Steel-Belted Radius

Reference Guide

Release 6.27
Published Date October 2019

Copyright Notice

FCC Statement

The following information is for FCC compliance of Class A devices: This equipment has been tested and found to comply with the limits for Class A digital device, pursuant to part 15 of the FCC rules. These limits are designed to provide reasonable protection against harmful interference when the equipment is operated in a commercial environment. The equipment

generates, uses, and can radiate radio- frequency energy and, if not installed and used in accordance with the instruction manual, may cause harmful interference to radio communications. Operation of this equipment in a residential area is likely to cause harmful interference, in which case users will be required to correct the interference at their own expense.

The following information is for FCC compliance of Class B devices: The equipment described in this manual generates and may radiate radio-frequency energy. If it is not installed in accordance with the instruction manual, it may cause interference with radio and television reception. This equipment has been tested and found to comply with the limits for a Class B digital device in accordance with the specifications in part 15 of the FCC rules. These specifications are designed to provide reasonable protection against such interference in a residential installation. However, there is no guarantee that interference will not occur in a particular installation.

If this equipment does cause harmful interference to radio or television reception, which can be determined by turning the equipment off and on, the user is encouraged to try to correct the interference by one or more of the following measures:

- Reorient or relocate the receiving antenna.
- Increase the separation between the equipment and receiver.
- Consult the dealer or an experienced radio/TV technician for help.
- Connect the equipment to an outlet on a circuit different from that to which the receiver is connected.

Caution: Changes or modifications to this product could void the user's warranty and authority to operate this device and may result in the equipment no longer complying with FCC requirements for Class A or Class B digital devices. In that event, your right to use the equipment may be limited by FCC regulations, and you may be required to correct any interference to radio or television communications at your own expense.

U.S. Government Rights

Commercial software and commercial software documentation: This documentation is commercial computer software documentation and the products (whether hardware or software) covered by this documentation are or contain commercial computer software. Government users are subject to the Pulse Secure, LLC. standard end user license agreement and any applicable provisions of the FAR and its supplements. No further rights are granted.

Products (whether hardware or software) covered by, and information contained in, this documentation are controlled by U.S. Export Control laws and may be subject to the export or import laws in other countries. Nuclear, missile, chemical, biological weapons end uses or end users, whether direct or indirect, are strictly prohibited. Export or re-export to countries subject to U.S. embargo or to entities identifie on US export exclusion lists, including, but not limited to, the denied persons and specially designated national lists, is strictly prohibited.

Disclaimer

THE SPECIFICATIONS AND INFORMATION REGARDING THE PRODUCTS IN THIS MANUAL ARE SUBJECT TO CHANGE WITHOUT NOTICE. ALL STATEMENTS, INFORMATION AND RECOMMENDATIONS IN THIS MANUAL ARE BELIEVED TO BE ACCURATE BUT ARE PRESENTED WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED.

THE SOFTWARE LICENSE AND LIMITED WARRANTY FOR THE ACCOMPANYING PRODUCT ARE SET FORTH IN THE INFORMATION PACKET THAT SHIPPED WITH THE PRODUCT AND ARE INCORPORATED HEREIN BY REFERENCE. IF YOU ARE UNABLE TO LOCATE THE SOFTWARE LICENSE OR LIMITED WARRANTY, CONTACT YOUR PULSE SECURE REPRESENTATIVE FOR A COPY.

apache/httpclient, that is used in SBR-E software is of license type "Apache License 2.0" and refer the following URL for more information: https://github.com/apache/httpcomponents-client/blob/4.5.x/LICENSE.txt.

bcgit/bc-java, that is used in SBR-E software is of license type "MIT" and refer the following URL for more information: https://github.com/bcgit/bc-java/blob/r1rv60/LICENSE.html.

google/gwt, that is used in SBR-E software is of license type "Apache License 2.0" and refer the following URL for more information: http://www.gwtproject.org/terms.html.

gwtbootstrap3/gwtbootstrap3, that is used in SBR-E software is of license type "Apache License 2.0" and refer the following URL for more information: https://github.com/gwtbootstrap3/gwtbootstrap3/blob/0.9.3/LICENSE.

kohsuke/WinSW, that is used in SBR-E software is of license type "MIT" and refer the following URL for more information: https://github.com/kohsuke/winsw/blob/winsw-v2.2.0/LICENSE.txt.

laaglu/lib-gwt-file, that is used in SBR-E software is of license type "GNU Lesser General Public License v3" and refer the following URL for more information:
http://www.gnu.org/license/lgpl.html.

Revision History

The following table lists the revision history for this document.

Revision	Date	Description
6.27	October 2019	No documentation updates for 6.27 Release
6.26	May 2019	6.26-R1 Updates

Content

About This Guide

The Steel-Belted Radius Reference Guide describes the configuration options for the Steel-Belted Radius software.

Before You Begin

This manual assumes that you have installed the Steel-Belted Radius software and the SBR Administrator. For more information, refer to the Steel-Belted Radius Installation and Upgrade Guide.

Audience

This manual is intended for network administrators responsible for implementing and maintaining authentication, authorization, and accounting services for an enterprise. This manual assumes that you are familiar with general RADIUS and networking concepts and the specific environment in which you are installing Steel-BeltedRadius.

If you use Steel-Belted Radius with third-party products such as Oracle or RSA SecurID, you should be familiar with their installation, configuration, and use.

What's in this Manual

This manual contains the following chapters and appendixes:

- Chapter 1, "Introduction," presents a summary of the files used by the various editions of Steel-Belted Radius and provides some general suggestions about modifying configuration files.
- Chapter 2, "Authentication Configuration Files," describes the files used to specify Steel-Belted Radius authentication configuration settings.
- Chapter 3, "Operations Files," describes the files used to specify Steel-Belted Radius operation and administrationsettings.
- Chapter 4, "Attribute Processing Files," describes the configuration and dictionary files that specify RADIUS attributes for third-party network devices.
- Chapter 5, "Address Assignment Files," describes the files used to configure address assignment functions in the GEE version of Steel-Belted Radius.
- Chapter 6, "Accounting Configuration Files," describes the files used to enable and configure Steel- Belted Radius accounting settings.
- Chapter 7, "Realm Configuration Files," describes the configuration files relating to proxy and directed realm administration in the GEE version of Steel-Belted Radius.
- Chapter 8, "Database Error Map Files," describes the database error files that specify how Steel-Belted Radius should classify errors returned by backend databases.
- Chapter 9, "EAP Configuration Files," describes the EAP configuration and helper files, which specify options for automatic EAP helper methods.
- Chapter 10, "SNMP Configuration Files," describes the SNMP configuration files used in the GEE versions of Steel-Belted Radius (Linux only).
- Chapter 11, "SQL Authentication Files," describes the files used to configure SQL authentication

in Steel-BeltedRadius.

- Chapter 12, "SQL Accounting Files," describes the files used configure SQL accounting in Steel-Belted Radius.
- Chapter 13, "LDAP Authentication Files," describes the files used to configure LDAP authentication in Steel-BeltedRadius.
- Chapter 14, "TACACS+ Configuration Files,'describes the files used to configure TACACS+ in Steel-Belted Radius.
- Appendix A, "Authentication Protocols," provides a matrix of authentication methods and their supported authentication protocols.
- Appendix B, "Vendor-Specific Attributes," describes the Steel-Belted Radius vendor-specific attributes.
- Appendix C, "SNMP Traps and Statistics," describes the proprietary SNMP traps and rate statistics generated by Steel-Belted Radius.
- Appendix D, "Windows Events," describes the Windows events that can be generated by Steel-Belted Radius.

TypographicalConventions

Table 1 describes the text conventions used throughout this manual.

Table 1: Typographical Conventions

Convention	Description	Examples
Bold typeface	Indicates buttons, field names, dialog names, and other user interface elements.	Use the **Scheduling** and **Appointment** tabs to schedule a meeting.
Plain sans serif typeface	Represents: • Code, commands, and keywords • URLs, file names, and directories	Examples: • Code: certAttr.OU = 'Retail Products Group' • URL: Download the JRE application from: http://java.sun.com/j2se/
Italics	Identifies: • Terms defined in text • Variable elements • Book names	Examples: • Defined term: An RDP client is a Windows component that enables a connection between a Windows server and a user's machine. • Variable element: Use settings in the Users > Roles > Select Role > Terminal Services page to create a terminal emulation session. • Book name: See the Steel-Belted Radius Administration Guide.

Editions/UsedIn

Steel-Belted Radius is available in multiple editions to meet the requirements of different types of customers. This manual uses the following abbreviations to identify editions of Steel-Belted Radius:

- GEE – Global Enterprise Edition

- EE – Enterprise Edition

The description of each configuration file used in Steel-Belted Radius identifies the editions that use that file. If an edition uses only some of the settings in a file, the edition identifier includes an asterisk. For example, the following label indicates that the GEE edition use all settings in the file and the EE edition does not use the file.

Used by:

GEE **Not**

used by:

EE ***Syntax***

- radiusdir represents the directory into which Steel-Belted Radius has been installed. By default, this is C:\Program Files (x86)\Pulse Secure\Steel-Belted Radius for Windows systems and /opt/PSsbr/radius on Linux.
- Brackets [ ] enclose optional items in format and syntax descriptions. In the following example, the first Attribute argument is required; you can include an optional second Attribute argument by entering a comma and the second argument (but not the square brackets) on the same line.

  <add | replace> = Attribute [,Attribute]

  In configuration files, brackets identify section

  headers: the [Processing] section of proxy.ini

  In screen prompts, brackets indicate the default value. For example, if you press Enter without entering anything at the following prompt, the system uses the indicated default value (/opt).

  Enter install path [/opt]:

- Angle brackets < > enclose a list from which you must choose an item in format and syntax descriptions.
- A vertical bar (|) separates items in a list of choices. In the following example, you must specify add or replace (but not both):

  <add | replace> = Attribute [,Attribute]

Related Documentation

The following documents supplement the information in this manual.

Steel-Belted Radius Documentation

The readme.txt file contains the latest information about features, changes, known problems, and resolved problems. If the information differs from the information found in the documentation set, defer to the information in the Release Notes.
In addition to this manual, the Steel-Belted Radius documentation includes the following manuals:

- The Steel-Belted Radius Reference Guide describes the configuration files and settings used by Steel-Belted Radius.
- The Steel-Belted Radius Getting Started Guide describes how to install, configure, and administer the Steel-Belted Radius software on a server running the Linux, or Windows operating system.

Requests for Comments (RFCs)

The Internet Engineering Task Force (IETF) maintains an online repository of Request for Comments (RFC)s online at http://www.ietf.org/rfc.html. Table 2 lists the RFCs that apply to this guide.

Table 2: RFCs

RFC Number	Title
RFC 1155	Structure and Identification of Management Information for TCP/IP-based Internets. M. Rose, K. McCloghrie, May 1990.
RFC 1213	Management Information Base for Network Management of TCP/IP-Based Internets: MIB-II. K. McCloghrie, M. Rose, March 1991.
RFC 2271	An Architecture for Describing SNMP Management Frameworks. D. Harrington, R. Presuhn, B. Wijnen, January 1998.
RFC 2284	PPP Extensible Authentication Protocol (EAP). L. Blunk, J. Volbrecht, March 1998.
RFC 2433	Microsoft PPP CHAP Extensions. G. Zorn, S. Cobb, October 1998.
RFC 2548	Microsoft Vendor-specific RADIUS Attributes. G. Zorn. March 1999.
RFC 2607	Proxy Chaining and Policy Implementation in Roaming. B. Aboba, J. Vollbrecht, June 1999.
RFC 2618	RADIUS Authentication Client MIB. B. Aboba, G. Zorn. June 1999.
RFC 2619	RADIUS Authentication Server MIB. G. Zorn, B. Aboba. June 1999.
RFC 2620	RADIUS Accounting Client MIB. B. Aboba, G. Zorn. June 1999.
RFC 2621	RADIUS Accounting Server MIB. G. Zorn, B. Aboba. June 1999.
RFC 2622	PPP EAP TLS Authentication Protocol. B. Aboba, D. Simon, October 1999.
RFC 2809	Implementation of L2TP Compulsory Tunneling via RADIUS. B. Aboba, G. Zorn. April 2000.
RFC 2865	Remote Authentication Dial In User Service (RADIUS). C. Rigney, S. Willens, A. Rubens, W. Simpson. June 2000.
RFC 2866	RADIUS Accounting. C. Rigney. June 2000.
RFC 2867	RADIUS Accounting Modifications for Tunnel Protocol Support. G. Zorn, B. Aboba, D. Mitton. June 2000.

RFC Number	Title
RFC 2868	RADIUS Attributes for Tunnel Protocol Support. G. Zorn, D. Leifer, A. Rubens, J. Shriver, M. Holdrege, I. Goyret. June 2000.
RFC 2869	RADIUS Extensions. C. Rigney, W. Willats, P. Calhoun. June 2000.
RFC 2882	Network Access Servers Requirements: Extended RADIUS Practices. D. Mitton. July 2000.
RFC 3162	RADIUS and IPv6. B. Aboba, G. Zorn, D. Mitton. August 2001.
RFC 3575	IANA Considerations for RADIUS (Remote Authentication Dial In User Service). B. Aboba, July 2003.
RFC 3579	RADIUS (Remote Authentication Dial In User Service) Support For Extensible Authentication Protocol (EAP). B. Aboba, P. Calhoun, September 2003.
RFC 3580	IEEE 802.1X Remote Authentication Dial In User Service (RADIUS) Usage Guidelines. P. Congdon, B. Aboba, A. Smith, G. Zorn, J. Roese, September 2003.

Third-Party Products

For more information about configuring your access servers and firewalls, consult the manufacturer's documentation provided with each device.

Contacting Technical Support

For technical support, open a support case using the Case Manager link at **https://www.pulsesecure.net/support/**

When you are running Legacy SBR Administrator, you can choose **Web > Steel-Belted Radius User Page** to access a special home page for Steel-Belted Radius users.

When you are running Web GUI, you can choose **Help** > **Home Page** > **Steel-Belted Radius Home Page** to access a special home page for Steel-Belted Radius users.

When you call technical support, please have the following information at hand:

- Your Steel-Belted Radius product edition and release number (for example, Global Enterprise Edition version 6.2).
- Information about the server configuration and operating system, including any OS patches that have been applied.
- For licensed products under a current maintenance agreement, your license or support contract number.
- Question or description of the problem, with as much detail as possible.
- Any documentation that can help resolve the problem, such as error messages, memory dumps, compiler listings, and error logs.

Chapter 1

Introduction

Steel-Belted Radius is a complete implementation of the RADIUS (Remote Authentication Dial In User Service) protocol that runs in your Windows, or Linux environment. It interfaces with a wide variety of network access devices, and authenticates remote and WLAN users against numerous back-end databases — allowing you to consolidate the administration of all your remote and WLAN users, however they connect to your network.

Configuration Files

The configuration and behavior of your Steel-Belted Radius server is determined by a set of configuration files. In many cases, you must edit these files manually. In a few cases, the contents of a configuration file are updated dynamically when you use the SBR Administrator application to change settings.

Configuration files reside in the radiusdir\Service (Windows) or radiusdir (Linux) directory. The number and names of configuration files depend on which edition of Steel-Belted Radius you are running and whether you are using optional add-on components.

Table 3 identifies the files that are used by each edition of Steel-Belted Radius. An asterisk in an entry means that the applicable edition uses some but not all of the settings in that file.

Table 3: Steel-Belted Radius Configuration Files

File	Function	GEE	EE
*.acc	Configures an SQL accounting method.	x	x
*.dcm	Master list of dictionary files.	x	x
*.dct	Vendor-specific dictionary file.	x	x
*.dhc	Configures specific DHCP address pools, where * is the name of an address pool listed in dhcp.ini.	x	
*.dir	Configures directed authentication and directed accounting realms.	x	
*.rr	Configures attribute value pools.	x	
*.pro	Configures proxy realms	x	
access.ini	Maps user or group account levels to administrative permissions. Used with admin.ini to grant administrators access privileges to administrative objects and actions.	x	
account.ini	Controls how RADIUS accounting attributes are logged.	x	x*
admin.ini	Maps administrative access levels to sets of access rights. Used with access.ini to grant administrators access privileges to administrative objects and actions.	x	

File	Function	GEE	EE
authlog.ini	Controls how RADIUS authentication requests are logged by Steel-Belted Radius.	x	
authReport.ini	Controls what authentication logs Steel- Belted Radius generates.	x	x
authReportAccept.ini	Controls options for the acceptance authentication log file.	x	x
authReportBadShared Secret.ini	Controls options for the invalid shared secret authentication log file.	x	x
authReportReject.ini	Controls options for the rejection authentication log file.	x	x
authReportUnknownC lient.ini	Controls options for the unknown client authentication log file.	x	x
blacklist.ini	Configures blacklist settings, which are used to block authentication requests that match a blacklist profile.	x	
bounce.ini	Configures the auto-restart function for the Windows version of Steel-Belted Radius, which causes Steel- Belted Radius to restart itself automatically whenever it experiences a shutdown.	x	
ccagw.ini	Configures support for 3Com CCA tunnel attributes.	x	x
classmap.ini	Specifies what Steel-Belted Radius does with RADIUS attributes encoded in one or more Class attributes included in accounting requests.	x	x
dhcp.ini	Configures DHCP address pools so that IP addresses can be assigned from a backend DHCP server.	x	
eap.ini	Configures EAP authentication methods used by Steel-Belted Radius	x	x
events.ini	Controls dilutions and thresholds for Steel-Belted Radius events used to signal failures and warnings.	x	x
filter.ini	Sets up rules for filtering attributes into and out of RADIUS packets.	x	
ldapauth.aut	Specifies settings for LDAP authentication in Steel-Belted Radius.	x	x*
lockout.ini	Configures settings that lock user accounts after repeated failed login attempts.	x	
peapauth.aut	Configures the EAP-PEAP authentication method.	x	x*
proxy.ini	Stores information that applies to all realms on the server.	x	
proxyrl.ini	Configures list of realms for forwarding	x	

File	Function	GEE	EE
	accounting packets.		
radConfigServer.ini	Configures settings related to scheduling the DB backup	x	x
radius.ini	Configures a variety of operational settings for Steel-Belted Radius.	x	x*
radsql.acc (Linux only)	Configures Oracle SQL accounting for the Linux version of Steel-Belted Radius.	x	x*
radsql.aut (Linux only)	Configures Oracle SQL authentication for the Linux version of Steel-Belted Radius.	x	x*
radsqljdbc.acc (Linux only)	Configures JDBC SQL accounting for the Linux version of Steel-Belted Radius.	x	x*
redirect.ini	Configures settings that redirect users after repeated failed login attempts.	x	x*
securid.ini	Specifies the prompt strings returned to RSA SecurID users during login and authentication.	x	
securidauth.aut	Configures the SecurID authentication method.	x	x
servtype.ini	Configures service type mappings, which allow a user to have multiple authorization attribute sets based on the service type the user is requesting.	x	
sidalt.aut	Configures token caching for RSA SecurID authentication.	x	x
spi.ini	Defines encryption keys and identifies the servers from which Steel-Belted Radius processes encrypted Class attributes in accounting requests	x	x
sqlacct.acc (Windows only)	Configures SQL accounting for the Windows version of Steel-Belted Radius.	x	x*
sqlauth.aut (Windows only)	Specifies the name of the TACACS+ server and the shared secret used to validate communication between the Steel-Belted Radius server and the TACACS+ server.	x	x
tlsauth.aut	Configures the TLS authentication method.	x	x
tlsauth.eap	Configures the operation of the TLS helper method.	x	x
ttlsauth.aut	Configures the TTLS authentication method.	x	x
uniport.aut	Configures the UniPort authentication	x	

File	Function	GEE	EE
	method.		
update.ini	Controls what information is updated when Steel-Belted Radius receives a HUP or USR2 signal.	x*	
vendor.ini	Maps vendor-specific dictionary files to identifiers used in the Steel-Belted Radius administrative database.	x	x*
winauth.aut	Configures the WinAuth authentication method.	x	x

Tips for Editing Configuration Files

When editing configuration files, observe the following guidelines:

- Configuration files are text files that you can edit using a standard text editor, such as Notepad on Windows and gedit on Linux. If you use a word processing application such as Microsoft Word to edit your configuration files, make sure that you save the modified file in ASCII text format.
- You should make a backup copy of your configuration files before you make any changes, so that you have a working archive copy in the event that you delete or misconfigure an important setting and want to revert to your previous configuration.
- You can enter comments in configuration files by starting the line containing the comment with a semicolon (;) as the first character of the line. To disable a setting, consider commenting it out (by putting a semicolon at the start of the line) instead of deleting it.
- Put comments on a separate line above or below configuration settings. You cannot include comments on the same line as a configuration setting.

Correct:
;Set to 0 on 5/30/2006
Session_Timeout = 0
Incorrect:
Session_Timeout = 0 ; Set to 0 on 5/30/2006

- The default configuration files provided with Steel-Belted Radius typically include section headers and settings that are commented out. In such cases, Steel-Belted Radius uses the value shown in the commented setting as the default, meaning that you do not need to change the setting if you want to use the default value.
- To change the value for a setting to something other than the default value, you must uncomment the setting by removing the semicolon at the start of the line. Note that the section headers (in square brackets) must also be uncommented for settings to be processed correctly.
- Make sure that lines containing settings or section headers have a text character in the first column. If a line has white space in the first column, it might not be processed correctly.
- You can edit configuration files while Steel-Belted Radius is running. However, changes to some files, such as radius.ini, require that you restart Steel-Belted Radius for the changes to take effect.

Chapter 2

Authentication Configuration Files

This chapter describes the usage and settings for the initialization files used by Steel-Belted Radius to authenticate users and to record the results of authentication events. Initialization files are loaded at startup time, and reside in the Steel-Belted Radius directory.

authlog.ini File

Used by:

GEE Not

used by:

EE

The authlog.ini initialization file contains information that controls how RADIUS authentication request attributes are logged in the comma-delimited yyyymmdd.authlogfile.

[Alias/name] Sections

You can create one or more [Alias/name] sections in authlog.ini (Table 4) to associate attributes of different names, but identical meaning. For example, one network access device vendor might call an attribute Auth- Connect-Type and another might call it Auth-Conn-Typ, yet the two attributes would both map to Auth-Conn- Type.

Each [Alias/name] section permits you to map one RADIUS authentication request attribute that is already being logged by Steel-Belted Radius to any number of other attributes. You can provide as many [Alias/name] sections as you want, using the following syntax for each section:

```
[Alias/name]
VendorSpecificAttri
bute=
VendorSpecificAttri
bute= M
```

Table 4: authlog.ini [Alias/name] Syntax

Parameter	Function
name	The preferred attribute name. The name attribute must be one that you are currently logging to a column in the Steel-Belted Radius authentication request log file (.authlog). Therefore, it must be listed in the [Attributes] section of authlog.ini.
VendorSpecificAttribute	Each entry is given on one line. An equal sign (=) must immediately follow each VSA name, without any intervening space. Improperly formatted entries are considered invalid and are ignored.

Each VendorSpecificAttribute in the list is logged to the name column in the authentication request log file. Because you are listing these attributes in an [Alias/name] section, make sure they are not listed in the [Attributes] section or they will be logged to their own columns as well as to the name column.

All of the attribute names that you reference in an [Alias/name] section must be defined in a dictionary file that is already installed on the Steel-Belted Radius server. This includes name and each VendorSpecificAttribute entry.

In the following example, the standard RADIUS attribute Auth-Conn-Type is mapped to the vendor-specific attributes Auth-Connect-Type and Auth-Conn-Typ. Values encountered for all three attributes are logged in the Auth-Octet-Packets column in the authentication request log file:

```
[Alias/Auth-Conn-
Type] Auth-Conn-
Typ=
Auth-Connect-Type=
```

[Attributes] Section

The [Attributes] section of authlog.ini lists all the attributes logged in the authentication request log file. When you install Steel-Belted Radius, the authlog.ini file is set up so that all standard RADIUS attributes and all supported vendor authentication attributes are listed.

You can configure what is logged to the authentication request log file by rearranging the order of attributes in the [Attributes] section. You can delete or comment out attributes you do not want or that do not apply to
your equipment. This lets you design the content and column order of any spreadsheets that you plan to create based upon the authentication request log file.

The syntax of the [Attributes] section is as follows:

```
[Attributes]
AttributeName=
AttributeName=
.
.
.
```

For example:

```
[Attrib
utes]
User-
Name
=
NAS-IP-
Address=
NAS-Port=
Service-
Type=
Framed-
Protocol=
Framed-IP-
Address=
Framed-IP-
Netmask=
Framed-
Compression=
```

The [Attributes] section lists one AttributeName on each line. You must ensure that an equal sign (=) immediately follows each AttributeName, with no spaces in between. Improperly formatted entries are considered invalid and are ignored.

Each AttributeName in the [Attributes] section must be defined in a standard RADIUS or vendor-specific dictionary file (.dct) installed on the Steel-Belted Radius server.

Note: The first five attributes in each authentication log file entry (Date, Time, RAS-Client, Full-Name, and ACC/REJ) are always enabled, and cannot be re-ordered or deleted. Therefore, these attributes do not appear in the authlog.ini file [Attributes] section.

[Configuration] Section

The [Configuration] section of authlog.ini specifies the location

of the yyymmdd.authlog file.

Table 5: authlog.ini [Alias/name] Syntax

Parameter	Function
LogDir	Specifies the destination directory on the local host where yyyymmdd.authlog files are stored. Default value is the directory where Steel-Belted Radius is installed. Note: You cannot write authlog files to a mapped or shared drive.

[Settings] Section

Steel-Belted Radius writes all authentication request data to the current authentication request log file (yyyymmdd.authlog) until that log file is closed. When Steel-Belted Radius closes an authentication request log file, it immediately opens a new one and begins writing authentication request data to it.

You can configure how often this rollover of the authentication request log file occurs.

The naming conventions of the authentication request log files support the fact that Steel-Belted Radius can create more than one file per day. The formats are as follows. In the examples below, y=year digit, m=month digit, d=day digit, and h=hour digit. The extra sequence number _nnnnn starts at _00000 each day.

Table 6: Authentication Log Rollover

File Generation Method	File Naming Convention
Default (24 hours)	yyyymmdd.authlog
Non-24-hour rollover	yyyymmdd_hhmm.authlog
Rollover due to size	yyyymmdd_nnnnn.authlog
Rollover due to size or startup when non-24-hour time in effect	yyyymmdd_hhmm_nnnnn.authlog

The [Settings] section of authlog.ini (Table 7) controls which entries are written to the authentication request log file, and ensure the compatibility of these entries with a variety of database systems. The following "rollover" settings can be present in the [Settings] section.

Table 7: authlog.ini [Settings] Syntax

Parameter	Function
Enable	• If set to 0, the authentication request log is disabled and other settings are ignored. • If set to 1, the authentication request log is enabled. Set Enable to 1 for Authentication servers. For efficiency, set Enable to 0 for non-authentication servers. Default value is 0.
LogFilePermissions (Linux only)	Specifies the owner and access permission setting for the auth log (yyyymmdd.authlog) file. Enter a value for the LogFilePermissions setting in owner:group permissions format, where: • **owner** specifies the owner of the file in text or numeric format. • **group** specifies the group setting for the file in text or numeric format. • **permissions** specifies what privileges can be exercised by Owner/Group/Other with respect to the file in text or numeric format. For example, ralphw:1007 rw-r- - - - - specifies that the file owner (ralphw) can read and edit the log file, members of group 1007 can read (but not edit) the log file, and other users cannot access the log file.
BufferSize	Specifies the size of the buffer used in the authentication request logging process, in bytes. Default value is 32768.
LineSize	Specifies the maximum number of characters in a line in the authentication request log. You can enter a number in the range 1024–32768. Default value is 4096.
MaxSize	• If set to a number greater than 0, specifies the maximum number of bytes for an authentication request log file. If the authentication request log file equals or exceeds this limit when the log size is checked, the log file is closed and a new file started. • If set to 0, the authentication request log has no maximum size. Default value is 0. **Note:** Because the size of the log file is checked once per minute, the log file can exceed the maximum size specified in this parameter.
QuoteBinary	• If set to 1, binary values written to the authentication request log file are enclosed in quotes. • If set to 0, quotes are not used. Set this value according to the format expected by the application that processes the authentication request log entries. Default value is 1.
QuoteInteger	• If set to 1, integer values written to the authentication request log file are enclosed in quotes. • If set to 0, quotes are not used. Set this value according to the format expected by the application that processes the authentication request log entries. Default value is 1.
QuoteIPAddress	• If set to 1, IP addresses written to the authentication request log file are enclosed in quotes. • If set to 0, quotes are not used. Set this value according to the format expected by the application that processes the authentication request log entries. Default value is 1.
QuoteText	• If set to 1, text strings written to the authentication request log file are enclosed in quotes.

Parameter	Function
	• If set to 0, quotes are not used. Set this value according to the format expected by the application that processes the authentication request log entries. Default value is 1.
RollOver	Specifies how often the current authentication request log file is closed and a new file opened (a rollover), up to one rollover per minute. • If set to 0, the authentication request log rolls over once every 24 hours, at midnight local time. • If set to a number in the range 1–1440, specifies the number of minutes until the next rollover. Default value is 0.
RollOverOnStartup	• If set to 1, each time Steel-Belted Radius is started, it closes the current authentication request log file and opens a new one. A sequence number _nnnnn is appended to the log file name, just as when MaxSize is reached. • If set to 0, each time Steel-Belted Radius is started, it appends entries to the previously open authentication request log file. Default value is 0.
Titles	• If set to 1, each time a new authentication request log file is created, the title line (containing column headings) is written to the file. • If set to 0, the line is not written. Default value is 1.
UTC	• If set to 1, time and date values are provided according to universal time coordinates (UTC, formerly known as Greenwich mean time or GMT). • If set to 0, time and date values reflect local time. Default value is 0.

authReport.ini File

Used by:

GEE, EE

Not Used

By: —

The authReport.ini initialization file controls whether Steel-Belted Radius generates the following reports:

- Authentication acceptance report
- Authentication rejection report
- Unknown authentication client report
- Invalid shared secret report

If enabled, these reports are written to the radiusdir\authReports directory on the Steel-Belted Radius server.

Note: The settings in the authReport.ini file are overwritten when the SBR Administrator is used to enable or disable these reports.

[AcceptReport] Section

The [AcceptReport] section of authReport.ini (Table 8) enables or disables generation of the authentication acceptance report. The settings for the authentication acceptance report are specified in the authReportAccept.ini File.

Sample syntax is as follows:

```
[AcceptRe
port]
Enable = 1
```

Table 8:authRport.ini [AcceptReport] Syntax

Parameter	Function
Enable	• If set to 1, Steel-Belted Radius periodically generates the authentication acceptance report. • If set to 0, the authentication acceptance report is not generated. Default value is 1.

[BadSharedSecretReport] Section

The [BadSharedSecretReport] section of authReport.ini (Table 9) enables or disables generation of the invalid shared secret report. The settings for the invalid shared secret report are specified in the authReportBadSharedSecret.ini File.

Sample syntax is as follows:

```
[BadSharedSecretR
eport] Enable = 1
```

Table 9: authReport.ini [BadSharedSecretReport] Syntax

Parameter	Function
Enable	• If set to 1, Steel-Belted Radius periodically generates the invalid shared secret report. • If set to 0, the invalid shared secret report is not generated. Default value is 1.

[RejectReport] Section

The [RejectReport] section of authReport.ini (Table 10) enables or disables generation of the authentication rejection report. The settings for the authentication rejection report are specified in the authReportReject.ini file.

Sample syntax is as follows:

```
[RejectRep
ort]
Enable = 1
```

Table 10: authReport.ini [RejectReport] Syntax

Parameter	Function

Enable	• If set to 1, Steel-Belted Radius periodically generates the authentication rejection report. • If set to 0, the authentication rejection report is not generated. Default value is 1.

[UnknownClientReport] Section

The [UnknownClientReport] section of authReport.ini (Table 11) enables or disables generation of the unknown authentication client report. The settings for the unknown authentication client report are specified in the authReportUnknownClient.ini file.

Sample syntax is as follows:

```
[UnknownClientRe
port] Enable = 1
```

Table 11: authReport.ini [UnknownClientReport] Syntax

Parameter	Function
Enable	• If set to 1, Steel-Belted Radius periodically generates the unknownauthentication client report. • If set to 0, the unknown authentication client report is not generated. Default value is 1.

authReportAccept.ini File

Used by:

GEE, EE

Not Used

By: —

The authReportAccept.ini initialization file specifies options for the authentication acceptance report, which is an ASCII comma-delimited file that contains information about successful authentications by the Steel-Belted Radiusserver.

[Attributes] Section

The [Attributes] section of authReportAccept.ini lists the attributes logged in the acceptance log.

You can configure what is logged to the acceptance report by entering attributes in the [Attributes] section in the sequence in which they should appear. This lets you design the content and column order of any spreadsheets that you plan to create based upon the acceptance report.

The syntax of the [Attributes] section is as follows:

```
[Attributes]
AttributeName=
AttributeName=
.
.
.
```

For example:

```
[Attrib
utes]
```

User-
Name
=
NAS-IP-Address=

The [Attributes] section lists one AttributeName on each line. You must ensure that an equal sign (=) immediately follows each AttributeName, with no spaces in between. Improperly formatted entries are considered invalid and are ignored.

Each AttributeName in the [Attributes] section must be defined in a standard RADIUS or vendor-specific dictionary file (.dct) installed on the Steel-Belted Radius server.

Note: The first four attributes in each acceptance report entry (Date, Time, RAS-Client, and Full- Name) are always enabled, and cannot be re-ordered or deleted. Therefore, these attributes do not appear in the [Attributes] section of the authReportAccept.ini file.

[Settings] Section

The [Settings] section of authReportAccept.ini (Table 12) specifies the operational characteristics of the authentication acceptance report.

If the MaxMinutesPerFile parameter is set to 0, the file name of the authentication acceptance report is accepts_ yyyymmdd.csv (where yyyymmdd identifies the date the report was generated.) If the MaxMinutesPerFile parameter is set to a value greater than 0, the file name of the report is accepts_yyyymmdd_hhmm.csv (where yyyymmdd identifies the date and hhmm identifies the time the report was generated.)

Sample syntax is as follows:

[Settings]
BufferSize =
131072
DaysToKeep = 1
LineSize = 4096
LogfilePermissions = ralphw:1007 rw-r- - - - -
MaxMinutesPerFile = 0
QuoteInteger = 1
QuoteIpAddress = 1
QuoteBinary = 1
QuoteText = 1
QuoteTime = 1
UTC = 0

Table 12: authReportAccept.ini [Settings] Syntax

Parameter	Function
BufferSize	The size of the buffer used in the logging process, in bytes. Default value is 131072.
DaysToKeep	Specifies the number of days the Steel-Belted Radius server should retain each authentication acceptance report. Default value is 1 (one day).

Parameter	Function
LineSize	The maximum size of a single log line. The allowable range is 1024 to 32768. Default value is 4096.
LogFilePermissions (Linux only)	Specifies the owner and access permission setting for the authentication acceptance log (accepts_yyyymmdd.csv) file. Enter a value for the LogFilePermissions setting in owner:group permissions format, where: • owner specifies the owner of the file in text or numeric format. • group specifies the group setting for the file in text or numeric format. • permissions specifies what privileges can be exercised by Owner/Group/Other with respect to the file in text or numeric format. For example, ralphw:1007 rw-r- - - - - specifies that the file owner (ralphw) can read and edit the log file, members of group 1007 can read (but not edit) the log file, and that other users cannot access the log file.
MaxMinutesPerFile	Specifies how often the current authentication acceptance report is closed and a new file opened. • If set to n (where n is a number greater than 0), a new report is generated every n minutes. • If set to 0, a new report is generated once every 24 hours, at midnight local time. Default value is 0. Note: The value entered for MaxMinutesPerFile determines the file name of the generated report.
QuoteBinary	• If set to 1, binary values written to the report are enclosed in quotes. • If set to 0, quotes are not used. Set this value according to the format expected by the application that processes the entries. Default value is 1.
QuoteInteger	• If set to 1, integer values written to the report are enclosed in quotes. • If set to 0, quotes are not used. Set this value according to the format expected by the application that processes the entries. Default value is 1.
QuoteIPAddress	• If set to 1, IP addresses written to the report are enclosed in quotes. • If set to 0, quotes are not used. Set this value according to the format expected by the application that processes the entries. Default value is 1.
QuoteText	• If set to 1, text strings written to the report are enclosed in quotes. • If set to 0, quotes are not used. Set this value according to the format expected by the application that processes the entries. Default value is 1.
QuoteTime	• If set to 1, time and date values written to the report are enclosed in quotes. • If set to 0, quotes are not used. Set this value according to the format expected by the application that processes the entries. Default value is 1.

Parameter	Function
UTC	• If set to 1, time and date values are provided according to universal time coordinates (UTC, formerly known as Greenwich mean time or GMT). • If set to 0, time and date values reflect local time. Default value is 0.

authReportBadSharedSecret.ini File

Used by: GEE, EE
Not Used By: —

The authReportBadSharedSecret.ini initialization file specifies options for the invalid shared secret report, which is an ASCII comma-delimited file that records information about requests received from known RADIUS clients that used an invalid shared secret. This condition is only detectable if the authentication request contained a Message-Authenticator attribute, which is required if credentials are of an EAP type but optional if credentials are PAP, CHAP, or MS-CHAP. (In the case of PAP, an invalid shared secret will not be detected, but will result in an Access-Reject response as the user password is decrypted into incorrect characters.)

If the MaxMinutesPerFile parameter is set to 0, the file name of the bad shared secret report is badSharedSecret_yyyymmdd.csv (where yyyymmdd identifies the date the report was generated.) If the MaxMinutesPerFile parameter is set to a value greater than 0, the file name of the report is badSharedSecret_ yyyymmdd_hhmm.csv (where yyyymmdd identifies the date and hhmm identifies the time the report was generated).

[Attributes] Section

The [Attributes] section of authReportBadSharedSecret.ini lists the attributes logged in the invalid shared secret report.

You can configure what is logged to the invalid shared secret report by entering attributes in the [Attributes] section in the sequence in which they should appear. This lets you design the content and column order of any spreadsheets that you plan to create based upon the silent discard/bad shared secret report.

The syntax of the [Attributes] section is as follows:

```
[Attributes]
AttributeName=
AttributeName=
```

For example:

```
[Attrib
utes]
User-
Name
=
NAS-IP-Address=
```

The [Attributes] section lists one AttributeName on each line. You must ensure that an equal sign (=) immediately follows each AttributeName, with no spaces in between. Improperly formatted entries are ignored.

Each AttributeName in the [Attributes] section must be defined in a standard RADIUS or vendor-specific dictionary file (.dct) installed on the Steel-Belted Radius server.

Note: The first three attributes in each invalid shared secret report entry (Date, Time, and RADIUS- Client) are always enabled, and cannot be re-ordered or deleted. Therefore, these attributes do not appear in the [Attributes] section of the authReportBadSharedSecret.ini file.

[Settings] Section

The [Settings] section of authReportBadSharedSecret.ini specifies the operational characteristics of the invalid shared secret report. Sample syntax is as follows:

```
[Settings]
BufferSize = 131072
DaysToKeep = 1
LineSize = 4096
LogfilePermissions = ralphw:1007 rw-r- - - - -
MaxMinutesPerFile = 0
QuoteBinary = 1
QuoteInteger = 1
QuoteIpAddress = 1
QuoteText = 1
QuoteTime = 1
UTC = 0
```

Table 13: authReportBadSharedSecret.ini [Settings] Syntax

Parameter	Function
BufferSize	The size of the buffer used in the logging process, in bytes. Default value is 131072.
DaysToKeep	Specifies the number of days the Steel-Belted Radius server should retain each invalid shared secret report. Default value is 1 (one day).
LineSize	The maximum size of a single log line. The allowable range is 1024 to 32768. Default value is 4096.
LogFilePermissions (Linux only)	Specifies the owner and access permission setting for the invalid shared secret report (badSharedSecret_yyyymmdd.csv) file. Enter a value for the LogFilePermissions setting in owner:group permissions format, where: • owner specifies the owner of the file in text or numeric format. • group specifies the group setting for the file in text or numeric format. • permissions specifies what privileges can be exercised by Owner/Group/Other with respect to the file in text or numeric format. For example, ralphw:1007 rw-r- - - - - specifies that the file owner (ralphw) can read and edit the log file, members of group 1007 can read (but not edit) the log file, and that other users cannot access the log file.

Parameter	Function
MaxMinutesPerFile	Specifies how often the current report is closed and a new file opened. • If set to n (where n is a number greater than 0), a new report file is generated every n minutes. • If set to 0, a new report file is generated once every 24 hours, at midnight local time. Default value is 0. Note: The value entered for MaxMinutesPerFile determines the file name of the generated report.
QuoteBinary	• If set to 1, binary values written to the report are enclosed in quotes. • If set to 0, quotes are not used. Set this value according to the format expected by the application that processes the entries. Default value is 1.
QuoteInteger	• If set to 1, integer values written to the report are enclosed in quotes. • If set to 0, quotes are not used. Set this value according to the format expected by the application that processes the entries. Default value is 1.
QuoteIPAddress	• If set to 1, IP addresses written to the report are enclosed in quotes. • If set to 0, quotes are not used. Set this value according to the format expected by the application that processes the entries. Default value is 1.
QuoteText	• If set to 1, text strings written to the report are enclosed in quotes. • If set to 0, quotes are not used. Set this value according to the format expected by the application that processes the entries. Default value is 1.
QuoteTime	• If set to 1, time and date values written to the report are enclosed in quotes. • If set to 0, quotes are not used. Set this value according to the format expected by the application that processes the entries. Default value is 1.
UTC	• If set to 1, time and date values are provided according to universal time coordinates (UTC, formerly known as Greenwich mean time or GMT). • If set to 0, time and date values reflect local time. Default value is 0.

authReportReject.ini File

Used by:
GEE, EE
Not Used By: —

The authReportReject.ini initialization file specifies options for the authentication rejection report, which is an ASCII comma-delimited file that records authentication rejections.

If the MaxMinutesPerFile parameter is set to 0, the file name of the authentication rejection report is rejects_ yyyymmdd.csv (where yyyymmdd identifies the date the report was generated.) If the MaxMinutesPerFile parameter is set to a value greater than 0, the file name of the report is rejects_yyyymmdd_hhmm.csv (where yyyymmdd identifies the date and hhmm identifies the time the report was generated).

[Attributes] Section

The [Attributes] section of authReportReject.ini lists the attributes logged in the authentication rejection report.

You can configure what is logged to the authentication rejection report by entering attributes in the [Attributes] section in the sequence in which they should appear. This lets you design the content and column order of any spreadsheets that you plan to create based upon the reject report.

The syntax of the [Attributes] section is as follows:

```
[Attributes]
AttributeName=
AttributeName=
.
.
.
```

For example:

```
[Attrib
utes]
Service
-Type=
Source-IP-
Address=
Source-UDP-
Port=
```

The [Attributes] section lists one AttributeName on each line. You must ensure that an equal sign (=) immediately follows each AttributeName, with no spaces in between. Improperly formatted entries are ignored.

Each AttributeName in the [Attributes] section must be defined in a standard RADIUS or vendor-specific dictionary file (.dct) installed on the Steel-Belted Radius server.

The following attributes in each authentication rejection report entry are always enabled, and cannot be re- ordered or deleted:

- Date – Identifies the date of the authentication rejection.
- Time – Identifies the time of the authentication rejection.
- RADIUS-Client – Identifies the RADIUS client that received the authentication rejection.
- User-Name – Identifies the name of the user that was rejected.
- Reject-Method – Identifies the most relevant authentication method that rejected the user. If this information is unavailable, the parameter is set to Unknown.

- Reject-Reason – Identifies one of the following reasons for the authentication rejection:
  - System error
  - Blacklisted user
  - Invalid request
  - Database or directory not available
  - User not in database
  - Credentials invalid
  - User name or credential incorrect
  - Post-processing error
  - Unknown
- Reject-Log – Identifies the reason for the authentication request in language supplied by the authentication method. If a reason is not supplied, the parameter is set to Unavailable.

These attributes do not appear in the [Attributes] section of the authReportReject.ini file.

[Settings] Section

The [Settings] section of authReportReject.ini specifies the operational characteristics of the authentication rejection report. Sample syntax is as follows:

```
[Setti
ngs]
UTC
= 0
BufferSize = 131072
MaxMinutesPerFile = 0
DaysToKeep = 1
LineSize = 4096
LogfilePermissions = ralphw:1007 rw-r- - - - -
QuoteInteger = 1
QuoteIpAddress = 1
QuoteText = 1
QuoteTime = 1
QuoteBinary = 1
```

Table 14: authReportReject.ini [Settings] Syntax

Parameter	Function
BufferSize	Specifies the size of the buffer used in the logging process, in bytes. Default value is 131072.
DaysToKeep	Specifies the number of days the Steel-Belted Radius server should retain each rejection report. Default value is 1 (one day).
LineSize	Specifies the maximum size of a single log line. The allowable range is 1024 to 32768.

Parameter	Function
	Default value is 4096.
LogFilePermissions (Linux only)	Specifies the owner and access permission setting for the authentication rejection report (rejects_yyyymmdd.csv) file. Enter a value for the LogFilePermissions setting in owner:group permissions format, where: • owner specifies the owner of the file in text or numeric format. • group specifies the group setting for the file in text or numeric format. • permissions specifies what privileges can be exercised by Owner/Group/Other with respect to the file in text or numeric format. For example, ralphw:1007 rw-r- - - - - specifies that the file owner (ralphw) can read and edit the log file, members of group 1007 can read (but not edit) the log file, and that other users cannot access the log file.
MaxMinutesPerFile	Specifies how often the current report is closed and a new file opened. • If set to n (where n is a number greater than 0), a new report file is generated every n minutes. • If set to 0, a new report file is generated once every 24 hours, at midnight local time. Default value is 0. Note: The value entered for MaxMinutesPerFile determines the file name of the generated report.
QuoteBinary	• If set to 1, binary values written to the report are enclosed in quotes. • If set to 0, quotes are not used. Set this value according to the format expected by the application that processes the entries. Default value is 1.
QuoteInteger	• If set to 1, integer values written to the report are enclosed in quotes. • If set to 0, quotes are not used. Set this value according to the format expected by the application that processes the entries. Default value is 1.
QuoteIPAddress	• If set to 1, IP addresses written to the report are enclosed in quotes. • If set to 0, quotes are not used. Set this value according to the format expected by the application that processes the entries. Default value is 1.
QuoteText	• If set to 1, text strings written to the report are enclosed in quotes. • If set to 0, quotes are not used. Set this value according to the format expected by the application that processes the entries. Default value is 1.
QuoteTime	• If set to 1, time and date values written to the report are enclosed in quotes. • If set to 0, quotes are not used. Set this value according to the format expected by the application that processes the entries. Default value is 1.
UTC	• If set to 1, time and date values are provided according to universal time coordinates (UTC, formerly known as Greenwich mean time or GMT). • If set to 0, time and date values reflect local time.

Parameter	Function
	Default value is 0.

authReportUnknownClient.ini File

Used by: GEE, EE
Not Used By: —

The authReportUnknownClient.ini initialization file specifies options for the unknown authentication client report, which is an ASCII comma-delimited file produced by the Steel-Belted Radius server that identifies requests received from unknown RADIUS clients.

If the MaxMinutesPerFile parameter is set to 0, the file name of the unknown authentication client report is unknownClient_yyyymmdd.csv (where yyyymmdd identifies the date the report was generated.) If the MaxMinutesPerFile parameter is set to a value greater than 0, the file name of the report is unknownClient_ yyyymmdd_hhmm.csv (where yyyymmdd identifies the date and hhmm identifies the time the report was generated.)

[Attributes] Section

The [Attributes] section of authReportUnknownClient.ini lists the attributes logged in the unknown client report.

You can configure what is logged to the unknown client log by entering attributes in the [Attributes] section in the sequence in which they should appear. This lets you design the content and column order of any spreadsheets that you plan to create based upon the unknown client log.

The syntax of the [Attributes] section is as follows:

```
[Attributes]
AttributeName=
AttributeName=
```

For example:

```
[Attrib
utes]
User-
Name
=
```

The [Attributes] section lists one AttributeName on each line. You must ensure that an equal sign (=) immediately follows each AttributeName, with no spaces in between. Improperly formatted entries are considered invalid and are ignored.

Each AttributeName in the [Attributes] section must be defined in a standard RADIUS or vendor-specific dictionary file (.dct) installed on the Steel-Belted Radiusserver.

Note: The first three attributes in each invalid shared secret report entry (Date, Time, and RADIUS- Client) are always enabled, and cannot be re-ordered or deleted. Therefore, these attributes do not appear in the [Attributes] section of the authReportBadSharedSecret.ini file.

[Settings] Section

The [Settings] section of authReportUnknownClient.ini specifies the operational characteristics of the unknown authentication client report. Sample syntax is as follows:

```
[Settings]
BufferSize =
131072
DaysToKeep = 1
LineSize = 4096
LogfilePermissions = ralphw:1007 rw-r- - - - -
MaxMinutesPerFile = 0
QuoteBinary = 1
QuoteInteger = 1
QuoteIpAddress = 1
QuoteText = 1
QuoteTime = 1
UTC = 0
```

Table 15: authReportUnknownClient.ini [Settings] Syntax

Parameter	Function
BufferSize	The size of the buffer used in the logging process, in bytes. Default value is 131072.
DaysToKeep	Specifies the number of days the Steel-Belted Radius server should retain each unknown client report. Default value is 1 (one day).
LineSize	The maximum size of a single log line. The allowable range is 1024 to 32768. Default value is 4096.
LogFilePermissions (Linux only)	Specifies the owner and access permission setting for the unknown authentication client report (unknownClient_yyyymmdd_hhmm.csv) file. Enter a value for the LogFilePermissions setting in owner:group permissions format, where: • owner specifies the owner of the file in text or numeric format. • group specifies the group setting for the file in text or numeric format. • permissions specifies what privileges can be exercised by Owner/Group/Other with respect to the file in text or numeric format. For example, ralphw:1007 rw-r- - - - - specifies that the file owner (ralphw) can read and edit the log file, members of group 1007 can read (but not edit) the log file, and that other users cannot access the log file.
MaxMinutesPerFile	Specifies how often the current report is closed and a new file opened. • If set to n (where n is a number greater than 0), a new report file is generated every n minutes. • If set to 0, a new report file is generated once every 24 hours, at midnight local time. Default value is 0. Note: The value entered for MaxMinutesPerFile determines the file name of the generated report.
QuoteBinary	• If set to 1, binary values written to the report are enclosed in quotes. • If set to 0, quotes are not used.

Parameter	Function
	Set this value according to the format expected by the application that processes the entries. Default value is 1.
QuoteInteger	• If set to 1, integer values written to the report are enclosed in quotes. • If set to 0, quotes are not used. Set this value according to the format expected by the application that processes the entries. Default value is 1.
QuoteIPAddress	• If set to 1, IP addresses written to the report are enclosed in quotes. • If set to 0, quotes are not used. Set this value according to the format expected by the application that processes the entries. Default value is 1.
QuoteText	• If set to 1, text strings written to the report are enclosed in quotes. • If set to 0, quotes are not used. Set this value according to the format expected by the application that processes the entries. Default value is 1.
QuoteTime	• If set to 1, time and date values written to the report are enclosed in quotes. • If set to 0, quotes are not used. Set this value according to the format expected by the application that processes the entries. Default value is 1.
UTC	• If set to 1, time and date values are provided according to universal time coordinates (UTC, formerly known as Greenwich mean time or GMT). • If set to 0, time and date values reflect local time. Default value is 0.

blacklist.ini File

Used by: GEE
Not used by: EE

The blacklist.ini configuration file enables and configures blacklist settings. Only one profile can be created for the purposes of blacklisting, and any login attempt that matches that profile is blocked. An authentication request matches the blacklist profile if the attributes in the request match the checklist attributes of the profile. The profile can contain multiple attributes, and if any of the attributes match those of the profile, the attempt is rejected.

The blacklist.ini file contains one configuration section called [Settings] (Table 16), which has the following settings:

```
[Settings]
Enable =
<0|1>
IncludeProxy =
<0|1> Profile =
```

profile

Table 16: blacklist.ini Syntax

Parameter	Function
Enable	• If set to 1, blacklisting is enabled. • If set to 0, blacklisting is disabled. Default value is 0.
IncludeProxy	• If set to 1, blacklisting is configured to include proxy requests, meaning that it is applied to all authentication requests. • If set to 0, blacklisting is configured only to local authentication requests. Default value is 0.
Profile	Specifies the name of the blacklist profile in the Steel-Belted Radius database.

