HP OpenView Proxy Server Using Radia

Radia Proxy Server Guide

Version: 4.0i

Note: All content in this manual applies to Windows users except for content identified solely "for UNIX," "for a UNIX platform," or "in a UNIX environment."

Manufacturing Part Number: T3420-90011

July 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.

OpenLDAP is a registered trademark of the OpenLDAP Foundation.

Acknowledgements

Technical Support

Please select Support & Services from the following web site:
<http://www.hp.com/managementsoftware/services>

There you will find contact information and details about the products, services, and support that HP OpenView offers.

The support site includes:

- Downloadable documentation
- Troubleshooting information
- Patches and updates
- Problem reporting
- Training information
- Support program information

Preface

About this Guide

Who this Guide is for

This book is for Radia system administrators who want to install the Radia Proxy Server in their enterprise environments to cache data at a location between the Radia Configuration Server and their subscribers. To use some of the features in this publication, you should be familiar with the Radia Management Portal, the Radia System Explorer, the Radia Configuration Server, and Radia Clients.

What this Guide is about

The *Radia Proxy Server Guide* describes:

- what the Radia Proxy Server is, and its static and dynamic cache processes.
- how to install and configure the Radia Proxy Server version 2.0 for Windows and 1.3.x for UNIX:
  - as an agent when servicing clients.
  - as a server when co-located with the Radia Configuration Server for HTTP download support.
- how to implement the Radia Proxy Server with your Radia Configuration Server and clients.
- how to administer the Radia Proxy Server using the Radia Management Portal version 2.0 or greater.

Note

To administer the Radia Proxy Server using the Radia Management Portal version 1.x, refer to procedures in the appropriate *Radia Management Portal Guide*. Archive versions are available from the HP OpenView support web site.

Summary of Changes

This printing of the *Radia Proxy Server Guide* contains the following changes to information and procedures for the following chapters.

Chapter 1: About the Radia Proxy Server

- Page 23, *Dynamic Cache Management*: clarified when the dynamic cache purge is triggered.

Chapter 2: Installation and Configuration

1.2 Page 35, *Installing the Radia Proxy Server Locally for Windows*, updated to show revised installation dialogs for this release. The End User License Agreement, shown in Figure 2.2 on page 37, is new for this release.

1.2 Page 45, Installing the Radia Proxy Server Locally for UNIX, updated to show revised installation dialogs for this release. The End User License Agreement, shown in Figure 2.12 on page 47, is new for this release.

- Page 56, *Requirements for Remote Installations from the Radia Management Portal*, added another requirement.
- Page 59, *Performing the Install Proxy Server Task,* updated steps and figures to show how to perform this task using Radia Management Portal version 2.0.
- Page 64, Figure 2.25 ~ Job Status page: new graphic.
- Page 69, *Table 2.2 ~ Configuration File rps.cfg Parameters*: Modified the definitions of these parameters to clarify a dynamic cache purge takes place only when the index file save takes place, and the index file save is skipped when there are no change to be applied:

  ```
  -dynamic-maxdays
  -dynamic-savefreq
  -dynamic-savetod
  ```

- Page 74, *Table 2.3 ~ Dynamic Cache Parameter Summary*, modified row discussing how to save the index file of the dynamic cache and purge the dynamic cache of aged files.
- Pages 75, *The Date-Based Purge of the Dynamic Cache* and *Saving the Index File* topics have replaced the old topic *Scheduling a Date-Based Purge of the Dynamic Cache*, to more accurately reflect the product.

  Added a note on page 76 about data-based purge.

- Page 80, *Table 2.4 ~ Configuring the RPS.CFG for a Co-located RPS,* changed the examples for the **–static-root** parameter from:

  Windows: C:/Novadigm/ConfigurationServer/DB/RESOURCE

  UNIX: /opt/Novadigm/ConfigurationServer/DB/RESOURCE

  to:

  Windows: C:/Novadigm/ConfigurationServer/DB

  UNIX: /opt/Novadigm/ConfigurationServer/DB

- Page 81, *Figure 2.29 ~ Sample Configuration for a Radia Proxy Server co-located with the Radia Configuration Server,* changed the entry for –static-root from:

  ```
  -static-root       "C:/Novadigm/ConfigurationServer/DB/RESOURCE"
  ```

  to:

  ```
  -static-root       "C:/Novadigm/ConfigurationServer/DB"
  ```

- Page 83, *To remove ZHTTPMGR support from a Radia Configuration Server:* Edited this procedure.
- Page 83, *Configuring the Radia Database for the Static Cache Preload*, refreshed Figures 2.30 to 2.32 to show System Explorer version 4.0.
- Page 89, *Configuring Radia Clients for Use with the Radia Proxy Server*, refreshed Figures 2.33 to 2.35 to show System Explorer version 4.0.

Chapter 3: Radia Proxy Server Administration

1.2 Page 104, *Synchronizing (preloading) the Radia Proxy Server*: modified steps and replaced images to show how to perform this using Radia Management Portal version 2.0.

- Page 106, *Purging the Dynamic Cache using the Radia Management Portal*: modified steps and replaced images to show how to perform this using Radia Management Portal version 2.0.
- Page 109, *Using Proxy Servers to Install Clients Remotely from the Radia Management Portal*: modified Step 1 of the overview.

Chapter 4: Troubleshooting

- Page 114, *Performance Problems*, expanded the discussion to include procedures on how to check the IRPStackSize value in Windows Registry, and links to obtain the valid IRPStackSize values for Windows operating systems.

- Page 118, *Reporting a Problem for a Radia Proxy Server*, topic title and contents modified from the previous topic: *Opening a Ticket for a Radia Proxy Server*.

Editorial Improvements

In addition to the changes listed above, this version contains various editorial and style updates to each chapter and section and the index.

Conventions

You should be aware of the following conventions used in this book.

Table P.1 ~ Styles

Element	Style	Example
References	*Italic*	See the *Publishing Applications and Content* chapter in this book.
Dialog boxes and windows	**Bold**	The **Radia System Explorer Security Information** dialog box opens.
Code	`Andale Mono`	`Radia_am.exe`
Selections	**Bold**	Open the **\Admin** directory on the installation CD-ROM.

Table P.2 ~ Usage

Element	Style	Example
Drives (system, mapped, CD)	Italicized placeholder	*SystemDrive*:\Program Files\Novadigm might refer to C:\Program Files\Novadigm on your computer. *CDDrive*:\client\radia_am.exe might refer to D:\client\radia_am.exe on your computer.
Files (in the Radia Database)	All uppercase	PRIMARY
Domains (in the Radia Database)	All uppercase	PRIMARY.SOFTWARE May also be referred to as the SOFTWARE domain in the PRIMARY file.
Classes (in the Radia Database)	All uppercase	PRIMARY.SOFTWARE.ZSERVICE May also be referred to as the ZSERVICE class in the SOFTWARE domain in the PRIMARY file.

The table below describes terms that may be used interchangeably throughout this book.

Table P.3 ~ Terminology*

* Depends on the context. May not always be able to substitute.

Term	May also be called
Application	software, service
Client	Radia Application Manager and/or Radia Software Manager
Computer	workstation, server
NOVADIGM domain	PRDMAINT domain **Note**: As of the 4.0 release of the database, the NOVADIGM domain is being renamed the PRDMAINT domain. Therefore, if you are using an earlier version, you will see the NOVADIGM domain in the database.
Radia Configuration Server	Manager, Active Component Server
Radia Configuration Server Database	Radia Database

Table P.4 ~ Radia Proxy Server Terminology

Term	Definition / Context
co-located Radia Proxy Server	A Radia Proxy Server installed on the same machine as the Radia Configuration Server so that resources can be downloaded from the Radia database using HTTP. Configured with a –static-type of server and with its –static-root pointing to the Radia database. Requires no preloading.
edmprof file	The Radia Configuration Server Settings file. This manual uses this non-platform specific, generic reference. • On UNIX operating systems, it is **.edmprof**, located on the **home** directory of the UNIX user ID that installs, starts, stops, and maintains the Radia Configuration Server. • On Windows operating systems, it is **edmprof.dat**, located in the **bin** folder of the Radia Configuration Server directory.
base installation directory	The location where your Radia Proxy Server is installed. By default, the Radia Proxy Server is installed into the following directory: • On a Windows machine, it is: ***<SystemDrive>*:\Novadigm\IntegrationServer** • On a UNIX machine, it is: **/opt/Novadigm/IntegrationServer**

Contents

1

About the Radia Proxy Server

At the end of this chapter, you will:

- Understand the Radia Proxy Server.
- Understand when to use a Radia Proxy Server.
- Know how to use the Radia Proxy Server.

What is the Radia Proxy Server?

When the Radia Proxy Server is used, it is the primary repository for Radia Client data. Once the Radia Client determines the resources needed for its desired state, it can request those resources from the Radia Proxy Server. Requests are made using either HTTP (recommended for all new installations) or TCP/IP. The Radia Proxy Server has the ability to service multiple, concurrent client requests using both protocols simultaneously.

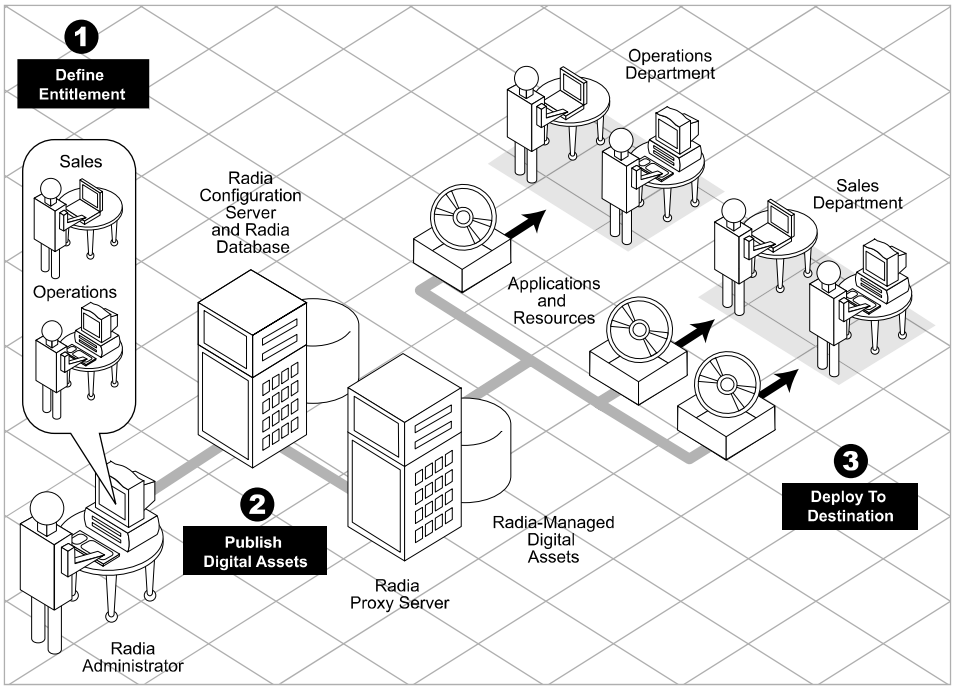


Figure 1.1 ~ Radia Infrastructure using Radia Proxy Server to Deploy Applications to Clients.

When to use a Radia Proxy Server

Radia Proxy Servers are beneficial in your environment if you have many client computers requesting the same resources from the same location. When data is cached on the Radia Proxy Server, the demand placed on the Radia Configuration Server is decreased, allowing the Radia Configuration Server to allocate more resources to other tasks.

Placing Radia Proxy Servers at strategic points in your network increases the rate at which data is transferred. The connection between subscribers and the Radia Proxy Server may be more efficient than the connection between the subscribers and the Radia Configuration Server. The factors that determine the efficiency of a connection between a server and a client computer include hardware capability, network bandwidth, workload on the servers, network traffic patterns, and the volume of software to be distributed.

Note

The Radia Proxy Server is not a generic proxy, but rather specifically designed to manage and distribute Radia resources.

This publication describes:

- Radia Proxy Server components and processes.
- The installation of the Radia Proxy Server components.
- The configuration and implementation of the Radia Proxy Server.
- The configuration of your Radia database and Radia clients for use with Radia Proxy Servers.
- The administration of the Radia Proxy Server.

Radia Proxy Server Processing

The logical flow for a typical client request to a Radia Proxy Server is as follows (assuming all components are enabled):

1. The Radia Client's resolution process indicates it needs resources, and it should request them from a Radia Proxy Server.
2. The Radia Client sends a request to the Radia Proxy Server, which is received by the Radia Proxy Server front-end protocol.

> **Note**
>
> The default (and recommended) protocol for client communication with the Radia Proxy Server is HTTP. Optionally, TCP is also available. A Radia Proxy Server can service multiple, concurrent client requests using both protocols simultaneously.

   The client request is validated, and passed to the cache manager.
3. The local static cache is searched, and if the data is found, the request is satisfied and the data is sent to the Radia Client.
4. If the data is not in the local static cache, the local dynamic cache is then searched. If the data is found, the request is satisfied.
5. If the file does not reside in the dynamic cache, Dynamic PassThru requests the file from the defined upstream host (typically, another Radia Proxy Server) using HTTP. If there is a connection failure with the upstream host, and multiple upstream hosts were defined (for fail-over support), Dynamic PassThru tries the next host on the list. The original request is satisfied, and the file is stored in the dynamic cache for future requests.

Figure 1.2 on page 19 illustrates the Radia Proxy Server process flow for handling client requests.

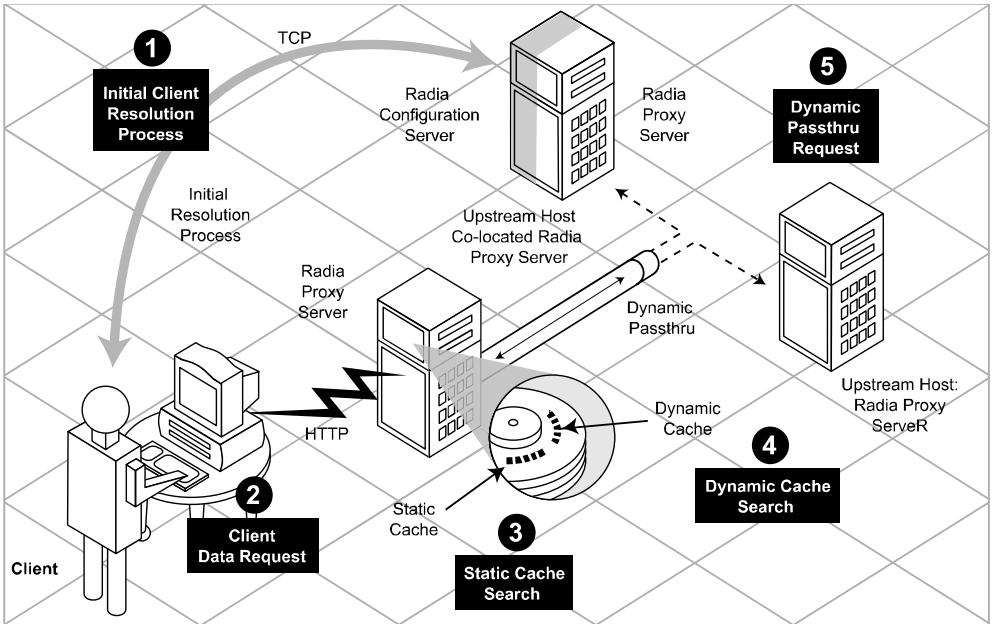


Figure 1.2 ~ Radia Proxy Server process flow.

Cache Definitions and Support

The Radia Proxy Server supports two types of cache locations: static and dynamic.

Static Cache

The *static cache* is the primary cache location for the Radia Proxy Server. The static cache can be configured to operate in one of two ways:

- **Static Type = Agent**
  When set to *agent*, the static cache is populated by the Preloader. For performance efficiency, this static cache should be preloaded with all resources that are expected to be distributed by the Radia Proxy Server to Radia Clients. The static cache is typically preloaded during off hours, so the required resources are available when requested by a Radia Client. For a discussion of this process, see *The Preloader Process* on page 21.
- **Static Type = Server (when co-located with the Radia Configuration Server)**
  When set to *server*, the static cache points to a native Radia Database on the same machine as the Radia Proxy Server. This co-located (or co-resident) Radia Proxy Server provides a means to download resources from the Radia database resources using HTTP. (The Radia Configuration Server can only download resources using TCP/IP.) For details on when downloading resources from the Radia database using HTTP is required, see *What is a Co-Located Radia Proxy Server* on page 24.

In both cases, the Radia Proxy Server views static cache as read only.

The Preloader Process

The Preloader populates the static cache of a Radia Proxy Server using a process that parallels the standard Radia Client resolution and deployment process. When the preloader runs, it uses TCP to connect to the assigned Radia Configuration Server for a resolution of the Radia Proxy Server's predefined distribution model. The required resources are then deployed to the Radia Proxy Server's static cache. In addition, resources no longer included in the Radia Proxy Server distribution model are removed from the static cache.

> **Note**
>
> As part of the Radia Proxy Server installation, a small version of the Radia Application Manager client is also installed. These components provide the functionality to support the Static Cache Preload process.

The preload resources can be deployed using TCP or HTTP. HTTP is available when the Radia Configuration Server has a co-located Radia Proxy Server.

The Preloader's Distribution Model

The Preload process is an application of the usual Radia distribution model for clients, with the following specific elements:

1. **Define Entitlement**
   The entitlement for Preloading a Radia Proxy Server is defined in the POLICY domain as follows:
   - **User**
     The machine identity of the Radia Proxy Server being managed (or preloaded).
   - **Applications**
     The software that is being preloaded to the Radia Proxy Server's static cache. This should include all applications normally requested by the set of Radia Clients that will be assigned to the Radia Proxy Server.
2. **Publish Digital Assets**
   - **Application Files**
     The components that make up the applications. When publishing MSI applications for distribution from a Radia Proxy Server, use the techniques discussed in this guide to have the ACPs preloaded to the Radia Proxy Servers but not distributed to the clients.
3. **Preload Radia Proxy Servers**
   - **Deployment Source and Protocol**
     Resources to preload the Radia Proxy Servers can come from the Radia Configuration Server, or, optionally, another Radia Proxy Server or Radia Staging Server. Resources can be deployed using TCP or HTTP. HTTP deployment from the Radia Configuration Servers uses the HTTP port of a co-located Radia Proxy Server.

- **Deployment Destinations**
  In this case, the Radia Proxy Server's static cache location is the Preload deployment destination.

The Preload distribution model is illustrated in Figure 1.3

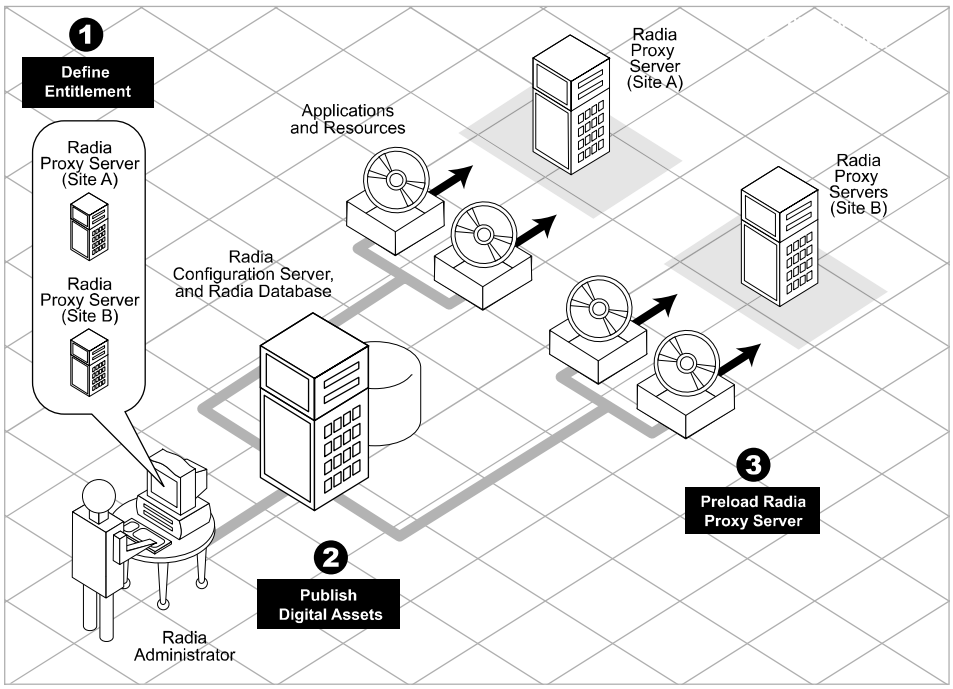


Figure 1.3 ~ The Radia Proxy Server Preload Distribution model.

Dynamic Cache

The *dynamic cache* is an optional, secondary cache location for the Radia Proxy Server. When enabled, the dynamic cache is populated on demand by the Dynamic PassThru component of the Radia Proxy Server using HTTP. When a requested resource is not found in the primary (static) cache, the dynamic cache is searched. If the requested resource is not found in the dynamic cache, the Dynamic PassThru process obtains the resource from an upstream host. The dynamic cache is viewed as a safety net for requests that fall through the static cache search.

Dynamic PassThru

When a client request is received for a resource that does not exist locally, the Radia Proxy Server can request these resources from an upstream host, such as a Radia Proxy Server co-located with the Radia Configuration Server, or another Radia Proxy Server. These resources are then returned to the requesting client, as well as stored locally in the dynamic cache for subsequent requests. Dynamic cache resources are transferred using HTTP.

For fail-over support, you can designate more than one upstream host for obtaining dynamic cache. If the Radia Proxy Server fails to connect with the first host on the list, it attempts to connect with the second listed host, and so on, to obtain the dynamic cache resources.

Dynamic Cache Management

Dynamic PassThru also manages this cache, purging files that have not been requested in a previously configured number of days. (This avoids keeping files in the dynamic cache after they exist in the static cache.) To support the purge process, an index file is maintained to keep track of when files were last used. The dynamic cache cleanup of "aged" files can be run from the Radia Management Portal using the Purge Dynamic Cache task. The purging of aged files will also run whenever the index file is saved. You can schedule the index file to be saved on a daily or more frequent basis. However, a scheduled save of the index file is skipped if dynamic cache resources were not used since the last save.

The dynamic cache can also be purged, as necessary, when a user-specified maximum file size is exceeded. The least-used files are deleted until the dynamic cache size is below the maximum file size. A "freespace" purge option allows you to purge down to a low-water mark, which is a certain percentage below the maximum file size. This "freespace" eliminates constant purging in an active dynamic cache environment.

Another purge option allows you to define "large files", and then exempt these large files from the first pass of a size-based purge. If large files are purged and then later requested, an undue load could be placed on the network. Use the large file options to alleviate this load.

What is a Co-Located Radia Proxy Server?

The Radia Configuration Server (RCS) communicates with other servers and clients using TCP/IP. It no longer supports a native HTTP data download capability. To obtain HTTP download capability, you need to use a co-located Radia Proxy Server.

A *co-located* Radia Proxy Server is a Radia Proxy Server placed on the same machine as the Radia Configuration Server to provide a source for downloading Radia database resources to another Radia server or clients using HTTP. It is defined with a static cache type of **server**, instead of agent. It does not manage its own static cache, but merely points to the Radia Database on the Radia Configuration Server. Thus, it needs no preloading (or synchronization), and normally has its dynamic cache disabled. This is shown in Figure 1.4 .

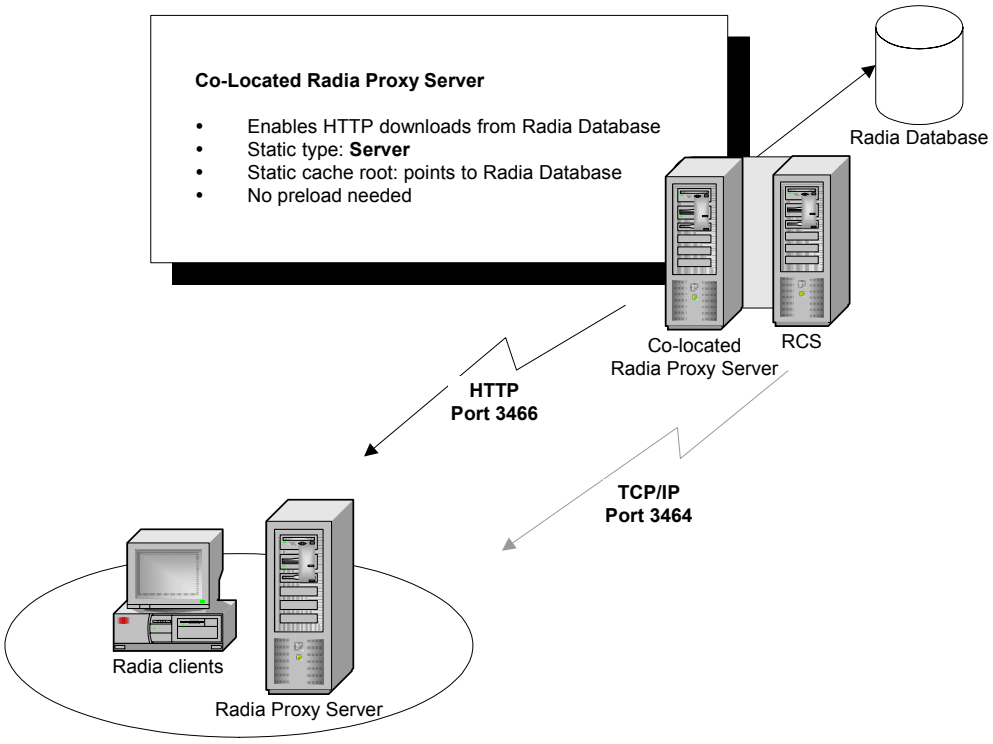


Figure 1.4 ~ Co-located Radia Proxy Server enables HTTP downloads from Radia Database.

