HP OpenView Server Management Using Radia

for the Windows and Unix operating systems

Software Version: 1.0

Configuration File Management Guide

Manufacturing Part Number: T3424-90088

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

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.

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

To register for an HP Passport ID, go to:

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

Contents

1 Introduction

The Configuration File Management (CFM) component of HP OpenView Server Management Using Radia provides a simple solution for administrators to manage changes to configuration files and implement these changes automatically.

> To use the Configuration File Management Client Agent, you must own a Server License To Use (LTU) for Application Manager using Radia.

Configuration files managed by CFM are text files that can be categorized as follows:

- Hierarchical files, such as INI files, which are divided into sections containing keys and values.
- Files that have a flat well-defined structure, such as hosts files, which contain records and fields.

To use CFM effectively, you will want to:

- Determine the desired change in behavior that affects the configuration file.
- Understand the current state of the configuration file, if one already exists.
- Decide what changes must be made or what file and entries must be created.
- Use the CFM domain in the System Explorer to create the desired instances, services, and connections, including setting up policy.

Once the desired instances, service, and connections are created, and a Radia client connect occurs, the standard client method applies the changes to the configuration file. If there is no configuration file, it will be created.

This guide describes how to install the Configuration File Management System, provides typical use cases and examples, and describes several classes in the CFM domain in detail.

Audience

System administrators who are managing servers in their environment using Radia products.

Prerequisites

You should be familiar with the following products:

- HP OpenView Configuration Server Using Radia
- HP OpenView Administrator Workstation Using Radia
- HP OpenView Application Manager Using Radia
- HP OpenView Management Portal Using Radia

Software Requirements

Radia Infrastructure

- HP OpenView Application Manager Using Radia for Windows, version 4.0.1 or later, or
- HP OpenView Application Manager Using Radia for Unix, version 4.1 or later

Additional Components

In addition to already established Radia products, Configuration File Management uses:

- New Configuration File Management (CFM) Domain and classes
- Server Management components for the Radia client

Chapters Summary

Installing Configuration File Management

This chapter describes the components required for Configuration File Management and includes instructions for installing the server components.

Managing Initialization (INI) Files

This chapter provides examples of how to use Configuration File Management (CFM) to create and update the unattended answer file that can be used with the System Optional Components Manager executable program (sysocmgr.exe) for Windows. This file's format conforms to INI file standards, containing sections, keys and values.

Managing Other Types of Configuration Files

This chapter provides examples of how to use Configuration File Management to create and update a hosts file. Hosts files serve as an example of other types of configuration files that can be managed by Configuration File Management because their contents can be easily specified as column values.

About the Configuration File Management Domain

Describes the CFM domain classes and instances necessary to manage your configuration files.

Related Documents

- *User's Guide for HP OpenView Configuration Server Using Radia*
- *Publisher Guide for HP OpenView Administrator Workstation Using Radia*
- *Installation and Configuration Guide for HP OpenView Application Manager Using Radia for Windows,* version 4.0.1 or later

- *Installation and Configuration Guide for HP OpenView Application Manager Using Radia for Unix,* version 4.1 or later
- *Application Management Profiles Guide for HP OpenView Server Management Using Radia*
- *Configuration Baseline Auditor Guide for HP OpenView Server Management Using Radia*
- *Installation and Configuration Guide for Windows Terminal Server and Citrix Support for HP OpenView Server Management Using Radia*

2 Installing Configuration File Management

Configuration File Management (CFM) requires an update to the Configuration Server database as well as the installation of a CFM component on the Radia client.

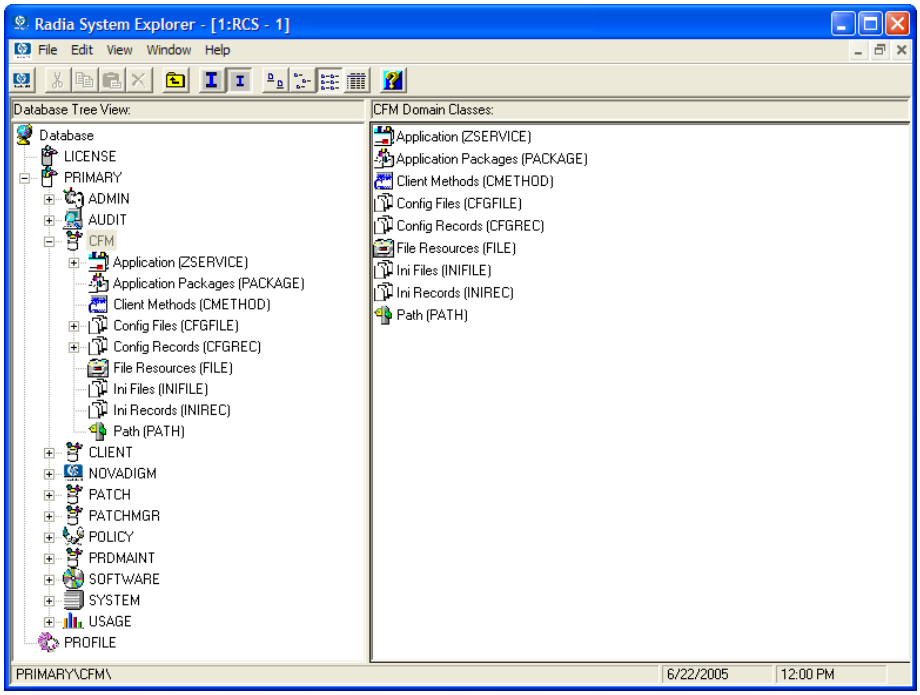
This chapter describes the steps required to update the Radia Configuration Server database. The required CFM client component is automatically installed by selecting the **Radia Server Management** option when installing the Radia client.

To use the Configuration File Management Client Agent, you must own a Server LTU for Application Manager using Radia.

To update the Radia Configuration Server database on Windows

1. Stop the **Radia Configuration Server** service.
2. Copy **CFM.xpc**, **CFM.xpi**, and import_CFM.bat to the Radia Configuration Server's bin directory (by default `SystemDrive:\Novadigm\ConfigurationServer\bin`).
3. Run import_cfm.bat.

   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 might also see a return code of `4` also, which you may ignore.
4. Start the Radia Configuration Server service.
5. Open the System Explorer to review the CFM domain as shown below.



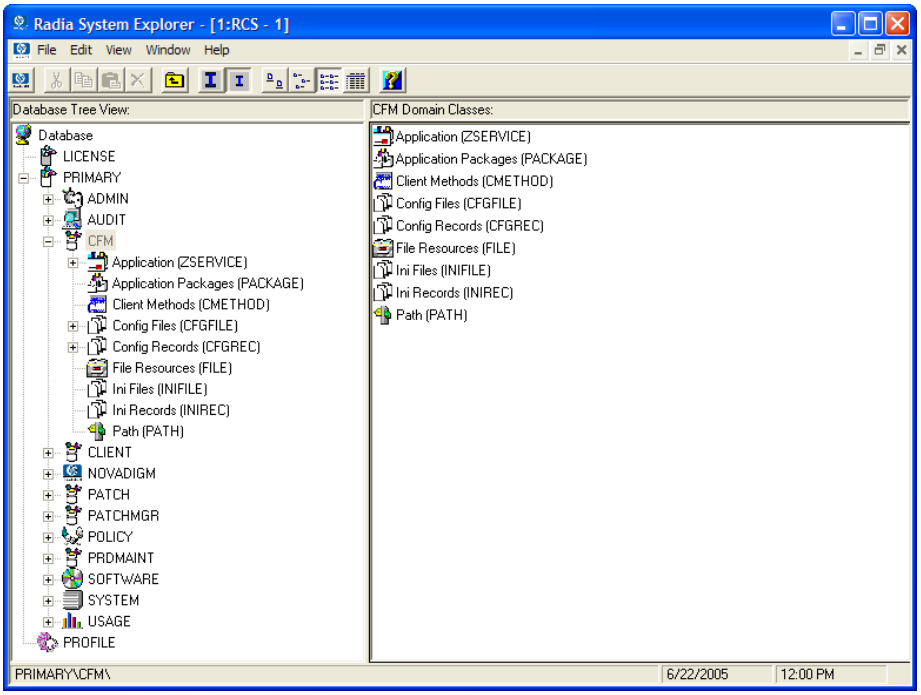
See About the Configuration File Management Domain on page 51 for more information about the CFM Domain.

See Managing Initialization (INI) Files on page 15 and Managing Other Types of Configuration Files on page 35 for examples of using the CFM Domain to manage configuration files.

To update the Radia Configuration Server database on UNIX

1. Stop the **Radia Configuration Server** service.
2. Copy CFM.xpc, CFM.xpi, and import_CFM.sh to the Radia Configuration Server's bin directory (by default `/opt/Novadigm/ConfigurationServer/BIN`).
3. Run import_cfm.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 might also see a return code of `4` also, which you may ignore.
4. Start the Radia Configuration Server service.
5. Open the System Explorer to review the CFM domain as shown below.



See About the Configuration File Management Domain on page 51 for more information about the CFM Domain.

See Managing Initialization (INI) Files on page 15 and Managing Other Types of Configuration Files on page 35 for examples of using the CFM Domain to manage configuration files.

3 Managing Initialization (INI) Files

Text files that follow the Windows-like initialization (INI) file format represent one type of configuration file that can be managed using Configuration File Management (CFM). These files are organized by sections that include keys and their associated values. Using CFM to manage these files simplifies the process of updating and managing the various entries that might be included in these file types.

This chapter provides examples of how to use Configuration File Management (CFM) to create and update the unattended answer file that can be used with the System Optional Components Manager executable program (sysocmgr.exe) for Windows. This program is used to add and remove components from a Windows system. The unattended answer file is used to determine which components should be installed, as well as the installation location for Windows applications. When used with sysocmgr.exe, this unattended file serves the same purpose as when using the **Add/Remove Windows Components** in the **Add or Remove Programs Control Panel**. This file's format conforms to INI file standards, containing sections, keys and values.

In our example, you will see how the file can be managed by CFM, as well as how the ZCREATE and ZUPDATE attributes defined for ZSERVICE can be used to carry out the sysocmgr.exe actions described by the unattended answer file.

Refer to *Installation and Configuration Guide for HP OpenView Application Manager Using Radia (Application Manager Guide)* for more information about ZSERVICE.

Use of the INIFILE and INIREC classes described in this chapter are limited to (Windows-like) INI file formats.

To describe how Configuration File Management can be used to manage INI files, this chapter includes the following sections:

- Setting Up the INI File
- Creating an INI File
- Modifying an INI File Entry
- Adding an INI File Entry
- Deleting an INI File Entry

Setting Up the INI File

Whether you wish to use Configuration File Management (CFM) to create and then manage one or more configuration files or only to manage existing files, the first thing you will want to do is to create an instance in **PRIMARY.CFM.INIFILE** for the file or files you want to manage.

Creating an instance in **PRIMARY.CFM.INIFILE** specifies that you want CFM to manage a particular file instance. When you create the instance, you identify the output file in which your configuration changes will be stored and managed.

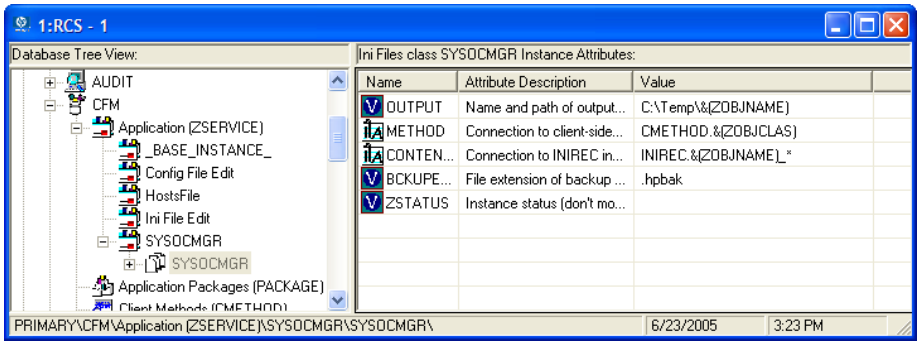
If you do not adjust the OUTPUT value, the output will be stored in the following default directory: `SystemDrive:\Program Files\Novadigm\Lib\username\radia\software\ZSERVICE\ServiceName` with a file name that matches the name of the INIFILE instance. See Table 1: INIFILE Class on page 52 for more information.