The following example enables the blacklist feature and specifies Steel-Belted Radius should use the BlockedNumbers profile to filter authentication requests.

```
[Setti
ngs]
Enabl
e = 1
IncludeProxy = 0
Profile=BlockedNumbers
```

The BlockedNumbers profile called by this blacklist.ini file specifies checklist attributes Steel-Belted Radius uses to reject authentication requests. The following entries in the BlockedNumbers profile identify Calling-Station-Id phone numbers used by rogue users you want to block.

```
Calling-Station-Id = 617-999-9119
Calling-Station-Id = 800-515-7889
```

lockout.ini File

Used by:
GEE Not used by:
EE

The lockout.ini configuration file enables and configures account lockout settings. Account lockout lets you disable an account after a configurable number of failed login attempts within a configurable period. For example, if a user enters an incorrect password three times within two minutes, Steel-Belted Radius can lock out the user's account temporarily. During the lockout period, the user cannot log in, even with the correct password. Attempts to authenticate against a locked out account cause Steel-Belted Radius to respond with an Access-Reject message immediately.

The lockout.ini file contains one configuration section called [Settings], which has settings similar to the following:

```
[Setti
ngs]
Enabl
e = 0
```

Rejects = 3
Within = 120
Lockout = 600

Table 17: lockout.ini Syntax

Parameter	Function
Enable	• If set to 0, lockout is disabled. • If set to 1, lockout is enabled. Default value is 0.
Lockout	Specifies the lockout period in seconds. Default value is 600 seconds (10 minutes).
Rejects	Specifies the number of rejected attempts prior to lockout. Default value is 3.
Within	Specifies the period in seconds during which a specified number of rejects causes a lockout. Default value is 120 seconds (two minutes).

[ClientExclusionList] Section

You can add a ClientExclusionList section to the lockout.ini file. Use this section to list clients which are excepted from the lockout functionality. Enter one client name per line. For example,

```
[ClientExclusionList]
exampleclient1
exampleclient2
```

[ExcludedUsers] Section

You can an ExcludedUsers section to the lockout.ini file. Use this section to prevent certain reserved user names, such as "anonymous", from being locked out. Enter one user name per line. For example,

```
[ExcludedUsers]
anonymous
```

Note: If you enable the lockout facility in Steel-Belted Radius and you use a tunneled authentication method (TTLS or PEAP) with a prefetch-capable method (native user, SQL, or LDAP) and an enabled EAP protocol (MS-CHAPv2, MD5-Challenge, LEAP, TLS), then you must enable Handle via Auto-EAP First in that prefetch-capable method to prevent the outer username (anonymous) from being added to the lockout list.

Otherwise, when Steel-Belted Radius receives an authentication request that uses an unconfigured EAP method, Steel-Belted Radius will reject the user (because the EAP method is not configured) and add the outer username (anonymous) to its lockout list. This will result in all users with an outer authentication name of anonymous being rejected until the lockout period expires.

redirect.ini File

Used by:
GEE Not used by:
EE

Account redirection lets you flag an account for special processing after a configurable number of failed login attempts within a configurable time period. The redirect.ini initialization file specifies the settings used for account redirection when users forget or mis-enter their passwords.

[Settings] Section

The [Settings] section of redirect.ini (Table 18) enables and configures account redirection settings.

Table 18: redirect.ini [Settings] Syntax

Parameter	Function
Enable	• If set to 0, account redirection is disabled. • If set to 1, account redirection is enabled. Default value is 0. Note: Account redirection and account lockout are incompatible. Do not enable account redirection if account lockout is enabled.
Lockout	The number of seconds in the account redirection lockout period. For example, a lockout period of 86,400 seconds locks a user out for one day if account redirection processing fails to authenticate the user. Default value is 600 seconds (10 minutes).
Profile	The name of the global profile that supplies the values and attributes used for the user after account redirection is triggered. Default value is Redirect.
Redirect	The number of seconds during which a user is in redirect state. If the redirectionv period elapses without another user authentication request, the user is returned to normal state. Default value is 120 seconds.
Rejects	The number of rejected attempts prior to redirection. Default value is 3.
Within	The period in seconds during which a specified number of rejects causes account redirection. Default value is 180 seconds (3 minutes).

For example, the following [Settings] section of redirect.ini specifies that, if a user fails authentication three times within 180 seconds, the user account is placed into redirect state. If the user does not submit another authentication request within 120 seconds of entering redirect state, the user account is restored to normal state.

```
[Setti
ngs]
Enabl
e = 0
Rejects = 3
Within = 180
```

Redirect = 120
Profile =
RedirectProfile
Lockout = 86400

If the user submits another authentication request within 120 seconds of entering redirect state, the user is accepted without authentication or authorization processing, the user's account is placed into accept-pending state, and the RADIUS accept message for the user contains the values and attributes specified in the global RedirectProfile profile. (These values or attributes could be used by an external customer process to direct the user to a secure web page that asks for alternative authentication information or billing information; the external process might then mail the user an access password if the user satisfies the external process requirements.)

When a user is in accept-pending state, the user's next authentication request determines whether Steel-Belted Radius accepts or locks out the user:

- If the next authentication is successful, the user account is returned to normal state.
- If the next authentication fails to accept the user, the user account is locked out for 86,400 seconds (one day). During this lockout period, authentication requests for this user are rejected automatically, even if the user enters the correct password.

[ClientExclusionList] Section

The [ClientExclusionList] section of redirect.ini identifies the RADIUS clients that are excluded from account redirection processing. Each entry in the [ClientExclusionList] section of redirect.ini consists of the name of a RADIUS client device, as configured in the Steel-Belted Radius database.

securid.ini File

Used by: GEE, EE
Not Used By: —

The securid.ini file lets you replace the default prompt strings used in RSA SecurID authentication with customized strings. Customized prompt strings are useful in situations where authentication is to be performed by means of RSA SecurID and the default prompt strings are too long for the screen on the authentication device.

If the securid.ini file is present in the Steel-Belted Radius server directory, Steel-Belted Radius uses prompt strings specified in the file instead of the default prompt strings. Sets of strings can be substituted in whole or in part. If a string is not represented by an entry in the securid.ini file, Steel-Belted Radius uses the default prompt string.

[Configuration] Section

The [Configuration] section of securid.ini specifies RSA SecurID access settings.

Enable = 1
AllowSystemPins = 0
CheckUserAllowedByClient = 1
DefaultProfile = DEFAULT

Table 19: securid.ini [Configuration] Syntax

Parameter	Function
Enable	• If set to 1, Steel-Belted Radius can authenticate users by means of RSA SecurID. • If set to 0, Steel-Belted Radius cannot authenticate users by means of RSA SecurID.

Parameter	Function
	Default value is 1.
AllowSystemPins	• If set to 1, users who are configured in the RSA Authentication Manager to receive a system-generated PIN when in New PIN mode are accepted. • If set to 0, users who are configured in the RSA Authentication Manager to receive system-generated PIN when in New PIN mode are rejected. Default value is 0.
CheckUserAllowed ByClient	• If set to 1, the RADIUS server verifies the user is allowed to connect through the network access device. • If set to 0, the RADIUS server does not verify the user is allowed to connect through the network access device. Default value is 1. Note: If this parameter is set to 1, RADIUS clients must be configured as Agent Hosts in RSA Authentication Manager.
DefaultProfile	Default profile to be assigned to a user if the RSA Authentication Manager does not return a profile. Default value is DEFAULT.

[Server_Settings] Section

The [Server_Settings] section of securid.ini specifies settings for Extended One-Time Password (EOTP or EAP-15) and Protected One-Time Password (POTP or EAP-32) authentication.

```
[Server_Settings]
Greeting =
Return_MPPE_Keys = 1
```

Table 20: securid.ini [Server_Settings] Syntax

Parameter	Function
Greeting	A string of as many as 80 characters returned to a network access device after a user is authenticated. For example, "Welcome to RSA Security Software."
Return_MPPE_Keys	• If set to 0, the module does not forward RADIUS MS-MPPE Send-Key and MS-MPPE-Recv-Key attributes. • If set to 1, the module includes RADIUS MS-MPPE-Send-Key and MS-MPPE-Recv-Key attributes in the final RADIUS Access-Accept response sent to the Access Point. This is necessary for the Access Point to key the WEP encryption. If the Access Point is authenticating only end users and WEP is not being used, this attribute may be set to 0. Default value is 1.

[Prompts] Section

If the securid.ini file is present in the Steel-Belted Radius server directory, Steel-Belted Radius uses prompt strings specified in the file instead of the default prompt strings. Sets of strings can be substituted in whole or in part. If a string is not represented by an entry in the securid.ini file, Steel-Belted Radius uses the default prompt string.

Substitution String Formats

Substitution strings use %s to mark locations at which variable text is to be substituted. Strings can have no %s placeholders, exactly one %s placeholder, or exactly two %s placeholders. When writing your own prompt

strings, you must supply strings with the expected number of %s placeholders. String names include a reminder suffix that reflects the number of %s placeholders:

- Strings that require two %s placeholders have names with a _S_S suffix. The first %s placeholder typically presents a number range ("4 to 8"). The second %s placeholder specifies "characters" or "digits" (or the equivalent, as configured in the Characters and Digits settings).
- Strings that require one %s placeholder have names with a _S suffix. The %s placeholder is replaced with a system-generated PIN.
- Strings that do not require %s placeholders have names with no suffix.

If a string in the securid.ini file is formatted incorrectly, it is ignored and the default prompt string is used.

Table 21 lists formatting conventions for the securid.ini file.

Table 21: Substitution String Formatting Conventions

Convention	Explanation
\b	Backspace; not typically used
\f	Formfeed
\n	Newline; typically used in conjunction with \r
\r	Carriage return; typically used in conjunction with \n
\t	Horizontal tab
\v	Vertical tab; not typically used
\\	Displayed backslash
\'	Displayed single-quote character
\"	Displayed double-quote character

If other characters in a substitution string are preceded by a backslash, the backslash is ignored and the character is displayed unchanged.

Quoted Strings

Trailing white space is ignored when an unquoted prompt string is read into Steel-Belted Radius. If you want a substitution string to include trailing white space, insert double-quote marks at the beginning and end of the string, enclosing the white space you want to include. For example, if you want a string to be displayed as the word PIN followed by a colon followed by a single space, you would enter StringName="PIN: " (with a space between the colon and the closing double-quote character).

Example 1: Verbose Substitution Strings

Figure 1 lists the default prompt strings, which may be too long for some SecurID displays. Although text lines in this display appear to wrap to a second line, text wrapping is not supported in securid.ini entries.

```
;[Prompts]
;InputNextCode = \r\nPlease Enter the Next Code from Your Token:
;InputMustChoose_S_S = \r\n Enter your new PIN, containing %s
```

```
%s,\r\n or\r\n<Ctrl-D>to cancel the New PIN procedure:
;InputCannotChoose = \r\n Press <Return> to generate a new PIN and display it on the screen,\r\n or\r\n <Ctrl-D> to cancel the New PIN procedure:
;InputMayChoose_S_S=\r\n Enter your new PIN, containing %s %s,\r\n or\r\n Press <Return> to generate a new PIN and display it on the screen,\r\n or\r\n <Ctrl-D> to cancel the New PIN procedure:
;InputReadyForPin =\r\n\r\nARE YOU PREPARED TO HAVE THE SYSTEM GENERATE A PIN? (y or n) [n]:
;InputReadyForPin_1_S =\r\n\r\nPIN: %s\r\n\r\n 10 second display or Hit RETURN to continue.
;InputReenterPin=\r\n Please re-enter new PIN:
;InputReenterPin_1=\r\nPINs do not match. Please try again.\r\n
;OutputReject =\r\n\r\nPIN rejected. Please try again.\r\n\r\nEnter PASSCODE:
;OutputChange = \r\n\r\nWait for the code on your card to change, then log in with the new PIN\r\n\r\nEnter PASSCODE:
;OutputAccepted = \r\nPASSCODE Accepted\r\n
;OutputDenied = \r\nAccess Denied\r\n\r\n\r\nEnter PASSCODE:
;OutputNoPassReqd = \r\nPASSCODE Not Required\r\n
;OutputDeniedFinal = \r\nAccess Denied\r\n\r\n
;Characters = characters
;Digits=digits
```

Figure 1 Verbose Substitution Strings

Example 2: 2 x 40 Display Substitution Strings

Figure 2 displays prompt strings designed for a 2 line x 40 character display. Although text lines in this display appear to wrap to a second line, text wrapping is not supported in securid.ini entries.

```
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;;
; BEGINNING OF 2 lines by 40 characters prompts, these use the full 40
; character width (not including "\r\n") and one or two lines
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;;
;[Prompts]
;InputNextCode=Please Enter the Next Code from\r\nYour Token
;InputMustChoose_S_S=Enter your new PIN (%s %s)
;InputCannotChoose = Press <Return> to generate a new PIN and\r\ndisplay it
;InputMayChoose_S_S = Enter new PIN (%s %s) or press\r\n<Return> to generate a new one
;InputReadyForPin=AREYOU PREPAREDTOHAVETHESYSTEM\r\nGENERATEA PIN? (y or n) [n]
;InputReadyForPin_1_S = PIN: %s, 10 second display or\r\npress <Return> to continue
;InputReenterPin=Please re-enter new PIN
;InputReenterPin_1=PINs do not match,\r\nPlease try again
;OutputReject = PIN Rejected, please try again\r\nEnter PASSCODE
```

```
;OutputChange = Wait for the code on your card to change\r\n
then  log in with new PIN, Enter PASSCODE
;OutputAccepted = PASSCODE Accepted
;OutputDenied = Access Denied\r\nEnter PASSCODE
;OutputNoPassReqd = PASSCODE Not Required
;OutputDeniedFinal = Access Denied
;Characters = chars
;Digits = digits
```

Figure 2 2x50 Display Substitution Strings

Example 3: Terse Substitution Strings

Figure 3 displays prompt strings designed to be parsed by a program at the client endpoint rather than read by a user.

```
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;;
; BEGINNING OF extremely terse prompts. These are appropriate for automatic
; interpretation by another program which parses the prompts. A well
trained
; end user could use these.
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;;
;[Prompts]
;InputNextCode=Next code
;InputMustChoose_S_S = Must choose
;InputCannotChoose = Cannot choose
;InputMayChoose_S_S=May choose(%s,%s)
;InputReadyForPin=Ready for pin
;InputReadyForPin_1_S=Ready for pin 1
;InputReenterPin = Reenterpin
;InputReenterPin_1 = Reenter pin 1
;OutputReject = Reject
;OutputChange = Change
;OutputAccepted = Accepted
;OutputDenied = Denied
;OutputNoPassReqd = No pass reqd
;OutputDeniedFinal = Denied final
;Characters = chars
;Digits = digits
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
```

Figure 3 Terse Substitution Strings

statlog.ini File

Used by: GEE
Not Used By: EE

The statlog.ini initialization file configures the Steel-Belted Radius statistics log file (yyyymmdd.statlog), which periodically records server statistics to a comma-delimited ASCII file. The statistics log provides a mechanism for creating snapshots of user-selected server statistics.

- The first line in a *.statlog file lists all column headings in double quotes ("Date", "Time", ...).

- The first column in a *.statlog file identifies the current date in yyyy-mm-dd format in double-quotes ("2006-02-13"). The *.statlog file uses the local date, not the date in the UTC time zone, when it records date information.
- The second column in a *.statlog file identifies the current time in hh:mm:ss format in double-quotes ("14:13:22"). The *.statlog file uses the local time, not the time in the UTC time zone, when it records time information.
- If statistics logging is enabled, a new statistics logging file is created the first time the server is started each day. At midnight, the server closes the old statistics log file and starts a new one with a file name that reflects the new date.
- If you restart the Steel-Belted Radius server and a *.statlog file exists for the current day, the server appends new information to the existing *.statlog file. When the server is restarted, the timer for capturing statistics snapshots is restarted; for example, a server configured to record statistics every 10 minutes captures statistics at 14:10:00. If the server is restarted at 14:15:00, it captures system statistics immediately (14:15:00) and 10 minutes thereafter (14:25:00); it does not try to capture statistics at 14:20:00 (10 minutes after the capture before the restart).
- If you change the order or contents of the list of statistics to be recorded and restart the server, Steel-Belted Radius detects the change and writes an entry with the new column headers to the current *.statlog file before writing new data records into the file.

Note: When you change the order or contents of the list of statistics recorded in the *.statlog file, Steel-Belted Radius creates the statloghdr.dat checkpoint file in the radiusdir directory. Do not modify or delete the statloghdr.dat file.

[Settings] Section

The [Settings] section of statlog.ini (Table 22) specifies whether the statistics log file is enabled, who can access the statistics log file, how frequently the server writes information to the statistics log file, and the number of days statistics log files are retained.

Table 22: statlog.ini [Settings] Syntax

Parameter	Function
Enable	• If set to 1, the Steel-Belted Radius server periodically writes statistics information to the yyyymmdd.statlog file. • If set to 0, the Steel-Belted Radius server does not update the yyyymmdd.statlog file. Default value is 0.
LogFilePermissions (Linux only)	Specifies the owner and access permission setting for the *.statlog file. Enter a value for the LogFilePermissions setting in owner:group permissions format, where: • owner specifies the owner of the file in text or numeric format. • group specifies the group setting for the file in text or numeric format. • permissions specifies what privileges can be exercised by Owner/Group/Other with respect to the file in text or numeric format. For example, ralphw:1007 rw-r- - - - - specifies that the file owner (ralphw) can read and edit the log file, members of group 1007 can read (but not edit) the log file, and other users cannot access the log file.

Parameter	Function
Interval-Seconds	Specifies the number of seconds (in the range 1–3600) that the Steel-Belted Radius server waits before writing new statistics information to the statistics log. Default value is 600 seconds (10 minutes).
Days-To-Keep	Specifies the number of days (in the range 1–365) the statistics log is retained by the Steel-Belted Radius server. When the specified number of days has elapsed, the statistics log is automatically purged. Default value is 7 days.

For example:

```
[Setti
ngs]
Enabl
e = 0
;LogfilePermissions = ralphw:1007 rw-r- - - - -
;Interval-Seconds = 600
;Days-To-Keep = 7
```

[Statistics] Section

The [Statistics] section of statlog.ini (Table 23) identifies the statistics you want included in the snapshot. Each entry in this section takes the format Source/Statistic, where Source identifies the LCI statistics container that holds the statistic counter you want and Statistic identifies the statistic by name.

Statistics are written to the log file in the order in which they are listed in the [Statistics] section.

Table 23: statlog.ini [Statistics] Syntax

Parameter	Function
Enable	Specifies is the name of the LCI statistics container that holds the specified statistic. Supported values for *Source* are: • Server • Authentication • Accounting • Proxy • Rate
Statistic	Specifies the name of the statistic you want to record in the log. Statistics in the Server LCI container are: • Accounting-Threads • Authentication-Threads • High-Acct-Threads • High-Acct-Threads-Since-Reset • High-Auth-Threads • High-Auth-Threads-Since-Reset • High-Total-Threads

Parameter	Function
	• High-Total-Threads-Since-Reset • Max-Acct-Threads • Max-Auth-Threads • Max-Total-Threads • Total-Threads Statistics in the Authentication LCI container are: • Accept • Dropped-Packet • Failed-Authentication • Failed-On-Check-List • Insufficient-Resources • Invalid-Request • Proxy-Failure • Reject • Rejected-By-Proxy • Silent-Discard • Total-Retry-Packets • Total-Transactions • Transactions-Retried Statistics in the Accounting LCI container are: • Dropped-Packet • Insufficient-Resources • Interim • Invalid-Client • Invalid-Request • Invalid-Shared-Secret • Off • On • Proxy-Failure • Start • Stop • Total-Retry-Packets • Total-Transactions • Transactions-Retried Statistics in the Proxy LCI container are: • Accounting • Authentication • Insufficient-Resources • Invalid-Response • Invalid-Shared-Secret • Timed-Out • Total-Retry-Packets • Total-Transactions • Transactions-Retried Statistics in the Rate LCI container are:

Parameter	Function
	• Acct-Start-Average-Rate • Acct-Start-Current-Rate • Acct-Start-Peak-Rate • Acct-Stop-Average-Rate • Acct-Stop-Current-Rate • Acct-Stop-Peak-Rate • Auth-Accept-Average-Rate • Auth-Accept-Current-Rate • Auth-Accept-Peak-Rate • Auth-Reject-Average-Rate • Auth-Reject-Current-Rate • Auth-Reject-Peak-Rate • • Auth-Request-Average-Rate • Auth-Request-Current-Rate • Auth-Request-Peak-Rate Statistics in the Rate LCI container are: • Proxy-Acct-Fail-Proxy-Average-Rate • Proxy-Acct-Fail-Proxy-Current-Rate • Proxy-Acct-Fail-Proxy-Peak-Rate • Proxy-Acct-Request-Average-Rate • Proxy-Acct-Request-Current-Rate • Proxy-Acct-Request-Peak-Rate • Proxy-Auth-Rej-Proxy-Average-Rate • Proxy-Auth-Rej-Proxy-Current-Rate • Proxy-Auth-Rej-Proxy-Error-Average-Rate • Proxy-Auth-Rej-Proxy-Error-Current-Rate • Proxy-Auth-Rej-Proxy-Error-Peak-Rate • Proxy-Auth-Rej-Proxy-Peak-Rate • Proxy-Auth-Request-Average-Rate • Proxy-Auth-Request-Current-Rate • Proxy-Auth-Request-Peak-Rate • Proxy-Fail-Badresp-Average-Rate • Proxy-Fail-Badresp-Current-Rate • Proxy-Fail-Badresp-Peak-Rate • Proxy-Fail-Badsecret-Average-Rate • Proxy-Fail-Badsecret-Current-Rate • Proxy-Fail-Badsecret-Peak-Rate • Proxy-Fail-Missingresr-Average-Rate • Proxy-Fail-Missingresr-Current-Rate • Proxy-Fail-Missingresr-Peak-Rate • Proxy-Fail-Timeout-Average-Rate • Proxy-Fail-Timeout-Current-Rate • Proxy-Fail-Timeout-Peak-Rate • Proxy-Retries-Average-Rate • Proxy-Retries-Current-Rate • Proxy-Retries-Peak-Rate

For Example:

[Statistics]
Server/Authentication-Threads
Server/Accounting-Threads Server/Total-Threads Server/Max-Acct-Threads
Server/Max-Auth-Threads Server/Max-Total-Threads
Server/High-Auth-Threads Server/High-Acct-Threads
Server/High-Total-Threads
Server/High-Acct-Threads-Since-Reset Server/High-Auth-Threads-Since-Reset
Server/High-Total-Threads-Since-Reset
Authentication/Accept
Authentication/Reject
Authentication/Silent-Discard
Authentication/Total-Transactions
Authentication/Dropped-Packet Authentication/Invalid-Request Authentication/Failed-Authentication
Authentication/Failed-On-Check-List
Authentication/Insufficient-Resources
Authentication/Proxy-Failure
Authentication/Rejected-By-Proxy
Authentication/Transactions-Retried Authentication/Total-Retry-Packets
Accounting/Start
Accounting/Stop
Accounting/Interim
Accounting/On
Accounting/Off
Accounting/Total-Transactions
Accounting/Dropped-Packet Accounting/Invalid-Request

Accounting/Invalid-Client
Accounting/Invalid-Shared-Secret
Accounting/Insufficient-Resources
Accounting/Proxy-Failure
Accounting/Transactions-Retried Accounting/Total-Retry-Packets
Proxy/Authentication
Proxy/Accounting
Proxy/Total-Transactions
Proxy/Timed-Out
Proxy/Invalid-Response
Proxy/Invalid-Shared-Secret
Proxy/Insufficient-Resources
Proxy/Transactions-Retried Proxy/Total-Retry-Packets
Rate/Auth-Request-Current-Rate Rate/Auth-Request-Average-Rate
Rate/Auth-Request-Peak-Rate Rate/Auth-Accept-Current-Rate Rate/Auth-Accept-Average-Rate
Rate/Auth-Accept-Peak-Rate Rate/Auth-Reject-Current-Rate Rate/Auth-Reject-Average-Rate
Rate/Auth-Reject-Peak-Rate Rate/Acct-Start-Current-Rate Rate/Acct-Start-Average-Rate
Rate/Acct-Start-Peak-Rate
Rate/Acct-Stop-Current-Rate Rate/Acct-Stop-Average-Rate Rate/Acct-Stop-Peak-Rate
Rate/Proxy-Auth-Request-Current-Rate Rate/Proxy-Auth-Request-Average-Rate Rate/Proxy-Auth-Request-Peak-Rate Rate/Proxy-Acct-Request-Current-Rate
Rate/Proxy-Acct-Request-Average-Rate Rate/Proxy-Acct-Request-Peak-Rate Rate/Proxy-Fail-

Timeout-Current-Rate Rate/Proxy-Fail-Timeout-Average-Rate Rate/Proxy-Fail-Timeout-Peak-Rate Rate/Proxy-Fail-Badresp-Current-Rate Rate/Proxy-Fail-Badresp-Average-Rate Rate/Proxy-Fail-Badresp-Peak-Rate Rate/Proxy-Fail-Badsecret-Current-Rate Rate/Proxy-Fail-Badsecret-Average-Rate Rate/Proxy-Fail-Badsecret-Peak-Rate Rate/Proxy-Fail-Missingresr-Current-Rate Rate/Proxy-Fail-Missingresr-Average-Rate Rate/Proxy-Fail-Missingresr-Peak-Rate Rate/Proxy-Retries-Current-Rate Rate/Proxy-Retries-Average-Rate Rate/Proxy-Retries-Peak-Rate Rate/Proxy-Auth-Rej-Proxy-Current-Rate Rate/Proxy-Auth-Rej-Proxy-Average-Rate Rate/Proxy-Auth-Rej-Proxy-Peak-Rate Rate/Proxy-Acct-Fail-Proxy-Current-Rate Rate/Proxy-Acct-Fail-Proxy-Average-Rate Rate/Proxy-Acct-Fail-Proxy-Peak-Rate Rate/Proxy-Auth-Rej-Proxy-Error-Current-Rate Rate/Proxy-Auth-Rej-Proxy-Error-Average-Rate Rate/Proxy-Auth-Rej-Proxy-Error-Peak-Rate.

tacplus.ini File

Used by: GEE, EE
Not Used By: —

The tacplus.ini initialization file provides the configuration information that enables the Steel-Belted Radius server to communicate with a TACACS+ server.

Note: Steel-Belted Radius does not support the use of IPv6 with TACACS+.

[ServerInfo] Section

The [ServerInfo] section of tacplus.ini (Table 24) provides information that allows Steel-Belted Radius to communicate with a TACACS+ server.

Table 24: tacplus.ini [ServerInfo] Syntax

Parameter	Function
SharedSecret	Specifies the shared secret between the TACACS+ server and Steel-Belted Radius.

TargetHost	Specifies the name or IPv4 address of the TACACS+ server.

For example:

```
[ServerInfo]
SharedSecret=123abc
TargetHost=197.43.160.
101
```

winauth.aut File

Used by: GEE, EE
Not Used By: —

The Windows Domain authentication method is configured by means of the winauth.aut file.

[Bootstrap]Section

The [Bootstrap] section (Table 25) specifies information that Steel-Belted Radius uses to load and start the Windows domain authentication module.

Table 25: winauth.aut [Bootstrap] Syntax

Parameter	Function
LibraryName	Specifies the name of the Windows Domain Authentication module. Default value is winauth. dll. Do not change this unless you are advised to do so by Pulse Secure Global Support Center.
Enable	• If set to 1, the authentication module is enabled. • If set to 0, the authentication module is disabled and does not appear in the Authentication Methods list in SBR Administrator. Default value is 1.
InitializationString	This entry is used to specify the name of the authentication method to appear in the Authentication Methods list in SBR Administrator. Default value is Windows domain authentication. The name of each authentication method must be unique. If you create additional.aut files to implement authentication against multiple databases, be sure that each InitializationString is set to a different method name.

[WindowsDomain] Section

The [WindowsDomain] section (Table 26) specifies the server's response to an expired Domain password. You can choose separate responses for Domain User and Domain Group authentication methods. Steel-Belted Radius takes the actions that you define in the [WindowsDomain] section when it receives an Expired password or User must change password at next logon status code after passing a username/password pair to a Windows Domain for authentication.

```
[WindowsDomain]
AllowExpiredPasswordsForUsers =
no
AllowExpiredPasswordsForGroups
= no AllowMachineLogin = yes
PrequalifyChecklist = no
ProfileForExpiredUsers = <place_name_of_Profile_here>
```

ProfileForExpiredUsersInGroups = <place_name_of_Profile_here>
RetryFailedAuthentications = no

Table 26: winauth.aut [WindowsDomain] Syntax

Parameter	Function
AllowExpiredPasswordsForUsers	A value of yes means that when the incoming username/password pair can be validated but the password has expired (under Domain User authentication), the server responds with an Access-Accept. A value of no means the server responds with an Access-Reject. If set to yes and you do not provide a ProfileForExpiredUsers value, the Access-Accept response contains the return list from the Domain User entry that matches the incoming username. This option is recommended for Domain Users. If set to yes and you provide a ProfileForExpiredUsers, the return list from that profile is used. Default value is no.
AllowExpiredPasswordsForGroups	If set to yes, the server responds with an Access-Accept when the incoming user-name/password pair can be validated but the password has expired (under Domain Group authentication). If set to no, the server responds with an Access-Reject. If set to yes and you provide a ProfileForExpiredGroups, the return list from that profile is used. This option is strongly recommended if you allow expired passwords for Domain Groups. If set to yes and you do not provide a ProfileForExpiredGroups value, the Access-Accept response contains the return list from the first Domain Group entry (alphabetically) in the server database. Default value is no.
AllowMachineLogin	Machine authentication is useful for sites that want a host on the network even when a user is not logged in. When machine authentication is enabled, a host's Windows domain credentials (certificate or domain password) are used to connect the host to the network. If set to yes, machine authentication is enabled. If set to no, machine authentication is disabled. Default value is yes. Note: You must enter 1 for the First-Handle-Via-Auto-EAP setting in the [Windows Domain User] and [Windows Domain Group] sections of eap.ini if you want to use machine authentication. For more information, see "eap.ini File".
ProfileForExpiredUsers	Names a profile entry in the Steel-Belted Radius database. This entry provides the return list for responses for all users who are accepted by the server under Domain User "expired password" conditions.
ProfileForExpiredUsersInGroups	Names a profile entry in the Steel-Belted Radius database. This entry provides the return list for responses for all users who are accepted by the server under Domain Group "expired password" conditions. This option is strongly recommended for Domain Groups.
PrequalifyCheckList	If set to yes, Steel-Belted Radius performs checklist processing on each domain ob- ject in the database before trying to authenticate a user request. If checklist processing fails, the object is skipped. If set to no, prequalification checklist processing is not performed. Default value is no.
RetryFailedAuthentications	If set to yes, Steel-Belted Radius attempts to authenticate a user after an authentica- tion failure. If set to no, Steel-Belted Radius does not attempt to reauthenticate a user after an authentication failure. Default value is no.
MatchDomainWithoutExtension	If set to yes, Steel-Belted Radius attempts to authenticate a domain user with User Principal Name. If set to no, Steel-Belted Radius does not attempt to authenticate a domain user with User Principal Name

Parameter	Function
	Note: This option will not be available as default. User needs to add this option manually to use UPN.

Chapter 3

Operations Files

This chapter describes the usage and settings for files used in Steel-Belted Radius operations and administration.

access.ini File

Used by: GEE
Not used by: EE

The access.ini file maps operating system user or group account names to levels of administrative privilege. The user account name and password used by an administrator when interacting with the Steel-Belted Radius server is granted access privileges according to the settings in this file.

[Settings] Section

The [Settings] section of access.ini contains overall configuration parameters; do not edit this section.

[Users] and [Groups] Sections

The syntax for the [Users] and [Groups] sections of the access.ini file (Table 27) is as follows:

```
[Users]
UserName = AccessLevel
_system.localhost          =
SnmpAgent  M
[Groups]
```

GroupName = AccessLevel
GroupName = AccessLevel
.
.

Note: If you use SNMP to monitor your Steel-Belted Radius server, the [Users] section of your access.ini file must contain the following entry:

_system.localhost = SnmpAgent
If you are not using SNMP, you should comment out or delete the
_system.localhost = SnmpAgent entry as a security precaution.

Table 27: access.ini Syntax

Parameter	Function
UserName GroupName	Each UserName or GroupName is the name of an authorized administrator account on the server. Depending on your platform, UserName and GroupName refer to a Windows domain user/group or a Linux /etc/passwd user/group. You must list user accounts in the [Users] section and group accounts in the [Groups] section. You should list groups in priority order; rights are granted based on the first group found of which the user is a member.
AccessLevel	The AccessLevel in each access.ini entry is the access level that you want to assign to that account. Each AccessLevel string must match the name of an [AccessLevel] section in admin.ini. You can define as many [AccessLevel] sections as you require. Once an [AccessLevel] section is defined in admin.ini, you can use access.ini to assign the access privileges associated with that level to users and group accounts.

A special access level called SuperAdmin grants read/write access to all types of administrative data. This access level is always defined, and can be assigned to a user or group account in access.ini without appearing in admin. ini. admin.ini File.

admin.ini File

Used by: GEE
Not used by: EE

The admin.ini file maps administrative access levels to sets of access rights. These access levels are enforced for administrators connecting to Steel-Belted Radius by means of the SBR Administrator or the LDAP configuration interface (LCI). Each [AccessLevel] section in the admin.ini file corresponds to an AccessLevel name entered in the access.ini file. You can create as many [AccessLevel] sections in the admin.ini file as you require.

Access rights are defined according to the categories of administrative data that an account is allowed to read and/or write. These data categories correspond to SBR Administrator panels and to objects directly under o=radius in the LDAP configuration schema.

Note: If you omit a keyword, access to that data category is specifically denied for all information and windows that correspond to that keyword. Misspelled keywords are considered omitted.

[AccessLevel] Section

The syntax for each [AccessLevel] section (Table 28) defined in admin.ini is as follows:

```
[AccessLevel]
Access = value
Certificates = value
CCMPublish = value
CCMServerList= value
Configuration = value
CurrentUsers = value
ImportExport = value IP-Pools =value
IPX-Pools = value
License = value
Profiles = value Proxy= value
RAS-Clients= value Report = value
RuleSets= value
Statistics = value
Tunnels = value Users= value
```

Table 28: admin.ini Syntax

Parameter	Function
AccessLevel	Specifies the name of the access level. The value used here must be identical to the value used in the access.ini file.
Access	Specifies whether administrators with this access level can read or write (update) administrative access data, which is controlled by the Administrators panel. Valid values are: • r – Read-only access • w – Write-only access • rw – Read/write access **Note**: When an administrator requests access, Steel-Belted Radius checks entries in the Administrators panel in SBR Administrator before checking the access.ini and admin.ini files. If an applicable administrative account exists in the Administrators panel, the user is given full access to the Steel-Belted Radius database, regardless of the configuration of the access.ini and admin.ini files.

Parameter	Function
Certificates	Specifies whether administrators with this access level can modify trusted root and server certificate information through the SBR Administrator. Valid values are: • r – Read-only access • w – Write-only access • rw – Read/write access
CCMPublish	Specifies whether administrators with this access level can publish server replication (ccmpkg) information through the SBR Administrator. Valid values are: • r – Read-only access • w – Write-only access • rw – Read/write access
CCMServerList	Specifies whether administrators with this access level can read or write (update) information in the Replication panel in the SBR Administrator. Valid values are: • r – Read-only access • w – Write-only access • rw – Read/write access
Configuration	Specifies whether administrators with this access level can read or write (update) information found in the Authentication Policies panel in the SBR Administrator. Valid values are: • r – Read-only access • w – Write-only access • rw – Read/write access
CurrentUsers	Specifies whether administrators with this access level can read or write (update) the Current Sessions Table, which can be displayed in the Reports panel of SBR Administrator. Write access allows the administrator to delete entries from the Current Sessions Table. Valid values are: • r – Read-only access • w – Write-only access • rw – Read/write access
ImportExport	Controls whether the Import and/or Export menu items are enabled in the SBR Administrator. • Read access allows file export. • Write access allows file import. Valid values are: • r – Read-only access (allows export but not import) • w – Write-only access (allows import but not export) • rw – Read/write access (allows import and export) Data categories without read access are disabled. If a user tries to export categories of data without having sufficient access rights, categories for which the user does not have read access are omitted from the export operation. Similarly, if a user tries to import categories of data without having sufficient access rights, categories for which the user does not have write access are omitted from the import operation. Note: Import and Export are subject to the particular rights that the user has to each type of item, such as Users or Tunnels.
IP-Pools	Specifies whether administrators with this access level can read or write (update) IP address pool data. Valid values are: • r – Read-only access • w – Write-only access • rw – Read/write access

Parameter	Function
IPX-Pools	Specifies whether administrators with this access level can read or write (update) IPX address pool data. Valid values are: • r – Read-only access • w – Write-only access • rw – Read/write access
License	Specifies whether administrators with this access level can add a new license. Valid values are: • w – Write-only access • rw – Read/write access
ImportExport	Controls whether the Import and/or Export menu items are enabled in the SBR Administrator. • Read access allows file export. • Write access allows file import. Valid values are: • r – Read-only access (allows export but not import) • w – Write-only access (allows import but not export) • rw – Read/write access (allows import and export) Data categories without read access are disabled. If a user tries to export categories of data without having sufficient access rights, categories for which the user does not have read access are omitted from the export operation. Similarly, if a user tries to import categories of data without having sufficient access rights, categories for which the user does not have write access are omitted from the import operation. Note: Import and Export are subject to the particular rights that the user has to each type of item, such as Users or Tunnels.
IP-Pools	Specifies whether administrators with this access level can read or write (update) IP address pool data. Valid values are: • r – Read-only access • w – Write-only access • rw – Read/write access
IPX-Pools	Specifies whether administrators with this access level can read or write (update) IPX address pool data. Valid values are: • r – Read-only access • w – Write-only access • rw – Read/write access
License	Specifies whether administrators with this access level can add a new license. Valid values are: • w – Write-only access • rw – Read/write access
Profiles	Specifies whether administrators with this access level can read or write (update) profile data. Valid values are: • r – Read-only access • w – Write-only access • rw – Read/write access
Proxy	Specifies whether administrators with this access level can read or write (update) proxy target data. Valid values are: • r – Read-only access • w – Write-only access • rw – Read/write access

Parameter	Function
RAS-Clients	Specifies whether administrators with this access level can read or write (update) RADIUS client data. Valid values are: • r – Read-only access • w – Write-only access • rw – Read/write access
Report	Specifies whether administrators with this access level can read or write (update) report data. Valid values are: • r – Read-only access • w – Write-only access • rw – Read/write access
Rulesets	Specifies whether certificates are replicated within a realm. Valid values are: • r – Read-only access • w – Write-only access • rw – Read/write access
Statistics	Specifies whether administrators can read Authentication, Accounting, and Proxy statistics generated by the server. Write access is not applicable. Valid values are: • r – Read-only access
Tunnels	Specifies whether administrators with this access level can read or write (update) RADIUS tunnel data. Valid values are: • r – Read-only access • w – Write-only access • rw – Read/write access
Users	Specifies whether administrators with this access level can read or write(update) user data. Valid values are: • r – Read-only access • w – Write-only access • rw – Read/write access Note: You must set the Users parameter to rw (read-write) for a user or group if you want the user or group to be able to import user information into Steel-Belted Radius.

[SNMPAgent] Section (GEE only)

If you use SNMP to monitor your Steel-Belted Radius server, the [SNMPAgent] section of admin.ini file must include the following section to give Read access to the SNMP agent.

```
[SnmpAgent]
RAS-Clients=r
Users=r
Profiles=r
Proxy=r
```

Tunnel
s=r
IP-
Pool
s=r
IPX-
Pool
s=r
Acce
ss=r
Configuration=r
Statistics=r
CurrentUsers=r
Report=r
ImportExport=r
License=r

bounce.ini File (Windows only)

Used by: GEE
Not used by: EE

The bounce.ini configuration file enables and configures the Steel-Belted Radius auto-restart feature. This feature causes Steel-Belted Radius to restart itself automatically whenever it experiences a shutdown.

If you enable the auto-restart feature, it results in the loading of two copies of the server executable, radius.exe. After the parent executable is run, the parent executable runs the child executable. The parent periodically sends a message to the child to see if it is still operating. If the child does not respond to the message within
60 seconds (a configurable time period), the parent terminates the child, waits for a configurable number of seconds to allow radius.exe to fully shut down, and then starts a new copy of the child.

Note: When auto-restart is enabled and the server is running normally, you typically see two instances of radius.exe in any tool (such as the Task Manager) that you use to monitor processes on the Windows host computer.

While auto-restart is enabled, all server startup and shutdown activity is logged to a file called bounce.log in the server directory. Other types of server activity continue to be logged to the server log file (yyyymmdd.log) in the serverdirectory.

[Settings] Section

The bounce.ini file contains one configuration section called [Settings] (Table 29).

[Settings]
Enable=1
MaxPong=2
5
MaxStartup
=60
MaxShutDow
n=60

PingInterval
=10

Note: When auto-restart is enabled and the server is running normally, you typically see two instances of radius.exe in any tool (such as the Task Manager) that you use to monitor processes on the Windows host computer.

Table 29: bounce.ini [Settings] Syntax

Parameter	Function
Enable	• If set to 1, the auto-restart feature is enabled. • If set to 0, the auto-restart feature is disabled and other settings in the bounce.ini file are ignored. Default value is 0.
MaxPong	Specifies the number of seconds that the parent waits for a response message from the child, before it decides the child is no longer operating and attempts to restart it. Default value is 25 seconds.
MaxShutdown	Specifies the number of seconds that the parent allows for normal shutdown of the child. If the child does not terminate within that time, the parent terminates the child. Default value is 60 seconds
MaxStartup	Specifies the number of seconds that the parent allows for starting up the child. If the child does not send a message within that time, the parent decides the startup was not successful and exits. Default value is 60 seconds.
PingInterval	Specifies the number of seconds between each message sent by the parent to the child to check whether it is running. Default value is 10 seconds.

ccagw.ini File

Used by: GEE, EE
Not used by: —

The ccagw.ini file contains information about 3COM CCA gateways. You must configure a [gateway] section for each 3COM CCA gateway to enable the return of required CCA tunnel attributes.

[gateway] section

Each [gateway] section of the ccagw.ini file (Table 30) contains information about a specific 3COM CCA gateway.

```
[Jupiter-Gateway]
Address =
200.47.98.142
TunnelRefresh = 3600
Description = Jersey City facility, East Coast
subscribers Secret=HollandTunnel
```

Table 30: ccagw.ini [gateway] Syntax

Parameter	Function
Address	Specifies the IPv4 address of the gateway. Note: You cannot use IPv6 addresses with ccagw.ini.
TunnelRefresh	Specifies the number of seconds before the tunnel refreshes. Default value is 0.
Description	Specifies a text string describing the gateway.
Secret	Specifies the shared secret used to authenticate communication between Steel-Belted Radius and the gateway device.

eval.ini File

Used by: GEE, EE
Not used by:

The eval.ini file contains temporary license key information for evaluation installations of Steel-Belted Radius. The eval.ini file does not contain any configurable settings.

events.ini File

Used by: GEE, EE
Not used by: —

The events.ini configuration file controls dilutions and thresholds for Steel-Belted Radius events that are used to communicate failures, warnings, and other information. Events are handled by the Windows Events Viewer.

Appendix D, "Windows Events," summarizes common event values.

Note: that only some of these events support thresholds or dilution.

[EventDilutions] Section

The [EventDilutions] section of events.ini specifies how many events must occur before Steel-Belted Radius generates an event report. This feature lets you "dilute" the rate at which frequently occurring events are logged.

Syntax is as follows:

```
[EventDilutio
ns]
EventName=DilutionCount
```

where EventName identifies a Steel-Belted Radius event and DilutionCount specifies how many times this event must occur before it is recorded in the Windows event log or to the SNMP manager program.

Example

The following example specifies that a Steel-Belted Radius server configured to authenticate against a SQL database reports every fifth SQLConnectFailure (warning event number 5008) error:

```
[EventDilutions]
; 5008 - nnnn attempts to connect to SQL server failed.
SQLConnectFailure=5
```

If an SQL error condition prevents the server from connecting to the database, Steel-Belted Radius retries the connection (and reports these attempts in the RADIUS log file (yyyymmdd.log). Steel-Belted Radius does not trigger warning event 5008 until the fifth connection attempt fails.

[Suppress] Section

The [Suppress] section of events.ini lets you suppress Steel-Belted Radius events. An event whose ID number (Windows) or trap number (Linux) appears in this section is not reported when the applicable informational, warning, or error condition occurs.

Example

The following settings suppress events relating to verification server timeouts (5013) or failures (5014).

```
[Suppress]
5013
5014
```

[Thresholds] Section

The [Thresholds] section of events.ini (Table 31) lets you specify thresholds that trigger an event report. Thresholds often come in pairs, where a warning event is generated when a resource becomes scarce (low threshold is crossed), and an information event is generated when the resource becomes available (high threshold is crossed).

The [Thresholds] section lets you tune Steel-Belted Radius event generation for items such as system memory, thread count, and file system space, and can differ for each computer depending on resources, configuration, and other applications.

Table 31: events.ini [Thresholds] Syntax

Parameter	Function
FileSystemFreeKBWarningIssue	When available system disk space falls to the specified value, issue the warning event RADMSG_FILE_SYSTEM_LOW or funkSbrTrapLowFSSpace (5007). Default value is 4096 KB (4MB).
FileSystemFreeKBWarningClear	When the number of kilobytes of available system disk space reaches the specified value, issue the informational event RADMSG_FILE_SYSTEM_NORMAL or funkSbrTrapFSNormal (103). Default value is 8092 KB (8MB).
ReserveMemoryKB	Reserve this amount of memory (in kilobytes) at system startup for cases of overload. If a memory allocation failure occurs, Steel-Belted Radius frees the reserved memory and reports the event. Default value is 2048 KB (2MB).
PoolPctAddressAvailWarningIssue	When the number of available addresses in any IP address pool drops below the specified percentage, issue a funkSbrTrapIPAddrPoolLow warning. Defaultvalue is 20 percent.
PoolPctAddressAvailWarningClear	When the number of available addresses in any IP address pool rises above the specified percentage, issue an informational message. Default value is 40 percent.

Example

This following example would produce a warning event (5001) when the number of available accounting or authentication threads falls below 10 percent, and an informational event (102) is issued when it rises above 20 percent.

[Thresholds]
ThreadAvailWarningIssue=10
ThreadAvailWarningClear=20

radConfigServer.ini

Used by: GEE, EE

Not used by: —

The radConfigServer.ini initialization file is introduced as part of SBR 6.2.5. This file configures backup, restore, publish and CCM functionalities of SBR database. However currently only backup configurations are present. In future, it could be used for other configurations.

[ScheduleBackup] Section

Used by: GEE, EE

Not used by: —

This section is used to schedule SBR DB backup periodically thereby automating the process.

[ScheduleBackup]
ScheduleInHours = 12
ScheduleDaily = yes
ScheduleWeekly = yes
ScheduleMonthly = yes

All the parameters in this section are disabled by default.

Note: Only one parameter can be enabled at a time. If more than one parameter is enabled, then the first parameter enabled in the order given above will be taken.

Example 1:

[ScheduleBackup]
ScheduleInHours = 12
ScheduleDaily = yes
;ScheduleWeekly = yes
;ScheduleMonthly = yes

In the above example both ScheduleInHours and ScheduleDaily are enabled. ScheduleInHours will be assumed as enabled. Hence, the backup will be scheduled for 12 hours. The order is hours, daily, weekly, monthly even if you shuffle the parameters.

The time of restart after setting the configurations will be taken as the start time of the schedule.

Table 32: radConfigServer.ini [ScheduleBackup] Syntax

Parameter	Function
ScheduleInHours	Schedules DB backup on hourly basis Default value is 12 hours. The range should be between 1 and 1000.

Parameter	Function
ScheduleDaily	Schedules DB backup on daily basis (24 hours) Default value is yes. This will be enabled when the value is given as yes/YES
ScheduleWeekly	Schedules DB backup on weekly basis Default value is yes. This will be enabled when the value is given as yes/YES
ScheduleMonthly	Schedules DB backup on monthly basis Default value is yes. This will be enabled when the value is given as yes/YES

Note: An xml file loaded during the startup of radConfigServer module called "sbr_ccm.xml" has few more configuration parameters. These are both applicable for manual and Auto backup.

sbr_ccm.xml Location:

"<SBR_INSTALLED_PATH>/Service" (Windows), "<SBR_INSTALLED_PATH>/radius (Linux)

1. Backup directory can be configured
   backup_filesystem_path" (line no: 116)

2. Maximum backups kept at a time
   "max_backups" (line no: 117)

Figure 4: Configurable parameters in sbr_ccm.xml

```
<component id="ConfigPublicationProvider.component_class">
    <object program_id = "radConfigServer.RadConfigPublicationProvider">
        <init
            backup_ruleset         = "BACKUP"
            temp_filesystem_path   = "temp"
            backup_filesystem_path = "backups"
            max_backups            = "10"
            resource_provider_uri  = "/CCM/publish/"
            database_path          = "/CCM/serverlist/"
        />
    </object>
```

radius.ini File

Used by: GEE, EE*
Not used by: —

The radius.ini initialization file is the main configuration file that determines the operation of Steel-Belted Radius. It contains information that controls a variety of Steel-Belted Radius functions and operations.

[Addresses] Section

Used by: GEE, EE*
Not used by: —

By default, the Steel-Belted Radius server tries to autoconfigure all IPv4 addresses that are reported by name services for the primary host name of the server on which Steel-Belted Radius is running, so that it can listen for incoming RADIUS packets on all available network interfaces. If IPv6 is enabled, Steel-Belted Radius autoconfigures its IPv6 addresses and then listens on all interfaces using IPv6 addresses.

You should explicitly configure the IP addresses that you want Steel-Belted Radius to use in the [Addresses]

section of radius.ini if Steel-Belted Radius is running on a multi-homed (more than one network interface) server and if any of the following statements apply to your network:

- One or more network interfaces on the server are connected to networks that should not carry RADIUS traffic.
- The server has more than one host name, and IP addresses exist for names other than the primary host name.
- The server has private IP addresses that are not published by name services.

Specifying IPv4 or IPv6 addresses causes the server to listen on only those addresses and ignore all other addresses.

Specifying AutoConfigureIPv4 or AutoConfigureIPv6 causes Steel-Belted Radius to attempt to discover and configure all IPv4 or IPv6 addresses that belong to the local host automatically.

Example 1

The following example configures Steel-Belted Radius to listen for RADIUS authentication and accounting requests on the IPv4 address 192.168.12.35 and on all local IPv6 interfaces. Note that IPv6 functionality must be enabled (by setting Enable to 1 in the [IPv6] section of radius.ini) before IPv6 addresses can be used.

```
[Addresses]
192.168.12.35
AutoConfigureIPv6
```

To route all of your proxy traffic through a single interface, set the value for ProxySource in the [Configuration] section of radius.ini to the appropriate IP address or addresses, which must be listed in the [Addresses] section.

Example 2

The following example routes all proxy traffic through the interface at 192.10.20.30:

```
[Addr
esses
]
192.1
0.20.3
0
192.10.20.31

[Configuration] ProxySource
= 192.10.20.30
```

GEE: The ProxySource setting in the [Configuration] section of radius.ini disables per-realm control of proxy outbound interfaces. If ProxySource is not set, sockets are opened and bound for each interface on the server. To route different proxy realms through specific interfaces using the proxy.ini file, refer to "[Interfaces] Section".

[AuditLog] Section

Used by: GEE, EE
Not used by: —

The [AuditLog] section (Table 32) specifies whether Steel-Belted Radius maintains an audit log file (yyyymmdd. auditlog) to record administrator activities and CCM events. Audit log records are stored in XML format.
Administrator activities include the following:

- Logging in and out by Steel-Belted Radius administrators
- Creating, modifying, and deleting Steel-Belted Radius objects (RADIUS clients, users, profiles, proxy targets, proxy realms, tunnels, administrators, authentication policies, or CCM nodes)
- Importing files

CCM events include publication, notification, and download of CCM files.

Note: The audit log does not track changes made through the LDAP configuration interface (LCI).

```
[AuditLog]
;Enable = 0
;LogfilePermissions=owner:groupmode
;DaysToKeep = 30
```

Table 33: radius.ini [AuditLog] Syntax

Parameter	Function
Enable	• If set to 0, audit logging is disabled. • If set to 1, audit logging is enabled. Default value is 0.
LogfilePermissions (Linux only)	Specifies the owner and access permission setting for the audit log (yyyymmdd.auditlog) file. Enter a value for the LogFilePermissions setting in owner:group permissions format, where: • owner specifies the owner of the file in text or numeric format. • group specifies the group setting for the file in text or numeric format. • permissions specifies what privileges can be exercised by Owner/Group/Other with respect to the file in text or numeric format. For example, ralphw:1007 rw-r- - - - - specifies that the file owner (ralphw) can read and edit the audit log file, members of group 1007 can read (but not edit) the audit log file, and other users cannot access the audit log file.
DaysToKeep	Specifies the number of days the Steel-Belted Radius server should retain each authentication acceptance report. Default value is 30 days.