For details on installing and configuring a co-located Radia Proxy Server, see the topic *Co-Locating a Radia Proxy Server with a Radia Configuration Server for HTTP Support* on page 79.

When to use a co-located Radia Proxy Server

You need a Radia Proxy Server co-located with the Radia Configuration Server for HTTP support in the following configurations:

- To support Radia Clients configured to retrieve resources from the Radia Database using HTTP (instead of TCP/IP).
- To support a subordinate, or downstream, Radia Proxy Server that is obtaining its dynamic cache resources from the Radia database. Dynamic cache resources must be obtained using HTTP.
- To support preloading a subordinate, or downstream, Radia Proxy Server using HTTP instead of TCP/IP.

Each configuration requiring a co-located Radia Proxy Server is discussed below:

- **Radia Clients using HTTP to obtain their resources**
  Radia Clients always obtain their policy resolution from the Radia Configuration Server. By default, the Radia Clients then obtain their resources from the Radia Database of the Radia Configuration Server using TCP/IP. To have Radia Clients obtain their resources from the Radia Database using HTTP, instead, you need to add a co-located Radia Proxy Server with the Radia Configuration Server, and then direct the clients to obtain their resources from the co-located Radia Proxy Server. For details on directing clients to obtain their resources from a Radia Proxy Server, see *Configuring Radia Clients for Use with the Radia Proxy Server* page 89.
- **Preloading a Radia Proxy Server from the Radia Database using HTTP**
  By default, a Radia Proxy Server is preloaded from the Radia Database using TCP/IP. Optionally, it can be preloaded from the Radia Database using HTTP, instead. This configuration would also require a Radia Proxy Server co-located with the Radia Configuration Server for HTTP support, as shown in Figure 1.5 . For details on using this option, refer to the topic: *Preloading Deployment Options* on page 86.

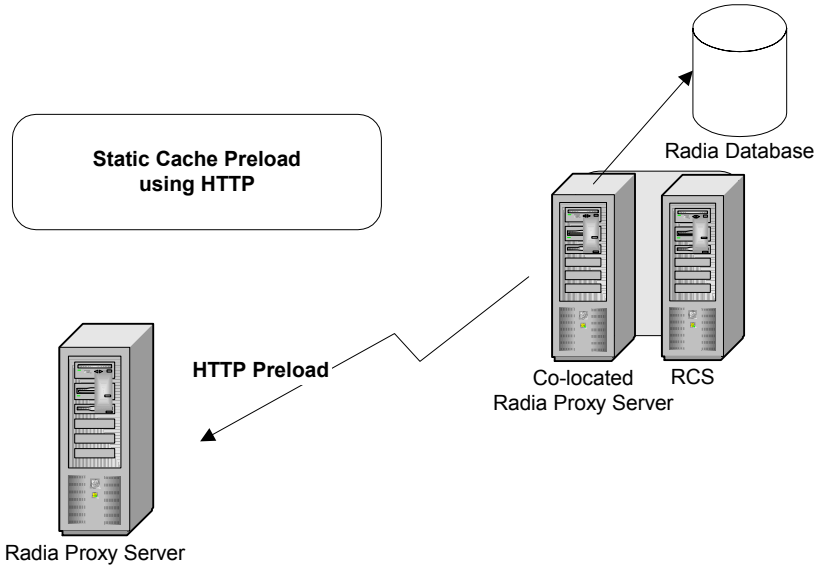


Figure 1.5 ~ The Radia Proxy Server Preload Distribution model.

- **Radia Proxy Server using the Radia Configuration Server as its dynamic cache upstream host**
  Enabling the dynamic cache for a Radia Proxy Server means that when a client requests resources from the Radia Proxy Server, and the resources are not in the local cache, the Dynamic Cache PassThru process immediately requests the resources from its predefined upstream host using HTTP.

  When a Radia Proxy Server is enabled for dynamic caching, and is using the Radia Configuration Server as the upstream host for obtaining its dynamic cache resources, you need to use a co-located Radia Proxy Server for HTTP download support. This is because the Dynamic PassThru process uses HTTP exclusively. Figure 1.6 shows this configuration.

  For details on specifying the dynamic cache and upstream host, see the topic *Configuring the Dynamic Cache Parameters* on page 73.

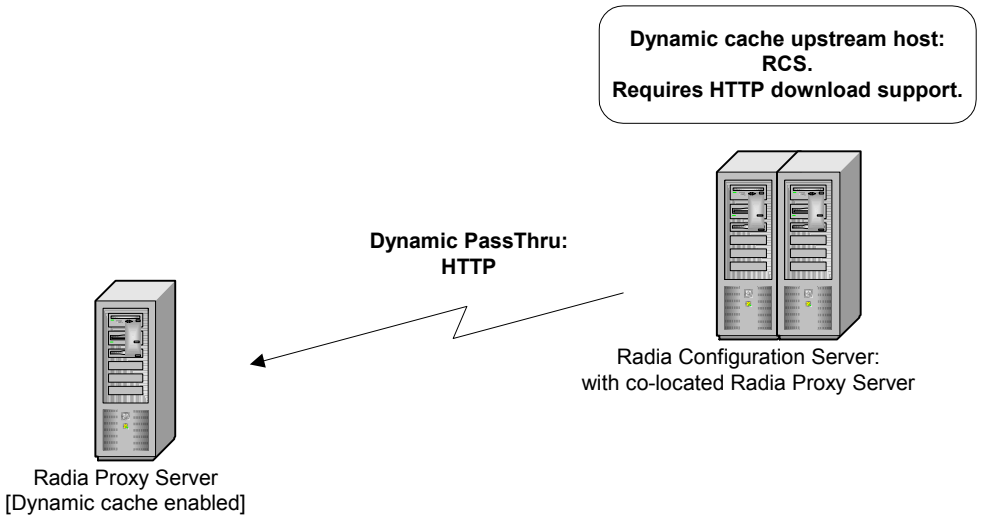


Figure 1.6 ~ Radia Proxy Server using the RCS as its dynamic cache upstream host.

Radia Integration Server and the Radia Proxy Server

The Radia Proxy Server runs as a loaded module under the control of the Radia Integration Server. The Radia Integration Server is a base component for various Radia infrastructure modules, such as the Radia Inventory Manager, the Radia Management Portal, and the Radia Policy Manager. It provides web services that are shared by loaded modules, resulting in a single entry point for all HTTP (web-based) requests. This integration provides performance, efficiency, and ease of maintenance in an adaptable and cohesive (server) framework.

The location of the Radia Integration Server is the base installation directory of the Radia Proxy Server. If the Radia Integration Server is not currently present, it is installed with the Radia Proxy Server.

Radia Management Portal and the Radia Proxy Server

The Radia Management Portal provides a web-based, single access point from which Radia administrators manage their entire Radia environment regardless of location or computing platform. Administrative tasks can easily be distributed to administrators in remote locations based on roles and policies.

The Radia Management Portal provides the ability to both install and perform many administrative functions for your Radia Proxy Servers. The remote install from the Radia Management Portal honors a pre-configured *.INI file, so the installed Radia Proxy Server is ready to go. In addition, the Radia Management Portal can be used to start, stop, preload (synchronize) the static cache, and purge the dynamic cache of any Radia Proxy Server in your Network.

In conjunction with the Radia Management Portal's ability to install Radia Clients remotely, Radia Proxy Servers can also be used to assist the Radia Management Portal in distributing client install scripts. This serves to spread the workload of deploying large numbers of clients between the Radia Management Portal and Radia Proxy Servers, just as the Radia Proxy Servers assist the Radia Configuration Server in deploying applications to the clients.

Summary

- Radia Proxy Servers enable an additional server to act as an extension of the Radia Configuration Server. The Radia Proxy Server stores a copy of the application software that Radia distributes, and delivers it to the Radia Client computers that are attached to the server.
- Each client will receive resources directly from the Radia Proxy Server. The recommended protocol for client communications is HTTP, although multiple, concurrent protocols are supported.
- A Radia Proxy Server's static cache is preloaded with the applications defined for it in the Radia Database. The preload process connects using TCP, but can have resources deployed using TCP or HTTP. HTTP is available when the Radia Configuration Server has a co-located Radia Proxy Server.
- A Radia Proxy Server enabled for dynamic cache will request resources from a predefined upstream host when they are not currently available in its local static or dynamic cache. Multiple upstream may be defined to provide fail-over support after a connection error. Dynamic cache resources are downloaded to the Radia Proxy Server using HTTP. The dynamic cache offers a number of purge options to remove old files and to keep it under a specified maximum size.
- A Radia Proxy Server co-located with the Radia Configuration Server is used whenever you need to obtain resources from the Radia Database using HTTP. The Radia Configuration Server no longer supports downloading resources using HTTP.
- The potential benefit of a Radia Proxy Server must be evaluated individually for each server and its attached client computers.

2

Installing and Configuring the Radia Proxy Server

At the end of this chapter, you will:

- Be able to install the Radia Proxy Server software locally using installation media, or remotely using the Radia Management Portal.
- Be able to access and apply any updates to bring the product to the latest level.
- Be able to configure the Radia Proxy Server after installation for different options, including the use of the dynamic cache.
- Understand how to configure the different Radia components for use with the Radia Proxy Server.

Installation and Configuration Overview

Before you can use a Radia Proxy Server, you must prepare your Radia environment. To configure your enterprise for using Radia Proxy Servers, you will need to complete the following tasks:

- ❑ Install the Radia Proxy Server on the designated server.
  - Read the topics on system requirements and installation notes, and then follow the steps to install the Radia Proxy Server locally or remotely (using the Radia Management Portal). Separate procedures are given local Windows and UNIX installations.
  - If you are installing the Radia Proxy Server co-located with the Radia Configuration Server (to provide a source for downloading resources from the Radia database using HTTP), see *Co-Locating a Radia Proxy Server with a Radia Configuration Server for HTTP Support* on page 79.
  - Also visit the HP Technical Support web site to check for the latest available updates to the product, such as a Service Pack update.
- ❑ Review and modify the Radia Proxy Server configuration parameters after installation. For example, edit the configuration file to enable and configure the dynamic cache.

  See *Configuring the Radia Proxy Server* on page 66.
- ❑ Create a distribution policy in the Radia Configuration Server database for preloading the Radia Proxy Server static cache.

  See *Configuring the Radia Database for the Static Cache Preload* on page 83.
- ❑ Preload the Radia Proxy Server static cache.

  See *Preloading the Radia Proxy Server* on page 103.
- ❑ Assign the appropriate subscribers to the Radia Proxy Server.

  See *Configuring Radia Clients for Use with the Radia Proxy Server* on page 89.

Radia Proxy Server System Requirements

Hardware Requirements and Recommendations

- **Static Type of Agent**
  In general, a Radia Proxy Server with a static type of Agent is most dependent on network bandwidth and disk I/O speed. Use the recommendations given in Table 2.1 to obtain desired performance on your Radia Proxy Servers. The Radia Proxy Server will run on lesser machines, but performance will probably suffer under peak loads.

- **Static Type of Server**
  A Radia Proxy Server co-located with the Radia Configuration Server for HTTP support has a static type of Server. Most hardware requirements for this Radia Proxy Server are more than accommodated by those of the Radia Configuration Server's needs, especially for processor speed, memory, and disk drive speed. See the *Radia Configuration Server Installation Guide* for specific requirements. For additional considerations, see Table 2.1 below.

Table 2.1 ~ Radia Proxy Server Hardware Recommendations

Component	Static Type of Agent	Static Type of Server
Overall	In general, the Radia Proxy Server is most dependent on network bandwidth and disk I/O speed.	A machine sized for the Radia Configuration Server (RCS) more than accommodates a co-located Radia Proxy Server.
Processor	A higher processor speed is more important than having multiple processors. Thus, we recommend as fast a processor as is practical (for example, 2 GHz).	Use RCS requirements for speed. However, multi-processors are strongly recommended for this configuration.
Memory	At least 512 MB. As with all systems, the more memory the better.	RCS requirements are fine.
Disk Space	Provide at least double the anticipated total volume of resources to be housed (that is, your anticipated static and dynamic cache).	RCS requirements are fine.
Disk Drive Speed	Fast access disk drives are highly advisable.	RCS requirements are fine.
Network Interface Card (NIC)	A fast network card is a plus. Use a 1 GB NIC if your network supports it.	A second network card is helpful to address configurations where network volume becomes a constraining factor (large volume due to the size or number of resources to be transferred to clients). When using two network cards, the RCS is homed to one NIC IP address and the Radia Proxy Server to the other.
Network Configuration	Minimize the number of router hops between client requesters and their respective Radia Proxy Server to improve performance.	If servicing clients, recommendation for a static type of Agent applies.

Windows System Requirements

- Windows NT 4 with Service Pack 6a, 2000 with Service Pack 3 or later, XP, or Server 2003.
- Connection to a computer running the Radia Configuration Server.
- Administrator rights to the computer to install the Radia Proxy Server.

UNIX System Requirements

- Solaris operating system Version 2.5.1 or above, SPARC CPU, Motif 1.2 libraries.
- HP-UX operating system Version 10.20 or above, PA Risc CPU, Motif 1.2 libraries.
- AIX operating system Version 4.3.1, Motif 1.2 libraries.
- RedHat Linux Version 6.2 or above, Intel Pentium processor or compatible CPU.
- Connection to a computer running Radia Configuration Server.

Radia Proxy Server Installation

There are two methods for installing the Radia Proxy Server:

- Install the Radia Proxy Server directly onto a local server.
- Install the Radia Proxy Server remotely through the Radia Management Portal. For detailed information about the Radia Management Portal, see the *Radia Management Portal Guide*.

Windows and UNIX Installation Notes

- To complete the Radia Proxy Server installation prompts, you need to understand the Radia Proxy Server Preload process that resolves and loads the static cache. For details, please see *Cache Definitions and Support* on page 20.
- Before you begin, locate your license file. You need this license file to install the products that you have purchased. If you need assistance, contact HP Technical Support (see page 4).

  During installation of the Radia Proxy Server, the license file is renamed **license.nvd**, and is copied to the Radia Integration Server's **module** directory.
- After installation, the Windows Radia Proxy Server service is started automatically; but the UNIX Radia Proxy Service is not.
- After installing the product from the Radia Management Infrastructure CD-ROM, check the HP Technical Support web site to see if a maintenance Service Pack is needed to bring the product to the latest level. For details, see *Applying Product Updates* on page 65.
- To install the Radia Proxy Server on a UNIX system, make sure the user who is installing the Radia Proxy Server is logged in as root, and has adequate rights to create and update the target installation directory.

Note to Solaris users

For the Radia Proxy Server to operate correctly on Solaris platforms, the UNIX user ID running the Radia Proxy Server must include the directory **/usr/sbin** in his UNIX PATH environment variable setting. Contact your local system administrator if you need assistance.

Note to HP users

In order for Radia to install correctly on HP-UX platforms, you must mount the Radia Infrastructure CD-ROM using pfs_mount.

The Radia Infrastructure CD-ROM is created using the Rock Ridge format. Since the HP-UX standard mount procedure is incompatible with the Rock Ridge file system type, HP has made available the PFS package (Portable File System) that allows their workstations to recognize this format. Specific instructions follow:

- Insert the CD-ROM and mount by typing:
  /usr/sbin/pfs_mount -v -x unix */cdrom/*mnt
  where ***/cdrom*** is your physical CD-ROM device.
- To un-mount, type:
  /usr/sbin/pfs_umount /mnt

See your local UNIX systems administrator and UNIX man pages for more information.

Installing the Radia Proxy Server to a Local Directory

This section describes how to install the Radia Proxy Server to a local site for both Windows and UNIX.

- Instructions for Windows follow.
- Instructions for UNIX begin on page 45.

Following the installation, proceed with the topic *Configuring the Radia Proxy Server* on page 66.

Installing the Radia Proxy Server Locally for Windows

This section describes how to install the Radia Proxy Server to a local site for Windows.

To install the Radia Proxy Server locally for Windows

1. Double-click the **setup.exe** file from the Radia Proxy Server installation source directory. The source directory is found on the Radia Infrastructure CD at the following path:
   \extended_infrastructure\proxy_server\win32

The **Welcome** window opens.

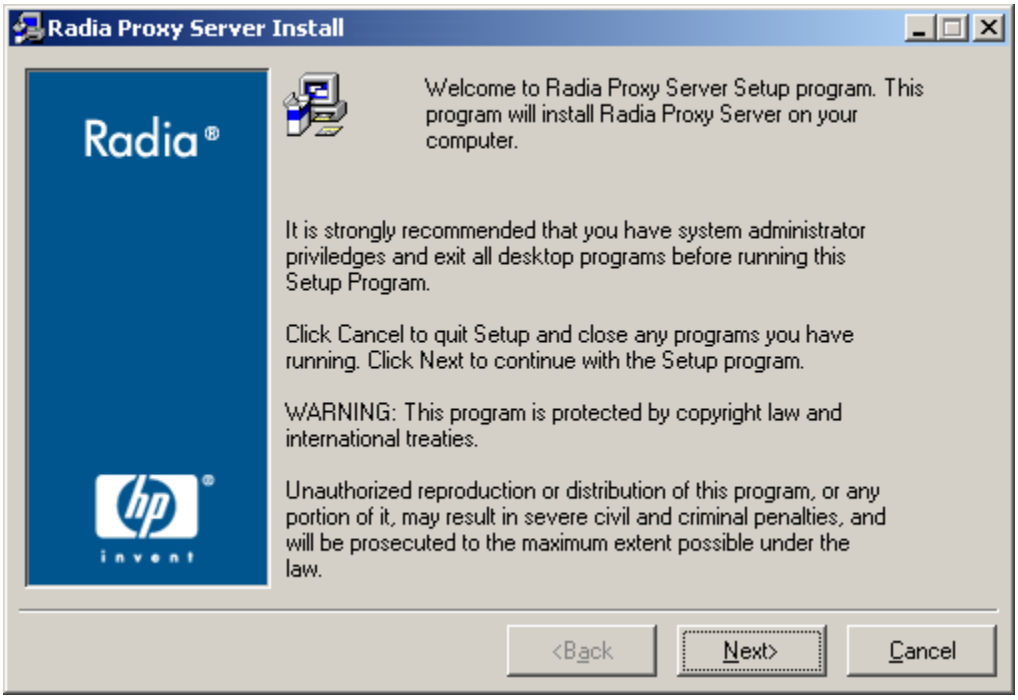


Figure 2.1 ~ Welcome window.

2. Click **Next**.

The **End-User License Agreement** window opens for you to read the licensing terms for this product. You must accept the terms before the Radia Proxy Server can be installed.

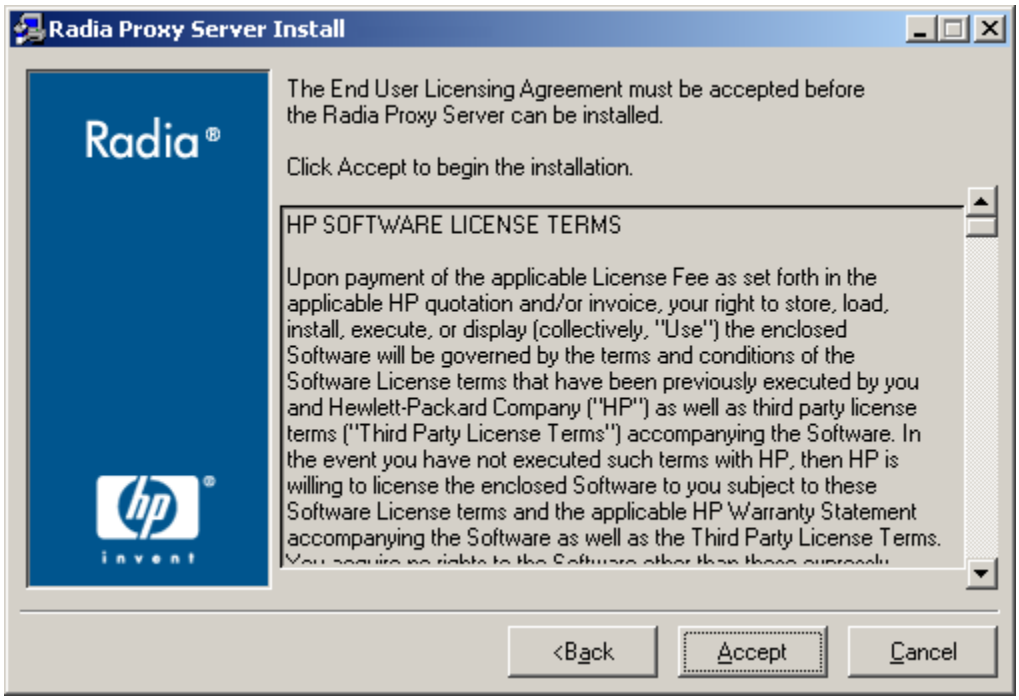


Figure 2.2 ~ End User Licensing Agreement.

3. Click **Accept** to agree to the terms of the software license and continue with the installation.

The **Radia Proxy Server Directory** window opens for you to select or enter the **base directory** for the Radia Proxy Server install. This Radia Proxy Server base directory specifies the location of the foundation Radia Integration Server component. The default is **C:\Novadigm\IntegrationServer.**

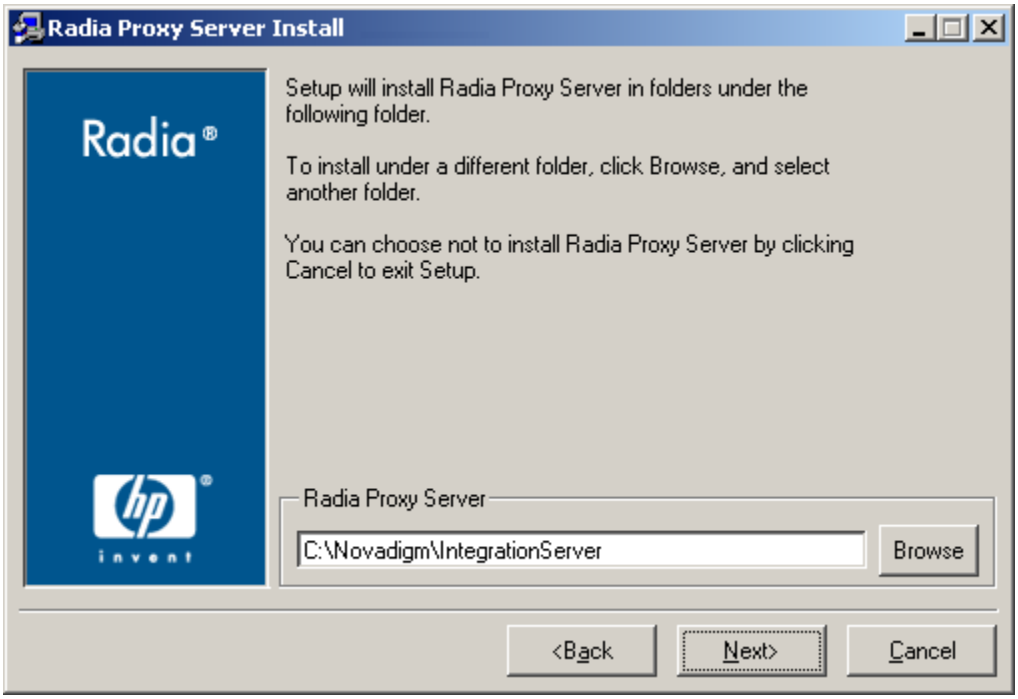


Figure 2.3 ~ Radia Proxy Server directory.

Note

If an instance of the Radia Integration Server component [httpd.tkd] is already on the target machine, then the **Radia Proxy Server Directory** window is bypassed and the Radia Proxy Server is installed automatically to the existing Radia Integration Server folder. Continue with Step 5.

4. Type the name of the base directory in which to install the Radia Proxy Server, accept the default directory shown in the text box, or click **Browse** and navigate to the directory in the **Browse** dialog box.

Note

For each value requested during this installation procedure, press the ENTER key to accept the default.

5. Click **Next**.

   The **License File** window opens.

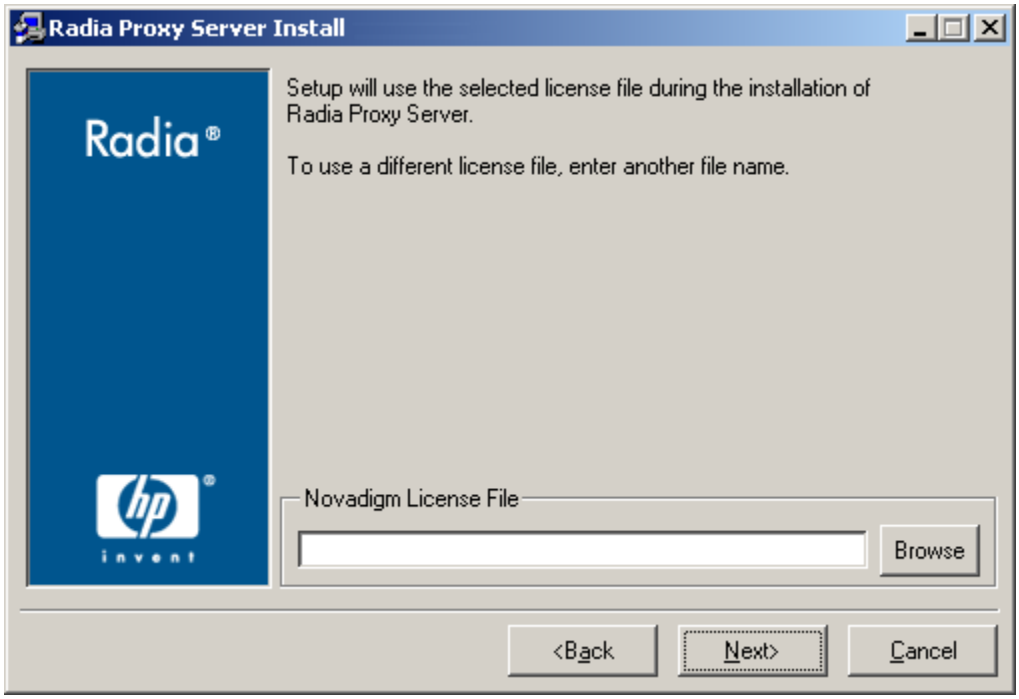


Figure 2.4 ~ License File window.

