would read as follows:

```
ovconfchg -ns cbainport -set input_data_path
"D:\IntegrationServer\etc\cba\CBA_Import"
```

- If you make a mistake and want to set a value back to its default, type

```
ovconfchg -ns cbainport -clear keyname
```

- If you want to set all values back to their defaults, type

```
ovconfchg -ns cbainport -clear -all
```

- If you want to see all of the non-default values, type

```
ovconfchg -edit
```

The keys and values that you may change are listed in the table below.

**Table 1: Importer keys and values**

Key	Default value	Description
db_name	HPOV_CBA	The name of the Configuration Baseline Auditor database created on the SQL Server.
host_ip	Localhost	The host name or IP address for the SQL Server.
port	1433	The port number for the SQL Server.

<b>Key</b>	<b>Default value</b>	<b>Description</b>
username	sa	The User ID for the Configuration Baseline Auditor database created on the SQL Server.
error_data_path	<i>SystemDrive:</i> /HP OpenView/data/ datafiles/cba/import/ error	The path where collected data will be stored if it cannot be imported into the Configuration Baseline Auditor database.
input_data_path	<i>SystemDrive:</i> /HP OpenView/data/datafile s/cba/import	The path where the Configuration Baseline Importer looks for data to import into the Configuration Baseline Auditor database. Note: This directory should be the same as the directory specified earlier in <code>put.cfg</code> .
archiving	Default value is whatever was chosen during the installation.	This indicates whether you want to archive collected data or not. Set this key to True or False.
archive_data_path	<i>SystemDrive:</i> /HP OpenView/data/datafile s/cba/import/archive	The path where collected data will be stored after it is successfully imported into the Configuration Baseline Auditor database if Archiving is set to TRUE. Note: Do not move any files from the <code>archive_data_path</code> directory to the <code>input_data_path</code> directory with the intention of re-importing the information. If a file is in the archive directory, it has already been successfully imported to the database.
pause_time	60	The time the Configuration Baseline Importer waits, in seconds, before checking to see if there is new data to import.

After making the changes, you must perform the following steps to ensure the changes get picked up.

- 1 Run `ovc -stop cbaimport`.

- 2 Run `ovc -start cbaimport`. For further information on the `ovc` command, see *SystemDrive:\Program Files\HP OpenView\help\en\ov\_java\control\control\html*.



If you want to confirm that the Configuration Baseline Importer is running, type `ovc -status`.

## Changing your Configuration Baseline Auditor Database password

If necessary, you can change the password that you set up during the initial configuration in your Configuration Baseline Auditor Database that you created on your SQL Server.

### To change your password

- 1 Go to *SystemDrive:\Program Files\HP Openview\bin* and double-click `cbaimport_chgpwd.bat`.
- 2 When prompted, type the new password and press **Enter**.
- 3 Run `ovc -stop cbaimport`.
- 4 Run `ovc -start cbaimport`. For further information on the `ovc` command, see *SystemDrive:\Program Files\HP OpenView\help\en\ov\_java\control\control\html*.

## Task 4 Installing the Configuration Baseline Auditor

Use the Configuration Baseline Auditor to compare different states within a subsystem.

### To install the Configuration Baseline Auditor

- 1 From the Server Management CD, go to `\Configuration Baseline\Importer` and double-click `setup.cmd`.

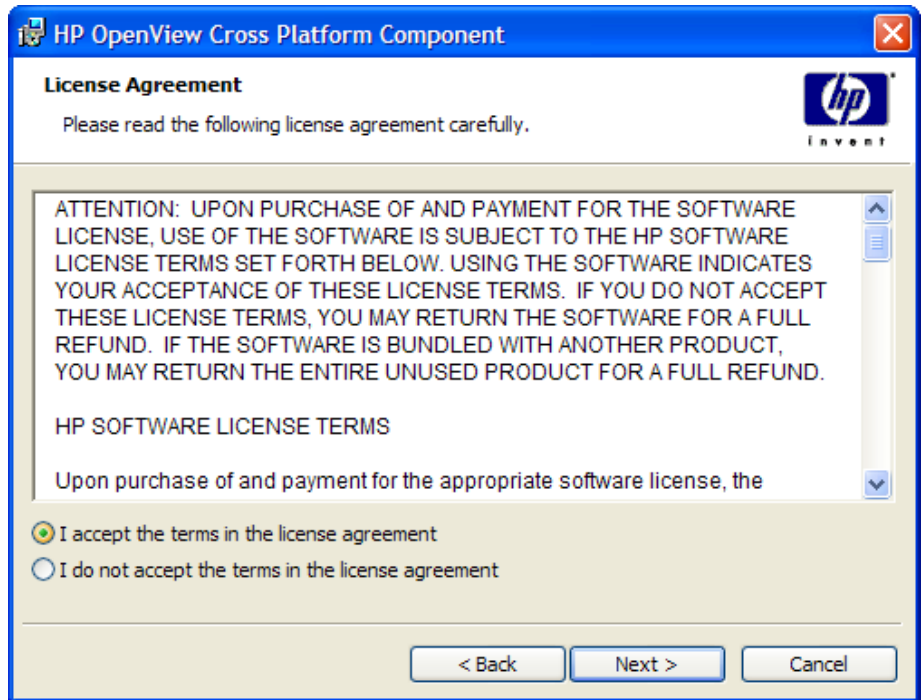
First, the HP OpenView common components will be installed. These components are a pre-requisite for the Configuration Baseline Importer and provide secure communication and control, configuration, and installation of other components. They also install a local java runtime that is necessary for the HP OpenView environment.

The Welcome to the Installer for HP OpenView Cross Platform Component window opens.

- ▶ The interface for the HP OpenView Cross Platform Component window opens only if you have not previously installed the HP OpenView common components on the machine. If the components are already installed, you will see several progress bars which will perform additional installations to the same directory used previously. When they are done, skip to step 6.

2 Click **Next**.

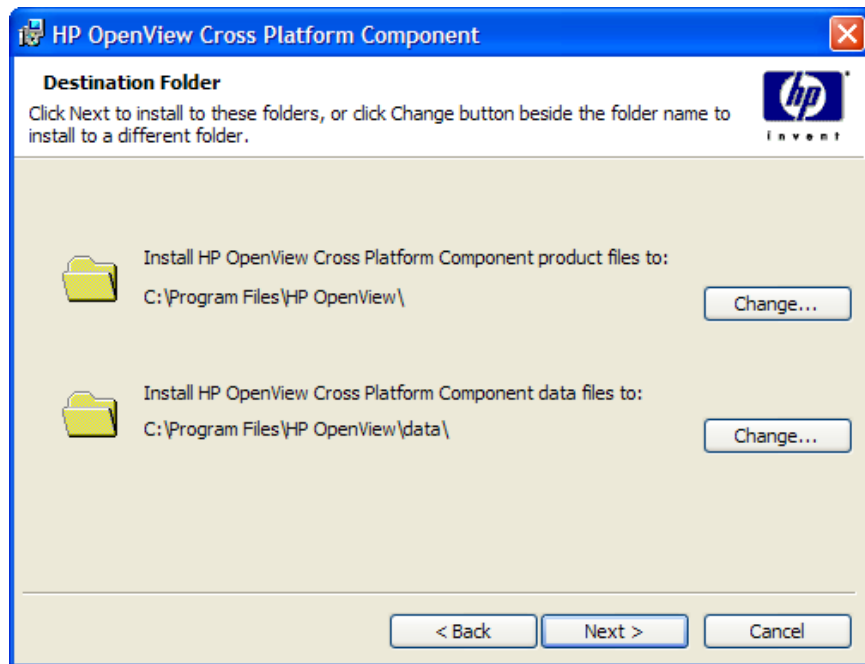
The End User Licensing Agreement window opens.



3 Select whether you accept the terms of the agreement or not and then click **Next**. If you do not accept the terms, the installation will be terminated.

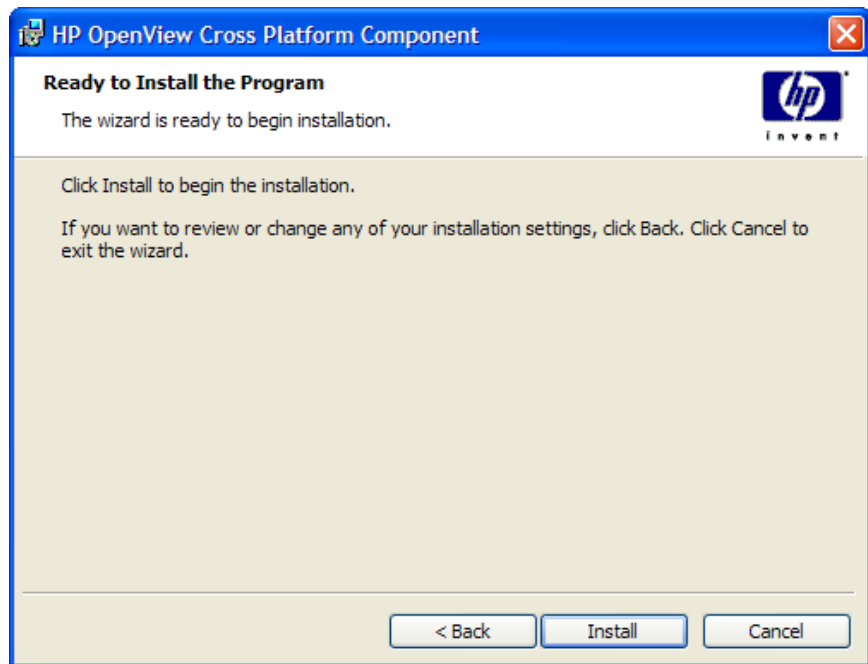
The Destination folder window opens.



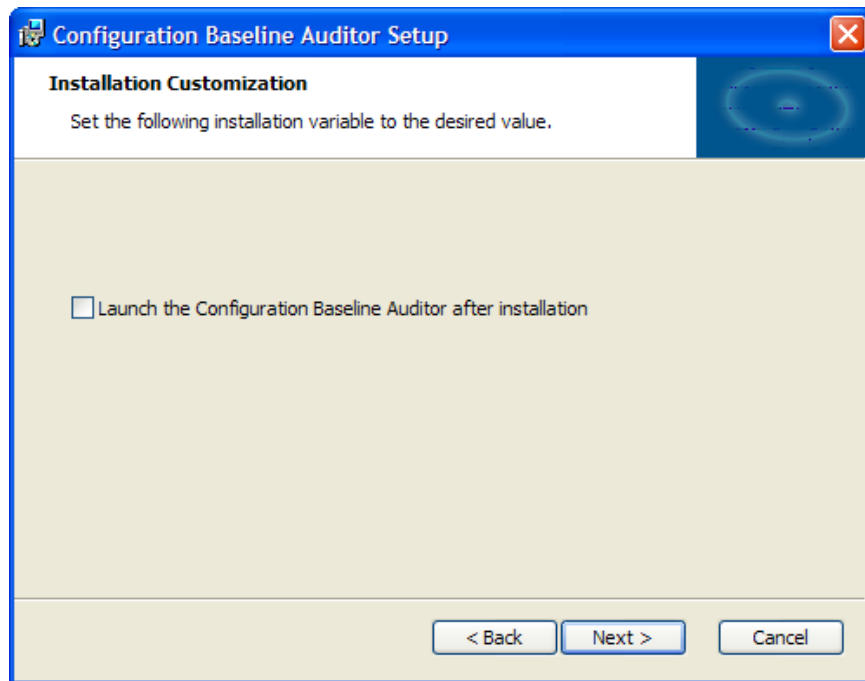


- 4 Accept the defaults or click **Change** to change the installation directories. Then, click **Next**.