[AuthRejectLog] Section

Used by: GEE
Not used by: EE

You configure the [AuthRejectLog] section of radius.ini (Table 33) to specify what types of authentication method rejection messages Steel-Belted Radius records in the RADIUS log file (yyyymmdd.log). You can specify that you want the server log file to record reject information generated by all authentication methods, reject information of one or more specific types, or the most relevant rejection information.

Processing an authentication request might result in multiple instances of an authentication method being given a chance to authenticate the user. If this occurs and at least one authentication method

succeeds in authenticating the user, no messages are recorded to the server log file. If this occurs and all instances fail to authenticate the user, you can specify that only the most relevant reason for the authentication failure is recorded. For example, if one method resulted in an authentication error of type InvalidCredentials and another results in an authentication error of type SystemError, only the InvalidCredentials message would be logged.

You can specify that more than one type of log message should be recorded by entering more than one filter type value for the Filter parameter.

Table 34: radius.ini [AuthRejectLog] Syntax

Parameter	Function
Enable	• If set to 0, authentication reject details are not recorded in the server log file. • If set to 1, authentication reject details of the specified type(s) are recorded in the server log file. Default value is 0.
Filter	Specifies the types of authentication reject messages to be recorded: • **All** – Record authentication rejection details from all authentication methods. • **MostRelevant** – When multiple authentication methods are tried and all fail, record the most relevant error messages (the messages with the greatest severity). If two messages have the same severity, both are listed. The following values are listed in order of greatest to least relevance: • PostProcessRejection – User was authenticated successfully but postprocessing caused rejection. • InvalidCredentialsOrUser – User was not authenticated because user was not found or credentials were invalid. • InvalidCredentials – User was not authenticated because user was known but the password or certificate was not correct. • UnsupportedCredentialType – User was not authenticated because the credentials presented were of the wrong type. • UserNotFound – User was not authenticated because user could not be found in the authentication database. • AccessError – Authentication failed because a database or remote server was inaccessible. • InvalidRequest – User was not authenticated because the request appeared to be malformed. • BlacklistedUser – User was not authenticated because user is blacklisted. • SystemError – User was not authenticated because of a system error such as a resource allocation error.

The following example would cause authentication reject details from all authentication methods to be recorded to the server log file.

```
[AuthRej
ectLog]
Enable =
1 Filter =
All
```

The following example would cause all authentication reject details of type SystemError to be recorded.

```
[AuthRej
ectLog]
Enable =
1
```

Filter= SystemError

The following example would cause all authentication reject details of type SystemError, BlacklistedUser, or UserNotFound to be recorded.

[AuthRej
ectLog]
Enable =
1
Filter= SystemError, BlacklistedUser, UserNotFound

[Configuration] Section

Used by: GEE, EE*
Not used by: —

The [Configuration] section of radius.ini (Table 34) contains parameters that control basic behavior of Steel- Belted Radius.

Table 35: radius.ini [Configuration] Syntax

Parameter	Function
AcctAutoStopEnable (GEE only)	The Proxy AutoStop feature forwards session termination information to downstream proxy RADIUS servers when a user session is closed, so that the resources associated with the user session can be freed. • If set to 0, the Proxy AutoStop feature is disabled. • If set to 1, the Proxy AutoStop feature is enabled. Default value is 0.
AddDestIPAddressAttrToRequest	• If set to 0, Steel-Belted Radius does not add destination address information to RADIUS requests. • If set to 1, Steel-Belted Radius adds a Funk-Dest-IP-Address attribute identifying the IP address to which the RADIUS request was sent to the attributes in the packet. All processing that could be performed on an attribute included in the request packet, such as checklist processing, can be performed on this attribute. Default value is 0. GEE: If you enable this attribute, the attribute is visible to the proxy module. If your environment proxies requests, you might want to configure Steel-Belted Radius to strip the attribute from the request before forwarding the request to a downstream server
AddDestUDPPortAttrToRequest	• If set to 0, Steel-Belted Radius does not add destination port information to RADIUS requests. • If set to 1, Steel-Belted Radius adds a Funk-Dest-UDP-Port attribute identifying the UDP port to which the RADIUS request was sent to the attributes in the packet. All processing that could be performed on an attribute included in the request packet, such as checklist processing, can be performed on this attribute. Default value is 0. GEE: If you enable this attribute, the attribute is visible to the proxy module. If your environment proxies requests, you might want to configure Steel-Belted Radius to strip the attribute from the request before forwarding the request to a downstream server.
AddFunkClientGroupToRequest	• If set to 0, Steel-Belted Radius does not add a Funk-Radius-Client-Group attribute to an incoming RADIUS request. • If set to 1, Steel-Belted Radius adds a Funk-Radius-Client-Group attribute to the RADIUS request. The value of the Funk-Radius-Client-Group attribute is set to the name of the client group. Default value is 0. **Note**: You should enable this option only if you configure RADIUS client groups in

Parameter	Function
	SBR Administrator. For more information on RADIUS client groups, refer to the Steel-Belted Radius Administration Guide.
AddFunkLocationGroupIdToRequest (GEE only)	• If set to 0, Steel-Belted Radius does not add a Funk-Location-Group-Id attribute to an incoming RADIUS request. • If set to 1, Steel-Belted Radius adds a Funk-Location-Group-Id attribute to an incoming RADIUS request if the request comes from a client in a configured location group. The value of the Funk-Location-Group-Id attribute is set to the name of the location group, which can be used for SQL, LDAP, and checklist processing. Default value is 0. **Note:** The Funk-Location-Group-id attribute is case-sensitive
AddSourceIPAddressAttrToRequest	• If set to 0, Steel-Belted Radius does not add source address information to RADIUS requests. • If set to 1, Steel-Belted Radius adds a Funk-Source-IP-Address attribute identifying the IP address from which the RADIUS request was received to the attributes in the packet. All processing that could be performed on an attribute included in the request packet, such as checklist processing, can be performed on this attribute. Default value is 0. GEE: If you enable this attribute, the attribute is visible to the proxy module. If your environment proxies requests, you might want to configure Steel-Belted Radius to strip the attribute from the request before forwarding the request to a downstream server.
Apply-Login-Limits	• If set to yes, the maximum number of concurrent connections for each user is enforced, and connection attempts above the limit are rejected. • If set to no, connections above the limit are allowed, but an event is noted in the server log file. Default value is yes.
AttributeEdit (GEE only)	• If set to 1, the attribute editing feature for proxy realms is enabled. • If set to 0, the feature is disabled. Default value is 1.
AuthenticateOnly	• If set to 1, no response attributes are included in the response packet to an AuthenticateOnly (Service-Type 8) request. • If set to 0, the normal response attributes are included in the response. Default value is 1.
AutoPasswor ds (GEE only)	If set to yes, support for SHA and UNIXcrypt passwords for authentication against the native database are enabled. Default value is no (disabled).
CheckMessageAuthenticator	Specifies whether validation of Message-Authenticator occurs on receipt of an Access-Request from a network access device or on receipt of an Access-Accept, Access-Reject, or Access-Challenge from a proxy (extended proxy only). • If set to 0, the validation of received Message-Authenticator attributes is disabled. • If set to 1, the validation of received Message-Authenticator attributes is enabled. Default value is 0. **Note**: Validation does not occur for ordinary proxy.
ClassAttributeStyle	• If set to 1, Steel-Belted Radius uses unencrypted Class attributes with multiple ASCII keys in Access-Accept packets. • If set to 2, Steel-Belted Radius uses enhanced/encrypted Class attributes in Access-Accept packets. Default value is 2.

Parameter	Function
	Note: The ClassAttributeStyle parameter must be set to a value of 2 before you can use attribute embedding. For information on attribute embedding, see "[Debug] Section".
DisableSecondaryMakeModelSelection	If set to 1, Steel-Belted Radius looks up the network access device entry by using the source address of the request and sets the make/model according to the information specified for the client. If set to 0, Steel-Belted Radius: 1. Looks up the network access device entry by using the source address of the request and sets the make/model according to the information specified for the client. 2. Uses the NAS-IP-Address attribute (if present) to look up the network access device entry. If the IP address is found, override the make/model information identified in Step 1. 3. Uses the NAS-Identifier attribute (if present) to look up the network access device by name. If the name is found, override the make/model information defined in Step 1 or Step 2. Default value is 0.
EnableEricssonViGHTTPDigestSupport	• If set to 1, the Ericsson ViG version of HTTP Digest Access authentication is enabled. • If set to 0, the Ericsson ViG version of HTTP Digest Access authentication is disabled. When the Ericsson ViG version of HTTP Digest Access authentication is enabled, Steel-Belted Radius looks for the ViG VSAs when it parses incoming packets, and, if it finds them, converts them to AVPs compatible with the current HTTP Digest Access authentication. Default value is 0. **Note:** This setting is ignored if the EnableHTTPDigestSupport setting is set to 0 (disabled).
EnableHTTPDigestSupport	• If set to 1, HTTP Digest Access authentication is enabled. • If set to 0, HTTP Digest Access authentication is enabled. When HTTP Digest Access authentication is enabled, Steel-Belted Radius interprets the inclusion of certain attribute-value pairs in an Access-Request message as a request to use HTTP Digest Access authentication. Default value is 0.
EnhancedDiagnosticLogging	• If set to no, standard diagnostic logging messages are written to the RADIUS log file when the log level is set to 0. • If set to yes, messages relating to proxy retries, proxy timeouts, and LDAP timeouts, as well as standard diagnostic logging messages, are written to the RADIUS log file (yyyymmdd.log) when the log level is set to 0. Default value is no.
ExtendedProxy (GEE only)	• If set to 1, you can set up realms for proxy RADIUS or directed authentication/ac-counting. • If set to 0, Steel-Belted Radius can proxy-forward to specific servers (identified using Proxy entries in the Administrator program), but proxy realms and directed realms are disabled. If the ExtendedProxy setting is not present in the [Configuration] section, realms are disabled by default. Default value is 1.
FramedIPAddressHint	If set to yes, the attribute Framed-IP-Address is treated as a hint. If this attribute appears in the Access-Request and the user's return list is configured to allocate Framed-IP-Address from a pool, the IP address in the Access-Request is returned instead of a newly-allocated IP address. Default value is no.
LogAccept	If set to 1, specifies that messages associated with Accepts that meet the current LogLevel should be recorded in the server log file.

Parameter	Function
	If set to 0, messages associated with Accepts are ignored. Default value is 1. GEE: The LogReject setting is re-read whenever the server receives a HUP signal. GEE: The LogAccept setting is re-read whenever the server receives a HUP signal.
LogFilePermissions (Linux only)	Specifies the owner and access permission setting for the system log (yyyymmdd.log) file. Enter a value for the LogFilePermissions setting in owner:group permissions format, where: • owner specifies the owner of the file in text or numeric format. • group specifies the group setting for the file in text or numeric format. • permissions specifies what privileges can be exercised by Owner/Group/Other with respect to the file in text or numeric format. For example, ralphw:1007 rw-r- - - - - specifies that the file owner (ralphw) can read and edit the log file, members of group 1007 can read (but not edit) the log file, and other users cannot access the log file.
LogfileMaxMBytes	• If set to 0 (or if setting is absent), the server log file size is ignored and log file names are date-stamped to identify when they were opened (YYYYMMDD.log). • If set to a value in the range 1–2047, the current server log file is closed when it reaches the specified number of megabytes (1024 x 1024 bytes), and a new server log file using the date and time it was opened as its filename (YYYYMMDD_HHMM.log) is opened. Default value is 0. Note: The size of the log file is checked once per minute. The log file might exceed the size specified in LogFileMaxMBytes, since it does not roll over until the next log size check occurs.
LogHighResolutionTime	• If set to no, the timestamp for entries in the Steel-Belted Radius log file (yyyymmdd. log) are recorded as hh:mm:ss (hours:minutes:seconds). • If set to yes, the timestamp for entries in the Steel-Belted Radius log file (yyyymmdd. log) are recorded as hh:mm:ss:xxx, where xxx represents the number of elapsed milli- seconds since the ss value changed. Default value is no
LogLevel	Sets the rate at which Steel-Belted Radius writes entries to the server log file (.LOG): • 0 – Production logging level • 1 – Informational logging level • 2 – Debug logging level Default value is 0. GEE: The LogLevel setting is re-read whenever the server receives a HUP signal.
LogReject	• If set to 0, messages associated with Rejects are ignored. • If set to 1, messages associated with Rejects that meet the current LogLevel should be recorded in the server log file. Default value is 1. GEE: The LogReject setting is re-read whenever the server receives a HUP signal.
NoNullTermination	• If set to 0, RADIUS reply attributes of type string are sent with a null character at the end of the string (null terminated string). • If set to 1, RADIUS reply attributes of type string are sent without the null character at the end of the string. Entering a value of 1 for this setting is the equivalent of changing all reply attributes of type string to type stringnz.

Parameter	Function
	Default value is 0. **Note:** After you change this setting, you must delete the saved-dicts.bin file and restart the Steel-Belted Radius service.
PhantomTimeout	The maximum number of seconds that a phantom session record remains active. As soon as the corresponding accounting start packet is received, a phantom record is discarded. If a phantom record still exists at the end of its timeout period, it is discarded and all resources associated with it are released.
PrivateDir (GEE only)	Sets the destination directory on the local host where server log files are stored. Default value is the Steel-Belted Radius directory. To use a non-default location, you must move or copy the database, the dictionary files, the *.xml files, and the system directory and its contents to the new directory. After you restart the Steel-Belted Radius service, the server log will be created in the new directory. **Note**: You cannot write server log files to a mapped or shared drive. **Note**: The PrivateDir parameter was called LogDir in previous releases of Steel-Belted Radius.
ProcessRealmBeforeTunnel (GEE only)	• If set to 0, Steel-Belted Radius checks whether a request matches the criteria established for tunnels before it tests whether a request matches the criteria for proxy and directed realms. • If set to 1, Steel-Belted Radius checks whether a request matches the criteria established for proxy and directed realms before it tests whether a request matches the criteria established for tunnels. Default value is 0.
ProxyFastFail (GEE only)	Specifies the number of seconds a Steel-Belted Radius server continues to forward packets to a proxy RADIUS target that appears to be down. A value of 0 disables the feature. Default value is 300.
ProxySource	Specifies the IP address of the interface through which all outgoing proxy traffic is routed. The IP address specified for ProxySource must be listed in the [Addresses] section of radius.ini. If a ProxySource address is not specified and per-realm control of proxy interfaces is not enabled, Steel-Belted Radius uses the first interface it finds on the server.
ProxyStripRealm	• If set to 1, the proxy realm decoration is stripped before sending the request down- stream. • If set to 0, no realm name stripping is performed. Default value is 1.
SelectIPPoolNameByNasAVPs	• If set to 0, the IP address pool for a RADIUS client is based on the source IP address in the UDP packet containing the access request. • If set to 1, the IP address pool for a RADIUS client is based on the value of the NAS-IP-Address or NAS-Identifier attribute included in the access request. If the NAS-IP-Address or NAS-Identifier attribute is not present, or if a RADIUS client matching the IP address or identifier cannot be found, the IP address pool for a RADIUS client is based on the source IP address in the UDP packet containing the access request. Default value is 0.
StartupTimeout	Specifies the number of seconds Steel-Belted Radius waits for its startup sequence to finish before timing out. Default value is 360 seconds.
TraceLevel	Specifies the RADIUS packet tracing level: • 0 – No packet tracing • 1 – Parsed content of packets is logged

Parameter	Function
	• 2 – Raw content and parsed content of the packet is logged Default value is 0. Note: Packet traces are written to the server log file and can be a useful tool for troubleshooting interoperability problems.
TreatAddressPoolsAsDisjoint	• If set to 1, Steel-Belted Radius treats each IP address pool as though it operates off its own disjoint address space. This disables the normal checks to ensure that an IP address is allocated only to a single address pool. • If set to 0, a single IP address can be allocated only to a single session and from a single IP address pool. Default value is 0. Note: To track allocated resources, Steel-Belted Radius uses the Class attribute to track IP addresses. This attribute contains the IP pool name and IP address.
UseNewAttributeMerge	• If set to 1, the new profile and user attribute merging calculation is performed. • If set to 0, the older calculation technique is used. Refer to "Resolving Profile and User Attributes" in the Steel-Belted Radius Administration Guide for an explanation of new attribute merging. Default value is 1.

[CurrentSessions] Section

Used by: GEE, EE
Not used by: —

The [CurrentSessions] section of radius.ini (Table35) controls the Current Sessions Table.

```
[CurrentSessions]
;CaseSensitiveUsernameCompare = 1
```

Table 36: radius.ini [CurrentSessions] Syntax

Parameter	Function
CaseSensitiveUsernameCompare	• If set to 1, when the server searches its Current Sessions Table for sessions that have the same username, it uses case-sensitive lookups. • If set to 0, the server ignores case. Default value is 1.

[Debug] Section

Used by: GEE, EE
Not used by: —

The [Debug] section of radius.ini (Table 36) helps debug problems with Steel-Belted Radius operations by incorporating thread identifiers in log messages. Thread identifiers help you parse the diagnostic log when messages about different RADIUS requests are interleaved.

The syntax for including thread identifiers in diagnostic log messages is as follows:

```
[Debug]
Log-Thread-ID = yes
```

Table 37: radius.ini [Debug] Syntax

Parameter	Function
Log-Thread-ID	• If set to yes, thread identifiers are included in Steel-Belted Radius log messages. • If set to no, thread identifiers are omitted from Steel-Belted Radius log messages. Default value is no.

[EapSettings] Section

Used by: GEE, EE
Not used by: —

The [EapSettings] section of radius.ini (Table 37) contains two parameters: AllowTLSFallback and MinimumProtocolVersion.

Table 38: radius.ini [EapSettings] Syntax

Parameter	Function
AllowTLSFallback MinimumProtocolVersion	When AllowTLSFallback option (in radius.ini under EapSettings) is commented out, it will be assumed as 1 as default, and it will allow backward compatibility and fallback. When AllowTLSFallback option is set to 0, it checks for MinimumProtocolVersion option (in radius.ini) in which specific protocol version is specified (TLSv10, TLSv11, TLSv12). If this option is not set to correct versions or if it is commented out, then TLSv12 will be assumed as default.

Absolute path of radius.ini file in Windows: <Installed path>/Pulse Secure/Steel-Belted Radius/Service Path of radius.ini file in Linux: <Installed path>/PSsbr/radius

Note: Only when AllowTLSFallback option is set to zero, MinimumProtocolVersion option is valid. Otherwise, MinimumProtocolVersion will be ignored.

[EmbedInClass] Section

Used by: GEE, EE
Not used by: —

The [EmbedInClass] section of radius.ini (Table 38) identifies attributes that are available during authentication processing which must be made available in accounting requests. Attribute embedding allows billing information to be embedded in a Class attribute returned to Steel-Belted Radius by a network access device. When Steel- Belted Radius receives an embedded attribute, it decodes the attribute and places it in the Accounting request according to the settings specified in the classmap.ini file.

Note: The ClassAttributeStyle parameter in the [Configuration] section of radius.ini must be set to a value of 2 before you can use attribute embedding.

The syntax for embedding attributes is as follows:

```
[EmbedInClass]
responseAttribute={Clear|Encrypt}[,Remove]
```

Table 39: radius.ini [EmbedInClass] Syntax

Parameter	Function
responseAttribute	Identifies the response attribute to be embedded in the RADIUS Class attribute.
Clear	Specifies that the retrieved information is included in the Class attribute in cleartext format.
Encrypt	Specifies that the retrieved information is encrypted before it is included in the Class attribute.
Remove	Optional parameter that removes the embedded attribute from the Accept-Response packet.

[FailedAuthOriginStats] Section (Windows only)

Used by: GEE, EE
Not used by: —

The [FailedAuthOriginStats] section of radius.ini enables you to identify when a specific network access device is associated with certain Windows Performance Monitor (perfmon) counters. This helps you to identify a specific region of your network that might be having difficulties. The syntax is as follows:

```
[FailedAuthOrigi
nStats]
RADIUSclient=IDn
umber
RADIUSclient=IDn
umber
```

where RADIUSclient is the name of a network access device as defined in the RADIUS Clients window, and IDnumber is a number in the range 1 to 16. These numbers map to the following Steel-Belted Radius perfmon counters:

```
Failed
Auths - 1
Failed
Auths - 2
.
.
.
Failed Auths - 16
```

For example, if you map a RADIUS client named herman to the number 3, then the perfmon counter Failed Auths - 3 tells you the number of failed authentication requests that have originated from RADIUS client herman.

```
[FailedAuthOriginS
tats] herman=3
```

[HiddenEAPIdentity] Section

Used by: GEE, EE
Not used by:

The [HiddenEAPIdentity] section of radius.ini allows the known inner identity of EAP/TTLS and EAP/SIM protocols to be included in the Access-Accept message returned in response to an authentication request.

The syntax is as follows:

```
[HiddenEAPIdentity]
IncludeInAcceptResponse=0|1
ResponseAttribute = attributeName[, replaceAttribute]
```

Table 40: radius.ini [HiddenEAPIdentity] Syntax

Parameter	Function
IncludeInAcceptResponse	• If set to 0, inclusion of the inner identity in Access-Accept responses is disabled. • If set to 1, Steel-Belted Radius includes the inner identity in the specified attribute of an Access-Accept response. Default value is 0.
attributeName	Identifies the attribute in which to include the inner identity in an Access-Accept message. If this value is omitted, the User-Name attribute is used. The attributeName value can be any string attribute, including a VSA, that is defined in an attribute dictionary.
[,replaceAttribute]	Identifies the Access-Accept attribute that retains the original value of the attribute specified in the attributeName argument. If a replacement value is not specified, the value of the original attribute is lost.

[IPPoolSuffixes] Section

Used by: GEE
Not used by: EE

The [IPPoolSuffixes] section of radius.ini lets you define suffixes that can be used to split the IP address pools reserved for a network access device into smaller subcategories.

The syntax is as follows:

```
[IPPoolSuff
ixes]
Suffix1
Suffix2
```

For example, to create three categories that append -Bronze, -Silver, and -Gold to IP Address Pool names, this section would be defined as follows:

```
[IPPoolSuffixes]
-Bronze
-Silver
-Gold
```

[IPv6] Section

Used by: GEE, EE
Not used by: —

[IPv6]

Enable = 0
DynamicNameResolution = 2
IPv6LinkLocalUnicastScopeId = 0
IPv6SiteLocalUnicastScopeId = 0

The [IPv6] section of radius.ini (Table 41) controls IPv6 network transport features.

Table 41: radius.ini [IPv6] Syntax

Parameter	Function
Enable	Determines whether IPv6 networking is enabled in Steel-Belted Radius. • If set to 0, IPv6 networking is disabled, and other values in the IPv6 section of radius.ini are ignored. • If set to 1, IPv6 networking is enabled. Default value is 0. **Note**: IPv4 networking is always enabled in Steel-Belted Radius.
DynamicNameResolution	Determines whether the Steel-Belted Radius server tries to use IPv6 name services (DNSv6) to resolve host names. • 0 – Do not use IPv6 name services. IPv4 name services are not affected by this setting. • 1 – Use only IPv6 name services. IPv4 name services are disabled by this setting. • 2 – Use IPv6 name services first; use IPv4 name services in case of failure. Default value is 2.
IPv6LinkLocalUnicastScopeId	Specifies an interface name (such as hme0) or index (4) for Linux hosts. Specifies an interface index (4) for Windows hosts. If set to 0, Steel-Belted Radius does not use link local addresses. Default value is 0.
IPv6SiteLocalUnicastScopeId	• Linux: Specifies an interface name (such as hme0) or index (4). • Windows: Specifies an interface index (4). If set to 0, Steel-Belted Radius selects the site local scope ID automatically. Default value is 0.

[LDAP] Section

Used by: GEE, EE
Not used by: —

The [LDAP] section of radius.ini (Table 41) sets the TCP port number that you want to use for communication between Steel-Belted Radius and LDAP clients.

The syntax is as follows:

[LDAP]
Enable = 1
TCPPort = portNumber

Table 42: radius.ini [LDAP] Syntax

Parameter	Function
Enable	• If set to 0, the LDAP Configuration Interface is disabled. • If set to 1, the LDAP Configuration Interface is enabled. Default value is 0. Note: Enabling LCI without changing the access password might leave your Steel-Belted Radius database vulnerable to access by any LDAP client. Read the "LDAP Configuration Interface" chapter of the Steel-Belted Radius Administration Guide before you enable this feature.
TCPPort	Specifies the TCP port number that you want to use for communication between Steel-Belted Radius and LDAP clients. Default value is 667.

[LDAPAddresses] Section

Used by: GEE
Not used by: EE

The [LDAPAddresses] section of radius.ini lets you specify the interfaces on which Steel-Belted Radius listens for LDAP Configuration Interface (LCI) requests. If you want to provide these settings, you must add a section called [LDAPAddresses] to the radius.ini file. This section should contain a list of IP addresses, one per line:

```
[LDAPAddr
esses]
199.198.1
97.196
196.197.198.199
```

If the [LDAPAddresses] section is omitted or empty, Steel-Belted Radius listens for LCI requests on all bound IP interfaces.

EE: The LDAP Configuration Interface is an optional add-on for the Enterprise edition of Steel-Belted Radius. You must license the LDAP Configuration Interface before you can configure or use it.

[MsChapNameStripping] Section

Used by: GEE, EE
Not used by: —

The [MsChapNameStripping] section of radius.ini specifies whether you want Steel-Belted Radius to try to strip domain information from usernames when it tries to match its user entry to the username/password hash forwarded by the enduser. This feature is useful in situations where the username in the Steel-Belted Radius database includes characters the enduser host considers domain information, which it deletes before computing its hash of the user's credentials.

If this feature is enabled:

1. Steel-Belted Radius scans the username in its database looking for delimiter characters that might indicate a domain is prefixed to the username. If a prefix delimiter character is found, the server strips that character (and all characters to the left of the delimiter), generates its own hash of the user's credentials, and compares the result to the hashed credentials forwarded by the enduser to determine if a match is found.

2. If a prefix delimiter is not found (or if the hashed credentials do not match after the prefix is stripped), Steel-Belted Radius scans the username looking for delimiter characters that might indicate a domain is suffixed to the username. If a suffix delimiter character is found, the server strips that character (and all characters to the right of the delimiter), generates its own hash of the user's credentials, and compares the result to the hashed credentials forwarded by the enduser to determine if a match is found.

3. If neither a prefix delimiter nor a suffix delimiter is found (or if a delimiter was found but the hashed credentials did not match), the server uses the entire username string to generate the hashed credentials and compares the result to the hashed credentials forwarded by the enduser to determine if a match is found.

The syntax for the [MsChapNameStripping] section is as follows:

```
[MsChapNameStrip
ping] Enable=1
Prefix
=\\
Suffix
=/@
```

Table 43: radius.ini [MsChapNameStripping] Syntax

Parameter	Function
Enable	• If set to 0 (or omitted), MS-CHAP name stripping is disabled. • If set to 1, MS-CHAP name stripping is enabled. Default value is 0.
Prefix	A list of as many as five ASCII characters to strip from the prefix. If a space character appears in the list, the entire list must be surrounded by quotation marks. Enter a double backslash (\\) to indicate you want to strip the backslash character. A double backslash counts as one character in the list. Default value is \\.
Suffix	A list of as many as five ASCII characters to strip from the suffix. If a space character appears in the list, the entire list must be surrounded by quotation marks. Enter a double backslash (\\) to indicate you want to strip the backslash character. A double backslash counts as one character in the list. Default value is /@.

[Ports] Section

Used by: GEE, EE
Not used by: —

The [Ports] section of radius.ini (Table 43) provides a method for setting the UDP ports used by Steel-Belted Radius.

- If one or more UDPAuthPort settings are specified in the [Ports] section of radius.ini, the port numbers in this section are the only ones on which the server listens for authentication requests. Similarly, if one or more UDPAcctPort settings are specified, they are the only ones on which the server listens for accounting requests.

You can specify as many as 64 port numbers on a Windows server and as many as 4096 ports on a Linux server. If this limit is exceeded, the RADIUS authentication subcomponent fails to initialize.

- If no UDPAuthPort or UDPAcctPort settings are present in the [Ports] section, the server attempts to read the port numbers associated with radius service (authentication) and radacct (accounting) in / etc/services. If successful, the server listens on these port numbers. No more than one port can be specified for the radius service or for the radacct service.
- If no UDPAuthPort settings are present in the [Ports] section and no radius service or radacct is listed in the /etc/services file, the server listens for authentication requests on UDP ports 1645 and 1812 for authentication and UDP ports 1646 and 1813 for accounting.

Note: When auto-restart is enabled and the server is running normally, you typically see two instances of radius.exe in any tool (such as the Task Manager) that you use to monitor processes on the Windows host computer.

If the server will function as a proxy forwarding server, you can specify a block of UDP port numbers from which the proxy RADIUS ports are allocated. Proxy RADIUS allocates port numbers in sets of eight. Port numbers in an allocated block do not have to be contiguous: if a UDP port number that falls in the proxy RADIUS range is in use, proxy RADIUS skips over it.

Table 44: radius.ini [Ports] Syntax

Parameter	Function
SecureTcpAdminAddress	Specifies the IP address of the administrative interface used for communication between SBR Administrator and the Steel-Belted Radius server. If not specified, any network interface on the Steel-Belted Radius server accepts a connection from SBR Administrator.
SecureTcpAdminPort	Specifies the TCP port used for communication between SBR Administrator and the Steel-Belted Radius server. Default value is 1813. Note: Consult Pulse Secure Global Support Center before changing the port number. Using a non-default port may cause communication problems between SBR Administrator and the Steel-Belted Radius server.
TCPControlAddress	Specifies the IP address of the administrative interface on the Steel-Belted Radius server used for SNMP and CCM/ replication communication. If not specified, any network interface on the Steel-Belted Radius server can be used for SNMP and CCM traffic.
TCPControlPort	Specifies the TCP port used for SNMP and CCM/replication communication. Default value is 1812. Note : Consult Pulse Secure Global Support Center before changing the port number. Using a non-default port may cause communication problems between SBR Administrator and the Steel-Belted Radius server.
UDPAuthPort	Specifies the UDP port(s) used for authentication. If you use more than one port, specify each port number on a separate line. Default values are 1645 and 1812. Note: Consult Pulse Secure Global Support Center before changing the port

Parameter	Function
	number. Using a non-default port may cause communication problems between SBR Administrator and the Steel-Belted Radius server.
UDPAcctPort	Specifies the UDP port(s) used for accounting. If you use more than one port, specify each port number on a separate line. Default values are 1646 and 1813. **Note**: Consult Pulse Secure Global Support Center before changing the port number. Using a non-default port may cause communication problems between SBR Administrator and the Steel-Belted Radius server.
UDPProxyPortBlockLength	Specifies the number of addresses in the port number range used for proxy RADIUS communication. Default value is 64.
UDPProxyPortBlockStart	Specifies the starting port number in the port number range used for proxy RADIUS communication. Default value is 28000. **Note**: If you change the default value, choose a number range that does not overlap with well-known UDP ports and proprietary UDP ports on your network. **Note**: You might need to configure network firewalls to allow ports in the specified number range to pass.

For example:

```
[Ports]
SecureTcpAdminPort = 1813
SecureTcpAdminAddress = 192.168.12.15
TcpControlPort = 1812
TCPControlAddress = 192.168.15.55
UDPAuthPort = 1645
UDPAuthPort = 1812
UDPAcctPort = 1646
UDPAcctPort = 1813
UDPProxyPortBlockStart = 28000
UDPProxyPortBlockLength = 64
```

The UDP port assignments entered in the [Ports] section of the radius.ini file override the UDP port assignments specified in the /etc/services file. For more information, see "services File".

[SecurID] Section

Used by: GEE, EE
Not used by: —
The [SecurID] section of radius.ini (

Table 44 contains items specific to RSA SecurID authentication for ISDN users. It provides information that allows Steel-Belted Radius to cache the user's credentials temporarily after a successful SecurID authentication. This technique is necessary to permit a second ISDN B-channel to be authenticated during the user's session. Steel-Belted Radius uses the cached token to authenticate the second channel.

Note: If this feature is not enabled, users who want to authenticate against a SecurID database through an ISDN connection that "bonds" both B-channels will fail to authenticate due to a SecurID security violation. ISDN users running only one B-channel are not affected.

Table 45: radius.ini [SecurID] Syntax

Parameter	Function
CachePasscodes	• If set to yes, RSA SecurID passcode caching is enabled. • If set to no, RSA SecurID passcode caching is disabled. Default value is no.
SecondsToCachePasscodes	The number of seconds to retain the cached SecurID passcode (PIN and token code). Default value is 60 seconds.

[Self] Section

Used by: GEE
Not used by: EE

The [Self] section of radius.ini lists all the realm names that indicate this Steel-Belted Radius server should handle a request. The syntax is as follows:

```
[Self]
Real
mNa
me
Real
mNa
me
.
.
.
```

You can use the [Self] section to map a realm name to the Steel-Belted Radius server. If you acquire a batch of new user accounts, users do not have to change how they enter usernames. They can enter the name User<Delimiter>RealmName or RealmName<Delimiter>User as usual.

When a username comes into Steel-Belted Radius, if the [Self] section lists RealmName, Steel-Belted Radius understands that it is the target, and handles the request locally instead of directing the request elsewhere.

[StaticAcctProxy] Section

The [StaticAcctProxy] section of radius.ini controls the delivery of Accounting messages to additional RADIUS accounting-enabled devices on the network, even when the initial RADIUS transaction is not a proxy RADIUS transaction. The syntax is as follows:

```
[StaticAcct
Proxy]
target    =
proxy
```

where proxy identifies the name of the RADIUS accounting-enabled device.

[Strip] Section

Used by: GEE
Not used by: EE

The [Strip] section specifies how Steel-Belted Radius manipulates the username by stripping the incoming User- Name attribute value of realm names and other "decorations."

The [Strip] section (and accompanying [StripPrefix] and [StripSuffix] sections) look like the following:

```
[Strip]
Authentication=
Yes
Accounting=No
StripPrefixCharacters=@
#%
StripSuffixCharacters=
"! " [StripPrefix]
PrefixStringToStrip
1
PrefixStringToStrip
2
.
.
.
[StripSuffix]
SuffixStringToStrip1
SuffixStringToStrip2
```

Table 46: radius.ini [Strip] Syntax

Parameter	Function
Authentication	If set to yes, the [StripPrefix] and [StripSuffix] rules are used to strip the username before an authentication request is processed. Default value is no.
Accounting	If set to yes, the [StripPrefix] and [StripSuffix] rules are used to strip the username before an accounting request is processed. Default value is no.
StripPrefixCharacters	A list of ASCII characters to strip from the prefix. If a space character appears in the list, the entire list must be surrounded by quotation marks.
StripSuffixCharacters	A list of ASCII characters to strip from the suffix. If a space character appears in the list, the entire list must be surrounded by quotation marks.

[StripPrefix] Section

Used by: GEE
Not used by: EE

The [StripPrefix] section lists prefixes that should be removed from the beginning of usernames, including the delimiter. If a space character appears in the list, the entire list must be surrounded by quotation marks.

```
[Strip]
Authentication=
yes
Accounting=ye
s
```

[Strip
Prefix
]
isp.c
om\
att.ne
t]

In this example, Steel-Belted Radius would strip the prefixes isp.com\ and att.net] from usernames in authentication and accounting requests.

[StripSuffix] Section

Used by: GEE
Not used by: EE

The [StripSuffix] section lists suffixes that should be removed from the end of usernames, including the delimiter.

For
exampl
e:
[Strip]
Authentication=yes
Accounting=yes
[StripSuffix]
@myrealm.com
@yahoo.com

In this example, Steel-Belted Radius would strip the suffixes @myrealm.com and @yahoo.com from usernames in authentication and accounting requests.

[UserNameTransform] Section

Used by: GEE, EE
Not Used By: —

The [UserNameTransform] section (Table 46) lets you specify a rule for transforming user names in RADIUS requests from the form in which they are received to a form in which they can be processed. This can be useful when the form in which users supply their names to the network access device is not compatible with the form in which the RADIUS server applies its rules for proxy forwarding or with the form that the authentication system requires.

The user name transformation rule used to convert input strings to output strings is based on an input format and an output format. The user name transformation rule is applied to user names appearing in RADIUS requests. The user name from the RADIUS request is parsed based on the input format.

- If the user name does not conform to the input format, the rule does not apply and the user name is unchanged.
- If the rule does apply, the parsed elements of the user name are formatted based on the output format to construct the transformed user name:
    1. The User-Name from the Access-Accept (or Acct-Start/Acct-Stop) is compared to the input format rule.

2. If the User-Name matches the rule, it is modified into the output format, and authentication continues.
3. If the User-Name does not match the input format, no modification occurs, and authentication continues.

The transformed user name replaces the original user name in RADIUS processing, just as if the transformed user name had been included in the request. The decision to proxy-forward the packet is based on the transformed user name, and all authentications are based on the transformed user name.

Format strings can be any sequence of characters, and can contain embedded variables enclosed in angle brackets (< >). The backslash (\) is an escape character within text, used to represent literal characters. Within variable names, a backslash is treated as a character, not as an escape; and therefore, variable names may not include right angle brackets (>).

The literal text should be composed of characters not expected to be found in the variable elements. Use punctuation characters such as a slash (/) or an at-sign (@), rather than letters or numbers.

The user name transformation rule can be applied to authentication packets, accounting packets, or both.

```
[UserNameTransform]
In=<input format>
Out=<output format>
Authentication=< yes |
no > Accounting=< yes
| no >
```

Table 47: radius.ini [UserNameTransform] Syntax

Parameter	Function
In	A format string identifying the input format for user names. For example, <user>@<realm>.
Out	A format string identifying the output format for user names. For example, <user>.
Authentication	Set to Yes to enable the transform for authentication requests. Default value is Yes.
Accounting	Set to Yes to enable the transform for accounting requests. Default value is Yes.
Proxy	Set to Yes to enable the transform for proxied requests. Default value is Yes.

For example, the following settings transforms george@acme.com to george: In = <user>@<realm>
Out = <user>

The following settings transform abc/martha@bigco.com to bigco.com::abc/martha: In = <prefix>/<user>@<realm>
Out = <realm>::<prefix>/user

[ValidateAuth] and [ValidateAcct] Sections

Used by: GEE
Not used by: EE

The [ValidateAuth] and [ValidateAcct] sections of radius.ini (Table 47) specify how Steel-Belted Radius validates usernames in authentication and accounting requests. These sections enable Steel-Belted Radius to examine the User-Name attribute in the incoming packet to determine whether it employs a valid character set.

```
[ValidateAuth]
User-Name                    =
RegularExpression
[ValidateAcct]
User-Name = RegularExpression
```

Table 48: radius.ini [ValidateAuth] and [ValidateAcct] Syntax

Parameter	Function
[ValidateAuth]	This section applies only to authentication servers.
[ValidateAcct]	This section applies only to accounting servers.
User-Name	Names the regular expression against which the User-Name attribute is validated. If the User-Name entry is absent from the section or the regular expression is blank, no validation occurs.
RegularExpression	The regular expression lists each valid character or range of characters. A dash (-) indicates a range of alphanumeric characters. For example, A-Z indicates every uppercase alphabetic character. A backslash (\) followed by a non-alphanumeric character indicates that character literally, for example \? indicates the question mark. \ is used as an escape character, as follows: \a bell (7) \b backspace (8) \t tab (0x09) \n newline (10) \v vertical tab (11) \f formfeed (12) \r return (13) \xnn hex value, where nn is a two-digit hexadecimal number \nnn decimal value, where nnn is a three-digit decimal number

The following example permits a string composed only of upper-case and lower-case characters, digits, periods, and commas:

```
User-Name = A-Za-z0-9.,
```

The following example permits upper-case and lower-case characters: `User-Name = A-Za-z`

sbrd.conf File (Linux only)

Used by: GEE, EE

Not Used By: —

The sbrd.conf file (Table 48) is an executable Bourne shell script that is invoked by the sbrd process to initialize the execution environment for Steel-Belted Radius.

Note: In Steel-Belted Radius v5.3, users were instructed to modify the sbrd script if they wanted to change its settings. The sbrd.conf file makes direct modification to the sbrd script unnecessary. Do not modify the sbrd script.

For example:

```
#!/bin/sh
##############
# sbrd.conf
############
##
ULIMIT_CORE_SIZ
E=""
ULIMIT_CORE_COU
NT=3
ULIMIT_OPEN_FILES=1024

RADIUSUMASK=""
RADIUS_HIGH_FDS=
1
ORACLE_MSB_FILE="ORACLE_HOME/rdbms/mesg/ocius.msb"

# Radius executable, options, and arguments
RADIUS="radius"
RADIUSOPTS=""
RADIUSARGS="sbr.xml"
RADIUS_PRIVATE_DIR="$RADIUSDI
R"

# Watchdog executable, options, and arguments
WATCHDOGENABLE=0
WATCHDOG="radiusd"
WATCHDOGOPTS="--config $RADIUSDIR/radiusd.conf --pidfile
$RADIUSDIR/radius.pid"
WATCHDOGARGS="$RADIUSDIR/$SELF"
```

Note: Do not include spaces in parameter settings in the sbrd.conf file.
Correct: ULIMIT_CORE_COUNT=3
Incorrect: ULIMIT_CORE_COUNT = 3

Table 49: sbrd.conf Syntax

Parameter	Function
ULIMIT_CORE_SIZE	Specifies the size of core files generated if Steel-Belted Radius fails. • If set to a value, ULIMIT_CORE_SIZE specifies the maximum size for core files in

Parameter	Function
	1024-byte blocks (Linux). • If set to disabled, Steel-Belted Radius uses the current environment without changes. • If set to "" (two double-quotes with no space between), Steel-Belted Radius uses the current environment, making adjustments as needed. Default value is "".
ULIMIT_CORE_COUNT	Specifies the number of core files maintained on the Steel-Belted Radius server. If the maximum number of core files already exists on the server, Steel-Belted Radius discards the oldest core files and generates a new core file if it fails. • If set to a number in the range 0–999,999,999, the server maintains the specified number of core files. • If set to unlimited, Steel-Belted Radius does not discard exsting core files if it generates a new one. • If set to disabled, Steel-Belted Radius uses the current environment without changes. • If set to "" (two double-quotes with no space between), Steel-Belted Radius uses the current environment, making adjustments as needed. Default value is 3.
ULIMIT_OPEN_FILES	Specifies the number of open files that the Steel-Belted Radius process can have open at one time. • If set to a number in the range 256–1024, the server maintains the specified number of open files. • If set to disabled, Steel-Belted Radius uses the current environment without changes. • If set to "" (two double-quotes with no space between), Steel-Belted Radius uses the current environment, making adjustments as needed. Default value is 1024.
RADIUSMASK	Specifies the file permissions that are withheld when new log files are created. • If set to a umask argument, log files are created with the specified permissions with- held from Owner, Group, and Other users. • If set to "", log files are created with the default access permissions established by the ambient umask for Owner, Group, and Other users. Refer to the Steel-Belted Radius Administration Guide for information on how to configure and use umask to control file permission settings.
RADIUS_HIGH_FDS	• If set to 0, management of file descriptors is disabled. You can set RADIUS_HIGH_FDS to 0 if you specified a value of 256 or lower for the ULIMIT_OPEN_FILES parameter. • If set to 1, management of file descriptors is enabled. You should set RADIUS_HIGH_ FDS to 1 if you specified a value greater than 256 for the ULIMIT_OPEN_FILES parame- ter. Default value is 1.
ORACLE_MSB_FILE	Specifies the absolute path to the locale-specific Oracle message file. • If you enter the path name to a message file, the file descriptor that is returned is greater than 255. If you enter "", Steel-Belted Radius uses the descriptor returned by a standard library open() call.

Parameter	Function
RADIUS	Default value is “radius”. Do not change this value unless instructed to do so by technical support.
RADIUSOPTS	Specifies options used when running Steel-Belted Radius. Default value is “”. Do not change this value unless instructed to do so by technical support.
RADIUSARGS	Default value is “sbr.xml”. Do not change this value unless instructed to do so by technical support
RADIUS_PRIVATE_DIR	Default value is “$RADIUSDIR”. Do not change this value unless instructed to do so by technical support.
WATCHDOGENABL	• If set to 0, the Steel-Belted Radius watchdog process, which restarts Steel-Belted Radius if it fails, is disabled. If set to 1, the Steel-Belted Radius watchdog is enabled. Default value is 0.
WATCHDOG	Specifies the name of the Steel-Belted Radius watchdog process. Default value is radiusd. Do not change this value unless instructed to do so by technical support.
WATCHDOGOPTS	Default value is --config $RADIUSDIR/radiusd.conf --pidfile $RADIUSDIR/radius.pid. Do not change this value unless instructed to do so by technical support.
WATCHDOGARGS	Default value is $RADIUSDIR/$SELF. Do not change this value unless instructed to do so by technical support.

Services File

The services file can be used to assign default UDP ports for RADIUS communications to and from the Steel- Belted Radius server. Steel-Belted Radius reads the services file at startup. Among the items of information in the services file are the port assignments for RADIUS authentication and accounting services. Figure 1 illustrates part of a sample services file.

Figure 1: Sample Services File

```
# This file contains port numbers for well-known services
# defined by IANA. Format:
#
# <service> <port number>/<protocol> [aliases...] [#<comment>]
#
echo            7/tcp
echo            7/udp
discard         9/tcp       sink null
discard         9/udp       sink null
systat          11/tcp      users       #Active users
systat          11/tcp      users       #Active users
daytime         13/tcp
```

The location of the services file depends on your operating system:

- Linux: /etc/ (may be mapped using NIS or NIS+)

- Windows: C:\WINDOWS\system32\drivers\etc\

If no entry for radius or radacct is found in the services file, Steel-Belted Radius uses the default UDP ports (1645 and 1812 for authentication, 1646 and 1813 for accounting).

Steel-Belted Radius can be configured to use any available UDP ports for authentication and accounting:

1. Use a text editor to open the services file.
2. To set the port for authentication, set the value of the radius parameter. For example: radius 1812/udp # RADIUS authentication protocol
3. To set the port for accounting, set the value of the radacct parameter. For example: radacct 1813/udp # RADIUS accounting protocol

   Note: Port number assignments made in the radius.ini file override the assignments made in this file. See "[Ports] Section" for more information.

You can determine the ports that Steel-Belted Radius is using at any time by examining the server log file for that time period.

Note: If another RADIUS server is running on the same host, you must modify the services file to avoid port number conflicts if the other RADIUS server binds to the default ports before Steel-Belted Radius starts.

servtype.ini File

Used by: GEE
Not used by: EE

The servtype.ini file configures service type mapping in Steel-Belted Radius. Service type mapping allows a single user to have multiple authorization attribute sets based on the service type the user is requesting. The service type is determined based on request attributes using rules that may differ depending on the network access device.

Using static configuration parameters in the servtype.ini file, you can specify, on a device-by-device basis, a mapping of request attributes and/or values to service type strings. These strings can be attached to the
username as a prefix or as a suffix. The elaborated username is used for both authentication and authorization, and for allowing different authorizations based on service type requested.

Refer to the Steel-Belted Radius Administration Guide for information on how to configure and use service type mapping.

[Settings] Section

The [Settings] section of servtype.ini (Table 49) controls how the service type string should be attached to the username prior to look-up in the Native User database.

Note: If Prefix and Suffix are both set to 0 in the [Settings] section, service type mapping is disabled.

Table 50: servtype.ini [Settings] Syntax

Parameter	Function
Prefix	Specifies whether the service type string should be prefixed to the username prior to

Parameter	Function
	look-up in the Native User database. • If set to 1, the service type string is prefixed to the username. • If set to 0, the service type string is not prefixed to the username. Default value is 0.
Suffix	Specifies whether the service type string should be suffixed to the username prior to look-up in the Native User database. • If set to 1, the service type string is suffixed to the username. • If set to 0, the service type string is not suffixed to the username. Default value is 0.
Default	Mapping name that is used when an Access-Request message is received from a network access device not listed in the [NAS] section of servtype.ini. If you do not configure a Default setting and the server cannot determine the mapping in any other way, the server ignores the service type and authenticates the user without it.

[NAS] Section

The [NAS] section of the servtype.ini file lets you map network access devices to [mapping] sections. The syntax for [NAS] is as follows:

```
[NAS]
NASname =
mappingName
NASname =
mappingName
```

Each NASname entry in the [NAS] section must match the name of a RADIUS client entry in the Steel-Belted Radius database. When an Access-Request is received, its NAS-IP-Address attribute is matched to a RADIUS client entry in the database. If a match can be found and the RADIUS client name matches a NASname in the [NAS] section, Steel-Belted Radius looks for a corresponding [Mapping] section in the servtype.ini file.

[MappingName] Section

Each [MappingName] section of the servtype.ini file identifies the strings to be added to the username for lookups in the Native User database, which allows Steel-Belted Radius to retrieve the appropriate return list, and specifies the rules an incoming Access-Request packet must meet before Steel-Belted Radius returns an Access- Accept message. The name of each [MappingName] section must match a mappingName entry in the [NAS] section.

The syntax for each [MappingName] section is as follows:

```
[mapping]
ServiceTypeStrin
g RADIUSattribute
= value
~RADIUSattribute =value
```

Each rule is a statement about an attribute that must be present in the incoming Access-Request packet.

Each rule must be indented with a tab character, followed by a RADIUSattribute = value string, followed by a carriage return. Every component of the rule is optional, so there are many syntax variations.

If a rule includes a RADIUSattribute field, this field must identify a standard or vendor-specific RADIUS attribute that is known to the server. If a rule provides an optional value field, this field must name a valid possible value for that attribute.

If the RADIUSattribute field for a rule is preceded by a tilde (~), then the specified RADIUSattribute, if present in the Access-Request packet, must have a value other than value for the rule to be true. If the RADIUSattribute is not present in the Access-Request packet, or if it is present and has the value specified, the rule is false and authorization fails.

Example

[Settings]
Prefix=1
Suffix=0
Default=defaultmap [NAS]
nas1=nas1map
nas2=nas2map
[nas1map] ppp: Framed-Protocol=1 Service-Type=2 vpn: Framed-Protocol=6 ~Service-Type=2 other: Framed-Protocol Service-Type
[nas2map] analog: NAS-Port-Type=1 isdn: NAS-Port-

Type=2
[defaultm
ap] ppp:

update.ini File

Used by:

GEE Not

used by:

EE

The update.ini initialization file controls what information is updated when Steel-Belted Radius receives a HUP or USR2 signal, which is sent by means of the signal command on Linux and by means of the radhup.exe and radusr2.exe programs on Windows.

When Steel-Belted Radius receives a HUP or USR2 signal, it performs the tasks specified in the [HUP] and [USR2] sections of the update.ini file. You can perform tasks selectively by modifying update.ini to toggle specific settings; for example, you can issue a HUP signal to initiate one set of tasks, and then modify update.ini and issue another HUP signal to initiate a different set of tasks.

The update.ini file installed with Steel-Belted Radius causes Steel-Belted Radius to re-read all settings when it receives a HUP signal and to clear its statistics when it receives a USR2 signal.

[HUP] and [USR2] Sections

The [HUP] section of update.ini specifies what tasks Steel-Belted Radius should perform when it receives a HUP signal. The [USR2] section of update.ini specifies what tasks Steel-Belted Radius should perform when it receives a USR2 signal.

```
[HUP]
ResetStats = 0
ResetThreadHighWaterMarks = 0
Update3GPP = 1
Update3GPP2 = 1
UpdateAdminAccess = 1
UpdateAutoStop = 1
UpdateCCAGateways = 1
UpdateDHCPPools = 1
UpdateEap = 1
UpdateLogAndTraceLevel = 1
UpdateLogfilePermissions = 1
UpdatePlugins = 1
UpdateProxy = 1
UpdateValuePools = 1
```

Table 51 lists the settings that may be present in the [HUP] or [USR2] section of update.ini.

Table 51: update.ini [HUP] and [USR2] Syntax

Parameter	Function
ResetStats	• If set to 0, do not reset Steel-Belted Radius statistics to 0 when a HUP or USR2 signal is received. • If set to 1, reset Steel-Belted Radius statistics to 0 when a HUP or USR2 signal is

Parameter	Function
	received. Default value is 0 in the [HUP] section. Default value is 1 in the [USR2] section.
ResetThreadHighWaterMarks	• If set to 0, do not reset Steel-Belted Radius high thread statistics (High-Auth-Threads-Since-Reset, High-Acct-Threads-Since-Reset, and High-Total-Threads-Since-Reset) when a HUP or USR2 signal is received. • If set to 1, reset Steel-Belted Radius high thread statistics to 0 when a HUP or USR2 signal is received. Default value is 0 in the [HUP] section. Default value is 1 in the [USR2] section.
UpdateAdminAccess	• If set to 0, do not update administrator access settings when a HUP or USR2 signal is received. • If set to 1, update administrator access settings when a HUP or USR2 signal is received. Default value is 1 in the [HUP] section. Default value is 0 in the [USR2] section.
UpdateAutoStop	• If set to 0, do not update the Proxy AutoStop settings (by re-reading the AcctAutoSt-opEnable setting in radius.ini) when a HUP or USR2 signal is received. • If set to 1, update the Proxy AutoStop settings (by re-reading the AcctAutoStopEnable setting in radius.ini) when a HUP or USR2 signal is received. Default value is 1 in the [HUP] section. Default value is 0 in the [USR2] section.
UpdateDHCPPools	• If set to 0, do not update EAP settings specified in eap.iniwhen a HUP or USR2 signal is received. • If set to 1, update EAP settings specified in eap.ini when a HUP or USR2 signal is received. Default value is 1 in the [HUP] section. Default value is 0 in the [USR2] section.
UpdateEAP	• If set to 0, do not update EAP settings specified in eap.ini when a HUP or USR2 signal is received. • If set to 1, update EAP settings specified in eap.ini when a HUP or USR2 signal is received. Default value is 1 in the [HUP] section. Default value is 0 in the [USR2] section.
UpdateLogAndTraceLevel	• If set to 0, do not update log and trace levels specified in radius.ini when a HUP or USR2 signal is received. • If set to 1, update log and trace levels specified in radius.ini when a HUP or USR2 signal is received. Default value is 1 in the [HUP] section.