6. Browse to or type the name of your License File in the text box. You must enter a valid license file to continue the installation.

   The license file will be copied to the /**modules** folder as **license.nvd** during the install.

7. Click **Next**.

   The **Radia Configuration Server IP Address** window opens.

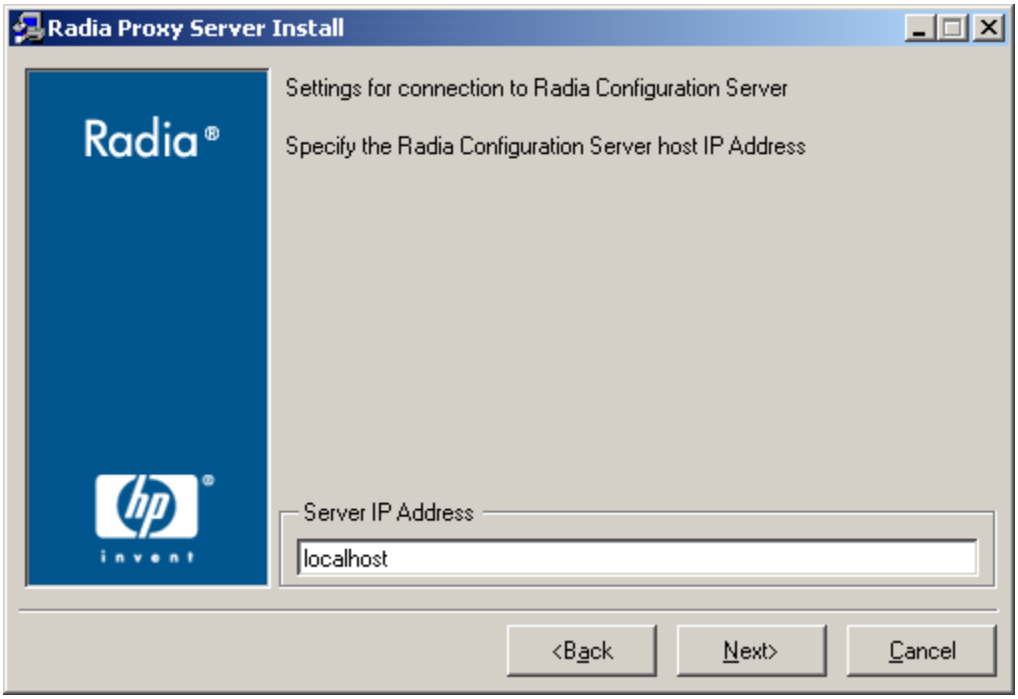


Figure 2.5 ~ Radia Configuration Server IP Address window.

8. In the **Server IP Address** text box, type the IP Address or DNS name of the host Radia Configuration Server. The Radia Proxy Server connects to this host during the preload process to obtain its static cache resolution and static cache files. The default is **localhost**.

Notes

Leave the default entry of **localhost** if you are co-locating this Radia Proxy Server with a Radia Configuration Server to enable HTTP download support. For additional details, see *Co-Locating a Radia Proxy Server with a Radia Configuration Server* on page 79.

The Radia Proxy Server Preload process must always connect to the host Radia Configuration Server, specified in the **Server IP Address** text box, for a static cache resolution. However, you can configure the Radia Proxy Server to obtain the static cache files from another Radia Proxy Server, if desired. For details, see the procedure *Preloading Deployment Options* on page 86.

9. Click **Next**.

The **Radia Configuration Server Port** window opens.

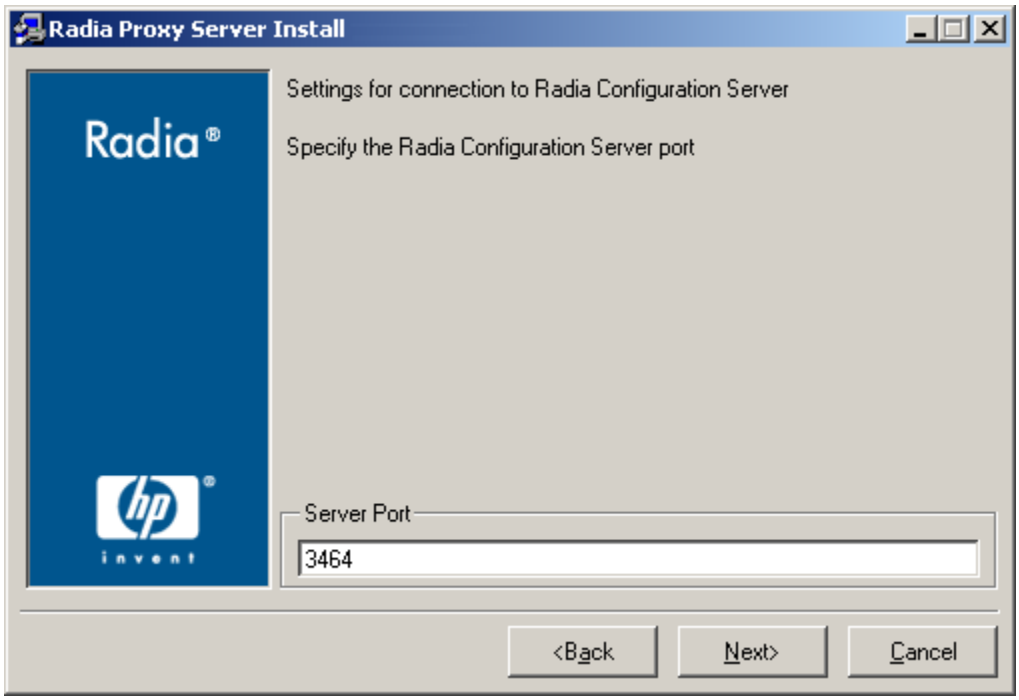


Figure 2.6 ~ Radia Configuration Server Port window.

10. Accept or type the TCP port number of the Radia Configuration Server to be used during the preload process for the static cache. The default port number is 3464.

11. Click **Next**.

The **Radia Configuration Server User ID** window opens.

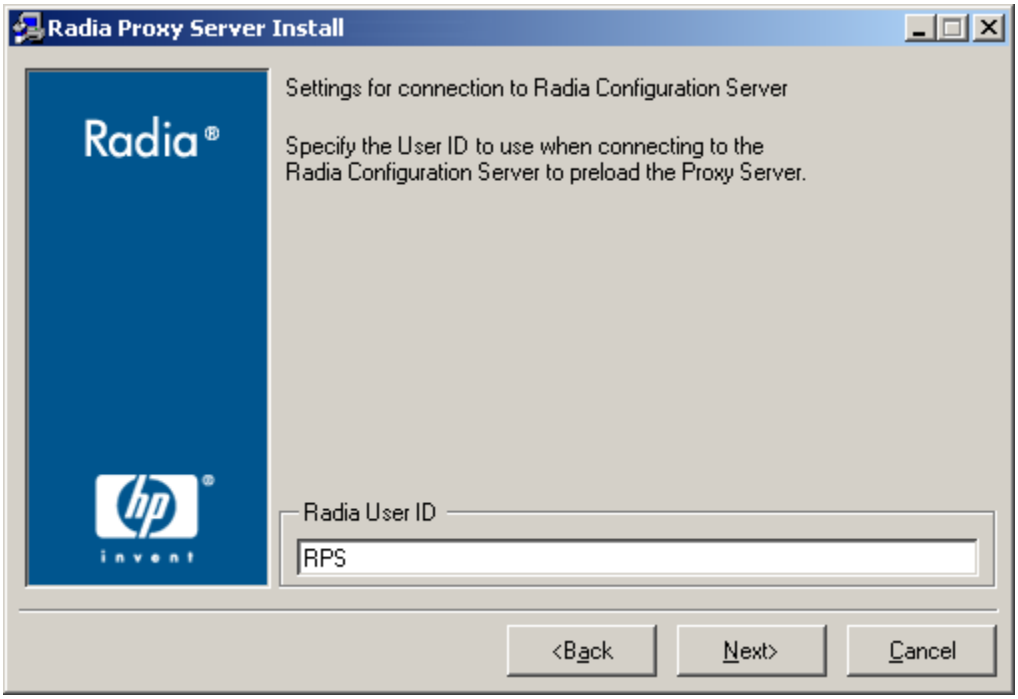


Figure 2.7 ~ Radia Configuration Server User ID window.

12. Specify the Radia User ID for this Radia Proxy Server to use when it connects to the Radia Configuration Server for its static cache preload resolution. The default is RPS.

Note

The Radia user ID entered must correspond to a Policy User class instance in the Radia Database, where the static cache distribution model for this Radia Proxy Server is defined. See *Configuring the Radia Database for the Static Cache Preload* on page 83.

13. Click **Next**.

The **Installation Settings** window opens.

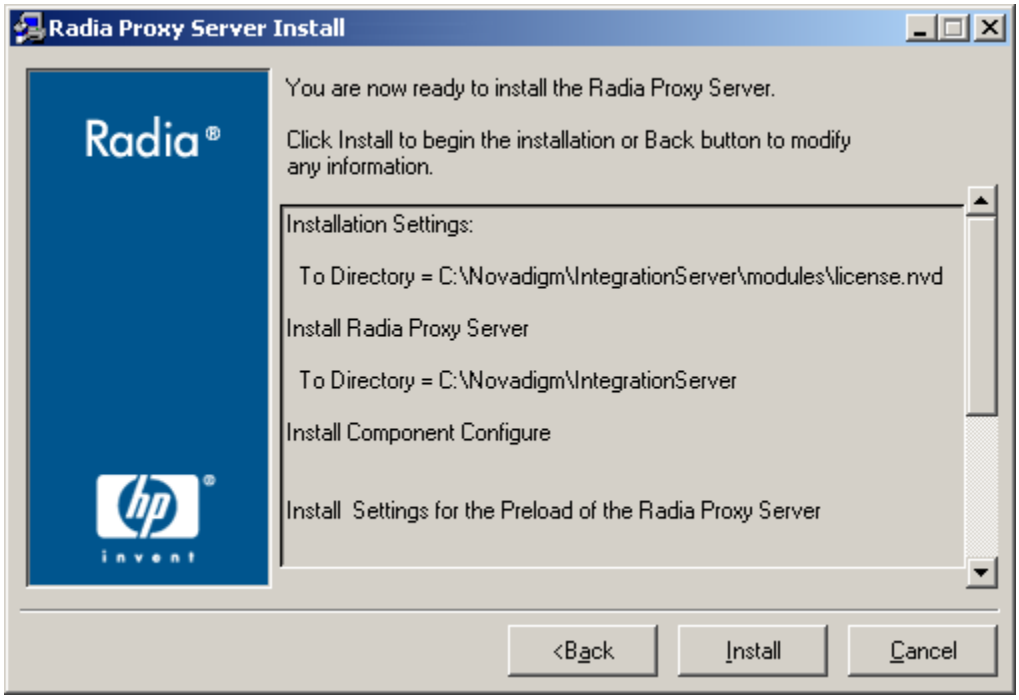


Figure 2.8 ~ Installation Settings window.

14. Review the settings in the **Installation Settings** window. If you want to change any of these settings, click **Back.**
15. Click **Install** to begin the Radia Proxy Server installation.

The **Installation Progress** window opens.

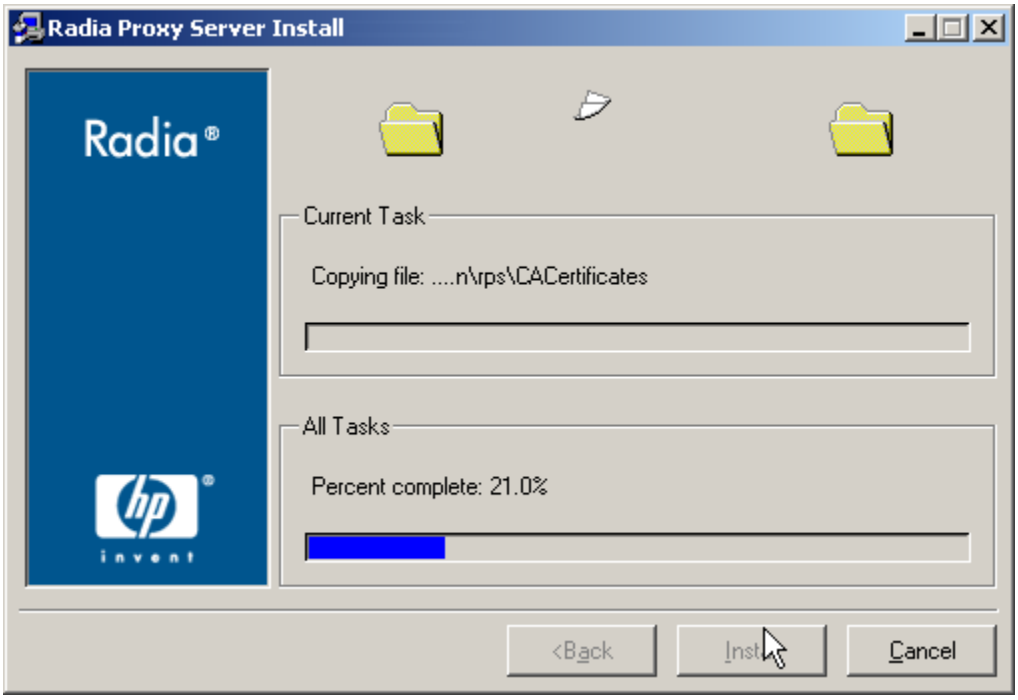


Figure 2.9 ~ Installation Progress window.

When the installation is finished, the **Successful Installation** window opens.

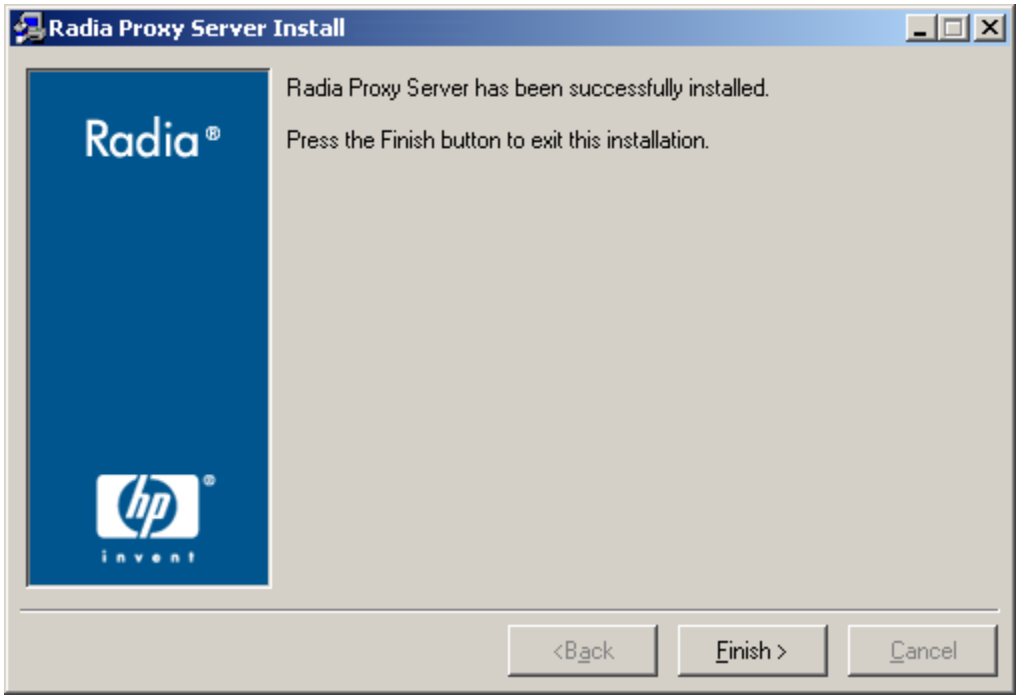


Figure 2.10 ~ Successful Installation window.

16. Click **Finish** to exit the installation program.

You have successfully installed the Radia Proxy Server Windows to a local directory. The Radia Proxy Server service is started.

Note

If you want to apply product updates at this time, see *Applying Product Updates* on page 65.

Proceed with *Configuring the Radia Proxy Server* as discussed on page 66.

Installing the Radia Proxy Server Locally for UNIX

This section describes how to install the Radia Proxy Server to a local site for UNIX.

Following the installation, proceed with the topic *Configuring the Radia Proxy Server* on page 66.

To install the Radia Proxy Server locally for UNIX

1. Change your current working directory to the directory containing the Radia Proxy Server media. This will be either your mounted CD-ROM directory or a temporary directory where you loaded the Radia Proxy Server media.
2. At the command prompt, type **./install**, and then press ENTER.

   The **Welcome** window opens.

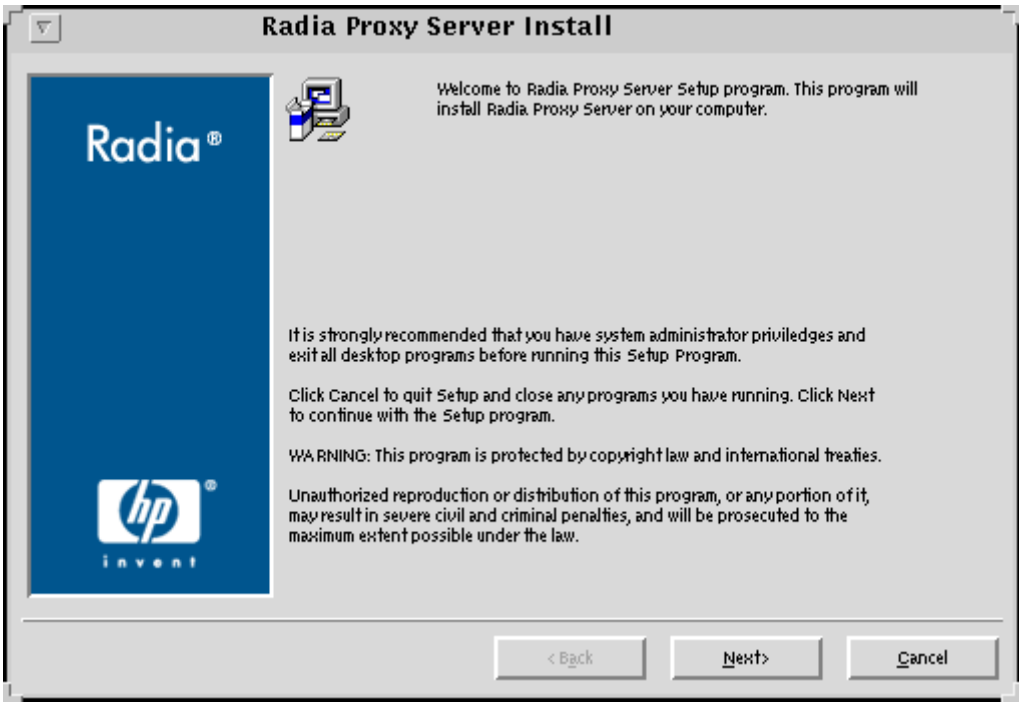


Figure 2.11 ~ Welcome window.

3. Click **Next**.

   The **End-User License Agreement** window opens for you to read the licensing terms for this product. You must accept the terms before the Radia Proxy Server can be installed.

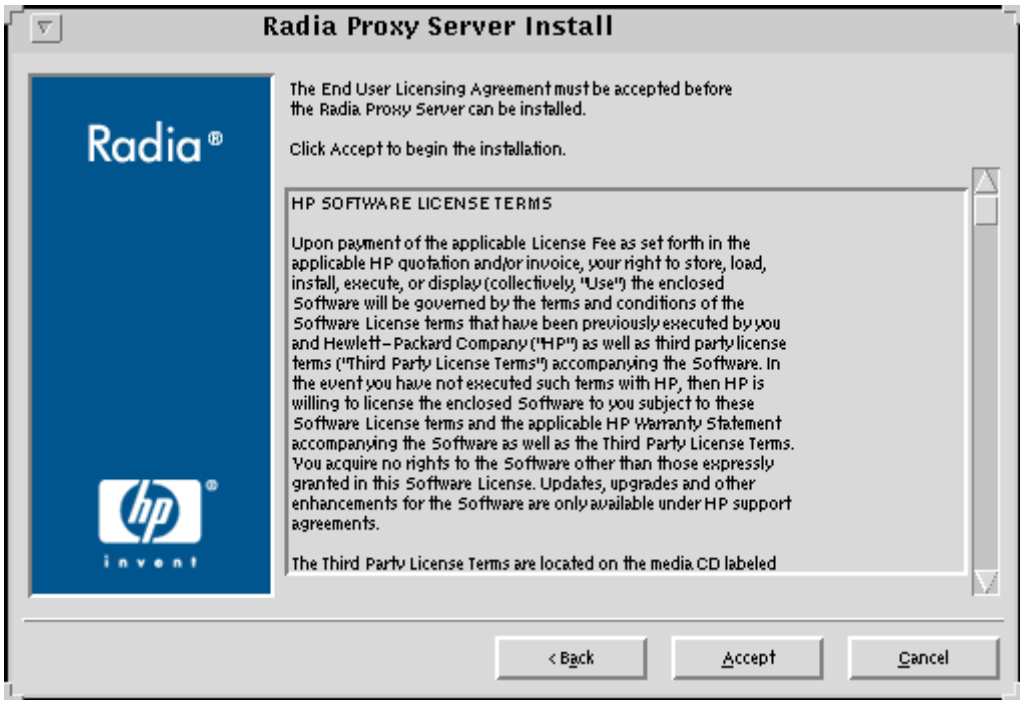


Figure 2.12 ~ End User Licensing Agreement.

4. Click **Accept** to agree to the terms of the software license and continue with the installation.

The **Installation Directory** window opens for you to select or enter the **base directory** for the Radia Proxy Server install.

Note

If an instance of the Radia Integration Server component [httpd.tkd] is already on the target machine, then the **Radia Proxy Server Directory** window is bypassed and the Radia Proxy Server is installed automatically to the existing Radia Integration Server folder. Continue with Step 7.

Figure 2.13 ~ Installation Directory window.

5. Accept the default Radia Proxy Server location, or type or browse to an alternate directory where you would like to install the Radia Proxy Server.

This **Radia Proxy Server** base directory specifies the location of the Radia Integration Server component, installed with the product.

6. Click **Next**.

The **License File** window opens.

Figure 2.14 ~ License File window.

7. Type the location and name of your License File, or click **Browse** to select the file from the **Browse** dialog box.
8. Click **Next**.

   The **Radia Configuration Server IP Address** window opens.

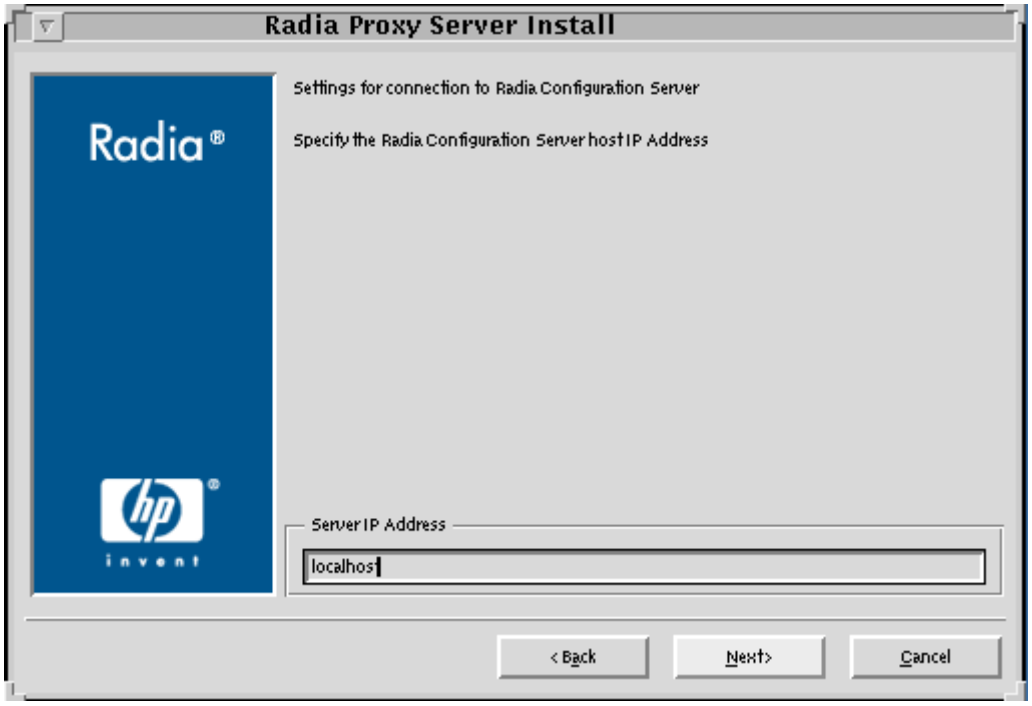


Figure 2.15 ~ Radia Configuration Server IP Address window.

9. Type the IP Address or hostname of the host Radia Configuration Server the Radia Proxy Server will connect to during a preload to obtain its static cache resolution and static cache files.