In this example, we will create a **SYSOCMGR** instance and an associated **SYSOCMGR ZSERVICE** instance.

To create a PRIMARY.CFM.INIFILE instance

1. Open the System Explorer.
2. Go to **PRIMARY.CFM.INIFILE**.
3. Create a new instance named **SYSOCMGR**.
4. Change the OUTPUT attribute to C:\TEMP\&(ZOBJNAME).

   The attributes in the **SYSOCMGR** instance should appear as follows:



Creating an INI File

You can use Configuration File Management to create your configuration files, if desired. In the following example, we will create the unattended answer file for sysocmgr.exe to include the following Components and Terminal Services sections.

A value of **on** installs the component, and a value of **off** prevents the component from being installed.

```
[Components]
netoc=off
TSEnable=on
TerminalServer=on

[Terminal Services]
ApplicationServer=1
PermissionsSetting=1
```

Each of the unattended answer file attributes is described in the following table.

Table 1: Unattended Answer File Attibutes

Section	Key	Description
Components	netoc	Installs additional optional networking components. If netoc=off, the [NetComponents] section is not processed.
Components	TSEnable	Enables Terminal Services.
Components	TerminalServer	Installs Terminal Services.
TerminalServices	ApplicationServer	Used to specify whether or not you wish the file Terminal Server Services to function as an Application Server (1) or as a Remote Administration Server (0).
TerminalServices	PermissionsSetting	Allows administrators to choose the user permissions required to run applications using Terminal Services. A value of **1** allows users to run applications that require additional

Section	Key	Description
		permissions not generally afforded to users on Windows 2000 and later. A value of **0** runs the environment using Windows 2000 and later compatible security.

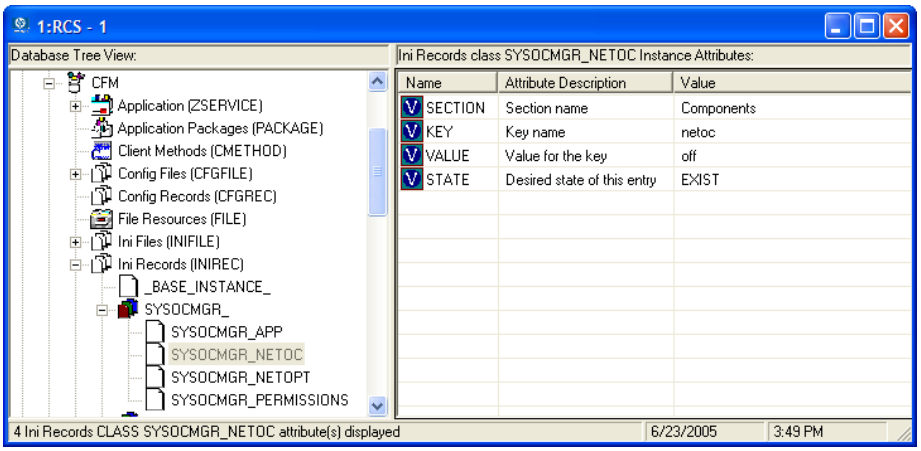
To create an INI file

These instructions assume you have followed the steps described in Setting Up the INI File on page 16.

1. Open the System Explorer.
2. Go to **PRIMARY.CFM.INIREC**.
3. Create a new instance named **SYSOCMGR_NETOC**. See About Naming Conventions on page 56.
4. Open the new instance (**SYSOCMGR_NETOC**) and specify the following values.

Attribute	Example Value
SECTION	Components
KEY	netoc
VALUE	off
STATE	EXIST

Once the example values have been entered, the attributes in the **SYSOCMGR_NETOC** instance should appear as follows:



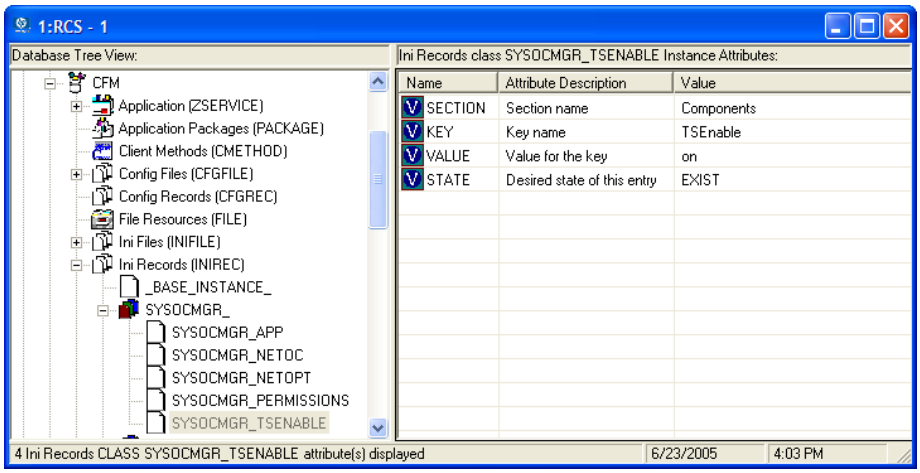
5 Go to **PRIMARY.CFM.INIREC**.

6 Create a new instance named **SYSOCMGR_TSENABLE**.

7 Open the new instance (**SYSOCMGR_TSENABLE**) and specify the following values.

Attribute	Example Value
SECTION	Components
KEY	TSEnable
VALUE	on
STATE	EXIST

Once the example values have been entered, the attributes in the **SYSOCMGR_TSENABLE** instance should appear as follows:



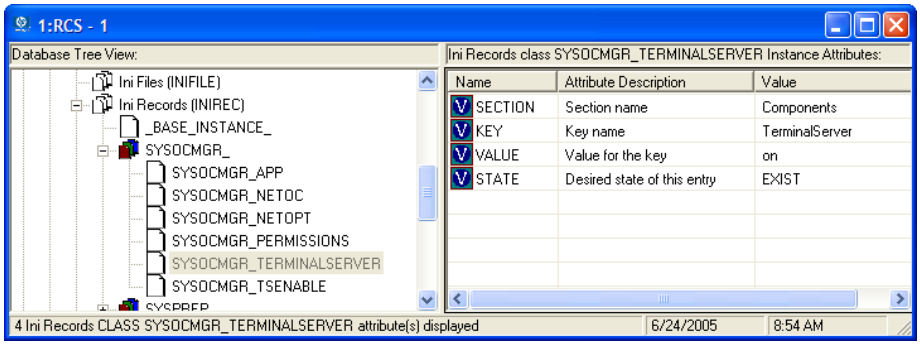
8 Go to **PRIMARY.CFM.INIREC**.

9 Create a new instance named **SYSOCMGR_TERMINALSERVER**.

10 Open the new instance (**SYSOCMGR_TERMINALSERVER**) and specify the following values.

Attribute	Example Value
SECTION	Components
KEY	TerminalServer
VALUE	on
STATE	EXIST

Once the example values have been entered, the attributes in the **SYSOCMGR_TERMINALSERVER** instance should appear as follows:



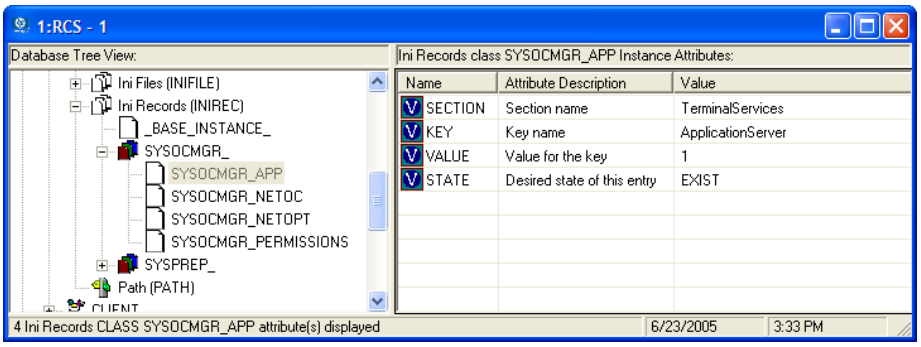
11 Go to **PRIMARY.CFM.INIREC**.

12 Create a new instance named **SYSOCMGR_APP**.

13 Open the new instance and specify the following values:

Attribute	Example Value
SECTION	TerminalServices
KEY	ApplicationServer
VALUE	1
STATE	EXIST

Once the example values have been entered, the attributes in the **SYSOCMGR_APP** instance should appear as follows:



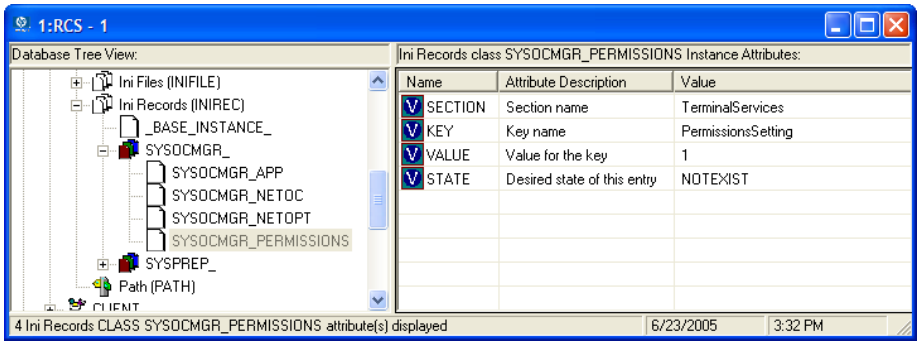
14 Go to **PRIMARY.CFM.INIREC**.

15 Create a new instance named **SYSOCMGR_PERMISSIONS**.

16 Open the new instance and specify the following values:

Attribute	Example Value
SECTION	TerminalServices
KEY	PermissionsSetting
VALUE	1
STATE	EXIST

Once the example values have been entered, the attributes in the **SYSOCMGR_PERMISSIONS** instance should appear as follows:



17 Go to **PRIMARY.CFM.ZSERVICE** and create a new ZSERVICE instance called **SYSOCMGR**. Use **SYSOCMGR** for the display name and for the instance name.

18 Configuration setting changes will be made through the Application Manager client. Therefore, open the **SYSOCMGR** ZSERVICE and add **INIFILE.SYSOCMGR** to any available _ALWAYS_ connection attributes.

Refer to *Installation and Configuration Guide for HP OpenView Application Manager Using Radia* (*Application Manager Guide*) for more information about ZSERVICE configuration.

19 To run the `sysocmgr.exe` file when the **SYSOCMGR** service is installed, add the following entry to ZCREATE and ZUPDATE.

```
C:\windows\system32\sysocmgr.exe
/i:c:\windows\inf\sysoc.inf /u:c:\temp\SYSOCMGR /r
```

where:

`/i` references the optional components information file

`/u:` specifies the unattended answer file location

`/r` supresses the reboot after installation

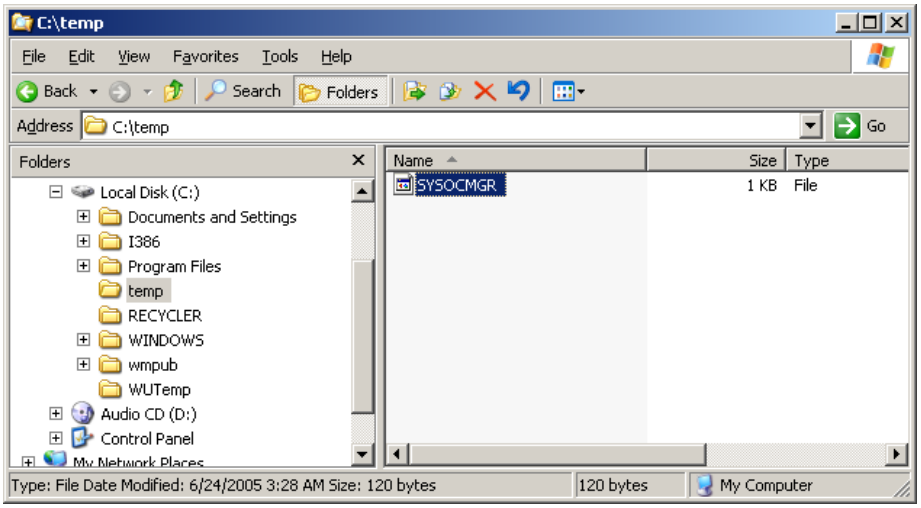
20 Connect the service to policy. For this example, you can connect the service to **POLICY.WORKGROUP.DEFAULT**.