Parameter	Function
	Default value is 0 in the [USR2] section.
UpdateLogfilePermissions (Linux only)	• If set to 0, do not update logfile permissions specified in the logfile configuration files when a HUP or USR2 signal is received. • If set to 1, update logfile permissions specified in the logfile configuration files when a HUP or USR2 signal is received. Default value is 1 in the [HUP] section. Default value is 0 in the [USR2] section.
UpdatePlugins	• If set to 0, do not update plug-ins that support dynamic re-reading of configuration settings when a HUP or USR2 signal is received. • If set to 1, update plug-ins that support dynamic re-reading of configuration settings when a HUP or USR2 signal is received. Default value is 1 in the [HUP] section. Default value is 0 in the [USR2] section. **Note:** The TLS, TTLS, and PEAP plug-ins currently support dynamic configuration updates.
UpdateProxy	• If set to 0, do not update realm configuration when a HUP or USR2 signal is received. • If set to 1, update realm configuration (by re-reading proxy.ini *.pro, and *.dir files) when a HUP or USR2 signal is received. Default value is 1 in the [HUP] section. Default value is 0 in the [USR2] section.
UpdateValuePools	• If set to 0, do not update attribute value pool settings (in *.rr files when a HUP or USR2 signal received. • If set to 1, update attribute value pool settings (in *.rr files when a HUP or USR2 signal is received. Default value is 1 in the [HUP] section. Default value is 0 in the [USR2] section.

Auto-Restart Files (Linux only)

Used by: GEE
Not used by: EE

When enabled, the auto-restart module acts as a watchdog daemon, monitoring the status of the Steel-Belted Radius executable and restarting it as needed. Automatic restart is disabled by default. Perl must be installed on the Steel-Belted Radius server if you want the automatic restart module. Perl support is not required for syslog but is available.

Perl SNMP Support

You can configure the auto-restart module to send SNMP traps to record auto-restart events. Perl SNMP support resides in the Perl SNMP_Session module, which provides access to remote SNMP agents. Refer to the ReleaseNotes.txt file for radiusd for information on how to install and configure the Perl SNMP_Session module.

Perl SNMP support allows Steel-Belted Radius to send SNMP traps to a variety of SNMP agents, including

the Sun Management Center, which is distributed with some Sun hardware platforms. Sun Management Center is not required to run radiusd.

Perl syslog Support

The optional perl package syslog.ph is used to log the watchdog daemon status. You can configure the auto-restart module to send syslog messages to record auto-restart events. To use syslog reporting, you can use the h2ph utility to create a syslog.ph file. The following example assumes site_perl/5.005 is in @INC:

```
su - root
cd /user/include/sys
/usr/perl15/bin/h2ph    -d/usr/perl15/site_perl/5.005 syslog.h
```

If you do not want to use syslog, you should use the -d or --logfile options for the radiusd command to open a regular log file (radiusd.log).

S90radius/sbrd Script

To enable the auto-restart module, you must edit the sbrd script (Linux) to ensure that a certain line in the script is uncommented (the hash mark # is removed from the start of the line), as follows:

1. If Steel-Belted Radius is already running, become superuser and type the following command to stop the server:
   Linux:/etc/init.d/sbrd stop
2. Edit the radius script (S90radius or sbrd). The line you want to edit for auto-restart appears as follows:

   #RADIUS="$RADIUSDIR/radiusd--server$RADIUSDIR/radius"

   The --server option identifies the location and name of the Steel-Belted Radius executable file, and must be present on the radiusd command line.
3. If the comment hash mark (#) is present at the start of the line, remove it.
4. Save and exit the file.
5. Type the following command to restart the server: sbrd invokes radiusd, which starts the RADIUS service:
   Linux:/etc/init.d/sbrd start

radiusd Script

If you enable the auto-restart module, the sbrd startup/shutdown script runs radiusd instead of the radius executable file. radiusd executes radius as a child process and monitors its health by a polling mechanism. Polling parameters are configurable by editing the radiusd.conf file in the server directory; the relevant timeouts and logging options are near the beginning of the file.

The default radiusd.conf settings cause the auto-restart feature to work as follows:

If the radius server executable fails to respond to status polling from radiusd within 17 seconds, radiusd attempts to stop radius using SIGTERM (a polite shutdown). If radius does not shut down within 60 seconds, SIGKILL
(a hard kill) is used to stop it. After shutdown by either method, radiusd starts a new radius child process. If this radius child does not respond to status polling within 60 seconds of startup, it is presumed dead; a misconfiguration of the server is assumed; and radiusd terminates with a critical error.

Note: The radius executable normally runs as a daemon. When the automatic-restart module is enabled, the radius executable is run as a child process of radiusd instead of being run as a daemon.

While the auto-restart module is enabled, all informational, debugging, warning, error, and critical messages from radiusd are recorded in the following locations:

- Syslog – Messages are written to the syslog system logging facility.
- Log file – If syslog is not available, messages are written to the server log file specified using the --logfile option on the radiusd command line; for example:

```
RADIUS="$RADIUSDIR/radiusd \
--server $RADIUSDIR/radius \
--logfile/var/log/radd.log"
```

If the --logfile option is not already included in the radiusd command line, you may add it.

Note: Options processed by radiusd are preceded by two dashes (--). Options preceded with a single dash are passed to Steel-Belted Radius.

Note: If Perl is not installed in the /usr/local/bin/ directory, the following error message occurs when you start the Steel-Belted Radius server:

./S90radius:/RadiusHome/radiusd:notfound

To fix this error, edit the first line of the radiusd file in the RADIUS directory so that the directory structure points to the correct Perl interpreter executable:

#!/usr/local/bin/perl

ScriptConfiguration

The radiusd.conf configuration file (Table 52) provides settings for the radiusd automatic-restart module.

Table 52: radiusd.conf Syntax

radiusd.conf Parameter	Function
WatchdogIntervalPing	Number of seconds the automatic-restart module waits between sending status inquiries. Default value is 5 seconds.
WatchdogIntervalMaxPong	Number of seconds the automatic-restart module waits for a reply before issuing a SIGTERM (shutdown) message. Default value is 17 seconds.
WatchdogIntervalMaxStartup	Number of seconds during which the server is expected to be able to start up. Default value is 60 seconds.
WatchdogIntervalMaxShutdown	Number of seconds during which the server is expected to be able to shut down. Default value is 60 seconds.
SnmpManager = hostname community port version	Identifier for an SNMP management station that should receive traps from the automatic-restart module. You can specify more than one SNMP management station. For each SNMP management station, enter the following: • hostname – IP address of the SNMP management station. • community – SNMP community string.

radiusd.conf Parameter	Function
	• port – UDP port number used for SNMP trap messages. UDP port 162 is the default. • version – SNMP version number. Default value is 1. If SnmpManager is undefined, SNMP traps may still be logged, but are not transmitted on the network.
SnmpInterface	Identifies the IP network interface to be used to generate SNMP trap messages. You can specify interfaces by name or by IP address. If you enter any, the first IPv4 interface the automatic-restart module finds is used. If you leave this parameter blank, generation of SNMP trap messages is disabled.
SnmpCommandTrap	Specifies how SNMP trap messages should be forwarded: • You can specify the pathname and filename for a module or executable whose syn- tax matches the SMC snmptrap utility. • You can specify SNMP_Session.pm to deliver SNMP traps to the management station using the Perl modules. If you leave the parameter blank, SNMP trap messages are not generated. Default value is blank.
SnmpCommandUptime	Specifies how the automatic-restart module determines elapsed time for timestamps in trap messages. You can specify the pathname and filename for a module or executable whose syntax matches the SMC uclock utility. If you leave the parameter blank, the automatic restart module calculates elapsed time relative to its own start time. Default value is blank.
SnmpEnterprise	Specifies the OID prefix for enterprise-specific trap messages, which is used to select the appropriate MIB for decoding traps. Default value is 1.3.6.1.4.1.1411.1.1. If you leave the parameter blank, SNMP trap messages are not generated.
SnmpGenericTrapType= 6	Specifies the enterprise-specific trap type, which must be 6 according to the SNMPv1 standard. Do not change this value without a specific reason.
SnmpTrapWatchdogStarted	Specifies the trap type for messages indicating that the automatic-restart module is started. Default value is 113. Enter 0 to disable this type of trap.
SnmpTrapWatchdogStopped	Specifies the trap type for messages indicating that the automatic-restart module is stopped. Default value is 114. Enter 0 to disable this type of trap.
SnmpTrapWatchdogRadius	Specifies the trap type for messages indicating that the RADIUS server is restarted.

radiusd.conf Parameter	Function
Started	Default value is 115. Enter 0 to disable this type of trap.
SnmpTrapWatchdogRadiusTerm	Specifies the trap type for messages indicating that the RADIUS server is not responding and that the automatic-restart module has sent the SIGTERM signal. Default value is 5028. Enter 0 to disable this type of trap.
SnmpTrapWatchdogRadiusKill	Specifies the trap type for messages indicating that the RADIUS server is not responding and that the automatic-restart module has sent the KILL signal. Default value is 5029. Enter 0 to disable this type of trap.
SnmpTrapWatchdogAborted	Specifies the trap type for messages indicating that the RADIUS server is not responding and that the automatic-restart module has given up and aborted. Default value is 10051. Enter 0 to disable this type of trap.
SnmpTrapWatchdogFailedInit	Specifies the trap type for messages indicating that the automatic-restart module failed to start, which may indicate a misconfiguration issue. Default value is 10052. Enter 0 to disable this type of trap.

application.properties File

Function: Configures KeyStore information of Java Web Server (Jetty).

Used by: GEE, EE

Table 53: application.properties Syntax

Parameter	Function
server.ssl.key-store-type	Specifies the type of Java keystore. Values: PKCS12 (.pfx, .p12), JKS (.jks) Default: PKCS12
server.ssl.key-store	Specifies absolute path to certificate. Example: /opt/customSSLCert.pfx
server.ssl.key-store-password	Specifies the password to open the certificate

Note: This file is introduced as a part of SBR 6.26 where a separate Java process/services runs in the server machine to host the GUI application.

Note: This file is used to configure custom certificate for Java Web Server. For more details, refer to **Appendix - H** in the Steel-Belted radius Administration guide.

Chapter 4

Attribute Processing Files

This chapter describes the usage and settings for the Steel-Belted Radius attribute processing and dictionary files, which specifies RADIUS attributes.

Overview

For each product listed in the vendor.ini file, Steel-Belted Radius provides a dictionary (.dct) file. Dictionary files enable Steel-Belted Radius to exchange attributes with RADIUS clients. Like initialization files, dictionary files are loaded at startup time, and reside in the Steel-Belted Radius directory:

*.dct
dictio
na.dc
m

- Dictionary files identify the attributes Steel-Belted Radius should expect when receiving RADIUS requests from a specific type of device.
- Dictionary files identify the attributes Steel-Belted Radius should include when sending a RADIUS response to a specific type of device. Figure 2 illustrates the format of a dictionary file.

Figure 2: Sample Dictionary File

```
##################################################################
#Juniper.dct - RADIUS dictionary for Juniper M-160 and M-40Es
# (See README.DCT for more details on the format of this file)
##################################################################
#Use the RADIUS specification attributes
#
@radius.dct
#
# Juniper specific parameters
#
MACRO Juniper-VSA(t,s) 26 [vid=2636 type1=%t% len1=+2 data=%s%]
ATTRIBUTE Juniper-Local-User-Name Juniper-VSA(1, string) r
```

```
ATTRIBUTE Juniper-Allow-Commands Juniper-VSA(2,
string)r ATTRIBUTE Juniper-Deny-Commands Juniper-
VSA(3, string) r ATTRIBUTEJuniper-Allow-Configuration
Juniper-VSA(4, string)r ATTRIBUTEJuniper-Deny-
ConfigurationJuniper-VSA(5,string)r
##################################################################
#Juniper.dct -Juniper Networks dictionary
##################################################################`
```

Dictionary File Location

Windows: Dictionary files must be placed in the same directory as the Steel-Belted Radius service. While starting up, Steel-Belted Radius scans its home directory for all files with an extension of .dct (standard dictionary files) and uses the list to create a "master" dictionary, which includes all known attributes.

Linux: Dictionary files must be placed in the same directory as the Steel-Belted Radius daemon. During initialization, Steel-Belted Radius reads the file dictiona.dcm in the server directory to get a list of files with an extension of .dct (standard dictionary files) and uses the list to create a "master" dictionary, which includes all known attributes.

Dictionary File Records

Records in a dictionary file must begin with one of the keywords listed in Table 53.

Table 54: Dictionary File Keywords

Keyword	Function
@	Include the referenced file
ATTRIBUTE	Define a new attribute
VALUE	Define a named integer value for an attribute
MACRO	Define a macro used to simplify repetitive definitions
OPTIONS	Define options beyond the scope of attribute definitions
#	Ignore this text (comment)

Editing Dictionary Files

The product-specific files shipped with Steel-Belted Radius reflect specific vendors' implementations of RADIUS clients. Therefore, you do not usually need to modify the dictionary files shipped with Steel-Belted Radius.

However, if your network access device vendor provides information about a new product, a new attribute, or a new value for an attribute, you can add this information to your existing Steel-Belted Radius configuration by editing dictionary files.

Before you edit an existing dictionary file or create a new one, you must do the following to integrate your changes into Steel-Belted Radius:

1. Add a new vendor-product entry to vendor.ini so that you can reference the new dictionary while configuring Steel-Belted Radius.
2. Place your dictionary file in the same directory as the Steel-Belted Radius service or daemon.

3. Edit the dictiona.dcm file so that it includes your new dictionary file.
4. Stop and restart the server.

Include Records

Records in a dictionary file that begin with the @ character are treated as special include records. The string that follows the @ character identifies the name of a dictionary file whose contents are to be included. For example, the entry @vendorA.dct would include all of the entries in the file vendorA.dct.

Include records are honored only one level deep. For example, if file vendorA.dct includes file radbase.dct and radbase.dct includes radacct.dct, vendorA.dct incorporates records in radbase.dct but not those in radacct.dct.

Master Dictionary File

The master dictionary dictiona.dcm consists of include records that reference vendor-specific dictionaries. The order in which vendor-specific dictionaries are included in the master dictionary has significance only if two vendor-specific dictionaries contain conflicting definitions for the same attribute or attribute value. The first definition of an attribute or attribute value takes precedence over later definitions of the same attribute or attribute value. For example, if master dictionary dictiona.dcm consists of the following include records:

@vendorA.dct
@vendorB.dct
@vendorC.dct

then attributes and attribute values defined in vendorA.dct override attributes and attribute values defined in vendorB.dct or vendorC.dct, and attributes and attribute values in vendorB.dct override attributes and values defined in vendorC.dct.

ATTRIBUTE Records

Attribute records conform to the following

syntax: ATTRIBUTE attrib_name attrib_id

syntax_type flags

Table 55: ATTRIBUTE Record Syntax

Parameter	Function
attrib_name	Name of the attribute (up to 31 characters with no embedded blanks).
attrib_id	Integer in the range 0 to 255 identifying the attribute's encoded RADIUS identifier.
syntax_type	Syntax type of the attribute.
flags	Defines whether an attribute appears in the checklist, the return list (or both), whether it is multi-valued and whether it is orderable.

Note: One limitation of standard dictionary files (the attrib_id of all the attribute records must be unique) is waived for the master dictionary file. Multiple vendors can define different attribute names for the same attribute identifier (assuming the attribute identifier is not already used in the base RADIUS specification). Since attributes in the Steel-Belted Radius database are stored by name (rather than by attrib_id), this introduces no ambiguity into the database.

The following example illustrates a typical attribute record: ATTRIBUTE Framed-IP-Netmask 9 ipaddr Cr

This attribute record specifies all of the following:

- An attribute named Framed-IP-Netmask is supported.
- The attribute's encoded RADIUS identifier is 9.
- The attribute must use the syntax of an IP address.
- Flag characters specify that the attribute can appear multiple times in a checklist (C) and at most one time in a return list for User or profile entries (r) in the Steel-Belted Radius database.

Attribute Name and Identifier

No two attribute records in a single dictionary file should have the same attrib_name or attrib_id. If a duplicate attrib_name or attrib_id is encountered, the later definition of the attribute is ignored in favor of the earlier one.

Syntax Type Identifier

Standard syntax_type identifiers are listed in Table 55.

Table 56: Syntax Type Identifiers

Syntax Type	Function
hexadecimal	Hexadecimal string.
hex4	4-byte (32-bit) unsigned hexadecimal number.
int1	1-byte (8-bit) unsigned decimal number.
int4, integer	4-byte (32-bit) unsigned decimal number.
signed-integer	4-byte (32-bit) signed decimal number. A number with a 1 in the first bit position is interpreted as a negative number.
ipaddr	IP address or IP netmask attribute.
ipaddr-pool	IPv4 address selected from an IP address pool.
ipxaddr-pool	IPX network number selected from an IPX address pool.
string	String attribute (includes null terminator).
stringnz	String attribute (without null terminator).
time	Time attribute (number of seconds since 00:00:00 GMT, 1/1/1970).

Note: Signed integer support is limited to attributes received in packets and processing relating to those attributes, such as accounting logs, authentication logs, authentication reports, and SQL plug- ins. SBR Administrator does not support signed integers, and treats signed and unsigned integers as unsigned integers.

Compound Syntax Types

In addition to the standard syntax_type identifiers listed in Table 54, the dictionary can accommodate compound syntax types for use in defining vendor-specific attributes. Instead of a single syntax_type identifier, one or more of the options listed in 56 can be combined inside square brackets to form a compound

syntax type.

Table 57: Compound Syntax Types

Option	Function
vid=nnn	The device manufacturer's SMI Network Management Private Enterprise code (assigned by ISO) in decimal form.
typeN=nnn	Type setting for vendor-specific attribute as defined in the RADIUS specification; N specifies the length of the field (in bytes), nnn specifies the decimal value of the field.
lenN=nnn	Length field for vendor-specific attribute as defined in the RADIUS specification; N specifies the length of the field (in bytes), nnn specifies the decimal value of the field (a plus sign prior to the value indicates that the length of the data portion is to be added to nnn to obtain the actual length).
fillN=nnn	Fill field setting for non-integer tunneled attributes; N specifies the size of the field to be filled with the value specified by nnn.
tag=nnn	Tunnel attributes include a tag field, which may be used to group attributes in the same packet which refer to the same tunnel. Since some vendors' equipment does not support tags, this syntax type is optional and must be present for the attribute to include a tag field. A value of 0 indicates that the field should be present but ignored.
data=syntax_type	The actual data to be included in the attribute; the syntax can be any of the standard syntax types.

An example of a vendor-specific attribute definition follows:
ATTRIBUTE vsa-xxx 26 [vid=1234 type1=1 len1=+2 data=string] R

Flag Characters

The flags setting consists of the concatenation of one or more flag characters from the list in Table 57.

Table 58: Flag Characters

Flag Character	Function
b or B	Indicates that an attribute may be bundled in a single Vendor-Specific-Attribute for a particular vendor id. It may be included as one of a series of subattributes within a single VSA.
c	Attribute can appear once within a user or profile check-list.
C	Attribute can appear multiple times within a user or profile check-list.
r	Attribute can appear once within a user or profile return-list.
R	Attribute can appear multiple times within a user or profile return-list.
t	Attribute can appear once within a tunnel attribute list.
T	Attribute can appear multiple times within a tunnel attribute list.
o or O	Attribute is orderable; the administrator can control the order in which such attributes are stored in the Steel-Belted Radius database (this flag makes sense only for multi-valued attributes).

VALUE Records

Value records are used to define names for specific integer values of previously defined integer attributes. Value records are never required, but are appropriate where specific meaning can be attached to an integer value of an attribute. The value record must conform to the following syntax:

VALUE attrib_namevalue_nameinteger_value

Table 59: VALUE Records

Flag Character	Function
attrib_name	Name of the attribute (up to 31 characters with no embedded blanks)
value_name	Name of the attribute value (up to 31 characters with no embedded blanks)
integer_value	Integer value associated with the attribute value

No two value records in a dictionary file should have the same attrib_name and value_name or the same attrib_ name and integer_value. If a duplicate is encountered, the later definition of the attribute value is ignored in favor of the earlier one (the earlier one is considered to be an override).

The following example illustrates the use of the VALUE record to define more user-friendly attribute values for the Framed-Protocol attribute:

```
ATTRIBUTE       Framed-Protocol      7       integer      Cr
VALUE           Framed-Protocol      PPP     1
VALUE           tFramed-Protocol     SLIP    2
```

Using these dictionary records, the administrator need not remember that the integer value 1 means PPP and the integer value 2 means SLIP when used in conjunction with the Framed-Protocol attribute. Instead, the Steel- Belted Radius Administrator program lets you choose from a list of attribute values including PPP and SLIP.

Macro Records

Macro records are used to streamline the creation of multiple vendor-specific attributes that include many common parameters. A macro record can be used to encapsulate the common parts of the record. The macro record must conform to the following syntax:

MACRO macro_name (macro_vars) subst_string

Table 60: MACRO Records

Parameter	Function
macro_name	Name of the macro
macro_vars	One or more comma-delimited macro variable names
subst_string	String into which macro variables are to be substituted; any sequence of characters conforming to the format %x% for which a macro variable called x has been defined undergo the substitution process

The following example illustrates the use of a macro that simplifies the specification of multiple vendor-

specific attributes:

```
MACROCisco-VSA(t, s) 26[vid=9type1=%t% len1=+2data=%s%]
ATTRIBUTE Cisco-xxx Cisco-VSA(1, string) R
ATTRIBUTE Cisco-yyy Cisco-VSA(4, int4) C
ATTRIBUTE Cisco-zzz Cisco-VSA(9, ipaddr)
r
```

The macro preprocessor built into the Steel-Belted Radius dictionary processing would translate the records in the preceding example to the following records before being processed.

```
ATTRIBUTE Cisco-xxx 26 [vid=9 type1=1 len1=+2 data=string] R
ATTRIBUTE Cisco-yyy 26 [vid=9 type1=4 len1=+2 data=int4] C
ATTRIBUTE Cisco-zzz 26 [vid=9 type1=9 len1=+2 data=ipaddr] r
```

OPTION Records

By default, each vendor-specific attribute is encoded in a single VSA attribute. The format of a VSA attribute is described in Table 60.

Table 61: OPTION Records

Bits	Content
0 - 7	Type: contains the value 26
8 - 16	Length of data in bytes
17 - 47	Vendor ID
48 - on	Vendor data

If you provide a parameter to the OPTION setting, however, multiple vendor-specific attributes can be present in the vendor-data portion of a single VSA record.

The OPTION record must conform to the following format: OPTION bundle-vendor-id = vid

Note: You must set the B flag for attribute bundling to occur. For a particular vendor-specific attribute to be bundled, you must set the OPTION record for the vendor's vendor-ID and set the B (or b) flag for the specific attribute.

The Nortel Rapport dictionary supports this option, for example. If you want to combine Nortel's vendor-specific attributes in a single VSA, you would provide the entry:

```
OPTION bundle-vendor-id=562
```

This is because 562 is Nortel's Vendor ID, as set in the MACRO record. The Nortel Rapport vendor-specific attributes now would be concatenated within the vendor-data portion of a RADIUS VSA attribute (up to 249 octets).

classmap.ini File

Used by: GEE, EE
Not Used By: —

The classmap.ini initialization file specifies what Steel-Belted Radius does with RADIUS attributes encoded in one or more Class attributes included in accounting requests it receives.

[AttributeName] Section

The [AttributeName] section of classmap.ini specifies whether RADIUS information encapsulated in a Class attribute should be appended to an accounting request or replace a current value in an accounting request. If one attribute is replaced by another, the original attribute can be added to the request with a different identifier.

```
[AttributeName]
<add | replace> = Attribute [,Attribute]
```

Table 62: classmap.ini [Attributename] Syntax

Parameter	Function
AttributeName	Name of the attribute encoded into the Class attribute by the authenticating server.
<add | replace>	Specifies whether the attribute value should be added to the accounting request (leaving all other values intact) or whether one value should replace another in the accounting request.
Attribute	Specifies the name of the attribute that should be added to the accounting request, which contains the original value of the attribute identified by AttributeName.
[,Attribute]	Specifies the name of the attribute in the accounting request that should contain the value of the attribute displaced when the value of AttributeName replaces the existing Attribute value. Valid only when the replace keyword is used.

Note: The RADIUS Class attribute cannot contain IPv6 attributes.

In the following example, the encapsulated User-Name attribute would replace theexisting User-Name in the accountingrequest.

```
[User-name]
replace = User-Name
```

In the following example, the encapsulated User-Name attribute would be placed in the accounting request as User-Name, and the original value for User-Name would be added to the request as Funk-Full-User-Name.

```
[User-name]
replace = User-Name, Funk-Full-User-Name
```

In the following example, the encapsulated User-Name attribute would be added to the accounting request as a new attribute, and the original User-Name attribute would remain unchanged.

```
[User-Name]
add = Funk-Full-User-Name
```

filter.ini File

Used by: GEE, EE
Not used by: —

Note: You should use the SBR Administrator to maintain settings in the filter.ini file. You should not edit the filter.ini file manually.

The filter.ini configuration file lets you set up rules for filtering attributes into and out of RADIUS packets.

Filter Rules

Each filter in the filter.ini file consists of the filter name in square brackets ([name]) followed by the rules for that filter.

Each rule takes one of the following three forms: keyword attribute value
keyword attribute
keyword

Table 62 lists valid syntax combinations.

Table 63: Filter Syntax

filter.ini Rule Syntax	Function
ALLOW	This keyword by itself specifies that all attributes, regardless of value, are to be allowed in the packet.
ALLOW attribute	This rule specifies that this attribute is allowed in the packet, regardless of its value.
ALLOW attribute value	The rule lists a specific attribute/value pair to allow in the packet.
EXCLUDE	The keyword by itself specifies that all attributes, regardless of value, are to be excluded from the packet. EXCLUDE is the default action for a filter.
EXCLUDE attribute	The rule specifies that this attribute is excluded from the packet, regardless of its value.
EXCLUDE attribute value	The rule specifies an attribute/value pair to exclude from the packet.
ADD attribute value	The rule lists a specific attribute/value pair to add to the packet. The attribute is added after all other rules are processed.
REPLACE attr1 WITH attr2	The rule specifies that any occurrence of attr1 are replaced by attr2, which retains attr1's value.
REPLACE attr1 WITH attr2 v2	The rule specifies that any occurrence of attr1 (regardless of value) is replaced by attr2 whose value is set to v2.
REPLACE attr1 v1 WITH attr2	The rule specifies that any occurrence of attr1 whose value is v1 is replaced by attr2 (which keeps value v1).
REPLACE attr1 v1 WITH attr2 v2	The rule specifies that any occurrence of attr1 whose value is v1 is replaced by attr2 having a value v2.

An attribute is ADDed to a packet only if it is legal to do so. Some attributes can appear only once in a RADIUS packet; others can appear multiple times. If an attribute that is the subject of an ADD rule is already present in the packet (after processing ALLOW and EXCLUDE rules) and the attribute can only appear once, the ADD rule is not processed and the second instance of the attribute is not added.

The Steel-Belted Radius dictionary file radius.dct provides string aliases for certain integer values defined in the RADIUS standard. You can use these strings in attribute filter rules.

Note: Filter rules provide you with tremendous flexibility. However, Steel-Belted Radius does notprevent you from creating an invalid RADIUS packet. Some attributes are not appropriate for certain types of requests. For example, adding a pooled Framed-Ip-Address attribute to an accounting request could cause a loss of available IP addresses.

Order of Filter Rules

The order of rules is important. General default rules that take no parameters, such as ALLOW (allow all attributes unless otherwise specified) or EXCLUDE (exclude all attributes unless otherwise specified) must appear as the first rule in the filter. Later rules supersede earlier rules; the last applicable rule "wins." ADD and REPLACE rules are applied after the ALLOW and EXCLUDE rules.

More specific rules with more parameters (ADD attribute value) act as exceptions to less specific rules with fewer parameters (ALLOW attribute, EXCLUDE). For example, you might want to ALLOW a certain attribute and EXCLUDE one or more specific values for that attribute. Or you might EXCLUDE all attributes, ALLOW specific attributes, and ADD specific attribute/value pairs.

You can use two basic approaches to designing a filter:

- Start the rule list with a default EXCLUDE rule (no parameters) and add ALLOW rules for any attributes or attribute/value pairs that you want to insert into the packet. ADD and REPLACE rules may be used.
- Start the rule list with a default ALLOW rule (no parameters) and add EXCLUDE rules for any attributes or attribute/value pairs that you want to remove from the packet. ADD and REPLACE rules may be used.

The default action for filter.ini is EXCLUDE. If a filter does not contain any rules, the filter removes all attributes from a packet when the filter is applied.

Values in Filter Rules

The value of an attribute is interpreted based on the type of the attribute in its attribute dictionary. Table 63 lists the meaning of each attribute type.

Table 64: Filter Rule Values

Attribute Type	Function
hexadecimal	A hexadecimal value is specified as a string. Special characters may be included using escape codes.
int1, int4, integer	1- or 4-byte unsigned decimal number (integer is equivalent to int4). Note: The Steel-Belted Radius dictionary file radius.dct provides string aliases for certain integer values defined in the RADIUS standard. You can use these strings in attribute filter rules.
ipaddr, ipaddr-pool	An IP address in dotted notation; for example: EXCLUDE NAS-IP-Address 127.0.0.1

Attribute Type	Function
ipxaddr-pool	A sequence of hex digits; for example: ALLOW Framed-IPX-Network 0042A36B
string	String attribute (includes null terminator). A string is specified as text. The text may be enclosed in double-quotes (“). The text is interpreted as a regular expression. Backslash (\) is the escape character. Escape codes are interpreted as follows: Code Meaning \a 7 \b 8 \f 12 \n 10 Code Meaning \r 13 \t 9 \v 11 \nnn nnn is a decimal value between 0 and 255 \xnn nn is a hexadecimal value between 00 and FF \c c is a single character, interpreted literally Literal backslashes (\) within a string and double-quotes (“) within quoted strings should be prefixed with an escape character. For example: ADD Reply-Message Session limit is one hour ADD Reply-Message “Session limit is one hour” ADD Reply-Message “Your user name is \”George\””
time	A time value is specified with a string indicating date and time: yyyy/mm/dd hh:mm:ss The date portion is mandatory; the time portion may be specified to whatever degree of precision is required, or may be omitted entirely. For example: 2006/4/3 14:00:00 and 2006/4/3 14 both refer to April 3, 2006 at 2:00 p.m. For example: ADD Ascend-PW-Expiration 2006/4/3

Referencing Attribute Filters

Steel-Belted Radius attribute filtering provides flexibility in packet processing. You can use the same filter for all packets in all realms. You can apply filtering to some realms, and not others. (To disable filtering for a realm, omit filtering parameters from the *.pro, *.dir, peapauth.aut, or ttlsauth.aut file.) Filtering is often used only for packets that are routed "out" to realms (the FilterOut parameter).

To reference the filtering rules defined in the filter.ini file in proxy or directed realm configurations, you must use the FilterOut and FilterIn parameters in the [Auth] and [Acct] sections of a RADIUS realm configuration file.

The full syntax used is:

```
[Auth]
FilterIn=name1
FilterOut=name2
[Acct]
FilterIn=name3
FilterOut=name4
```

where name1, name2, and so forth provide the names of filters, sections in the filter.ini file called [name1], [name2], and so forth. The name values in this syntax are completely independent of each other. They may be all the same, all different, or some combination of same and different.

When using the FilterIn and FilterOut parameters in the [Auth] and [Acct] sections, be sure to use the filter name without the square brackets ("name", not "[name]").

Note: If a [name] section is not found in the filter.ini file, it is equivalent to assigning a filter that EXCLUDEs all attributes. In other words, assigning a filter name that cannot be found causes the final packet to be emptied of all attributes.

Note: Do not allocate IP addresses from Steel-Belted Radius IP address pools in accounting filters. These addresses will be allocated but never released.

sample.rr File

Used by: GEE
Not used by: EE

Attribute value pools allow Steel-Belted Radius to assign and return attribute sets dynamically when an Authorization Request is processed. This functionality is supported by the use of a vendor-specific attribute (VSA) called Funk-Round-Robin-Group. The value for this attribute is a string, and should be set to the name of a .rr suffix file that defines an attribute value pool. This value can therefore be set for a user or profile by using
the SBR Administrator or the LDAP Configuration Interface (LCI), or by any other return list mechanism (such as databaseretrieval).

Attribute value pooling allows for a dynamic allocation of attribute values sets, so that attributes needed to configure changeable and complex situations do not have to be assigned in static profiles. This functionality is supported by the use of a vendor-specific attribute called Funk-Round-Robin-Group. The

value for this attribute is a string, and should be set to the name of a .rr suffix file that defines an attribute value pool.

A .rr file is defined as follows:

```
[Sets]
SetName1        =
Weight1
SetName2        =
Weight2
...
[SetN
ame1
]
AttributeName1.1        =       AttributeValue1.1
AttributeName1.2 = AttributeValue1.2
...
```

Steel-Belted Radius maintains "round-robin" statistics for each attribute value pool so that weight calculations can be performed properly. When a user who belongs to a profile that has been assigned to a particular attribute value pool logs in, the round-robin values are incremented to determine which Attribute Value set should be assigned to the user. This attribute set is added to the return list of the Access-Accept.

Attribute value pooling can be used in several ways. For example, the Acme Company wants off-site employees to be able to establish tunnels to the company network. The Acme Company maintains three tunnel connection endpoints to which end users can create VPNs into the corporate network, each of these with different capacities. The company would define an attribute value pool of three attribute sets, each describing how to establish a tunnel with one of these connection points. These attribute sets should be weighted according to the capacity of the three connection points. Figure 3 illustrates a sample acme.rr file.

Figure 3: Sample *.rr file (acme.rr)

```
;acme.rr
[Sets]
VPN1=20
VNP2=12
VPN3=7
[VPN1]
Tunnel-Server-Endpoint = 8.4.2.1
Tunnel-Password = GoodGuess
[VPN2]
Tunnel-Server-Endpoint = 8.4.2.2
Tunnel-Password = BestGuess
[VPN3]
Tunnel-Server-Endpoint = 8.4.2.4
Tunnel-Password = OurSecret
```

To make this attribute value pool visible, the Acme Company would define a Funk-Round-Robin-Group VSA and assign it to the users (or the profile assigned to these users) and make the value of the VSA point to the acme.rr file shown in Figure 3.

```
Funk-Round-Robin-Group = acme.rr
```

Refer to the Steel-Belted Radius Administration Guide for more information on using attribute value pooling.

spi.ini File

Used by: GEE, EE
Not Used By: —

The spi.ini initialization file defines encryption keys and identifies the servers from which Steel-Belted Radius processes encrypted Class attributes in accounting requests. The spi.ini file allows one Steel-Belted Radius server to decode accounting requests for sessions that were authenticated on a different Steel-Belted Radius server. Class attributes received from servers not specified in spi.ini are ignored.

All Steel-Belted Radius servers that may receive authentication and accounting requests from a common network access device must be configured with similar spi.ini files, which must list the IP addresses of all the servers in that "cluster." This allows one server to authenticate a user and generate an encrypted Class attribute that can be decrypted and processed by any other server in the cluster.

[Keys] Section

The [Keys] section of spi.ini specifies the list of encryption keys used to encode subattributes encapsulated within Class attributes.

```
[Keys]
CurrentK
ey = n 1
= value
2
=
val
ue
M
```

Table 65: spi.ini [Keys] Syntax

Parameter	Function
CurrentKey	Specifies the encryption key that is currently active, where n is 0 or the number of a key listed in the [Keys] section: • 0 – Generate and use a unique random key to encrypt Class attributes. Used only when the Steel-Belted Radius server does not exchange encrypted Class attributes with other servers. • n – Use the specified key to encrypt Class attributes. Default value is 0.
n = value	Specifies the number and value of the encryption key.

In the following example, the Steel-Belted Radius server generates a unique random key to encrypt Class attributes.

```
[Keys]
CurrentKe
y = 0
```

In the following example, the second key (swordfish) is currently active and used to encrypt Class attributes. The other keys in this section can be used to decrypt Class attributes received from other servers in the same cluster.

[Hosts] Section

The [Hosts] section of spi.ini identifies the IP address of servers from which received Class attributes are parsed for encapsulated/encrypted subattributes. Class attributes from servers not identified in the [Hosts] section of spi.ini are passed without special processing.

The information in the [Hosts] section is used to compute the server's identifier, which is included in the Class attribute. If one of a host's interfaces is included in the [Hosts] section, that interface is used to compute the server identifier. If more than one interface for a host is listed, the IP address of the last interface listed is used. If no matching address is found, the host's primary IP address is used. Addresses not corresponding to a host interface are used to configure the collection of other servers whose Class attributes are accepted.

In the following example, three servers are identified as belonging to a cluster.

```
[Hosts]
192.168.1
5.21
192.168.23.121
192.168.23.205
```

vendor.ini File

Used by: GEE, EE
Not Used By: —

The vendor.ini initialization file contains information that allows Steel-Belted Radius to work with the products of othervendors.

[Vendor-ProductIdentification] Section

The [Vendor-Product Identification] section of vendor.ini (Table 65) identifies and provides information about the network access devices that can be used with Steel-Belted Radius.

Table 66: vendor.ini [Vendor-Product Identification] Syntax

Parameter	Function
vendor-product	Specifies the name of the product. A product name must be unique, cannot include blanks and must consist of 31 or fewer characters. These product names are used only in the Make/model list in the RADIUS Clients panel. This list is used when adding a new RADIUS client or when selecting a vendor-specific attribute.
dictionary	Specifies the dictionary file to use for this product. The dictionary file must be located in the same directory as the Steel-Belted Radius daemon or service. You do not need to specify an extension on the dictionary name; Steel-Belted Radius automaically attaches an extension of .DCT to the dictionary names listed in this parameter.
call-filter-attribute	Specifies the attribute used for call filter functions. Used only by Ascend/Lucent network access devices.
challenge-response- attribute	Specifies the attribute number in which a network access device sends responses to challenge sequences. If not specified, the default behavior is to expect responses to be encoded in the User-Password attribute.
data-filter-attribute	Specifies the attribute used for data filter functionality. Used only by Ascend/Lucent network access devices.
discard-after	Used for inbound proxy RADIUS servers that send username information in a "decorated" format. For example, if a proxy RADIUS server sends usernames of the form

Parameter	Function
	username@company, then specifying @ results in the @ delimiter character and all text after the @ delimiter character being discarded for authentication purposes; the string username is used.
discard-before	Used for inbound proxy RADIUS servers that send username information in a "decorated" format. For example, if a proxy RADIUS server sends usernames of the form company$username, then specifying $ results in the $ delimiter character and all text before the $ delimiter character being discarded for authentication purposes; the string username is used.
help-id	Help context for the vendor's product in the vendor information help file.
ignore-acct-ss	If set to Yes, the digital signature of accounting packets based on the shared secret is ignored. This accommodates devices that do not properly sign accounting packages. Default value is No.
ignore-ports	Determines whether Steel-Belted Radius may infer that one user has logged off if the port that was assigned to that user is now being used by another user. • If set to No, an inference is made and the previous user is removed from the Active Users list. • If set to Yes, no inference is made and both users are deemed active. Default value is No.
max-eap-fragment	Specifies a maximum EAP fragment length on a make/model basis. The maximum EAP fragment length emitted by TLS or TTLS is the lesser of the maximum specified in their .eap/.aut files and this setting. Default value is 1020. This may be inefficient, however, as the fragment length must be set to a number low enough to work with all of a customer's Access Points.
port-number-usage	• If set to per-port-type, entries in the Active List containing duplicate port numbers and port types are deleted. • If set to unique, entries in the Active List containing duplicate port numbers are deleted; port type information is ignored. Default value is per-port-type.
product-scan-acct	Specifies the name of the section in the vendor.ini file that contains rules for dynamically determining the product associated with an accounting request by the contents of the request packet.
product-scan-auth	Specifies the name of the section in the vendor.ini file that contains rules for dynamically determining the product associated with an authentication request by the contents of the request packet.
send-class-attribute	If set to No, the Class attribute is not sent to the client on Access-Accept. (This feature is designed to accommodate devices that don't handle the Class attribute properly.) Default value is Yes.
send-session-timeout-on-challenge	• If set to Yes, the Session-Timeout attribute is sent to the client on Access-Challenge responses that include EAP messages. This attribute advises a network access device on how long it should wait for a user response to the challenge. • If set to No, the Session-Timeout attribute is not sent to the client on Access-Challenge responses that include EAP messages. Default value is Yes.

Product-ScanSettings

Used by: GEE
Not used by: EE

After you define a Vendor-Product entry in vendor.ini, the name of this entry can be selected in the RADIUS Clients window as a possible value for the Make/model field. The Product-Scan-Auth and Product-Scan-Acct settings can be used within a Vendor-Product entry to permit dynamic make/model selection to occur. These settings enable Steel-Belted Radius to examine the incoming packet to determine the make/model of the network access device that originated the packet.

A dynamic Vendor-Product entry might appear as follows:

```
Vendor-Product =
DeviceNameInRASClientsList Product-
Scan-Auth = MakeModelSelect
Product-Scan-Acct = MakeModelForAccounting
[MakeModelForAuthentication]
Product =
String
Product =
String
.
.
Product =
[MakeModelForAccount
ing] Product = String
Product = String
.
.
.
Product =
```

Table 67: vendor.ini Product-Scan Syntax

Parameter	Function
Vendor-Product	Creates a label that appears as a selection in the Make/Model list in the RADIUS Clients window of the SBR Administrator.
Product-Scan-Auth=name	Applies only to authentication servers. name references a section heading that appears elsewhere in vendor.ini.
[name]	Provides rules that govern dynamic make/model selection. These rules apply on authentication requests if the value name is assigned to Product-Scan-Auth; they apply on accounting requests if the value name is assigned to Product-Scan-Acct.
Product=String . . . Product=	Product is a product name. String is a regular expression to match against attributes in the packet. Character by character, Product must match a Vendor-Product value defined elsewhere in the vendor.ini file. The default vendor.ini provided with Steel-Belted Radius includes a number of Vendor-Product values from which you may choose. Each value corresponds to a vendor-specific RADIUS attribute dictionary. The list of product names and strings is tried in order. If the packet does not come from the first device, the next is tried, and so on until the last entry in the list is tried. You can set up a default at the end of the list by making sure the last Product entry in the list has no String assigned. If no match is found earlier in the list, Steel-Belted Radius assumes that the packet comes from the type of device specified in the final entry.

The following example would be appropriate in a configuration whose RASs were mostly Ascend devices:

```
Product-Scan-Auth = Bigco Special Scan
.
.
.
[Bigco Special Scan]
Ascend MAX Family=
\x2c?
Nortel Versalar Remote Access Concentrator=
\x1a?\x00\x00\x06\x30
US Robotics NETServer = \x1a?\x00\x00\x01\xad
Ascend MAX Family =
```

The preceding example sets up dynamic make/model selection for authentication and states that the identity of the client device should be determined by seeking matches in the following order:

1. Is the attribute with identifier number 0x2c (Acct-Session-Id), with a value of any length (indicated by the question mark character), found in the incoming authentication packet? If so, the originating network access device is a member of the Ascend MAX Family; use that vendor-specific dictionary.

2. Is the vendor-specific attribute with identifier number 0x1a (Vendor-Id), with a value of any length (indicated by the question mark character), present in the packet? If so, does it have the value 1584 (0x630) which indicates a Nortel Networks Versalar RAC? If so, use that vendor-specific dictionary (provided with Steel-Belted Radius).
3. Is the Vendor-Id attribute present, with any length, and if so, does it have the value 429 (0x1ad) which indicates a US Robotics NETServer? If so, use that vendor-specific dictionary (provided with Steel- Belted Radius).
4. If no match can be found using the rules specified in this section, then use the vendor-specific dictionary for the Ascend MAX Family.

Note: When auto-restart is enabled, and the server is running normally, you typically see two instances of radius.exe in any tool (such as the Task Manager) that you use to monitor processes on the Windows host computer.

Chapter 5

Address Assignment Files

This chapter describes the usage and settings for the Steel-Belted Radius initialization (.ini) files that are used to enable, disable, and configure IP address assignment, which is available for the GEE version of Steel- Belted Radius.

dhcp.ini File

pool.dhc Files

dhcp.ini File

Used by: GEE
Not used by: EE

The dhcp.ini configuration file configures DHCP address pools so that IPv4 addresses can be assigned from a backend DHCP server, rather than from a standard Steel-Belted Radius IP address pool.

Note: Steel-Belted Radius does not support DHCP allocation of IPv6 addresses.

[Settings] Section

The [Settings] section of the dhcp.ini file (Table 67) controls DHCP address allocation.

Table 68: dhcp.ini [Settings] Syntax

Parameter	Function
Enable	• If set to 1, DHCP address allocation is enabled. • If set to 0, DHCP address allocation is disabled. Default value is 0.
Attempts	Specifies the number of times a DHCP DISCOVER or REQUEST message is sent if no response is received. Default value is 3.
AttemptTimeout	Specifies the waiting period, in seconds, for a response to a DISCOVER or REQUEST message, before resending the message. Default value is 5 seconds.
OverallTimeout	Specifies the number of seconds for acquiring an IP address before DHCP address assignment is presumed to have failed. This timeout applies only to the DISCOVER/ REQUEST sequence used to acquire an address initially, not to address renewal or release. Default value is 15 seconds. Note: While the timeout for the individual DISCOVER and REQUEST transactions is

Parameter	Function
	specified by Attempts and AttemptTimeout, Overall Timeout specifies the timeout for the entire sequence.
htype	Specifies the client hardware type (0–255). This parameter is typically omitted, because the value is generated automatically.
Hlent	Specifies the length of the client hardware address (1–16). This parameter is typically omitted, because the value is generated automatically.
Chaddr-prefix	Specifies the string that identifies the initial bytes of the client hardware address (chaddr). This string can include escape codes, including \nnn for decimal values and \ xnn for hex values. This parameter is typically omitted, because the value is generated automatically.
ServerPort	Specifies the UDP port number on which the DHCP server(s) listen. This setting should be specified only for non-standard DHCP configurations. Default value is 67, which is the standard DHCP server port.
LocalPort	Specifies the UDP port number that Steel-Belted Radius, acting as a relay agent, uses during DHCP communication. This setting should be specified for only non-standard DHCP configurations. Default value is 67, which is the standard DHCP server port.
Pad	Specifies the minimum number of bytes for a DHCP request message. Messages smaller than this number are padded with 0s. Certain DHCP servers discard messages smaller than a certain value. This option allows interoperability with such servers. Default value is 300.

The following is a sample dhcp.ini file:

```
[Setti
ngs]
Enabl
e = 1
Attempts = 3
AttemptTimeout = 2
OverallTimeout = 10
```

[Pools] Section

The [Pools] section lists all DHCP pool names (specified in the pool.dhc file) in the following format:

```
[P
ool
s]
po
ol
1
```

po
ol 2
For
examp
le:

[Pools]
DHCP_SERVER1
DHCP_SERVER_SA
LES

pool.dhc Files

Each pool listed in the [Pools] section of the dhcp.ini file must be a corresponding pool.dhc file that configures that pool.

[Settings] Section

Table 69: pool.dhc [Settings] Syntax

Parameter	Function
LeaseTime	Set to the lease time, in seconds, to request from the DHCP server. Default value is 1 day.
MinLeaseTime	Set to the minimum lease time, in seconds. Offers from DHCP servers with lease time less than this minimum are ignored. Default value is the value set for LeaseTime.
TargetAddress	Set to the address to which DISCOVER messages are sent. Default value is 255.255.255.255, the local broadcast address. This entry should normally remain unchanged, to allow DHCP DISCOVER messages to be broadcast.

[Request] Section

The [Request] section allows options in the DHCP DISCOVER and REQUEST messages to be constructed from attributes in the RADIUS Access-Request and from

pre-configured literal values in the following way:

[Request]
DHCP option = RADIUS attribute or literal value DCHP option = RADIUS attribute or literal value
.
.
.

The DHCP option contains of the following fields (brackets ([ ]) indicate optional text). Fields are not separated by spaces.

[vendor-specific]option [offset] format

Table 70: pool.dhc [Request] Syntax

Parameter	Function
vendor-specific	Set to v if this is a vendor-specific option, or omit otherwise.
option	option
offset	Set to a period followed by the number of bytes into the option where the value is located, or a plus-sign (+) to indicate a list of values in the DHCP option – each to be mapped to an instance of the RADIUS attribute.
format	Set to the format of the DHCP option, which can be one of the following: n32 a 32-bit integer n16 16-bit integer n8 8-bit integer s or string string i or ip IP address

The following are examples of DCHP option fields:

- 1ip (The "Subnet Mask" option as an IP address)
- 3+ip (The "Router" option as a list of IP address, each to be mapped to an instance of the RADIUS attribute)
- 6.4ip (The "DNS Server" option as a second IP address in list (each IP address is 4 bytes))
- 12s (The "Host Name" as a string)

The RADIUS attribute can be set to the name of any attribute defined in any dictionary. A literal value can be specified instead of a RADIUS attribute. This value must be text enclosed in double-quotes ("").

The string is interpreted based on the format of the DHCP option:

- IP addresses must be specified in dotted notation; for example, 127.0.0.1 for IPv4 networks.
- Integers are expressed in decimal format; for example, 100.
- Strings are expressed as any text sequence.

The text can include escape sequences, where the backslash character (\) is the escape character. Table 70 lists escape sequences.

Table 71: Escape Sequences

Parameter	Function
\a	7

Parameter	Function
\b	8
\f	12
\n	10
\r	13
\t	9
\y	11
\nnn	A decimal value between 0 and 255.
\xnn	A hexadecimal value between 00 and FF
\\	A literal backslash \
\"	A double-quote
\char	A single character, interpreted literally

Note: You must use an escape character to include a literal backslash (\) or double-quote (") in the string.

An escape sequence can be used to set an option to an arbitrary binary value. This is useful, for example, when setting the Vendor Class Identifier option (60).

The following example sets the DHCP Host Name option to the RADIUS Calling-Station-Id, and sets the DHCP Vendor Class Identifier option to a binary string:

```
[Request]
12s = Calling-Station-Id
60s = "\x01\x02\x03\x04\x05"
```

[Reply] Section

The [Reply] section allows RADIUS Access-Accept attributes to be constructed from options the DHCP server returns in an ACK message, in the following way:

```
[Reply]
RADIUS attribute = DHCP
option RADIUS attribute =
DHCP option
.
.
.
```

See the [Request] section for information on how to specify the RADIUS attribute and the DHCP option values.

Note: In contrast to the [Request] section, the left and right sides of the equal sign are reversed to account for the direction in which the data is being set.

The following example returns the RADIUS Framed-IP-Netmask attribute from the DHCP Subnet Mask option and sets the RADIUS Framed-MTU attribute from the DHCP Interface MTU option:

```
[Reply]
Framed-IP-
Netmask    =    1ip
Framed-MTU      =
26n16
```

ReconfiguringPools

DHCP pool information is loaded at startup from the dhcp.ini file and all associated pool.dhc files. DHCP pools can be added, deleted, and modified dynamically by doing the following:

1. Modify the dhcp.ini file and the pool.dhc files as required.

2. Restart the RADIUS process/service:

   - Linux: Issue the HUP signal to the Steel-Belted Radius process. kill -HUP ProcessID
   - Windows: Run RADHUP.EXE from the command shell.

Steel-Belted Radius reads the modified files and configures its DHCP pools

Chapter 6

Accounting Configuration Files

This chapter describes the usage and settings for the Steel-Belted Radius accounting initialization (.ini) files, which enable, disable, and configure accounting features of the server. Initialization files are loaded at startup time, and reside in the Steel-Belted Radius directory.

account.ini File

Used by: GEE, EE*
Not Used By: —

The account.ini file contains information that controls how RADIUS accounting attributes are logged to a comma- delimited text file by Steel-Belted Radius. Specifically, the account.ini file controls file creation settings, such as
file creation frequency, maximum size, and default directory, and file content, such as what information is recorded for each received accounting request.