Notes

Leave the default entry of **localhost** if you are co-locating this Radia Proxy Server with a Radia Configuration Server to enable HTTP download support. For additional details, see *Co-Locating a Radia Proxy Server with a Radia Configuration Server* on page 79.

The Radia Proxy Server Preload process must always connect to the host Radia Configuration Server, specified in the **Server IP Address**, for a static cache resolution. However, you can configure the Radia Proxy Server to obtain the static cache files from another Radia Proxy Server, if desired. For details, see the procedure *To change the deployment source or protocol for a Preload* on page 86.

10. Click **Next**.

The **Radia Configuration Server Port** window opens.

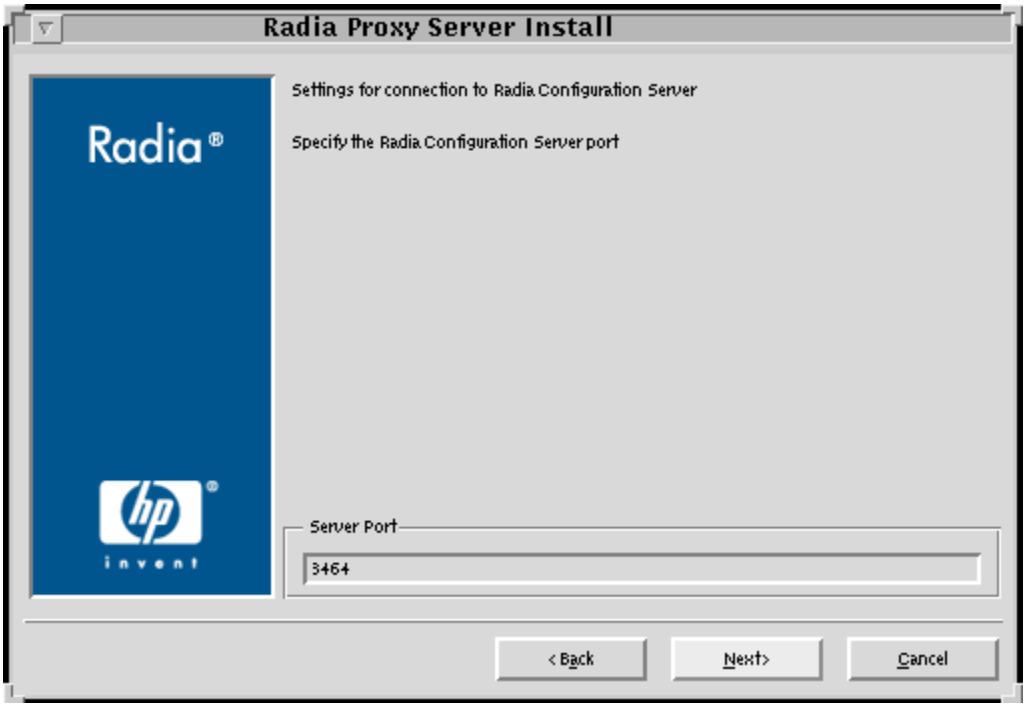


Figure 2.16 ~ Radia Configuration Server Port window.

11. Accept or type the TCP port number of the Radia Configuration Server to be used to connect to the Radia Proxy Server for the static cache Preload resolution. The default port number is 3464.

12. Click **Next**.

The **Radia Configuration Server User ID** window opens.

Figure 2.17 ~ Radia Configuration Server User ID window.

13. Type the user ID that the Radia Proxy Server will use to connect to the Radia Configuration Server to obtain its static cache preload resolution. The default is RPS.

Notes

The Radia User ID entered must correspond to a Policy User class instance in the Radia Database, which defines the applications to be loaded in the static cache. For details, see *Creating a Distribution Model for a Static Cache Preload* on page 84.

14. Click **Next**.

The **Installation Settings** window opens.

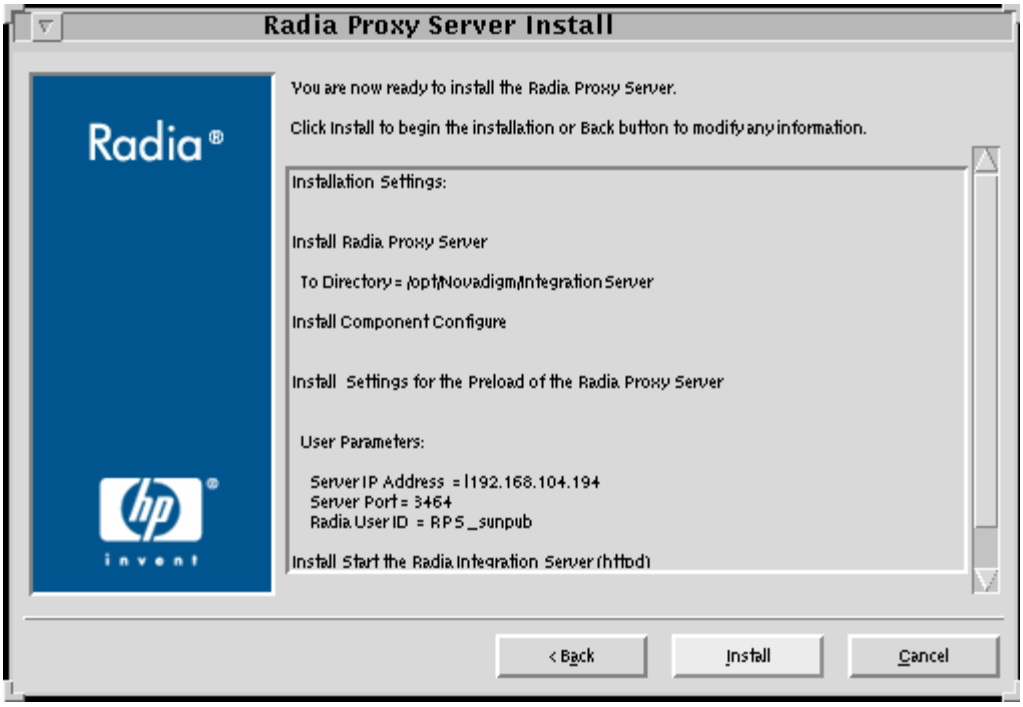


Figure 2.18 ~ Installation Settings window.

15. Review the settings in the **Installation Settings** window. If you want to change any of these settings, click **Back** until you reach the area of the installation you would like to change.

16. Click **Install** to begin the Radia Proxy Server installation.

The **Installation Progress** window opens.

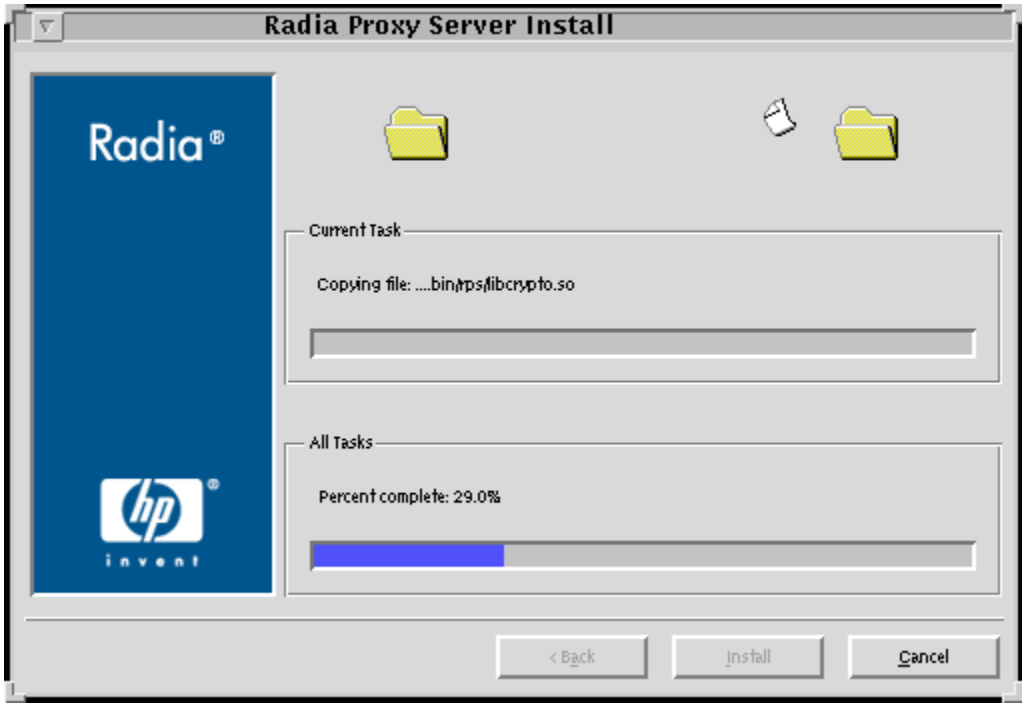


Figure 2.19 ~ Installation Progress window.

17. When the installation is finished, the **Successful Installation** window opens.

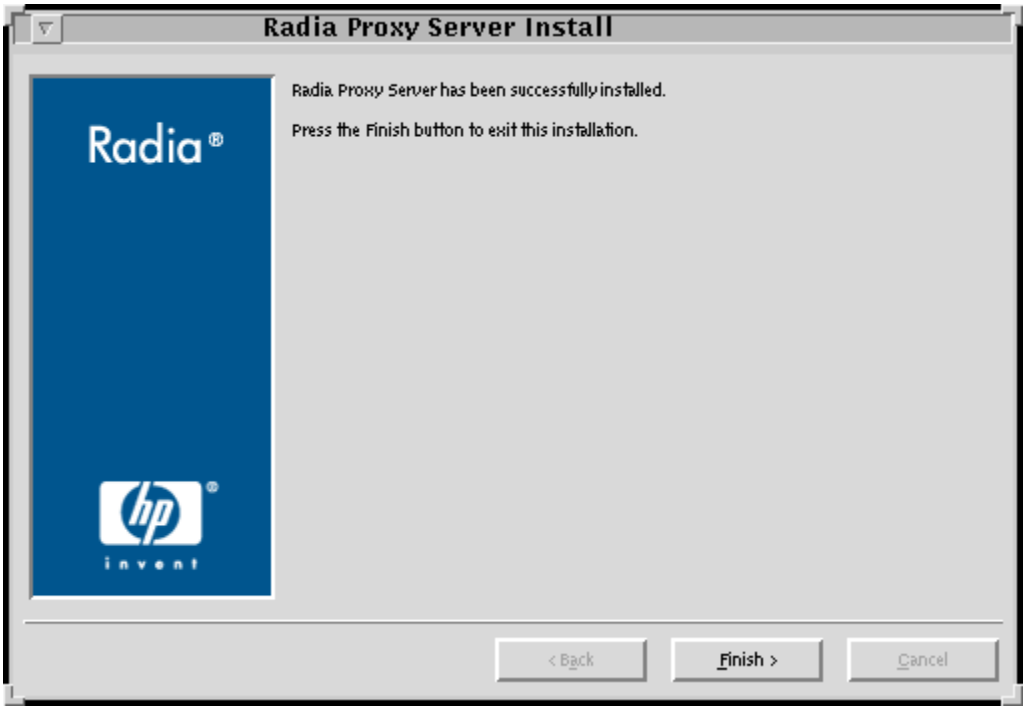


Figure 2.20 ~ Successful Installation window.

18. Click **Finish** to exit the installation program.

You have successfully installed the Radia Proxy Server locally for UNIX.

- To apply Product Updates at this time, see *Applying Product Updates* on page 44.
- To complete the configuration of the Radia Proxy Server for various options, go to *Configuring the Radia Proxy Server* on page 66.
- To start or stop the Radia Proxy Server for UNIX, use the commands discussed in the next topic, *Starting and Stopping the Radia Proxy Server for UNIX*.

Starting and Stopping the Radia Proxy Server for UNIX

To start the Radia Proxy Server for UNIX

1. Change your current directory to the directory where you installed the Radia Proxy Server (**/opt/Novadigm/IntegrationServer/** by default).

2. Type:

./nvdkit httpd.tkd

3. Press ENTER
4. The Radia Proxy Server is started on your computer.

To stop the Radia Proxy Server for UNIX

Note

The following are general guidelines and the commands are examples that may vary slightly depending on the UNIX type you are using.

1. Obtain the process ID for the Radia Proxy Server by listing all the UNIX processes and extracting the process ID for nvdkit.

   ```
   ps -f | grep nvdkit | sed /grep/d | awk '{ print $2 }'
   ```

2. Run the following command.

   ```
   kill <PID>
   ```

   The Radia Proxy Server is stopped on your computer.

Installing the Radia Proxy Server to a Remote Location

The Radia Proxy Server can be installed remotely using the Radia Management Portal from any web browser. During the installation, you will receive status information, and if the installation fails, it can be rescheduled.

The Radia Management Portal also allows you to select a pre-configured CFG file during the Install Proxy Server task. Using this option means the installed Radia Proxy Server can be fully configured and ready to run. See *Preparing and Locating Configuration Files for Remote Proxy Server Installs* on page 58 for more information.

Important Note

The following sections require you to be familiar with the Radia Management Portal. For detailed information, see the *Radia Management Portal Guide for Windows* or the *Radia Management Portal Guide for Windows UNIX*.

Requirements for Remote Installations from the Radia Management Portal

In order to install Radia Infrastructure products from the Radia Management Portal, you must be aware of the following requirements.

- For Windows, the remote computer must be running Windows NT, 2000, XP, or Server 2003.

Note

In some cases, Windows XP may need to be configured to support a remote installation. See the HP Technical Support web site for more information.

- For HP-UX, the remote computer must be running the HP-UX operating system Version 10.20 or above, PA Risc CPU.
- For Solaris, the remote computer must be running the Solaris operating system Version 2.5.1 or above, SPARC CPU.
- The installation files for the Radia product must be stored in the Radia Integration Server's **\media** directory. The Radia Management Portal installation program will copy these files automatically when you opt to install the components for remote installations from the installation dialog boxes. See *Installing the Radia Management Portal for Windows* in the Radia Management Portal Guide for more information.

  If you did not copy these files using the installation program, you must manually copy these files from the appropriate CD-ROM to the Radia Integration Server's **\media\extended_infrastructure** directory. The directory structure of this directory should mirror the CD-ROM layout.

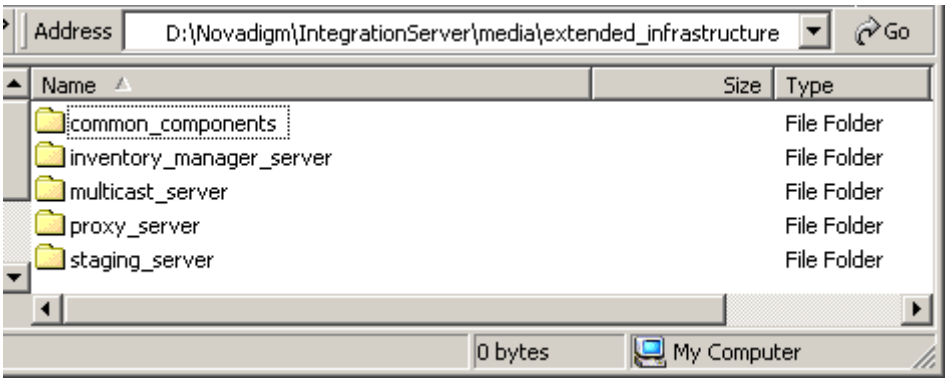


Figure 2.21 ~ Directory structure for \media\extended_infrastructure directory.

- A packing list, which contains a list of the files to be transferred across the network, must exist in the directory with the installation files. The Radia Management Portal creates the packing list when you launch the remote installation.
- For Radia Management Portal version 2.x and above, the Devices container must have an entry for each machine to which you are installing the Radia Proxy Server. See the *Radia Management Portal Guide for Windows* for more information.

Preparing and Locating Configuration Files for Remote Proxy Server Installs

Optionally, use these procedures to prepare one or more fully configured PRS.CFG files before you run the Radia Management Portal **Install Proxy Server** task. This allows you to install a Radia Proxy Server that is already configured and ready to go.

The pre-configured RPS.CFG files must be placed in a specific media location for the Radia Management Portal to use them. When you run the Install Proxy Server task from the Radia Management Portal, the task will prompt you to select a specific CFG file, if multiple ones exist.

To prepare a pre-configured RPS.CFG file for use with the Install Proxy Server task

1. Prepare a fully configured RPS.CFG file.

   Perform a local installation of the Radia Proxy Server on a test machine running the same platform as the intended Radia Proxy Server platform. Edit the resulting rps.cfg file using the directions given in *Configuring the Radia Proxy Server* on page 66.

2. Place the configured rps.cfg file in the appropriate Radia Management Portal media directory. The location of a configured rps.cfg file will vary according to the platform on which you are installing the Radia Proxy Server: win32, hpux, or solaris. For example, the location for a Windows Radia Proxy Server installation is similar to this:

   C:\Novadigm\IntegrationServer\media\extended_infrastructure\proxy_server\win32\media\etc

   Use the steps below to place the rps.cfg file at the appropriate \media location for the Radia Management Portal.

   a. Go to the directory where the Radia Management Portal is installed.

   The default is either
   <*SystemDrive*>:\Novadigm\Radia Integration Server

   OR

   <*SystemDrive*>:\Novadigm\IntegrationServer

   b. Go to the following folder location in the Radia Management Portal directory:
   \media\extended_infrastructure\proxy_server\<*platform*>\media
   where <*platform*> is win32, hpux, or solaris, according to the platform on which you are installing the Radia Proxy Server.

   c. Add a **\etc** folder to the **\media** directory.

   d. Copy the rps.cfg file to this platform-specific **\media\etc** folder. For example, if the Radia Management Portal is installed on **C:\Novadigm\IntegrationServer**, and the Radia Proxy Server will be installed on a Windows platform, then place the rps.cfg file in the following location:
   C:\Novadigm\IntegrationServer\media\extended_infrastructure\proxy_server\win32\media\etc

3. Run the **Install Proxy Server** task from the Radia Management Portal, as usual. See *Performing the Install Proxy Server Task* on page 59. The installation task will transfer the fully configured rps.cfg file. If you prepare more than one configuration file, use the **RPS Config File** drop-down list to select the appropriate one during the Install Proxy Server task.

Performing the Install Proxy Server Task

Use these steps to install one or more Radia Proxy Servers from the Radia Management Portal version 2.0 or later. If you are using an earlier version of the Radia Management Portal, refer to the *Radia Management Portal Guide* for that release for detailed steps.

> **Caution**
>
> You may also want to check for the latest information available on this topic on the HP OpenView support web site.

To install the Radia Proxy Server to a remote location

1. Access the Radia Management Portal from any web browser.
2. Use the **Navigation aid** to select the place in your infrastructure where you want to install the Radia Proxy Server. For example, you can select a single device from the Devices container or you can select a group of devices from the Groups container.

> **Note**
>
> If a group of devices is selected as the audience (target), the Proxy Server will be installed on all members of the group.

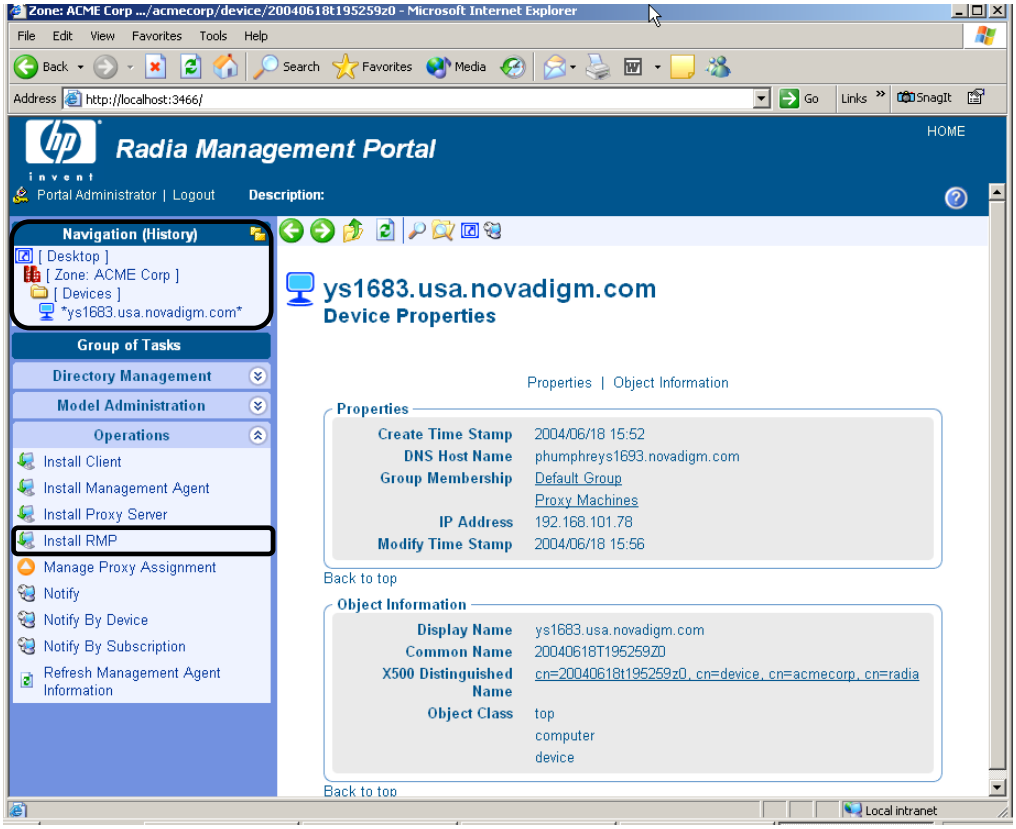


Figure 2.22 ~ Radia Management Portal – Starting Location is a single Device.

3. From the **Operations** task group, click **Install Proxy Server**.

4. Click **Next**.

The **Install Proxy Server—RPS Options** dialog box opens.

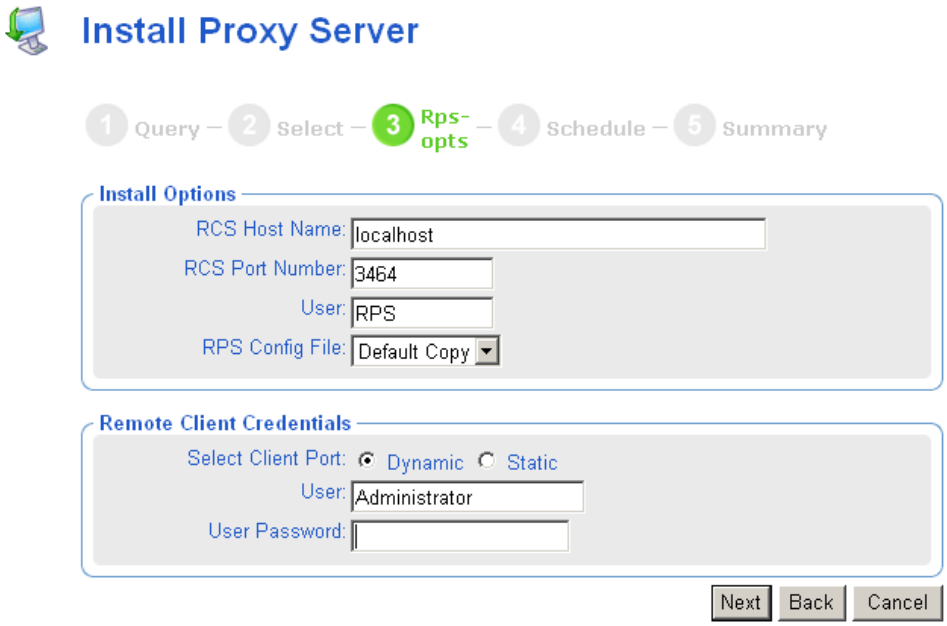


Figure 2.23 ~ Install Proxy Server—Install Options dialog box.

5. In the **RCS Host Name** text box, type the name or IP address for the Radia Configuration Server.
6. In the **RCS Port number** text box, type the port number for the Radia Configuration Server.
7. In the **RCS User** text box for Install Options, type the user ID to use to connect to the Radia Configuration Server.
8. If available, select which RPS configuration file to use during the installation from the **RPS Config File** drop-down list. This field only appears if multiple customized rps.cfg configuration files have been added to the Radia Management Portal.

> **Note**
>
> To make customized Radia Proxy Server configuration files available for selection during this task, see *Preparing and Locating Configuration Files for Remote Proxy Server Installs* on page 58.

9. In the **User** text box for Remote Client Credentials, type the administrator ID to obtain administrative authority on the target device's domain.

Tip

In order to take advantage of the Install Radia Proxy Server task, consider creating a standard administrator ID across the domains in your network.

10. *(Windows only)* In the **User Password** text box, type the administrator password to obtain administrative authority on the target device's domain.

If you do not enter the password, and administrative authority is required, the job may fail. Check the job status for specific information.

Note

If you do not enter the password, and administrative authority is required, the job may fail. Check the job status for specific information.

11. Click **Next**.

The **Schedule** dialog box opens.

12. In the **Schedule** dialog box, specify when you want this job to run. See *Scheduling Jobs* in the *Radia Management Portal Guide* for more information.

13. Click **Next**.

The **Install Proxy Server—Summary** dialog box opens.