Refer to the *Application Manager Guide* for more information.

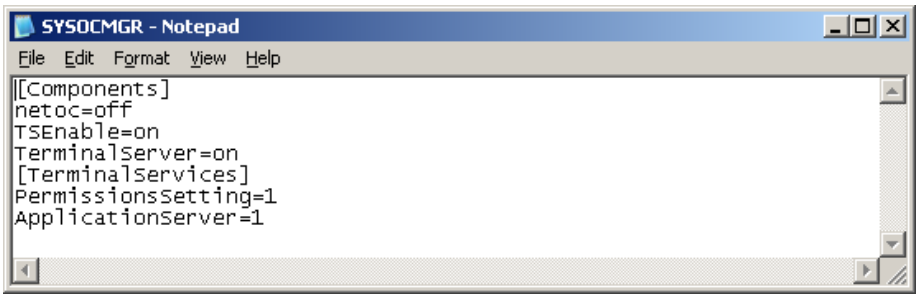
21 For this example, open a command prompt and run **`"c:\program files\novadigm\radskman.exe" dname=software`** to run an Application Manager connect to Radia.

22 Go to `C:\Temp`.

The `SYSOCMGR` file was created in the `C:\Temp` directory because we specified an OUTPUT value when creating the **SYSOCMGR** instance. If you accepted the default OUTPUT value, the file would be located in the following default directory: *`SystemDrive`*`:\Program Files\Novadigm\Lib\`*`username`*`\radia\software\ZSERVICE\SYSOCMGR`. See Table 1 on page 49 for more information.



23 Use a text editor to open the `SYSOCMGR` file. Below is an example of what the `SYSOCMRG` file that we created in this example looks like.

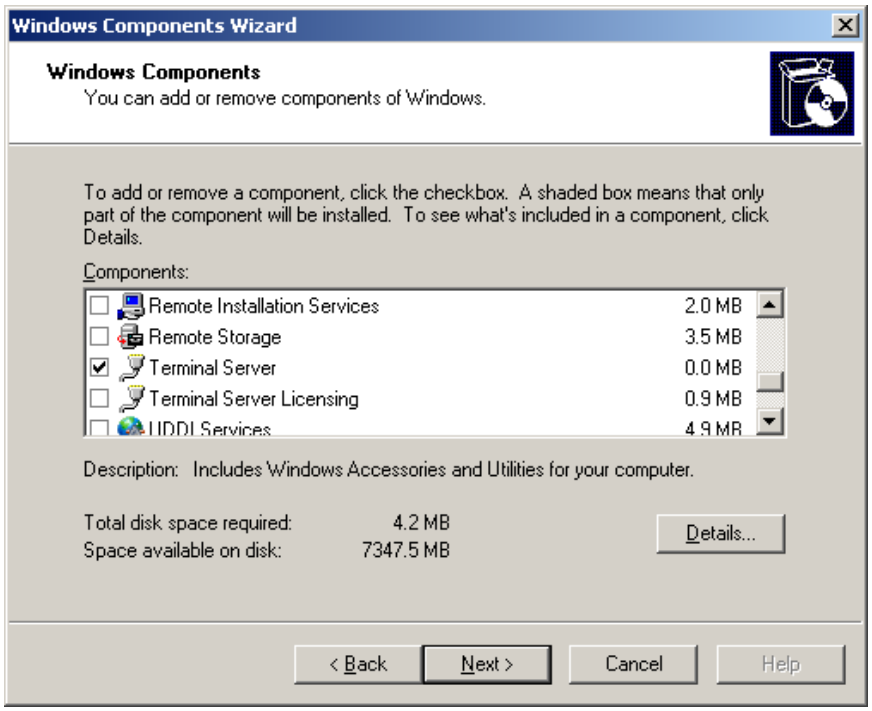


24 A complete install of Terminal Server requires a reboot, but you can verify it has been installed:

a Open the **Add or Remove Programs Control Panel.**

b Select **Add/Remove Windows Components**.

c Scroll down in the list to see the entry for **Terminal Server** is now selected.

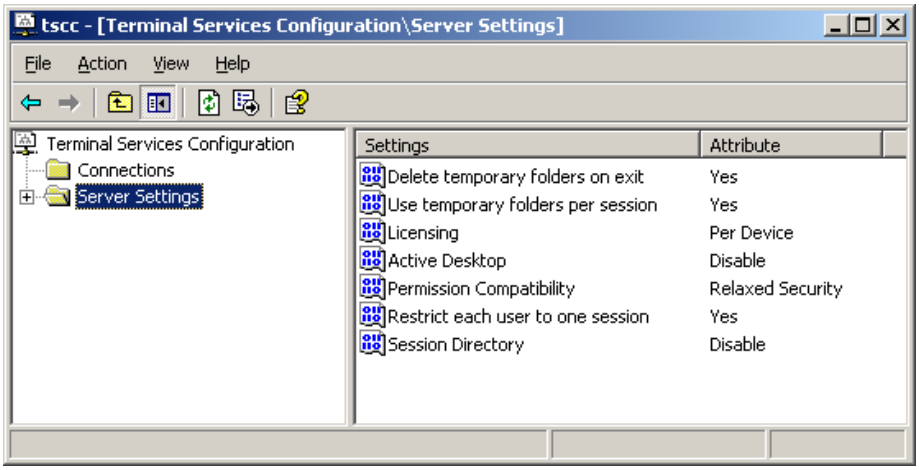


25 To verify that the Terminal Server Services is now functioning as an Application Server and that Permissions have been adjusted as expected:

a Go to **Administrative Tools Control Panel** and open the **Terminal Services Configuration**.

b Click **Server Settings**.

c Verify that **Licensing** is set to **Per Device** (rather than **Remote Desktop for Administration**) and **Permission Compatibility** is set to **Relaxed Security** (instead of **Full Security**).



Modifying an INI File Entry Entry

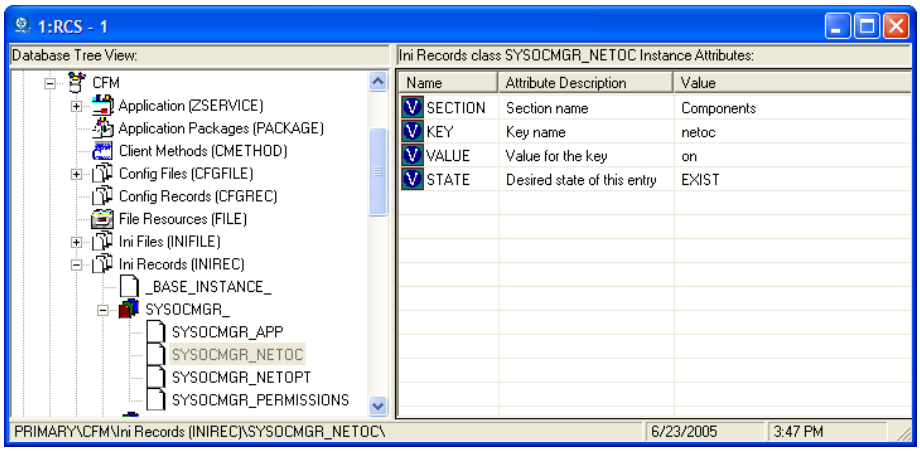
The real power of Configuration File Management (CFM) is in its ability to modify, insert, and remove information from configuration files. This allows you to control configurations without having to visit each machine and manually modify files.

In the following example, we will update **SYSOCMGR_NETOC** so that the netoc value is changed to **on**.

To update the INI file

1. Open the System Explorer.
2. Go to **PRIMARY.CFM.INIREC**.
3. Open the instance (**SYSOCMGR_NETOC**) and change the **netoc** value to **on** to enable the processing of optional network components in the unattended answer file as shown below.

   The attributes in the **SYSOCMGR_NETOC** instance should appear as follows:

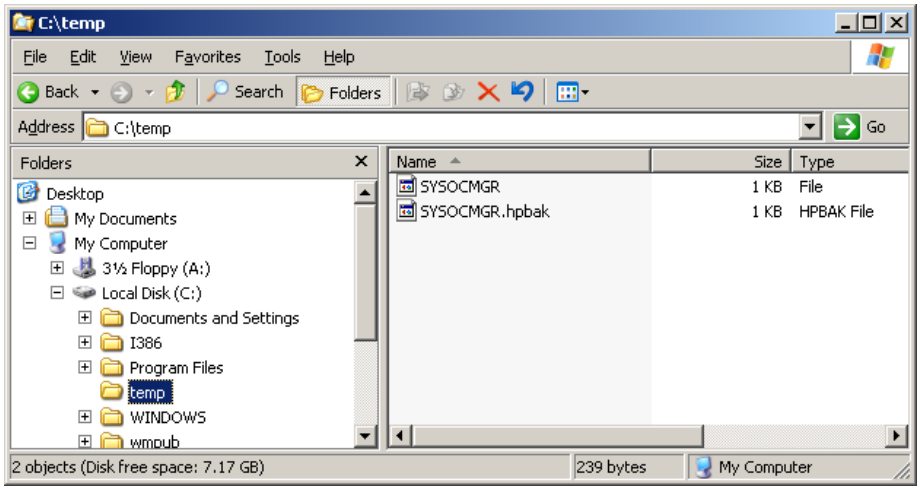


4 To update the file with the new value, open a command prompt and run an Application Manager connect to Radia:

```
"c:\program files\novadigm\radskman.exe" dname=software
```

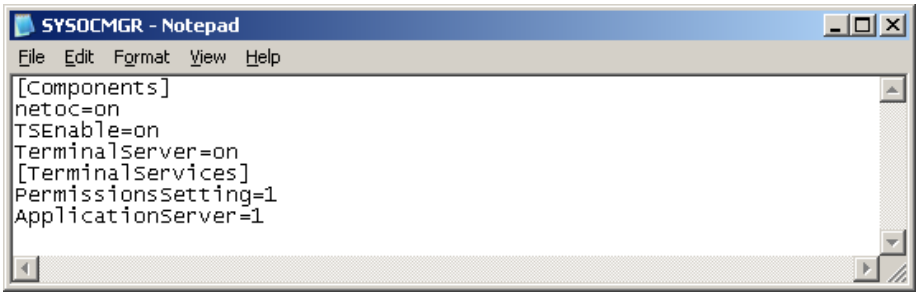
5 Go to `C:\Temp`.

The `SYSOCMGR` file is created in the `OUTPUT` directory value that was specified when creating the **SYSOCMRGR** instance. In our example the location is `C:\Temp\SYSOCMGR`.



As shown in the image above, a backup file is created when the file is modified. This file uses the `hpbak` extension by default. To change this value, modify the **BCKUPEXT** attribute when creating the **SYSOCMGR** instance. See Setting Up the INI File on page 16 for more information.

6 Use a text editor to open the `SYSOCMGR` file. Below is an example of what the `SYSOCMRG` file that we modified in this example looks like.



Adding an INI File Entry

In this example, because we enabled processing of optional network components in the unattended answer file by changing the netoc value to on, we will add the NetOptionalComponents section and its DHCPServer component to the unattended answer file to enable the server to be a DHCP Server.