[Alias/name] Sections

The [Alias/name] sections of account.ini are used to associate attributes of different names, but identical meaning. For example, one network access device vendor might call an attribute Acct-Octet-Pkt and another might call it Acct-Oct-Packets, yet the two attributes mean the same thing.

Each [Alias/name] section permits you to map one RADIUS accounting attribute that is already being logged by Steel-Belted Radius to any number of other attributes. You can provide as many [Alias/name] sections as you want, using the following syntax for each section:

```
[Alias/name]
VendorSpecificAttrib
ute=
VendorSpecificAttribute=
.
.
```

Table 72: account.ini [Alias|name] Syntax

Parameter	Function
name	The preferred attribute name. The name attribute must be one that you are currently logging to a column in the Steel-Belted Radius accounting log file (.act). Therefore, it must be listed in the [Attributes] section of account.ini.
VendorSpecificAttribute	Each entry is given on one line. An equal sign (=) must immediately follow each VSA name, without any intervening space. Improperly formatted entries are considered invalid and are ignored.

Each VendorSpecificAttribute in the list is logged to the name column in the accounting log file. Because you are listing these attributes in an [Alias/name] section, verify they are not listed in the [Attributes] section, or they will be logged to their own columns as well as the name column.

All of the attribute names that you reference in an [Alias/name] section must be defined in a dictionary file that is already installed on the Steel-Belted Radius server. This includes name and each VendorSpecificAttribute entry.

In the following example, the standard RADIUS attribute Acct-Octet-Packets is mapped to the vendor-specific attributes Acct-Octet-Pkt and Acct-Oct-Packets. Values encountered for all three attributes are logged in the
Acct-Octet-Packets column in the accounting log file:

```
[Alias/Acct-Octet-
Packets]        Acct-
Octet-Pkt=
Acct-Oct-Packets=
```

[Attributes] Section

The [Attributes] section of the account.ini file lists all the attributes logged for each received accounting request in the accounting log file. When you install Steel-Belted Radius, the account.ini file is set up so that all standard RADIUS attributes and all supported vendors' accounting attributes are listed.

You can change the order of columns in the accounting log file by rearranging the sequence of attributes in the [Attributes] section. You can delete or comment out any attributes that are not relevant to your billing system or which do not apply to the equipment that you are using. This lets you design the content and column order of any spreadsheets that you plan to create based upon the accounting log file.

The syntax is as follows:

```
[Attributes]
AttributeName=
AttributeName=
```

For example:

```
[Attrib
utes]
User-
Name
=
NAS-
Port=
Framed-IP-
Address=
Acct-Status-
Type=    Acct-
Delay-Time=
Acct-Session-
Id=
```

The [Attributes] section lists one AttributeName on each line. You must ensure that an equal sign (=) immediately follows each AttributeName, with no spaces in between. Improperly formatted entries are considered invalid and are ignored.

Each AttributeName in the [Attributes] section must be defined in a standard RADIUS dictionary file or a vendor- specific dictionary file on the Steel-Belted Radius server.

Note: The first six attributes in each log file entry (Date, Time, RAS-Client, Record-Type, Full-Name, and Auth-Type) are always enabled, and cannot be re-ordered or deleted. Therefore, these attributes do not appear in the account.ini file [Attributes] section.

[Configuration] Section

Used by: GEE
Not used by: EE

Table 73: account.ini [Configuration] Syntax

Parameter	Function
LogDir	Sets the destination directory on the local host where accounting log files are stored. Default value is the Steel-Belted Radius directory. Note: You cannot write accounting log files to a mapped or shared drive. Note: With directed realms, you can maintain separate accounting log locations for each realm.

[Settings] Section

Steel-Belted Radius writes all accounting data to the current accounting log file (.act) until that log file is closed. After closing the file, Steel-Belted Radius opens a new one and begins writing accounting data to it. You can configure how often this rollover of the accounting log file occurs.

The naming conventions for accounting log files permit more than one file to be generated during a day. Table 73 lists the file naming conventions used for different rollover periods. In the examples below, y=year digit, M=month digit, d=day digit, h=hour digit, and m=minute digit. When more than one file is generated during a day, the sequence number _nnnnn starts at _00000 each day.

Table 74: Accounting File Rollover

File Generation Method	File Naming Convention
Default (24 hours)	yyyyMMdd.act
Non-24-hour rollover	yyyyMMdd_hhmm.act
Rollover due to size	yyyyMMdd_nnnnn.act
Rollover due to size or startup when non-24-hour time in effect	yyyyMMdd_hhmm_nnnnn.act

The [Settings] section of the account.ini file (Table 73) controls how entries are written to the accounting log file, and ensures the compatibility of these entries with a variety of database systems.

Table 75: account.ini [Settings] Syntax

File Generation Method	File Naming Convention
BufferSize	The size of the buffer used in the accounting logging process, in bytes. Default value is 131072 bytes.
Carryover	• If set to 1, each time a new accounting log file is created, a start record for each session that is currently active is written to the file. • If set to 0, the list is not written. Default value is 1.
Enable	• If set to 1, the accounting log feature is enabled. • If set to 0, no .act files are created on this server. Accounting servers should have Enable set to 1; for efficiency, non-accounting servers should have Enable set to 0. Default value is 1.
LineSize	Number in the range 1024–32768 that specifies the maximum size of a single accounting log line. Default value is 4096.
LogFilePermissions (Linux only)	Specifies the owner and access permission setting for the accounting log file. Enter a value for the LogFilePermissions setting in owner: group permissions format, where: • owner specifies the owner of the file in text or numeric format. • group specifies the group setting for the file in text or numeric format. • permissions specifies what privileges can be exercised by Owner/Group/Other with respect to the file in text or numeric format. For example, ralphw:1007 rw-r----- specifies that the file owner (ralphw) can read and edit the log file, members of group 1007 can read (but not edit) the log file, and that other users cannot access the log file.
MaxSize	The maximum size of an accounting log file, in bytes. If the accounting log file reaches or exceeds this size when it is checked, the log file is closed and a new file started. A value of 0 (the default) means unlimited size. Note: Because the size of the log file is checked once per minute, the log file can exceed the maximum size specified in this parameter.
QuoteBinary	• If set to 1, binary values written to the accounting log file are enclosed in quotes. • If set to 0, quotes are not used. Set this value according to the format expected by the accounting application that processes the entries. Default value is 1.

File Generation Method	File Naming Convention
QuoteInteger	• If set to 1, integer values written to the accounting log file are enclosed in quotes. • If set to 0, quotes are not used. Set this value according to the format expected by the accounting application that processes the entries. Default value is 1.
QuoteIPAddress	• If set to 1, IP addresses written to the accounting log file are enclosed in quotes. • If set to 0, quotes are not used. Set this value according to the format expected by the accounting application that processes the entries. Default value is 1.
QuoteText	• If set to 1, text strings written to the accounting log file are enclosed in quotes. • If set to 0, quotes are not used. Set this value according to the format expected by the accounting application that processes the entries. Default value is 1.
QuoteTime	• If set to 1, time and date values written to the accounting log file are enclosed in quotes. • If set to 0, quotes are not used. Set this value according to the format expected by the accounting application that processes the entries. Default value is 1.
RollOver	Specifies how often the current accounting log file is closed and a new file opened (a rollover), up to one rollover per minute. Non-zero values indicate the number of minutes until the next rollover. If set to 0, the accounting log file rolls over once every 24 hours, at midnight local time. Default value is 0.
RollOverOnStartup	• If set to 1, each time Steel-Belted Radius is started, it closes the current accounting log file and opens a new one. A sequence number _nnnnn is appended to the log file name, just as when MaxSize is reached. • If set to 0, each time Steel-Belted Radius is started, it appends entries to the previously open accounting log file. Default value is 0.
Titles	• If set to 1, each time a new accounting log file is created, the titleline (containing column headings) is written to the file. • If set to 0, the line is not written. Default value is 1.
UTC	• If set to 1, time and date values are provided according to Universal Time Coordinates (UTC, formerly known as Greenwich Mean Time or GMT). • If set to 0, time and date values reflect local time. Default value is 0.

[TypeNames] Section

Each entry in the [TypeNames] section of account.ini maps a possible value of the Acct-Status-Type attribute to a string. The value of this attribute is written into the fourth column of each accounting log record. The syntax is as follows:

```
[TypeNames]
TypeID      =
TypeName
TypeID      =
TypeName
.
.
.
```

Table 76: account.ini [TypeNames] Syntax

Parameter	Function
TypeID	Each TypeID is a numeric value that corresponds to a possible value of the Acct-Status-Type attribute. This attribute appears in every incoming RADIUS accounting packet to identify the types of data it is likely to contain.
TypeName	Each TypeName value is a string. This string is written to the accounting log to identify the type of packet.

The standard Acct-Status-Type values 1, 2, 3, 7, and 8 are already listed in the [TypeNames] section of account.ini as follows:

```
[Type
Name
s]
1=Sta
rt
2=Sto
p
3=Int
erim
7=On
8=Off
```

You can edit the [TypeNames] section to add vendor-specific packet types to this list, which makes your accounting log files easier to read and use. For example:

```
[Type
Name
s]
1=Sta
rt
2=Sto
p
```

3=Int
erim
7=On
8=Off
639=Asce
ndType
28=3Com
Type

If no string is given for a particular Acct-Status-Type, Steel-Belted Radius uses the numeric value of the incoming Acct-Status-Type attribute, formatted as a string.

Chapter 7

Realm Configuration Files

This chapter describes the configuration files relating to proxy and directed realm administration in Steel-Belted Radius.

Table 76 lists the files in the Steel-Belted Radius directory you must edit to configure realms.

Table 77: Realm Configuration Files

File Name	Purpose
radius.ini	Enables and disables realm features.
proxy.ini	Specifies the order of realm selection methods, the realm selection rules, and other settings for all realms on the server.
RealmName.pro (GEE only)	For each proxy realm that you want to configure on the Steel-Belted Radius server, you must create a file called RealmName.pro, where RealmName is the name of the realm, and you must register this RealmName by listing it in the [Realms] section of the proxy.ini file.
RealmName.dir (GEE only)	For each directed authentication and/or accounting realm that you want to configure on the Steel-Belted Radius server, you must create a file called RealmName.dir, where RealmName is the name of the realm, and you must register this RealmName by listing it in the [Directed] section of proxy.ini.
filter.ini	Stores filters for RADIUS attributes; these filters may be referenced from the [Auth] or [Acct] section of a RealmName.pro or RealmName.dir file. Note: Do not edit the filter.ini file manually. Use the SBR Administrator to configure rules for filtering RADIUS attributes.

Proxy Realm Configuration Files

Used by: GEE
Not used by: EE

This section describes how to set up the proxy realm configuration files.

Sample radius.ini Realm Settings

The following settings in your radius.ini file enables the realm feature and the attribute filtering feature. These two features must be enabled for the sample proxy realm configuration files to work:

```
[Configuration]
ExtendedProxy=1
AttributeEdit=1
```

Note: For radius.ini syntax details, see "[Configuration] Section".

Examples

The following proxy.ini file registers a proxy realm called sample.com and adds that realm to the list of target realms for static proxy accounting.

```
[Realms]
sample.com
[StaticAcct]
7=CustAOnOf
f
8=CustAOnOf
f
[CustAOnOff]
realm=sample.
com
```

Note: For syntax details, see "proxy.ini File".

The following proxy.ini file entry specifies that otto@rtt.other.com and carol@3g.other.com would both map to the other.com proxy realm.

```
[Realms]
other.com = *.other.com
```

The following proxy.ini file specifies that otto@rtt.other.com and carol@3g.other.com would map to the other. com proxy realm and that caitlin@groton.other.com would map to the groton.other.com proxy realm.

```
[Realms]
other.com=*.othe
r.com
groton.other.com
```

Sample Proxy RADIUS (.pro) File

The following complete file must be called sample.com.pro for it to work with the sample proxy.ini file.

```
[Auth
]
Enabl
e = 1
TargetsSection = AuthTargets
RoundRobin = 2
StripRealm = 0
RequestTimeout = 5
NumAttempts = 3
FilterOut
= CustAOut FilterIn
=                CustAIn
```

```
MessageAuthenticator
=                    0
UseMasterDictionary
= yes

[Acct]
Enabl
e = 1
TargetsSection   =   AcctTargets
RoundRobin = 1
StripRealm = 0
RequestTimeout = 5
NumAttempts  =  3
FilterOut          =
CustAOut
;    FilterIn    =
RecordLocally
= 1
;      Block      =      1
UseMasterDictionary
=yes
[AuthTarg
ets]
bunion=1
desktop=
1
[AcctTarg
ets]
desktop
[Called-Station-ID]
8885551212
5551234
[FastFail]
MinFailure
s = 3
MinSeconds = 3
ResetSeconds = 30
```

Note: For syntax details, see "Proxy RADIUS Configuration (.pro) File"

This example expects the Steel-Belted Radius database to contain Proxy entries with target names Desktop and Bunion. These entries are required to provide the network routing information (IP address, RADIUS shared secret, and UDP ports) that allows forwarded packets to reach the target servers at the customer site.

Sample filter.ini File

The following complete sample filter.ini file defines the two attribute filters referenced in the sample.com.pro file.

```
[Cust
AOut]
ALLO
W
EXCLUDE        NAS-IP-
Address  ADD  NAS-IP-
Address 1.2.3.4
[Cust
AIn]
EXCL
UDE
ALLOW    Session-
Timeout    ALLOW
Idle-Timeout
ALLOW    Service-Type
Framed  ADD  Service-
Type Framed
ADD Framed-IP-Address CustAPool
```

The CustAOut filter in this example is designed to be applied to request packets coming into the Steel-Belted Radius server that are directed out to the realm. It allows all of the attributes in the packet to go out to the realm, with the exception of the RADIUS client's IP address. It replaces this IP address with the specific "dummy" address 1.2.3.4. This filter enhances overall security by not publishing routing information to the network when it's not necessary to do so.

The CustAIn filter in this example is designed to be applied to response packets returning to the Steel-Belted Radius server, which are relayed, in turn, to the RADIUS client. Most attributes are excluded; however, if any timeout values are returned, they'll be allowed through. If the Service-Type attribute is present in the response and it has the value Framed (a string alias for the Service-Type integer value 2), it is allowed in the packet. Steel- Belted Radius adds the Service-Type attribute to the packet if it is not already there, and assigns it the value Framed (2).

The CustAIn filter in this example expects the Steel-Belted Radius database to contain an IP address pool entry called CustAPool, which specifies the customer's valid address ranges. If this entry is not present, the CustAIn filter fails. CustAPool is referenced in the filter's final entry, which assigns a value to the Framed-IP-Address attribute. As shown in the example, this entry causes Steel-Belted Radius to (1) add the Framed-IP-Address attribute to the packet; (2) select an available address from CustAPool, and (3) assign this value to the Framed-IP- Address attribute.

Directed Realm Configuration Files

Used by: GEE
Not used by: EE

This section discusses how to set up the directed realm configuration files.

Sample radius.ini Realm Settings

The same radius.ini excerpt works for our sample directed realm as for our sample proxy realm. The ExtendedProxy setting needs to be enabled (set to 1). The AttributeEdit setting does not apply to directed realms.

```
[Configuration]
ExtendedProxy=1
```

Note: For syntax details, see "[Configuration] Section".

Sample proxy.ini File

The following proxy.ini file registers the proxy realm called sample.com and registers a directed authentication and/or accounting realm called sample2.com. It defines several directed accounting methods, including those we plan to reference from the sample2.com.pro realm configuration file.

```
[Real
ms]
sampl
e.com

[Directed]
sample2.com

[DirectedAcctMethods]
CustBAcctSQL = c:\radius\CustomerB\theirsql.acc
CustCAcctAttributes = c:\radius\CustomerC\account.ini
CustCAcctSQLConfig = c:\radius\CustomerC\sqlacct.acc
CustDAcctSQLConfig3 = c:\radius\CustomerD\mysql.acc
```

Note: For syntax details, see "proxy.ini File".

The following proxy.ini file specifies that otto@rtt.other.com and carol@3g.other.com would both map to the other.com directed realm.

```
[Directed]
other.com = *.other.com
```

The following proxy.ini file specifies that otto@rtt.other.com and carol@3g.other.com would map to the other. com directed realm and that caitlin@groton.other.com would map to the groton.other.com directed realm.

```
[Directed]
other.com=*.othe
r.com
groton.other.com
```

Sample Directed Realm (.dir) File

The following configuration file must be called sample2.com.dir for it to work with the sample radius.ini file and proxy.ini file.

```
[Auth
]
Enabl
e = 1
StripRealm = 1
UseMasterDictionary
= yes [Acct]
Enable = 1
RecordLocally = 1
UseMasterDictionary = yes
[AuthMethods]
Native User
[AcctMethods
]
CustCAcctAttr
ibutes
CustCAcctSQL
Config
[Called-
Station-Id]
8885551212
55512340
```

Note: For syntax details, see "Directed Realm Configuration (.dir) File".

This sample file configures both directed authentication and directed accounting. It also strips realm routing information from the User-Name prior to authentication.

The [Acct Methods] section of this file lists the two accounting methods for the sample2.com realm. These are CustCAcctAttributes, which specifies how to log attributes to a .act accounting log file on the local server, and CustCAcctSQLConfig, which configures accounting to an external SQL database. Both methods are configured in the [DirectedAcctMethods] section of our sample proxy.ini file, above.

proxy.ini File

Used by: GEE
Not used by: EE

The proxy.ini file specifies the order of realm selection methods, the realm selection rules, and other settings for all realms on the server. Settings for a realm are provided in its RealmName.pro or RealmName.dir file. After you edit proxy.ini, you must apply your changes as follows:

- If you've configured any proxy realms, you can load your new realm configuration without stopping and restarting the server.

- Linux: Issue the HUP signal to the Steel-Belted Radius process. kill -HUP ProcessID
- Windows: Run RADHUP.EXE from the command shell. Steel-Belted Radius re-reads proxy.ini, filter.ini, and all *.pro and *.dir files in the server directory, and resets its realm configuration.

- If you've configured any directed realms and if you've added or changed:
  - Any directed accounting methods: you must stop and restart the server to load your new configuration.
  - Directed authentication methods in which external database (SQL or LDAP) authentication is used, you must stop and restart the server to load your new configuration.
  - Directed authentication methods in which local or pass-through (Native, UNIX, Domain, Host, SecurID, or TACACS+) authentication is used, you can load your realm configuration by using a HUP signal.

[Configuration] Section

The [Configuration] section of proxy.ini permits you to define prefix and suffix conventions for realm name parsing and specifies whether to use the master RADIUS dictionary to process inbound proxy responses.

You can enable prefix and suffix conventions for realm name parsing if you specify a different delimiter character for each. All prefixed name decorations must use the prefix delimiter, and all suffixed name decorations must use the suffix delimiter.

If you set the prefix and suffix delimiter to the same character, both prefix and suffix conventions are enabled, but (since suffixes are checked first) prefixes may be misinterpreted.

You should choose different delimiter characters for tunnels, proxies, and realms.

Table 78: proxy.ini [Configuration] Syntax

Parameter	Function
TypeID RealmPrefix	Specifies the character used to identify prefixed name decorations; for example, RAS1/RAS2/joeuser. Default value is /. **Note:** Enter \\ to specify the backslash character, since a single backslash in a configuration file indicates a line continuation.
RealmSuffix	Specifies the character used to identify suffixed name decorations; for example, joeuser@RAS1@RAS2. Default value is @. **Note**: Enter \\ to specify the backslash character, since a single backslash in a configuration file indicates a line continuation.
UseMasterDictionary	• If set to yes, inbound proxy responses use the master Steel-Belted Radius dictionary when attributes are filtered in.

Parameter	Function
	• If set to no, proxy responses use the client-specific dictionary when attributes are filtered in. Default value is yes. NOTE: The UseMasterDictionary setting configured in individual .dir or .pro files overrides the global setting configured in the proxy.ini file.

[Realms] Section

The [Realms] section of proxy.ini lists all of the proxy realms known to the server. The syntax is as follows:

```
[Real
ms]
Realm
Name
RealmName[=         match_rule]
RealmName   [=<undecorated>]
```

Table 79: proxy.ini [Realms] Syntax

Parameter	Function
RealmName	EEach entry must match the name of a RealmName.pro file in the same directory as proxy.ini.
= match_rule	Optional. Specifies a rule for mapping the domain information in a User-Name to a proxy realm by means of prefix or suffix wildcards.
= <undecorated>	Optional. Marker indicating the specified realm is used to process requests containing undecorated User-Name information.

[Directed] Section

The [Directed] section of proxy.ini (Table 79) lists the names of all of the directed authentication and/or accounting realms on the server. The syntax for the [Directed] section is as follows:

```
[Direc
ted]
Realm
Name
RealmName[=
match_rule]  RealmName
[=<undecorated>]  M
```

Table 80: proxy.ini [Directed] Syntax

Parameter	Function
RealmName	Each entry must match the name of a RealmName.dir file in the same directory as proxy.ini.
= match_rule	Optional. Specifies a rule for mapping the domain information in a User-Name to a directed realm by means of prefix or suffix wildcards.
= <undecorated>	Optional. Marker indicating the specified realm is used to process requests containing undecorated User-Name information.

[Processing] Section

If this section is present, it lets you specify which realm selection rules are applied and the order in which they are applied. If no [Processing] section is present, routing continues in its default behavior.

Note: If the script keyword appears in the [Processing] section, Steel-Belted Radius executes the realm selection script first, before trying other built-in methods. For more information, refer to the Steel-Belted Radius Scripting Guide.

```
[Processing]
RealmSelector
.
.
.
```

Table 81: account.ini [TypeNames] Syntax

Parameter	Function
RealmSelector	This can be one of five identifiers: Attribute-Mapping, DNIS, Prefix, Suffix, or Undecorated. Only the rules corresponding to the values listed are applied, and they are applied in the order you specify them.

The following example enables undecorated User-Names, suffix delimiters, prefix delimiters, and DNIS rules (in that order).

```
[Processing]
Undecorated
Suffix
Prefi
```

x
D
NI
S

[AttributeMap] Sections

The [AuthAttributeMap] and [AcctAttributeMap] sections of proxy.ini let you map the presence, absence, or specific value of an attribute in the incoming packet to a specific realm. This is referred to as attribute mapping. An [AuthAttributeMap] or [AcctAttributeMap] section consists of one or more RealmName entries. Each RealmName must match the name of a realm configuration file (RealmName.pro or RealmName.dir) in the same directory as proxy.ini.

Note: Attribute mapping is supported by proxy realms and directed realms. You cannot use this feature when forwarding packets to a proxy target that is not accessed through a realm.

Each RealmName entry is a list of statements that can be true or false regarding the attributes in an incoming RADIUS packet; we call these statements rules. Rules found in [AuthAttributeMap] apply to authentication packets; rules found in [AcctAttributeMap] apply to accounting packets. In all other respects, [AuthAttributeMap] or [AcctAttributeMap] are the same. The syntax for individual rules may vary; the following example shows all of the possible syntax variations:

[AuthAttributeMap
] RealmName
Attribute=Value
Attribute
~Attribute=Value
~Att
ribut
e M
[AcctAttribute
Map] M
For example:
[AuthAttribut
eMap]
CustTRealm
Framed-
Protocol=1
Service-
Type=2
CustQRealm
Framed-Protocol=PPP
~Service-
Type=Framed
NativeRealm

Each attribute mapping rule must begin with a space or tab character, followed optionally by a tilde (~),

then the name of a standard or vendor-specific RADIUS Attribute that is in one of the Steel-Belted Radius dictionary files. If a Value is present, it is preceded by an equal sign (=), and must specify a valid possible value for that attribute. The rule is terminated by a carriage return. Tilde (~) indicates that the rule is satisfied only if the attribute or attribute/value pair is not present in the packet.

Each RealmName entry in an [AuthAttributeMap] or [AcctAttributeMap] section is examined in sequence from top to bottom. Within each RealmName entry, each rule is evaluated in sequence from top to bottom. The results are as follows:

- If all of the rules in a RealmName entry evaluate to true, the packet is routed to the realm called RealmName and the remaining entries in the attribute map are ignored. If any of the rules in a RealmName entry evaluate to false, this entry does not result in a mapping. Steel-Belted Radius evaluates the next entry in the map.
- If Steel-Belted Radius encounters a RealmName entry that contains no rules, the packet is automatically directed to that realm.

Table 81 explains how the various types of rules are evaluated.

Table 82: Attribute Mapping Rules

Syntax Variation	Function of the Attribute Mapping Rule
Attribute=Value	If the Attribute is present in the request packet and it has the Value shown, then this rule is true. If the Attribute is not present, or if it is present but does not have the Value shown, then this rule is false. Note: The Steel-Belted Radius dictionary file radius.dct provides string aliases for certain integer values defined in the RADIUS standard. You can use these strings in attribute mapping rules.
Attribute	If the Attribute is present in the request packet, then regardless of its value, this rule is true. If the Attribute is not present, then this rule is false. Note: You won't often use the Attribute rule without a Value, because most of the RADIUS packets coming into your configuration are going to contain the same set of RADIUS attributes, but with different Values.
~Attribute=Value	Note the tilde (~) operator. This rule is looking for a specific attribute that may have any value except the one listed. If Attribute is present in the request packet and it does not have the Value shown, then this rule is true. If Attribute is not present, or if it is present but does have the Value shown, then this rule is false. Note: The following is not valid syntax: Attribute=~Value
~Attribute	Note the tilde (~) operator and the absence of a Value. If Attribute is not present in the request packet, then this rule is true. If Attribute is present, then this rule is false.

When setting up [AuthAttributeMap] or [AcctAttributeMap] rules for your configuration, you should distinguish between the different realms whose requests you are processing. Consider how specific your rules must be to identify each realm uniquely. Is the presence of a particular attribute sufficient (Ascend-IP-Address), or must the attribute have a specific value before you can be sure of its source (NAS-IP-Addr=n.n.n.n)? Make sure that your logic does not permit a crossing of requests between realms.

If a realm destination has been identified by applying an [AuthAttributeMap] entry to the attributes in a session's authentication request, Steel-Belted Radius uses the same realm for that session's accounting requests (if the realm is enabled for accounting). Generally, this is the desired behavior for the realm. You should provide an [AcctAttributeMap] entry only if there is no [AuthAttributeMap] entry for a realm and you want to map the realm using one or more accounting attributes.

[DirectedAcctMethods] Section

The [DirectedAcctMethods] section of the proxy.ini file lists one or more external database accounting configuration files (.acc) or local accounting initialization files (.ini) on the local server, and assigns each of these files a name by which it may be referenced in a RealmName.dir file.

The syntax for the [DirectedAcctMethods] section is as follows:

```
[DirectedAcctMethods]
Description=PathAndFile
Description=PathAndFile
.
.
```

where Description is the name by which you want to reference the accounting method and PathAndFile is the full pathname of a .acc or .ini file on the local server.

- Linux:
  /usr/lib/extras/acctlib.acc
  /usr/lib/extras/ouracct.ini
- Windows:
  c:\radius\extras\acctl
  ib.acc}
  c:\radius\extras\oura
  cct.ini

This is the file that implements the accounting method. The location of this file must not be the Steel-Belted Radius directory.

- If your PathAndFile identifies a .acc file, external database accounting is performed as configured in the file. You may reference the Steel-Belted Radius SQL accounting module in the [Bootstrap] section of this .acc file.
- If your PathAndFile identifies a .ini file, you may omit the [Bootstrap] section from this file. Normal Steel-Belted Radius logging is performed, except that:
  - Accounting log entries (for requests that are routed to this accounting method) are written to accounting log files (.act) in the specified Path, rather than in the server directory.
  - Logging details (which attributes are logged, and in which order) are controlled by the [Settings] and [Attributes] sections of the .ini file listed in PathAndFile, rather than the account.ini file found in the server directory.

[StaticAcct] Section

Static proxy accounting lets you send duplicate copies of certain types of accounting request to proxy realms (or any RADIUS-aware device), in addition to the normal routing of the original accounting request. The number of duplicates is not limited

The [StaticAcct] section of proxy.ini maps possible values of the Acct-Status-Type attribute to a list of proxy realms that receive statically-forwarded, duplicate copies of all accounting packets of that type.

Acct-Status-Type is a RADIUS standard attribute that identifies the type of accounting request. Table 82 lists the names and meanings assigned to Acct-Status-Type values 1, 2, 3, 7, and 8. Additional values for Acct-Status- Type have been defined by network access device vendors for use with their equipment; you can also use these values in the [StaticAcct] section.

Table 83: Acct-Status-Type Attribute Values

Acct-Status-Type Value	Name	Meaning
1	Start	A user session has started
2	Stop	A user session has stopped, request contains final statistics
3	Interim	A user session is in progress, request contains current statistics
7	Accounting-On	The network access device has started
8	Accounting-Off	The network access device is about to shut down

The syntax for a [StaticAcct] section is as follows:

```
[StaticAcct]
number=name
number=name
```

where each number is a possible value of the Acct-Status-Type attribute, and each name identifies a section called [name] that appears elsewhere in the proxy.ini file.

When it receives an accounting request with an Acct-Status-Type of number, Steel-Belted Radius uses the [StaticAcct] section to match number with name, and statically forwards a duplicate copy of the packet to all of the proxy realms listed in the [name] section.

Each [name] section consists of a list name in square brackets ([name]) followed by a list of proxy realms. Each of these realms must have a RealmName.pro file in the same directory as proxy.ini. Directed realms do not support static proxy accounting.

The syntax for a [name] section is as follows:

```
[name]
realm=RealmName
realm=RealmName
```

The [name] section is used only if its name is mapped to a number in the [StaticAcct] section of the proxy.ini file.

The following excerpt from a proxy.ini file demonstrates some of the flexibility of static proxy forwarding. Copies of all session-related accounting packets (Start, Stop, and Interim) are proxy-forwarded to a realm called billing. Copies of all device-related accounting packets (Accounting-On and Accounting-Off) are

proxy-forwarded, not only to billing, but also to a realm called operations.'

```
[Real
ms]
billin
g
oper
ation
s
[Stati
cAcct]
1 = SessionObserverList
2 = SessionObserverList
3 = SessionObserverList
7 = RASObserverList
8 =
RASObserverLi
st
[SessionObserv
erList] realm =
billing
[RASObserverL
ist] realm =
billing
realm= operations
```

[Interfaces] Section

If your server has more than one network interface, you can assign the outgoing proxy traffic for a particular realm to a particular interface card:

1. List the IP addresses associated with each network interface card in the [Addresses] section of the radius.ini file.

2. Create an [Interfaces] section for the proxy.ini file. This should consist of a list of one or more pairs in the following format:

```
[Interfaces]
InterfaceName = IPAddress
```

where InterfaceName is a label you assign to the given IPAddress.

3. Extend the existing entries in the [name] sections in .pro files for proxy realms with the InterfaceName defined in the [Interfaces] section so that they are in the following format:

```
[TargetSectio
n]
Target=NumAttempts,InterfaceName
```

where InterfaceName is the name of the interface defined above in the [Interfaces] section.

For example:
[Targets]
Bert=3,ABCInt
erface
Ernie=1,XYZInt
erface

Note: The ProxySource setting in the [Configuration] section of radius.ini disables per-realm control of proxy outbound interfaces. If ProxySource is not set, sockets are opened and bound for each interface on the server.

Proxyrl.ini File

Used by: GEE
Not used by: EE

The proxyrl.ini file supports a feature called smart static accounting, which lets you specify that the accounting packets for a proxy or directed realm should be forwarded to a list of one or more proxy realms. These groups of realms can also be used for static accounting configured in proxy.ini.

This file consists of a number of sections that you name. Each section name is referenced in the StaticAcctRealms parameter in the [Acct] section of a .pro or .dir file. Following the section name, you can list a number of proxy realm names, in the following format:

[realm-list-
name-1]
proxy-
realm-1
proxy-
realm-2
M
[realm-list-name-2]
.
.
.
For
example:
[StaticAcctTa
rgets1]
AcctSrvr1
AcctSrvr4

Note: You must be sure that the list of static accounting servers doesn't include any realms that use the list or an infinite loop occurs. If a realm that is included in a realm's list of static accounting servers and is specified in proxy.ini as doing static accounting, it gets duplicate accounting packets.

Proxy RADIUS Configuration (.pro) File

Used by: GEE
Not used by: EE

For each proxy realm that you want to configure on the Steel-Belted Radius server, you must create a file called RealmName.pro, where RealmName is the name of the realm, and you must add this RealmName to the [Realms] section of the proxy.ini file.

Note: If you create or edit a RealmName.pro file, you can apply your configuration changes dynamically, without stopping the server.

- Linux: Issue the HUP signal to the Steel-Belted Radius process. kill -HUP ProcessID
- Windows: Run RADHUP.EXE from the command shell.

After you do this, Steel-Belted Radius re-reads proxy.ini, filter.ini, and all .pro and .dir files in the server directory, and resets its realm configuration accordingly.
If you edit radius.ini while configuring a realm, you must stop and restart Steel-Belted Radius to load your new configuration.

[Auth] Section

The [Auth] section of a RealmName.pro file (Table 82) configures authentication for the proxy realm. The key parameters in these sections are:

- TargetsSection, which names the target selection strategy you want to use.
- FilterIn and FilterOut, which name the attribute filters you want applied to request and response packets,respectively.

Table 84: RealmName.pro [Auth] Syntax

Parameter	Function
Enable	If set to 1, enables forwarding of authentication packets to the realm called Realm- Name. If set to 0, the realm called RealmName is disabled for authentication. Default value is 0.
FilterOut=name	The FilterOut=name parameter causes Steel-Belted Radius to apply the filtering rules found in the [name] section of filter.ini. These rules are applied while Steel-Belted Radius is processing the incoming RADIUS request packet, and before it directs the packet "out" to the destination realm. You may also think of this as filtering various attributes and values "out" of the request before directing it to the realm.
FilterIn=name	The FilterIn=name parameter causes Steel-Belted Radius to apply the filtering rules found in the [name] section of filter.ini. These rules are applied after Steel-Belted Radius has received a response "in" from the destination realm, and while it is preparing the RADIUS response packet for its client. You may also think of this as filtering various attributes and values "in" to the response before returning it to the client.
MessageAuthenticator	If set to 1, a Message-Authenticator is inserted into each request forwarded to any target server in the realm. Default value is 0. Note: Both the proxy and the target RADIUS server requires this functionality.

Parameter	Function
NumAttempts	The number of times a timeout may occur when attempting to contact servers within the realm, before a failure is declared and the attempts to forward the request are stopped. Default value is 3.
RequestTimeout=x, y, z	A list of times, in seconds, to wait when attempting to contact a target server before timing out. The first value is the time to wait before the first timeout, and so on. The number of items in the list should be no greater than the NumAttempts setting. If NumAttempts is greater, than the last number listed is reused for subsequent timeouts. Default value is 5. Note: You can specify RequestTimeout or RequestTimeoutMills, but not both.
RequestTimeoutMills=x, y, z	A list of times, in milliseconds, to wait when attempting to contact a target server before timing out. The first value is the time to wait before the first timeout, and so on. The number of items in the list should be no greater than the NumAttempts setting. If NumAttempts is greater, than the last number listed is reused for subsequent timeouts. Note: You can specify RequestTimeout or RequestTimeoutMills, but not both.
RoundRobin	Specifies the number of target servers that are participating in round-robin load balancing. The count begins from the top of the list in the [name] section identified by TargetsSection. Other listed targets are used only after the round-robin targets fail for a particular request. Default value is 2.
StripRealm	• If set to 1, strip the realm name from the username before forwarding. • If set to 0, name stripping is disabled. Note: For proxy realms, realm name stripping is disabled (StripRealm = 0) by default. If you want to enable it, you must explicitly set StripRealm to 1.
TargetsSection=name	name identifies a section called [name] that appears elsewhere in the .pro file. This section lists all the targets in a proxy realm. When it receives a request for this proxy realm, Steel-Belted Radius selects a target from this list. Having the TargetsSection setting available in the [Auth] and [Acct] sections permits you to name different target selection parameters for proxy RADIUS authentication and accounting. Default value of name is AuthTargets, indicating the name of the section is [AuthTargets].
UseMasterDictionary	• If set to yes, inbound proxy responses for this realm use the master Steel-Belted Radius dictionary when authentication attributes are filtered in. • If set to no, proxy responses for this realm use the client-specific dictionary when authentication attributes are filtered in. Default value is yes. Note: This value overrides the global setting configured in the UseMasterDictionary parameter in the proxy.ini file.

[Acct] Section

The [Acct] section of a RealmName.pro file (Table 83) configures accounting. The key parameters in these sections are:

- TargetsSection, which names the target selection strategy you want to use.
- FilterIn and FilterOut, which name the attribute filters you want applied to request and response packets, respectively.

Table 85: RealmName.pro [Acct] Syntax

Parameter	Function
Enable	• If set to 1, enables forwarding of accounting packets to the realm called RealmName. • If set to 0, the realm called RealmName is disabled for accounting. Default value is 0.
Block	• If set to 0, the Steel-Belted Radius server sends an accounting acknowledgement immediately (for example, after Steel-Belted Radius records an accounting message). • If set to 1, the Steel-Belted Radius server waits for a response from the target realm before sending an accounting acknowledgement. Default value is 1. **Note**: Set the Block parameter to 0 if your network access device is not able to deal with long acknowledgment delays to accounting requests gracefully.
FilterIn=name	The FilterIn=name parameter causes Steel-Belted Radius to apply the filtering rules found in the [name] section of filter.ini. These rules are applied after Steel-Belted Radius has received a response "in" from the destination realm, and while it is preparing the RADIUS response packet for its client. You may also think of this as filtering various attributes and values "in" to the response before returning it to the client.
NumAttempts	the number of times a timeout may occur when attempting to contact servers within the realm, before a failure is declared and the attempts are stopped. Default value is 3.
RecordLocally	• If set to 1, log the packet locally before forwarding. • If set to 0, forward the packet and do not log locally. Default value is 1.
RequestTimeout=x, y, z	A list of times, in seconds, to wait when attempting to contact a target server before timing out. The first value is the time to wait before the first timeout, and so on. The number of items in the list should be no greater than the NumAttempts setting. If NumAttempts is greater, than the last number listed is reused for subsequent timeouts. **Note**: You can specify RequestTimeout or RequestTimeoutMills, but not both.
RequestTimeoutMills=x, y, z	A list of times, in milliseconds, to wait when attempting to contact a target server before timing out. The first value is the time to wait before the first timeout, and so on. The number of items in the list should be no greater than the NumAttempts setting. If NumAttempts is greater, than the last number listed is reused for subsequent timeouts. **Note**: You can specify RequestTimeout or RequestTimeoutMills, but not both.
RoundRobin	Specifies the number of target servers that are participating in "round-robin" load balancing. The count begins from the top of the list in the [name] section identified by TargetsSection. Other listed targets are only used after the round-robin targets fail for a particular request. Default value is 1.

Parameter	Function
StaticAcctRealms	If a setting is supplied for this parameter, accounting packets are forwarded to a list of realms. The setting given must be a section name defined in the proxyrl.ini file that lists the realms to which the accounting packets should be forwarded. See "Proxyrl.ini File" .
StripRealm=n	If set to 1, strip the realm name from the username before forwarding. If set to 0, name stripping is disabled. Default value is 0. NOTE: For proxy realms, realm name stripping is disabled (StripRealm = 0) by default. If you want to enable it, you must explicitly set StripRealm to 1.
TargetsSection=name	name identifies a section called [name] that appears elsewhere in the .pro file. This section lists all the targets in a proxy realm. When it receives a request for this proxy realm, Steel-Belted Radius selects a target from this list. Having the TargetsSection parameter available in the [Auth] and [Acct] sections permits you to name different target selection parameters for proxy RADIUS authentication and accounting. The default value of name is AcctTargets; in which case the name of the section is [AcctTargets].
UseMasterDictionary	If set to yes, inbound proxy responses for this realm use the master Steel-Belted Radius dictionary when accounting attributes are filtered in. If set to no, proxy responses for this realm use the client-specific dictionary when accounting attributes are filtered in. Default value is yes. Note: This value overrides the global setting configured in the UseMasterDictionary parameter in the proxy.ini file.

[AutoStop] Section

The [AutoStop] section of a realm configuration file permits you to activate the Proxy AutoStop feature. When this feature is enabled, an AutoStop request is automatically recorded and associated with the session in the current session's database when the initial Accounting-Start message is received. This AutoStop message may be used later to simulate an Accounting-Stop message which is fed back into the request processing engine, causing it to be forwarded to the appropriate realms and for the normal processes of ending the user session to be enacted.

Note: As the AutoStop record is generated when the session begins, it is simply a duplicate of the original Start request and does not have access to information about the lifetime of the user's actual activity.

Note: AutoStop records are not saved on persistent storage: this means that if Steel-Belted Radius is restarted, this information is lost and hence Accounting-Stop messages cannot be simulated for these user sessions.

Table 86: RealmName.pro [AutoStop] Syntax

Parameter	Function
Enable	Set to 0 to disable AutoStop for the current realm. Set to 1 to enable AutoStop for the current realm. Default value is 0.

Table 86 lists the parameters in other configuration files you must enable (set to 1) for AutoStop to operate.

Table 87: account.ini [TypeNames] Syntax

File	Section	Parameter
RealmName.pro	[Acct]	Enable
RealmName.pro	[Acct]	RecordLocally
radius.ini	[Configuration]	AcctAutoStopEnable

[Called-Station-ID] Section

The [Called-Station-ID] section of a RealmName.pro file allows the target realm to be selected based on DNIS. The [Called-Station-ID] section lists each DNIS string that identifies the realm. If this string is found in the Called- Station-Id attribute of an incoming RADIUS request, the request is assumed to be addressed to this realm.

The syntax is as follows:

```
[Called-Station-ID]
String
String
.
.
.
```

where String is a DNIS string.

For example:

```
[Called-Station-ID]
8005551212
8005551213
6175551212
```

You can also use wildcards, as in the following example:

```
[Called-Station-ID]
800*
508*
```

Target Selection Rules

Each [name] section of a RealmName.pro file specifies a set of rules that Steel-Belted Radius can use to

select a target for proxy-forwarding within the proxy realm. Each [name] section consists of a list of target servers. For
any particular request, if the first listed server fails to respond (or is presumed down), the other servers are tried in the order listed. A [name] section is activated by referencing it from the [Auth] and/or [Acct] sections.

Table 88: Proxy Realm Target Selection

To activate...	Use...
a [name] section for authentication	TargetName=name in the [Auth] section
the same [name] section for accounting	TargetName=name in the [Acct] section
some [other] section for accounting	TargetName=other in the [Acct] section

The full syntax is as follows:

```
[Auth]
TargetsSection=name
B
[Acct]
TargetsSection=name
A [nameA]
Server = n
Server = n
.
.
.
[name
B]
Server = n

Server = n
.
.
```

where Server is the name of a server that you've configured as a target for standard proxy RADIUS forwarding, and n is explained in the next section. Server must match a Proxy entry in the Steel-Belted Radius database. This Proxy entry provides the address and shared secret for the target server. All other settings in the Proxy entry (retry policy, proxy accounting) are overridden by the settings that you configure in the RealmName.pro file.

Note: If your server has multiple interface cards, you may add a parameter referring to the interface to each line to order the outgoing proxy traffic for the realm through a particular interface. See "[Interfaces] Section".

Round-Robin Load Balancing

If you have multiple target servers in a realm, you can select whether to use them in round-robin fashion (load balancing), primary/backup fashion, or a combination of both. The value of the RoundRobin entry in the [Auth] or [Acct] section indicates the number of targets that are to be used in round-robin fashion. The

count begins from the top of list in the [name] section. Other listed targets are used only if the round-robin targets fail for a particular request. If RoundRobin is 0 or 1, all requests are routed to the first target in the [name] list, assuming that it is up, the others are tried in the order listed.

If RoundRobin is 2 or greater (say, n), each request is routed to a different target server, in rotation among the first n listed targets. Requests are then load- balanced evenly among those targets. For any particular request, if one target fails to respond, other targets are attempted. The round-robin targets are tried first; if they all fail to respond; any additional targets are then tried in the order in which they appear in the list.

In the following example, RoundRobin is 3. Under normal circumstances, requests are balanced in round-robin fashion among the first three targets. The first request goes to Bert; the next goes to Ernie; the next to George; the next to Bert; the next to Ernie; the next to George; and so on. If any of these servers go down at some point, the other two are tried, in list order. The fourth target (Mary) receives requests only when other targets are down.

```
[Auth]
RoundRo
bin=3
NumAttem
pts=8
TargetsSection=Targets

[Targ
ets]
Bert=
1
Ernie
=1
Geor
ge=1
Mary
=5
```

Selecting a Backup Server

If RoundRobin is set to 0, Steel-Belted Radius makes a selection from the "other" servers in the list only if the primary server is down.

For Example:

```
[Auth]
RoundR
obin=0
NumAttempts=8
TargetsSection=Tar
gets
[Targ
ets]
Bert=
1
```

```
Ernie
=1
```

In this case, Bert is used until there is a problem; then Ernie becomes the server of second choice.

Realm Retry Policy

Each target selection rule in the [name] section permits you to name a target and assign it a numeric value:

```
[nam
e]
Serv
er = n
Server = n
.
.
```

The n setting indicates the number of times to retry requests to this target server when it doesn't respond (when no response is received from the server within the amount of time set by RequestTimeout in the [Auth] or [Acct] section).

The number of attempts to all servers within the entire realm is given by the NumAttempts value in the [Auth] or [Acct] section. For example, let's say that NumAttempts is 8 and there are three target servers, each with n set to 3:

```
[Auth]
NumAtte
mpts=8
TargetsSection=Targets
[Targ
ets]
Bert=
3
Ernie
=3
Geor
ge=3
```

Let's say that all three servers are down when a request comes into the realm. The first target (Bert) is tried 3 times; then the second target (Ernie) is tried 3 times; and the third target (George) is tried 2 times. At this point, the number of tries to all servers in the realm is 8, which equals NumAttempts. Steel-Belted Radius returns a failure response from the realm.

Note: A third attempt to George could not be made unless you edited the RealmName.pro file, increased NumAttempts to 9, and reloaded Steel-Belted Radius.

[FastFail] Section

The [FastFail] section of a realm configuration file permits you to fine-tune retry policies for individual realms, and for specific targets within a realm. If you provide a [FastFail] section, the ProxyFastFail parameter in the radius.ini [Configuration] section is ignored.

Table 89: RealmName.pro [FastFail] Syntax

Parameter	Function
MinFailures=x MinSeconds=y	These parameters define a tolerance level for failures to reach a target server within a realm. Such "failures" are judged according to the NumAttempts and RequestTimeout settings that you defined in the [Auth] or [Acct] sections. A target is presumed down once x consecutive failures have occurred and at least y seconds have elapsed. Once a target is presumed down, Steel-Belted Radius directs proxy requests to another target in the same realm, if available. It does not wait for responses from the failed target. However, it sends strobe requests periodically to the failed target to detect when that server comes back up. Once a response is received to one of these strobe requests, that server is no longer presumed down. **Note**: Strobe requests are sent to the "down" target server only if there are proxy requests addressed to its realm.
ResetSeconds=z	Once the realm's tolerance level is exceeded, this parameter specifies how long a target may be presumed down. The ResetSeconds value indicates the maximum number of seconds during which a server can be presumed down in the absence of strobe requests. If z seconds elapse with no strobe requests sent to the down server, the server is reset to "up." The status of a target that is presumed down is reset to "up" when one of the following occurs: • A response to a strobe request is received from the server. • There has been no request sent to the server for z seconds.

[ModifyUser] Section

The [ModifyUser] section of a realm configuration file permits you to decorate a realm, where the realm is determined by other means, such as DNIS or attribute mapping.

This is used mainly to enhance directed realms. For example, the following two users are in the database: george@gm and george@ford. Either user could log in as george, as Steel-Belted Radius would determine the realm, for example, by DNIS. Based on the realm, Steel-Belted Radius would append either @gm or @ford to the user name, and then use the Native User directed method to authenticate.

This methodology could also be used in a double-proxy situation. The first proxy uses DNIS to determine a realm, then decorates the name and forwards it to the next hop server. This second proxy (which may be a legacy RADIUS server that doesn't understand DNIS) could then handle realms based on the name decoration.

Table 90: RealmName.pro [ModifyUser] Syntax

Parameter	Function
AddPrefix=prefix AddSuffix=suffix	These parameters define the User-Name prefix and suffix.

[SpooledAccounting] Section

Proxy spooling is configured within the [SpooledAccounting] section of a RealmName.pro file.

```
[SpooledAccounting]
Enable=1
RolloverSeconds=
600
RolloverSize=1048
576
Directory=.\all_acct
_data
RetryInterval=60
ShutdownDelay=2
0
```

Table 91: RealmName.pro [SpooledAccounting] Syntax

Parameter	Function
Enable	• If set to 1, proxy spooling is enabled. • If set to 0, proxy spooling is disabled. Default value is 0.
RolloverSeconds	Specifies the rollover interval in seconds. After the interval elapses, the current spool file is closed and a new one is created. Default value is 600 (10 minutes.)
RolloverSize	Specifies the rollover file size limit in bytes. After the file size exceeds this limit, the current spool file is closed and a new one is created. If both RolloverSeconds and RolloverSize are set, the first parameter that exceeds its limit initiates rollover. Default value is 1,048,576 bytes (1 megabyte).
Directory	Specifies the directory where the spool (.psf) files are stored. The directory must be manually created in the RADIUS service directory. Default value is .\RealmName Note: Each realm must have its own directory for spool files. Otherwise, packets for multiple realms would be interspersed and a problem in one realm could prevent subsequent packets to other realms from being forwarded.
RetryInterval	Specifies the interval in seconds prior to retrying a proxy request if the target system (the downstream server where accounting data for this realm is sent) is down. Default value is 60.
ShutdownDelay	Specifies the amount of time (given as the number of seconds) prior to the execution of a shutdown request during which the final undelivered spooled packets in the spool file can be sent to their target. This value should be set according to the amount of

Parameter	Function
	accounting data normally received for this realm, and other relevant network conditions. If the target system is down when Steel-Belted Radius shuts down, this setting is not applied, and unspooling terminates immediately (and Steel-Belted Radius shuts down immediately). Upon restart, unspooling of accounting data restarts from the beginning of the oldest spool file. Default value is 20.

Note: Do not enable proxy spooling for realms that are not enabled for accounting.

Retry Sequence

If Steel-Belted Radius receives an accounting packet for a realm, and the target system is down, Steel-Belted Radius implements the RealmName.pro retry configuration, as in the following example:

```
[Acct]
RequestTimeout
=5,        3,        5
NumAttempts=3
```

In this example, Steel-Belted Radius attempts to proxy forward the accounting packet to the target IP address, as it would in a non-SpooledAccounting scenario. Three attempts are made; the first waits for five seconds before timing out, the second three seconds, and the third five seconds.

If there is still no response from the target after three attempts, the RetryInterval in the [SpooledAccounting] section is applied. If RetryInterval equals 60, then five seconds after the last unsuccessful NumAttempts is completed, Steel-Belted Radius waits another sixty seconds and then attempts the entire retry policy again.

Directed Realm Configuration (.dir) File

Used by: GEE
Not used by: EE

A *directed realm* specifies target methods for directed authentication and/or directed accounting. Its realm configuration file is called RealmName.dir. By default, an sample .dir file (example.dir) is installed with Steel- BeltedRadius.

The *directed authentication* feature permits the server to bypass its Authentication Methods list and map an incoming RADIUS request to one or more specific authentication methods. Steel-Belted Radius chooses the destination method based on routing information found in the request packet. The destination methods may be any authentication methods already configured on the local Steel-Belted Radius server, regardless of how they were configured; for example, a method may have been configured using the Administrator windows, the LDAP configuration interface, or a .aut configuration file.

If no *directed authentication* method is configured, every request percolates through the same Authentication Methods list, as defined in the Authentication Policies panel in SBR Administrator. This behavior may or may not be ideal for every customer. Directed authentication lets you tailor an authentication methods list to a customer's needs.

Directed accounting is also possible. The destination accounting method may be the Steel-Belted Radius accounting log, an external database configured using a .acc file, or a distinct accounting log file that contains entries only for this customer.

To activate these features, you must create RealmName.dir files, place them in the Steel-Belted Radius directory, and list them in the [Directed] section of proxy.ini. Subsequently, any requests that arrive addressed to one of these realm names are processed on the local server using the instructions you've provided in proxy.ini and in the corresponding RealmName.dir file.

After you edit a RealmName.dir file, you must apply your changes as follows. If you have added or changed:

- Any directed accounting methods, you must stop and restart the server to load your new configuration.
- Directed authentication methods in which external database (SQL or LDAP) authentication is used, you must stop and restart the server to load your new configuration.
- Directed authentication methods in which local or pass-through (Native, UNIX, Domain, Host, SecurID, or TACACS+) authentication is used, you can apply your configuration changes dynamically, without stopping the server.
  - Linux: Issue the HUP signal to the Steel-Belted Radius process. kill -HUP ProcessID
  - Windows: Run the RADHUP.EXE program from the command shell. (RADHUP.EXE is located in the server directory that you specified at installation time, usually C:\Program Files (x86)\Pulse Secure\ Steel-BeltedRadius\Service.)