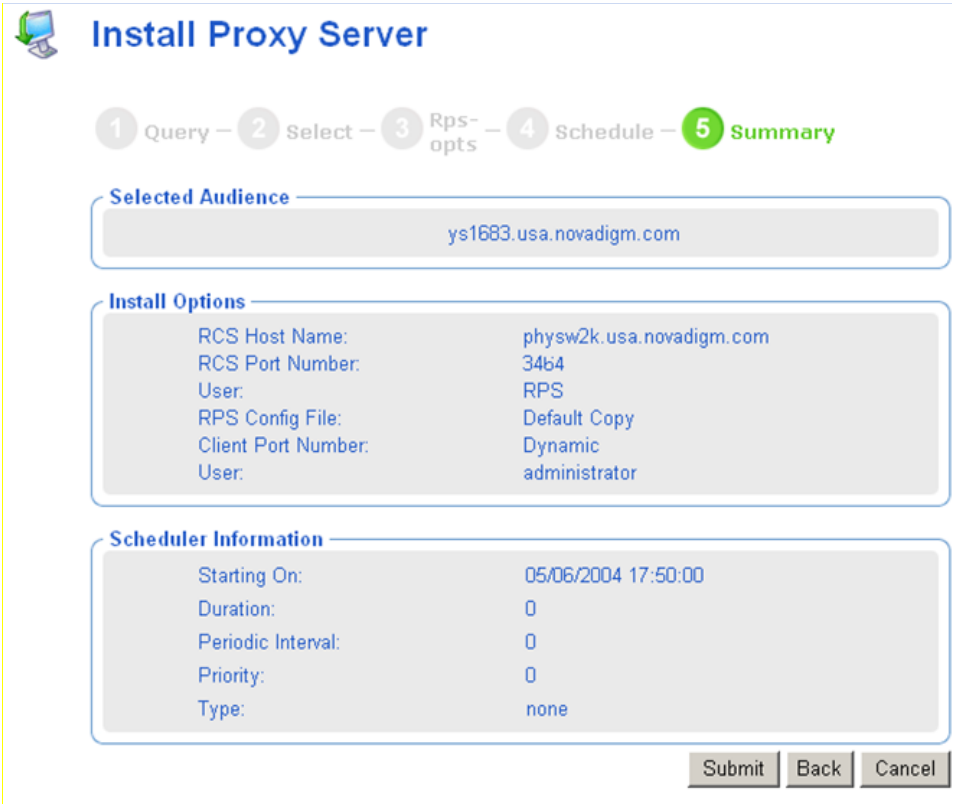


Figure 2.24 ~ Install Proxy Server—Summary dialog box.

14. Click **Submit**.

The **Job Status** page opens with list of the jobs. This page automatically refreshes every 60 seconds. Press **F5** to manually refresh it.

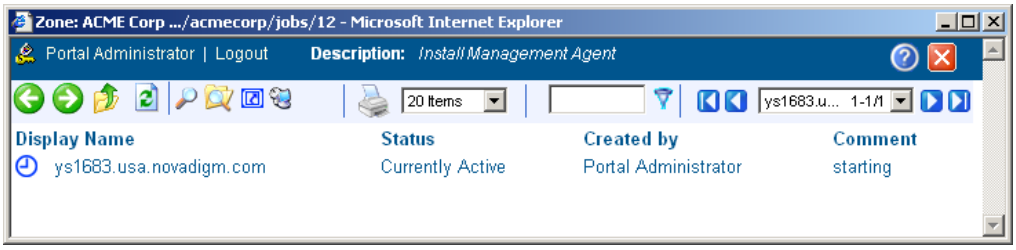


Figure 2.25 ~ Job Status page.

- Click  to go up one level in the job or directory tree. For example, after viewing job details, click this icon to return to the Job Group Summary.
- Click  if you want to refresh the status of the installation.
- Click  to view detailed properties for the job or job group. This gives you detailed information on the job status.
- Click  to add a shortcut for Jobs to your Desktop.
- Click  to obtain a printable view of the Jobs Status page.

15. When you are done viewing the job status, click  to close the **Job Status** page, and return to the Radia Management Portal.

You have finished installing the Radia Proxy Server. The Radia Proxy Server service for Windows is automatically started after it is installed.

Note

When the Radia Management Agent is also installed on the devices running a Radia Proxy Server, those devices automatically become members of the Radia Proxy Server group in the Cross References, Infrastructure Services container of the Radia Management Portal. See *About the Zone Containers* in the *Radia Management Portal Guide* for more information.

Radia Proxy Server Directory Structure

The Radia Proxy Server is installed, by default, into the following folders:

- **Novadigm\IntegrationServer** folder, for Windows.
- **/opt/Novadigm/IntegrationServer** folder, for UNIX.

Installing the Radia Proxy Server adds the following subdirectories to the **IntegrationServer** folder.

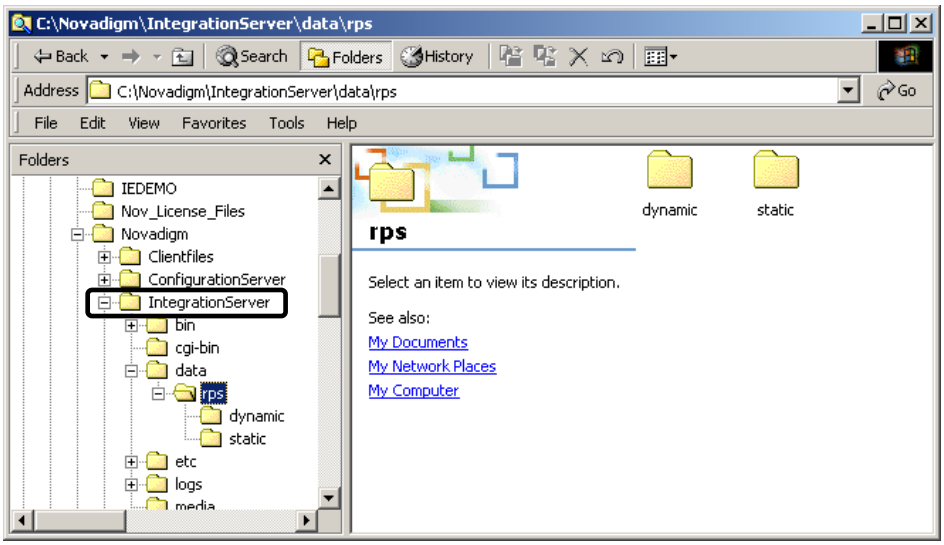


Figure 2.26 ~ Radia Proxy Server directory structure.

Applying Product Updates

Product updates for licensed users of the Radia Proxy Server are available for download from the HP OpenView web site:

- Check this site after installing the product from the Radia Infrastructure CD to see if later updates are available to bring the product to the latest level.
- Check this site on a regular basis to learn about available product maintenance.

Configuring the Radia Proxy Server

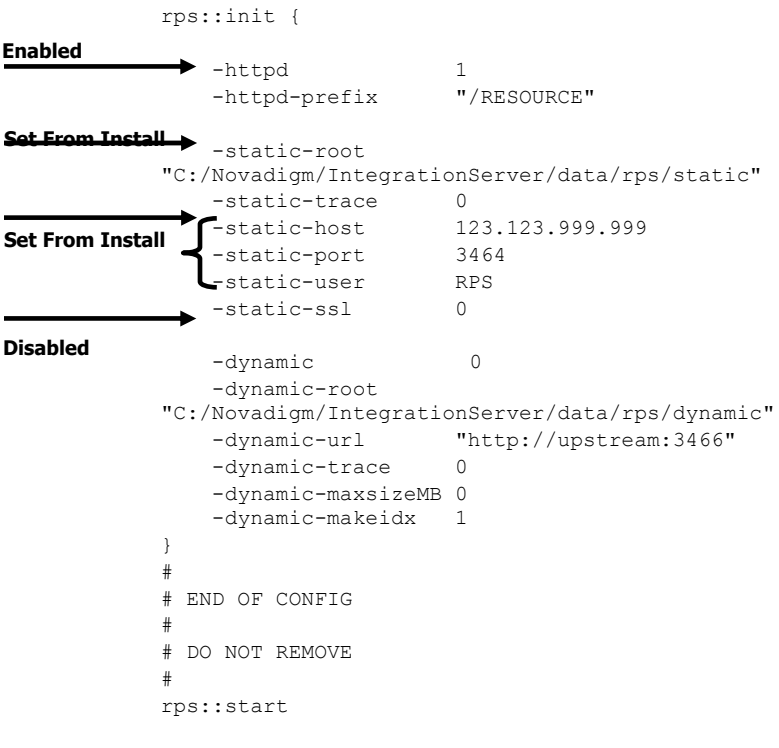
Radia Proxy Server Configuration File Overview

The Radia Proxy Server installation creates a configuration file, **rps.cfg**, located in the **/etc** folder of the Radia Proxy Server base installation directory. Following a local install, review the configuration parameters and make modifications to the rps::init section for the front-end communications protocol or the static or dynamic cache.

Table 2.2 on page 69 defines all parameters in the **rps.cfg** file and their default values.

Sample rps.cfg file after an install

The following figure shows a sample rps.cfg following an installation. Your rps.cfg file may show additional entries.




```
rps::init {
    -httpd             1
    -httpd-prefix      "/RESOURCE"
    -static-root
"C:/Novadigm/IntegrationServer/data/rps/static"
    -static-trace      0
    -static-host       123.123.999.999
    -static-port       3464
    -static-user       RPS
    -static-ssl        0
    -dynamic            0
    -dynamic-root
"C:/Novadigm/IntegrationServer/data/rps/dynamic"
    -dynamic-url       "http://upstream:3466"
    -dynamic-trace     0
    -dynamic-maxsizeMB 0
    -dynamic-makeidx   1
}
#
# END OF CONFIG
#
# DO NOT REMOVE
#
rps::start
```

Figure 2.27 ~ Sample rps.cfg file.

By default, after the installation:

- Http front-end protocol is enabled; Stager front-end protocol (for TCP) is disabled
- Static cache is configured with entries from the install. The -static-root entry defines the data store location, and the -static-host, -static-port and -static-user entries define the static cache preload operation.
- Dynamic cache operations are disabled.

Refer to the following topics to configure your rps.cfg file:

- *RPS.CFG Configuration Parameters Table* on page 68. This defines all of the configuration parameters available in this release.
- *Configuring the Dynamic Cache Parameters* on page 73.
- *Configuring the Radia Proxy Server in an Internet Proxy (Firewall) Environment* on page 78.
- *Co-Locating a Radia Proxy Server with a Radia Configuration Server* on page 79.

RPS.CFG Configuration Parameters Table

Table 2.2 on page 69 summarizes the parameters in the **rps.cfg** file and their default values. Parameters are listed in alphabetical order.

The configuration parameters in the rps.cfg file fall into three categories:

- **Front-end client communication options: -httpd (http) or -stager (TCP/IP)**
  By default, the Radia Proxy Server is configured with the **–httpd** front end enabled. This uses the HTTP front end protocol to communicate with Radia clients. When needed, the -**stager** front-end option is also available to communicate with Radia 2.x clients using TCP/IP; for details, see *Using TCP/IP for Client Communication* on page 92.
- **Static cache settings**
  The group of **–static*** parameters configure the static cache and its TCP connection to the Radia Configuration Server to obtain a preload resolution. Modify the –static* parameters in the rps.cfg file to change the values set during the installation or to set additional options, such as tracing.

  Note

  To configure a Radia Proxy Server co-located with the Radia Configuration Server, you need to modify the static cache parameters after installation. See the topic *Co-Locating a Radia Proxy Server with a Radia Configuration Server* on page 79.

- **Dynamic cache settings**
  The dynamic cache is not configured during the install. You must manually enable it (that is, change **–dynamic** from 0 to **1** in the rps.cfg) and specify its options by editing the rps.cfg file. For details, see Configuring the Dynamic Cache Parameters on page 73.

Table 2.2 ~ Configuration File rps.cfg Parameters

Parameter	Default	Description
-dynamic	0	When set to **1**, the dynamic cache is enabled and an entry in **–dynamic-url** is required to specify the upstream host. When set to 0, the dynamic cache is disabled, and all –dynamic* parameters are disregarded.
-dynamic-allow-shared-resource-purge	0	When set to **0**, safeguards against purging dynamic cache from a Radia Database. Recommend keeping at 0. Set to **1** to remove safeguard.
-dynamic-defdmn		HTTP upstream URL (Radia Configuration Server's) domain.
-dynamic-defcls		HTTP upstream URL (Radia Configuration Server's) class.
-dynamic-freespace	10	A percentage of the -dynamic-maxsizeMB to be left as free disk space after executing a size-based purge. The default is 10 %. When a size-based purge is triggered, the purge will delete files until the cache file size is reduced by this percentage below the maximum size. Used with –dynamic-maxsizeMB.
-dynamic-largefile-mindays	0	Specifies the minimum days a 'large file' should be retained (during a size-based purge). Disabled when set to 0 (default). Used with –dynamic-maxsizeMB and –dynamic-largefile-size.
-dynamic-largefile-size	0	Specifies the size of a 'large file' in bytes. To specify the size in MB, enclose in quotation marks, such as: "250 mb". When not equal to 0, files of this size or greater are excluded from deletion during the first pass of a size-based purge. Disabled when set to 0 (default) Used with –dynamic-maxsizeMB.
-dynamic-maxdays	7	The number of days to keep un-requested resources in the dynamic cache (if a file is regularly requested, it would never be deleted). Defines "aged" files for dynamic cache cleanup. Files that have not been requested for longer than this period are deleted from the cache whenever the Purge Dynamic Cache task is run from the Radia Management Portal, and whenever the dynamic index file is saved. (See *–dynamic-savetod* and *–dynamic-savefreq* for frequency.)
-dynamic-maxsizeMB	0	Maximum size of dynamic cache. **0** indicates no maximum. To set a maximum size, enter the number of MB, such as 200 for 200 MB.

Table 2.2 ~ Configuration File rps.cfg Parameters

Parameter	Default	Description
-dynamic-makeidx	1	When set to **0**, turns off making the dynamic cache idx file (no purge). A date-based purge of "aged" files is performed whenever the index file is saved. (See *-dynamic-maxdays*).
-dynamic-prefix	"/RESOURCE"	HTTP upstream URL prefix (append Domain.Class.OID).
-dynamic-proxy-host		To pass through an Internet proxy machine or a Firewall proxy when connecting to the Dynamic Cache Upstream Host machine, enter the Internet proxy's hostname or IP address.
-dynamic-proxy-pass		To pass through an Internet proxy machine or a Firewall proxy when connecting to the Dynamic Cache upstream host machine, enter the connecting User's password.
-dynamic-proxy-port		To pass through an Internet proxy machine or a Firewall proxy when connecting to the Dynamic Cache Upstream Host machine, enter the Internet Proxy port number to use.
-dynamic-proxy-user		To pass through an Internet proxy machine or a Firewall proxy when connecting to the Dynamic Cache Upstream Host machine, enter the connecting User name.
-dynamic-root	*See description.*	The fully-qualified location to store the dynamic cache. For example: "C:/Novadigm/IntegrationServer/data/rps/dynamic"
-dynamic-savefreq	90	When –dynamic-savetod is disabled (-1), specifies how often (in minutes) to save index file and cleanup cache. **The save index file process is skipped when there are no changes to be applied.**
-dynamic-savetod	-1	Specifies time of day (hh:mm:ss) to save dynamic index file and cleanup cache each day. Overrides –dynamic-savefreq. Disabled when set to **–1**. T**he save index file process is skipped when there are no changes to be applied.**
-dynamic-trace	0	When set to **1**, information is recorded for diagnostic tracing.

Table 2.2 ~ Configuration File rps.cfg Parameters

Parameter	Default	Description
-dynamic-url	"http://*upstream*:3466"	HTTP upstream URL (append prefix). Replace *Upstream* with the upstream host the Radia Proxy Server makes a connection with for obtaining dynamic cache. The upstream host can be Radia Proxy Server co-located with the Radia Configuration Server, or another Radia Proxy Server. To change the default port (3466) modify the PORT setting in the httpd.rc file located in the **/etc** directory where you installed the Radia Proxy Server. Optionally, define multiple upstream hosts for fail-over support. Enclose all entries in one set of quotation marks. Space separate each upstream host. For example: `"http://upsteam1:3466 http://upstream2:3466"`
-httpd	1	When set to **1**, HTTP front end is enabled, supporting HTTP communication with Radia clients. Do not change.
-httpd-prefix	"/RESOURCE"	The URL prefix registered to Radia Integration Server. Do not change.
-httpd-trace	0	*This parameter is reserved for future use.*
-stager	0	Staging on or off. When set to **1**, the stager (native TCP/IP) front end is enabled. Not recommended for use with Radia 3.x clients. Typically used to transition only from Radia 2.x to Radia 3.x.
-stager-addr	0.0.0.0	Restricts the IP address used by the Radia Proxy Server (if using multiple IP addresses on one computer).
-stager-port	3461	The listening port used by the stager front end.
-stager-trace	0	When set to **1**, information is recorded for diagnostic tracing.
-static-host	xxx.xxx.xxx.xxx	Agent upstream host (example, Radia Configuration Server). Used for initiating the Preloader connect.
-static-logsize	1000000	Specifies the size of the Preloader connect.log file in bytes. Connect.log is in the Radia Integration Server's **/logs/rps** folder. When the logsize is reached, a backup file (.BAK) is created. By default, this file is connect.bak. If a backup file already exists, it will be overwritten.
-static-port	3464	Agent upstream TCP port. Used during preload.
-static-proxy-host		To pass through an Internet proxy machine or a Firewall proxy when connecting to the Agent upstream host during the Preload, enter the Internet proxy's hostname or IP address.
-static-proxy-pass		To pass through an Internet proxy machine or a Firewall proxy when connecting to the Agent upstream host during the Preload, enter the connecting User's password.

Table 2.2 ~ Configuration File rps.cfg Parameters

Parameter	Default	Description
-static-proxy-port		To pass through an Internet proxy machine or a Firewall proxy when connecting to the Agent upstream host during the Preload, enter the Internet Proxy port number.
-static-proxy-user		To pass through an Internet proxy machine or a Firewall proxy when connecting to the Agent upstream host during the Preload, enter the connecting User name.
-static-root	*See description*	The fully qualified location of the Preloader's data store (IDMDATA). For example: "C:/Novadigm/IntegrationServer/data/rps/static"
-static-type	agent	When set to **agent**, the static cache is populated by the Preloader. Set to **server** to point the cache to a native Radia Configuration Server Database (where the Radia Proxy Server is installed on the same computer as the Radia Configuration Server).
-static-ssl	0	When set to **1**, indicates the Preloader is to use SSL.
-static-trace	0	When set to 1, information is recorded for diagnostic tracing.
-static-user	RPS	Agent upstream identity on the Radia database in Policy.User (ZUSERID). Used during preloading. The services connected to this user ID will be preloaded into the static cache on the Radia Proxy Server.

Editing the RPS.CFG File

To edit the RPS.CFG File

Before modifying the rps.cfg file, stop the service for the Radia Proxy Server. See *Starting and Stopping the Radia Proxy Server for Windows* on page 98 or *Starting and Stopping the Radia Proxy Server for UNIX* on page 55.

1. Make your modifications to the rps::init section of the rps.cfg file.
2. After making your modifications, restart the service.

Syntax Notes

- When the following parameters are disabled (that is, set to 0), all other options related to the parameter are disregarded:

  -dynamic
  -httpd
  -stager

For example, if –dynamic is 0, the entire set of –dynamic* parameters are disregarded.

- Use quotation marks to enclose entries that include special characters or spaces. For example, the following –dynamic-root entry uses quotation marks to enclose the fully-qualified location of the dynamic cache store.

  -dynamic-root "C:/Novadigm/IntegrationServer/data/rps/dynamic"

- Use slashes [/] to specify the paths for the –dynamic-root and –static-root parameters in both Windows and UNIX environments. For example:

  -dynamic-root "C:/Novadigm/IntegrationServer/data/rps/dynamic"

Configuring the Dynamic Cache Parameters

When dynamic caching is enabled, client requested resources not available on the Radia Proxy Server's local cache can be requested on demand from a designated upstream host. The resources are downloaded from the upstream host using HTTP, placed in the dynamic cache of the Radia Proxy Server, and provided to the client.

Caution

Dynamic cache use is never recommended for a Radia Proxy Server co-located with a Radia Configuration Server. For details, please see *Co-Locating a Radia Proxy Server with a Radia Configuration Server* on page 79.

The minimal rps.cfg entries required for dynamic caching are simply the -dynamic and -dynamic-url parameters, which enable the dynamic cache and define the upstream URL for obtaining the requested files. (The –dynamic-root location is set during the install.)

```
rps::init {
...
   -dynamic  1
   -dynamic-url  http://<upstream>:3466
...
}
```

Figure 2.28 ~ Minimal rps.cfg entries required for dynamic caching.

The balance of the dynamic cache entries are optional. If absent from the rps::init section, the default values are assumed. Use Table 2.3 as a guide to configuring the dynamic cache for your environment.

Table 2.3 ~ Dynamic Cache Parameter Summary*

Refer back to Table 2.2 ~ Configuration File rps.cfg Parameters for complete definitions of each parameter.

To do this:	Use these rps.cfg parameters
Specify an upstream host and http port; either a Radia Configuration Server enabled for HTTP downloads, or another Radia Proxy Server. Specify multiple hosts for fail-over support. **Required.**	**-dynamic 1** **-dynamic-url "http://*upstream:*3466"** or -dynamic-url "http://upstream:3466 http://upstream2:3466 http://upstreamN:3466"
Specify where to store the dynamic cache on the Radia Proxy Server.	-dynamic-root
Save the index file that is used to track when files were last requested. A save also purges "aged" dynamic cache files not requested in a specific number of days. Schedule the index file save daily or every nn minutes. **Note:** A scheduled index file save (and thus the data-based purge) is skipped if there are no changes in the index file at that time.	-dynamic-maxdays -dynamic-makeidx -dynamic-savefreq -dynamic-savetod
Set a maximum cache size. If the maximum cache size is exceeded, the least recently used files are deleted until the maximum size is reached. Options: 1. Set a low-water mark (freespace) for a size-based purge. 2. Define 'large files' to be exempt during the first pass of the size-based delete.	-dynamic-freespace -dynamic-maxsizeMB -dynamic-largefile-size -dynamic-largefile-mindays
Revise the safeguard against purging the dynamic cache stored on a shared resource, such as the Radia Configuration Server.	-dynamic-allow-shared-resource-purge
Set diagnostic tracing.	-dynamic-trace
Provide server, port, user, and password information to pass through an Internet Proxy or Firewall required to connect to the upstream host. Also, enable SSL protocol.	-dynamic-proxy-host -dynamic-proxy-port -dynamic-proxy-user -dynamic-proxy-pass -dynamic-ssl
Override the default prefix.domain, and class OID for obtaining resources from the upstream URL. Not normally needed.	-dynamic-prefix -dynamic-defdmn -dynamic-defcls

Coding Multiple Upstream Hosts for Dynamic Cache Fail-over Support

When dynamic cache is enabled, the upstream host and port are defined in the **–dynamic-url** parameter of the rps.cfg configuration parameter. Multiple hosts may now be specified in the –dynamic-url parameter, using a blank separated list, to provide for a second, third, or more, upstream host with which to attempt to connect if a connection with the previously listed host fails. If all defined hosts in the list fail, an error will be returned to the Radia Proxy Server user.

For example, if you code the –dynamic-url as:

```
-dynamic-url   "http://111.111.111.11:3466 http://upstream2:3466 http://upstream3:3466"
```

the Radia Proxy Server will first attempt to connect to http://111.111.111.11 on port 3466 to obtain its dynamic cache. If that connection fails, it will attempt to connect to the second upstream host in the list: http://upstream2 on port 3466. If the second connection fails, it will attempt the third entry: http://upstream3 on port 3466. If the third connection fails, an error message is generated.

Note

Use one set of quotation marks to enclose the entire list of upstream hosts. Otherwise, you will receive a syntax error.

The Date-Based Purge of the Dynamic Cache

Every time a file is requested from the Dynamic Cache, the request date is recorded and maintained. If a file has not been requested in a pre-defined maximum number of days, it is deleted whenever a date-based purge occurs. Use the **-dynamic-maxdays** parameter in rps.cfg to specify the maximum number of days to keep unrequested files in the dynamic cache. The default is 7 days.

A date-based purge can be triggered in the following ways:

1. Explicitly from the Radia Management Portal by running the task: **Purge Dynamic Cache**. See the later topic, *Purging the Dynamic Cache using the Radia Management Portal* on page 106.
2. Whenever the dynamic cache index file is saved. For details, see the following topic, *Saving the Index File*.

Saving the Index File

An index file is maintained to keep track of when files in the Dynamic Cache were last used. Every so often this index file is saved to update the “last-used” date in the dynamic cache files. Whenever the index file is saved, a date-based purge of the dynamic cache also takes place to cleanup “aged” files.

The schedule for saving the index file is established using either the **–dynamic-savetod** or **–dynamic-savefreq** parameters. Use the **-dynamic-savefreq** parameter to schedule the index file save process every *nn* minutes.

Important Note

For a date-based purge to occur, the index file save process must run. The index file save process is **skipped** when there are no changes to be applied to it. Therefore, if a dynamic cache has had no resources requested of it or added to it since a previous save, the index file save is skipped and the date-based purge is also skipped.

To use either of these options, first ensure the following dynamic cache parameters are specified:

```
-dynamic                 1
-dynamic-makeidx         1
-dynamic-maxdays        <max days to hold unrequested files>
```

- Use the **-dynamic-savetod** parameter to set a daily schedule for saving the dynamic index and purging the dynamic cache. A -dynamic-savetod entry overrides a -dynamic-savefreq entry. Valid values are –1 (disabled) or *hh:mm:ss*. Seconds may be omitted.

  Example: To set a time-of-day for the purge to occur daily, use the **-dynamic-savetod** parameter with the time specified in the format hh:mm:ss. For example, the following set of parameters will automatically run a purge of the dynamic cache each day at 3:00 AM to delete files that have not been requested for seven days.

```
-dynamic                 1
-dynamic-makeidx         1
-dynamic-maxdays         7
-dynamic-savetod         3:00
```

- If -dynamic-savetod is not specified, use the **-dynamic-savefreq** parameter to specify how often (in minutes) to save the index file and purge the dynamic cache.