To add an INI file entry

1 Open the System Explorer.

2 Go to **PRIMARY.CFM.INIREC**.

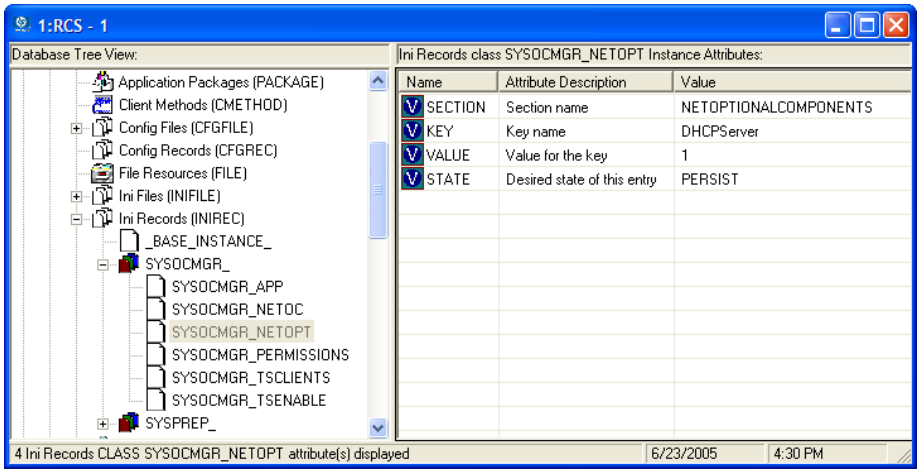
3 Create a new instance named **SYSOCMGR_NETOPT** and enter the values shown below.

Attribute	Example Value
SECTION	NETOPTIONALCOMPONENTS
KEY	DHCPServer
VALUE	1

Attribute	Example Value
STATE	PERSIST

The PERSIST value indicates that the DHCPServer will remain set to 1 even if the **SYSOCMGR** service is removed from the target server. See About Initialization (INI) Files on page 51 for more information about valid values for STATE.

Once the example values have been entered, the attributes in the **SYSOCMGR_NETOPT** instance should appear as follows:

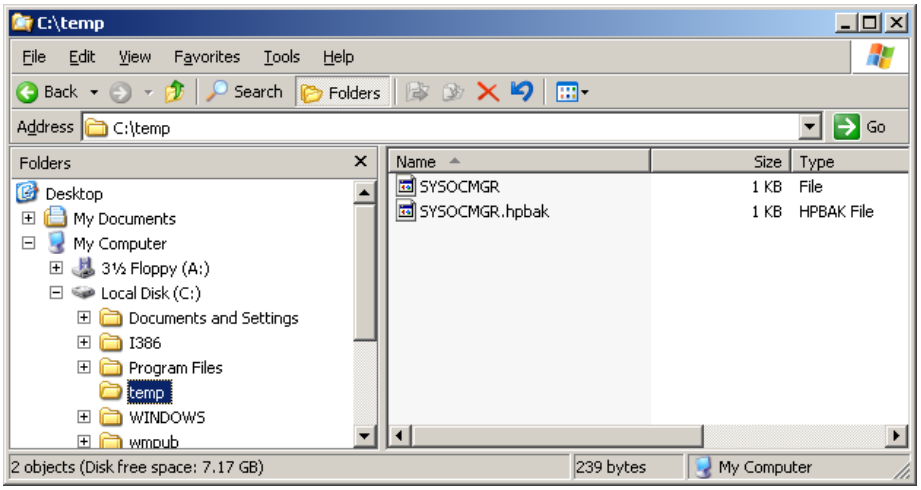


4 To update the file with the new value, open a command prompt and run an Application Manager connect to Radia:

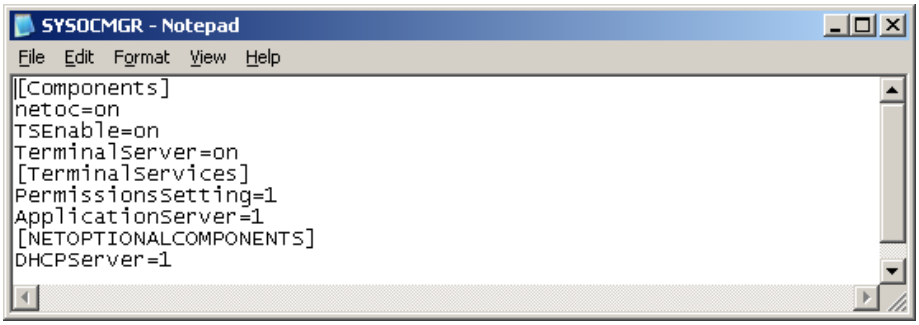
```
"c:\program files\novadigm\radskman.exe" dname=software
```

5 Go to `C:\Temp`.

The `SYSOCMGR` file is created in the OUTPUT directory value that was specified when creating the SYSOCMRGR instance. In our example the location is `C:\Temp\SYSOCMGR`.



6 Use a text editor to open the `SYSOCMGR` file. Below is an example of what the `SYSOCMRG` file that we modified in this example looks like.



7 To verify that the DHCP server is now enabled:

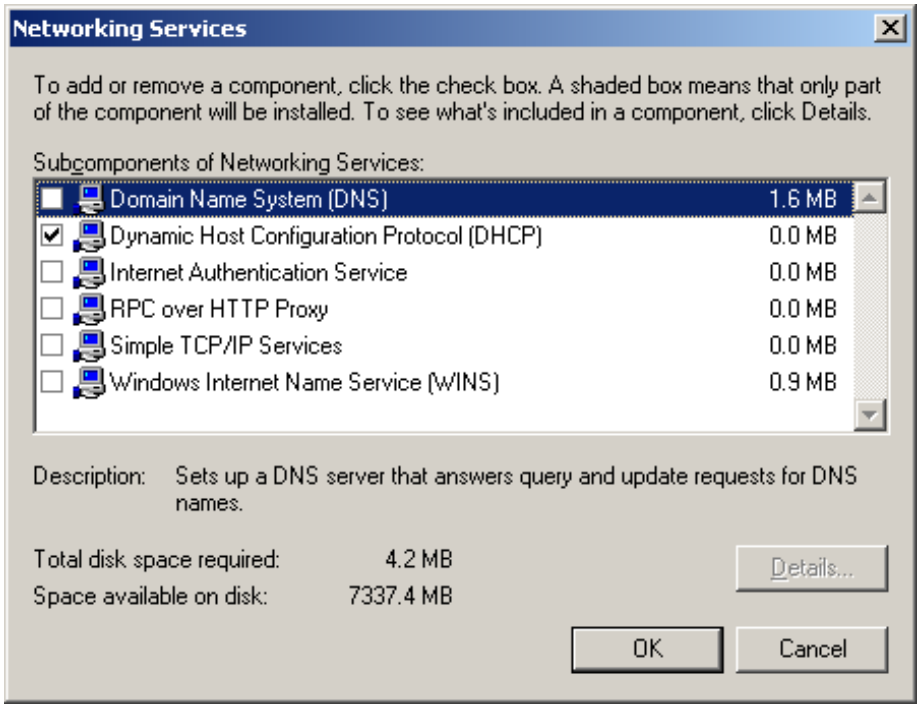
a Open the **Add or Remove Programs Control Panel.**

b Select **Add/Remove Windows Components**.

c Scroll down and highlight **Networking Services**.

d Click **Details** to see more specific information.

Dynamic Host Configuration Protocol (DHCP) should be selected in the list of networking components.



Deleting an INI File Entry

You can also use Configuration File Management (CFM) to delete file entries.

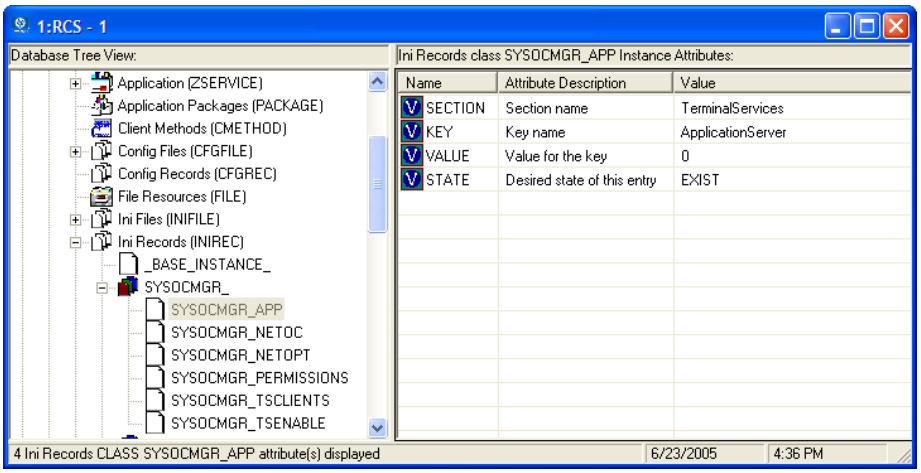
- If you wish to manage the entry, but do not want it to appear in the unattended answer file, set STATE to NOTEXIST.
- If you no longer wish to manage the file entry, delete the instance.

In this example, we will first change the ApplicationServer value to **0** and Terminal Server value to **off**, so that the Terminal Server Services will function as a Remote Administration Server. Then, because we no longer need to specify permissions, we will delete the PermissionsSetting key and value.

To first modify an INI file entry

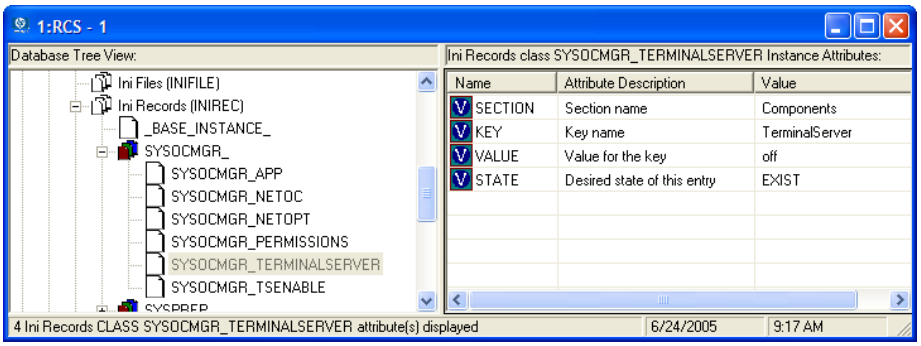
1. Open the System Explorer.
2. Go to **PRIMARY.CFM.INIREC**.
3. Open the instance **SYSOCMGR_APP** and change the ApplicationServer value to **0**.

The attributes in the **SYSOCMGR_APP** instance appear as follows:



4 Open the instance **SYSOCMGR_TERMINALSERVER** and change the ApplicationServer value to **`off`**.

The attributes in the **SYSOCMGR_TERMINALSERVER** instance appear as follows:



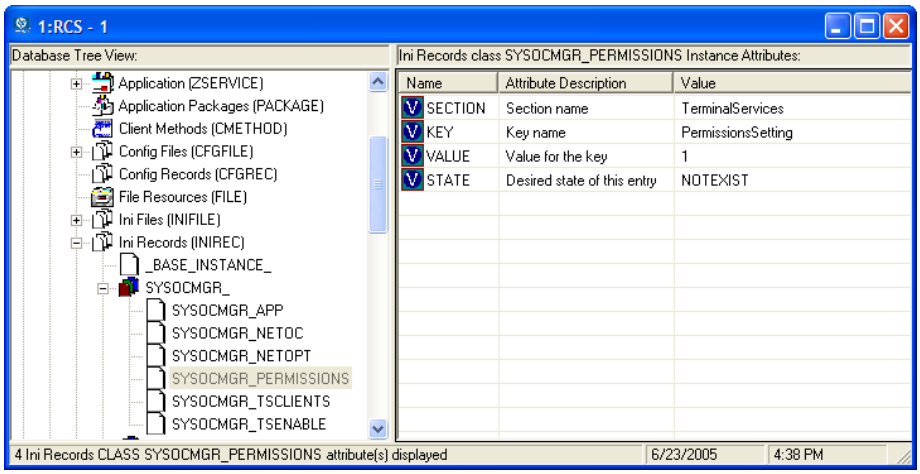
To delete an INI file entry

1 Go to **PRIMARY.CFM.INIREC**.

2 If you wish to continue to manage the file entry, open the instance named **SYSOCMGR_PERMISSIONS,** change the STATE value to NOTEXIST, and continue with the remaining instructions in this section.