Steel-Belted Radius re-reads proxy.ini, filter.ini, and all .pro and .dir files in the server directory, and resets its realm configuration accordingly.

Note: If you edit radius.ini while configuring a realm, you must restart Steel-Belted Radius to load your new configuration.

[Auth] Section

Directed authentication is enabled in a realm by setting the Enable parameter in the [Auth] section of the corresponding RealmName.dir file, where RealmName is the name of the realm. The syntax is as follows:

```
[Auth
]
Enabl
e = 1
StripRealm=1
UseMasterDictionary
= yes
```

Table 92: RealmName.dir [Auth] Syntax

Parameter	Function
Enable	• If set to 1 in the [Auth] section of a RealmName.dir file, the directed authentication realm called RealmName is enabled.

Parameter	Function
	• If set to 0, the realm is disabled. By enabling a directed authentication realm, you make it possible for Steel-Belted Radius to override the Authentication Methods list on the local server by providing an alternate list - for requests addressed to this realm only. Details of this list are provided in the [AuthMethods] section of the same RealmName.dir file.
StripRealm	• If set to 1, Steel-Belted Radius strips the realm name from the username before attempting to authenticate the user's request. • If set to 0, realm name stripping is disabled. Note: For directed realms, realm name is enabled (StripRealm = 1) by default. If you want to disable it, you must explicitly set StripRealm to 0.
UseMasterDictionary	• If set to yes, inbound proxy responses for this realm use the master Steel-Belted Radius dictionary when authentication attributes are filtered in. • If set to no, proxy responses for this realm use the client-specific dictionary when authentication attributes are filtered in. Default value is yes. Note: This value overrides the global setting configured in the UseMasterDictionary parameter in the proxy.ini file.

[AuthMethods] Section

If directed authentication is enabled, the [AuthMethods] section of a RealmName.dir file lists one or more authentication methods to be used.

The syntax is as follows:

```
[AuthMethods]
Description
Description
M
```

where Description is the "official name" of an authentication method configured on the Steel-Belted Radius server. For example:

```
[AuthMethods]
Native
User
SecurID
User
SecurID
Prefix
SecurID
Suffix
SecurID
TACACS+
```

```
User
TACACS+    Prefix
TACACS+    Suffix
Windows Domain
User
WindowsDomain
Group  UNIX User
UNIXGroup
<InitializationString=SQL>
<InitializationString=LDAP>
```

If you want your [AuthMethods] section to reference an external authentication method, a Description string must match the names of that method. If you want your [AuthMethods] section to reference an external database, enter the InitializationString value from the [Bootstrap] section of the corresponding. aut file.

Note: There is no interaction between the settings in the Authentication Policies panel and in RealmName.dir files, or between different RealmName.dir files. For example, if you disable the UNIX User method (for Linux) or Windows Domain User method (for Windows) in the Authentication Policies panel while it is enabled in a RealmName.dir file, it remains enabled in RealmName.dir.

[Acct] Section

Directed accounting is enabled in a realm by setting the Enable parameter in the [Acct] section of the corresponding RealmName.dir file, where RealmName is the name of the realm. The syntax is as follows:

```
[Acct]
Enabl
e = 1
StripRealm = 0
RecordLocally    =    0
UseMasterDictionary
= yes
```

Table 93: RealmName.dir [Acct] Syntax

Parameter	Function
Enable	• If set to 1 in the [Acct] section of a RealmName.dir file, the directed accounting realm called RealmName is enabled. • If set to 0, the realm is disabled. By enabling a directed accounting realm, you make it possible for Steel-Belted Radius to override the normally configured accounting methods on the local server by providing an alternate list - for requests addressed to this realm only. Details of this list are provided in the [AcctMethods] section of the same RealmName.dir file. Default value is 0.
RecordLocally	• If set to 1, Steel-Belted Radius writes accounting records to its main accounting log file in addition to the accounting destinations specified in [AcctMethods].

Parameter	Function
	• If set to 0, this feature is disabled. Default value is 0.
StaticAcctRealms (GEE only)	If a value is supplied for this parameter, accounting packets are forwarded to a list of realms. The setting given must be a section name defined in the proxyrl.ini file that lists the realms to which the accounting packets should be forwarded. See "Proxyr.ini File".
StripRealm	• If set to 1, Steel-Belted Radius strips the realm name from the username before attempting to authenticate the user's request. • If set to 0, realm name stripping is disabled. Note: For directed realms, username stripping is enabled (StripRealm = 1) by default. If you want to disable it, you must explicitly set StripRealm to 0.
UseMasterDictionary	• If set to yes, inbound proxy responses for this realm use the master Steel-Belted Radius dictionary when accounting attributes are filtered in. • If set to no, proxy responses for this realm use the client-specific dictionary when accounting attributes are filtered in. Default value is yes. Note: This value overrides the global setting configured in the UseMasterDictionary parameter in the proxy.ini file.

[AcctMethods] Section

If directed accounting is enabled, the [AcctMethods] section of a RealmName.dir file lists one or more accounting methods to be used. The syntax is as follows:

```
[AcctMeth
ods]
Descriptio
n
Descriptio
n M
```

where Description is the "official name" of a directed accounting method configured in the proxy.ini file.

[Called-Station-ID] Section

The [Called-Station-ID] section of a RealmName.dir file allows Steel-Belted Radius to select a realm to be used for directed authentication and/or accounting based on DNIS information supplied in an incoming RADIUS packet.

The [Called-Station-ID] section lists each DNIS string that identifies the realm. If this string is found in the Called- Station-Id attribute of an incoming request, the directed authentication and/or accounting rules found in the corresponding RealmName.dir file are applied to the request.

The syntax is as follows:

```
[Called-
Station-ID]
String
```

String
.
.
.

where String is a DNIS string.

[ModifyUser] Section

The [ModifyUser] section of a realm directed file permits you to decorate a realm, where the realm is determined by other means, such as DNIS or attribute mapping. This is used mainly to enhance directed realms. For example, the following two users are in the database: george@gm and george@ford. Either user could log in as george, as Steel-Belted Radius would determine the realm, for example, by DNIS. Based on the realm, Steel- Belted Radius would append either @gm or @ford to the user name, and then use the Native User directed method to authenticate.

Table 94: RealmName.dir [ModifyUser] Syntax

Parameter	Function
AddPrefix=prefix AddSuffix=suffix	These parameters define the User-Name prefix and suffix.

radius.ini Realm Settings

Used by: GEE, EE*
Not Used By: —

The [Self] section of radius.ini lets you list all of the realm names that should be handled by this Steel-Belted Radius server, rather than being proxied to other targets.

GEE: The [Configuration] section of radius.ini provides two parameters that you can use to enable or disable realm features for the Steel-Belted Radius server: ExtendedProxy and AttributeEdit. Both parameters are enabled (set to 1) by default. You can disable either feature by setting the corresponding parameter to 0.

Note: If you edit radius.ini while configuring a realm, you must stop and restart the Steel-Belted Radius server to load your new realm configuration.

Chapter 8

Database Error Map Files

This chapter describes the database error files, which specify how Steel-Belted Radius should classify errors returned by backend databases. Hard errors cause Steel-Belted Radius to drop and re-establish the connection to the backend database. Soft errors cause the connection to the database to be maintained.

mssql.ini File

mysql.ini File

oracle.ini File

mssql.ini File

Used by: GEE, EE
Not used by: —

The mssql.ini configuration file specifies the errors returned by a backend MS-SQL database that should be classified as soft errors. Error codes not listed in mssql.ini are presumed to be hard errors, which cause Steel- Belted Radius to drop and re-establish the connection to the MS-SQL database. Note that database-dependent RADIUS transactions will fail while the connection to the MS-SQL database is being re-established.

Each entry in the mssql.ini configuration file consists of an error number (positive integer), followed by a descriptive comment. For best performance, you should use the mssql.ini file to identify only the most common soft errors.

Note: If you Incorrectly define a hard error as a soft error and the error is encountered during processing, you may be required to restart Steel-Belted Radius to reset the database plug-in.

[SoftErrors] Section

The [SoftErrors] section identifies each MS-SQL error code that should be classified as a soft error. To include a comment or description for the error code, enter a semi-colon after the error code, followed by the comment.

```
[SoftErrors]
151 ; '%.*ls' is an invalid money value.
206 ; Operand type clash: %ls is incompatible with %ls
210 ; Syntax error converting datetime from binary/varbinary string.
212 ; Expression result length exceeds the maximum. %d max, %d
found.  220 ; Arithmetic overflow error for data type %ls, value = %ld.
229 ; %ls permission denied on object '%.*ls', database '%.*ls', owner '%.*ls'.
.
.
```

.

[oracle.ini File]

Used by: GEE, EE
Not used by: —

The oracle.ini configuration file specifies the errors returned by a backend Oracle database that should be classified as soft errors. Error codes not listed in oracle.ini are presumed to be hard errors, which cause Steel- Belted Radius to drop and re-establish the connection to the Oracle database. Note that database-dependent RADIUS transactions will fail while the connection to the Oracle database is being re-established.

Each entry in the oracle.ini configuration file consists of an error number (positive integer), followed by a descriptive comment. For best performance, you should use the oracle.ini file to identify only the most common soft errors.

Note: If you Incorrectly define a hard error as a soft error and the error is encountered during processing, you may be required to restart Steel-Belted Radius to reset the database plug-in.

[SoftErrors] Section

The [SoftErrors] section identifies each Oracle error code that should be classified as a soft error. To include a comment or description for the error code, enter a semi-colon after the error code, followed by the comment.

```
[SoftErrors]
00001 ; unique constraint (string.string) violated
00036 ; maximum number of recursive SQL levels (string)
exceeded 00054 ; resource busy and acquire with NOWAIT
specified
00055 ; maximum number of DML locks exceeded
00057 ; maximum number of temporary table locks
exceeded 00060 ; deadlock detected while waiting for
resource
00100 ; no data found
.
.
.
```

Chapter 9

EAP Configuration Files

This chapter describes the EAP configuration and helper files, which specify options for automatic EAP helper methods. These files are loaded at startup time and resides in the Steel-Belted Radius directory.

eap.ini File

peapauth.aut File

tlsauth.aut File

tlsauth.eap File

ttlsauth.aut File

eap.ini File

Used by: GEE, EE
Not used by: —

Note: You should use the SBR Administrator to maintain settings in the eap.ini file. You should not edit the eap.ini file manually.

The eap.ini configuration file configures the sequence in which EAP authentication types are tried when authenticating users by means of the different Steel-Belted Radius authentication methods.

Each authentication method that you want EAP authentication to be performed against must be configured within this eap.ini file.

This file must contain one section for each authentication method that you use, and the title of the section must identify the authentication method:

- Native User
- SecurID
- SecurID User
- SecurID Prefix
- SecurID Suffix
- LDAP
- SQL
- SQL-ORACLE
- Windows Domain User
- Windows Domain Group
- EAP-TLS(GEE/EE)
- EAP-TTLS(GEE/EE)
- EAP-PEAP(GEE/EE)

- winauth
- defaultMethods

```
[Native-User]
EAP-Only = 0
First-Handle-Via-Auto-EAP = 0
EAP-Type = TTLS, LEAP, MD5-Challenge
Available-EAP-Types=MD5-Challenge,MS-CHAP-V2,LEAP,TLS  Available-EAP-Only-Values=0,1
Available-Auto-EAP-Values=1
```

Note: Steel-Belted Radius is configured with an eap.ini file that should work for most environments.

Table 94 lists the parameters in each section.

Table 95: eap.ini Syntax

Parameter	Function
EAP-Only	• If set to 0, the authentication method accepts all types of user credentials. • If set to 1, the authentication method is given only EAP credentials or acts only as a back-end server to an automatic EAP protocol method. For authentication methods expected to handle EAP-TTLS inner authentications, this parameter should be set to 0 or 1 depending on the type of credentials used in the inner authentication. Note: If you are using SecurID with PEAP, set this value to 0. Since the PEAP plug-in converts the inner EAP/Generic Token credentials to PAP for security reasons, setting this value to 1 causes SecurID processing to be skipped when using EAP/Generic Token, ultimately leading to the user being rejected.
EAP-Type	A comma-separated list of the EAP protocols to support for this authentication method. The first protocol in the list is the primary protocol. Protocols that appear later in the list are used with this authentication method only if the client responds with an EAP NAK and specifies such a protocol or if another authentication method triggers the use of the protocol but cannot complete the request. Valid values include the following: • LEAP • Generic-Token • MD5-Challenge • TTLS (GEE/EE only) • TLS (GEE/EE only) • MS-CHAP-V2 (GEE/EE only). Leave the EAP-Type list empty to disable EAP for this authentication method.
First-Handle-Via-Auto-EAP	• If set to 1 and the user credentials are EAP, an appropriate automatic EAP helper method is called before the authentication method. The purpose of calling the automatic EAP helper method is to convert the user's EAP credentials into a format acceptable to the authentication method. • If set to 0, the authentication method itself handles the request directly, before any automatic helper methods. Default varies based on type of user. Refer to the comments in the eap.ini file for more information.

Parameter	Function
	Note: If you want to use machine authentication, you must enter 1 for this setting in the [Windows Domain User] and [Windows Domain Group] sections of eap.ini. Note: You must set the AllowMachineLogin setting in the [WindowsDomain] section of winauth.aut to Yes if you want to use machine authentication. For more information, see "winauth.aut File" .
Available-EAP-Types	A comma-separated list of the EAP protocols that can be selected when configuring the Steel-Belted Radius server by means of the SBR Administration. Valid values include the following: • TTLS (GEE/EE only) • TLS (GEE/EE only) • MS-CHAP-V2 (GEE/EE only) • LEAP • Generic-Token • MD5-Challenge
Available-EAP-Only-Values	Controls whether the Use EAP authentication only checkbox in the EAP Setup window (accessed through the Authentication Policies panel in SBR Administrator) is enabled. Network administrators can use this parameter to control whether SBR Administrator users can select EAP authentication options. • If set to 0,1, users can check and uncheck the Use EAP authentication only checkbox. • If set to 0, the Use EAP authentication only option is disabled and the checkbox is inactive. • If set to 1, the Use EAP authentication only option is enabled and the checkbox is inactive. Default varies based on type of user. Refer to the comments in the eap.ini file for more information.
Available-Auto-EAP-Values	Controls whether the Handle via Auto-EAP first checkbox in the EAP Setup window (accessed through the Authentication Policies panel in SBR Administrator) is enabled. Network administrators can use this parameter to control whether SBR Administrator users can select auto-EAP options. • If set to 0,1, users can check and uncheck the Handle via Auto-EAP first checkbox. • If set to 0, the Handle via Auto-EAP first option is disabled and the checkbox is inactive. • If set to 1, the Handle via Auto-EAP first option is enabled and the checkbox is inactive. Default varies based on type of user. Refer to the comments in the eap.ini file for more information.

peapauth.aut File

Used by: GEE, EE*
Not used by: –

Note: You should use the SBR Administrator to maintain settings in the peapauth.aut file. You should not edit the peapauth.aut file manually.

Settings for the EAP-PEAP plug-in are stored in the peapauth.aut file. The peapauth.aut configuration file is

read each time the Steel-Belted Radius server restarts (or, if you are using the GEE edition of Steel-Belted Radius, each time the Steel-Belted Radius server receives a HUP signal).

[Bootstrap] Section

The [Bootstrap] section of the peatauth.aut file (Table 94) specifies information that Steel-Belted Radius uses to load the EAP-PEAP authentication method.

Table 96: peapauth.aut [Bootstrap] Syntax

Parameter	Function
LibraryName	Specifies the name of the EAP-PEAP module. Default value is peapauth.dll for Windows and peapauth.so for Linux. Do not change this unless you are advised to do so by Pulse Secure Global Support Center.
Enable	Specifies whether the EAP-PEAP authentication module is enabled. • If set to 0, EAP-PEAP is disabled, and the authentication method does not appear in the Authentication Methods list in the Authentication Policies panel. • If set to 1, EAP-PEAP is enabled. Default value is 0.
InitializationString	Specifies the name of the authentication method to appear in the Authentication Methods list in the Authentication Policies panel. The name of each authentication method must be unique. If you create additional .aut files to implement authentication against multiple databases, the InitializationString value in each file must specify a unique method name. Default value is EAP-PEAP.

[Server_Settings] Section

The [Server_Settings] section (Table 96) lets you configure the basic operation of the EAP-PEAP plug-in.

Table 97: peapauth.aut [Server_Settings] Syntax

Parameter	Function
TLS_Message_Fragment_Length	Set to the maximum size TLS message length that may be generated during each iteration of the TLS exchange. Some Access Points may have problems with RADIUS responses or EAP messages that exceed the size of one Ethernet frame (1500 bytes including IP/UDP headers). The default value (1020) prevents the RADIUS challenge response (carried in a UDP packet) from exceeding one Ethernet frame. This is likely to be the safest setting. Setting a smaller value affects the number of RADIUS challenge/response round-trips required to conclude the TLS exchange. While a value of 1400 may result in 6 round-trips, a value of 500 may result in 15 round-trips. The minimum value is 500.
Return_MPPE_Keys	Setting this attribute to 1 causes the module to include RADIUS MS-MPPE-Send-Key and MS-MPPE-Recv-Key attributes in the final RADIUS Accept response sent to the Access Point. This is necessary for the Access Point to key the WEP encryption. If the Access Point is authenticating only end-users and WEP is not being used, this attribute may be set to 0.

Parameter	Function
	Default value is 1.
DH_Prime_Bits	Specifies the size of the prime number that the module uses for Diffie-Hellman exponentiation. Selecting a larger prime number makes the system less susceptible to certain types of attacks but requires more CPU processing to compute the Diffie-Hellman key agreement operation. Valid values are 512, 1024, 1536, 2048, 3072, and 4096. Default value is 1024.
Cipher_Suites	Specifies the TLS cipher suites (in order of preference) that the server is to use. These cipher suites are documented in RFC 2246, "The TLS Protocol Version 1.2," and other TLS- related RFCs and draft RFCs. Default value is: 0x3C, 0x3D, 0x67, 0x6B, 0x40, 0x6A, 0x9C, 0x9D, 0x9E, 0x9F, 0xA2, 0xA3.
PEAP_Min_Version	Specifies the minimum version of the PEAP protocol that the server should negotiate: • If set to 0, the server negotiates version 0, which is compatible with Microsoft's initial PEAP implementation (shipped in Microsoft XP Service Pack 1). • If set to 1, the server negotiates version 1, which is compatible with Cisco's initial PEAP implementation (shipped in Cisco ACU). Default value is 0. **Note**: The value entered in this setting must be less than or equal to the value entered for the PEAP_Max_Version setting.
LibraryName	Specifies the name of the EAP-PEAP module. Default value is peapauth.dll for Windows and peapauth.so for Linux. Do not change this unless you are advised to do so by Pulse Secure Global Support Center.
Enable	Specifies whether the EAP-PEAP authentication module is enabled. • If set to 0, EAP-PEAP is disabled, and the authentication method does not appear in the Authentication Methods list in the Authentication Policies panel. • If set to 1, EAP-PEAP is enabled. Default value is 0.
InitializationString	Specifies the name of the authentication method to appear in the Authentication Methods list in the Authentication Policies panel. The name of each authentication method must be unique. If you create additional .aut files to implement authentication against multiple databases, the InitializationString value in each file must specify a unique method name. Default value is EAP-PEAP.

[Inner_Authentication] Section

Used by: GEE
Not used by: EE

The [Inner_Authentication] section (Table 96) lets you specify the way in which the inner authentication step is to operate.

Table 98: peapauth.aut [Inner_Authentication] Syntax

Parameter	Function
Directed_Realm	Omitting this setting causes the inner authentication request to be handled like any other request received from a RAS. Specifying the name of a directed realm causes the request to be routed based on the methods listed in the directed realm. Default is to process the inner authentication through standard request processing.

Note: The filters named in these settings must be defined in the filter.ini file.

[Request Filters] Section

Request filters (Table 98) affect the attributes of inner authentication requests.

Table 99: peapauth.aut [Request Filters] Syntax

Parameter	Function
Transfer_Outer_Attribs_to_New	This filter affects only a new inner authentication request (rather than continuations of previous requests). If this filter is specified, all attributes from the outer request are transferred to the inner request and this filter is applied. The transfer occurs and the filter is applied before any attributes specified in the inner authentication are added to the request. If this filter is not specified, no attributes from the outer request are transferred to the inner request.
Transfer_Outer_Attribs_to_Continue	This filter affects only a continued inner authentication request (rather than the first inner authentication request). If this filter is specified, all attributes from the outer request are transferred to the inner request and this filter is applied. The transfer occurs and the filter is applied before any attributes specified in the inner authentication are added to the request. If this filter is not specified, no attributes from the outer request are transferred to the inner request
Edit_New	This filter affects only a new inner authentication request (rather than continuations of previous requests). If this filter is specified, it is applied to the inner request that is the cumulative result of attributes transferred from the outer request (see Transfer_Outer_Attribs_To_New in this table) and attributes included in the inner authentication request sent through the tunnel by the client. If this filter is not specified, the request remains unaltered.
Edit_Continue	This filter affects only a continued inner authentication request (rather than a new inner authentication request). If this filter is specified, it is applied to the inner request that is the cumulative result of attributes transferred from the outer request (see Transfer_Outer_Attribs_To_Continue in this table) and attributes included in the inner authentication request sent through the tunnel by the client. If this filter is not specified, the request remains unaltered.

Note: The filters named in these settings must be defined in the filter.ini file. The filters named in these settings must be defined in the filter.ini file.

[Response Filters] Section

Response filters (Table 99) affect the attributes in the responses returned to authentication requests.

Table 100: peapauth.aut [Response Filters] Syntax

Parameter	Function
Transfer_Inner_Attribs_To_Accept	This filter affects only an outer Access-Accept response that is sent back to a network access device. If this filter is specified, the filter is applied to the inner authentication response and all resulting attributes are transferred to the outer authentication response. If this filter is not specified, no inner authentication response attributes are transferred to the outer authentication response.
Transfer_Inner_Attribs_To_Reject	This filter affects only an outer Access-Reject response that is sent back to a network access device. If this filter is specified, the filter is applied to the inner authentication response and all resulting attributes are transferred to the outer authentication response. If this filter is not specified, no inner authentication response attributes are transferred to the outer authentication response.

Note: The filters named in these settings must be defined in the filter.ini file.

[Session Resumption] Section

The [Session_Resumption] section (Table 100) lets you specify whether session resumption is permitted and under what conditions session resumption is performed.

Note: For session resumption to work, the network access device must be configured to handle the Session-Timeout return list attribute, because the network access device must be able to tell the client to reauthenticate after the session timer has expired.

Table 101: peapauth.aut [Session_Resumption] Syntax

Parameter	Function
Session_Timeout	Set this attribute to the maximum number of seconds you want the client to remain connected to the network access device before having to re-authenticate. • If set to a number greater than 0, the lesser of this value and the remaining resumption limit (see description below) is sent in a Session-Limit attribute to the RADIUS client on the RADIUS Access Accept response. • If set to 0, no Session-Limit attribute is generated by the plug-in. This does not prevent the authentication methods performing secondary authorization from providing a value for this attribute. Default value is 0. Entering a value such as 600 (10 minutes) does not necessarily cause a full re-authentication to occur every 10 minutes. You can configure the resumption limit to make most re-authentications fast and computationally cheap.
Termination_Action	Specifies the value to return for the Termination-Action attribute sent for an accepted client. This is a standard attribute supported by most Access Points and determines what happens when the session timeout is reached. Valid values are: • -1: Do not send the attribute. • 0: Send the Termination-Action attribute with a value of 0.

Parameter	Function
	• 1: Send the Termination-Action attribute with a value of 1. Default value is -1. Note that this does not prevent the authentication methods performing secondary authorization from providing a value for this attribute.
Resumption_Limit	Set this attribute to the maximum number of seconds you want the client to be able to re-authenticate using the TLS session resumption feature. This type of re-authentication is fast and computationally cheap. It does, however, depend on previous authentications and may not be considered as secure as a complete (computationally expensive) authentication. Specifying a value of 0 disables the session resumption feature. Default value is 0.

tlsauth.aut File

Used by: GEE, EE
Not used by: –

Note: You should use the SBR Administrator to maintain settings in the tlsauth.aut file. You should not edit the tlsauth.aut file manually.

Settings for the EAP-TLS authentication method are stored in the tlsauth.aut file. The tlsauth.aut configuration file is read each time the Steel-Belted Radius server restarts (or, if you are using the GEE edition of Steel- Belted Radius, each time the Steel-Belted Radius server receives a HUP signal).

[Server_Settings] Section

The [Server_Settings] section contains the settings that control the basic operation of the EAP-TLS authentication method.

Table 102: tlsauth.aut [Server_Settings] Syntax

Parameter	Function
TLS_Message_Fragment_Length	Maximum TLS message length that may be generated during each iteration of the TLS exchange. Anecdotal evidence suggests that some Access Points may have problems with RADIUS responses or EAP messages that exceed the size of one Ethernet frame (1500 bytes including IP/UDP headers). The default value (1020) prevents the RADIUS challenge response (carried in a UDP packet) from exceeding one Ethernet frame. This is likely to be the safest setting. Setting a smaller value affects the number of RADIUS challenge/response round-trips required to conclude the TLS exchange. While a value of 1400 may result in 6 round-trips, a value of 500 may result in 15 round-trips. The minimum value is 500.
Verify_User_Name_Is_Principal_Name	Certificates issued by Microsoft's Windows 2000 Certificate Server typically include a Subject Alternative Name/Other Name attribute, where Principal Name set to something likeuser@certtest.acme.com. The Windows XP client that supports EAP-TLS in conjunction with 802.1X extracts this attribute value from the client's certificate and uses it to respond to the Access Point's EAP Identity Request. The Access Point, in turn, packages up this value as the RADIUS User-Name attribute in requests it sends to a RADIUS server.

Parameter	Function
	• If set to 1, the EAP-TLS module verifies that the contents of the RADIUS User-Name attribute match the 'Principal Name' of the certificate used to authenticate the user. • If set to 0, no such check is performed. The value should be set to 0 if the certificates used do not include a 'Principal Name' or if the client being used does not report the contents of 'Principal Name' as the user's identity in response to an EAP Identity Request. Default value is 0.
Return_MPPE_Keys	Setting this attribute to 1 causes the EAP-TLS module to include RADIUS MS-MPPE-Send-Key and MS-MPPE-Recv-Key attributes in the final RADIUS Accept response sent to the Access Point. This is necessary for the Access Point to key the WEP encryption. If the Access Point is authenticating only end-users and WEP is not being used, this attribute may be set to 0. Default value is 1.
DH_Prime_Bits	Specifies the size of the prime number that the module uses for Diffie-Hellman exponentiation. Selecting a larger prime number makes the system less susceptible to certain types of attacks but requires more CPU processing to compute the Diffie-Hellman key agreement operation. Valid values are 512, 1024, 1536, 2048, 3072, and 4096. Default value is 1024.
Cipher_Suites	Specifies the TLS cipher suites (in order of preference) that the server is to use. These cipher suites are documented in RFC 2246, "The TLS Protocol Version 1.2," and other TLS- related RFCs and draft RFCs. Default value is: 0x3C, 0x3D, 0x67, 0x6B, 0x40, 0x6A, 0x9C, 0x9D, 0x9E, 0x9F, 0xA2, 0xA3.
Profile	Specifies a profile that is to be used to select attributes sent back on an Access-Accept. By default, additional attributes are not sent back.
Verify_Client_Certificate_Published	Specifies that the EAP-TLS module should check that the client certificate is published in Active Directory for accocunt users. Default value is 0 (disabled)

[CRL_Checking] Section

The [CRL_Checking] section (Table 102) lets you specify settings that control how Steel-Belted Radius performs certificate revocation list (CRL) checking.

Table 103: tlsauth.aut [CRL_Checking] Syntax

Parameter	Function
Enable	Specifies whether CRL checking is enabled. Default value is 0 (disabled).
Retrieval_Timeout	Specifies the time (in seconds) that EAP-TLS will wait for a CRL checking transaction to complete when the CRL check involves a CRL retrieval. When CRL retrieval takes longer than the specified time, the user's authentication request will result in a reject. Default value is 5 seconds.
Expiration_Grace_Period	Specifies the time (in seconds) after expiration during which a CRL is still considered acceptable. EAP-TLS will always attempt to retrieve a new CRL when it is presented with a certificate chain and it finds an expired CRL in its cache.

Parameter	Function
	• If set to 0 (strict expiration mode), EAP-TLS will not accept a CRL that has expired. • If set to a value greater than 0 (lax expiration mode), EAP-TLS will consider the expired CRL as an acceptable stand-in from the time the CRL expires to the time the grace period ends. Default value is 0 (strict expiration mode).
Allow_Missing_CDP_Attribute	Specifies whether the omission of a CDP attribute in a non-root certificate is acceptable. Without a CDP attribute, EAP-TLS will not know how to retrieve a CRL and will not be able to perform a revocation check on the certificate. • If set to false, EAP-TLS will not accept a CRL with a missing CDP attribute. • If set to true, EAP-TLS will allow such certificates and skip CRL checking for them. Default value is true.
Default_LDAP_Server_Name	Specifies what LDAP server name to use if the CDP contains a value that begins with the string //ldap:\\\. This style of CDP (generated by some CAs) does not include the identity of the LDAP server. Specify the name of the LDAP that contains the CRLs if you expect to encounter certificates with this style CDP. If you don't specify a server name and such certificates are encountered, the CRL retrieval will fail.
Enable_CRL_Cache_Timeout	Specifies whether CRL cache timeout is enabled. Valid values are: • If set to 0, the CRL is refreshed whenever it expires. • If set to 1, the CRL begins to expire when the age of the CRL in the cache exceeds the number of hours specified in the CRL_Cache_Timeout_period parameter or when the scheduled CRL expiration time occurs, whichever comes first. After a CRL has expired (because its scheduled expiration time has passed or because the CRL cache has timed out), Steel-Belted Radius uses the expiration grace period to determine whether it should use the current CRL.
CRL_Cache_Timeout_Period	Specifies the maximum age, in hours, that a CRL can exist in the cache before it begins to expire. • If you enter 0, Steel-Belted Radius always regards the CRL in the cache as expired and downloads a new CRL every time it receives a client certificate request. • If you enter a number greater than 0, the CRL begins to expire when the age of the CRL in the cache exceeds the number of hours specified in this parameter or when the scheduled CRL expiration time occurs, whichever comes first. Note: You must set Enable_CRL_Cache_Timeout to 1 or the CRL_Cache_Timeout_Period parameter is ignored.
Default_LDAP_Server_Name	Specifies what LDAP server name to use if the CDP contains a value that begins with the string //ldap:\\\. This style of CDP (generated by some CAs) does not include the identity of the LDAP server. Specify the name of the LDAP that contains the CRLs if you expect to encounter certificates with this style CDP. If you don't specify a server name and such certificates are encountered, the CRL retrieval will fail.

[Session_Resumption] Section

The [Session_Resumption] section (Table 102) lets you specify whether session resumption is permitted and under what conditions session resumption is performed.

Note: For session resumption to work, the network access device must be configured to handle the Session-Timeout return list attribute, because the network access device must be able to tell the client to reauthenticate after the session timer has expired.

Table 104: tlsauth.aut [Session_Resumption] Syntax

Parameter	Function
Session_Timeout	Set this attribute to the maximum number of seconds you want the client to remain connected to the network access device before having to re-authenticate. • If set to a number greater than 0, the lesser of this value and the remaining resumption limit (see description below) is sent in a Session-Limit attribute to the RADIUS client on the RADIUS Access Accept response. • If set to 0, no Session-Limit attribute is generated by the plug-in. Note that this does not prevent the authentication methods performing secondary authorization from providing a value for this attribute. Default value is 0. Entering a value such as 600 (10 minutes) does not necessarily cause a full re-authentication to occur every 10 minutes. You can configure the resumption limit to make most re-authentications fast and computationally cheap.
Termination_Action	Specifies the value to return for the Termination-Action attribute sent for an accepted client. This is a standard attribute supported by most Access Points and determines what happens when the session timeout is reached. Valid values are: • -1: Do not send the attribute. • 0: Send the Termination-Action attribute with a value of 0. • 1: Send the Termination-Action attribute with a value of 1. Default value is -1. Note that this does not prevent the authentication methods performing secondary authorization from providing a value for this attribute.
Resumption_Limit	Set this attribute to the maximum number of seconds you want the client to be able to re-authenticate using the TLS session resumption feature. This type of re-authentication is fast and computationally cheap. It does, however, depend on previous authentications and may not be considered as secure as a complete (computationally expensive) authentication. Specifying a value of 0 disables the session resumption feature. Default value is 0.

Sample tlsauth.aut File

```
[Bootstrap]
LibraryName=tlsaut
h.dll  Enable=0
InitializationString=EA
P-TLS
[Server_Settings]
; Note that all trusted root certificates
; must have a .der file extension and
; must be placed in the ROOT directory
; immediately below the directory
; containing the SBR 'radius' daemon and
```

; the radius.ini file.
; Indicates the maximum TLS Message fragment
; length EAP-TLS will handle. If not
; specified, this parameter defaults to 1020.
; It can be set as high as 4096,
; but sizes over 1400 bytes are likely to cause
; fragmentation of the UDP packet
; carrying the message and some RADIUS client
; may be incapable of dealing with
; this fragmentation.
;TLS_Message_Fragment_Length = 1020
; Indicates whether or not the EAP-TLS module
; it to check whether the User Name
; provided in the RADIUS request matches the
; principal name in the client's
; certificate. The default is not to perform
; this check.
;Verify_User_Name_Is_Principal_Name = 0
; Indicates whether or not the EAP-TLS module
; should return the
; MS-MPPE-Send-Key and MS-MPPE-Recv-Key
; attribute upon successfully
; authenticating the user. The default is
; to return these attributes.
;Return_MPPE_Keys = 1
; Specifies the size of the prime to use
; for DH modular exponentiation. The
; choices are 512, 1024, 1536, 2048, 3072
; and 4096. The default is 1024 bits.
;DH_Prime_Bits = 1024
; Specifies the TLS cipher suites (in order
; of preference) that the server is
; to use. These cipher suites are documented
; in RFC 2246 and other TLS related
; RFCs or draft RFCs. The default is: 0x3C, 0x3D, 0x67, 0x6B, 0x40, 0x6A, 0x9C, 0x9D, 0x9E, 0x9F, 0xA2, 0xA3
;Cipher_Suites = 0x3C, 0x3D, 0x67, 0x6B, 0x40, 0x6A, 0x9C, 0x9D, 0x9E, 0x9F, 0xA2, 0xA3
; Specifies a profile that is to be used
; to select attributes sent back on an
; Access-Accept. The default is not to send
; any additional attributes.
; Profile = <profile-name>
[CRL_Checking]
; Specifies whether CRL checking is to be enabled.
; The default is to disable CRL checking.
; Enable = 0
; Specifies the time (in seconds) that EAP-TLS
; will wait for a CRL checking

; transaction to complete when the CRL check
; involves a CRL retrieval. When
; CRL retrieval takes longer than the
; specified time, the user's authentication
; request will result in a reject. The
; default value is 5 seconds.
; Retrieval_Timeout = 5
; Specifies the time (in seconds) after
; expiration during which a CRL is
; still considered acceptable. EAP-TLS will
; always attempt to retrieve a
; new CRL when it is presented with a
; certificate chain and it finds an
; expired CRL in its cache. EAP-TLS
; will consider the expired CRL as an
; acceptable stand-in from the time the
; CRL expires to the time the grace
; period ends.
;Expiration_Grace_Period = 0
; Specifies whether the omission of a
; CDP attribute in a non-root certificate
; is acceptable. Without a CDP attribute,
; EAP-TLS will not know where to
; retrieve a CRL from and will not be
; able to perform a revocation check on
; the certificate. The default is allow
; such certificates and to skip CRL
; checking for them.
;Allow_Missing_CDP_Attribute = 1
; Specifies what LDAP server name to
; use if the CDP contains a value that
; begins with the string "//ldap:\\\".
; This style of CDP (generated by some
; CAs does not include the identity of
; the LDAP server. Specify the name of
; the LDAP that contains the CRLs if you
; expect to encounter certificates
; with this style CDP. If you don't specify
; a server name and such certificates
; are encountered, the CRL retrieval will fail.
; Default_LDAP_Server_Name = <hostname>
[Session_Resumption]
; Specifies the maximum length of time (in seconds)
; the RAS/AP will be
; instructed to allow the session to persist
; before the client is asked
; to re-authenticate. Specifying a 0 will

```
; causethe  Session-Timeout attribute
; not to be generated by the plug-in. The default is 0.
;Session_Timeout = 0
; Specifies the value to return for the
;Termination-Actionattribute
; sent in an accepted client. If omitted in
; this file, the Termination-Action
; attribute will not be sent.
Termination_Action = 0
; Specifies the length of time (in seconds)
; during which an authentication
; request that seeks to resume a previous TLS
; session will be considered
; acceptable. Specifying 0 will cause session
; resumption support to be
; disabled. The default is 0.
Resumption_Limit = 3600
```

tlsauth.eap File

Used by: GEE, EE
Not used by: –

Note: You should use the SBR Administrator to maintain settings in the tlsauth.eap file. You should not edit the tlsauth.eap file manually.

Settings for the EAP-TLS automatic EAP helper are stored in the tlsauth.eap file. The tlsauth.eap configuration file is read each time the Steel-Belted Radius server restarts (or, if you are using the GEE edition of Steel- Belted Radius, each time the Steel-Belted Radius server receives a HUP signal).

[Server_Settings] Section

The [Server_Settings] section (Table 103) contains the settings that control the basic operation of the EAP-TLS authenticationprocess.

Table 105: tlsauth.eap [Server_Settings] Syntax

Parameter	Function
TLS_Message_Fragment_Length	Maximum TLS message length that may be generated during each iteration of the TLS exchange. Anecdotal evidence suggests that some Access Points may have problems with RADIUS responses or EAP messages that exceed the size of one Ethernet frame (1500 bytes including IP/UDP headers). The default value (1020) prevents the RADIUS challenge response (carried in a UDP packet) from exceeding one Ethernet frame. This is likely to be the safest setting. Setting a smaller value affects the number of RADIUS challenge/response round-trips required to conclude the TLS exchange. While a value of 1400 may result in 6 round-trips, a value of 500 may result in 15 round-trips. The minimum value is 500.
Verify_User_Name_Is_Principal_	Certificates issued by Microsoft's Windows 2000 Certificate Server usually include a

Parameter	Function
Name	Subject Alternative Name/Other Name attribute, where Principal Name set to something likeuser@certtest.acme.com. The MS Windows XP client that supports EAP-TLS in conjunction with 802.1X extracts this attribute value from the client's certificate and uses it to respond to the Access Point's EAP Identity Request. The Access Point, in turn, packages up this value as the RADIUS User-Name attribute in requests it sends to a RADIUS server. • If set to 1, the EAP-TLS module verifies that the contents of the RADIUS User-Name attribute match the 'Principal Name' of the certificate used to authenticate the user. • If set to 0, no such check is performed. The value should be set to 0 if the certificates used do not include a 'Principal Name' or if the client being used does not report the contents of 'Principal Name' as the user's identity in response to an EAP Identity Request. Default value is 0.
Return_MPPE_Keys	Setting this attribute to 1 causes the EAP-TLS module to include RADIUS MS-MPPE-Send-Key and MS-MPPE-Recv-Key attributes in the final RADIUS Accept response sent to the Access Point. This is necessary for the Access Point to key the WEP encryption. If the Access Point is authenticating only end-users and WEP is not being used, this attribute may be set to 0. Default value is 1.
DH_Prime_Bits	Specifies the size of the prime number that the module uses for Diffie-Hellman exponentiation. Selecting a larger prime number makes the system less susceptible to certain types of attacks but requires more CPU processing to compute the Diffie- Hellman key agreement operation. Valid values are 512, 1024, 1536, 2048, 3072, and 4096. Default value is 1024.
Cipher_Suites	Specifies the TLS cipher suites (in order of preference) that the server is to use. These cipher suites are documented in RFC 2246, "The TLS Protocol Version 1.2," and other TLS-related RFCs and draft RFCs. Default value is: 0x3C, 0x3D, 0x67, 0x6B, 0x40, 0x6A, 0x9C, 0x9D, 0x9E, 0x9F, 0xA2, 0xA3.

[Secondary_Authorization] Section

The [Secondary_Authorization] section lets you specify whether secondary authorization is performed and, if it is, what information is used in the secondary authorization request.

Table 106: tlsauth.eap [Secondary_Authorization] Syntax

Parameter	Function
Enable	Specifies whether secondary authorization checking is enabled. • If set to 0, this feature is disabled, and the EAP-TLS plug-in accepts the user upon proof of ownership of a private key that matches a valid certificate. If this setting is 0, no other settings in this section are applicable to the plug-in's operation. • If set to 1, a secondary authorization check against a traditional authentication method such as an SQL plug-in is performed. Default value is 1.
Convert_User_Name_To_Subject_	Once the EAP-TLS module has concluded its processing, it may still defer to a traditional authentication method (core or plug-in) for final authorization. To do so, it must provide

Parameter	Function
CN	a user name and password to the traditional authentication method. If set to 1, the EAP-TLS module parses the Subject attribute of the client's certificate for the least significant 'CN=' and takes the value of this attribute (for example, 'George Washington') as the user name being passed to the traditional authentication method. Important: Convert_User_Name_To_Subject_CN and Convert_User_Name_To_Principal_Name cannot both be set to 0 and cannot both be set to 1. Default value is 1.
Convert_User_Name_To_ Principal_Name	Once the EAP-TLS module has concluded its processing, it may still defer to a traditional authentication method (core or plug-in) for final authorization. To do so, it must provide a user name and password to the traditional authentication method. • If set to 0, the user name passed to the traditional authentication method is the user name retrieved from the Subject field of the client certificate (see description of Convert_User_Name_To_Subject_CN above). • If set to 1, the EAP-TLS module uses the principal name (Subject Alternate Name or Other Name) from the client certificate (for example, 'joe@acme.com') as the user name being passed to the traditional authentication method. Default value is 0. Important: Convert_User_Name_To_Subject_CN and Convert_User_Name_To_Principal_Name cannot both be set to 0 and cannot both be set to 1.
FixedPassword	By default, the secondary authorization check includes a user name but no other user credentials, because no password or similar credential for the client is available at the conclusion of the TLS handshake. Some authentication methods (Native User, LDAP, and SQL) can be configured to not require user credentials. If you plan to use secondary authorization against an authentication method (for example, LDAP) that cannot be configured to ignore the lack of user credentials, you may specify a fixed password that the plug-in uses on all secondary authorization checks. Default is to perform the check without user credentials.
Include_Certificate_Info	If set to 1, the EAP-TLS plug-in adds four attributes to the request before the secondary authorization check is performed: The Funk-Peer-Cert-Subject attribute contains the value of the Subject attribute in the client certificate. The Funk-Peer-Cert-Principal attribute contains the value of the principal name (Subject Alternate Name or Other Name) attribute of the client certificate. The Funk-Peer-Cert-Issuer attribute contains the value of the Issuer attribute in the client certificate. The Funk-Peer-Cert-Hash attribute contains a hexadecimal ASCII representation of the SHA1 hash of the client certificate. These attributes are ignored if the authentication method that will perform the authentication check does not use them. Default value is 0.

[CRL_Checking] Section

The [CRL_Checking] section (Table 106) lets you specify settings that control how Steel-Belted Radius performs certificate revocation list (CRL) checking.

Table 107: tlsauth.eap [CRL_Checking] Syntax

Parameter	Function
Enable	Specifies whether CRL checking is enabled. Default value is 0 (disabled).
Retrieval_Timeout	Specifies the time (in seconds) that EAP-TLS will wait for a CRL checking transaction to complete when the CRL check involves a CRL retrieval. When CRL retrieval takes longer than the specified time, the user's authentication request will result in a reject. Default value is 5 seconds.
Expiration_Grace_Period	Specifies the time (in seconds) after expiration during which a CRL is still considered acceptable. EAP-TLS will always attempt to retrieve a new CRL when it is presented with a certificate chain and it finds an expired CRL in its cache. • If set to 0 (strict expiration mode), EAP-TLS will not accept a CRL that has expired. • If set to a value greater than 0 (lax expiration mode), EAP-TLS will consider the expired CRL as an acceptable stand-in from the time the CRL expires to the time the grace period ends. Default value is 0 (strict expiration mode).
Allow_Missing_CDP_Attribute	Specifies whether the omission of a CDP attribute in a non-root certificate is acceptable. Without a CDP attribute, EAP-TLS will not know how to retrieve a CRL and will not be able to perform a revocation check on the certificate. • If set to false, EAP-TLS will not accept a CRL with a missing CDP attribute. • If set to true, EAP-TLS will allow such certificates and skip CRL checking for them. Default value is true.
	Specifies what LDAP server name to use if the CDP contains a value that begins with the string //ldap:\\\. This style of CDP (generated by some CAs) does not include the identity of the LDAP server. Specify the name of the LDAP that contains the CRLs if you expect to encounter certificates with this style CDP. If you don't specify a server name and such certificates are encountered, the CRL retrieval will fail.

[Session_Resumption] Section

The [Session_Resumption] section lets you specify whether session resumption is permitted and under what conditions session resumption is performed. The [Session_Resumption] section consists of the parameters listed in **Table 107**.

Note: For session resumption to work, the network access device must be configured to handle the Session-Timeout return list attribute, because the network access device must be able to tell the client to reauthenticate after the session timer has expired.

Table 108: tlsauth.eap [Session_Resumption] Syntax

Parameter	Function
Session_Timeout	Set this attribute to the maximum number of seconds you want the client to remain connected to the network access device before having to re-authenticate. • If set to a number greater than 0, the lesser of this value and the remaining resumption limit (see description below) is sent in a Session-Limit attribute to the RADIUS client

Parameter	Function
	on the RADIUS Access Accept response. • If set to 0, no Session-Limit attribute is generated by the plug-in. Note that this does not prevent the authentication methods performing secondary authorization from providing a value for this attribute. Default value is 0. Entering a value such as 600 (10 minutes) does not necessarily cause a full re-authentication to occur every 10 minutes. You can configure the resumption limit to make most re-authentications fast and computationally cheap.
Termination_Action	Specifies the value to return for the Termination-Action attribute sent for an accepted client. This is a standard attribute supported by most Access Points and determines what happens when the session timeout is reached.Valid values are: • -1: Do not send the attribute. • 0: Send the Termination-Action attribute with a value of 0. • 1: Send the Termination-Action attribute with a value of 1. Default value is -1. Note that this does not prevent the authentication methods performing secondary authorization from providing a value for this attribute.
Resumption_Limit	Set this attribute to the maximum number of seconds you want the client to be able to re-authenticate using the TLS session resumption feature. This type of re-authentication is fast and computationally cheap. It does, however, depend on previous authentications and may not be considered as secure as a complete (computationally expensive) authentication. Specifying a value of 0 disables the session resumption feature. Default value is 0.

Sample tlsauth.eap

File [Bootstrap]
LibraryName=tlsauth.
dll Enable=1
; Maximum TLS Message fragment length
TLS_Message_Fragment_Length = 1020
; Indicates whether the EAP-TLS module is to check
; whether the User Name provided in the RADIUS request
; matches the principal name in the client's certificate.
Verify_User_Name_Is_Principal_Name = 1
; Indicates whether the EAP-TLS module should return
;the MS-MPPE-Send-Keyand MS-MPPE-Recv-Key attributeupon
; successfully authenticating the user.
Return_MPPE_Keys = 1
; Specifies the size of the prime to use for DH modular
; exponentiation.
DH_Prime_Bits =
1536
[Secondary_Authori
zation]
; Indicates whether secondary authorization is to be

```
; performed. Set to 1 to require a secondary authorization
;checkagainsttraditionalauthenticationmethod
; (for example, SQL
plug-in) Enable = 1
; Indicates whether the plug-in should substitute the CN
; contained in the client certificate for the RADIUS User
; Name before the secondary authorization check
Convert_User_Name_To_Subject_CN = 1
; Indicates whether the plug-in should substitute the
; principal name contained in the Subject Alternate Name
; (Other Name) field of the client certificate for the
; RADIUS User Name before secondary authorization check.
Convert_User_Name_To_Principal_Name = 0
;Indicates whether the secondary authorization check
; should use no user credentials or a fixed password.
FixedPassword = test
;Indicates whether attributes containing information
; about the client certificate should be added to the
;request before secondary authorization is performed.
;Theattributesinclude Funk-Peer-Cert-Subject,
; Funk-Peer-Cert-Principal, Funk-Peer-Cert-Issuer, and
; Funk-Peer-Cert-Hash. The default is not to include
;these attributes.
;Include_Certificate_Info = 0
[Session_Resumption]
; Maximum length of time (in seconds) the RAS/AP will
; allow the session to persist before the client is asked
; to re-
authenticate.
Session_Timeout=
600
; The value to return for the Termination-Action attribute
; sent in an accepted client.
Termination_Action= 0
; The length of time (in seconds) during which an
;authentication request that seeks to resume a previous
; TLS session will be considered acceptable.
Resumption_Limit = 3600
```

Configuring Secondary Authorization

The EAP-TLS plug-in may be configured to perform a secondary authorization check that typically requires a traditional authentication method that can be configured to authenticate users without the presence of credentials.

Examples for the Oracle SQL plug-in, the LDAP plug-in, and Native User authentication are provided below.

SQLAuthentication

The .aut file below shows an example of how the Oracle SQL plug-in on Linux can be configured so that password information is not required as input or output.

To configure these two plug-ins to cooperate, no password has been given in the SQL= string entry in the [Settings] section, and the Password= entry in the [Results] section has been similarly left empty.

```
[Bootstrap]
LibraryName=radsql_auth_ora.s
o Enable=1
InitializationString=Oracle SQL
Auth [Settings]
; OracleInstance is database instance, ($ORACLE_SID from shell)
Connect=OracleUser/OraclePassword@OracleInstance
;Otherthanprocedures,non-interactiveSQLstatementsarenot
; terminated
SQL=SELECTFullName FROM orasqlauth WHERE username=%Name/50s
ParameterMarker=?
ConcurrentTimeout
=10
ConnectTimeout=
10
QueryTimeout=10
WaitReconnect=2
MaxWaitReconnec
t=15
; LogLevel and TraceLevel for this plugin, regardless of radius.ini
LogLevel=0
Trace
Level
=0
[Resu
lts]
;Empty definition of Password= indicates password to be ignored,
; since EAP-TLS is assumed to have already authenticated the user.
Password=
FullName=1/255s
;Profile=2/48
;Alias
=3/4
8
[Failu
re]
;Accept=0
;Profile=xyz
;FullName=Remote User
```

If the SQL authentication method used for secondary authorization is intended to be used only in conjunction with EAP-TLS, use SBR Administrator to set EAP-Only=1 and EAP-Type=TLS in the appropriate

section of the eap. ini file to prevent unintended use of this SQL authentication method for traditional authentication requests.

LDAP Authentication

The .aut file below shows an example of how the LDAP plug-in can be configured so that password information is not required as input or output.

To configure the EAP-TLS and LDAP plug-ins to cooperate properly, the BindName= option has been utilized in the [Settings] section to log into the LDAP server and no %password= setting has been specified in the [Response] section.

```
[Settings]
MaxConcurrent=2
Timeout=20
ConnectTimeout=
5
QueryTimeout=10
WaitReconnect=2
MaxWaitReconnect=
30
BindName=uid=admin,ou=administrators,o=bigco.com
Bindpassword=adminPassword
LogLevel=2
UpperCaseName=
0
PasswordCase=orig
inal
Search=DoLdapSe
arch SSL=0
[Ser
ver]
s1=
[Ser
ver/
s1]
Host=199.185.162.147
Port=
389
[Requ
est]
%Username=User-
Name [Response]
%profile=TheUserProfile
[Search/DoLdapSearch]
Base=ou=Special
Users,o=bigco.com Scope=2
Filter=(uid=<User-
Name>)
Attributes=AttrList
```

```
Timeout=20
%DN=dn
[Attributes/AttrList]
userprofile
```

If the LDAP authentication method used for secondary authorization is intended to be used only in conjunction with EAP-TLS, use SBR Administrator to set EAP-Only=1 and EAP-Type=TLS in the appropriate section of the eap. ini file to prevent unintended use of this LDAP authentication method for traditional authentication requests.

Native User Authentication

The only requirement for using EAP-TLS in conjunction with Native User authentication is that appropriate values must be set in SBR Administrator for the First-Handle-Via-Auto-EAP and EAP-Type settings in the eap.ini file.

ttlsauth.aut File

Used by: GEE, EE*
Not used by: –

Note: You should use the SBR Administrator to maintain settings in the ttlsauth.aut file. You should not edit the ttlsauth.aut file manually.

Settings for the EAP-TTLS authentication method are stored in the ttlsauth.aut file. The ttlsauth.aut configuration file is read each time the Steel-Belted Radius server restarts (or, if you are using the GEE edition of Steel- Belted Radius, each time the Steel-Belted Radius server receives a HUP signal).