  Example: The following set of parameters automatically run a purge of the dynamic cache every 120 minutes for files unrequested for seven days.

```
-dynamic                 1
-dynamic-makeidx         1
-dynamic-maxdays         7
-dynamic-savefreq      120
```

Specifying a Size-Based Purge of the Dynamic Cache

Use the following options to automatically have files deleted from the dynamic cache to keep its size in check.

- **Automatically run a size-based purge when the maximum size is exceeded.**

The maximum size of the dynamic cache is specified in the **–dynamic-maxsizeMB** parameter in rps.cfg. When this maximum size is exceeded, a size-based purge of the dynamic cache will run automatically, purging the least recently used files until the target size is reached.

For example, the following entries specify an automatic purge of the least recently used files if the dynamic cache exceeds 300 MB. The purge process deletes the least recently used files until the cache is below the maximum size, or, to the size required by the new -dynamic-freespace parameter *(next feature)*.

```
-dynamic              1
-dynamic-maxsizeMB  300
```

- **Define a "freespace" amount to be available after a size-based dynamic cache purge**

  When a size-based purge is triggered, you may specify a –dynamic-freespace option to purge the dynamic cache down to a low-water mark (as a percentage of the maximum file size). This allows you to bring the cache size down to a predefined manageable size whenever it reaches the maximum file size. Use this option to eliminate repetitive calls to the purge process if upstream downloads occur frequently.

  To specify the **–dynamic-freespace** parameter, define a percentage of free space required upon completion of a size-based purge. For example: the following options will trigger a purge of the dynamic cache when it reaches 300 MB, and will purge the least recently used files until the dynamic cache size is more than 10% below 300, or under 270 MB.

```
-dynamic                1
-dynamic-maxsizeMB     300
-dynamic-freespace      10   <percentage below the maxsizeMB after the purge>
```

- **Exempt large files from the first-pass of a size-based purge**

  Normally, when a size-based purge takes place, the least recently used files are purged from the dynamic cache first. If large files are purged and then later requested, an undue load could be placed on the network. Use the following two "large file" parameters to allow the purge process to skip over the files that meet these large file criteria during an initial purge.

```
-dynamic-largefile-size     <defines the min large file size, in bytes>
-dynamic-largefile-mindays  <defines the min days large files stay in cache>
```

  Example: The following entries define large files as "25 MB" and the minimum number of days to remain in the cache as 15. So, the first pass of the purge will exempt files 25 MB or larger that have remained in the cache less than 15 days, even if they meet the normal "least recently used" criteria.

```
-dynamic                       1
-dynamic-maxsizeMB           300
-dynamic-freespace            10
-dynamic-largefile-size      "25 MB"
-dynamic-largefile-mindays    15
```

If, after purging the non-large files, the desired (reduced) cache size is not achieved, then the purge process will be repeated without regard to the large file settings.

Configuring the Radia Proxy Server in an Internet Proxy (Firewall) Environment

The following examples illustrate how to use the set of –static-proxy* and –dynamic-proxy* configuration parameters that provide support for environments with an Internet proxy or firewall server. Use these parameters to define the host, port, user name and passwords needed to pass through the firewall or Internet proxy for a static cache preload or a dynamic cache request.

Examples of TCP Parameters for Static Cache Preloader

The static cache Preloader uses TCP to connect to the Radia Configuration Server for the initial resolution. In the examples below, **443** is defined in order to get through a firewall, which typically opens ports **80** and **443.**

- Simple Connection

```
-static-host   "<RCS IP Address or hostname">
-static-port   3464
```

- Connection through Web Proxy

```
-static-proxy-host     "<Web-Proxy IP address or Hostname>"
-static-proxy-port     8080
-static-host           "<RCS IP Address or Hostname>"
-static-port           443
```

- Connection through Web Proxy performing Basic User Authentication

```
-static-proxy-host    "<Web-Proxy IP address or Hostname>"
-static-proxy-port    8080
-static-proxy-user    "<Web-Proxy Username>"
-static-proxy-pass    "<Web-Proxy Password>"
-static-host      <RCS IP Address or Hostname>
-static-port      443*
```

Examples of HTTP Parameters for Dynamic Cache PassThru

The dynamic cache uses an HTTP port to connect to a Radia Configuration Server enabled for HTTP-download support, or another Radia Proxy Server. Additional dynamic cache parameters are available to pass through an Internet or Firewall Proxy, with or without basic user authentication. The following examples illustrate how to use these Firewall Proxy parameters.

- Simple Connection to upstream HTTP host and port

```
-dynamic          1
-dynamic-url      http://upstream:3466
```

- Connection through Web Proxy

```
-dynamic          1
-dynamic-url      http://upstream:3466
-dynamic-proxy-host   "<Web-Proxy IP Address or Hostname>"
-dynamic-proxy-port   8080
```

- Connection through Web Proxy performing Basic User Authentication

```
-dynamic          1
-dynamic-url      http://upstream:3466
-dynamic-proxy-host   "<Web-Proxy IP address or Hostname>"
-dynamic-proxy-port   8080
-dynamic-proxy-user   "<Web-Proxy-username>"
-dynamic-proxy-pass   "<Web-Proxy-Password>"
```

Co-Locating a Radia Proxy Server with a Radia Configuration Server for HTTP Support

The Radia Configuration Server no longer supports native HTTP download capability. When you use a Radia configuration that requires obtaining resources from the Radia database using HTTP, you need to co-locate a Radia Proxy Server on the same machine as the Radia Configuration Server. The co-located Radia Proxy Server is defined with a static type of **server**, instead of agent. This means it does not store its own static cache, but merely points to that of the Radia Database. It does not need to be preloaded. Configurations that Require a Co-located Radia Proxy Server

You need a Radia Proxy Server co-located with the Radia Configuration Server to provide HTTP download support in the following configurations:

- To support Radia clients configured to obtain their resources from the Radia database using HTTP, instead of TCP/IP.
- To support a subordinate, or downstream, Radia Proxy Server, configured to obtain its dynamic cache from the Radia database.
- To preload a subordinate, or downstream, Radia Proxy Server from the Radia database using HTTP, instead of TCP/IP.

For details on each of these configurations, see *When to use a co-located Radia Proxy Server* on page 25.

Once you co-locate a Radia Proxy Server on the Radia Configuration Server, you can use its IP address and port number (generally: 3466) to obtain the Radia database resources using HTTP.

Installing and Configuring a Co-located Radia Proxy Server

Use these procedures to establish a co-located Radia Proxy Server with your Radia Configuration Server in order to service HTTP requests for resources to be downloaded from the Radia Database.

Note

The ability to service HTTP requests was previously provided through ZHTTPMGR, which is no longer supported. A co-located Radia Proxy Server provides improved performance and scalability over ZHTTPMGR for enabling the Radia Configuration Server for HTTP communications. Steps to remove ZHTTPMGR begin on page 83.

To install and configure a co-located Radia Proxy Server

1. Install the Radia Proxy Server on the same machine as the Radia Configuration Server. Accept the default values throughout the install prompts.

   For details, see *Radia Proxy Server Installation* topics on page 34.

2. For Windows, stop the Radia Integration Service (which starts automatically after the Radia Proxy Service is installed).

3. Modify the following RPS.CFG file parameters using the guidelines given in Table 2.4. The rps.cfg file is located in the **/etc** folder of the base install directory.

Table 2.4 ~ Configuring the RPS.CFG for a Co-located RPS

Parameter	Default	Set To	Description
-static-root	See description	See description	The fully qualified location of the Preloader's data store. Enter the location of the Radia Database. Use slashes (/) as delimiters in the entry for Windows as well as UNIX. By default, this location is: Windows: C:/Novadigm/ConfigurationServer/DB UNIX: /opt/Novadigm/ConfigurationServer/DB
-static-type	agent	server	Set to **server** to point the static cache to a native Radia Database.
-dynamic	0	0	Use of the dynamic cache is NOT RECOMMENDED for a co-located Radia Proxy Server. For details, see Safeguarding the Radia Database from a Dynamic Cache Purge on page 82

Refer to Table 2.2 ~ Configuration File rps.cfg Parameters on page 69 for a complete listing of rps.cfg parameters.

The following figure shows a sample rps.cfg file configured for a Radia Proxy Server co-located with the Radia Configuration Server.

```
rps::init {
    -httpd             1
    -httpd-prefix      "/RESOURCE"
    -static-root       "C:/Novadigm/ConfigurationServer/DB"
    -static-type       server
    -static-trace      0
    -static-host                              <leave default>
    -static-port       3464
    -static-user       RPS
    -static-ssl        0
    -dynamic           0
    -dynamic-root                               <leave default>
    -dynamic-url       http://upstream:3466     <leave default>
    -dynamic-trace     0
    -dynamic-maxsizeMB 0
    -dynamic-makeidx   0
}
#
# END OF CONFIG
#
# DO NOT REMOVE
#
rps::start
```

Figure 2.29 ~ Sample Configuration for a Radia Proxy Server co-located with the Radia Configuration Server.

4. Restart the co-located Radia Proxy Server.
5. If necessary, switch the dynamic cache upstream port number (specified in the -dynamic-url parameter of rps.cfg) for any Radia Proxy Servers that were previously pointing to the Radia Configuration Server port as its upstream host. These Radia Proxy Servers should now point to the co-located Radia Proxy Server port (3466) as their upstream port.

Safeguarding the Radia Database from a Dynamic Cache Purge

The recommended approach for configuring a Radia Proxy Server co-located with the Radia Configuration Server is to point the static cache at the Radia Database, set the -static-type to "server", and disable the dynamic cache.

The Radia Proxy Server includes automatic protection to guard against purging files from the Radia Database in the event that a co-located Radia Proxy Server's dynamic cache is enabled. The **–dynamic-allow-shared-resource-purge** parameter, when set to **0** (default), prevents Radia Database files from being deleted during any dynamic cache purge process. This might occur when a co-located Radia Proxy Server is unusually configured (against recommendation) as follows:

- dynamic cache is enabled
- dynamic cache root points to the Radia Database.

Using the default (**-dynamic-allow-shared-resource-purge** set to **0**) protects your Radia Database from any dynamic cache purges.

Setting the parameter to **1** removes the safeguard and allows a purge of dynamic-cache files from the Radia Database.

Removing the Use of ZHTTPMGR for HTTP Support

If you previously used ZHTTPMGR to enable your Radia Configuration Server for HTTP, you should remove that support and replace it with a co-located Radia Proxy Server.

To remove ZHTTPMGR support from a Radia Configuration Server

1. Locate the Radia Configuration Server Settings file, EDMPROF. The specific EDMPROF file name and location for Unix and Windows are listed below:
   - On UNIX operating systems, it is **.edmprof**, located on the **home** directory of the UNIX user ID that installs, starts, stops, and maintains the Radia Configuration Server.
   - On Windows operating systems, it is **edmprof.dat**, located in the **bin** folder of the Radia Configuration Server directory.

   If a Radia Configuration Server was enabled for HTTP support through the use of ZHTTPMGR, the EDMPROF file will have the following entries in the MGR_ATTACH_LIST and MGR_HTTP sections:

   ```
   [MGR_ATTACH_LIST]
   CMD_LINE=(zhttpmgr ADDR=<RCS IP Addr, PORT=8080) RESTART=YES

   [MGR_HTTP]
   HTTP_PORT=8080
   ```

2. In the [MGR_ATTACH_LIST] section, comment out the CMD_LINE entry for zhttpmgr.
3. In the [MGR_HTTP] section, comment out the HTTP_PORT entry.

Configuring the Radia Database for the Static Cache Preload

Each Radia Proxy Server whose static type is set to **agent** (that is, not a Radia Proxy Server co-located with the Radia Configuration Server) requires a Preload distribution model defined in the Radia Database.

This Preload distribution model defines:

- The resources to be loaded onto the Radia Proxy Server's static cache when the Preload runs.
- Where the resources should be distributed from and under which protocol. By default, the resources are distributed from the host Radia Configuration Server using TCP. However, you can configure the distribution source to be a co-located Radia Proxy Server for HTTP download. Or, you can configure the distribution source to be another (preloaded) Radia Proxy Server. See *Preloading Deployment Options* on page 86.

- For Windows Installer Enabled Applications, the ACP resources to be loaded onto the Radia Proxy Server, but not distributed to the clients. See *Preloading Windows Installer Enabled Applications* on page 86

Use the following procedures to create a static cache distribution model in the Radia Database for each Radia Proxy Server to be preloaded. This can be done before or after the Radia Proxy Servers are installed and configured.

Creating a Distribution Model for a Static Cache Preload

To create a distribution model for the Preload of the Static Cache

1. Create a POLICY.USER instance in the Radia Database that matches the User ID entered during the Radia Proxy Server install, or later specified in the **–static-user** parameter in the **/etc/rps.cfg** file. The User ID default is RPS. However, the User ID is often set to: RPS_<*machine_name*>.
2. Connect the POLICY.USER instance to the set of applications to be preloaded to the Radia Proxy Server's static cache. For details on how to entitle a POLICY.USER instance to application services, see the *Radia Application Manager Guide*.

To configure a Radia Proxy Server to preload the all database applications

The following steps serve as one example of preloading a Radia Proxy Server. This example preloads the Radia Proxy Server with all available applications as well as HP applications. Your preload should specify the set of applications normally required by the clients assigned to that Radia Proxy Server.

1. Create a Policy.Workgroup instance called NVD_STG_PRELOAD_APPLICATIONS with a friendly name NVD Radia Stager-Proxy Preload Applications.
2. Manually enter the following always 'offers' fields to preload all applications onto the Radia Proxy Server:

   ```
   SOFTWARE.ZSERVICE.*
   NOVADIGM.ZSERVICE.*
   ```

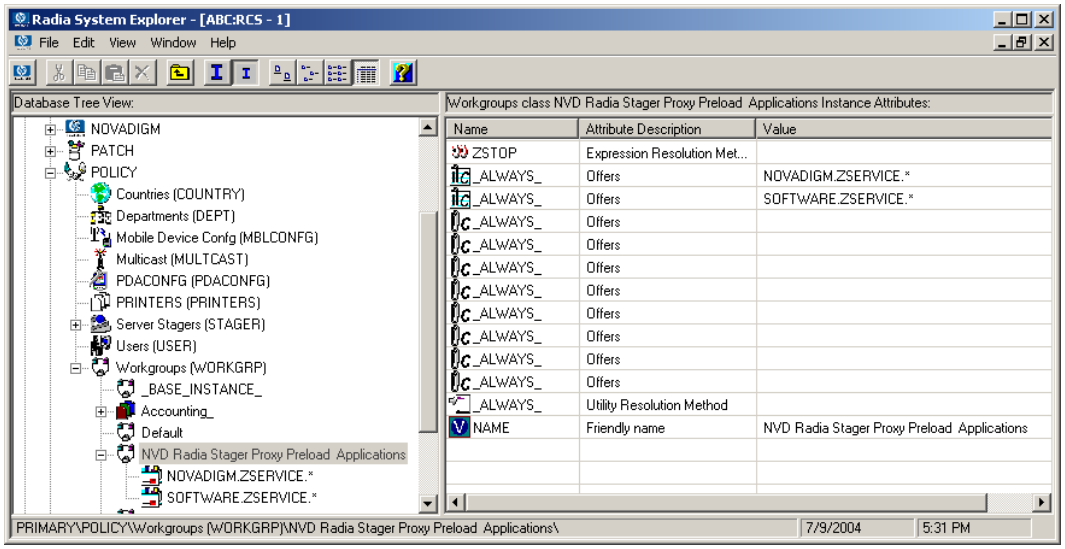


Figure 2.30 ~ Sample NVD Workgroup to entitle a Radia Proxy Server to application services.

3. Connect the Policy.User instance to this Policy.Workgroup instance.

To manually run the Preload, see *Testing the Radia Proxy Server Preload* on page 115.

To run the Preload from the Radia Management Portal, see *Preloading the Radia Proxy Server* on page 103.

Preloading Deployment Options

A Radia Proxy Server's preload resolution is always performed on the host Radia Configuration Server. However, you can direct the actual distribution of the static cache resources to come from another (previously preloaded) Radia Proxy Server, instead of from the Radia Configuration Server. Or, you can set the deployment protocol to use the HTTP port of a Radia Proxy Server co-located with the Radia Configuration Server. These deployment options are set the same way you configure the clients to obtain their resources from the Radia Proxy Server: by attaching a pre-configured STAGER instance to the Radia Proxy Server's User instance in the Radia database.

To change the deployment source or protocol for a Preload

1. Using the Radia System Explorer on the Radia Configuration Server, create a STAGER instance to be used to preload the Radia Proxy Server from another Radia Proxy Server, or to preload using the HTTP port of a co-located Radia Proxy Server.
2. Specify the following attributes in the STAGER instance:

Table 2.5 ~ Stager Instance Attributes needed to Preload a Proxy from co-located or remote Radia Proxy Server using HTTP

STAGER Attribute	Set to Value
ZPCPROTL	HTTP
ZDATAURL	http://<*Your RPS IP Addr or Hostname*>:3466 Replace <*Your RPS IP Addr or Hostname*> with the IP address or Hostname of the Radia Proxy Server to be used to obtain the static cache Preload files, in lieu of the Radia Configuration Server. For a co-located Radia Proxy Server, this will be the same address as the Radia Configuration Server.

3. Locate the POLICY.USER instance for the Radia Proxy Server. This is usually RPS or RPS_<*machine_name*>.
4. Connect a POLICY.STAGER instance to the RPS_<*machine_name*> instance in the POLICY.USER domain of the Radia Database.

The following Figure shows the STAGER instance STAGER_PROXY_01 connected to the RPS_Proxy_02 POLICY.USER instance. When the RPS_Proxy_02 server connects to the RCS for its Preload, its Preload distribution is resolved using the entries defined it its POLICY.USER instance. The connection to the STAGER_PROXY_01 instance means the Preload is deployed using HTTP from the Radia Proxy Server URL specified in the STAGER instance.

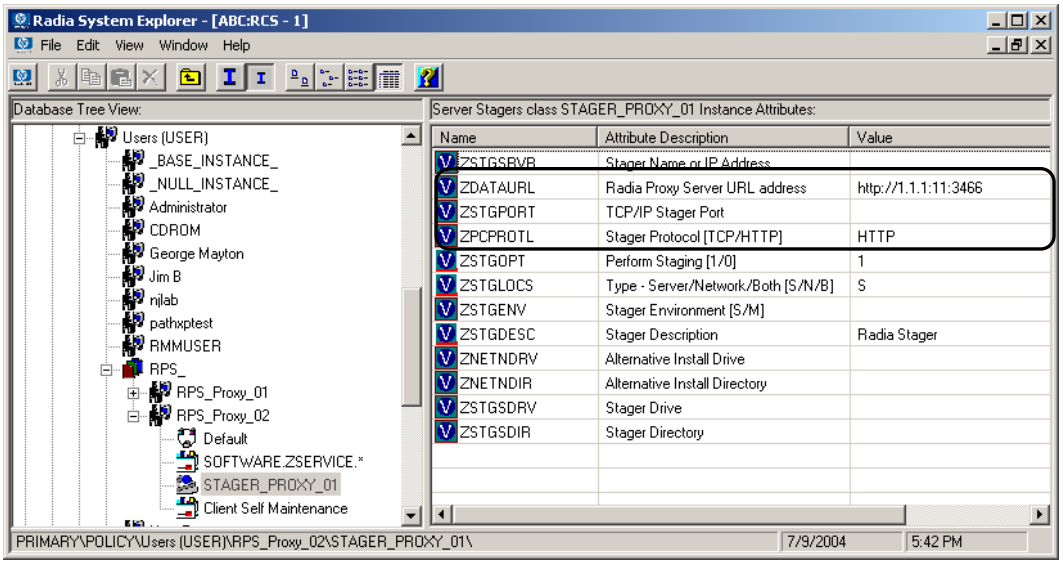


Figure 2.31 ~ STAGER instance to Preload from an HTTP port or another Radia Proxy Server.

Preloading Windows Installer Enabled Applications

There are two packages associated with each Windows Installer enabled application, the MSI package that includes the list of MSI Features, and the Administrator Control Point (ACP) package. The (ACP) package instance contains all of the file resources that are required by a Windows Installer product. (For more information on publishing Windows Installer Enabled applications, see the *Radia Application Manager Guide* or the *Radia Software Manager Guide*.) If you are using Radia Proxy Servers or Radia Staging Servers, you will need to deploy the ACP package to these two types of servers while installing only the MSI package on the client computer. There are three models for doing this:

- Create separate Application (ZSERVICE) instances for the MSI package and the ACP Package. Deploy the MSI Package to the client computers and the ACP Package to the Radia Staging Server and Radia Proxy Servers.
- Create separate Application (ZSERVICE) instances for the MSI and ACP packages. When creating the ACP packages, name them all with a suffix of _ACP. In a connection instance, type SOFTWARE.PACKAGE.*_ACP. This will connect all ACP packages to this one service. Deploy this service to your Radia Staging Servers and Radia Proxy servers only.

Caution

Be sure that you have adequate disk space on your Radia Proxy Servers and Radia Staging Servers before connecting *all* of the ACP packages to this service. Depending on your network condition, loading this data may take a long time.

- Create one Application (ZSERVICE) instance for both the MSI and ACP packages. In the procedure below, a ZSTOP expression will be placed on the ACP package to prevent the client computers from downloading the ACP package. Radia Staging Servers and Radia Proxy Servers ignore expressions, and will install the ACP package.

To enable an ACP package for preloading only

1. Use Radia System Explorer to navigate to the ACP package that you want to preload.
2. Right-click on the ACP package and choose **Edit Instance** from the shortcut menu.
3. Click **ZSTOP000 Stop Resolution (000)** and type a value of **1**.

   Typing a value of **1** will prevent client computers from resolving the ACP package, but will allow the Radia Proxy Server or Radia Staging Server to preload the ACP package.

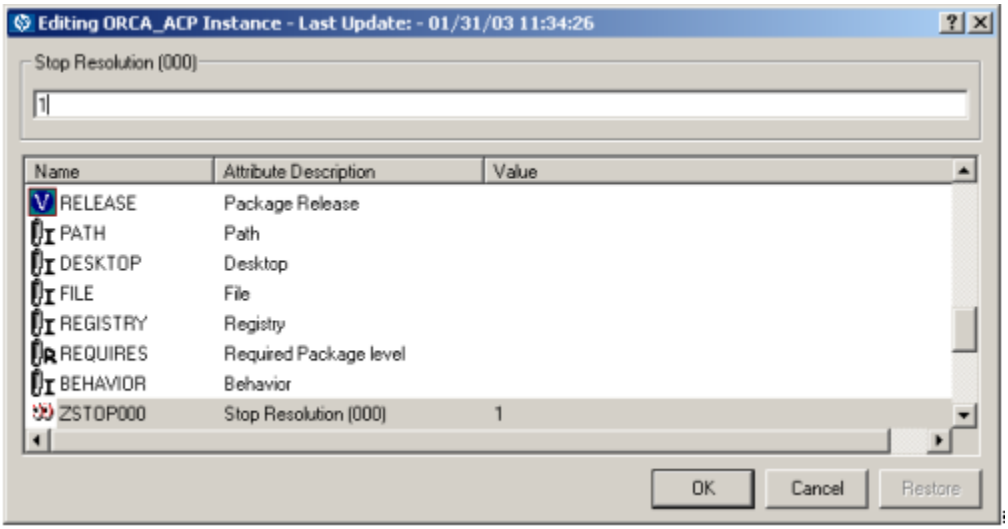


Figure 2.32 ~ Set the ZSTOP000 to 1.

4. Click **OK**.
4. Click **Yes** to confirm the change.
5. Connect the ACP package and the MSI package to the same service.

Configuring Radia Clients for Use with the Radia Proxy Server

The Radia Proxy Server functions as an extension of the Radia Configuration Server. When used, the Radia Proxy Server becomes the primary repository for Radia Client data. Once a Radia Client determines which resources it needs to achieve its 'desired state', it can request the resources from the Radia Proxy Server.

After installing a Radia Proxy Server, configure the desired set of Radia Clients to request their needed resources from the Radia Proxy Server instead of the Radia Configuration Server. This is done in the POLICY Domain of the Radia Configuration Server's database.

Using HTTP for Client Communication

When the Radia Proxy Server front-end protocol is HTTP, use these steps to configure your clients in the Radia database. The HTTP front-end is enabled by default in the rps.cfg file (that is: -httpd is set to 1).

> **Note**
>
> Although client requests can be made using HTTP or TCP/IP, HTTP is the recommended protocol for communication with the Radia Proxy Server. To use TCP/IP, see *Using TCP/IP for Client Communication* on page 92.

To configure Radia Clients for a Radia Proxy Server using HTTP

1. Using the Radia System Explorer, create and edit a POLICY.STAGER instance to define the Radia Proxy Server as the deployment source for subscribers.
2. Update the following STAGER class attributes to specify the Radia Proxy Server information, as follows.
   - **ZPCPROTL —** Set the value of ZPCPROTL to indicate HTTP is the protocol the clients are to use for communicating with the Radia Proxy Server.
   - **ZDATAURL —** When using HTTP, set the **ZDATAURL** field to store the URL and port for the Radia Proxy Server. Storing the Radia Proxy Server URL in the Radia Configuration Server Database allows the Radia Proxy Server IP address or Hostname to be centrally controlled. Anytime this IP address or Hostname changes, the ZDATAURL field must be updated.

   Thus, the recommended settings for enabling clients to request data from a Radia Proxy Server using HTTP are:

   ```
   ZPCPROTL = HTTP
   ZDATAURL = http://<Your RPS IP Address or Hostname>:3466
   ```

Replace *Your RPS IP Address or Hostname* with the IP address or Hostname of your Radia Proxy Server.