If you no longer wish to manage the PermissionsSetting value, instead of changing the STATE value, you would delete the **SYSOCMGR_PERMISSIONS** instance.

Once the example values have been entered, the attributes in the **SYSOCMGR_PERMISSIONS** instance should appear as follows:

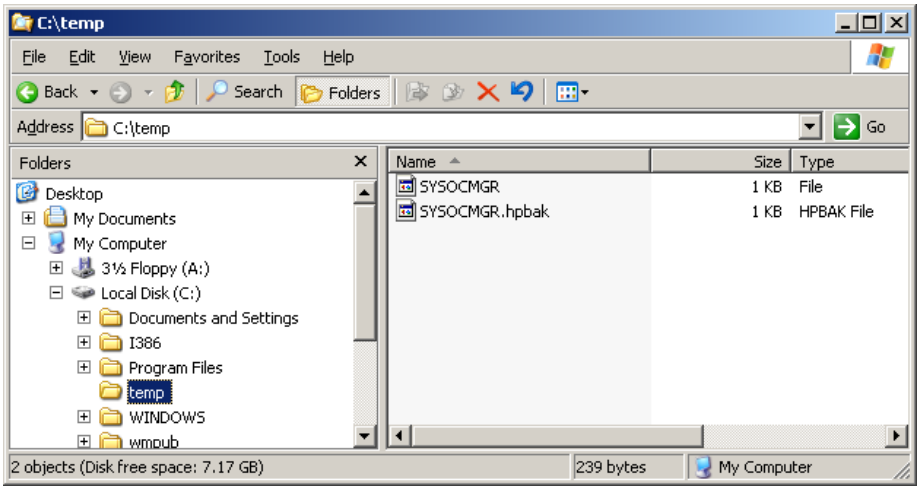


3 To update the file with the new value, open a command prompt and run an Application Manager connect to Radia:

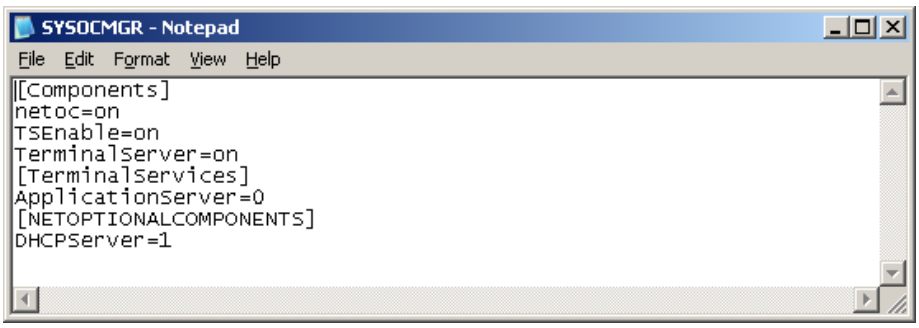
```
"c:\program files\novadigm\radskman.exe" dname=software
```

4 Go to C:\Temp.

The `SYSOCMGR_INIFILE` file is created in the OUTPUT directory value that was specified when creating the **SYSOCMRGR** instance. In our example the location is `C:\Temp\SYSOCMGR`.

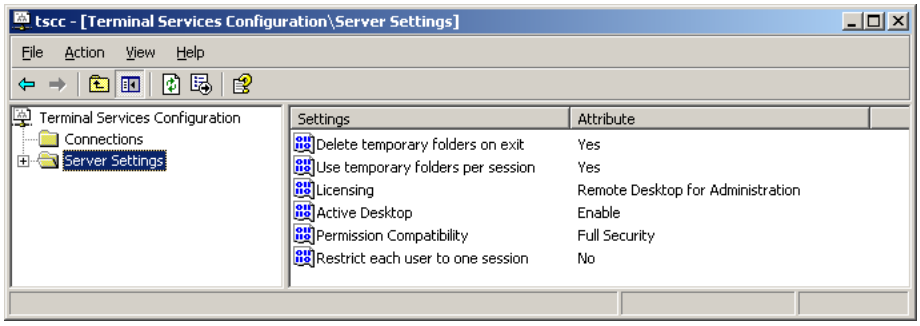


5 Use a text editor to open the `SYSOCMGR` file. Below is an example of what the `SYSOCMRG` file that we modified in this example looks like.



6 To verify that the Terminal Server Services is now running as a Remote Administration Server, the server will need to be rebooted.

7 After reboot, the Terminal Services Configuration Control Panel again reflects the settings for Remote Desktop Administration.



Review About Initialization (INI) Files on page 49. Then, you can experiment with creating and removing INI files, as well as manipulating entries.

4 Managing Other Types of Configuration Files

This chapter provides examples of how to use Configuration File Management to create and update a hosts file. Hosts files serve as an example of other types of configuration files that can be managed by Configuration File Management because their contents can be easily specified as column values.

To describe how Configuration File Management can be used to manage hosts files, this chapter includes the following sections:

- Setting Up the Hosts File
- Creating a Hosts File
- Modifying a Hosts File Entry
- Adding a Hosts File Entry
- Specifying the Location for a Hosts File Entry
- Deleting a Hosts File Entry

Setting Up the Hosts File

Whether you wish to use Configuration File Management (CFM) to create and then manage configuration files or to only manage existing files, the first thing you will want to do is to create an instance in **PRIMARY.CFM.CFGFILE** for the file or files to be managed.

Creating an instance in **PRIMARY.CFM.CFGFILE** specifies that you want CFM to manage a particular file instance. You can also identify the output file in which your configuration changes will be stored. If you do not provide an OUTPUT value, the output will be stored in the following default directory: `SystemDrive:\Program Files\Novadigm\Lib\username\radia \software\ZSERVICE\ ServiceName`. See Table 3 on page 51 for more information.

In this example, we will create a **HOSTSFILE** CFGFILE instance and an associated **HOSTSFILE** ZSERVICE instance.

To create a PRIMARY.CFM.CFGFILE instance

1 Open the System Explorer.

2 Go to PRIMARY.CFM.CFGFILE.

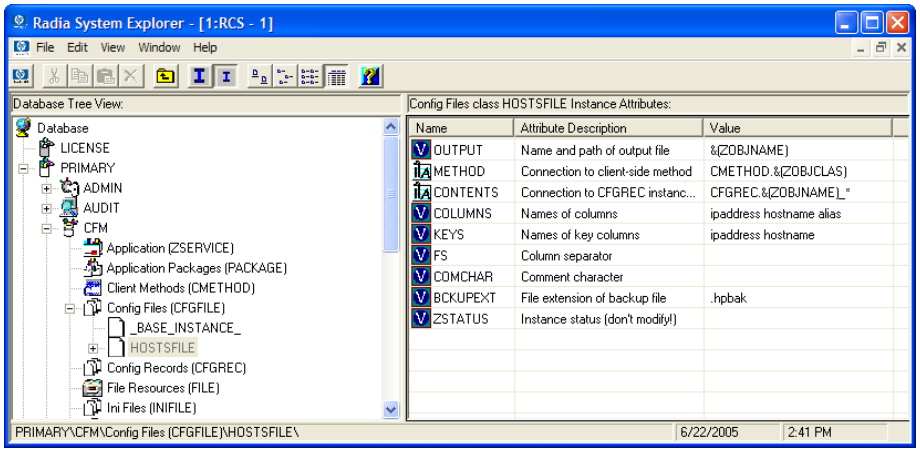
3 Create a new instance named **HOSTSFILE**.

4 Open the new instance and specify the following values:

Attribute	Value
COLUMNS	ipaddress hostname alias
KEYS	ipaddress hostname

For more information about the **CFGFILE** attributes, see About Configuration Files on page 51.

Once the example values have been entered, the attributes in the **HOSTSFILE** instance in the **CFGFILE** class should appear as follows:



Creating a Hosts File

Configuration File Management can be used to initially create configuration files if desired. In this example, we will add a web and an application server to the hosts file to expedite the resolution of these servers on the network:

```
10.10.10.1 www.internal.companya.usa.com EmployeeWeb

10.10.10.2 AppServer.companya.usa.com AppServer
```

On Microsoft Windows systems, hosts files are found in *SystemDrive*:\Windows\system32\drivers\etc\hosts. On Unix and Linux systems they are located in /etc/hosts. This example will use a sample Windows location throughout.

To create a hosts file

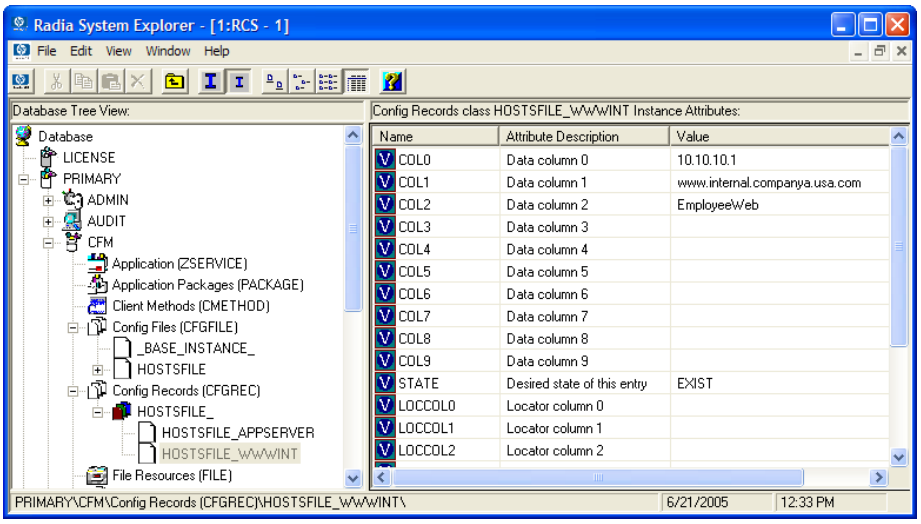
These instructions assume you have followed the steps described in Setting Up the Hosts File on page 35.

1. Go to **PRIMARY.CFM.CFGREC**.
2. Create a new instance named **HOSTSFILE_WWWINT**.
3. Open the new instance and specify the following values:

Attribute	Example Value
COL0	10.10.10.1
COL1	www.internal.companya.usa.com
COL2	EmployeeWeb
STATE	EXIST

For more information about the **CFGREC** attributes, see About Configuration Files on page 51.