[Bootstrap] Section

The [Bootstrap] section of the ttlsauth.aut file (Table 107) specifies information that Steel-Belted Radius uses to load the EAP-TTLS authentication method.

Table 109: ttlsauth.aut [Bootstrap] Syntax

Parameter	Function
LibraryName	Specifies the name of the EAP-TTLS module. Default value is ttlsauth.dll for Windows and ttlsauth.so for Linux. Do not change this unless you are advised to do so by Pulse Secure Global Support Center.
Enable	Specifies whether the EAP-TTLS authentication module is enabled. • If set to 0, EAP-TTLS is disabled, and the EAP-TTLS authentication method does not appear in the Authentication Methods list in the Authentication Policies panel. • If set to 1, EAP-TTLS is enabled. Default value is 0.
InitializationString	Specifies the name of the authentication method to appear in the Authentication Methods list in the Authentication Policies panel. The name of each authentication method must be unique. If you create additional .aut files to implement authentication against multiple databases, the InitializationString value

Parameter	Function
	in each file must specify a unique method name. Default value is EAP-TTLS.

[Server_Settings] Section

The [Server_Settings] section (Table 109) lets you configure the basic operation of the EAP-TTLS plug-in.

Table 110: ttlsauth.aut [Server_Settings] Syntax

Parameter	Function
TLS_Message_Fragment_Length	Specifies the maximum size TTLS message length that may be generated during each iteration of the TTLS exchange. This value affects the number of RADIUS challenge/ response round-trips required to conclude the TLS exchange. A value of 1400 may result in 6 round-trips, while a value of 500 may result in 15 round-trips. Some Access Points may have problems with RADIUS responses or EAP messages that exceed the size of one Ethernet frame (1500 bytes including IP/UDP headers). Minimum value is 500. Maximum value is 4096. Default value is 1020, which prevents the RADIUS challenge response (carried in a UDP packet) from exceeding one Ethernet frame.
Return_MPPE_Keys	Setting this attribute to 1 causes the module to include RADIUS MS-MPPE-Send-Key and MS-MPPE-Recv-Key attributes in the final RADIUS Accept response sent to the Access Point. This is necessary for the Access Point to key the WEP encryption. If the Access Point is authenticating only end-users and WEP is not being used, this attribute may be set to 0. Default value is 1.
DH_Prime_Bits	Specifies the size of the prime number that the module uses for Diffie-Hellman exponentiation. Selecting a larger prime number makes the system less susceptible to certain types of attacks but requires more CPU processing to compute the Diffie-Hellman key agreement operation. Valid values are 512, 1024, 1536, 2048, 3072, and 4096. Default value is 1024.
Cipher_Suites	Specifies the TLS cipher suites (in order of preference) that the server is to use. These cipher suites are documented in RFC 2246, "The TLS Protocol Version 1.2," and other TLS- related RFCs and draft RFCs. Default value is: 0x3C, 0x3D, 0x67, 0x6B, 0x40, 0x6A, 0x9C, 0x9D, 0x9E, 0x9F, 0xA2, 0xA3.
Require_Client_Certificate	• If set to 1, specifies that the client must provide acertificate as part of the TTLS exchange. • If set to 0, no client certificate is required. Default value is 0.

[Inner_Authentication] Section

Used by: GEE
Not used by: EE

The [Inner_Authentication] section (Table 110) lets you specify the way in which the inner authentication step is to operate.

Table 111: ttlsauth.aut [Inner_Authentication] Syntax

Parameter	Function
Directed_Realm	Omitting this setting causes the inner authentication request to be handled like any other request received from a RAS. Specifying the name of a directed realm causes the request to be routed based on the methods listed in the directed realm. Default is to process the inner authentication through standard request processing.

[Request Filters] Section

Request filters (Table 111) affect the attributes of inner authentication requests.

The filters named in these settings must be defined in the filter.ini file.

Table 112: ttlsauth.aut [Request Filters] Syntax

Parameter	Function
Transfer_Outer_Attribs_to_New	This filter affects only a new inner authentication request (rather than continuations of previous requests). If this filter is specified, all attributes from the outer request are transferred to the inner request and this filter is applied. The transfer occurs and the filter is applied before any attributes specified in the inner authentication are added to the request. If this filter is not specified, no attributes from the outer request are transferred to the inner request.
Transfer_Outer_Attribs_to_ Continue	This filter affects only a continued inner authentication request (rather than the first inner authentication request). If this filter is specified, all attributes from the outer request are transferred to the inner request and this filter is applied. The transfer occurs and the filter is applied before any attributes specified in the inner authentication are added to the request. If this filter is not specified, no attributes from the outer request are transferred to the inner request.
Edit_New	This filter affects only a new inner authentication request (rather than continuations of previous requests). If this filter is specified, it is applied to the inner request that is the cumulative result of attributes transferred from the outer request (see Transfer_Outer_Attribs_To_New in this table) and attributes included in the inner authentication request sent through the tunnel by the client. If this filter is not specified, the request remains unaltered.
Edit_Continue	This filter affects only a continued inner authentication request (rather than a new inner authentication request). If this filter is specified, it is applied to the inner request that is the cumulative result of attributes transferred from the outer request (see Transfer_Outer_Attribs_To_Continue in

Parameter	Function
	this table) and attributes included in the inner authentication request sent through the tunnel by the client. If this filter is not specified, the request remains unaltered.

[Response Filters] Section

Response filters (Table 112) affect the attributes in the responses returned to authentication requests.

Note: The filters named in these settings must be defined in the filter.ini file.

Table 113: ttlsauth.aut [Response Filters] Syntax

Parameter	Function
Transfer_Inner_Attribs_To_Accept	This filter affects only an outer Access-Accept response that is sent back to a network access device. If this filter is specified, the filter is applied to the inner authentication response and all resulting attributes are transferred to the outer authentication response. If this filter is not specified, no inner authentication response attributes are transferred to the outer authentication response.
Transfer_Inner_Attribs_To_Reject	This filter affects only an outer Access-Reject response that is sent back to a network access device. If this filter is specified, the filter is applied to the inner authentication response and all resulting attributes are transferred to the outer authentication response. If this filter is not specified, no inner authentication response attributes are transferred to the outer authentication response.

[CRL_Checking] Section

The [CRL_Checking] section (Table 113) lets you specify settings that control how Steel-Belted Radius performs certificate revocation list (CRL) checking.

Table 114: ttlsauth.aut [CRL_Checking] Syntax

Parameter	Function
Enable	If set to 1, specifies that CRL checking is enabled for EAP-TTLS. Default value is 0.
Retrieval_Timeout	Specifies the time (in seconds) that EAP-TTLS waits for a CRL checking transaction to complete when the CRL check involves a CRL retrieval. When CRL retrieval takes longer than the specified time, the user's authentication request is rejected. Default value is 5 seconds.
Expiration_Grace_Period	Specifies the time (in seconds) after expiration during which a CRL is still considered acceptable. EAP-TTLS always attempts to retrieve a new CRL when it is presented with a certificate chain and it finds an expired CRL in its cache.

Parameter	Function
	• If set to 0 (strict expiration mode), EAP-TTLS does not accept a CRL that has expired. • If set to a value greater than 0 (lax expiration mode), EAP-TTLS considers the expired CRL as an acceptable stand-in from the time the CRL expires to the time the grace period ends. Default value is 0 (strict expiration mode).
Allow_Missing_CDP_ Attribute	Specifies whether the omission of a CDP attribute in a non-root certificate is acceptable. Without a CDP attribute, EAP-TLS does not know how to retrieve a CRL and cannot perform a revocation check on the certificate. • If set to 0, EAP-TLS does not accept a CRL with a missing CDP attribute. • If set to 1, EAP-TLS allows such certificates and skips CRL checking for them. Default value is 1.
Default_LDAP_Server_Name	Specifies what LDAP server name to use if the CDP contains a value that begins with the string //ldap:\\\. This style of CDP (generated by some CAs) does not include the identity of the LDAP server. Specify the name of the LDAP that contains the CRLs if you expect to encounter certificates with this style CDP. If you don't specify a server name and such certificates are encountered, CRL retrieval fails.

[Session_Resumption] Section

The [Session_Resumption] section (Table 114) lets you specify whether session resumption is permitted and under what conditions session resumption is performed.

Note: For session resumption to work, the network access device must be configured to handle the Session-Timeout return list attribute, because the network access device must be able to tell the client to reauthenticate after the session timer has expired.

Table 115: ttlsauth.aut [Session_Resumption] Syntax

Parameter	Function
Session_Timeout	Set this attribute to the maximum number of seconds you want the client to remain connected to the network access device before having to re-authenticate. • If set to a number greater than 0, the lesser of this value and the remaining resump- tion limit (see description below) is sent in a Session-Limit attribute to the network access device on the RADIUS Access Accept response. • If set to 0, no Session-Limit attribute is generated by the plug-in. This does not pre- vent the authentication methods performing secondary authorization from providing a value for this attribute. Default value is 0. Entering a value such as 600 (10 minutes) does not necessarily cause a full re-authentication to occur every 10 minutes. You can configure the resumption limit to make most re-authentications fast and computationally cheap.
Termination_Action	Specifies the value to return for the Termination-Action attribute sent for an accepted client. This is a standard attribute supported by most Access Points and determines what happens when the session timeout is reached. Valid values are: • -1: Do not send the attribute.

Parameter	Function
	• 0: Send the Termination-Action attribute with a value of 0. • 1: Send the Termination-Action attribute with a value of 1. Default value is -1. Note that this does not prevent the authentication methods performing secondary authorization from providing a value for this attribute.
Resumption_Limit	Set this attribute to the maximum number of seconds you want the client to be able to re-authenticate using the TLS session resumption feature. This type of re-authentication is fast and computationally cheap. It does, however, depend on previous authentications and may not be considered as secure as a complete (computationally expensive) authentication. Specifying a value of 0 disables the session resumption feature. Default value is 0.

[Integrity_Settings]

The [Integrity_Settings] section (Table 115) specifies the list of quarantine profiles that can be used by the optional Endpoint Assurance Server software to specify how to process users designated for isolation.

```
[Integrity_Settings]
;Quarantine_Profiles=QUARANTINE        QUARANTINE2
```

Table 116: ttlsauth.aut [Integrity_Settings] Syntax

Parameter	Function
Quarantine_Profiles	Identifies the list of Steel-Belted Radius profiles that can be assigned to users designated for isolation by the Endpoint Assurance Server software. To enter more than one profile name, enter each name on the same line, separating the profile names with a space. Default value is no quarantine profiles.

Sample ttlsauth.aut File

```
[Bootstrap]
LibraryName=ttlsaut
h.dll Enable=1
InitializationString=EA
P-TTLS
; Maximum TLS Message fragment length EAP-TLS will handle.
TLS_Message_Fragment_Length = 1020
;Indicates whether the EAP-TLS module should return the
;MS-MPPE-Send-Key and MS-MPPE-Recv-Key attribute upon successful
; authentication of
user.
Return_MPPE_Key
s = 1
;Size of the prime to use for DH modular exponentiation.
DH_Prime_Bits=1536
```

```
; TLS cipher suites (in order of preference)
; that the server is to use.
Cipher_Suites = 0x3C, 0x3D, 0x67, 0x6B, 0x40, 0x6A, 0x9C, 0x9D, 0x9E, 0x9F, 0xA2, 0xA3
```

GEE only

```
[Inner_Authentication]
;Specifies how inner authentication routing operates.
Directed_Realm=        ttls_realm
[Request_Filters]
Transfer_Outer_Attribs_to_New                                    =
My_Xfer_Out_New_Filter
Transfer_Outer_Attribs_to_Continue                               =
My_Xfer_Out_Con_Filter  Edit_New=  My_Edit_New_Filter
Edit_Continue                          =
My_Continue_Filter
[Response_Filters]
Transfer_Inner_Attribs_To_Accept    =    My_Xfer_Acc_Filter
Transfer_Inner_Attribs_To_Reject    =    My_Xfer_Rej_Filter
[Session_Resumption]
; Maximum length of time (in seconds) the RAS/AP will allow
; the session to persist before the client is asked
;          to          re-
authenticate.
Session_Timeout=
600
; Value to return for the Termination-Action attribute sent
;    sent    in    an    accepted    client.
Termination_Action=  0
; Maximum length of time (in seconds) during which an authentication
; request that seeks to resume a previous TLS session will be
;considered              acceptable.
Resumption_Limit     =     3600
[Integrity_Settings]
; Specifies the list of valid quarantine profiles, which can be used
; by the Endpoint Assurance Server to specify isolated access.
; The default is no valid quarantine profiles.
;Quarantine_Profiles=QUARANTINE
```

For this to work, you must also provide the following settings in the [EAP-TTLS] section of the eap.ini file:

First-Handle-Via-Auto-EAP=0 EAP-Type = TTLS

Chapter 10

SNMP Configuration Files

This chapter describes how to configure and use the optional Simple Network Management Protocol (SNMP) package to monitor your Steel-Belted Radius server.

NOTE: SNMP is supported on the GEE edition of Steel-Belted Radius running on Linux servers. SNMP is not supported on any edition of Steel-Belted Radius running on a Windows server.

pssnmpd.conf

testagent.sh

pssnmpd.conf

Used by:

GEE Not

used by:

EE

The pssnmpd.conf file configuration file stores settings for the SNMP agent. After you install the SNMP agent for Steel-Belted Radius on a Linux server, you can modify the pssnmpd.conf configuration file to reflect your networkenvironment.

Note: When you install Steel-Belted Radius, you are prompted to enter your SNMP settings by the installation script. The installation script updates the pssnmpd.conf file based on the values you enter.

Access Control Section

The com2sec keyword maps each source/community pair to a security name. The com2sec entry is used to determine a security name from the traditional community string, taking into account where a request has come from.

The syntax for the com2sec keyword is:

com2sec *security_name source community*

where:

- *security_name* identifies the security name you want to create.
- *source* can be a hostname, a subnet, or the word default. You can specify a subnet as an IP address and mask (nnn.nnn.nnn.nnn/nnn.nnn.nnn.nnn) or as an IP address and Classless Inter-Domain Routing (CIDR) bits (nnn.nnn.nnn.nnn/nn).
- *community* is an SNMP community string, which acts as a password to authenticate

SNMP communications.

Note: If you use a CIDR address to identify a subnet, the host portion of the CIDR address must be 0. For example, if you are using the equivalent of a Class C subnet such as 192.168.1.x, you must enter the network address as 192.168.1.0/24 (which is the equivalent of 192.168.1.0/255.255.255.0).

The first source/community combination that matches an incoming packet is selected.

For example, the following creates two security names (local and mynetwork) and maps them to two different subnet/community name pairs.

	sec.name	source	community
com2sec	local	localhost	local_community
com2sec	mynetwork	192.168.1.0/24	remote_community

Security Names Section

The group keyword maps security names into group names. The group keyword gives general control by mapping between a security name (for a particular protocol version), and the internal name used in the access line.

The syntax for the **group** keyword is:

group *name model*

security where:

- *name* is the name of an access group
- *model* identifies the security model you want to use: v1 or v2c.
- *security* is a security name.

For example, the following maps the two security names to four group/model pairs.

#	sec.model	sec.name	
group	LocalGroup	v1	local
group	LocalGroup	v2c	local
group	LANGroup	v1	mynetwork
group	LANGroup	v2c	mynetwork

Access View Section

The view keyword specifies what portions of the MIB tree a specified group can view or modify. The syntax for the view keyword is:

viewname{include | excluded}*subtreemask*

where:

- *name* is the identifier used for the view.
- included/excluded lets you include or exclude specific portions of the MIB tree from the view.
- *subtree* identifies the portion of the MIB tree that this name refers to in numeric or named form.
- *mask* specifies what elements of the MIB subtree should be regarded as relevant. When the entire MIB can be viewed, you can omit the mask field.

Group Access Section

The access keyword to specify who has access to part or all of the MIB tree. The syntax for the access keyword is: access *name context model level prefix read write notify* **where:**

- *name* is the name of a group.
- *context* specifies the context for the view. For SNMPv1 or SNMPv2c, context should be empty.
- *model* is the security model: any, v1, or v2c.
- *level* can be used to ensure that the request is authenticated or encrypted. For SNMPv1 or SNMPv2c, level should be noauth.
- *prefix* specifies how the context setting should be matched against the context of the incoming PDU. Enter exact or prefix.
- *read* specifies the view to be used for READ access.
- *write* specifies the view to be used for WRITE access.
- *notify* specifies the view to be used for NOTIFY access.

For example, the following specifies that the LocalGroup uses the all view for READ, WRITE, and NOTIFY access.

#			sec mod	sec level	prefi	rea	write	notify
#		conte						
acces	LocalGroup	“”	any	noauth	exac	all	all	all
acces	LANGroup	“”	any	noauth	exac	all	non	none

System Contact Section

You can specify your system contact information in the pssnmpd.conf file or in the MIB. If you configure your system contact information in the pssnmpd.conf file, the objects are locked and cannot be modified by means of SNMP.

System contact information consists of the following:

- syslocation – The physical location of the managed device.

- syscontact – The person or department responsible for maintaining the managed device.
- sysname – The name of the managed device.

This information is stored in the system group of the MIB-II tree. The syntax for specifying system contact information is: syslocation *string*

syscontact

string

sysname

string

Traps

Section

Traps can be used by network entities to signal abnormal conditions to management stations. You should identify the NMS that receives trap messages generated by the Steel-Belted Radius server.

Note: You can configure Steel-Belted Radius to use either SNMPv1 or SNMPv2c traps. You cannot configure Steel-Belted Radius to generate both types of traps simultaneously.

The **trapcommunity** keyword specifies the default community string to be used when sending traps.

Syntax for the **trapcommunity** keyword is:

trapcommunity *string*

The **trapcommunity** keyword must precede the trap2sink keyword in the **pssnmpd.conf file.**

- The trapsink and trap2sink keywords specify whether you want Steel-Belted Radius to use either SNMPv1 traps or SNMPv2c traps. Do not enable both types of traps at the same time.
    - The **trapsink** keyword specifies the host or hosts to which the Steel-Belted Radius server should send SNMPv1 trap messages.
    - Syntax for the **trapsink** keyword is:
      trapsink host [community [port]]

where:

- *host* specifies the host name or IP address of the NMS.
- *community* specifies the community string the NMS expects.
- *port* specifies the port on which the NMS is listening for SNMPv1 trap messages.

For example:

 # send v1 traps

trapsink nms.system.com secret

- The trap2sink keyword specifies the host or hosts to which the Steel-Belted Radius server should send SNMPv2c trap (notification) messages.

  Syntax for the **trap2sink** keyword is:

  trap2sink *host [community [port]]*

where:

- *host* specifies the host name or IP address of the NMS.
- *community* specifies the community string the NMS expects.
- *port* specifies the port on which the NMS is listening for SNMPv2c trap messages.

For example:

 # send v2 traps

trap2sink nms.system.com secret

SNMP Proxy Section

The optional **proxy** keyword configures the SNMP agent to forward incoming SNMP requests to another agent. Syntax for the **proxy** keyword is:

proxy[SNMPCMD ARGS] *HOST OID [REMOTE OID]*

where: SNMPCMD keyword and arguments indicate how to authenticate the proxy SNMP.

Table 117: SNMPCMD Keywords

Command	Function
-c *community*	Set the community string for SNMP v1/v2c transactions.
-d	Dump the sent and received SNMP packets in hexadecimal format.
-r *retries*	Specifies the number of retries to be used in the request. Default value is 5.
-t *timeout*	Specifies the number of seconds between retries. Default value is 1.
-v {1 | 2c | 3}	Specifies the SNMP protocol to use. Default value is 1.

[snmp] Section

The SNMP agent uses the **pssnmpd.conf** file to store static agent configuration information, such as community strings. The SNMP agent uses the **persist** directory to store information set during the running of the agent, which needs to be persistent from one run to the next.

The **persistentDir** keyword in the [snmp] section of **pssnmpd.conf** specifies the location of the **persist directory**. By default, the **persist** directory is located in the radiusdir\snmp directory on your server.

The syntax for specifying the location of the persist directory is as follows:

[snmp]

persistentDir

radiusdir/snmp/persist [snmpd]

Section

By default, **pssnmpd** listens for incoming SNMP requests on UDP port 161 on all IP interfaces. You can specify a different UDP port in the **pssnmpd.conf** file. The syntax for specifying a listening port is as follows:

[snmpd]

agentaddress ***port_number***

Note: If you change the SNMP port number in pssnmpd.conf, you must also enter the same port number in testagent.sh.

Note: If you run more than one SNMP agent on your server, each agent must use a unique UDP port number.

Subagent Section

By default, the SNMP subagent in Steel-Belted Radius communicates with the SNMP agent on a configurable TCP port. The sbr_admin_parameters keyword specifies host, port, and interval values.

Syntax for the sbr_admin_parameters keyword is:

 # sbr_admin_parameters **host**=localhost port=***port***

tryinterval=***interval*** where:

- ***port*** identifies the TCP port the Steel-Belted Radius server uses for SNMP subagent-agent communication. The default value is TCP port 6669.
- ***interval*** specifies the number of seconds information can remain in the SNMP subagent cache. If your SNMP management station will issue queries intermittently, set the **tryinterval** value to a small number (1-5) to ensure timely information. If your SNMP management station will poll the server periodically, set the **tryinterval** value to a larger number to avoid flooding the server with queries. The default is 10 seconds.

radiusdir Section

The **sbr_private_directory** keyword specifies the location where Steel-Belted Radius is stored on your server. Syntax for the **sbr_private_directory** keyword is:

sbr_private_directory ***radiusdir***

The Steel-Belted Radius installer overwrites ***radiusdir*** with the appropriate value for your system. You should not need to change this value.

testagen

t.sh **Used**

by: GEE

Not used

by: EE

You can run the **testagent.sh** script to verify that the pssnmpd SNMP agent is functioning. Before you do so, you must configure the testagent.sh file with the community string for your network.

The syntax for the **testagent.sh** file is asfollows:

snmpget_path **-M** ***mib_directory -c community port sysDescr***

Table 118: testagent.sh Syntax

Command	Function
snmpget_path	Specifies the path for the snmpget utility. Default value is radiusdir/snmp/bin/snmpget.
-M *mib_directory*	Specifies the directory for the MIBs used by the SNMP agent. Default value is */radiusdir/snmp/mibs*.
-c *community*	Specifies the community string for your network. Default value is **COMMUNITY.**
port	Specifies the default port for SNMP traffic. Default value is **localhost:161.**
sysDescr	Specifies the MIB variable to be retrieved. Default value is system.sysDescr.0.

Chapter 11

SQL Authentication Files

This chapter describes the files used to configure SQL authentication in Steel-Belted Radius.

SQL Authentication Header Files

Used by: GEE, EE*

Not Used By: —

The header files used to configure SQL authentication methods must have the **.aut** extension; for example, **sqlauth.aut**. The format of a header file is comparable to that of a Windows **INI** file: it is composed of several sections; section names are enclosed in brackets; each section may contain multiple parameter/value pairs.

[Bootstrap] Section

The [Bootstrap] section of the SQL authentication header file (Table 117) specifies information that Steel-Belted Radius uses to load and start an SQL authentication method.

[Bootstrap]
LibraryName=sqlauth.
dll Enable=0
InitializationString=S
QL

Table 119: *.aut [Bootstrap] Syntax

Parameter	Function
LibraryName	Specifies the name of the SQL authentication module. • Enter **radsql_auth_jdbc.so** for JDBC on Linux). • Enter **SQLAUTH.DLL** for SQL on Windows.
-M*mib_directory*	Specifies whether the SQL authentication method is enabled. • If set to 0, the authentication method is disabled and does not appearin the Authentication Methods list in the Authentication Policies panel. • If set to 1, the authentication method is enabled. • Default value is 0.
InitializationString	Specifies the name of the authentication method to appear in the Authentication Methods list in the Authentication Policies panel. In the sample header file, this entry is set to SQL. You can modify this name as needed. The name of each authentication method must be unique. If you create additional .aut files to implement authentication against multiple databases,

Parameter	Function
	the InitializationString value in each file must specify a unique method name.

[FailedSuccessResultAttributes] Section

The [FailedSuccessResultAttributes] section of the SQL authentication header file (Table 118) can be used to map any RADIUS attribute returned from the database. Attributes can be specified in two ways:

- Attributes can be specified with a literal value enclosed in single quotes. Values must be enclosed with single quotes, even when they represent numeric values.
- Attributes can be specified with a numeric value that corresponds to the ordering of values returned from the SQL **select** statement.

Precede attribute names with @ and enter them as they appear in the dictionary **(.dct)** files. Enclose attribute values (including integers and IP addresses) in single quotes. For example:

[FailedSuccessResultAttributes]

@Reply-Message = 'Please re-enter your password.'

@Filter-Id = '3'

[Failure]

***Section* Used**

by: GEE

Not used by: EE

The [Failure] section of the SQL authentication header file can be used to determine the result of the authentication process (accept or reject) when connectivity to all of the configured SQL databases has failed. For example:

```
[Failur
e]
Accept
=       1
Profile
= XYZ
FullName=Unauthenticated!
```

Note: The Profile option and the Alias option cannot be used together. Read the following descriptions and choose the one that suits your needs.

Table 120: *.aut [Failure] Syntax

Parameter	Function
Accept	• If set to 1, Steel-Belted Radius returns an Access-Accept packet with the Profile, FullName, and/or Alias attributes specified in the corresponding [Failure] section parameters. • If set to 0, the user is rejected.
Profile	Specifies the name of a Steel-Belted Radius profile whose checklist and return list attributes are applied to the user's connection.
FullName	By indicating a FullName, Steel-Belted Radius returns a value in the class attribute, allowing for all [Failure] connections to be accounted.
lias	As an alternative to using the Profile parameter, you can use the Alias parameter to name an existing Steel-Belted Radius Native User entry. Steel-Belted Radius then applies the checklist and return list attributes of this User entry to the user's connection. **Note:** The Alias feature permits the Maximum Concurrent Connection limit (which is configured in the Add Users window) to be applied to the user's connection. **Note:** For security, Native User entries without passwords cannot be authenticated. Therefore, setting up Native User entries in preparation for using the Alias parameter with SQL authentication does not pose a "back door" security risk. **Note:** The Native User authentication method displayed in the Authentication Policies panel does not need to be activated for the Alias feature to work. **Note**: Individual attributes retrieved from the external database override profile attributes of the same name.
Profile	Specifies the name of a Steel-Belted Radius profile whose checklist and return list attributes are applied to the user's connection.
FullName	By indicating a FullName, Steel-Belted Radius returns a value in the class attribute, allowing for all [Failure] connections to be accounted.

[Results]

Section **Used**

by: GEE, EE*

Not used by:

—

The [Results] section of the SQL authentication header file maps the columns named in its **SELECT** query to the type of data that Steel-Belted Radius expects these columns to contain.

[Results]

Password

=1/48

Profile=2

/48

The following parameters may be present in a [Results] section. Each parameter represents a type of data required to authenticate an Access-Request, and if desired, apply authorization information as well.

Note: The Profile option and the Alias option cannot be used together. Read the following descriptions and choose the one that suits your needs.

Table 121: *.aut [Results] Syntax

Parameter	Function
%LoginLimit *(GEEonly)*	Specifies the name of the variable identifying the Maximum Concurrent Connection limits.
%Password	The value returned from this column is understood to be the user's password. The value returned by the SQL query is then matched with the user's password received in the Access-Request. By default, Steel-Belted Radius expects the user's password to stored in the SQL table in clear text format. If you want to configure Steel-Belted Radius to expect that the password value is encrypted with UNIXcrypt, then set PasswordFormat to 3 in the [Settings] section of the SQL authentication header file.
%Profile	The value returned from this column is interpreted as the name of the profile to associate with the user. The value returned by the SQL query is matched with an existing Profile entry of the same name. If the value is prof1, and a Profile called prof1 exists in the Steel-Belted Radius database, any return list or checklist attributes in prof1 are applied to the user's connection. If the value cannot be matched with an existing Profile in the Steel-Belted Radius database, the user is rejected due to "Insufficient Resources."
%ProxyRealm *(GEE)*	Specifies the realm to which the authentication must be proxied. If ProxyRealm is not set, Routed Proxy does not occur.
%ProxyUserName*(GEE)*	Specifies the User-Name attribute, which must be sent in the proxy request. If ProxyUserName is not set, the User-Name from the original request packet is used. **Note:** Enter the value for %ProxyUserName in capital letters.
%Alias	Specifies the value returned from this column that is matched with an existing Steel-Belted Radius Native User entry of the same name. For example, if the value is max1, and a native user called max1 exists in the Steel-Belted Radius database, then any return list or checklist attributes, as well as any concurrent connection limit configured for max1, are applied to the user's connection. If you want to apply concurrent connection limits to users who are being authenticated by means of SQL, you must set up a Native User entry with

Parameter	Function
	no password. **Note:** Use of %Alias is not recommended. Instead, use %Profile. **GEE:** The %LoginLimit value lets you implement the concurrent connection limits previously available through %Alias. Generally, even if a very large number of users resides in the SQL database, you need to add only one or two Native User entries to the Steel-Belted Radius database. The concurrent connection limit associated with a single Native User entry may be applied to any number of users in the SQL database. Often a Native User entry with a connection limit of 1, and a second Native User entry with a connection limit of 2, is sufficient for an entire SQL database. For example, analog users may be allowed a connection limit of 1, while ISDN users are allowed a connection limit of 2. **Note:** The Native User authentication method displayed in the Authentication Policies panel does not need to be activated for the %Alias feature to work.
%FullName	The value returned from this column is interpreted as the full name of the user. This feature is often used to distinguish the user's full name from the actual User-Name sent in the Access-Request.
RADIUS attributes	Any RADIUS attribute (preceded by an @) can be returned from the database and mapped into the [Results] section. Use attribute names as they appear in the appropriate .dct files.

Consider the following **SELECT** statement:

SELECT user_pwd, attribs, fullname **FROM** rasusers **WHERE** user_id = %name

where **user_pwd, attribs, fullname,** and **user_id** are the names of columns in the SQL table, and rasusers is the name of the SQL table itself. The [Results] section of this header file must map the SQL table columns **user_ pwd, attribs,** and **fullname** to authentication and/or authorization data types; for example.

[Results]
Passwor
d=1
Profile=2
FullNam
e=3

Columns in the SQL query are identified in the [Results] section by number; 1 represents the first column in the **SELECT** query (from left to right), and if other columns are also referenced, 2 represents the second, and 3 the third.

Along with a number representing the column order, each entry in the [Results] section also specifies the storage format of the column in the SQL table, using the same slash (/), length, and type conventions as the SQL query.

Default [Results] Parameters

The DefaultResults flag in the [Settings] section of **sqlauth.aut** specifies whether default values for **Password, Profile, Alias,** and **FullName** are automatically bound to the returned SQL data. The default **sqlauth.aut** file sets it to 0.

With **DefaultResults=0**, the results list is no longer automatically bound, and only explicit columns in the [Results] section, or embedded Parameters to a stored procedure, are used. This is the recommended setting.

The **DefaultResults=1** option remains only for backward-compatibility with old **.aut** files that rely on the default results behavior to ensure that the set of default columns are automatically bound.

[Server] Section

Used by: GEE

Not used by: EE

Steel-Belted Radius can maintain multiple SQL server connections and authenticate users against authentication databases in a round-robin fashion. This convention distributes the authentication workload across several servers.

The [Server] section of the SQL authentication header file gives Steel-Belted Radius a pool of servers from which to create the round-robin list. The [Server] section names each server that might be used. It also provides rules for when each of the possible servers should be included in (or excluded from) the round-robin list.

[Server]

ServerName=TargetNum

ber

ServerName=TargetNum

ber

.

.

Table 122: *.aut [Server] Syntax

Parameter	Function
ServerName	The name of the header file section that contains configuration information for that server..
TargetNumber	An *activationtargetnumber*, a number that controls when this server is activated for backup purposes. TargetNumber is optional and may be left blank.

A Steel-Belted Radius server maintains connectivity with its SQL servers according to the following rules:

- The priority of the server by order. The first entry in the [Server] section has the highest priority.
- By activation target number. The rule for the activation target is that if the number of SQL

servers to which Steel-Belted Radius is connected is less than the activation target, Steel-Belted Radius connects to the server and includes it in the round-robin list. While the number of active servers is equal to or greater that the activation target, Steel-Belted Radius does not use that server in the round-robin list. An activation target of 0 indicates that, in the current configuration, this machine is never used.

[Server/name] Sections

Used by:

GEE Not

used by:

EE

You must provide a [Server/***name***] section for each server you've named in the [Server] section, as follows, depending on your operating system:

- Windows:

  [Server/***name***]

  Connect=DSN=***dsnname***;UID=***username***;PWD=***password***

  where the values for ***dsnname***, ***username***, and ***password*** are specific to the SQL database you are using.

Note: Do not use the SA account or leave the password blank.

Last Resort Server

You may identify a "last resort" SQL server by providing a **LastResort** parameter in one of these [Server/name] sections, and setting its value to 1. If a SQL query against some other server results in "no record found," the authentication server tries the last resort server before accepting or rejecting the user.

In the following example, server s3 is the last resort server; in the example, the @mydb string refers to the service name for an Oracle database in the **tnsnames.ora** file (the server won't connect to the Oracle database without this).

- Windows:

  [Server]
  s1=2s

2
=
2
s
3
=
1

[Server/s1]
Connect=DSN=***dsnname***;UID=***username***;PWD=***password***

[Server/s2]
Connect=DSN=***dsnname***;UID=***username***;PWD=***password***

[Server/s3]
Connect=DSN=***dsnname***;UID=***username***;PWD=***password*** LastResort = 1

You might use the **LastResort** parameter to identify your master accounts database. This enables Steel-Belted Radius to authenticate the user in the case where a user account is newly added to the master accounts database but has not yet been propagated to all the SQL databases.

[Settings] Section

The [Settings] section of the SQL authentication header file defines parameters that control the database connection.

[Settings]
Connect=DSN=<dsn_name_here>;UID=<username_for_dB>;PWD=<password_for_dB>

SQL=SELECT password, profile FROM userlist WHERE name = %name/40
ParameterMarker=?
MaxConcurrent=1
ConcurrentTimeout=30
ConnectTimeout=25
QueryTimeout=25
WaitReconnect=2
MaxWaitReconnect=360
PasswordFormat = 0
DefaultResults = 0
ErrorMap=oracle.ini

Note: 1. "PWD" (password for DB) value of "Connect" will be overwritten with their encrypted equivalents after restart.

2. Encrypted password starts with "#ENC#" to denote that it is encrypted.

3. After encryption, If the password content needs to be modified then replace the complete encrypted

string with a new password text and restart SBR for the new encrypted string to be written.

Table 123: *.aut [Settings] Syntax

Parameter	Function
ConcurrentTimeout	Specifies the number of seconds a request may wait for execution before it is discarded. Since there may be only up to MaxConcurrent SQL statements executing at one time, new requests must be queued as they arrive until other statements are processed.
Connect	Specifies the string that must be passed to the database client engine to establish a connection to the database. This string has (or refers to) information about the name of the database, its location on the network, the password required to access it, and so forth. The format of the connect string depends on the type of database you use: Oracle: Connect=<dB_username>/<dB_password> JDBC: Connect=DSN=<jdbc:provider:driver:dsn_name_here>;UID=<username_ for_d B>;PWD=<password_for_dB>
ConnectDelimiter	(JDBC only) Specifies the character used to separate fields (DSN, UID, PWD) in the connect string. Default value is ; (semicolon). If the JDBC connect string requires use of semicolons as part of a field value, you can use this parameter to specify a different delimiter, such as ^ (caret).
ConnectTimeout	Specifies the number of seconds to wait when attempting to establish the connection to the database before timing out. This value is ignored if the client database engine does not support this feature.
DefaultResults	• If set to 0, no default values are assumed and the user must explicitly enter all result items (if you are not calling a stored procedure). • If set to 1, the default values for Results are used. This is the back- ward-compatibility setting and the setting if no value is specified in the file. In this case, each Result item must be explicitly specified.
Driver	(JDBC only) Specifies the third-party JDBC driver to load for authentication. For example: Driver=com/provider/jdbc/sqlserver/ SQLServerDriver **NOTE:** Third-party JDBC drivers must be installed in /radius/jre/lib/ext. Refer to the JDBC driver documentation for information on how to install the JDBC driver and supporting files.
ErrorMap	Specifies the name of the file that contains the native error codes that should be treated as soft errors (that is, errors that do not require Steel-Belted Radius to disconnect from and reconnect to the remote database). Note: Steel-Belted Radius includes three default error map files: mssql. ini (for Microsoft SQL using ODBC), mysql.ini (for MySQL using JDBC), and oracle.ini (for Oracle using OCI). See Chapter 8, "Database Error Map Files" for information on configuring error map files.

Parameter	Function
LogLevel	Activates logging for the SQL authentication component and sets the rate at which it writes entries to the server log file (.LOG). The LogLevel may be the number 0, 1, or 2, where 0 is the lowest logging level, 1 is intermediate, and 2 is the most verbose. If the LogLevel that you set in the .aut file is different than the LogLevel in radius.ini, the radius.ini setting determines the rate of logging. **GEE:** The LogLevel is re-read whenever the server receives a HUP signal.
MaxConcurrent	Specifies the maximum number of instances of a single SQL statement that may be executing at one time.
MaxWaitReconnect	Specifies the maximum number of seconds to wait after successive failures to reconnect after a failure of the database connection. WaitReconnect specifies the time to wait after failure of the database connection. This value is doubled on each failed attempt to reconnect, up to a maximum of MaxWaitReconnect.
ParameterMarker	Specifies the character or sequence of characters used as the parameter marker in a parameterized SQL query. Normally, this is the question mark (?), but this could vary among database vendors.
PasswordFormat	• If set to 0, Steel-Belted Radius tries to determine password format automatically. • If set to 3, Steel-Belted Radius expects the password value encrypted with UNIXcrypt. By default, the PasswordFormat parameter is not listed in the [Settings] section of the .aut file.
QueryTimeout	Specifies the number of seconds to wait for a response to a query before timing out. This value is passed to the client database engine, which may or may not implement the feature.
SQL	Specifies the SQL statement used to access the password information in the database. The SQL statement may be broken over several lines by ending each line with a backslash. The backslash must be preceded by a space character, and followed by a newline character. The subsequent lines may be indented for better readability. Example: SQL SELECT password, profile, fullname \ = FROM usertable \ WHERE username = %name/63s
SuccessResult	Specifies the string that is the expected result of a successful authentication, to be compared to the %result parameter. If a value is specified for this field, it is used in the following manner upon execution of the SQL statement: if the value of %result is not equal to the value given for this field, the user is rejected. The test for textual equality is not case sensitive. No such test, or rejection, is performed if no value is specified for this field. This is a useful technique for coordinating with the custom functionality of stored procedures.

Parameter	Function
UpperCaseName	Specifies whether the user's login name should be converted to uppercase characters before using it in the SQL statement execution. • 0 – Use the name exactly as received. • 1 – Convert the name to uppercase.
WaitReconnect	Specifies the number of seconds to wait after a failure of the database connection before trying to connect again.

[Strip] Sections

The [Strip] sections of the SQL authentication header file allow User-Name stripping to occur. These sections enable Steel-Belted Radius to identify the username that the SQL database expects by stripping the incoming User-Name attribute value of realm names and other "decorations."

You may or may not need to employ User-Name stripping for SQL authentication. Your need for this feature depends upon the naming conventions that you employ on your network and in your SQL database entries. Steel-Belted Radius's usual name parsing features work independently of this feature.

The following [Strip] syntax is available to enable and configure User-Name stripping for SQL authentication: [Strip]
Authentication=Yes

[StripPrefix]
String String
.
.

[Stri
psuf
fix]
Strin
g
Strin
g
.
.

Table 124: *.aut [Strip] Syntax

Parameter	Function
Authentication	• If set to No, prefix and suffix stripping is disabled for authentication. • If set to Yes, prefix and suffix stripping is enabled for authentication packets. When an authentication packet comes into the Steel-Belted Radius server and a SQL authentication method is active, stripping of the incoming User-Name attribute value occurs prior to SQL authentication as follows: a. Prefixes listed in the [StripPrefix] section are stripped from the incoming User-Name attribute value.

Parameter	Function
	b. Suffixes listed in [StripSuffix] are stripped. c. Any other name processing that is appropriate at this point (for example, tunnel or proxy name parsing) is performed. d. The fully stripped name is authenticated against the SQL database.
[StripPrefix]	Lists strings that are to be stripped from the beginning of the User-Name value. The strings are listed in order of priority. A string that appears earlier in the list takes precedence over later strings. In the following example, if the incoming User-Name is "seattleUser201", the stripped name is "User201". If the incoming User-Name is "seatac2000", the stripped name is "tac2000": [StripPrefix] seattle sea
String	Each String that you provide in a [Strip] section may be a character string, or a regular expression according to the following rules: ? is a wildcard character. A dash (-) indicates a range of alphanumeric characters; brackets must enclose lists of characters or ranges. For example, [A-Za-z] means any letter and [0-9.,] means any number, including decimal points and commas. A backslash (\) followed by a non-alphanumeric character indicates that character literally, for example \' indicates the question mark. \ is also used as an escape character, as follows: \a bell (7) \b backspace (8) \t tab (9) \n newline (10) \v vertical tab (11) \f formfeed (12) \r return (13) \xnn hex value, where nn are 2 hex digits \nnn decimal value, where nnn are 3 decimal digits
[StripSuffix]	Lists strings that are to be stripped from the end of the User-Name value. Conventions are the same as for [StripPrefix].

Chapter 12

SQL Accounting Files

This chapter describes the files used for SQL accounting in Steel-Belted Radius.

SQL Accounting Header (.acc) File

Used by:

GEE, EE

Not Used

By: —

The header file used to configure SQL accounting methods must have a **.acc** extension: for example, **sqlacct. acc.** The format of a header file is comparable to that of a Windows **INI** file: it is composed of several sections; section names are enclosed in brackets; each section may contain multiple parameter/value pairs.

[Bootstrap]Section

The [Bootstrap] section of the SQL accounting header file (Table 124) specifies information used to load and start the SQL accounting module.

[Bootstrap]
LibraryName=sqlacct.dl
l Enable=0
InitializationString=

Table 125: *.acc [Bootstrap] Syntax

Parameter	Function
LibraryName	Specifies the name of the SQL accounting module. Linux: Enter **radsql_acct_jdbc.**so (for JDBC). Windows: Enter **sqlacct.dll.**
Enable	Specifies whether the SQL accounting method is enabled. • If set to 0, SQL accounting is disabled. • If set to 1, SQL accounting is enabled.. Default value is 0.
InitializationString	Not used.

[Settings] Section

The [Settings] section of the SQL accounting header file defines parameters that control the database connection.

Table 126: *.acc [Settings] Syntax

Parameter	Function
ConcurrentTimeout	Specifies the number of seconds a request may wait for execution before it is discarded. Since there may be up to MaxConcurrent SQL statements executing at one time, as new requests arise they must be queued, waiting for other statements to complete. ConcurrentTimeout may be overridden for any particular statement in the [Type/statement] section for that statement.
Connect	Specifies the string that must be passed to the database client engine to establish a connection to the database. This string has (or refers to) information about the name of the database, its location on the network, the password required to access it, and so forth. The format of the connect string depends on the type of database you use: Oracle: Connect=<dB_username>/<dB_password> JDBC: Connect=DSN=<jdbc:provider:driver:dsn_name_here>;UID=<username_for_dB>;PWD=<password_for_dB> Note: 1. "PWD" (password for DB) value of "Connect" will be overwritten with their encrypted equivalents after restart. 2. Encrypted password starts with "#ENC#" to denote that it is encrypted. 3. After encryption, if the password content needs to be modified then replace the complete encrypted string with a new password text and restart SBR for the new encrypted string to be written.
ConnectDelimiter	(JDBC only) Specifies the character used to separate fields (DSN, UID, PWD) in the connect string. Default is ; (semicolon). If the JDBC connect string requires use of semicolons as part of a field value, you can use this parameter to specify a different delimiter, such as ^ (caret).
ConnectTimeout	Specifies the number of seconds to wait when attempting to establish the connection to the database before timing out. This value is passed to the client database engine, which may or may not implement the feature.
Driver	(JDBC only) Specifies the third-party JDBC driver to load for accounting. For example: Driver=com/provider/jdbc/sqlserver/SQLServerDriver NOTE: Third-party JDBC drivers must be installed in /radius/jre/lib/ext. Refer to the JDBC driver documentation for information on how to install the JDBC driver and supporting files.
LogLevel	Activates logging for the SQL accounting component and sets the rate at which it writes entries to the server log file (.LOG). The LogLevel may be the number 0, 1, or 2, where 0 is the lowest logging level, 1 is intermediate, and 2 is the most verbose. If the LogLevel that

Parameter	Function
	you set in the .acc file is different than the LogLevel in **radius.ini**, the **radius.ini** setting determines the rate of logging. Default value is 2. **GEE**: The LogLevel is re-read whenever the server receives a HUP signal.
MaxConcurrent	Specifies the maximum number of instances of a single SQL statement that may be executing at one time. MaxConcurrent may be overridden for any particular statement in the [Type/*statement]*section forthatstatement.
MaxWaitReconnect	Specifies the maximum number of seconds to wait after successive failures to reconnect after a failure of the database connection. WaitReconnect specifies the time to wait after failure of the database connection. This value is doubled on each failed attempt to reconnect, up to a maximum of MaxWaitReconnect.
ParameterMarker	Specifies the character or sequence of characters used as the parameter marker in a parameterized SQL query. Default is ? (question mark).
QueryTimeout	Specifies the number of seconds to wait for the execution of a SQL statement to complete before timing out. This value is passed to the database engine, which may or may not implement the feature. QueryTimeout may be overridden for any particular statement in the [Type/ statement] section for that statement.
UpperCaseName	Specifies whether the user's login name should be converted to uppercase characters before using it in the SQL statement execution. Set this entry to 1 to convert the name to uppercase, set it to 0 to use the name exactly as received.
UTC	This entry should be set to 0 to show time information in local time, or 1 to show time information in universal time coordinates (UTC).
WaitReconnect	Specifies the number of seconds to wait after a failure of the database connection before trying to connect again.

[Type] Sections

Each entry in the [Type] section of the SQL accounting header file maps an Acct-Status-Type attribute value to a statement name that you may assign arbitrarily. The statement name is then used to look up another section in the header file that describes that statement. The secondary section names are composed as [Type/***statement***], where ***statement*** is the arbitrarily assigned name for the statement.

For example, to perform separate accounting updates for network access device and user activity, you might provide the following [Type] and [Type/***statement***] sections:

[
T
y

p
e
]
1
=
u
s
e
r
2
=
u
s
e
r
3
=
u
s
e
r
7
=
n
a
s
8=nas
639=n
as
28=na
s

```
[Type/user]
        SQL=INSERT INTO usagelog \
                (Time,          NASAddress,
                SessionID, \    Type,  Name,
                BytesIn, BytesOut) \
VALUES \
        (%TransactionTime/t,%NASAddress,\
        @Acct-Session-Id,@Acct-Status-Type,   \
        %FullName/40s,  @Acct-Input-Octets,  \
        @Acct-Output-Octets)

[Type/nas]
        SQL=INSERT INTO ...
```

Note the numeric values used in the [Type] section above. The Acct-Status-Type values 1, 2, 3, 7, and 8 have been reserved by the RADIUS accounting standard with names and meanings, as described in Table

126.

Table 127: Acct-Status-Type Values

Acct-Status-Type Value	Name	Meaning
1	Start	A user session has started.
2	Stop	A user session has stopped, request contains final statistics.
3	Interim	A user session is in progress, request contains current statistics.
7	Accounting-On	The network access device has started.
8	Accounting-Off	The network access device is about to shut down.

Additional values for Acct-Status-Type have been defined by network access device vendors for use with their equipment. These vendor-specific values may also be listed in the [Type] section.

[Type/statement] Sections

Table 127 lists the parameters that may be present in a [Type/statement] section of the SQL accounting header file.

Table 128: *.acc [Type|statement] Syntax

Parameter	Function
SQL	Specifies the exact SQL statement used to update the SQL database with accounting information. The SQL statement may be broken over several lines by ending each line with a backslash. The backslash must be preceded by a space character, and followed by a newline. The subsequent lines may be indented for better readability. For example: **SQL=INSERT INTO accounting** **(TransTime, FullName, Authenticator, NASName,** **\ NASAddress, Type, PacketsIn, PacketsOut) ** **VALUES (%TransactionTime/t, %FullName/40s, ** **%AuthType/40s, %NASName/40s, %NASAddress, ** **%Type, @Acct-Input-Packets/n, ** **@Acct-Output-Packets/n)** **Note:** You should include the /t (timestamp) data type qualifier with the %TransactionTime argument in SQL statements. If you do not, the %TransactionTime output is formatted as character, with differing results on JDBC and Oracle.
MaxConcurrent	If present, MaxConcurrent overrides the value of MaxConcurrent specified in the [Settings] section for this particular statement.
ConcurrentTimeout	If present, ConcurrentTimeout overrides the value of ConcurrentTimeout

Parameter	Function
	specified in the [Settings] section for this particular statement.
QueryTimeout	If present, QueryTimeout overrides the value of QueryTimeout

[TypeNames] Section

Each entry in the [TypeNames] section of the SQL accounting header file maps an Acct-Status-Type attribute value to a string. If a %Type parameter is present in the corresponding SQL statement, this %Type parameter contains the given string.

If no string is given for a particular Acct-Status-Type, when an accounting request of that type is received, %Type is set to the numeric value of the Acct-Status-Type attribute, formatted as a string.

The syntax for the [TypeNames] section is as follows:

[TypeNames]
TypeID=TypeName
TypeID=TypeName
.
.
.

You can include RADIUS standard and vendor-specific accounting packet types; for example:

[TypeNames]
1=Start
2=Stop
3=Interim
7=On
8=Off
639=AscendType
28=3ComType

Working with Stored Procedures

A stored procedure is a sequence of SQL statements that form a logical unit and perform a particular task. You can use stored procedures to encapsulate a set of queries or operations that can be executed repeatedly on a database server. For example, you can code operations on an employee database, such as password lookup, as stored procedures that can be executed by application code. For more information on stored procedures, see the Steel-Belted Radius Administration Guide.

The SQL example in the previous section could be replaced by a custom stored procedure. This stored procedure might look something like the following:

```
PROCEDURE myProc
    (
        ttime in varchar2,
        nasaddr in varchar2, sessid in varchar2,
        ttype in varchar2,
        uname in varchar2, bytein in varchar2,
        byteout in varchar2
    );
END myProc;

CREATE OR REPLACE PACKAGE BODY myPack1 IS
PROCEDURE

myProc (

ttime in varchar2,
nasaddr in varchar2, sessid in varchar2,
ttype in varchar2,
uname in varchar2, bytein in varchar2,
byteout in varchar2

)

I
S
B
E
G
I
N

INSERT INTO usagelog

    ( Time, NASAddress, SessionID, Type, Name, BytesIn, BytesOut )

    VALUES
```

```
        ( ttime, nasaddr, sessid, ttype, uname, bytein, byteout );

    END
    myProc;
    END
    myPack1;
```

When you invoke the stored procedure, delineate each parameter as an input (!i), output (!o), or input/output (!io) variable.

This stored procedure can be invoked with the following connect string in the **radsql.acc** file:

```
SQL=BEGIN    myPack1.myProc(%TransactionTime!i,
        %NASAddress!i,  @Acct-Session-Id!i,  %Type!i,
        %FullName!i,  @Acct-Input-Packets!i,
        @Acct-Output-Packets!i);  END;
```

Load Balancing Example (GEE only)

The following excerpt from a .acc example file configures load balancing between two SQL servers (so that the work load is shared nearly equally between two servers). The tradeoff with this technique is that the data is split between two servers and must be reintegrated when processed. For example, the Accounting-START for an end- user may be stored on one server and the corresponding Accounting-STOP on the other.

```
[S
er
ve
r]
s1
=2
s2
=2

[Server/s1]
Connect=system/*********@thor

[Server/s2]
Connect=system/*********@odi
n

[Type]
1=Use
r
2=Use
r
3=Use
r

[Type/User]
SQL=INSERT INTO acct1(TransTime, FullName, \
        Authenticator,  NASName,  NASAddress,
```

```
        Type, \
PacketsIn, PacketsOut) \
VALUES (%TransactionTime/t, %FullName/40s, \
%AuthType/40s, %NASName/40s, %NASAddress, \
%Type, @Acct-Input-Packets/n, \
@Acct-Output-Packets/n)
```

Chapter 13

LDAP Authentication Files

This chapter describes the files used to configure LDAP authentication in Steel-Belted Radius.

LDAP Authentication Header (.aut) File

Used by:

GEE, EE

* Not Used By: —

The LDAP authentication header file is located in the same directory that contains the Steel-Belted Radius service (normally **C:\Program Files (x86)\Pulse Secure\Steel-Belted Radius\Service**) or daemon. The header file must have the extension .aut and is usually called ldapauth.aut.