The Ready to Install the Program window opens.



- 5 Click **Install**.
- 6 Click **Finish** when the HP OpenView Cross Platform component installation is complete. You will see several progress bars appear while the rest of the components are installed.  
When they are done, the Welcome to the Configuration Baseline Auditor Setup window opens.
- 7 Click **Next**.  
The End-User Licensing Agreement window opens.
- 8 Click **Next**.  
The Launch the Configuration Baseline Auditor window opens.



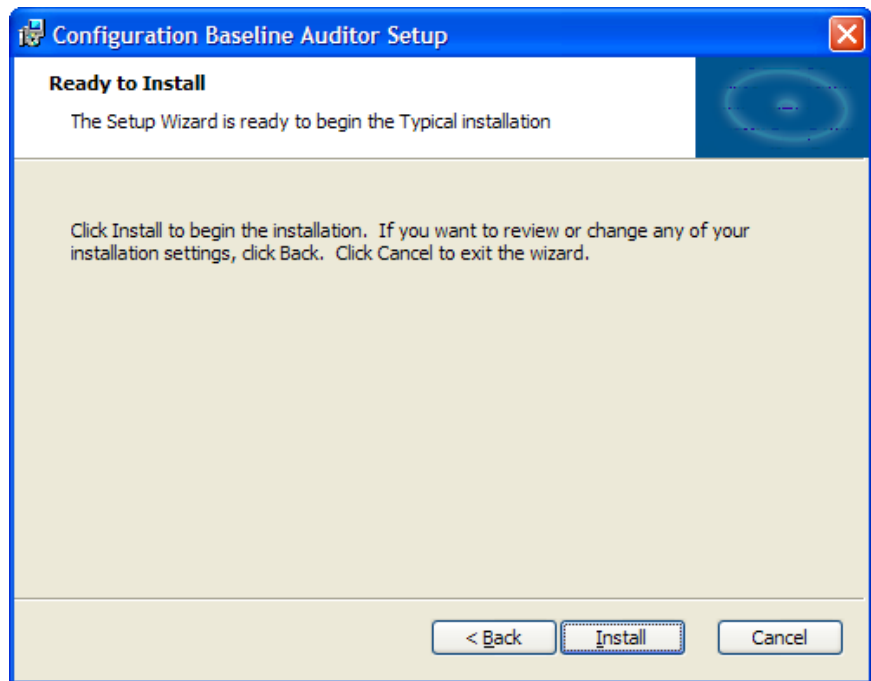
- 9 Click the checkbox if you would like to launch the Configuration Baseline Auditor.



If you do not have any data imported into your Configuration Baseline Auditor Database, you will receive an error message “Could not retrieve data from the database”. Close the Configuration Baseline Auditor and see the following chapters to create a subsystem and capture data.

- 10 Click **Next**.

The Install window opens.



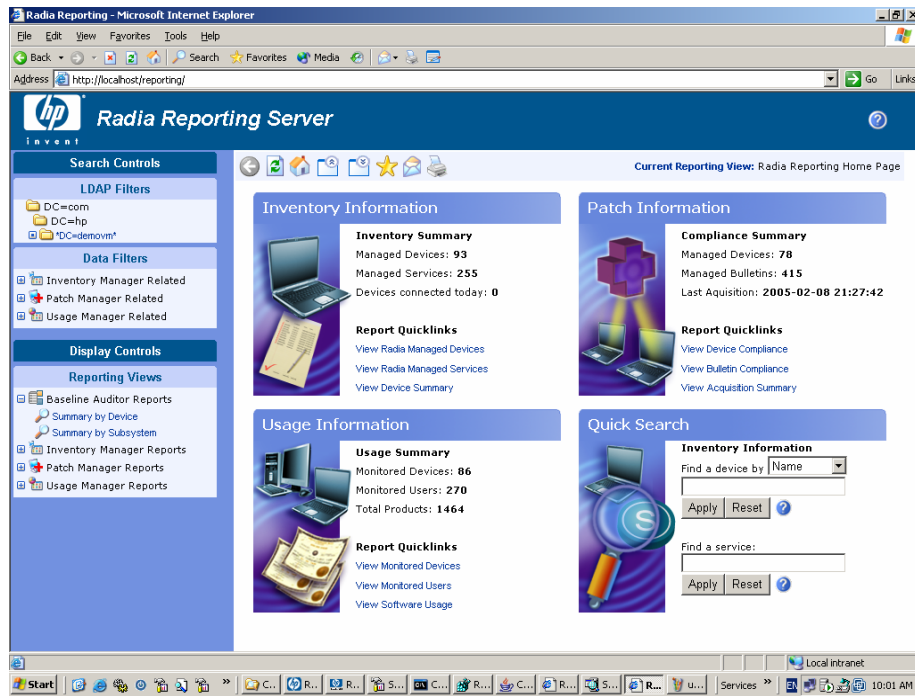
11 Click **Install**.

12 Click **Finish** to exit the Setup Wizard.

You may be asked to restart your machine. Click **Yes**.

#### **Task 5**    Configuring the Radia Reporting Server

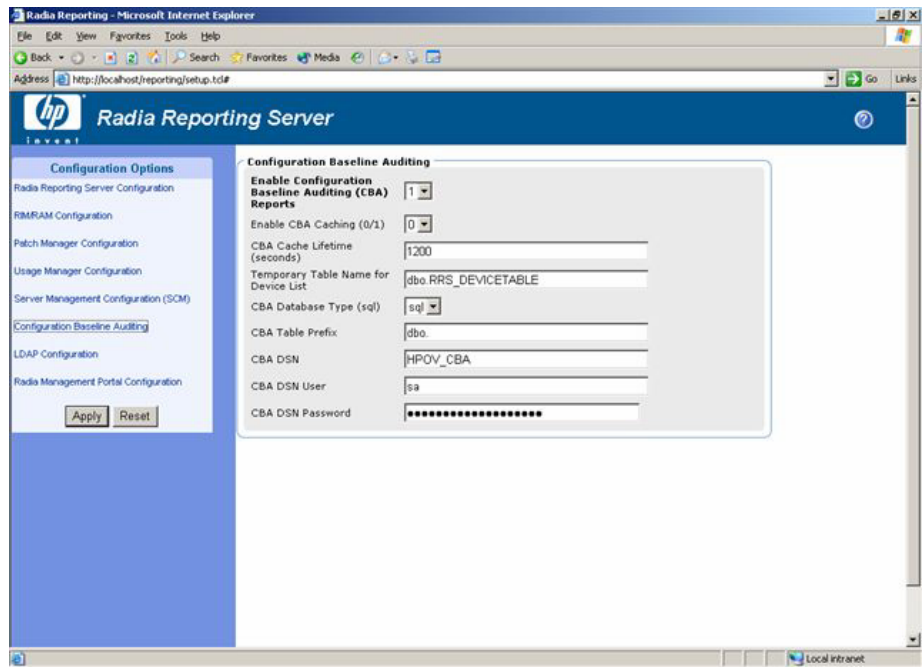
The Reporting Server has been enhanced to enable reporting against your Configuration Baseline Auditor SQL database to provide summary reports of Configuration Baseline Auditor data.



## To enable Configuration Baseline Auditor Reporting

- 1 Open a Web browser and type the address **http://ip\_address/reporting/setup.tcl** and press **Enter** on your keyboard.

This launches the interface for configuring the Reporting Server.



- 2 In **Configuration Options**, click **Configuration Baseline Auditing** and then review the following options.

**Table 2: Configuration Options for the Reporting Server**

Option	Description	Defaults
Enable Configuration Baseline Auditing Reports	Set this option to 1 to enable reporting or to 0 to disable reporting.	1
Enable CBA Caching	Set this option to 1 to enable caching or to 0 to disable caching. Caching stores the report information to the Reporting Server so it can be retrieved locally instead of retrieving the information from the database.	0
CBA Cache Lifetime (in seconds)	Defines how long to keep cached information on the	1200 (20 minutes)

Option	Description	Defaults
	Reporting Server before purging it and retrieving subsequent requests from the database.	
Temporary Table Name for Device List	This is an internal Radia Reporting Server requirement.	dbo.RRS_DEVICETABLE is the default. Typically, this is not modified.
CBA Database Type	Sql is for Microsoft SQL Server and is currently the only option.	SQL
CBA Table Prefix	Owner name of the CBA database.	dbo
CBA DSN	ODBC Data Set Name.	HPOV_CBA
CBA DSN User	ODBC Data Set Name User ID	Sa
CBA DSN Password	ODBC Data Set password. Note: This password will be encrypted by the Reporting Server.	sa

- 3 Click **Apply**.

#### Enabling the Reporting Server Configuration File

You must enable the new Reporting Server configuration file, `ReportingServer\bin\config.new.tcl`. To do this:

- 1 Close any open Reporting Server windows.
- 2 Rename your old configuration file, `config.tcl` to `old_config.tcl` or save it elsewhere.
- 3 Rename `config.new.tcl` to `config.tcl`.
- 4 Launch the Reporting Server.

You will learn about how to define subsystems, collect data and compare states in the following chapters.



## 3 Defining Subsystems

The key to successful implementation of the Configuration Baseline Auditor is to properly define subsystems. Remember, a subsystem is a logical group of resources (files and registry settings) for which you define scan filters to determine what information will be captured. When defining a subsystem, you need to:

- 1 Specify the subsystems to track in your environment. Subsystems should be associated with a well-known set of resources. Typically, this is an application or service. In some cases, it might be a common infrastructure component such as a Java runtime engine.
- 2 Determine the resources to be included or excluded from a subsystem. Common techniques for determining resources are to examine installation program metadata and scan for differences after an application has been installed.
- 3 Determine the attributes to monitor for changes.
  - Resources such as executable files should be scanned for changes in size, CRC and MD5 signatures. MD5 signatures for Windows executables are always generated, but MD5 generation is optional for non-executable files.
  - Data files that routinely change as part of normal processing should be excluded from a subsystem, or at least be defined to ignore their volatile attributes.
  - Permission attributes should be monitored for protected resources.

Use the BASELINE domain in the System Explorer to define subsystems.

## Example: Creating and Configuring a Subsystem

This section provides a simple example of how to create and configure a subsystem. See About the Baseline Domain on page 79 for more information about the BASELINE domain.

In this example, we will create a subsystem called Microsoft SQL Server 2000 that uses file and registry filters. File filters define the specific files that you want to scan in the subsystem. Registry filters define specific registry keys to scan.

Note that to use this example, you must have Microsoft SQL Server 2000 installed. If you do not have Microsoft SQL Server 2000, you can choose an application that is on your machine and define filters that match that application.