Once the example values have been entered, the attributes in the **HOSTSFILE** instance in the **CFGREC** class should appear as follows:

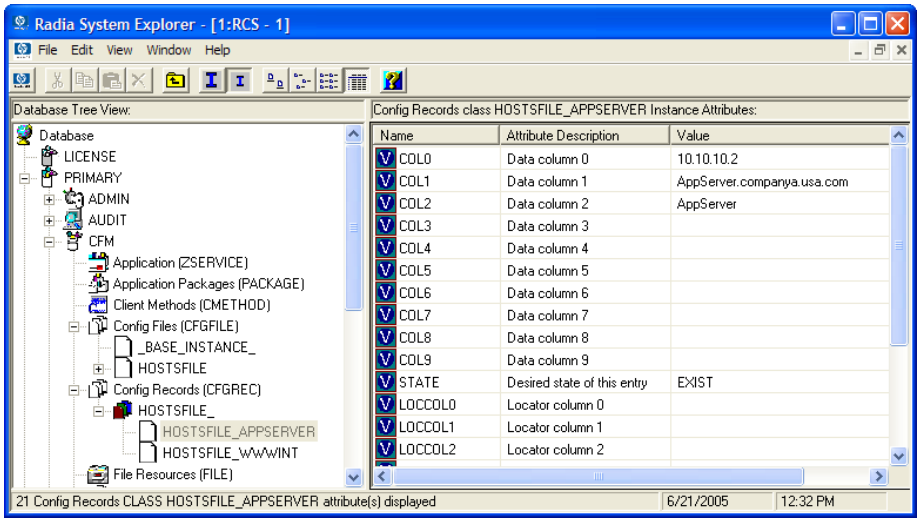


4 Create another instance named **HOSTSFILE_APPSERVER**.

5 Open the instance and specify the following values.

Attribute	Example Value
COL0	10.10.10.2
COL1	AppServer.companya.usa.com
COL2	AppServer
STATE	EXIST

Once the example values have been entered, the attributes in the **HOSTSFILE** instance in the **CFGREC** class should appear as follows:



6 Go to **PRIMARY.CFM.ZSERVICE** and create a new instance called **HOSTSFILE**. Use **HOSTSFILE** for the display name and **HOSTSFILE** for the instance name.

7 Open **ZSERVICE.HOSTSFILE** and add the value CFGFILE.HOSTSFILE to one of the available _ALWAYS_ attributes.

8 Refer to *Application Manager Guide for HP OpenView Application Manager Using Radia* (*Application Manager Guide*) for more information about ZSERVICE configuration.

9 Connect the service to policy. For this example, you can connect the service to **POLICY.WORKGRP.DEFAULT**.

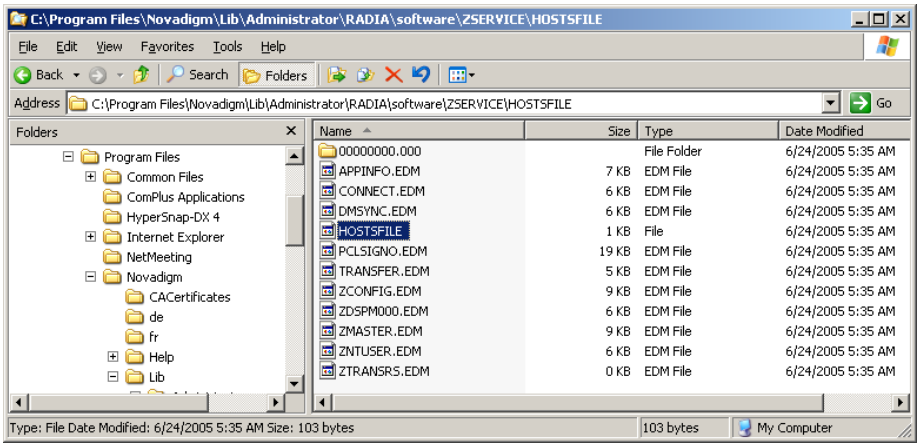
Refer to the *Application Manager Guide* for more information.

10 Open a command prompt and run "**c:\program files\novadigm\radskman.exe" dname=software** to run an Application Manager connect to Radia.

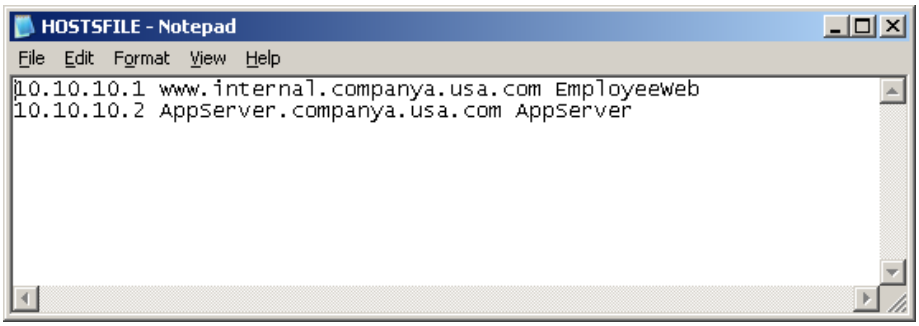
The HOSTSFILE file is created in the following default directory: *SystemDrive*:\Program Files\Novadigm\Lib\username\radia\software\ZSERVICE\HOSTSFILE.

If you would like to change the location where the file is stored, use the OUTPUT attribute as described in About Configuration Files on page 51.

11 Go to *SystemDrive*:\Program Files\Novadigm\Lib*username*\radia\software\ZSERVICE\HOSTSFILE.



12 Use a text editor to open HOSTSFILE. Below is an example of what the HOSTSFILE file that we created in this example looks like.



Modifying a Hosts File Entry

As mentioned previously, the real power of Configuration File Management (CFM) is in its ability to modify, insert, and remove information from configuration files. This allows you to control configurations without having to visit each machine and manually modify files.

In this example, we will perform a simple modification to our example hosts file that was created using CFM, by changing the alias for *www.internal.companya.usa.com* as follows:

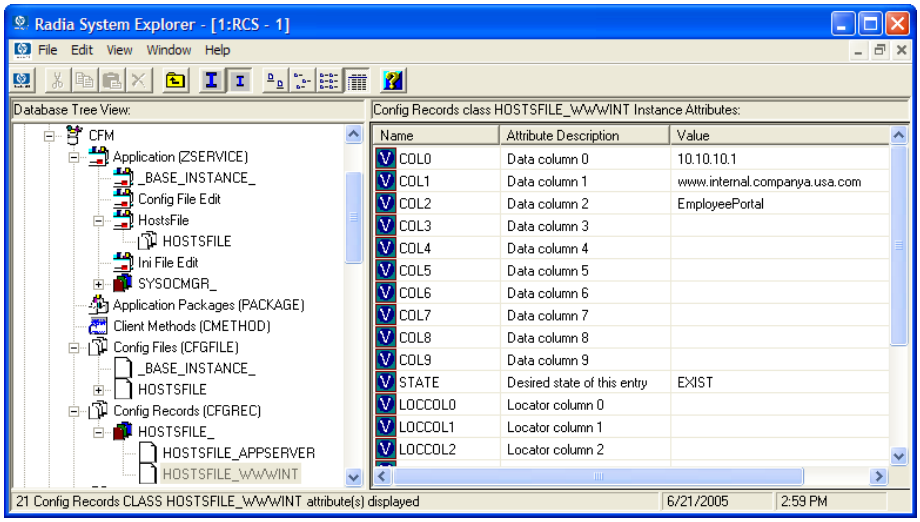
```
10.10.10.1 www.internal.companya.usa.com EmployeePortal
```

To modify a hosts file entry

1. Open the System Explorer.
2. Go to **PRIMARY.CFM.CFGREC**.
3. Open the instance named **HOSTSFILE_WWWINT** and change the following values:

Attribute	Original Value	New Value
COL2	EmployeeWeb	EmployeePortal

Once the COL2 value has been modified, the attributes in the **HOSTSFILE_WWINT** instance should appear as follows:



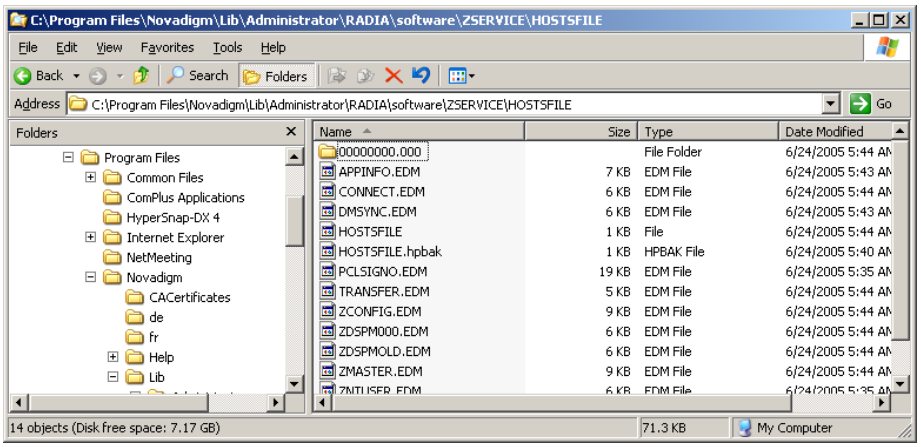
4 To update the file with the new value, open a command prompt and run an Application Manager connect to Radia:

"c:\program files\novadigm\radskman.exe" dname=software

The HOSTSFILE file is modified in the following default directory: *SystemDrive*:\Program Files\Novadigm\Lib*username*\radia\software\ZSERVICE\HOSTSFILE.

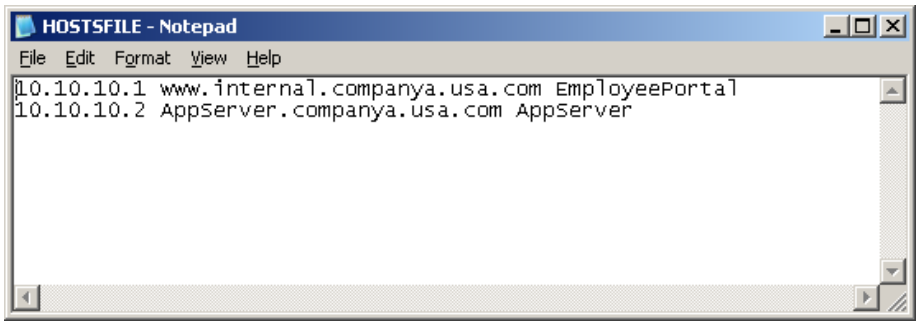
To change the location where the file is stored, use the OUTPUT attribute as described in About Configuration Files on page 51.

5 Go to *SystemDrive*:\Program Files\Novadigm\Lib*username*\radia\software\ZSERVICE\HOSTSFILE.



As shown in the image above, a backup file was created in this directory using the default `hpbak` extension. You can use the **BCKUPEXT** attribute to change the extension. See About Configuration Files on page 51 for more information.

6 Use a text editor to open `HOSTSFILE`. Below is an example of what the `HOSTSFILE` file that we modified in this example looks like.



Adding a Hosts File Entry

You might also want to use Configuration File Management (CFM) to add new entries to a configuration file. In this example, we add the following hosts file entry:

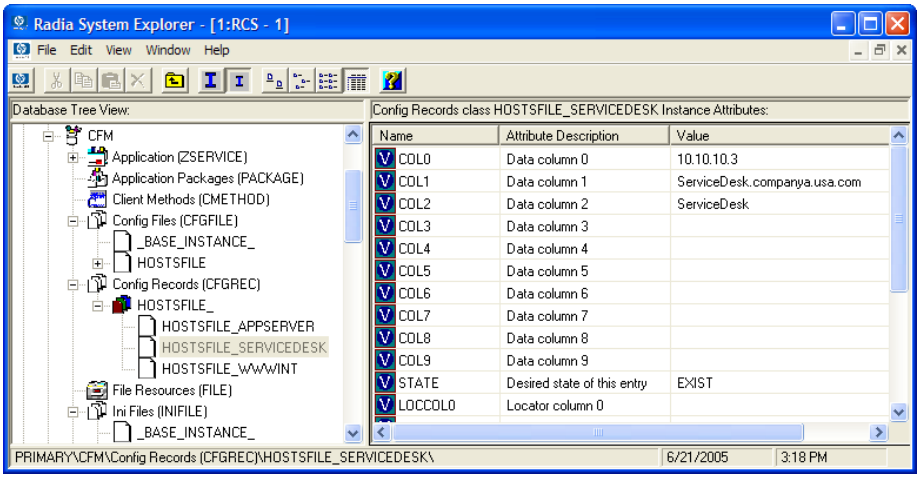
```
10.10.10.3 ServiceDesk.companya.usa.com ServiceDesk
```

To add a hosts file entry

1. Open the System Explorer.
2. Go to **PRIMARY.CFM.CFGREC**.
3. Create a new instance named **HOSTSFILE_SERVICEDESK.**
4. Open the new instance and specify the following values.