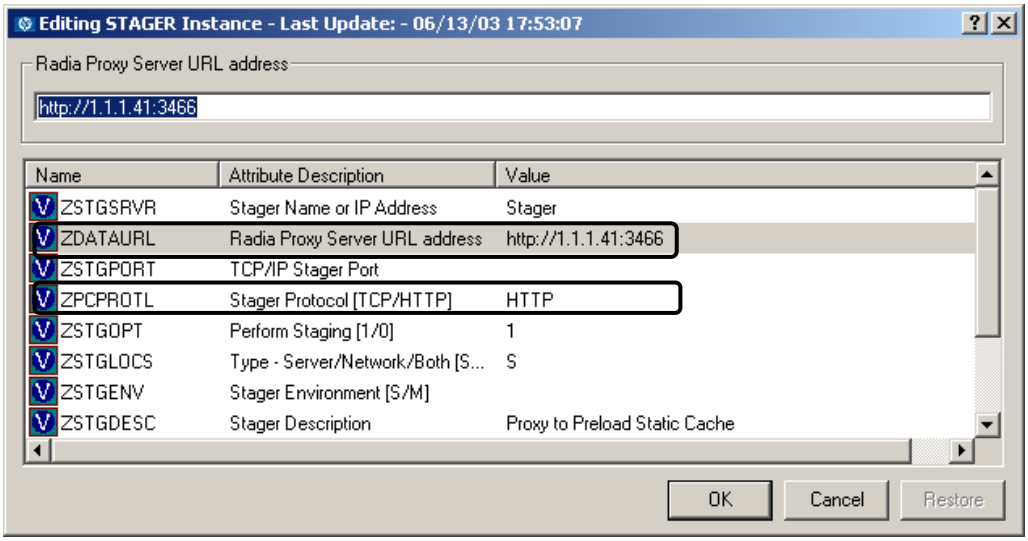


Figure 2.33 ~ STAGER instance attributes for a Radia Proxy Server serving clients using HTTP.

3. Connect the configured STAGER instance to the POLICY instances that represent the set of Radia Clients that will use this Radia Proxy Server. This is usually one or more workgroups or departments.

To connect a STAGER instance to a Workgroups instance

In the following example, we connect the Default Workgroup to a pre-configured STAGER instance so that *all* subscribers will receive applications from the Radia Proxy Server.

1. Open Radia System Explorer, and navigate to the PRIMARY.POLICY.STAGER class.
2. Double-click the **Server Stagers (STAGER)** to see the STAGER instances.
3. Double-click the **Workgroups (WORKGRP)** to see the WORKGRP instances.
4. Connect the predefined STAGER instance to the appropriate WORKGRP instance (Default, in our example).

In this example, all clients that are members of the Default Workgroup will request their resources from the Proxy Server named in the STAGER instance Proxy_01 using HTTP communication.

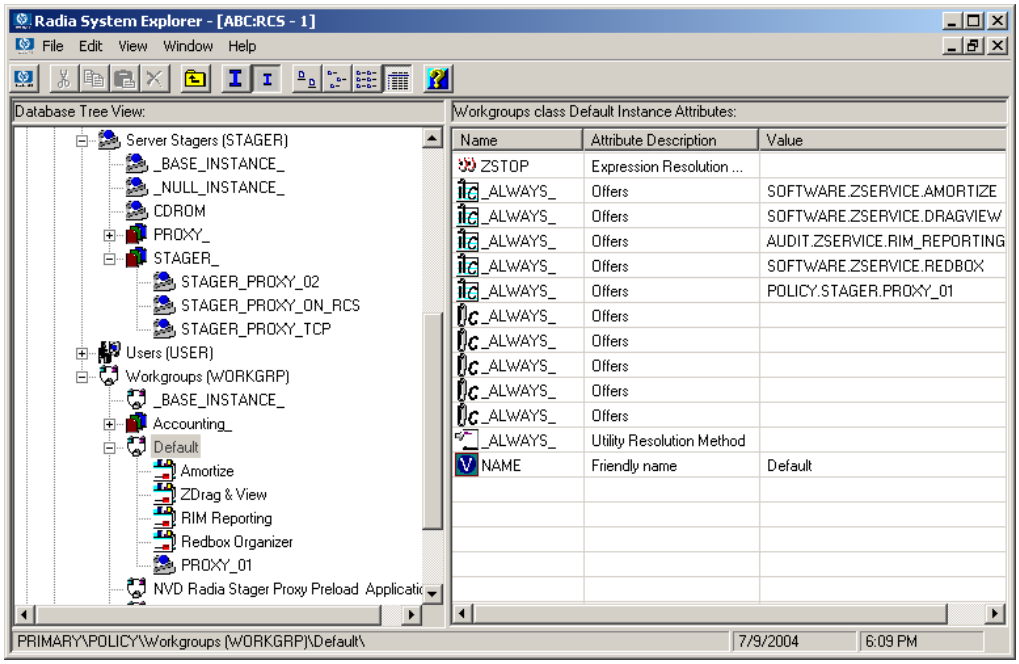


Figure 2.34 ~ Clients in the Default Workgroup will obtain resources from PROXY_01 using HTTP.

Using TCP/IP for Client Communication

Note

HTTP is the recommended protocol for client communication with the Radia Proxy Server. This topic discusses using the alternative TCP/IP protocol, when necessary.

Under certain circumstances, you may need to use the TCP/IP protocol to communicate with your clients. One example is when migrating from Radia 2.x to Radia 3.x Clients.

- To have the Radia Proxy Server communicate with clients using TCP/IP, you must enable the -stager front-end in the Radia Proxy Server Configuration file, rps.cfg. Set **–stager** to **1**. By default, the **–stager-port** listening port is set to **3461**. For details, refer to the –stager* parameters in Table 2.2 on page 69.
- To have the Radia Clients communicate with this Radia Proxy Server using TCP/IP, create a POLICY.STAGER instance in the Radia Database that sets the following definitions for the STAGER class attributes:

  ```
  ZSTGSRVR = <Radia Proxy Server IP Address or Hostname>
  ZDATAURL = leave blank
  ZSTGPORT = 3461 (the default Radia Proxy Server TCP/IP Port)
  ZPCPROTL = TCP
  ```

- If you are currently using TCP/IP to communicate between Radia clients and Radia Staging servers, and would like to continue using TCP/IP with the Radia Proxy Server, you need only change the IP address and port number of the Radia Staging Server to the IP address and port number of the Radia Proxy Server within Policy class. The policy instances will be unaffected.
  - Update the ZSTGSRVR attribute with the IP address of your Radia Proxy Server.
  - Update the ZSTGPORT attribute with the port number of your Radia Proxy Server for TCP, normally 3461.

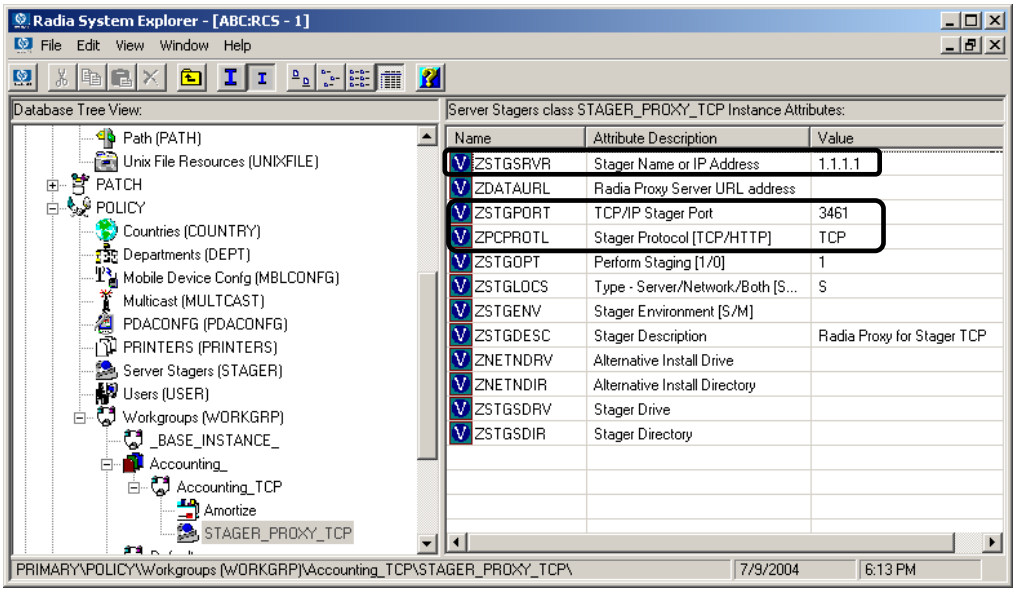


Figure 2.35 ~ STAGER instance attributes for Radia Proxy Server serving clients using TCP.

For more information about using the existing Radia Staging Requestor or modifying Stager instances, refer to the *Radia Staging Server Guide*.

Additional Configuration Topics

Enabling the Radia Configuration Server for HTTP Support

The Radia Configuration Server no longer supports downloading resources from the Radia Database using the HTTP protocol. Previously, this was provided through the ZHTTPMGR method. To obtain database resources from the Radia Database using HTTP, use a Radia Proxy Server co-located with the Radia Configuration Server.

For details, see *Co-Locating a Radia Proxy Server with a Radia Configuration Server* on page 79.

Summary

- You can install the Radia Proxy Server locally or remotely.
- The Radia Proxy Server component must be installed separately on each server to be used as a Radia Proxy Server.
- The Radia Integration Server is a foundation component of the Radia Proxy Server. Start or stop the Radia Integration Service (httpd) to start or stop the Radia Proxy Server.
- Review the rps.cfg in the /etc folder of the Radia Proxy Server install directory after an installation. Modify the parameters to set options, such as enabling and configuring the dynamic cache.
- The Radia Configuration Server no longer supports the HTTP protocol. When you need to obtain resources from the Radia Database using HTTP, install a Radia Proxy Server co-located with your Radia Configuration Server. The co-located Radia Proxy Server and port become the source for downloading resources from the Radia database using HTTP.
- Create a preload distribution model in the Radia Database for each Radia Proxy Server in your infrastructure installed remotely from the Radia Configuration Server.
- The Radia Clients must be configured to communicate with the Radia Proxy Server. The default and recommended protocol is HTTP. TCP/IP is also supported.

3

Radia Proxy Server Administration

At the end of this chapter, you will:

- Know how to start and stop the Radia Proxy Server.
- Know how to preload the Radia Proxy Server from the Radia Management Portal.
- Know how to purge the dynamic cache from the Radia Management Portal.
- Know how the Radia Proxy Server can be used in conjunction with the Radia Management Portal to deploy Radia client installs.
- Know which logs to use to troubleshoot a Radia Proxy Server.

There may be special circumstances involved in your Radia Proxy Server implementation. This chapter explores these possible situations:

- Starting and stopping the Radia Proxy Server.
- Populating the Radia Proxy Server before any clients try to install software.

Starting and Stopping the Radia Proxy Server for Windows

Since the Radia Proxy server is a loaded module under the control of the Radia Integration Server, in order to start and stop the Radia Proxy Server, start and stop the Radia Integration Server service on your computer. To do so, use the Service Control Manager window, as you would with any other Windows services. The Radia Integration Server service is started automatically after the Radia Proxy Server is installed on Windows platforms.

On UNIX platforms, the Radia Integration Server service is not started automatically after installation, it must be started manually after the installation program is finished. See *Starting and Stopping the Radia Proxy Server for UNIX* on page 55 for instructions.

For this example, we are using Windows 2000.

To start the Radia Proxy Server service

1. Go to **Start**, **Programs**, **Administrative Tools**, and then **Services**.
2. The **Service Control Manager** window opens.

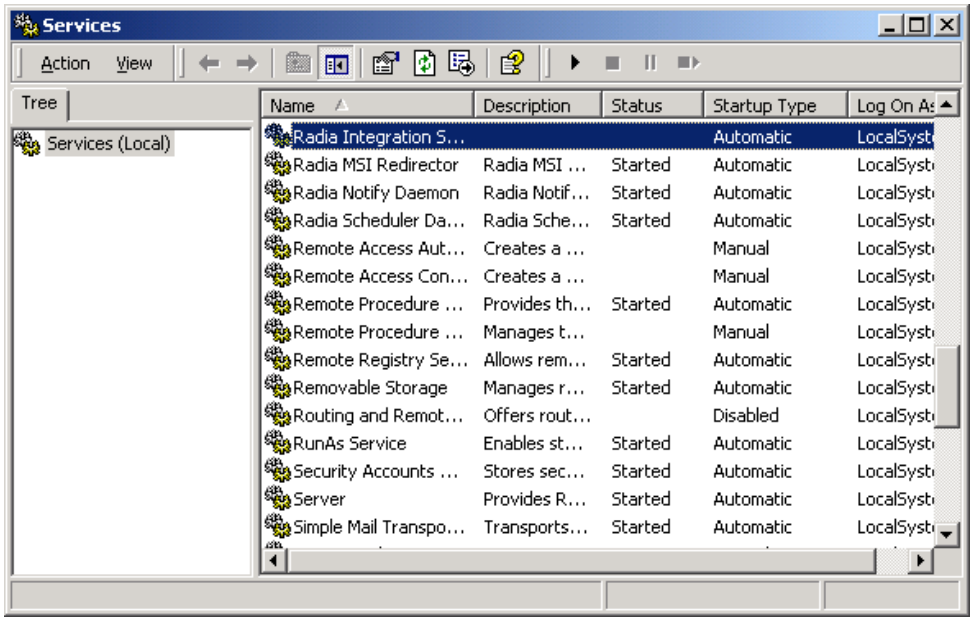


Figure 3.1 ~ Service Control Manager window.

3. Right-click on the service **Radia Integration Server**, and select **Start** from the shortcut menu that opens.

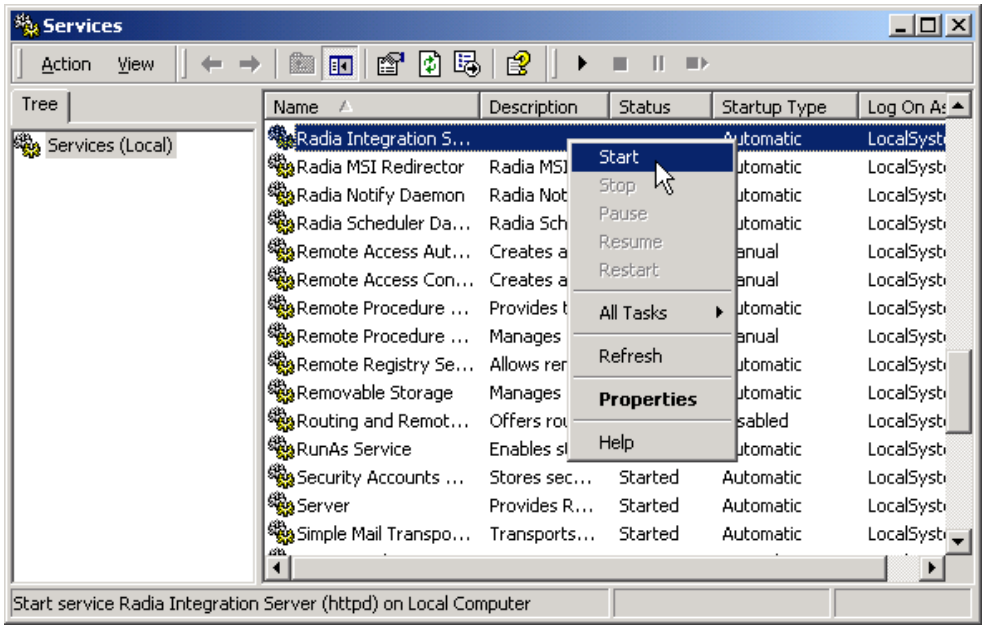


Figure 3.2 ~ Start the Radia Integration Server service.

The Radia Proxy Server is now running on your computer.

To stop the Radia Proxy Server service

1. Go to **Start**, **Programs**, **Administrative Tools**, **Services**.
2. The **Service Control Manager** window opens.

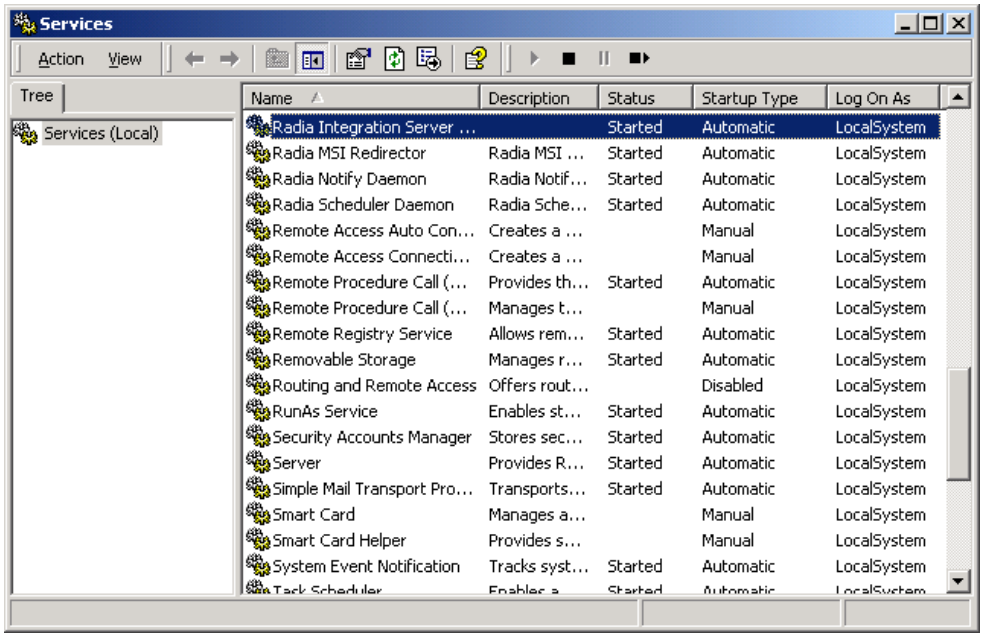


Figure 3.3 ~ Radia Integration Server service started.

3. Right-click on the service **Radia Integration Server**, and then select **Stop** from the shortcut menu that opens.

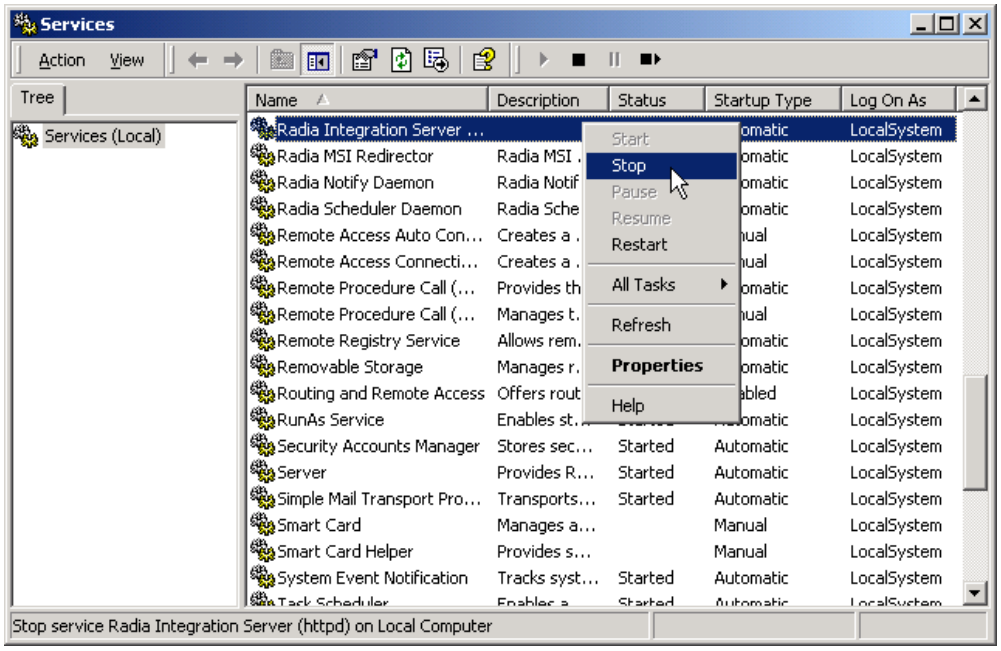


Figure 3.4 ~ Stop the Radia Integration Server service.

4. The Radia Proxy Server has been stopped.

> **Note**
>
> You can also use the Radia Management Portal to start and stop the Radia Proxy Server. Refer to the *Radia Management Portal Guide* for detailed information. There are separate guides for *Windows* and *UNIX.*

Radia Management Portal and the Radia Proxy Server

The Radia Proxy Server is one of the managed assets under the control of the Radia Management Portal. You can use the Radia Management Portal to remotely manage your Radia Proxy Servers. The *Radia Management Portal Guide* contains detailed information about managing your Radia Proxy Server remotely.

In addition to starting, stopping, and preloading the Radia Proxy Server, there are additional tasks available through the Radia Management Portal.

- **Activity Log Collection**
  Tracks all activity for the Radia Proxy Server.
- **Statistical Data Collection**
  Tracks historical data and verifies the age of the data cache.
- **Dynamic Cache Purge**
  Every time a file is requested, the request date is recorded and maintained. If a file has not been requested in a pre-defined maximum number of days, it is deleted. This purging of "aged" files occurs whenever the index file is saved, or, it can be initiated through the Radia Management Portal.
- **Preloader Reporting**
  Reports back to Radia Management Portal when preloading is finished.

> **Note**
>
> The following section requires you to be familiar with the Radia Management Portal. For detailed instructions and information, see the *Radia Management Portal Guide.* There are separate guides for Windows and UNIX.

Preloading the Radia Proxy Server

After installation, through the Radia Management Portal, you can choose to preload your Radia Proxy Server with services defined in the Radia Database. These services will be copied to the static cache located on the Radia Proxy Server.

> **Note**
>
> For a discussion of how to prepare Windows Installer Enabled applications for preloading to a Radia Proxy Server, see *Preloading Windows Installer Enabled Applications* on page 87.

Synchronizing (preloading) the Radia Proxy Server using the Radia Management Portal

From any web browser use the Radia Management Portal's **Synchronize Proxy Server** task to force the Radia Proxy Server to connect to the Radia Configuration Server to preload the files to the static cache on the Radia Proxy Server.

To synchronize the Radia Proxy Server using the Radia Management Portal

1. Access the Radia Management Portal from any web browser.
2. Use the Navigation aid to select the Radia Proxy Server that you want to synchronize.

> **Tip**
>
> Use the **Radia Proxy Server** group in the **Zone, Cross References, Infrastructure Services** container to synchronize all proxy servers in a zone at once.

Figure 3.5 ~ Synchronize Radia Proxy Server.

3. In the **Operations** task list, click **Synchronize Proxy Server**.

The **Schedule** window opens.

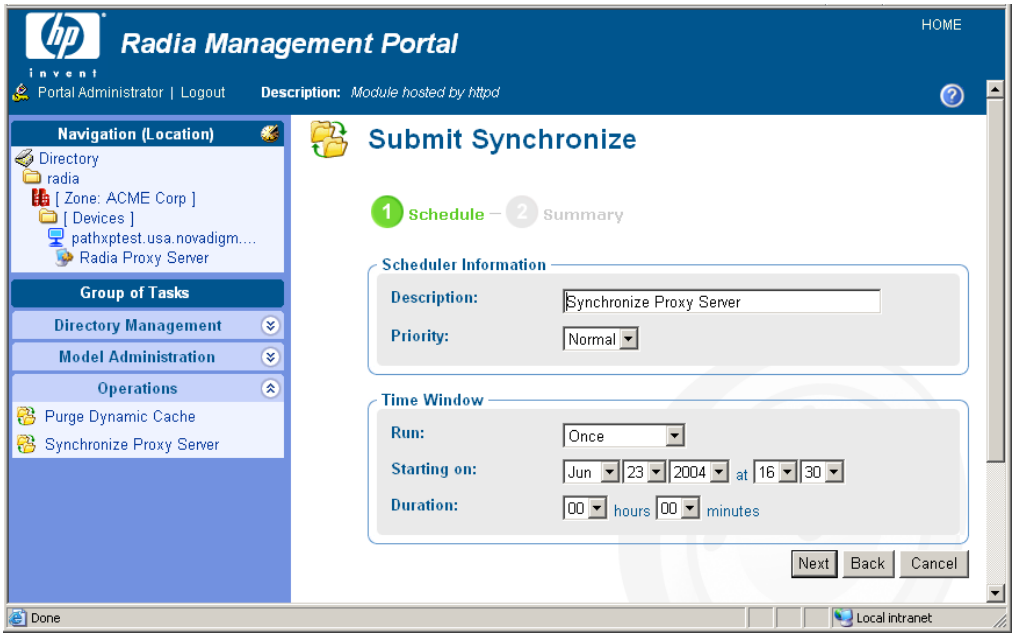


Figure 3.6 ~ Schedule window.

4. In the **Schedule** window, specify when you want this job to run.
5. Click **Next**.

The **Summary** window opens.