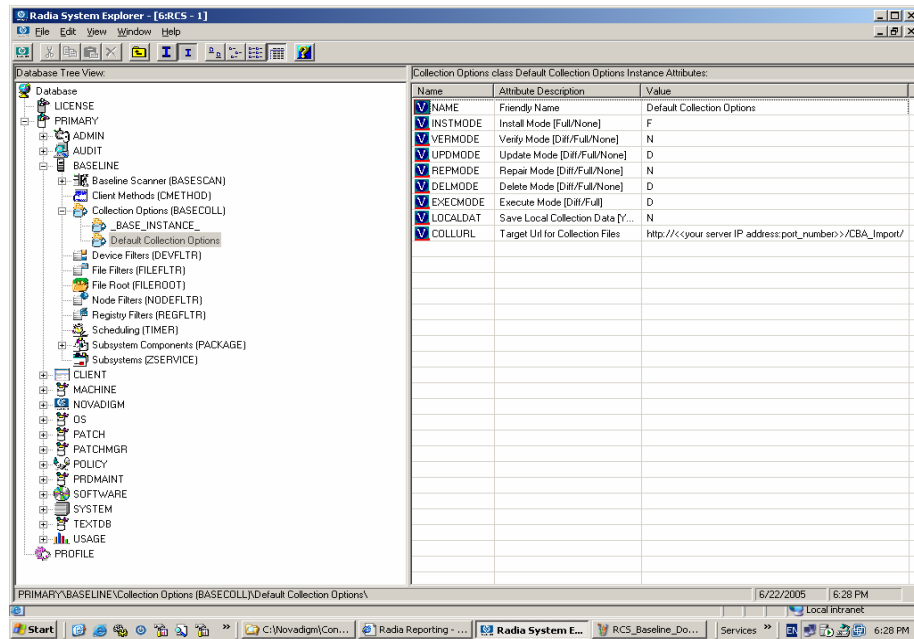
The steps below assume knowledge of how to use the System Explorer. If you are not familiar with this tool, please refer to the *System Explorer Guide* for details about how to use this tool.

### Creating a Subsystem

#### Task 1 Specify Collection Options

Collection options define the type of collection to be performed and when. Types of collections include full, differential or none. The collection is performed during a Radia client connect.

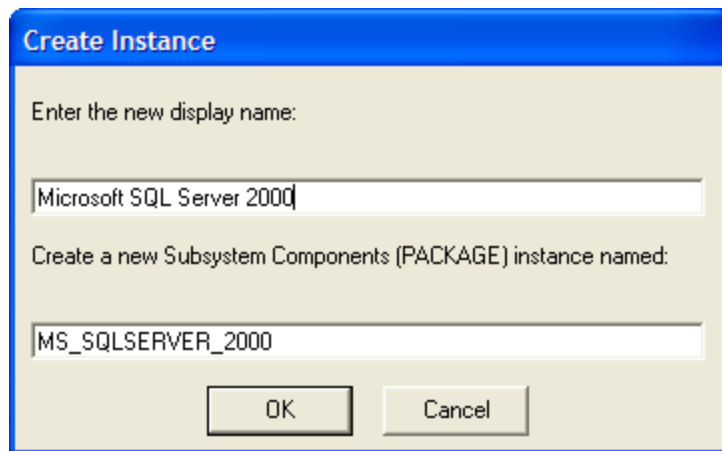
- 1 Open the System Explorer.
- 2 Go to **PRIMARY.BASELINE.BASECOLL**.
- 3 Open the **Default Collection Options** instance.



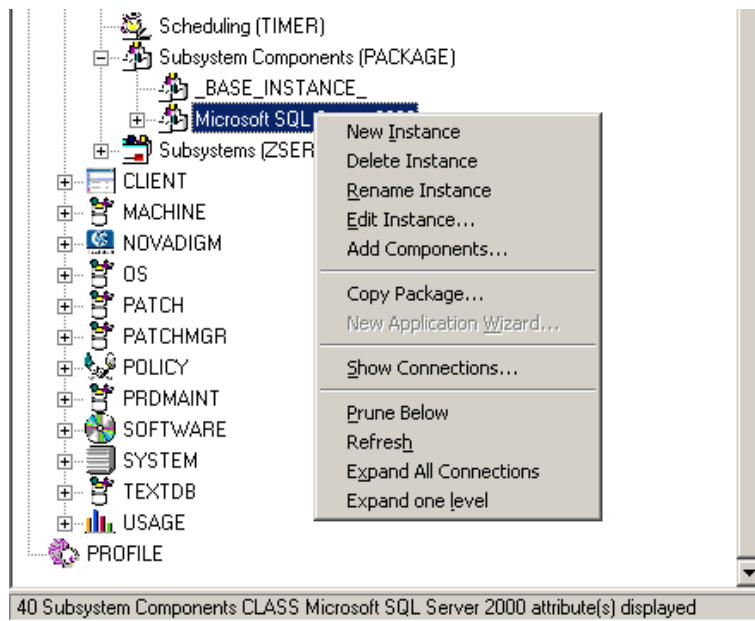
- 4 Open the **COLLURL** attribute and modify it to point to your Radia Integration Server. For example:  
`http://your server IP address:port_number/CBA_Import/`  
 becomes  
`http://192.168.1.1:3466/CBA_Import/`
- 5 For this example, do not change any of the other default options.
- 6 Click **OK** and **Yes** to confirm your changes.

## Task 2 Create a package and specify the file and registry filters

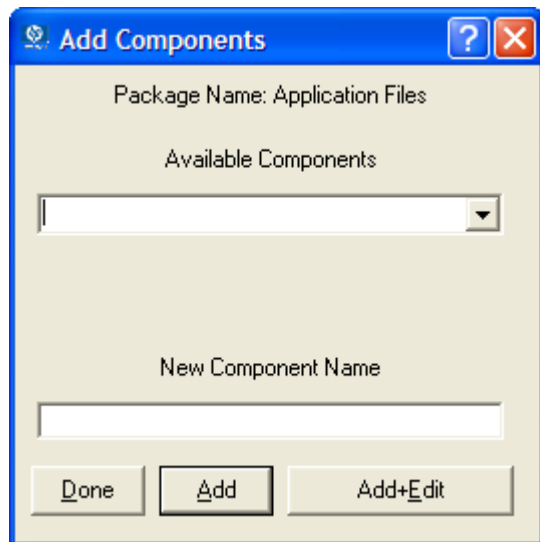
- 1 If necessary, open the System Explorer.
- 2 Go to **PRIMARY.BASELINE.PACKAGE**.
- 3 Create a new package instance. In our example, we will name the package **MS\_SQLSERVER\_2000**.



- 4 Click **OK**.
- 5 Right-click the new package and select **Add New Components**.

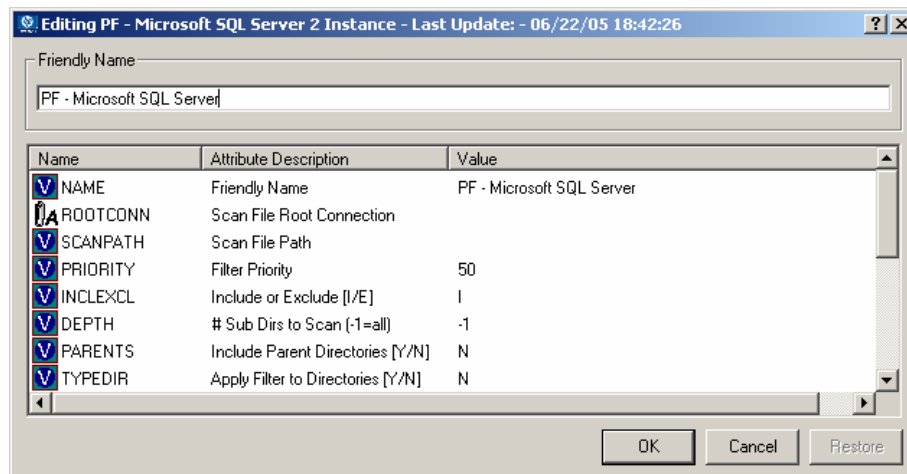


The Add New Components window opens.



- 6 From the **Available Components** drop-down menu, select **File Filters**.
- 7 In the **New Component Name** text box, type the friendly name for the directory containing the files that you want to capture. For example, you might type **PF - Microsoft SQL Server** to indicate that you want to capture the files from Program Files\Microsoft SQL Server.
- 8 Click **Add + Edit**.

The Edit Class window opens.

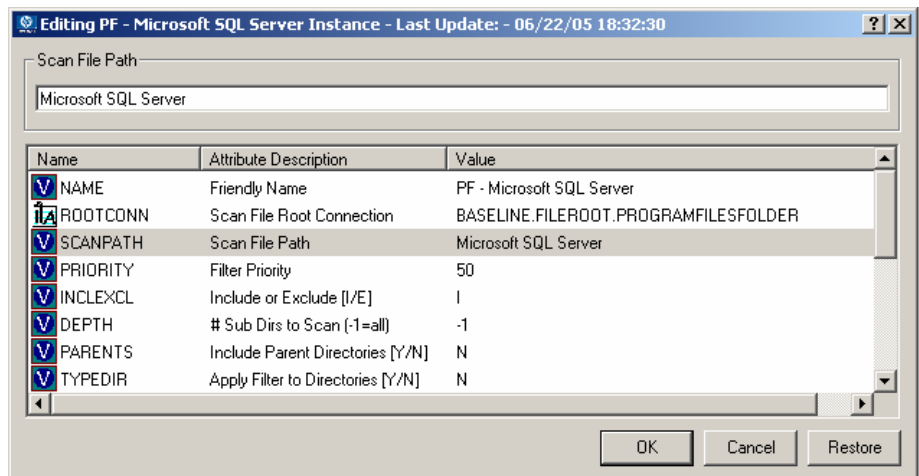


- 9 You have two options for defining the path contents. You can use the logical folder settings that are predefined in the **BASELINE** domain in the **FILEROOT** class. If this option is chosen, the path name defined in the **SCANPATH** attribute is appended to this **FILEROOT** chosen to form a fully qualified path name. You may also choose to use define the **SCANPATH** attribute as a fully qualified name (i.e., `C:\Program Files\Microsoft SQL Server`). When using the **ROOTCONN** option, connect the appropriate logical folder instance (in the **FILEROOT** class) to the package.



You may need to review the instances in the **BASELINE.FILEROOT** class to determine what the appropriate instance is for your needs.

- 10 For this example, go to the **ROOTCONN** attribute and type the connection to the logical folder, such as **BASELINE.FILEROOT.PROGRAMFILESFOLDER**.



- 11 When a **ROOTCONN** is defined, set the **SCANPATH** attribute to the folder where the files that you want to scan are stored. If you opened Windows Explorer, typically Microsoft SQL Server files are stored in `C:\Program Files\Microsoft SQL Server`. Therefore, in this example, you would type **Microsoft SQL Server**.



Do not specify the full path (`C:\Program Files`) since you connected this instance to a logical folder to define where the folder can be found on a particular device.

You may also choose to use define the **SCANPATH** attribute as a fully qualified name (i.e., C:\Program Files\Microsoft SQL Server) and leave the **ROOTCONN** attribute empty.

12 Click **OK**.

13 Click **Yes** to save your changes.

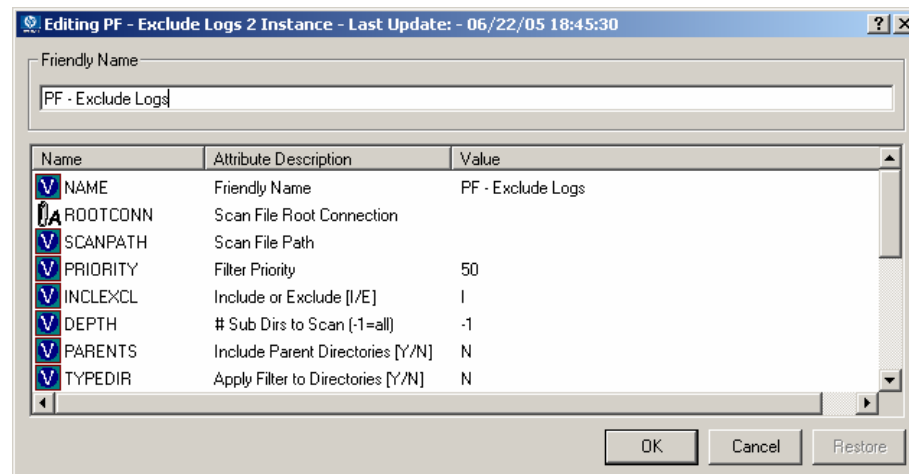
The Add Components dialog box opens again.