Attribute	Example Value
COL0	10.10.10.3
COL1	ServiceDesk.companya.usa.com
COL2	ServiceDesk
STATE	EXIST

5. Once the example values have been entered, the attributes in the **HOSTSFILE_SERVICEDESK** instance should appear as follows:



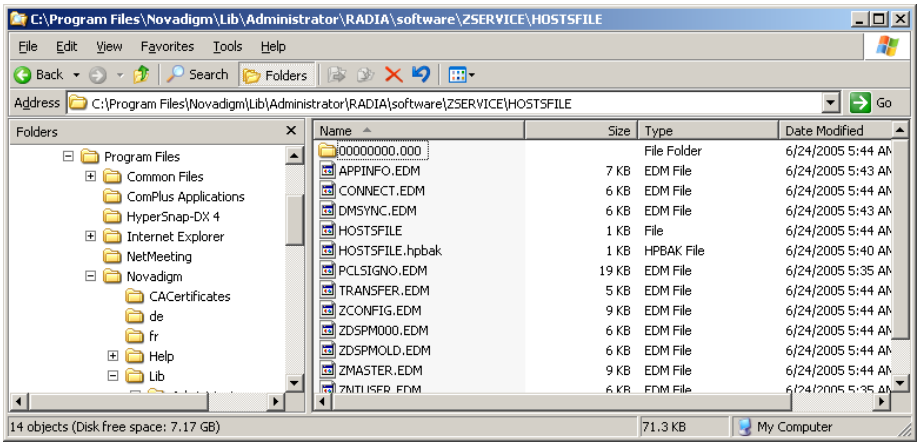
6 To update the file with the new value, open a command prompt and run an Application Manager connect to Radia:

"c:\program files\novadigm\radskman.exe" dname=software

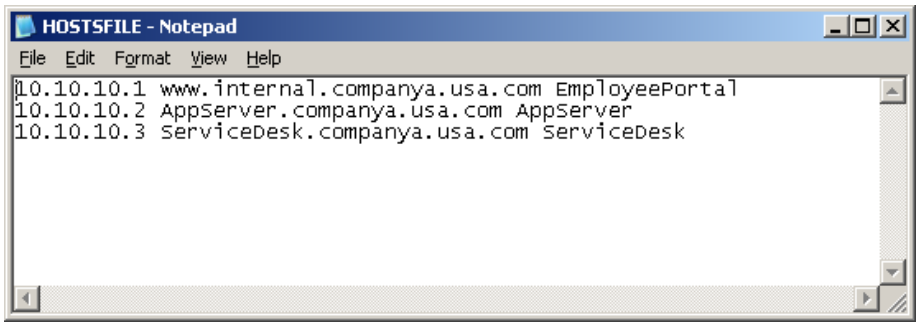
The HOSTSFILE file is modified in the following default directory: *SystemDrive*:\Program Files\Novadigm\Lib*username*\radia\software\ZSERVICE\HOSTSFILE.

To change the location where the file is stored, use the OUTPUT attribute as described in About Configuration Files on page 51.

7 Go to *SystemDrive*:\Program Files\Novadigm\Lib*username*\radia\software\ZSERVICE\HOSTSFILE.



8 Use a text editor to open `HOSTSFILE`. Below is an example of what the `HOSTSFILE` file that we modified in this example looks like.



Specifying the Location for a Hosts File Entry

You can also use Configuration File Management (CFM) to specify a location for a configuration file entry. For example, you might need to add a hostname to an existing IP address so that you can use a single machine to host multiple Web or ftp servers. In such cases, you can use CFM to update the hosts file, and to specify that you wish to keep this information with the rest of the entries for that IP address.

If you do not specify a location for the file entry, CFM appends the entry to the end of the file. For more information about the rules for updating configuration files, see About Configuration Files on page 57.

In this example, we will add an entry for a second hostname for the IP address 10.10.10.1.

```
10.10.10.1 ftp.internal.companya.usa.com InternalFTP
```

To specify the location of a hosts file entry:

1 Open the System Explorer.

2 Go to **PRIMARY.CFM.CFGREC**.

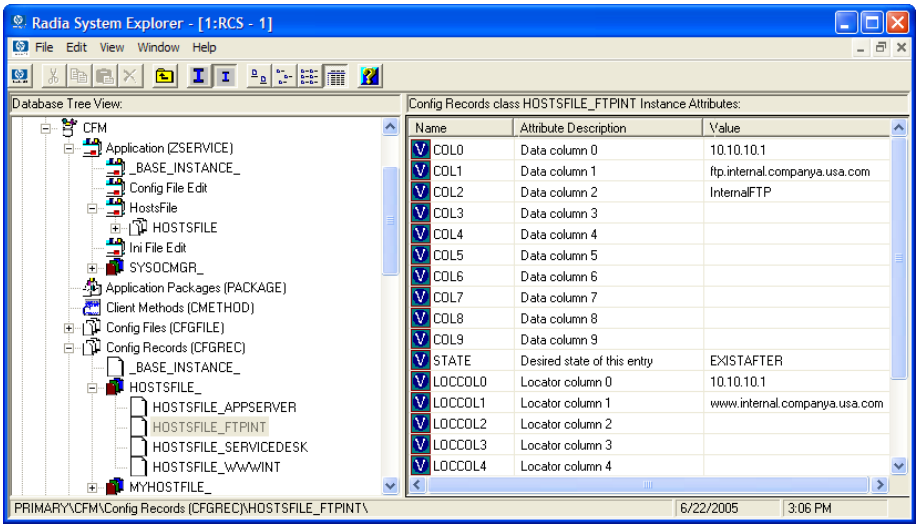
3 Create a new instance named **HOSTSFILE_FTPINT.**

4 Open the new instance and specify the following values.

Attribute	Example Value
COL0	10.10.10.1
COL1	ftp.internal.companya.usa.com

Attribute	Example Value
COL2	InternalFTP
STATE	EXISTAFTER
LOCCOL0	10.10.10.1
LOCCOL1	www.internal.companya.usa.com

5 Once the example values have been entered, the attributes in the **HOSTSFILE_FTPINT** instance should appear as follows:



In this example, CFM uses the **LOCCOL0** value (10.10.10.1) and **LOCCOL1** (www.internal.companya.usa.com) to locate the line entry to be used as a reference point for **EXISTAFTER**. Once CFM finds the 10.10.10.1 and www.internal.companya.usa.com entries, it inserts the new hostname entry for InternalFTP *after* it.

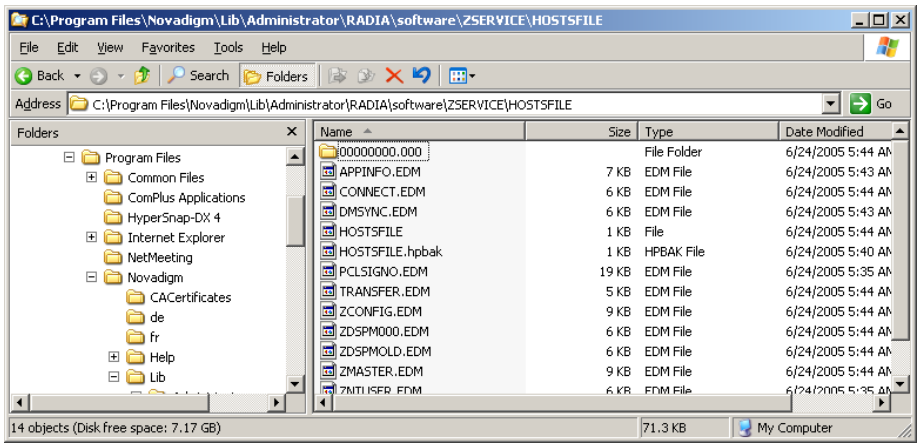
6 To update the file with the new value, open a command prompt and run an Application Manager connect to Radia:

`"c:\program files\novadigm\radskman.exe" dname=software`

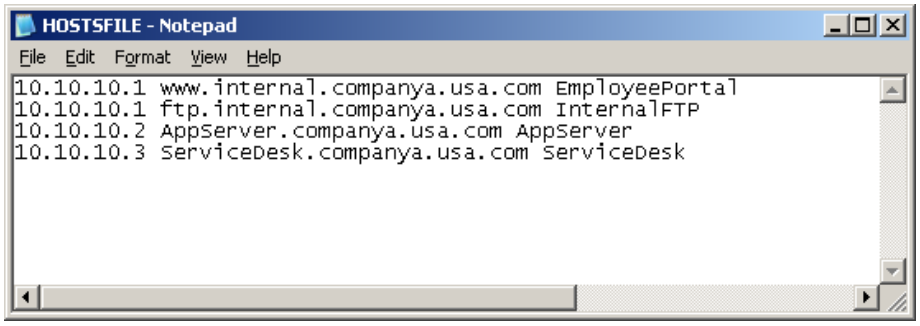
The HOSTSFILE file is modified in the following default directory: `SystemDrive:\Program Files\Novadigm\Lib\usernam e\radia\software\ZSERVICE\HOSTSFILE.`

To change the location where the file is stored, use the OUTPUT attribute as described in About Configuration Files on page 51.

7 Go to *SystemDrive*:\Program Files\Novadigm\Lib*username*\radia\software\ZSERVICE\HOSTSFILE.



8 Use a text editor to open HOSTSFILE. Below is an example of what the HOSTSFILE file that we modified in this example looks like. As specified using the **EXISTAFTER** attribute, the ftp.internal.companya.usa.com entry is inserted *after* the www.internal.companya.usa.com entry.



Deleting a Hosts File Entry

You can also use Configuration File Management (CFM) to delete file entries.

- If you wish to manage the entry, but do not want it to appear in the hosts file, set STATE to NOTEXIST.
- If you no longer wish to manage the file entry, delete the instance.

In this example, we delete the first entry for the internal web portal:

```
10.10.10.1 www.internal.companya.usa.com EmployeePortal
```

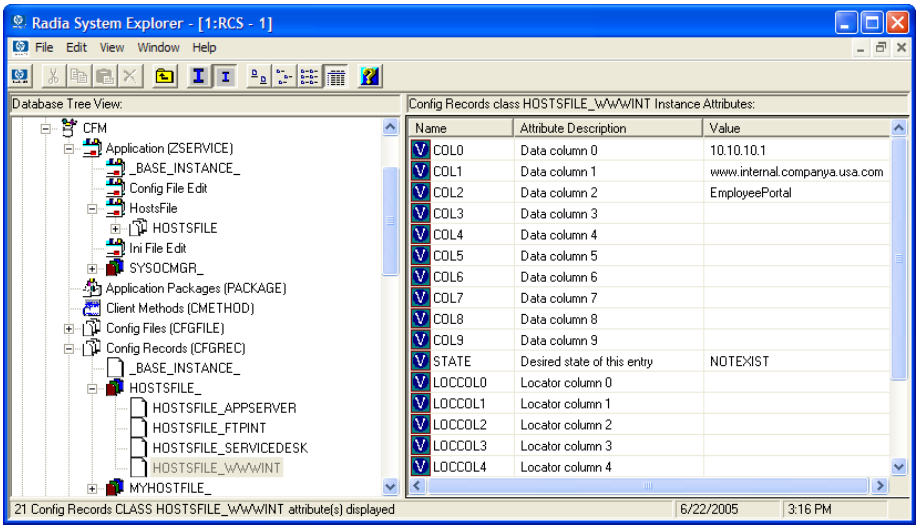
To delete a hosts file entry:

1 Go to **PRIMARY.CFM.CFGREC**.

2 If you wish to continue to manage the file entry, open the instance named **HOSTSFILE_WWWINT**, change the desired state of this entry value to NOTEXIST, and continue with the remaining instructions in this section.

> If you no longer wish to manage the EmployeePortal file entry, instead of changing the desired state of this entry, you would delete the **HOSTSFILE_WWWINT** instance.