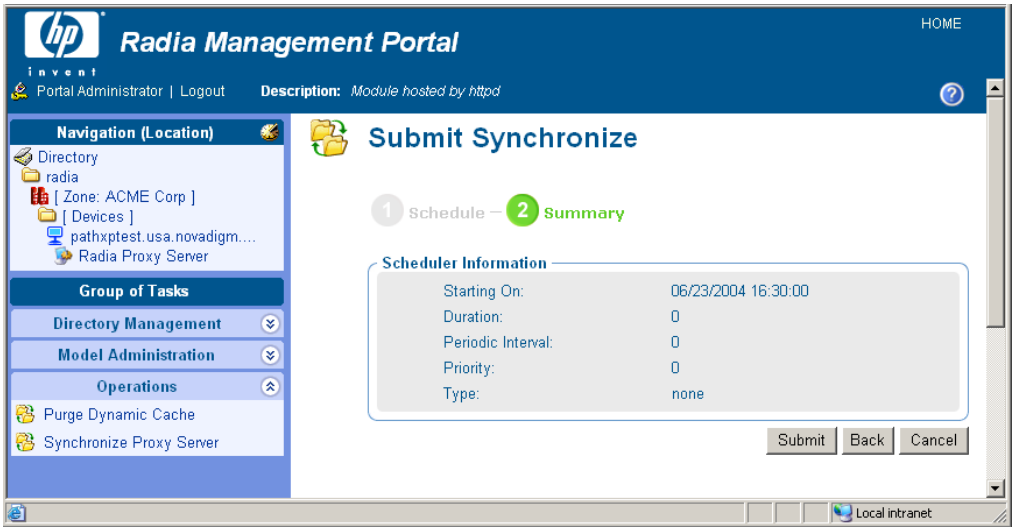


Figure 3.7 ~ Summary window.

6. Click **Submit**.

A list of the jobs appears.

The status of the synchronize proxy job will report the following events:

- Submission of the job request to the Proxy Server.
- Start of session between Proxy Server and Radia Configuration Server (for preloading the files to the static cache on the Radia Proxy Server).
- Job completed.

Purging the Dynamic Cache using the Radia Management Portal

Use the **Purge Dynamic Cache** task of the Radia Management Portal to purge the dynamic cache of the Radia Proxy Server of "aged" files.

See the Radia Management Portal Guide for more information.

Note

The same purge process occurs whenever the dynamic cache index file is saved. To set the dynamic cache parameters that schedule a save of the index file, see *The Date-Based Purge of the Dynamic Cache* on page 75.

To purge the dynamic cache of the Radia Proxy Server

1. Access the Radia Management Portal, and use the Navigation aid to select the Radia Proxy Server whose cache you want to purge.

Note

To schedule a dynamic cache purge of all Radia Proxy Servers in your network at once, go to the **Radia Proxy Server** group in the **Zone, Cross References, Infrastructure Services** container to begin the task. When your audience is a group of devices, the Query window is bypassed.

Figure 3.8 ~ Location of a device's Radia Proxy Server.

2. In the **Operations** task group, click **Purge Dynamic Cache**.

   The **Schedule** window opens.

3. In the **Schedule** window, specify when you want this job to run. For more information, see *Scheduling Jobs* in the *Radia Management Portal Guide*.

4. Click **Next**.

   The **Submit Purge—Summary** window opens.

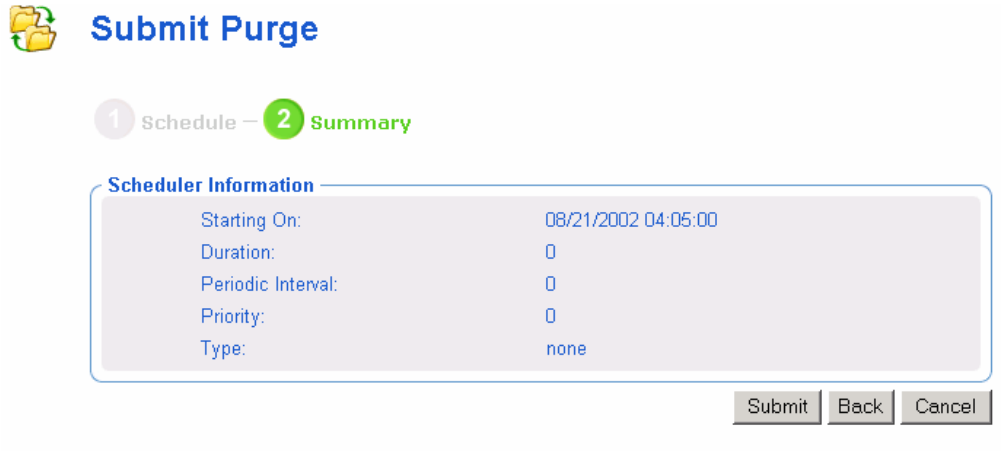


Figure 3.9 ~ Submit Purge—Summary window

5. Click **Submit**.

A list of the jobs appears. Now, you can use the **View Properties** task of the Radia Management Portal to view detailed information, such as the status of the job.

Using Proxy Servers to Install Clients Remotely from the Radia Management Portal

Just as Radia Proxy Servers are used to efficiently offload the distribution of applications to clients, the Radia Management Portal supports the use of Proxy Servers to efficiently offload the task of remotely installing clients.

This feature requires the following Radia components:

- Radia Management Portal (RMP) at version 1.2 or above
- Radia Proxy Server at version 1.1 or above, with an installed Radia Management Agent from RMP version 1.2 or above

From the Radia Management Portal, Administrators may use the **Manage Proxy Assignment** task to assign a set of manageable nodes (that is, computers which have a Radia Management Agent installed on them) to a Radia Proxy Server. This assigned proxy server becomes the code source for a later RMP request to install a Radia Client on the managed node. Thus, the workload of downloading scripts to install Radia clients remotely can be distributed between the Portal and a number of "proxy" servers.

All tasks are performed from the Radia Management Portal. For details, refer to the topics *Managing Proxy Assignments* and *Installing Clients* in the *Radia Management Portal Guide for Windows* or *Radia Management Portal Guide for UNIX*. An overview of the process involves five steps:

1. Use the **Add Group** task to make a group of all devices to be assigned to a Proxy Server *(Recommended)*.
2. Use the **Manage Proxy Assignment** task to assign the group of devices to the Proxy Server.

   At this time, the Radia Management Portal delivers the client install payload to the Proxy Server.
3. Use the **Install Client** task to schedule a job to have the Radia Client remotely installed on the group of devices.

   The Radia Management Portal schedules a client install job for each node in the group.
4. When each client install job executes, the node contacts the assigned Proxy Server.
5. The Proxy Server synchronizes its client payload with the Radia Management Portal using delta processing, and then deploys it to the node.

The initial delivery and delta synchronization of the client install payload from the Radia Management Portal to the Radia Proxy Server rely upon the underlying Radia Integration Server components of both the Portal and Proxy Servers.

Summary

- You can start and stop the Radia Proxy Server Service locally or remotely via the Radia Management Portal.
- You can purge the Radia Proxy Server's dynamic cache from the Radia Management Portal.
- You can preload the Radia Proxy Server's static cache using the Radia Management Portal.
- From the Radia Management Portal, you can designate the Radia Proxy Server to deliver client installs for an assigned set of computers. This feature uses the underlying Radia Integration Server component to synchronize the client payload with the Radia Management Portal and then deliver it to machines requesting the client installs.

4

Troubleshooting

At the end of this chapter, you will:

- Be familiar with the Radia Proxy Server log files.
- Be familiar with troubleshooting the most common error messages.
- Know how to collect the required files and version information when working with HP Technical Support.
- Know how to preload the Radia Proxy Server manually, from any web browser.

Radia Proxy Server Installation Directory

By default, the Radia Proxy Server is installed into the following directories:

Windows: **SystemDrive:\Novadigm\IntegrationServer**

UNIX: **/opt/Novadigm/IntegrationServer**

Note

Radia Proxy Server Release 1.0 for Windows installed into a different default directory: <*SystemDrive*>: \Novadigm\Radia Integration Server

About the Log Files

The Radia Proxy Server writes several logs, which can be used to track progress and diagnose problems. The log files are stored in the **logs** subdirectory of the Radia Proxy Server installation directory.

The log files are:

- **httpd-*port*.log**
  Replace *port* with your port number, for example, httpd-3466.log. . Logs the Radia Proxy Server activities of the TCL web server that it runs on. Can be found in the **logs** subdirectory of the Radia Proxy Server installation directory.
- **httpd-*port.YY.MM.DD*.log**
  This log contains the web server activity for each day. If the log is empty, it means that there was no activity that day.
- **httpd-3466.error.txt**
  This log contains messages written to any logs that contain the prefix **ERROR**. This allows you to view all errors in a single location.
- **CONNECT.LOG**
  Log created in the **\logs\rps** subdirectory of the Radia Proxy Server installation directory when the Proxy Server connects to the Radia Configuration Server to preload the static cache. Displays information related to the preload and the modules involved, such as RADCONCT.

  Each time you start the web server a new log is written. The old log is saved as httpd-*port.nn*.log.

Changing the Logging Level

By default the trace level is set to 3, which is the informational tracing level. This displays INFO, WARNING, and ERROR messages.

To change the log level, you can either run the following at a command prompt from the Radia Proxy Server installation directory:

nvdkit httpd.tkd –log_level 4

OR

Modify the log level in the httpd.rc configuration file found in the \etc folder of the Radia Proxy Server install directory. Use the following procedures.

To change the log level in the httpd.rc file

1. Stop the Radia Integration Service.
2. Open the file ***SystemDrive*:\Novadigm\IntegrationServer\etc\httpd.rc** for Windows, which is located on the computer that is running the Radia Proxy Server. The following is an excerpt from this file.

```
# Config Array
# Element   Default
# =======   =======
# HOST      [info hostname]
# PORT      3466
# DEBUG     0
# DOCROOT   [file join $home htdocs]
# IPADDR    {}
# WEBMASTER support@novadigm.com
# UID       50
# GID       100
# NAME      $tcl_service
# LOG_LEVEL 3
# LOG_LIMIT 7
#
Overrides Config {
    PORT    3466
    LOG_LEVEL    4
}
#
# (Re)Initialize Logging
#
Log_Init
```

Figure 4.1 ~ Excerpt of httpd.rc showing LOG_LIMIT increased to level 4.

3. Type LOG_LEVEL and the appropriate trace level, space delimited, within the **Overrides Config** starting and ending brackets **{ }.** Select the appropriate trace level, as follows.

Table 4.1 ~ Trace Levels

Trace Level	Description
0	No logging.

Table 4.1 ~ Trace Levels	
Trace Level	**Description**
1	Logs errors only.
2	Logs warnings and errors.
3	Logs informational messages, warnings, and errors. *Recommended trace level setting for customers.*
4	Logs all debug information. *Recommended for experienced customers only.*
5 – 9	Full trace *Not recommended for customer use.*

4. Save the file changes.
5. Restart the **Radia Integration Server** service.

Common Problems and Solutions

Performance Problems

Anti-virus software can reset IRPStackSize to a non-recommended, low value, causing performance issues for RIS (Radia Integration Server)-based products. If you are experiencing performance problems in a Windows environment, check that the IRPStackSize in the Windows Registry is set to an adequate value for your operating system. If IRPStackSize is set too low, it may severely impair your Radia Integration Server's network performance, especially under heavy load situations. On the Radia Proxy Server, for example, if IRPStackSize is set below the normal range for the operating system, application deployment will be slowed considerably.

To check for an adequate IRPStackSize value

Use your operating system's registry editor to check if the value of IRPStackSize is set too low. If it is, back up the Windows Registry and then increase the IRPStackSize value to be within the recommended range.

The IRPStackSize setting can be found in the following registry location:

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\LanmanServer\Parameters]
```

Note

For Windows XP and 2000, the IRPStackSize value is not created during the installation of Windows 2000 or Windows XP. If there is no value for it at the above location, the default value is used—which is no problem. If IRPStackSize is found at the above registry location, verify that it falls within the recommended values.

To obtain the recommended values for IRPStackSize as well as detailed instructions of how to change the IRPStackSize value, see the following Symantec and Microsoft documents:

- How to Change the IRPStackSize for Computers with Windows NT-Based Operating Systems
- IRPStackSize Parameter in Windows 2000
- Antivirus Software May Cause Event ID 2011

These links were active as of this writing.

Errors and Solutions

These topics address some of the errors you may find when analyzing a proxy server log.

- If you receive the following error, you are not properly licensed for the Radia Proxy Server (or the license expired) and will need to obtain the correct license file through Tech Support in order to run this product.

```
ERROR: "c:/novadigm/radia integration server/modules/rps.tkd", Not licensed for
this product
```

- If you receive the above error in the CONNECT.LOG during preload, you must check the Radia Configuration Server's log. The '650' error means something went wrong on the server side and the application was not delivered correctly.

```
[17:10:21 [RADCONCT / 000005a4] SYSTEM --- RADCONCT exit status [650]

[17:10:21 [RADCONCT / 000005a4] SYSTEM --- RADCONCT [Server stopped application
configuration.]

NVD000010A [radconnect_term] 17:10:21 [RADCONCT / 000005a4] SYSTEM --- RADCONCT
Return Code [650]

NVD000005E [radconct_cleanu] 17:10:21 [RADCONCT / 000005a4] SYSTEM --! RADCONCT
Exit code [650]
```

- If you receive the following error in the CONNECT.LOG during preload, you have not assigned any services/applications to the default RPS user instance. (Services will be preloaded whether they are Mandatory or Optional. Stop Expressions are also ignored.)

  If additional logging is need, you may be asked to increase the log level. You can change the log level at the command prompt by running "nvdkit httpd.tkd –log_level 4", or adding the log level to the httpd.rc configuration file found in the **/etc** folder.

```
13:43:36 Warning: RPS/Static: sync: Radskman rc:[109] [Presently there are no
applications available in the software catalog. Please contact your system
administrator for assistance.] (CHILDSTATUS -1 109)
```

Testing the Radia Proxy Server Preload

Use the following command to manually perform a Radia Proxy Server preload distribution. This is often used in a test environment, but can also be used in a production environment, if desired.

To manually perform the Preload from a web browser

You can manually perform a Radia Proxy Server's preload by typing the following command into a web browser:

```
http://<proxy_machine_name>:3466/proc/rps/sync
```

You will not get any feedback from the process—just a blank screen.

Check the Radia Proxy Server machine for activity and check the following Radia Proxy Server directory for data files.

```
<Radia_Integraton_Server>\DATA\RPS\STATIC\000D\
```

where: *<Radia_Integration_Server>* represents the base installation directory where the Radia Proxy Server is installed. By default, this is: C:\Novadigm\IntegrationServer for Windows, and /opt/Novadigm/IntegrationServer for UNIX.

Troubleshooting Questions

To expedite a solution, you should be prepared to answer the following questions before you contact technical support. This will expedite the solution.

- What has changed in your environment?
- What are the build numbers of your RPS.TKD, NVDKIT, and HTTPD.TKD modules? See *Collecting Information for HP Technical Support* below.
- What error messages were received?
- What actions was the Radia Proxy Server performing when the problem occurred? For example, was the RPS serving files or preloading data?
- If you use HTTP, can you download existing data from the Radia Proxy Server.

  To test this, open an Internet Explorer browser session and type:

  http://host:port/RESOURCE/SOFTWARE/FILE/an existing resourceID in static or dynamic cache)

Collecting Information for HP Technical Support

If you need to contact HP Technical Support for assistance, be sure to collect the following information:

1. **Version information** for nvdkit, httpd.tkd and rps.tkd. For details, see *How to Obtain Version Information*, below.
2. The **logs** directory including all subdirectories, stored by default in the following locations:

   Windows: ***SystemDrive*:\Novadigm\IntegrationServer\logs**
   UNIX: **/opt/Novadigm/IntegrationServer/logs**
3. The **etc** directory files (no subdirectories), stored by default in the following locations:

   Windows: ***SystemDrive*:\Novadigm\IntegrationServer\etc**
   UNIX: **/opt/Novadigm/IntegrationServer/etc**

4. For preload problems, also collect the Radia Proxy Server lib directory and contents (except any **000D** compressed data subdirectory), stored by default in the following locations:

   Windows: ***SystemDrive*:\Novadigm\IntegrationServer\etc\rps**
   UNIX: **/opt/Novadigm/IntegrationServer/etc/rps**

How to Obtain Version Information

To gather the version information for RPS.TKD, NVDKIT, and HTTPD.TKD

1. From a command prompt, navigate to where the Radia Proxy Server was installed; this is the location for nvdkit.exe.

   Your license file (license.nvd) needs to be copied to this folder.

2. Type the `NVDKIT commands for each of the components, as given in the following table.`

Table 4.2 ~ Radia Proxy Server Components and Commands to Obtain Version and Build

Component	Location	Command to Obtain Version Number
RPS.TKD	<RPS-Install>/modules	NVDKIT VERSION MODULES RPS.TKD
NVDKIT	<RPS-Install>	NVDKIT VERSION
HTTPD.TKD	<RPS-Install>	NVDKIT VERSION HTTPD.TKD

The following figure shows an example of obtaining the version information.

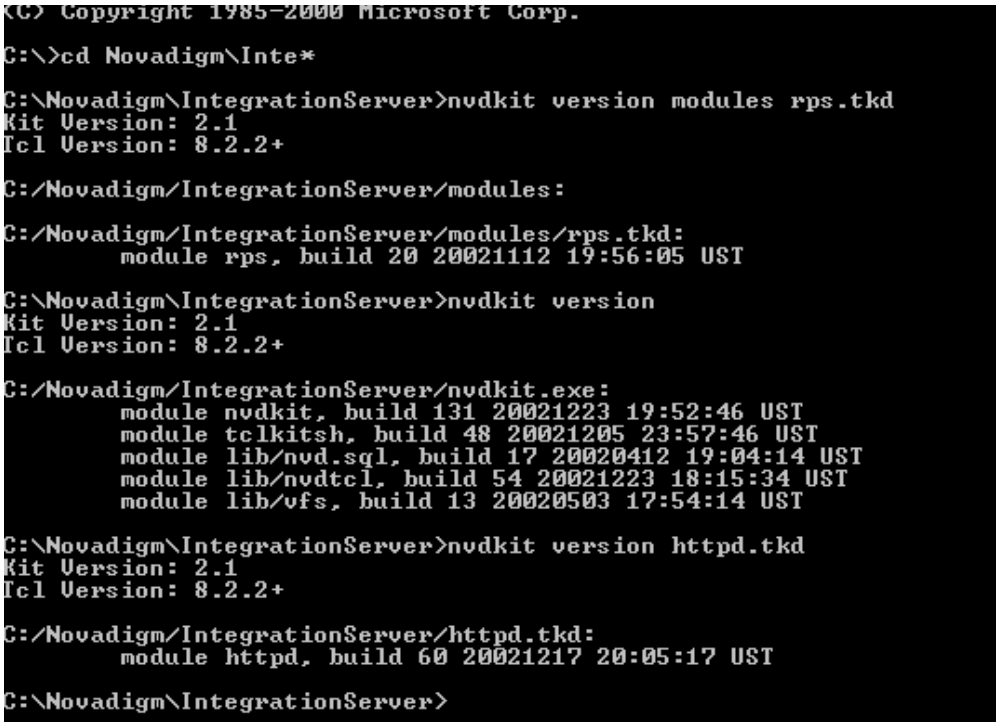


Figure 4.2 ~ Obtaining version and build numbers for RPS.TKD, NVDKIT, and HTTPD.TKD.

3. Collect this output for HP Technical Support.

Reporting a Problem for a Radia Proxy Server

Once you have exhausted the options laid out in this topic and reviewed your logs for typical errors, go to the HP OpenView Technical Support web site (see page 4) to report or resolve the problem.

Whenever you contact HP Technical Support for assistance regarding a Radia Proxy Server, be prepared to collect the items discussed in the topic *Collecting Information for HP Technical Support* on page 116. At a minimum, collect your HTTPD:3466.log plus an output showing the versions of your RPS, NVDKIT, and HTTPD.TKD modules available.

Summary

- Review the HTTP-port.log files and the CONNECT.log files to troubleshoot Radia Proxy Server errors.
- Review the common error messages and solutions given in this topic.
- Collect the version information, required log files, and other relevant directories and files before contacting HP Technical Support or submitting a support case.
- You can enter a command from any web browser to manually preload a Radia Proxy Server.

Glossary

Application Manager

See *Radia Application Manager*.

applications

Also called software, data, or services.

Applications are one type of content that Radia can manage on subscriber computers. Use the Radia Publisher to create packages of data to be managed on your subscribers' computers.

attribute

Also called *field, variable,* or *property*.

An **attribute** is a single, descriptive data item in a class. The class template contains a definition (e.g., the name, data type, description, and length) for each attribute that makes up the class. Class instances contain a set of attributes and each attribute contains a value.

attribute property

An **attribute property** controls some aspect of how an attribute is processed on the Radia Configuration Server and client computer. Each attribute defined in a class template has a set of Radia Configuration Server properties and a set of client properties.

client computer

A **client** computer is a subscriber's computer that has the Radia Client software installed on it.

dynamic cache

The **dynamic cache** is the Radia Proxy Server's secondary cache populated through Dynamic PassThru. When clients request data that is not available in the static cache, Dynamic PassThru sends a request to an upstream host, either a Radia Proxy Server co-located with a Radia Configuration Server, or another Radia Proxy Server, to satisfy the request. Multiple upstream hosts may be defined for fail-over support: if the first named host fails to connect, Dynamic PassThru sends the request to the next upstream host on the list.

Dynamic PassThru

The HTTP process used to populate the **dynamic cache.** When clients request data that is not available in the static cache or existing dynamic cache, Dynamic PassThru sends a request to an upstream host, either a Radia Proxy Server co-located with a Radia Configuration Server or another Radia Proxy Server, to satisfy the request. Multiple upstream hosts may be defined for fail-over support: if the first named host fails to connect, Dynamic PassThru sends the request to the next upstream host on the list.

instance

Also called *class instance*.

An **instance** is a Radia Database object containing a specific occurrence of a class. This is analogous to a row in a relational data table or a record in a traditional flat file. The attributes of an instance contain the data describing one specific entity of that class.

Management Portal

See *Radia Management Portal*.

method

A **method** is a program that performs functions that are meaningful in the context from which they are called.

Methods can be written in REXX or in a language that produces an executable that can validly run on the platform where it is invoked. The HP-supplied REXX run-time environment interprets REXX methods.

Client methods run on the subscriber's computer, while Radia Configuration Server methods run on the Radia Configuration Server computer.

object

An **object** is a data structure containing variables stored in a file with an .EDM suffix on the client computer. An object can consist of one or more instances. Each instance contains the same set of variables. The values held in the variables can vary from instance to instance.

Use the Radia Client Explorer to view, edit, or create objects.

preloading

Preloading is the process of loading the application to be distributed before any clients request the software. It requires a resolution of the Radia Proxy Server's distribution model on the Radia Configuration Server before the applications can be loaded, either directly from the Radia Configuration Server, or indirectly from another Radia Proxy Server. The preload resolution is performed using TCP. The files may be deployed using TCP or HTTP.

Radia Application Manager

The **Radia Application Manager** (radia_am.exe) is the Radia Client that manages mandatory services. The systems administrator uses the Radia System Explorer to specify the services that the Radia Application Manager manages on the subscriber's computer. No user interface is available.

Radia Client Explorer

The **Radia Client Explorer** can be used to view or edit local objects, or create new objects. You can also use the Radia Client Explorer to view objects located on a file server or on other computers to which you are connected via a local area network (LAN).

Radia Configuration Server

Also called *Active Component Server* or *Manager*.

The **Radia Configuration Server** distributes applications to client computers. It maintains the Radia Database, which stores information that the Radia Configuration Server needs to manage digital assets for distribution to client computers.

Radia Database

The **Radia Database** stores all of the information necessary to manage digital assets on a client computer, including:

- The software and data that Radia distributes.
- The "desired state" of each client computer with respect to the managed content.
- The policies determining which subscribers can subscribe to which packages.
- Security and access rules for administrators.

Use the Radia System Explorer to manipulate the Radia Database.

Radia Management Portal

The **Radia Management Portal** is a core Radia product, used to manage many different Radia assets.

Radia Manager

See *Radia Configuration Server*.

Radia Service

A set of digital assets managed as a Radia unit.

Radia Software Manager

The **Radia Software Manager** (radia_sm.exe) is the Radia Client used to manage optional services. The systems administrator uses the Radia System Explorer to specify the services that are available to the subscriber.

The subscriber installs and manages services that are available from the Radia Software Manager user interface (Service List).

Radia Staging Requestor

The **Radia Staging Requestor** resides on the client computer, and communicates with the Radia Staging Server to retrieve data from, and supply data to, the Radia Staging Server.

Radia System Explorer

The **Radia System Explorer** is used to manipulate the contents of the Radia Database.

resource

Also called *file*.

A **resource** is a single component that is bundled into a package. Examples of resources are files, desktop links, and sets of registry keys.

REXX

Radia **REXX** is an interpreted language that provides a simple way to customize various aspects of Radia processing.

service

Also called a software application, application, or software.

A **service** is a group of related packages.

Software Manager

See *Radia Software Manager*.

STAGER instance

The **STAGER** instance in the **POLICY** domain of the Radia Database contains information necessary for the client to connect to the Radia Proxy Server or the Radia Staging Server to obtain its needed resources. A STAGER instance can also specify the protocol and deployment source used to preload a Radia Proxy Server.

Staging Requestor

See **Radia Staging Requestor**.

static cache

The **static cache** is the Radia Proxy Server's primary cache, managed by the Preloader. After installing the Radia Proxy Server, this cache is preloaded with services as defined in the Radia Configuration Server distribution model. The services may be deployed from the Radia Configuration Server or another Radia Proxy Server.

subscriber

A **subscriber** is the person who uses managed applications on a client computer.

System Explorer

See *Radia System Explorer*.

variable

A **variable** is a piece of named storage that contains a changing value. The variable's value forms a part of the client's resolved distribution model and can influence the resolution process through messaging or symbolic substitution.

Lists

Figures

Tables

Procedures

Index

E

F

H

I

L

M

N

O

P

R

S

T

U

V

W

Z