14 From the **Available Components** drop-down menu, select **File Filters**.

15 In the **New Component Name** text box, type the friendly name for the directory containing the files that you want to capture. For example, you might type **PF - Exclude Logs** to indicate that you do not want to capture log files from the logs directory in Program Files\Microsoft SQL Server.

16 Click **Add + Edit**.

The Edit Class window opens.

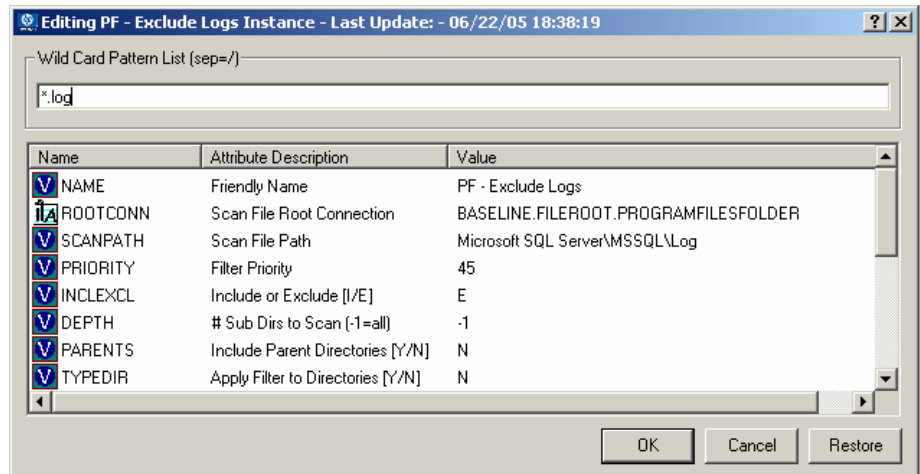


17 Connect the appropriate logical folder instance (in the **FILEROOT** class) to the package.



You may need to review the instances in the **BASELINE.FILEROOT** class to determine what the appropriate instance is for your needs.

- 18 For this example, go to the ROOTCONN attribute and type the connection to the logical folder, such as **BASELINE.FILEROOT.PROGRAMFILESFOLDER**.



- 19 When a ROOTCONN is defined, set the SCANPATH attribute to the folder where the files that you want to scan are stored. If you opened Windows Explorer, typically Microsoft SQL Server files are stored in C:\Program Files\Microsoft SQL Server\MSSQL\LOG. Therefore, in this example, you would type **Microsoft SQL Server\MSSQL\LOG**.



Do not specify the full path (C:\Program Files) since you connected this instance to a logical folder to define where the folder can be found on a particular device.

You may also choose to use define the SCANPATH attribute as a fully qualified name (i.e., C:\Program Files\Microsoft SQL Server\MSSQL\LOG) and leave the ROOTCONN attribute empty.

- 20 Set the INCLEXCL attribute to E to exclude the logs from the scan.
- 21 Change the PRIORITY from the default of 50 to a higher priority, such as 45.
- 22 Set the WILDCARD attribute to \*.log to ensure that all logs with the .log extension will be excluded.
- 23 Click **OK**.
- 24 Click **Yes** to save your changes.

The Add Components dialog box opens again.

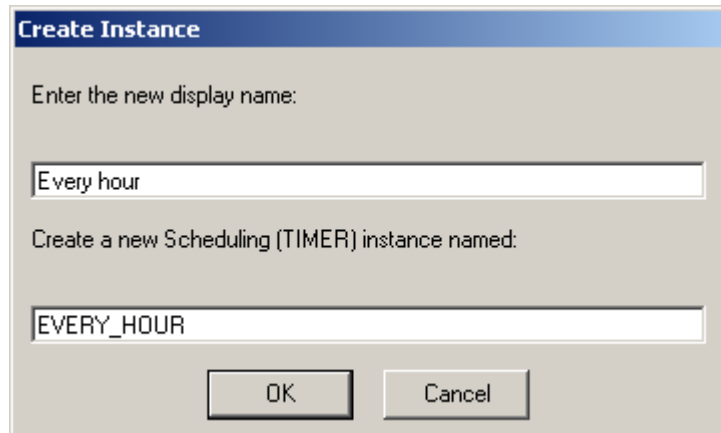


- 25 From the **Available Components** drop-down list box, select **Registry Filters**.
- 26 In the **New Component Name** text box, type a friendly name for the registry filter such as **HKLM - Microsoft SQL Server**.
- 27 Click **Add + Edit**.  
The Editing HKLM – Microsoft SQL Server window opens.
- 28 Set the **SCANPATH** attribute to the path to be scanned in the registry editor, such as `Software\Microsoft\Microsoft SQL Server`.
- 29 Click **OK**.
- 30 Click **Yes** to save your changes. The Add Components window opens again.
- 31 Click **Done**.

You have created a package that contains three filters—two for files and one for registry keys.

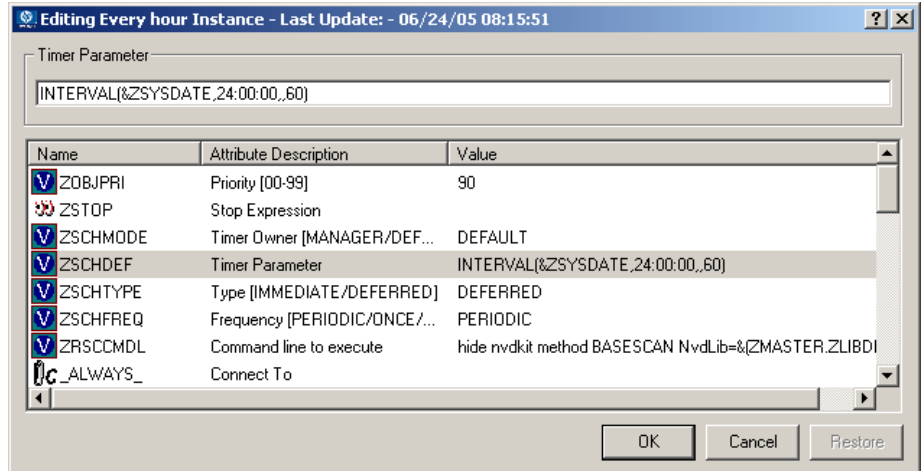
### Task 3 Create a periodic timer

- 1 If necessary, open the System Explorer.
- 2 Go to **PRIMARY.BASELINE.TIMER**.
- 3 Create a new timer instance which will run every hour starting at midnight. In our example, we will name the timer **EVERY\_HOUR**.



- 4 Click **OK**.

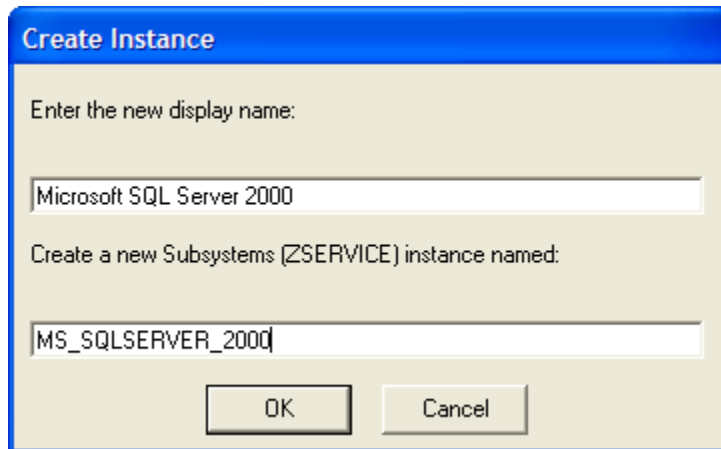
- Open the new instance and change the ZSCHDEF attribute to reflect an interval baseline scan occurring every hour. Set the **ZSCHDEF** attribute to **INTERVAL(&ZSYSDATE,24:00:00,,60)** to indicate that this timer should expire every 60 minutes starting at midnight.



- Click **OK**.

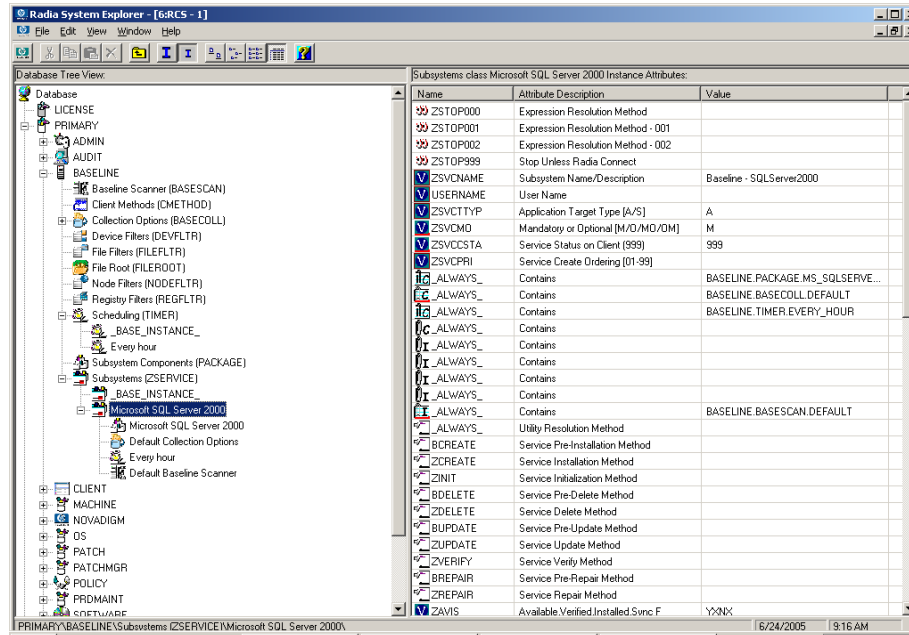
#### Task 4 Create a service

- Create a new ZSERVICE instance called Microsoft SQL Server 2000.



- Click **OK**.
- Drag your **Microsoft SQL Server 2000** package and drop it on the new ZSERVICE instance.

- 4 In the Select Connection Attribute dialog box, click **Copy**.
- 5 Click **Yes** to confirm that you want to make the connection. Then, click **OK**.
- 6 Drag your **Every Hour** timer and drop it on the new ZSERVICE instance.
- 7 In the Select Connection Attribute dialog box, click **Copy**.
- 8 Click **Yes** to confirm that you want to make the connection. Then, click **OK**.



In most cases, subsystems will be managed as mandatory services through the Application Manager client. Open the Microsoft SQL Server ZSERVICE and change the following attributes:

- **ZSVCNAME = Baseline - SQLServer2000**  
Modify this attribute because this is what appears as the service definition in the SQL database service name and the Software Manager catalog.
- **DESCRIPT = Microsoft SQLServer 2000**  
This will be the subsystem description that will be stored as the subsystem name in the SQL Database and displayed in the Configuration Baseline Auditor.