3 Once the example values have been entered, the attributes in the **HOSTSFILE_WWWINT** instance should appear as follows.



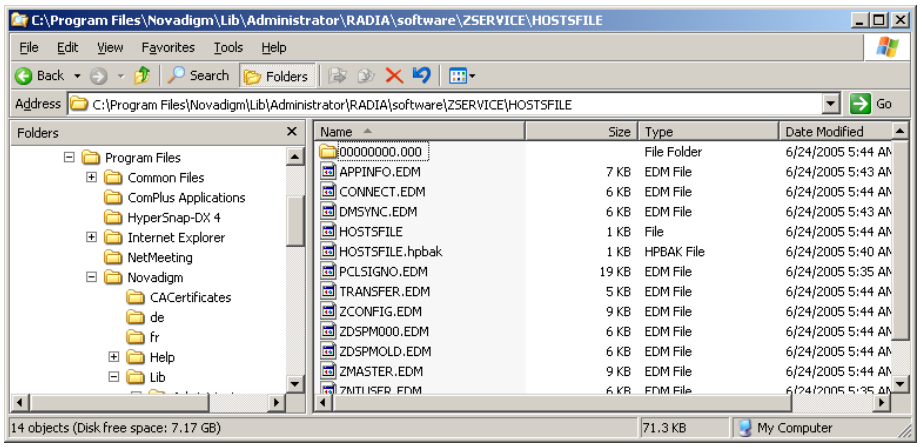
4 To update the file with the new value, open a command prompt and run an Application Manager connect to Radia:

```
"c:\program files\novadigm\radskman.exe" dname=software
```

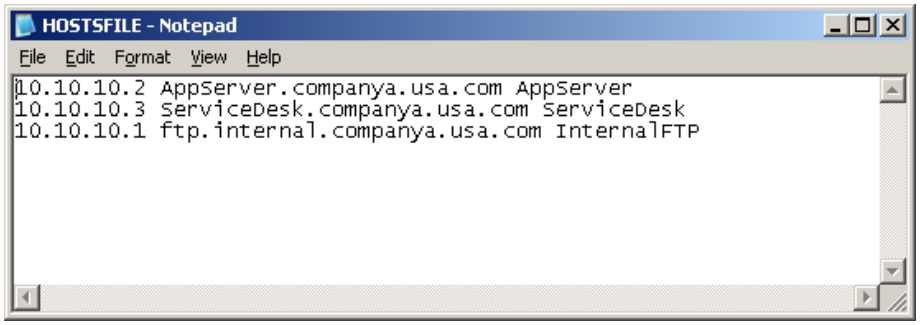
The `HOSTSFILE` file is modified in the following default directory: *`SystemDrive`*`:\Program Files\Novadigm\Lib\`*`username`*`\radia\software\ZSERVICE\HOSTSFILE`.

To change the location where the file is stored, use the OUTPUT attribute as described in About Configuration Files on page 51.

5 Go to *SystemDrive*:\Program Files\Novadigm\Lib*username*\radia\software\ZSERVICE\HOSTSFILE.



6 Use a text editor to open HOSTSFILE. Below is an example of what the HOSTSFILE file that we modified in this example looks like.



Review About Configuration Files on page 51. Then, you can experiment with creating and removing INI files, as well as manipulating entries.

5 About the Configuration File Management Domain

The CFM domain contains the classes and instances necessary to manage your configuration files.

About Initialization (INI) Files

The Radia Database contains two classes to manage INI files—INIFILE Class (INIFILE) and INIREC Class (INIREC).

Use the INIFILE Class (INIFILE) to specify the INI file to manage.

Table 1: INIFILE Class

Attribute	Default Value	Description
OUTPUT	&(ZOBJNAME)	Name of the output file. If you do not modify this value, the output will be stored in `SystemDrive:\Program Files\Novadigm\Lib\username\radia\software\Zservice\InstanceName` If you do specify the output for the file, be sure to follow these rules to prevent failures: • If you specify a file without an extension (for example, `C:\Temp`) make sure that there is not an existing directory with the same name in that location.
METHOD	CMETHOD.&(ZOBJLCAS)	Connection to Method instance. Do not modify.

Attribute	Default Value	Description
CONTENTS	INIREC.&(ZOBJNAME)_*	Connection to INIREC instances. This attribute connects any INIREC instances that begin with the same name as the INIFILE instance, modified with an *_Identifier*. Do not modify.
BCKUPEXT	.hpbak	The file extension for backup files. If this attribute is defined, a backup will be created each time changes are made to the original file. If this attribute is left blank, no backup is created.
ZSTATUS	N/A	*Do not modify this attribute.* Required by the Radia client to remember the install state of the object.

Use INI File Record (INIREC) class to specify the changes to be made to the INI file.

Table 2: INIREC class

Attribute	Default Value	Description
SECTION	N/A	Name of the section you wish to add or update. For example: SETTINGS
KEY	N/A	Name of the key you wish to add or update. For example: TIMEOUT
VALUE	N/A	Value to be associated with the KEY attribute. For example: 1
STATE	EXIST	The desired state for the entry. You can enter one of the following: • **EXIST** – ensures the entry exists in the file specified in the INIFILE instance. If the service is removed, the entries will also be removed. • **NOTEXIST** – ensures that the entry does not exist in the file specified in the INIFILE instance. • **PERSIST** – ensures that the entry exists in the file and that the entry will remain in the file even if the service is

Attribute	Default Value	Description
		removed. *Note that the file remains only if the configuration file is created outside of Radia's LIB directory or if it includes entries not being managed by CFM.*

About Configuration Files

The Radia Database contains two classes to manage Configuration files—CFGFILE class (CFGFILE) and CFGREC class (CFGREC).

Use the CFGFILE class (CFGFILE) to describe the structure of the source file and its characteristics.

Table 3: CFGFILE Class

Attribute	Default Value	Description
OUTPUT	&(ZOBJNAME)	Name of the output file. If you do not modify this value, the output will be stored in `SystemDrive:\Program Files \Novadigm\Lib\username\radia \software\Zservice\ServiceName` If you do specify the output for the file, be sure to follow these rules to prevent failures: • If you specify a file without an extension (for example, `C:\Temp`) make sure that there is not an existing directory with the same name in that location.
METHOD	CMETHOD.&(ZOBJLCAS)	Connection to Method instance. Do not modify.
CONTENTS	CFGREC.&(ZOBJNAME)_*	Connection to CFGREC instances. Do not modify.

Attribute	Default Value	Description
COLUMNS	N/A	The column names used in the configuration file. These column names are used to specify the types of values and the order in which they appear in the configuration file. For example: ipaddress hostname alias
KEYS	N/A	Columns that uniquely identify a specific line in the file so it can be located. Do not include a KEY attribute value for those columns whose values you wish to modify or manage. CFM uses the KEY values to determine whether or not the entry exists. For example: ipaddress hostname
FS	A single space.	Field separator. If you do not specify a value for the column separator, a space or tab will be recognized as a separator. Common values for separators include a space (for either a space or **`Tab`**), **`\t`** (for Tab only) and a comma (,).
COMCHAR	#	Comment character
BCKUPEXT	.hpbak	Allows you to specify the file extension for backup files. If this attribute is defined, a backup will be created each time changes are made to the original file. If this attribute is left blank, no backup is created.
ZSTATUS	N/A	*Do not modify this attribute.* Required by the Radia client to remember the install state of the object.

Use the Config Record (CFGREC) to manage the records in the configuration file.

Table 4: CFGREC class

Attribute	Default Value	Description
COL0…COL9	N/A	Values for the columns that were specified in the CFGFILE class. For example, if the columns are: ipaddress hostname alias You might specify the following values: COL0 = 10.128.131.1 COL1 = www.internal.companya.usa.com COL2= EmployeePortal
STATE	EXIST	The desired state for the entry. You can enter one of the following: • **EXIST** – ensures the entry exists in the file specified in the CFGFILE instance. If the service is removed, the entries will also be removed. • **EXISTBEFORE** – ensures the entry exists *before* the first line that matches the specified reference location. If no line matches the reference location, the line will be added to the end of the file. If the service is removed, the entries will also be removed. • **EXISTAFTER** – ensures the entry exists *after* the first line that matches the specified reference location. If the service is removed, the entries will also be removed. • **NOTEXIST** – ensures the entry does not exist in the file specified in the CFGFILE instance. • **PERSIST** – ensures that the entry exists in the file and that

Attribute	Default Value	Description
		the entry will remain in the file even if the service is removed. *Note that the file remains only if the configuration file is created outside of Radia's LIB directory or the file contains records not being managed by CFM..* • **PERSISTBEFORE** - ensures that the entry exists *before* the first line that matches the specified reference location and that the entry will remain in the file even if the service is removed. *Note that the file remains only if the configuration file is created outside of Radia's LIB directory or the file contains records not being managed by CFM..* • **PERSISTAFTER** - ensures that the entry exists *after* the first line that matches the specified reference location and that the entry will remain in the file even if the service is removed. *Note that the file remains only if the configuration file is created outside of Radia's LIB directory or the file contains records not being managed by CFM.*
LOCCOL0…LOCCOL9	N/A	Value used to locate a line within the file that is then used as a reference point for EXISTBEFORE, EXISTAFTER, PERSISTBEFORE or PERSISTAFTER. You can use regular expressions to match column entries without specifying an actual value. For example, if you wanted to check that a valid IP address appears in the first column, you might enter:

Attribute	Default Value	Description
		`([0-9]+).([0-9]+).([0-9]+).([0-9]+)` for LOCCOL0. As another example, if you wanted to ensure that the host name in the second column ends with hp.com you would enter: `[^.]*.hp.com` for LOCCOL1. ^. means “everything except a period” * means “any number of characters” This way, the hostname can be any number of characters, but cannot contain a period (.) and will end in .hp.com. See on page About Regular Expressions on page 59 for more information.

About Naming Conventions

When creating INIREC instances, it is recommended that you use the naming convention as follows:

```
INIFILEINSTANCENAME_SECTIONNAME_KEYNAME
```

This allows you to quickly determine which instance you want to modify. It also ensures that the default connection set in the INIFILE class `(INIREC&(ZOBJNAME)_*)` will work.

This same convention is recommended for use with CFGREC instances.

Rules for Updating Configuration Files

You may want a good understanding of how the configuration files are updated based on the information you specify in the Radia Database. Below are the rules used during updates to INI files.

- If the section does not exist, it will be added to the end of the file and the key=value line will be added after that.
- If the section exists and the key exists in the section, the value will be updated.
- If the section exists and the key exists in the section multiple times, the value will be updated in the first occurrence of the key.
- If the section exists, but the key is not present, the key=value line will be added to the section after the last key-value line.
- If the section is present multiple times, the changes are done in the first occurrence.

  Key and section name comparisons for INI files are *not* case sensitive.

- All lines that are not affected by the update will be preserved as is.

The only difference between the rules for INI files and configuration files are that:

- When matching keys for configuration files, they are case sensitive.
- Configuration records use a reference location to allow you to specify where a record belongs.

A About Regular Expressions

This appendix discusses the format requirements for the regular expressions that are used in Configuration File Management (CFM). It also includes references for more information.

Regular Expression Specifications

The regular expressions format used in Configuration File Management is the Tcl programming language (version 8.2). This is a variant of the IEEE standard POSIX (Portable Operating System Interface) regular expression specification.

Additional References

For more information about these format requirements, see the following references:

- B. Welch, K. Jones: "Practical Programming in Tcl and Tk", 4th Ed., Prentice Hall, 2003
- T. Stubblebine: "Regular Expressions Pocket Reference", O'Reilly, 2003
- The Open Group Base Specification: "IEEE Std 1003.1, Regular Expressions", 2004; online at: **http://www.opengroup.org/onlinepubs/009695399/basedefs/xbd_chap09.html**
- IEEE Standard for Information Technology: "Portable Operating System Interface (POSIX)", American National Standards Institute, 1994

Index

F

H

I

K

L

M

N

O

P

R

S

T

U

V

Z