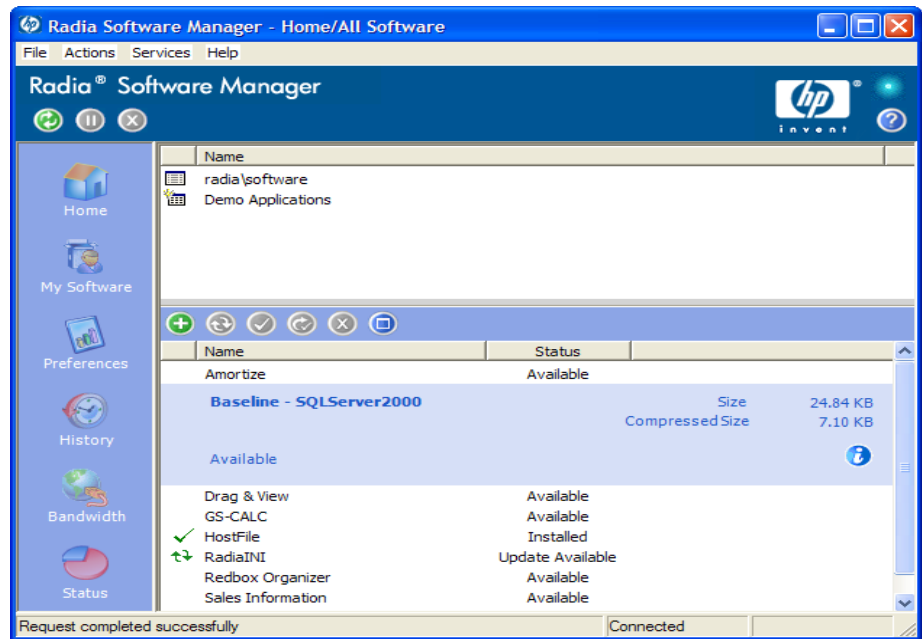
- 9 Click **OK**.
- 10 Click **Yes**.
- 11 Now that your service is configured, you must connect it to a policy instance. For this example, go to **PRIMARY.POLICY.WORKGROUPS**.
- 12 Drag the **Microsoft SQL Server 2000 ZSERVICE** instance to the **Default** workgroup. Click **Copy** and then click **Yes** to confirm the connection.
- 13 Click **OK**.

**Task 5** Run a client connect to initiate a scan


Now that your subsystem is configured, the next time a client connect or a timer expires and runs, the scan will be performed.

For unit testing purposes, you may want to use the Software Manager client. To do so, set ZSVCTYP to S and ZSVCMO to O so the service will appear in the Software Manager catalog. Refer to the *Software Manager Guide* for more information.

In this example, we have set up our subsystem to appear in the Software Manager catalog.



- 1 Notice **Baseline – SQLServer2000** in the catalog.

- 2 Click **Baseline – SQLServer2000** and then click the **Install**  button.
- 3 The Baseline scanner runs a scan based on the subsystem you defined.
- 4 When the scan is complete, the status bar will say “Request completed successfully”.



If you would like to perform a simple test, you can go into the *SystemDrive:\Program Files\Microsoft SQL Server* folder and make changes (add, remove or modify) to the files or directories.

Use the System Explorer to make sure that the **VERMODE** attribute in the **Collection Options (BASECOLL)** instance is set to **D**.

Then, select **Baseline – SQLServer2000** again and click **Verify**  which is set to perform a differential scan by default.



## 4 Viewing the Results

There are two ways to review the results of scans performed with the Configuration Baseline Auditor—using the Configuration Baseline Auditor or the Reporting Server.

- Use the Configuration Baseline Auditor to view the results of your scheduled scans and select the subsystems that you want to compare. The Configuration Baseline Auditor is java-based and runs in a server/administrator environment.

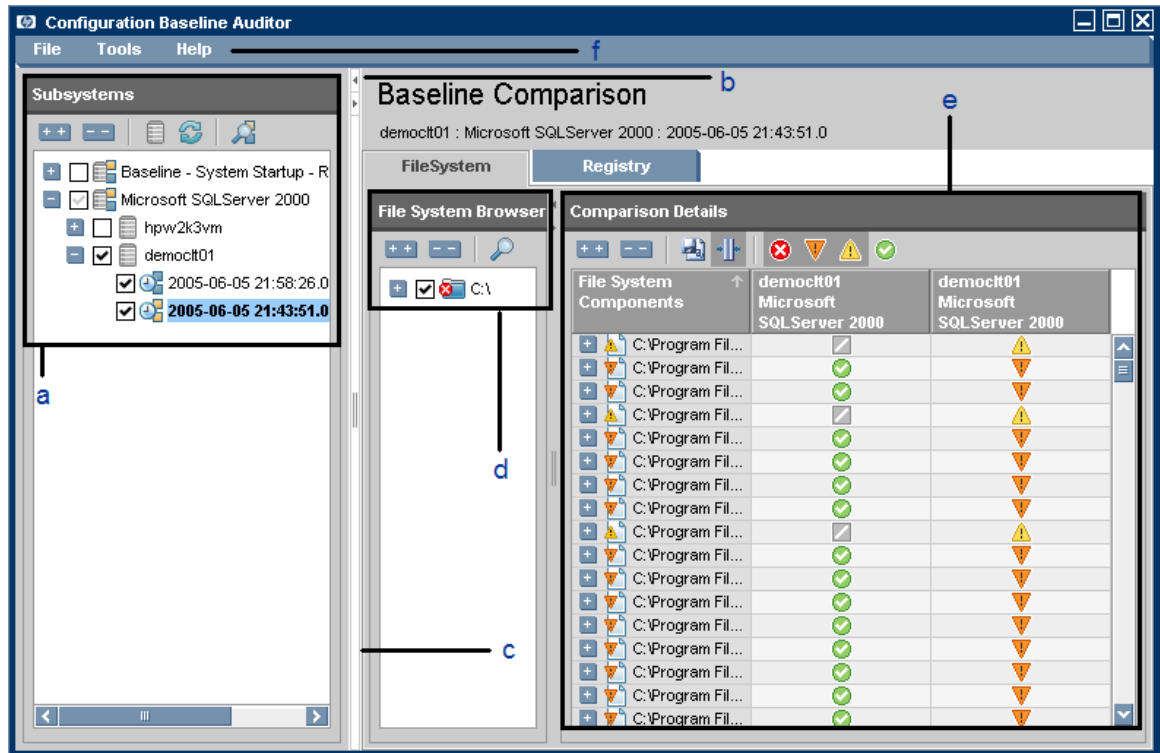
Use the Configuration Baseline Auditor to view and compare detailed information collected with each scan.

- The Reporting Server allows reporting against your Configuration Baseline Auditor SQL database to provide summary reports of Configuration Baseline Auditor data.

Use the Reporting Server when you need summary information about which servers and subsystems are under management and to see when the latest collections have occurred for these subsystems.

# About the Configuration Baseline Auditor

Review the diagram below for an introduction to the user interface.



**Figure 2: Configuration Baseline Auditor**

## a Subsystems

Lists the subsystems that you have defined. Expand a subsystem to view the states captured for the subsystem.

A baseline is a state that you select as the standard against which you want to compare other states for the subsystem. A state is set as the baseline by selecting it first, or by right-clicking any state and selecting **Set as Baseline State**. Baselines are shown in bold.



Expand all subsystems.





Collapse all subsystems.



Toggle between View by Subsystem and View by Node.



Refresh.



Compare selected states.

b Hide/show buttons for the Subsystem browser.

c Window resizer

Click and drag to resize the window.

d File System Browser/Registry Browser

Allows you to view the files or registry entries that have been scanned.



Expand all.



Collapse all.



Show details for selected items.

e Comparison details

Allows you to compare the attributes for the states you selected.



Expand all.



Collapse all.



Export report as CSV, HTML or XML.



Enable or disable autofit for columns.



Hide/show missing items.



Hide/show different items.



Hide/show added items.



Hide/show identical items.

#### f **Menu bar**

Go to **File** and select **Connection** to connect to another SQL database.

Go to **File** and select **Exit** to close the Configuration Baseline Auditor.

Go to **Tools** and select **Options** to set the default filters that will be in effect when performing a new comparison.



You can also perform many tasks in the interface via pop-up menus. Right-click within the interface to become familiar with the tasks that may be performed this way.

# Using the Configuration Baseline Auditor Interface

One of the most general uses of the Configuration Baseline Auditor is to set up a reference machine and capture the baseline of a subsystem. Then, you can check to see if subsystems on other machines in your environment match the baseline, or if there are differences in the configuration.

As you learned in the introduction to the interface, the left side of the interface displays subsystem definitions. Beneath each subsystem is a state that has been captured. You can select one or more states to review the details of the capture and, if you've selected more than one state, make comparisons.

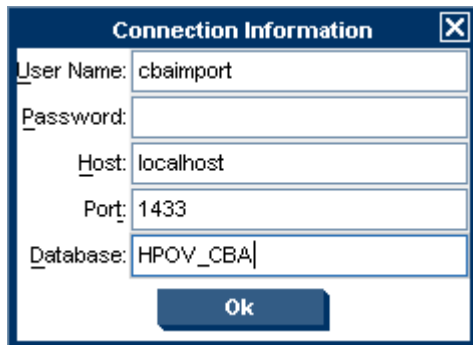
Use the filters available in the Comparison Details window to limit the scope of the comparison. For example, you can limit the scope so that you can view only items that have been added. If you expand the components, you can see the attributes. Finally, you can export the displayed information to a CSV, XML or HTML file.

In the example below, we assume that you ran a full scan of your Microsoft SQL Server 2000 subsystem. Imagine that we later installed Service Pack 3a for Microsoft SQL Server 2000 and ran another scan. The steps below will show you how to compare the states of the subsystem to see how the subsystem has changed.

## To compare states

- 1 From the **Start** menu, go to **HP Configuration Baseline Auditor**, and select **Configuration Baseline Auditor**.

The Connection Information dialog box opens.



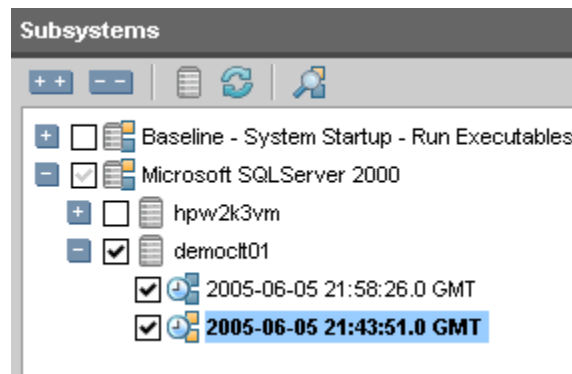
Connection Information	
User Name:	cbaimport
Password:	
Host:	localhost
Port:	1433
Database:	HPOV_CBA
Ok	

- 2 In the **User Name** text box, type your user name for the SQL database.

- 3 In the **Password** text box, type the password for the SQL database.
- 4 In the **Host** text box, type the location of the SQL database.
- 5 In the **Port** text box, type the port for the SQL database.
- 6 In the **Database** text box, type the name of the SQL database.
- 7 Click **OK**.

The Configuration Baseline Auditor window opens.

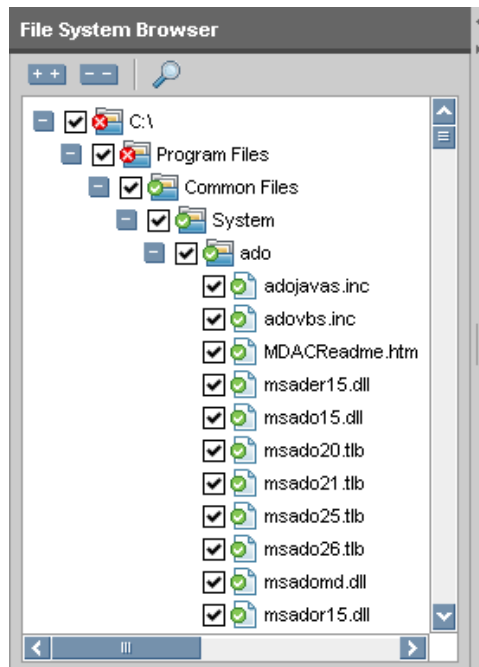
- 8 In the **Subsystems** area, select the subsystems or nodes that you want to compare. Remember, you can toggle between subsystems and nodes.
- 9 Specify a baseline. A state is set as the baseline by selecting it first, or by right-clicking any state and selecting **Set as Baseline State**. Baselines are shown in bold.




- 10 Click **Compare Selected States** .

When the comparison completes, the **File System Browser** and the **Comparison Details** areas are populated.

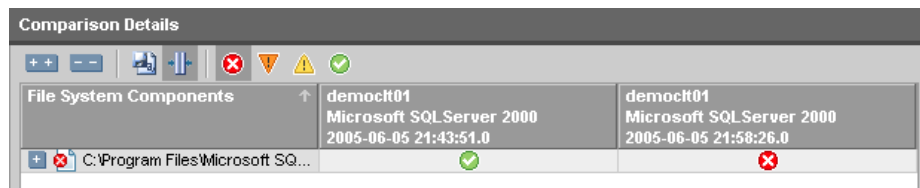




12 Click  show details for selected items.

The **Comparison Details** area is populated.

13 Use the buttons available to filter the items to review. For example, you can hide identical items, but show missing, different, or added items. In this example, we hid everything except for missing items.



14 Click to plus sign to expand the items listed in the **File System** or **Registry Components** column for detailed information.

Comparison Details		
File System Components	democft01 Microsoft SQL Server 2000 2005-06-05 21:43:51.0	democft01 Microsoft SQL Server 2000 2005-06-05 21:58:26.0
C:\Program Files\Microsoft SQL...	✓	✗
<b>File Name</b>	XPADSI.EXE	----
<b>Physical Root</b>	C:\	----
<b>Physical Path</b>	Program Files\Microsoft SQL Server\M...	----
<b>Logical Root</b>	ProgramFilesFolder\	----
<b>Logical Path</b>	Microsoft SQL Server\MSSQL\Binn	----
<b>Product Version</b>	8.0.1.94	----
<b>Vendor</b>	Microsoft Corporation	----
<b>File MD5</b>	883EA23715CD1C79E75BFD5EA8C6E...	----
<b>File dateTime</b>	2000-08-06 08:50:34.0	----
<b>Product</b>	Microsoft SQL Server	----
<b>File Size</b>	65600	----
<b>File Version</b>	2000.80.194.0	----

Notice that the file `XPADSI.EXE` existed in the original Microsoft SQL Server scan. However, when Service Pack 3a was installed, this file was removed and this difference is shown in the later scan.

#### To remove devices or subsystems from the database

You can right-click on a device or subsystem to remove the information from the SQL database.

- ▶ The delete option is available only if you are using the View by Node option in the Subsystems pane on the left. If you are using the View by Subsystem option, you can not delete items.

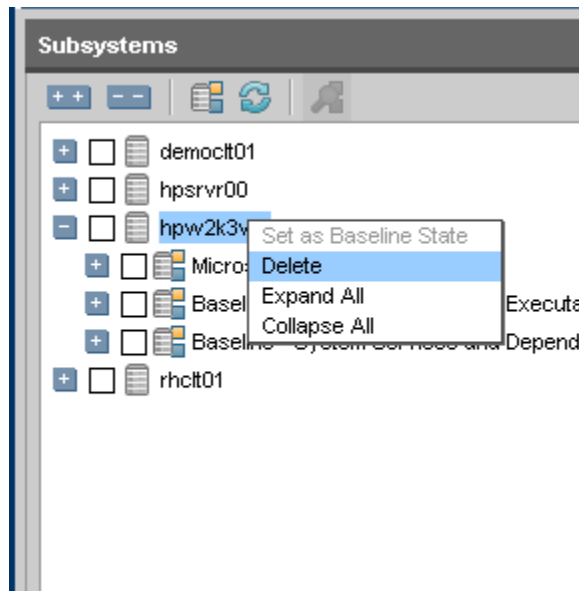
When a device is deleted, the device, all subsystems associated with the device, and all states associated with all subsystems are marked for permanent deletion from the SQL database.

When a subsystem is deleted, the subsystem associated with the specific device, and all states associated with the specific subsystem are marked for permanent deletion from the SQL database. The device will not be deleted.

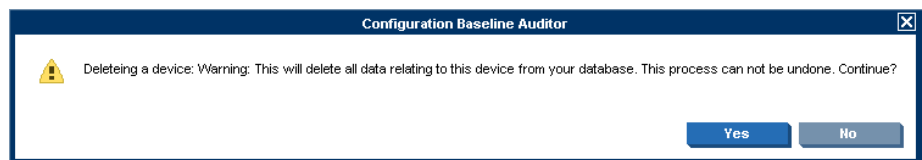
The Configuration Baseline Auditor interface will no longer display the items selected for permanent deletion.

#### To delete a device

- 1 In the Subsystems pane, right-click a device and select **Delete**.



A confirmation dialog box opens.

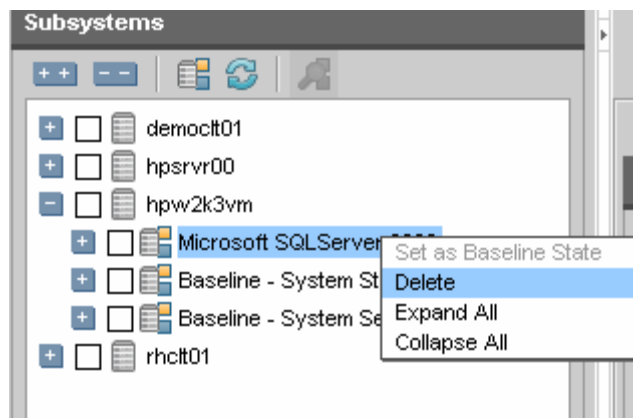


- 2 Click **Yes** to delete the device and all related subsystem data from the SQL database. Click **No** to cancel the request.

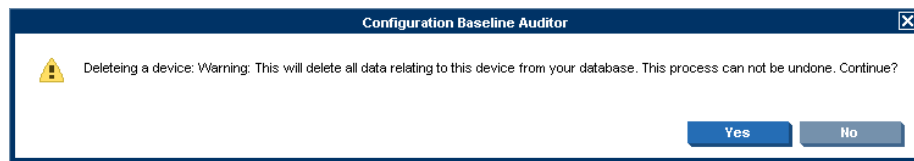
To delete a subsystem and all associated states

- 1 In the Subsystems pane, right-click the subsystem to be deleted and select **Delete**.





A confirmation dialog box opens.

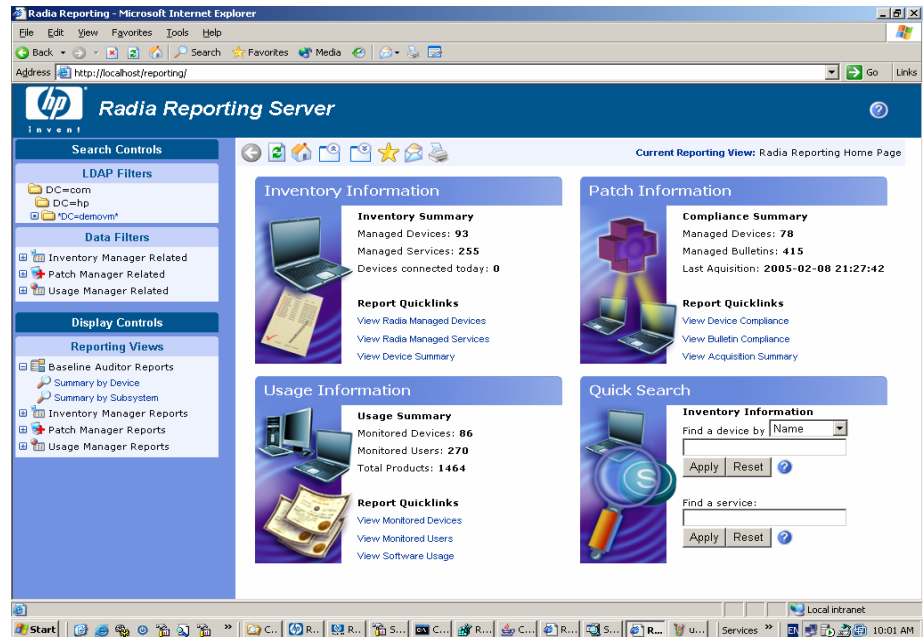


- 2 Click **Yes** to delete the subsystem and all related state data from the SQL database. Click **No** to cancel the request.

# Generating Configuration Baseline Auditor Reports

To generate Configuration Baseline Auditor Reports

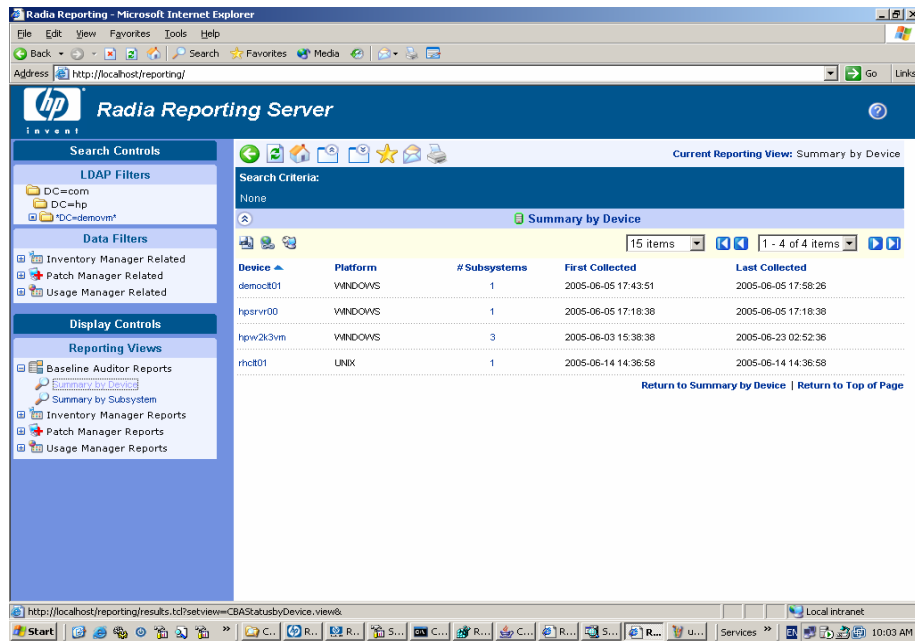
- 1 In your Web browser session, type the address `http://ip_address/reporting/` to launch the Reporting Server home page. If enabled, **Baseline Auditor Reports** will be available in the **Reporting Views** pane.



- 2 Click **Baseline Auditor Reports** to see two reporting options.

- **Summary by Device**
- **Summary by Subsystem**

In the image below, you can see a Summary by Device report.

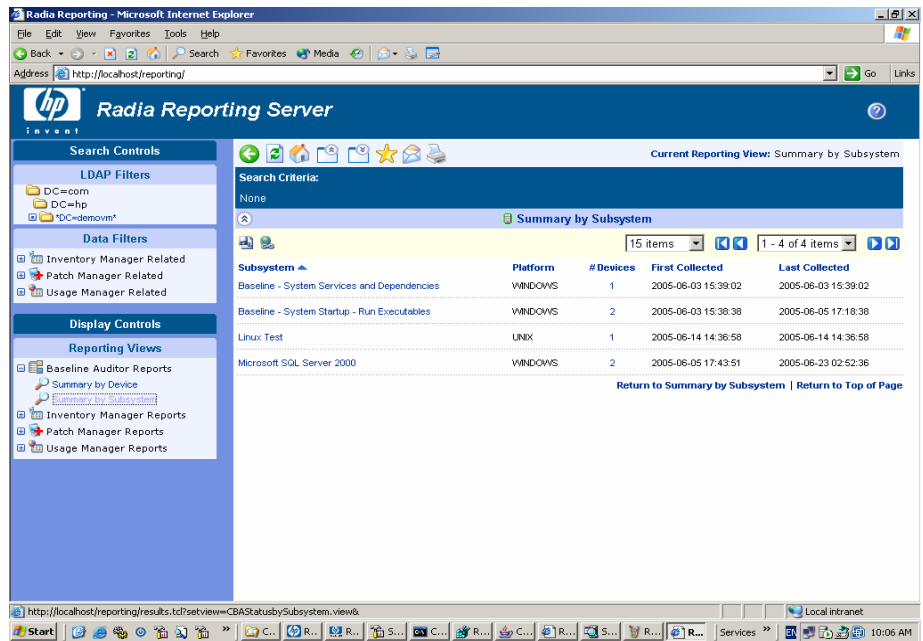


This report provides the following information.

**Table 3: Summary by Device Report**

Column Name	Description
Device	Click the device name to display a list of the subsystems that exist in the database for this device.
Platform	Describes the operating system platform.
# Subsystems	The number of unique subsystems in the SQL database for each device. Click on the number to show the subsystem information for this device. This is displayed in the <b>Summary by Device, by Subsystem</b> report.
First Collected	The first date/time that data was collected for any of the subsystem scans for the device.
Last Collected	The last date/time that data was collected for any of the subsystem scans for the device.

In the image below, you can see a Summary by Subsystem report.



This report provides the following information.

**Table 4: Summary by Subsystem Report**

Column Name	Description
Subsystem	Click the subsystem name to display a list of the devices that exist in the SQL database for this subsystem.
Platform	Describes the operating system platform.
# Devices	The number of unique devices in the SQL database for a subsystem. Click the number to show the names of the devices associated with a subsystem. This is displayed in the <b>Summary by Subsystem, by Device</b> report.
First Collected	The first date/time that data was collected for any of the subsystem scans for all devices.
Last Collected	The last date/time that data was collected for any of the subsystem scans for all devices.

The **Summary by Subsystem, by Device** report shows the number of states associated with a specific device for a specific subsystem. Click the **# States** column to see detailed state information.

The screenshot shows the Radia Reporting Server interface in a Microsoft Internet Explorer browser window. The address bar shows `http://localhost/reporting/`. The page title is "Radia Reporting Server".

**Search Controls:**

- LDAP Filters:** DC=com, DC=hp, \*DC=demovm\*
- Data Filters:** Inventory Manager Related, Patch Manager Related, Usage Manager Related
- Reporting Views:** Baseline Auditor Reports, Summary by Device, Summary by Subsystem, Inventory Manager Reports, Patch Manager Reports, Usage Manager Reports

**Search Criteria:**

- WINDOWFILTER FILTERS
- CBASQL Platform Name (WINDOWS)
- CBASQL Subsystem Name (Microsoft SQL Server 2000)

**Summary by Subsystem** (15 items, 1 - 1 of 1 items):

Subsystem	Platform	# Devices	First Collected	Last Collected
Microsoft SQL Server 2000	WINDOWS	2	2005-06-05 17:43:51	2005-06-23 02:52:36

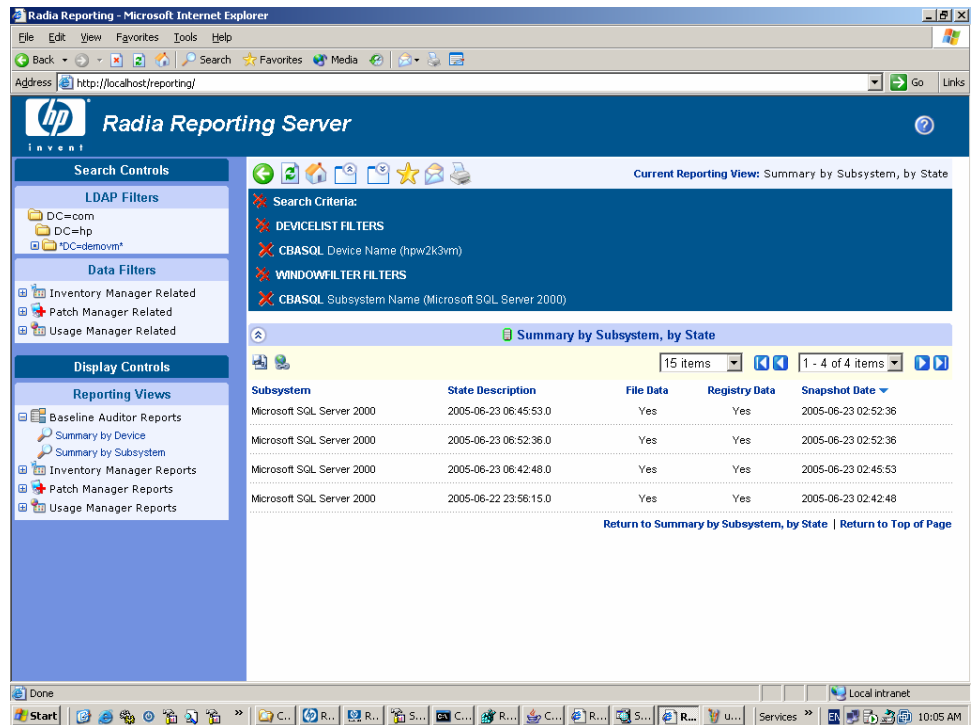
[Return to Summary by Subsystem](#) | [Return to Top of Page](#)

**Summary by Subsystem, By Device** (15 items, 1 - 2 of 2 items):

Subsystem	Device	# States	First Collected	Last Collected
Microsoft SQL Server 2000	democ01	2	2005-06-05 17:43:51	2005-06-05 17:58:26
Microsoft SQL Server 2000	hpw2k3vm	4	2005-06-22 19:56:15	2005-06-23 02:52:36

[Return to Summary by Subsystem, By Device](#) | [Return to Top of Page](#)

The **Summary by Subsystem, by State** report shows further detail for each State.

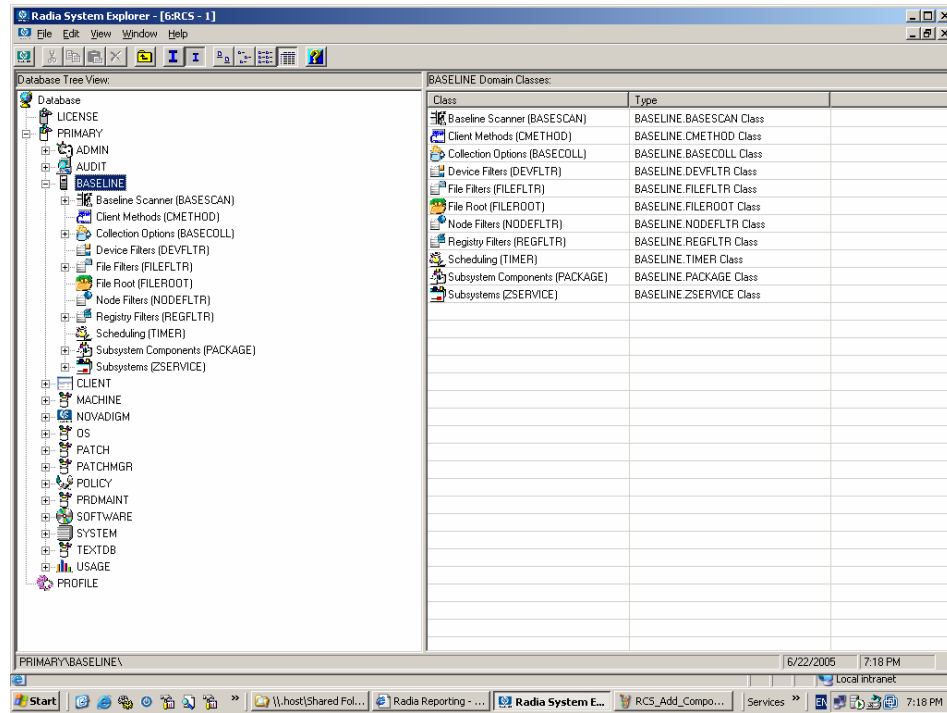


**Table 5: Summary by Subsystem, by State report**

Column Name	Description
Subsystem	Click the name of the subsystem to display the device names for which subsystem information exists in the database.
State Description	The description assigned to the subsystem during the scan request. It is automatically set to the time/date (GMT) when the scan was started on the target device.
File Data	Indicates whether file system data was captured during the scan.
Registry Data	Indicates whether Windows registry data was captured during the scan.
Snapshot Date	The date/time the scan was started.

# A About the Baseline Domain

Now that you have configured a subsystem and its collection options, you may want to become more familiar with the BASELINE domain. This domain contains several standard Radia class definitions, as well as several specific definitions associated with the Configuration Baseline Auditor.



The standard Radia classes are:

- Subsystems (ZSERVICE)
- Subsystem Components (PACKAGE)
- Scheduling (TIMER)

The specific definitions associated with the Configuration Baseline Auditor are described in the following sections.

## Baseline Scanner

The Baseline Scanner class defines the agent methods of the Configuration Baseline Auditor which capture subsystem information for a managed server.

This class is for internal use only.



Do not alter the instances in this class.

The Baseline Scanner package (currently file `basescan.tkd`) is distributed with the HP OpenView Application Manager Using Radia (RAM) client install and is maintained using normal RAM client update procedures through the PRDMAINT domain.

For more information on configuring client maintenance refer to the *Database Reference Guide for the HP OpenView Configuration Server Using Radia*.

## Client Methods (CMethod)

This class is for internal use only.



Do not alter the instances in this class.

## Collection Options (BASECOLL)

Use this class to define the scope of the collection during a client connect. The collection may be done during an install, verify, update, repair, delete or ad hoc execution. You can select what type of collection to perform, if any.

- **Full (F)**  
A full scan is performed and all data gathered by the baseline scanner and its filters is transferred to the Radia Integration Server.
- **Differential (D)**  
A differential scan is performed and only the data which has changed since the last scan was taken is transferred to the Radia Integration Server. If this is the first time that the scan is performed and the collection type is set to **Differential**, a full scan will be performed. In the future, only the differences will be collected.
- **None (N)**  
No scanning is performed.



**Table 6: Collection Options (BASECOLL) attributes**

Attribute	Description	Default values
NAME	Friendly name.	N/A
INSTMODE	Indicates what type of scan to perform when a service is installed.	F
VERMODE	Indicates what type of scan to perform when a service is verified.	N
UPDMODE	Indicates what type of scan to perform when a service is updated.	D
REPMODE	Indicates what type of scan to perform when a service is repaired.	N
DELMODE	Indicates what type of scan to perform when a service is deleted.	D
EXECMODE	Indicates what type of scan to perform when a service is executed.	D
LOCALDAT	Indicates whether to save a local copy of the data collected (and stored in an XML file).	N
COLLURL	Defines the target URL on the Radia Integration Server where the collected XML data files will be stored.	<pre>http://your server IP address:port_number/CBA_ Import/</pre> <p><i>You must update this attribute with the IP address and port for your Radia Integration Server.</i></p>

## Device Filters (DEVFLTR)

This class is for future use.

## File Filters (FILEFLTR)

Use this class to define what specific files should be scanned.

One or more file filters can be connected to a Subsystem (ZSERVICE) instance.

**Table 7: FILEFLTR attributes**

<b>Attribute</b>	<b>Description</b>	<b>Default values</b>
NAME	Friendly name	N/A
ROOTCONN	Specifies a connection to a logical folder such as PROGRAMFILESFOLDER.	N/A
SCANPATH	Defines the path to scan.	Specifies where the files that you want to scan are stored. If a ROOTCONN attribute is defined, the path name defined in the SCANPATH attribute is appended to associated FILEROOT to form a fully qualified path name. For example, PROGRAMFILESFOLDER\Microsoft SQL Server. You may also choose to use define the SCANPATH attribute as a fully qualified name (i.e., C:\Program Files\Microsoft SQL Server\MSSQL\LOG) and leave the ROOTCONN attribute empty.
PRIORITY	Sets the filter priority; 0 is the highest.	50
INCLEXCL	Specifies the type of filter—whether it includes information or excludes information.	I
DEPTH	Specifies the number of subdirectories to scan. Type -1 to scan all subdirectories.	-1
PARENTS	Specifies whether to include the parent directories in the captured data.	N

<b>Attribute</b>	<b>Description</b>	<b>Default values</b>
TYPEDIR	Indicates whether to apply the filter to directory names.	N
TYPEEXEC	Indicates whether to apply the filter to executable names.	Y
TYPEDATA	Indicates whether to apply the filter to data files (non-executable files).	Y
EXACT	Specifies exact file names to match. Use a forward slash to separate file names if you are specifying more than one.	N/A
WILDCARD	Specifies a wildcard file name pattern to match. Use a forward slash to separate multiple generic names.  Use the asterisk (*) or the question mark (?) to identify patterns.  For example: ABC???EF.exe/*.dll/*.ocx	*
REGEXP	Enter a regular expression. A regular expression is a string pattern. It can be used to match file and registry name patterns. CBA uses the TCL regular expression parser.  <i>Refer to the TCL documentation about regular expression parsing.</i>	N/A
SIZEDATA	Indicates whether to collect file size data during the scan.	Y
TIMEDATA	Indicates whether to collect file time data during the scan.	Y

<b>Attribute</b>	<b>Description</b>	<b>Default values</b>
CRCDATA	Indicates whether to compute a full CRC for the data content during the scan process. A CRC is automatically generated for Windows executables even when this option is set to 'N'.	N
MD5DATA	Indicates whether to compute a full MD5 signature for the data content during the scan process. An MD5 signature is automatically generated for Windows executables even when this option is set to 'N'.	N
PERMDATA	Indicates whether to collect file permission and access right data during the scan process.	N

## File Root (FILEROOT)

Use this class to define the base location of a file. For example, you can specify a drive such as C drive, or a well known folder location, like `ProgramFiles` folder.

## Node Filters (NODEFLTR)

This class is for future use.

## Registry Filters (REGFLTR)

Use this class to identify the registry keys to be scanned.

**Table 8: Registry Filters (REGFLTR) Attributes**

<b>Attribute</b>	<b>Description</b>	<b>Default values</b>
NAME	The friendly name.	_BASE_INSTANCE_
SCANROOT	Indicates where to begin the scan by specifying the root name of the path.	HKLM
SCANPATH	Specifies the pathname where the files that you want to scan are stored. For example, Software\Microsoft\Microsoft SQL Server.	N/A
PRIORITY	Specifies the filter priority. 0 is the highest.	50
INCLEXCL	Specifies whether the filter type is an include or exclude.	I
DEPTH	Specifies the subkey scan depth. Enter a positive integer number or -1 for all subkeys.	-1
PARENTS	Indicates whether to include parent keys in the captured scan data.	N
TYPEKEY	Indicates whether to apply filter to registry key names.	N
TYPEVALU	Indicates whether to apply filter to registry value names.	Y
EXACT	Specifies exact file names to match. Use a forward slash to separate multiple file names.	N/A

Attribute	Description	Default values
WILDCARD	<p>Specifies a wildcard file name pattern to match. Use a forward slash to separate multiple generic names.</p> <p>Use the asterisk (*) or the question mark (?) to identify patterns.</p> <p>For example: VAL????00/VALU*/*VALU</p>	*
REGEXP	<p>Enter a <b>regular expression</b>. A regular expression is a string pattern. It can be used to match file and registry name patterns. CBA uses the TCL regular expression parser.</p> <p><i>Refer to the TCL documentation about regular expression parsing.</i></p>	N/A
REGDATA	Indicates whether to collect registry value data information during the scan process.	Y
MAXDATA	Specifies the maximum size of the registry value data information to be collected. Maximum data size collected is 254 bytes followed by a null.	255
PERMDATA	Indicates whether to collect registry key permission and access right data during the scan process.	N

## Subsystem Components (PACKAGE)

Use this class to define the components for the subsystem. When you create an instance, you can specify file filters, registry filters or both.

## Subsystems (ZSERVICE)

Use this class to define the service, noting the following.

- The value for the ZSVCNAME attribute is what appears in the Application Manager or Software Manager.
- The value for the DESCRIPT attribute is the friendly name used within the Radia Database. It is also used to identify the subsystem in the SQL database and when displaying subsystems in the Configuration Baseline Auditor.





## B Troubleshooting

To troubleshoot the Configuration Baseline Auditor, you must first attempt to determine where the problem occurred. Some questions you might ask yourself are:

- **Did you configure your subsystem properly?**

For example, did you update the IP address and port number in COLLURL attribute in the `Collection Options` to specify the directory on the Radia Integration Server to store the collected information?

- **Is the Importer running?**

From a command prompt, run `ovc -status` to see the status of the Importer.

- **Is your client connect working?**

View the Radia `connect.log` file located in your `SystemDrive:\Program Files\Novadigm\Log` directory to view the status of the data capture request.

- **Did you specify the same directory in the put.cfg file and in the Configuration Baseline Importer?**

The directory specified in `put.cfg` *must* match the directory where the Configuration Baseline Importer will look for data to import into the SQL Server Database. This directory is set during the installation of the Configuration Baseline Importer.

### Logging

You can find the answers to some of this information in more in the following logs:

**Table 9: Logs**

Component	Location of Log
Radia Integration Server	<code>SystemDrive:\Novadigm\IntegrationServer\logs</code>

Component	Location of Log
HP OpenView Configuration Baseline Auditor interface and the Importer	<i>SystemDrive</i> :\Program Files\HP Openview\data\log
Client connect issues	<i>SystemDrive</i> :\Program Files\Novadigm\Log and review the connect.log. Note that it may help to search for the keyword “basescan” to review the actions related to the scan.
ZSERVICE object	<i>SystemDrive</i> :\Program Files\Novadigm\Lib\username\radia\software\ZSERVICE\servicename\00000000.000

## Increased Logging Details in Radia’s method.log

You can use the following options to provide detailed debugging information and logging in Radia’s *method.log* (stored by default in *SystemDrive*:\program files\novadigm\Log). The *method.log* is created when a timer instance is run.

To use these parameters, go to BASELINE.TIMER .InstanceName and open the ZRSCCMDL attribute which is set to `hide nvdkit method BASESCAN NvdLib=&(ZMASTER.ZLIBDRV) &(ZMASTER.ZLIBDIR)` by default.

You can use any of the debug options described in the table below as follows:

```
BASESCAN debug option
NvdLib=&(ZMASTER.ZLIBDRV) &(ZMASTER.ZLIBDIR)
```

NVDLIB tells the scanner where the service is installed on the local machine so that the script can run properly.

**Table 10: Debug options**

Debug options	Description
-debugtree	Debugs the Radia tree
-debugsend	Debugs HTTP send issues
-debugobject	Debugs captured object issues

<b>Debug options</b>	<b>Description</b>
-debugfile	Debugs file scan issues
-debugregistry	Debugs registry scan issues
-debugfilter	Debugs file/registry filter issues if the -debugfile or -debugregistry is turned on.
-debugid	Debugs element id issues
-debugscan	Debugs general scan issues
-debuglogical	Debugs logical name issues



# Glossary

## baseline

A baseline is a state that you select as the standard against which you want to compare other states for the subsystem.

## Baseline Scanner

Performs periodic scans on monitored devices to collect data for subsystems. The data collected by the Baseline Scanner is used to produce reports comparing subsystem states.

## device

A physical machine on which subsystems exist. The Baseline Scanner can collect hardware related state data for monitored devices.

## node

A node is typically associated with an operating system running on a device. There can be multiple nodes per device. For example, a dual boot device would have two nodes.

## subsystem

A subsystem is a logical group of resources (files and registry settings) for which you define scan filters to determine what information will be captured.

## Subsystem state (state)

A subsystem state (state) is the data captured for a subsystem at a specific point in time.



# Index

## A

About the Configuration Baseline Auditor  
user interface, 64  
archive\_data\_path, 38

## B

baseline, 64  
Baseline Domain  
About, 79  
drag and drop support, 14  
BASELINE domain, 10  
importing, 13

## C

Client Methods (CMethod), 80  
Collecting and Importing Data, 10  
Collection options, 50  
Collection Options (BASECOLL), 80  
COLLURL, 51  
Configuration Baseline Auditor, 10  
benefits, 7  
installing, 13, 39  
typical uses, 7  
what is it, 7  
Configuration Baseline Auditor Database  
changing the password, 39  
Configuration Baseline Auditor Reporting  
configuring, 45  
Configuration Baseline Auditor values  
ensuring changes are captured, 38  
return to default, 37  
review non-default values, 37  
set all values back to defaults, 37  
Configuration Baseline Importer, 24  
Configuration Baseline Importer, 10

installing, 13  
Configuration Baseline Importer  
installing, 24  
Configuration Baseline Importer  
modifying default values, 37  
Configuration Baseline Scanner, 10  
Configuring the Radia Reporting Server, 44  
Configuring Your Environment, 13  
Count by Device, 74  
Count by Subsystem, 74

## D

db\_name, 37  
Defining Subsystems, 10  
delete devices, 71  
delete subsystems, 71  
Device Filters (DEVFLTR), 81

## E

error  
Could not retrieve data from the database, 43  
error message, 43  
error\_data\_path, 38  
example  
creating a subsystem, 50

## F

File filters, 50  
File Filters (FILEFLTR), 81  
File Root (FILEROOT), 84

## G

Generating Configuration Baseline Auditor Reports,  
74

## H

Hardware and Software Requirements, 8  
host\_ip, 37  
HP OpenView common components, 24, 40

## I

input\_data\_path, 38

## L

Logging, 89  
logical folder, 54, 56

## M

Microsoft SQL Server Database, 13  
setup, 17

## N

Node Filters (NODEFLTR), 84

## P

pause\_time, 38  
port, 37  
Process flow, 11  
put.cfg, 16

## R

Radia Environment, 13  
configuring, 13  
Radia Integration Server, 10  
Radia Reporting Server, 10, 11  
configuring, 13  
Radia's method.log, 90  
registry filters, 50  
Registry Filters (REGFLTR), 84  
remove devices, 71  
remove subsystems, 71

Reporting, 10  
Reporting Server, 63  
ROOTCONN attribute, 54, 56

## S

SCANPATH attribute, 54, 56  
Specifying the data collection directory, 16  
state, 7  
status, 39  
subsystem  
defining, 7  
Subsystem Components (PACKAGE), 86  
subsystems, 7  
Subsystems  
defining, 49  
Subsystems (ZSERVICE), 87  
summary reports, 44  
Supported Platforms, 8  
System Explorer  
required version, 9

## T

Troubleshooting, 89

## U

user interface, 64  
username, 38  
Using the Configuration Baseline Auditor Interface,  
67

## V

Viewing the Results, 63

## X

XML files  
storing, 16