The structure of the LDAP authentication header file is comparable to that of a Windows INI file. An LDAP authentication header file consists of several sections, where each section may contain multiple entries. Section names are enclosed in square brackets, for example [Bootstrap]. Each entry in the section appears on one line, and is of the form **parameter = value**. A section ends at the next section, or at the end of the file. Everything to the right of a semicolon (;) is ignored until the end of that line.

LDAP Authentication Variable Names

When Steel-Belted Radius extracts RADIUS attribute values from the incoming Access-Request and adds them to the Variable Table, the name that it gives to each variable is the same as the name of the corresponding attribute, for example User-Name or Calling-Station-ID. You may refer to the variable by this name in any subsequent entry in the .aut header file. This convention means that RADIUS attribute names are treated as reserved keywords. However, the .aut header file syntax also permits you to assign the value of an incoming RADIUS attribute to any variable.

When the LDAP Search request returns LDAP attribute values, they are added to the Variable Table. Steel-Belted Radius gives each variable the name of the corresponding LDAP attribute. In the schema illustrated above,
this would produce variable names such as User-Secret and Last-Name. For the names to use in your own .aut header file, consult your LDAP database schema. Like RADIUS attribute names, LDAP attribute names are treated as reserved keywords. However, the .aut header file syntax permits you to assign the value of a returned LDAP attribute to any variable.

[Bootstrap] Section

Used by:

GEE, EE

Not used

by: —

The [Bootstrap] section of the LDAP authentication header file specifies information that Steel-Belted Radius uses to load and start the LDAP Authentication plug-in. After you edit **ldapauth.aut** and restart Steel-Belted Radius, the InitializationString value that you entered in the [Bootstrap] section of **ldapauth.aut** appears in the Authentication Methods list in the Authentication Policies panel. You can then enable, disable, or prioritize this method like any other entry in the list.

You can configure more than one LDAP authentication method. Each requires its own .aut file in the same directory as **ldapauth.aut**. The [Bootstrap] section of each .aut file must provide a LibraryName of **ldapauth.so** (for Linux) or **ldapauth.dll** (for Windows). The InitializationString in each **.aut** file must be unique, so that you can distinguish between authentication methods in the Authentication Policies panel.

Table 129: *.aut [Bootstrap] Syntax

Parameter	Function
LibraryName	Specifies the name of the LDAP authentication module. Value must be l**dapauth.so** for Linux and **ldapauth.dll** for Windows.
Enable	• If set to 1, the LDAP authentication module is enabled. • If set to 0, the LDAP authentication module is disabled, and the authentication method is unavailable and does not appear in the Authentication Methods list in the Authentication Policies panel. Default value is 0.
InitializationString	Specifies the identifier for the authentication method, which appears in the Authentication Methods list in the Authentication Policies panel. The name of each authentication method must be unique. If you create additional .aut files to implement authentication against multiple databases, be sure that each .aut file has a unique InitializationString value. Default value is LDAP.

[Attributes/name] Sections

Used by:

GEE, EE

Not used

by: —

LDAP database entries may have many attributes, many of which may be irrelevant to the authentication process. An LDAP Search returns all of the attributes associated with an LDAP entry. Therefore, when specifying an LDAP Search for authentication purposes, you may want to provide a list of specific LDAP attributes relevant to Steel-Belted Radius. Only these attributes are placed in the Variable Table.

Each [Attributes/***name***] section in the LDAP authentication header file lists LDAP attributes relevant to a specific LDAP Search request. The syntax is as follows:

[Attributes/
name]
attribute
attribute

.

.

.

where ***attribute*** is the name of an LDAP attribute and name is an arbitrary name for the section. You must type the attribute names exactly as they appear in your LDAP database schema. Use one line per attribute. For example:

**[Attributes/InterestingAttributes
] User-
Secret
RADIUS-
Profile
Inactivity-
Timeout**

An [Attributes/***name***] section is associated with a Search request by referencing it from within a [Search/***name***] section using the Attributes parameter. For example:

[Search/DoLdapSearch]
Attributes =
InterestingAttributes

If the **Attributes** parameter is omitted from a [Search/***name***] section, Steel-Belted Radius retains all of the attributes associated with the LDAP entry. Of these attributes, Steel-Belted Radius uses only those referenced in the **.aut** header file; all others stay in the Variable Table until the authentication transaction is complete and the table is discarded.

For BindName authentication, you must ensure that the [Attributes/name] section lists the attribute in which the user's password is stored and that your [Response] section assigns the value of this attribute to the outgoing
%Password parameter. Steel-Belted Radius completes authentication by comparing the returned %Password value with the password that arrived in the Access-Request. For example:

**[Attributes/InterestingAttributes
] User-
Secret
RADIUS-
Profile
Inactivity-
Timeout**

**[Response]
%Password = User-Secret**

%Profile = RADIUS-Profile
Vendor-Specific-NAS-Attribute = Inactivity-Timeout

[Response] Section

Used by:

GEE, EE* Not

used by: —

During an authentication transaction, the [Response] section is the last section in the LDAP authentication header file to be processed. At this point in processing, all Bind and Search requests to the LDAP database have
been completed. The [Response] section tells Steel-Belted Radius what to do with the information that it has retrieved from the incoming Access-Request and from the LDAP database. The goal at this point is for Steel- Belted Radius to complete authentication and issue an Access-Response to the RADIUS client.

The [Response] section syntax is as follows:

[Response]
attribute =
variable

attribute = *variable*

.
.
.

where ***attribute*** is the name of a RADIUS attribute or other special item needed to complete authentication, and ***variable*** is the name of a variable in the Variable Table. The end result of the [Response] syntax is that the value in the variable is assigned to the attribute.

An IP pool can be returned for any attribute of the appropriate type. If the returned string appears to be an IP address (that is, in the format, **a.b.c.d**), it is considered as an IP address; otherwise, it is considered as an address pool, from which an IP address is allocated.

attribute may be the name of a RADIUS attribute, or it may be one of the following keywords, which identify various special items associated with Steel-Belted Radius. Each of these keywords begins with the percent sign (%) to distinguish it clearly from the RADIUS attributes.

Table 130: *.aut [Response] Syntax

Item	Function
%LoginLimit (GEE only)	The name of the variable specifying the Maximum Concurrent Connection limits.
%Password	For BindName authentication, you must provide a %Password entry in the [Response] section and you must assign it the value of the password

Item	Function
	attribute retrieved from the LDAP database. Steel-Belted Radius validates the password received in the Access-Request by comparing it with the value assigned to %Password. If the passwords don't match, the request is rejected. Note: The user's password may be in clear text, or encrypted with UNIXcrypt or a SHA1+Base64 hash. For Bind authentication, omit %Password. Once processing reaches the [Response] section, the password has already been validated.
%Profile	The name of a Profile entry in the Steel-Belted Radius database. If the password has been validated (by BindName or Bind), with %Profile listed in the [Response] section, %Profile may be set to any variable, for example: %Profile = **userpolicy** When the search filter is set to find a user or object in the LDAP database that includes the **userpolicy** LDAP attribute, this value is retrieved and returned to the Steel-Belted Radius database so that it may be matched with an existing Profile entry of the same name. If the userpolicy LDAP attribute is multi-valued, the first value of userpolicy is used and subsequent values are ignored. If the value of **userpolicy** is "**prof1**" and a Profile called **prof1** exists in the Steel-Belted Radius database, any return list or checklist attributes in **prof1** are applied to the user's connection. If the value returned from LDAP cannot be matched with an existing Profile in the Steel-Belted Radius database, the user is rejected due to "Insufficient Resources."
%ProxyRealm (GEE only)	The realm to which the authentication must be proxied. If ProxyRealm is not set, Routed Proxy does not occur.
%ProxyUserName (GEE only)	The User-Name attribute, which must be sent in the proxy request. If ProxyUserName is not set, the User-Name from the original request packet is used. **Note:** Enter the value for %ProxyUserName in capital letters.
%Alias	The name of a Native User entry in the Steel-Belted Radius database. If the password has been validated (by BindName or Bind), with %Alias listed in the [Response] section, %Alias may be set to any variable, for example: %Alias = userpolicy **Important**: You are strongly recommended to use %Profile, as use of %Alias has been deprecated. **GEE**: The %LoginLimit value lets you implement the concurrent connection limits previously available through %Alias. **Note**: Native User entries without passwords automatically cannot be authenticated. This is a safety feature built into Steel-Belted Radius. Therefore, setting up Native User entries in preparation for using the Alias parameter with LDAP authentication does not pose a "back door" security risk.

Item	Function
	Generally, even if a very large number of users reside in the LDAP database, you need to add only one or two Native User entries to the Steel-Belted Radius database. The concurrent connection limit associated with a single Native User entry may be applied to any number of users in the LDAP database. Often a Native User entry with a connection limit of 1, and a second Native User entry with a connection limit of 2, is sufficient for the entire LDAP database. For example, analog users may be allowed a connection limit of 1, while ISDN users are allowed a connection limit of 2. **Note**: The Native User authentication method displayed in the Authentication Policies panel does not need to be activated for the Alias feature to work.
%FullName	The fully distinguished name of the User, for Steel-Belted Radius accounting purposes. This is the exact name against which authentication was performed. Depending on what may have occurred during Steel-Belted Radius name parsing, this name may or may not be different from the value of the User-Name attribute as it originally arrived in the Access-Request.

[Search/name] Sections

Used by:

GEE, EE* Not

used by: —

Each [Search/***name***] section in the LDAP authentication header file specifies the complete details of one LDAP Search request. You can use the same Search request on various databases, because the details of the database connection are specified separately.

For BindName authentication, you must ensure that each [Search/***name***] section searches for a database entry that matches the incoming username and retrieves from it an attribute containing that user's password. Steel-Belted Radius must compare this password to the one it received in the incoming Access-Request packet.

A [Search/***name***] section may retrieve other LDAP attributes as well; however, if you are authenticating with BindName, the user's password is a minimum requirement. Use the Attributes parameter to specify the list of items you want returned.

For example:

[Search/DoLDAPSearch]
Base = ou=Special Users, o=bigco.com Scope = 1
Filter = uid=<User-Name>
Attributes = InterestingAttributes
Timeout = 20
%DN = dn

[Attributes/InterestingAttributes] User-

**Secret
RADIUS-
Profile
Inactivity-
Timeout**

**[Response]
%Password = User-Secret
%Profile = RADIUS-Profile
Vendor-Specific-NAS-Attribute = Inactivity-**

Timeout

Table 131: *.aut [Search/name] Syntax

Parameter	Function
%DN	Specifies a variable into which the distinguished name that results from the Search should be placed.
Attributes	Specifies the LDAP attributes relevant to Steel-Belted Radius, by referencing an [Attributes/*name*] section elsewhere in the same .aut file.
Base	Specifies the distinguished name (DN) of the entry that serves as the starting point for the search. This filter is a template for an LDAP distinguished name string. The filter follows conventional LDAP syntax and may be as simple or as complex as LDAP syntax permits. It may also include replacement variables from the Variable Table. Each replacement variable consists of the variable name enclosed in angle brackets (<>). Upon execution of the LDAP Search request, the value of the variable replaces the variable name.
OnFound (GEE only)	Specifies the next request section when data is found. The value of this parameter is a string, name. The name specifies an LDAP Search request by referencing a [Search/name] section elsewhere in the same .aut file. If there is no next request section, the overall operation succeeds. This can be overridden using the $reject keyword, which causes the operation to fail when data is found.
OnNotFound (GEE only)	Specifies the next request section when data is not found. The value of this parameter is a string, name. The name specifies an LDAP Search request by referencing a [Search/name] section elsewhere in the same .aut file. If there is no next request section, the overall operation fails. This can be overridden using the $accept keyword, which causes the operation to succeed when data is not found.
Search	(Optional) Specifies an LDAP Search request by referencing a [Search/name] section elsewhere in the same .aut file. Steel-Belted Radius tries this Search request next, if the current Search yields no result. Note that each [Search/name] section may contain at most one Search parameter.
Filter	Specifies the filter to apply to the search. This filter is a template for an LDAP Search string. The filter follows conventional LDAP syntax and may be as simple or as complex as LDAP syntax permits, with multiple attribute/value assertions in boolean combination. It may also include replacement variables from the Variable Table. Each replacement variable consists of the variable name enclosed in angle brackets (<>). Upon execution of the LDAP Search request, the value of the variable replaces the variable name. For example, a Search template that uses the User-Name and Service-Type attributes from

Parameter	Function
	the RADIUS request might look like this: **(&(uid = <User-Name>)(type = <Service-Type>))**
Scope	Specifies the scope of the search; 0 (search the base), 1 (search all entries one level beneath the base), or 2 (search the base and all entries beneath the base at any level).

The Search parameter can be used in one [Search/name] section after another to create a serial "chain" of Search requests. Every Search in the chain is tried. If any Search fails to return data, the Access-Request is rejected.

An example of a two-part chained Search follows:

```
[Settings]
Search = DoLdapSearch
[Search/DoLdapSea
rch] Base = ...
Filter = ...
Search = GetMoreLdapInfo

[Search/GetMoreLdap
Info] Base = ...
Scope =
... Filter
= ...
```

Search sequencing is flexible. GEE users can proceed to a new search even if the current search returns no data by using the OnNotFound parameter. All editions can override search results using the **$reject** and **$accept** keywords. The following is an example of flexible searching:

```
[Search/DoSe
arch2] Base =
o=xyz.com
Scope = 2
Filter = uid=<User-
Name> Attributes =
AttrList Timeout = 20
%DN = dn
OnFound =
DoSearch8
OnNotFound =
DoSearch9

[Search/DoSe
arch8] Base =
o=xyz.com
Scope = 2
Filter = uid=<User-Name>
```

Attributes = AttrList
Timeout = 20
%DN = dn
OnFound = DoSearch9
OnNotFound = DoSearch9

[Search/DoSearch9] Base = o=xyz.com
Scope = 2
Filter = uid=<User-Name> Attributes = AttrList Timeout = 20
%DN = dn
OnNotFound = $accept

[Request]

Section Used

by: GEE, EE*

Not used by:

—

The [Request] section of the LDAP authentication header file indicates which RADIUS attribute values Steel- Belted Radius extracts from the incoming Access-Request. Steel-Belted Radius places these values in the Variable Table before moving on to the LDAP Bind and Search requests indicated in the file.

The syntax is as follows:

[Request]
attribute = variable
attribute = variable

where ***attribute*** is the name of a RADIUS attribute or other special item associated with the incoming Access- Request, and ***variable*** is the name of a variable in the Variable Table. The end result of the [Request] syntax is that the value in the incoming attribute is assigned to this variable.

attribute may be the name of a RADIUS attribute, or it may be one of the following keywords, which identify various special items also associated with the connection request. Note that each of these keywords begins with the percent sign (%) to strongly distinguish it from the RADIUS attributes.

Table 132: *.aut [Request] Syntax

Item	Function
%OriginalUserName (GEE only)	The original full identification of the user, prior to any processing (that is, **user@realm**).
%User (GEE only)	The user portion of OriginalUserName (the section before @).
%UserName	The full user identification (user and realm strings) after all stripping and processing has been performed.
%Name	Synonym for UserName.
%EffectiveUser (GEE only)	The name of the user (the section before @) as presented to the authentication method. This may be a modified version of the original user name.
%Realm (GEE only)	The realm portion of the original user identification (the section after @) as presented to the authentication method. This may be a modified version of the original realm name.
%EffectiveRealm (GEE only)	The realm portion of the user identification as presented to the method. This may be a modified version of the original realm name.
%NASName	The name of the network access device that originated the request. This may be the name of the RADIUS Clients entry in the database or the value of the NAS-Identifier or NAS-IP-Address attribute.
%NASAddress	The address of the NAS device, in dotted notation.
%NASModel	The make/model of the NAS device, as specified in the Steel-Belted Radius database.
%Password	The PAP password.

Item	Function
%AllowedAccessHours (GEE only)	The time periods in which the user is allowed to access the network.
%RADIUSClientName	The name of the network access device, as specified in a RADIUS Clients entry in the Steel-Belted Radius database.

variable may be omitted from any [Request] entry. If so, the value in the incoming ***attribute*** is assigned to a variable named ***attribute***.

[Request]
attribute=

In the following [Request] section example, the **nasid** variable receives the value of the NAS-Identifier attribute from the request packet, the Service-Type variable receives the value of the Service-Type attribute, and the
%NASAddress variable receives the NAS address in dotted notation.

[Request]
NAS-Identifier = nasid Service-Type = %NASAddres

s= *[Defaults]*

Section Used

by: GEE, EE

Not used by: —

The [Defaults] section of the LDAP authentication header file lets you add entries to the variable table before the request is processed. You can reference these variables in your query, even if they are not supplied in the request. Any variable not listed in the [Defaults] section is initialized to a null value.

The format of each [Defaults] entry is:

variable=value

where variable is the name of a variable and value is the value you want to assign to it. For example:

[Defaults]
Default-User=SStudent

[Search/Radius]
Base = ou=people,dc=funk,dc=com
Filter=uid=<Default-User>
Scope = 2
Attributes = RadiusAttrs Timeout

= 20
%DN = dn

In this example, the **Default-User** variable is not created during request processing by the LDAP plugin. Instead, the **Default-User** variable is inserted into the variable table by the entry in the [Defaults] section, and then substituted into the Filter setting in the [Search/Radius] section.

You can use the [Defaults] section to specify values for any variable, including temporary variables and those that represent RADIUS attributes or LDAP attributes. This way, if the Access-Request packet and LDAP database do not provide Steel-Belted Radius with all of the values that it needs to respond to an Access-Request, in each case it has an acceptable alternative value that can be used instead.

You can store multiple values for any variable; if that variable is mapped to a RADIUS attribute, all values are returned in the RADIUS response. Multiple entries set within this section are considered multiple values of the same variable.

Variable values are not additive between this section and each search. Therefore, if a search returns one or more values, all current values are replaced.

Note: The [Defaults] section is the only section in the header file that lets you assign static values to variables.

[Server/name] Sections

Used by:

GEE, EE

Not used

by: —

Several sections of the LDAP authentication header file work together to configure the connection between the Steel-Belted Radius server and the LDAP database server(s) that are being used to provide external database authentication. The sections are [Server], [Server/***name***], and [Settings].

Each [Server/***name***] section of the LDAP authentication header file contains configuration information about a single LDAP server. You must provide a [Server/***name***] section for each server you've named in the [Server] section. For example:

[Server] s1=
s2=

[Server/s1] Host = ldap_1 Port = 389

.
.
.

[Server/s2]
Host = 130.4.67.1
LastResort = 1

.
.

.

Table 132 lists the settings that may be present in a [Server/name] section:

Table 133: *.aut [Server/name] Syntax

Item	Function
Bind	For Bind authentication, you must specify a Bind template in the [Settings] section of the LDAP authentication header file. The Bind template must follow conventional LDAP syntax. It may be as simple or as complex as LDAP syntax permits, with multiple attribute/value assertions in boolean combination. It may also include replacement variables from the Variable Table. Each replacement variable consists of the variable name enclosed in angle brackets (<>). Upon execution of the LDAP Bind request, the value of the variable replaces the variable name. For example, a Bind template that uses the User-Name attribute from the RADIUS request might look like this: **uid=<User-Name>, ou=Special Users, o=bigco.com**
BindName	For BindName authentication, the BindName parameter specifies the distinguished name (DN) to be used in the Bind request that connects to the LDAP server. The [Server/name] section lets you specify a unique BindName for a specific server. Use the [Settings] section to specify a default BindName to use for all servers. For Bind authentication, omit all Bind, BindName and BindPassword parameters and use the Bind parameter in the [Settings] section. See "[Settings] Section"
BindPassword	For BindName authentication, you must provide a BindPassword. The BindPassword specifies the password to be used in the Bind request that connects to the LDAP server. The [Server/name] section lets you specify a unique BindPassword for a specific server. Use the [Settings] section to specify a default BindPassword to use for all servers. For Bind authentication, omit the BindName and BindPassword parameters. Use the Bind parameter instead. Note: 1. "BindPassword" if enabled will be overwritten with their encrypted equivalents after restart. 2. Encrypted password starts with "#ENC#" to denote that it is encrypted. 3. After encryption, if the password content needs to be modified then replace the complete encrypted string with a new password text and restart SBR for the new encrypted string to be written.
Certificates	Specifies the path of the certificate database for use with SSL. This path must not end in a filename. The certificate database must be the **cert7.db** and **key3.db** files used by Netscape Communicator 4.x or later.
ConnectTimeout	Specifies the number of seconds to wait when attempting to establish the connection to the database before timing out. This value is passed to the client database engine, which may or may not implement the feature.
FlashReconnect	If the server is down when performing a Bind or a Search, setting this parameter to 1 triggers a reconnection attempt before rejecting the request. Therefore, requests are not rejected due to inactivity timeouts. This setting applies to a particular server. To apply it for all servers, place it in the [Settings] section.

Item	Function
Host	You may identify a "last resort" LDAP server by providing a LastResort parameter in one of these [Server/name] sections, and setting its value to 1. If an LDAP query against some other server results in "no record found," the authentication server tries the last resort server before accepting or rejecting the user. You might use the LastResort parameter to identify your master accounts database. This enables Steel-Belted Radius to cover the case in which a user account is newly added but has not yet been propagated to all the LDAP databases.
LdapVersion	Specifies the version of LDAP protocol, if needed to override the default given in the [Settings] section.
MaxConcurrent	Specifies the maximum number of instances of a single LDAP request thatmay be executing at one time.
MaxWaitReconnect	Specifies the maximum number of seconds to wait after successive failures to reconnect after a failure of the database connection. WaitReconnect specifies the time to wait after failure of the database connection. This value is doubled on each failed attempt to reconnect, up to a maximum of MaxWaitReconnect.
Password	Specifies the password string, which can include variables, used to specify a Bind prior to any search within a request. If this parameter is not specified, the packet's password is used.
Port	The TCP port of the LDAP server, or 0 to use the standard port. Default value is 0.
QueryTimeout	Specifies the number of seconds to wait for the execution of an LDAP request to complete before timing out. This value is passed to the database engine, which may or may not implement the feature.
Search	The value of this parameter is a string, name. The name specifies an LDAP Search request by referencing a [Search/name] section elsewhere in the same .aut file.
SSL	• If set to 0, SSL is not used over the LDAP connection. • If set to 1, SSL is used over the LDAP connection. Default value is 0.
WaitReconnect	Specifies the number of seconds to wait after a failure of the database connection before trying to connect again.

[Server]

Section Used

by: GEE, EE*

Not used by:

— **Enterprise**

Edition

The [Server] section of the LDAP authentication header file lists the LDAP server used to perform

authentication. The syntax is as follows:

[Server]
ServerName=TargetNumber

where ***ServerName*** is the name of the header file section that contains configuration information for that server, and ***TargetNumber*** is an *activation target number*, a number that controls when this server is activated for
backup purposes. ***TargetNumber*** is optional and may be left blank. For example:

[Server] s1 =

Global Enterprise Edition/Service Provider Edition

The [Server] section of the LDAP authentication header file lists the LDAP servers that may be used to perform authentication. If you are running the GEE edition of Steel-Belted Radius, you can specify more than one server in the [Server] section for load-balancing or backup. When more than one server is specified, Steel-Belted Radius authenticates against these databases in a round-robin fashion.

The syntax is as follows:

[Server]
ServerName=TargetNu
mber
ServerName=TargetNu
mber

.
.
.

where ***ServerName*** is the name of a header file section that contains configuration information for that server, and ***TargetNumber*** is an *activation target number*, a number that controls when this server is activated for backup purposes. ***TargetNumber*** is optional and may be left blank. For example:

[Server] s1 =
s2 =

[Server/s1]
.
. ;Connection details for server s1
.
[Server/s2]

.
. ;Connection details for server s2
.

A Steel-Belted Radius server maintains connectivity with its LDAP servers according to the following rules:

- The priority of the server by order. The first entry in the [Server] section has the highest priority.
- By activation target number. The rule for the activation target is that if the number of LDAP servers that Steel-Belted Radius is connected to is less than the activation target, Steel-Belted

Radius connects to the server and includes it in the round-robin list. While the number of active servers
is equal to or greater that the activation target, Steel-Belted Radius does not use that server in the round-robin list. An activation target of **0** indicates that, in the current configuration, this machine is never used.

[Settings]

Section Used

by: GEE, EE*

Not used by:

—

The [Settings] section of the LDAP authentication header file forms a basis for all Bind and Search requests to the LDAP database server(s).

Search sequencing is flexible. You can override search results using the **$reject** and **$accep**t keywords.

GEE: You can proceed to a new search even if the current search returns no data by using the **OnNotFound** parameter.

For examples of using flexible searching, see "[Server/name] Sections".

The parameters in the [Settings] section apply to all LDAP servers listed in the header file. The following parameters are usually present. If any of these parameters is not provided in the [Settings] section, the parameter assumes a system default value.

The values set in [Settings] for some parameters, such as ConnectTimeout, MaxConcurrent, or WaitReconnect, provide defaults that apply to all servers. These default values can be overridden for a particular server by entering the same parameter with a different value in a [Server/name] section.

Table 134: *.aut [Settings] Syntax

Item	Function
Bind	For Bind authentication, you must specify a Bind template in the [Settings] section of the LDAP authentication header file. The Bind template must follow conventional LDAP syntax. It may be as simple or as complex as LDAP syntax permits, with multiple attribute/value assertions in boolean combination. It may also include replacement variables from the Variable Table. Each replacement variable consists of the variable name enclosed in angle brackets (<>). Upon execution of the LDAP Bind request, the value of the variable replaces the variable name. For example, a Bind template that uses the User-Name attribute from the RADIUS request might look like this: **uid=<User-Name>, ou=Special Users, o=bigco.com**
BindName	For BindName authentication, you must omit the Bind parameter from the LDAP authentication header file. Use the BindName and BindPassword parameters instead. In the [Settings] section, BindName and BindPassword specify a default LDAP Bind template to use for all servers. You can also use BindName and BindPassword in [Server/name] sections to override this default for an individual server

Item	Function
	See "[Server/name] Sections".
ConnectTimeout	Specifies the number of seconds to wait when attempting to establish the connection to the database before timing out. This value is passed to the client database engine, which may or may not implement the feature. Default value is 25 seconds. **Note:** The value specified in this parameter can be overridden in individual [Server/name] sections of this file.
FilterSpecial CharacterHandling	• If set to 1, specifies that non-alphanumeric characters, such as (or), should be converted to an ASCII hex value preceded by a backslash when they are encountered in a user name during authentication. • If set to 0, non-alphanumeric characters are not converted during authentication.
FlashReconnect	If a server is down when performing a Bind or a Search, setting this parameter to 1 triggers a reconnection attempt before rejecting the request. Therefore, requests are not rejected due to inactivity timeouts. This setting applies to all servers. To apply it for a particular server, place it in the appropriate [Server/name] section.
LdapVersion	Specifies the version of LDAP protocol. Default value is 2.
LogLevel	Activates logging for the LDAP authentication component and sets the rate at which it writes entries to the Steel-Belted Radius server log file (.LOG). This value may be the number 0, 1, or 2, where 0 is the lowest logging level, 1 is intermediate, and 2 is the most verbose. If the LogLevel that you set in the .aut file is different than the LogLevel in radius.ini, the radius.ini setting determines the rate of logging. **GEE**: The LogLevel is re-read whenever the server receives a HUP signal.
MaxConcurrent	Specifies the maximum number of instances of a single LDAP request that may be executing at one time. **Note**: The value specified in this parameter can be overridden in individual [Server/name] sections of this file.
MaxScriptSteps	Specifies the maximum number of seconds to wait after successive failures to reconnect after a failure of the database connection. WaitReconnect specifies the time to wait after failure of the database connection. This value is doubled on each failed attempt to reconnect, up to a maximum of MaxWaitReconnect. NOTE: The value specified in this parameter can be overridden in individual [Server/name] sections of this file.
OnFound (GEE only)	Specifies the next request section when data is found. The value of this parameter is a string, *name*. The name specifies an LDAP Search request by referencing a [Search/*name*] section elsewhere in the same **.aut file**. If there is no next request section, the overall operation succeeds. This can be overridden using the $reject keyword, which causes the operation to fail when data is found.
OnNotFound **(GEE only)**	Specifies the next request section when data is not found. The value of this parameter is a string, name. The name specifies an LDAP Search request by referencing a [Search/name] section elsewhere in the same .aut file. If there is no next request section, the overall operation fails. This can be overridden using the **$accept** keyword, which causes the operation to succeed when data is not found.

Item	Function
Password	Specifies the password string, which can include variables, used to specify a Bind prior to any search within a request. If this parameter is not specified, the packet's password is used.
PasswordCase	• If set to U or Upper, the password returned from the LDAP database is converted to upper-case before authentication. • If set to L or Lower, the password is converted to lowercase. • If set to O or Original, the password is not altered before authentication. Default value is Original.
PasswordFormat	By default, the PasswordFormat parameter is not listed in the [Settings] section of the LDAP authentication header file. With no listing, Steel-Belted Radius expects the user's password in the LDAP table to be in cleartext format. If you want to configure Steel-Belted Radius to automatically handle password values correctly when it detects that they have been encrypted using UNIXcrypt or a SHA1+Base64 hash, set PasswordFormat to auto.
QueryTimeout	Specifies the timeout value in seconds for an individual search performed against the LDAP server. Default value is 10 seconds.
ScriptTraceLevel	Specifies the level of detail for line-by-line script tracing in the log. • If set to 0, no traces are logged. • If set to 1, traces are only logged when the SbrTrace() function is executed by the script. • If set to 2, a trace is generated for every line executed by the script. Default value is 0.
Search	• If set to 0, SSL is not used over the LDAP connection. • If set to 1, SSL is used over the LDAP connection. Default value is 0. NOTE: The value specified in this parameter can be overridden in individual [Server/name] sections of this file.
Timeout	Specifies the maximum number of seconds for the overall timeout for each request, which includes the delay in acquiring resources, attempts against multiple LDAP servers, and so forth. Default value is 20 seconds.
UpperCaseName	• If set to 0, time values are displayed using the local time. • If set to 1, time values are displayed using universal time coordinates (UTC).
WaitReconnect	Specifies the number of seconds to wait after a failure of the database connection before trying to connect again. **Note:** The value specified in this parameter can be overridden in individual [Server/name] sections of this file.

[Failure] Section

Used by: GEE

Not used by: EE

The [Failure] section of the LDAP authentication header file (Table 133) can be used to determine the result of the authentication process (accept or reject) when connectivity to all of the configured LDAP databases has failed. For example:

[Failure]
Accept =
1 Profile
= XYZ
FullName = Mr Stanley Smith

Note: The Profile option and the Alias option cannot be used together. Read the following descriptions and choose the one that suits your needs.

Table 135: *.aut [Failure] Syntax

Item	Function
Accept	• If set to 1, Steel-Belted Radius returns an Access-Accept packet with the Profile, FullName, and/or Alias attributes specified in the corresponding [Failure] section parameters. • If set to 0, the user is rejected.
Profile	This is the name of an existing Steel-Belted Radius Profile entry, whose checklist and return list attributes are applied to the user's connection.
FullName	This string is the full user name, which is used in the Class attribute in the Access-Accept message.
Alias	As an alternative to using the Profile parameter, you can use the Alias parameter to name an existing Steel-Belted Radius Native User entry. Steel-Belted Radius then applies the checklist and return list attributes of this User entry to the user's connection. **Note**: The Alias feature permits the Maximum Concurrent Connection limit (settable in the Users panel) to be applied to the user's connection. **Important**: You are strongly recommended to use Profile, as use of Alias has been deprecated. The LoginLimit value lets you implement the concurrent connection limits previously available through Alias. If you want to apply concurrent connection limits to users who are being authenticated by means of LDAP, you must set up a Native User entry specifically for this purpose, with all of the appropriate checklist and return list attributes, and with no password. You can set up as many such accounts as you require. These entries store a specific set of checklist and return list attributes for LDAP authentication, for use only with the Alias parameter. **Note**: Native User entries without passwords cannot be authenticated. This is a safety feature built into Steel-Belted Radius. Therefore, setting up User entries in preparation for using the Alias parameter with LDAP authentication does not pose a "back door" security risk. NOTE: The Native User authentication method displayed in the Authentication Policies panel does not need to be activated for the Alias feature to work.

Chapter 14

TACACS+ Configuration (tac_plusd.cfg) File

This chapter describes the files used to configure TACACS+ in Steel-Belted Radius.

Used by: GEE

Not Used By: EE

TACACS+ configuration file is in the same directory that contains the Steel-Belted Radius process or daemon. The file must have the extension .cfg and is usually called tac_plusd.cfg file. TACACS+ feature is supported only in Linux platform and not Windows platform.

A single configuration file is sufficient for configuring "tac_plus".

Configuration Syntax

The configuration section is covered by the following sections are to be placed between

id = tac_plus = {
and the corresponding
}

Note:
Comments in Configuration Files:
Comments can appear anywhere in the configuration file, starting with # character and extending to the end of the current line. Should you need to disable this special meaning of the # character, e.g. if you have a password containing a # character, simply enclose the string containing it within double quotes.

The recommended order for writing a configuration file is

1. Global Definitions
2. Hosts
3. Users and Groups

The reasoning behind that non-random order is that parts of the configuration may use other parts, and this need to exist before being used.

Global Definitions

The global configuration section may contain the following configuration directives,

Limits and Timeouts

Several global limits and timeouts may be specified exclusively at global level: retire limit = n

The daemon instance will terminate after processing 'n' requests. The spawnd instance will spawn a new instance if necessary.
Default:
unset

retire
timeout =
s

The daemon instance will terminate after 's' seconds. spawnd will spawn a new instance if necessary.
Default: unset

User backend options

These options are relevant for configuring the MAVIS user

backend: user backend = mavis

Get user data from the MAVIS backend. Without that directive, only locally defined users will be available and the MAVIS backend may be used for authenticating known users (with password = mavis or similar) only.

pap backend = mavis [prefetch]

Verify PAP passwords using the MAVIS backend. This needs to be set to either mavis or prefetch to authenticate PAP requests using the MAVIS backend. If unset, the PAP password from the users' profile will be used.

If prefetch is specified, the daemon will first retrieve the users' profile from the backend and then authenticate the user based on information eventually found there.
This directive implies user backend =

mavis login backend = mavis

[prefetch] [chpass]

Verify Login passwords using the MAVIS backend. This needs to be set to either mavis or prefetch to authenticate login requests using the MAVIS backend. If unset, the login password from the users' profile will be used.

If prefetch is specified, the daemon will first retrieve the users' profile from the backend and then authenticate the user based on information eventually found there.
This directive implies user backend = mavis
For non-local users, if the chpass attribute is set and the user provides an empty password at login, the user is given the option to change his password. This requires appropriate support in the MAVIS backend modules.

Hosts

The TACACS+ daemon (tac_plus) will talk to known NAS addresses only. Connections from unknown addresses will be rejected. If tac_plus must encrypt its packets, then an encryption key must be specified. The identical key must also be configured on any NAS which communicates with tac_plus.

To specify a global key, use a statement like

host = 0.0.0.0/0 {
key = "your key
here"
}
or alternatively

host = world {
key = "your key
here" address =

0.0.0.0/0
}
where world is not a keyword, but just some arbitrary character string.

Note: Double quotes are required if the key contains spaces. The TACACS+ daemon will reject connections from hosts that have no encryption key defined. Double quotes within double-quoted strings may be escaped using the backslash character \ (which can be escaped by itself),
e.g.:
key = "quo\\te me\"."
translates to the ASCII sequence quo\te me".

Any CIDR range within a host definition needs to be unique, and the most specific definition will match. Host object attributes (key, prompt, enable passwords) may only exist once.

Generally, the syntax for host declarations conforms

to host =name_or_ip-range { key-value pairs }

In most cases, hosts can be referred to either by IP address or by name. The key-value pairs permitted in host sections of the configuration file are explained below.

key = string

This sets the key used for encrypting the communication between server and

NAS. address = cidr

Adds the address range specified by CIDR to the current host

definition. enable [level] = (permit | deny | login | (clear | crypt)

password)

This directive may be used to set host specific enable passwords, to use the login password or to permit (without password) or refuse any enable attempt level defaults to 15.
Enable passwords specified at host level have a lower precedence as those defined at user or group level.

Users and Groups

User and group declarations are similar. The major difference is, that groups are static, while user declarations may be added at run-time via the MAVIS back-end.

A user may be member of a group (also known as role or profile). Actual group membership may depend on various factors, e.g. the NAS the user is on, the NAC or time ranges. Each group may in turn be member of another group and so on.

The basic form of a user declaration is user = username
{ ... }

The basic form of a group declaration is group = groupname { ... }

A user or group declaration may contain key-value pairs and service declarations.

User-only options

The following declarations are valid in user context

only: login = ((clear | crypt) password | mavis |

permit | deny)

The login password authenticates shell log-ins to the server. login = crypt aFtFBT4e5muQE
login = clear Ci5c0

For crypt, DES- or MD5-hashed passwords may be used.
If the mavis keyword is used instead, the password will be looked up via the MAVIS backend. It will not be cached. This functionality may be useful if you want to authenticate at external systems, despite static user declarations in the configuration file.

pap = ((clear | crypt) password | mavis | permit | deny)
The pap authenticates PAP log-ins to the server. Just like with login, the password doesn't need to be in clear text, but may be DES (or MD5) hashed, or may be looked up via the MAVIS backend.

arap = (clearpassword | permit | deny)
For ARAP authentication, a cleartext password is required.

chap = (clearpassword | permit | deny)
For CHAP authentication, a cleartext password is required.

ms-chap = (clearpassword | permit | deny)
For MS-CHAP authentication, a cleartext password is required.

opap = (clearpassword | permit | deny)
For outgoing PAP authentication, a cleartext password is required.

password = (clearpassword | permit | deny)
This directive sets all previously undefined passwords (with the exception of OPAP) to the given cleartext password. This directive is retained for backwards compatibility only, and usage is deprecated.

password [acl [not] acl] { ... }
This directive allows specification of ACL-dependent passwords. Example:

```
acl jumpstation = { nac ==
10.255.0.85 } user = marc {
password acl
jumpstation { login =
permit
pap = permit
}
password {
login = clear
myLoginPassword pap
= clear myPapPassword
}
}
```

User and Group options

The following key-value pairs are valid in both user and group context:

Group membership

Group membership is specified using member directive:

As group membership is resolved when parsing the configuration file, groups and hosts need to be already defined before being used as argument in a member definition.

Just like users, groups can be member of (other) groups:

```
group = test1 {
... } group =
test2 { ... }

group = test2 {
    ...
    member = test1
    member =
    test2@nas1
    ...
}
```

Service Restrictions

default service = (permit | deny)
This defines whether a service not explicitly defined in the user profile should be permitted or denied.

prohibit service = service ...
Use this to override the default service specification. E.g., to explicitly deny the X.25 service.

prohibit service = x25

Service Definitions

Service definitions may appear in user and group sections.

There are a couple of generic configuration attributes which may appear in arbitrary service definitions.

default attribute = (permit | deny)
This directive specifies to accept or reject unknown attributes sent by NAS (default: deny).

(set | optional) attribute = value
Defines mandatory and optional attribute-value pairs. Example as follows, set priv-lvl = 15

Other configuration attributes are service specific and only valid in certain contexts. SHELL (EXEC) Service
Shell startup should have an appropriate service defined.
service = shell { }

Valid configuration directive within the curly brackets are, default cmd = (permit | deny)
This directive specifies to accept or reject unknown attributes sent by NAS (default:

deny). Non-Shell Services

For Juniper Networks-specific authorization service, use

```
service = junos-exec {
    set local-user-name = NOC
    # see the Junos documentation for more attributes
}
```

Likewise, for Raritan Dominion SX IP Console Servers:

```
service = dominionsx {
    set port-list = "1 3 4  15"
    set user-type = administrator # or operator, or observer
}
```

Examples

Here we declare two users Fred and Lily and two groups of admin and staff.
Fred is a member of group admin and group admin is in turn a member of group staff. Lily is not a member of any group.

```
group = admin {
    # group admin is a member of
    group staff  Member = staff
    Service =
        shell {
        Set priv-
        lvl = 15
    }
}

Group = staff {
    # group staff is not a member of any group
}
user = fred {
    # fred is a member of group admin on
    0.0.0.0/0  member = admin

    # fred is a member of group staff when logging in on 10.0.0.0/8
    member = staff@10.0.0.0/8

    # fred is a member of group admin when logging in on hosts defined in hostgroup
    test123  member = admin@test123

    # fred may only log in from a client in 172.16.0.0/24
    ...  client = 172.16.0.0/24

    # ... or from whatever address is defined in some host object
    test123  client = test123
}
```

Configuring Non-Local Users via MAVIS

MAVIS configuration is optional. It is not needed if satisfied with user configuration in the main configuration file. MAVIS backends may dynamically create user entries based on external authentication such as LDAP. For PAP and LOGIN, in the global section delegate authentication to the MAVIS sub-system. Statically defined users are still valid and have a higher precedence.

```
Pap    backend    =
mavis          Login
backend = mavis
```

Configuring Local Users for MAVIS Authentication

Under certain circumstances, it may be desired to keep the user definition in the plain text configuration file, but authenticate against some external system such as LDAP. To do so, specify one in the corresponding user definition

Login = mavis Pap = mavis Password = mavis

Configuring User Authentication

User Authentication can be specified separately for PAP, ARAP, CHAP and normal logins. ARAP, CHAP and global user authentication must be given in clear text. The following assigns the user mary five different passwords for ARAP, inbound and outbound CHAP, inbound PAP, outbound PAP and normal login respectively.

```
User = mary {
    Arap = clear "arap
    password" Chap = clear
    "chap password"
    Pap = clear "inbound pap
    password" Opap = clear
    "outbound pap password" Login =
    crypt XQj4892fjk
}
```

If user backend = mavis is configured in the global section, users not found in the configuration file will be looked up by the MAVIS backend. This option can be used in conjunction with more sophisticated backends such as LDAP or whenever it is desired not to duplicate pre-existing database user data to the configuration file. For users looked up by the MAVIS backend, will cause PAP and/or Login authentication to be performed by the MAVIS backend ignoring any corresponding password definitions in the user's profile.

Pap backend = mavis And/or
Login backend = mavis

If the users defined in configuration file must authenticate using the MAVIS backend, simply set the corresponding PAP or Login password field to mavis (user backend = mavis need not be added in this case)
User = mary {login = mavis}

Configuring Authorization

Authorization must be configured on both the NAS and SBR (TACACS+) to operate correctly. By default, the NAS will allow everything until it is configured to make authorization requests to SBR (TACACS+). On SBR (TACACS+) by default, deny authorization of anything isn't explicitly permitted. Authorization allows the TACACS+ to deny commands and services outright, or to modify commands and services on a per-user basis. Authorization is divided into two separate parts that is commands and services.

Authorization Algorithm

The complete algorithm by which the daemon processes its configured AV pairs against the list the NAS sends, is given below.

Find the user (or group) entry for this service (and protocol), then for each AV pair sent from the NAS:

1. If the AV pair from the NAS is mandatory:
    a. Look for an exact attribute, value match in the user's mandatory list. If found, add the AV pair to the output.
    b. If an exact match doesn't exist, look in the user's optional list for the first attribute match. If found, add the NAS AV pair to the output.
    c. If no attribute match exists, deny the command if the default is to deny, or,
    d. If the default is permit, add the NAS AV pair to the output.
2. If the AV pair from the NAS is optional:
    a. Look for an exact attribute, value match in the user's mandatory list. If found, add DAEMON's AV pair to output.
    b. If not found, look for the first attribute match in the user's mandatory list. If found, add DAEMON's AV pair to output.
    c. If no mandatory match exists, look for an exact attribute, value pair match among the daemon's optional AV pairs. If found add the DAEMON's matching AV pair to the output.
    d. If no exact match exists, locate the first attribute match among the daemon's optional AV pairs. If found add the DAEMON's matching AV pair to the output.
    e. If no match is found, delete the AV pair if the default is deny, or
    f. If the default is permit add the NAS AV pair to the output.
3. After all AV pairs, have been processed, for each mandatory DAEMON AV pair, if there is no attribute match already in the output list, add the AV pair (but add only ONE AV pair for each mandatory attribute).

Appendix A

Authentication Protocols

This appendix provides a matrix of authentication methods and their supported authentication protocols.

Table 136: Authentication Protocols

Method	PAP	CHAP	MS-CHAP	MS-CHAP-V2	LEAP	EAPM-SCHAP-V2	EAP-MD5	PAP/ Token card	EAP/ Token card
Microsoft PEAP available inner authentication protocols	No	No	No	No	No	Yes	No	No	No
Cisco PEAP available inner authentication protocols	No	No	No	No	Yes	Yes	Yes	No	Yes
TTLS available inner authentication protocols	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Local	Yes	Yes	Yes	Yes	Yes	Yes	Yes	N/A	N/A
Local (password saved as Allow PAP only, {SHA} or {crypt}).	Yes	No	No	No	No	No	No	N/A	N/A
Windows Domain Authentication									
Windows Domain Group	Yes	No	Yes	Yes	Yes	Yes	No	N/A	N/A
Windows Domain User	Yes	No	Yes	Yes	Yes	Yes	No	N/A	N/A
UNIX authentication methods									
UNIX User	Yes	No	No	No	No	No	No	N/A	N/A
UNIX Group	Yes	No	No	No	No	No	No	N/A	N/A
Other Authentication Plug-ins									
RSA SecurID	Yes	No	No	No	No	No	No	N/A	Yes
TACACS+	Yes	Yes	No	No	No	No	Yes	N/A	N/A
LDAP									
BIND (this includes AD and eDirectory/NDS)	Yes	No	No	No	No	No	No	N/A	N/A

Method	PAP	CHAP	MS-CHAP	MS-CHAP-V2	LEAP	EAPM-SCHAP-V2	EAP-MD5	PAP/ Token card	EAP/ Token card
BINDNAME (password stored in clear text)	Yes	Yes	Yes	Yes	Yes	Yes	Yes	N/A	N/A
BINDNAME (password stored in SHA Crypt text)	Yes	No	No	No	No	No	No	N/A	N/A
BINDNAME (password stored as MD4 hash of unicode value text)	Yes	No	No	Yes	No	Yes	No	N/A	N/A
BINDNAME (password stored as enc-md5)	Yes	Yes	No	No	No	No	No	N/A	N/A
SQL									
Password stored in clear text	Yes	Yes	Yes	Yes	Yes	Yes	Yes	N/A	N/A
Password password stored in SHA Crypt text	Yes	No	No	No	No	No	No	N/A	N/A
Password stored as {MD4} hash of unicode value text	Yes	No	No	Yes	No	Yes	No	N/A	N/A
Password stored as {enc- md5}	Yes	Yes	No	No	No	No	No	N/A	N/A

Appendix B

Vendor-Specific Attributes

Table 136 describes the vendor-specific attributes used with core Steel-Belted Radius.

Table 137: Steel-Belted Radius Vendor-Specific Attributes

Attribute Name	Purpose
Funk-Allowed-Access-Hours	May be placed in the checklist for a user or profile entry to control the exact time periods during which a user may be allowed access. Funk-Allowed-Access-Hours is a variable-length string that identifies time periods in a 7-day week of 24-hour days. This string consists of one or more day specifiers (each of which may list one or more days and/or ranges of days) followed by one or more ranges of 24-hour times, in minutes.
Funk-Concurrent-Login-Limit	Reserved for use by plug-ins and the Service Level Manager module for the Service Provider edition of Steel-Belted Radius.
Funk-Full-User-Name	Reserved for use by plug-ins and the Service Level Manager module for the Service Provider edition of Steel-Belted Radius.
Funk-Location-Group-Id	Added to an inbound authentication or accounting request when the request is matched to a location group and AddFunkLocationGroupIdToRequest is set to 1 in the radius.ini file. The value of the attribute is the name of the location group. **Note**: The Funk-Location-Group-id attribute is case-sensitive.
Funk-Peer-Cert-Hash	Added to the list of attributes available for secondary authorization processing when EAP-TLS is loaded as an automatic EAP helper. The attribute is added to the request only if Include_ Certificate_Info in the [Secondary_Authorization] section of tlsauth.eap is set to 1. The value of the attribute is the hexadecimal ASCII representation of the SHA1 hash of the client's certificate.
Funk-Peer-Cert-Issuer	Added to the list of attributes available for secondary authorization processing when EAP-TLS is loaded as an automatic EAP helper. The attribute is added to the request only if Include_ Certificate_Info in the [Secondary_Authorization] section of tlsauth.eap is set to 1. The value of the attribute is the contents of the Issuer attribute of the client's certificate.
Funk-Peer-Cert-Principal	Added to the list of attributes available for secondary authorization processing when EAP-TLS is loaded as an automatic EAP helper. The attribute is added to the request only if Include_ Certificate_Info in the [Secondary_Authorization] section of tlsauth.eap is set to 1. The value of the attribute is the contents of the Subject Alternate Name or Other Name attribute of the client's certificate.
Funk-Peer-Cert-Subject	Added to the list of attributes available for secondary authorization processing when EAP-TLS is loaded as an automatic EAP helper. The attribute is added to the request only if Include_ Certificate_Info in the [Secondary_Authorization] section of tlsauth.eap is set to 1. The value of the attribute is the contents of the Subject attribute of the client's certificate.
Funk-Round-Robin-Group	May be placed in the return list for a user or profile entry to dynamically assign an attribute set from an Attribute Value Pool at log-in time. The value of this attribute must be set to the .rr file name which defines the Attribute Value Pool.

Attribute Name	Purpose
Funk-Source-IP-Address	Added to the list of attributes available for request processing if AddSourceIPAddressAttrToRequest is set to 1 in the [Configuration] section of the radius.ini file. The value of the attribute is the IP address from which the packet containing the request was received.
Funk-Source-IPv6-Address	Reserved for future use.
Funk-Tribe-Name	Reserved for the Service Level Manager module to Steel-Belted Radius/Service Provider Edition.

Appendix C

SNMP Traps and Statistics

Steel-Belted Radius contains support for setting and retrieving configuration information via standard SNMP utilities. This appendix summarizes the proprietary SNMP traps and rate statistics generated by Steel-Belted Radius.

- The fnkradtr.mib MIB defines the content of the traps that are generated by the Steel-Belted Radius server for SNMPv1.
- The fnkradtr-v2.mib MIB defines the content of the traps that are generated by the Steel-Belted Radius server for SNMPv2.
- The fnkrate.mib MIB defines the peak, current, and average rate statistics maintained by Steel-Belted Radius.

Note: The SNMP subagent in Steel-Belted Radius may generate traps that do not reference Funk enterprise IDs. For information on generic SNMP traps specified by IETF-specified MIBs, refer to the appropriate RFC. For information on generic netSnmp traps specified by netSnmp-specific MIBs, refer to the netSnmp documentation.

Trap Variables

Table 137 lists the trap variables for the proprietary SNMP traps used by Steel-Belted Radius.

Table 138: Trap Variables

Variable Name	Identifies
funkSbrTrapVarComp	The component within the SBR server that issued the trap. 1 – Core 2– Accounting 3– Authentication
funkSbrTrapVarSev	The severity of the event that caused the trap. 1 – Informational 2 – Warning 3 – Error
funkSbrTrapVarSWName	The identity of the software that is the RADIUS server.
funkSbrTrapVarThreadsAvail	The number of threads available in the thread worker pool.

Variable Name	Identifies
funkSbrTrapVarBytesAvail	The number of bytes available in the file system.
funkSbrTrapVarPrivateDir	The file system path to the private directory used by the RADIUS server.
funkSbrTrapVarNumberOfOccurrences	The dilution factor for the trap. The trap is sent on once for every 'n' occurrences of this event.
funkSbrTrapVarSQLConnects	The number of connection attempts to a SQL database.
funkSbrTrapVarSQLDisconnects	The number of disconnects from a SQL database (due to an error encountered during an operation).
funkSbrTrapVarSQLTimeouts	The number of timeouts encountered when trying to perform a transaction against a SQL database.
funkSbrTrapVarServiceDispatcherErrCode	The error code returned in response to an attempt to start the RADIUS service on Windows.
funkSbrTrapVarSetStatusErrCode	The error code returned in response to an attempt to inform the service control dispatcher of the status of the RADIUS service on Windows.
funkSbrTrapVarGetDiskFreeSpaceErrCode	The error code returned in response to an attempt to call GetDiskFreeSpaceEx to determine the amount of free disk space available on Windows.
funkSbrTrapVarIniString	The .ini file setting used to specify a configuration value.
funkSbrTrapVarDbType	The type of database being employed by the RADIUS server.
funkSbrTrapVarFailedSystemName	The name of the remote system failing connectivity from the RADIUS server.
funkSbrTrapVarUserName	The name of the user to whom the trap refers.
funkSbrTrapVarPersistStoreName	The name of the persistent storage to which the trap refers.
funkSbrTrapVarDiagnosticMessage	A generic diagnostic message that may be helpful in determining and addressing the possible root causes of the trap.
funkSbrTrapVarIPAddrPoolName	The name of the IP address pool to which the trap refers.
funkSbrTrapVarIPAddrAvail	The number of addresses available in the IP address pool.
funkSbrTrapVarConnectedSystemName	The name of the remote system with which the RADIUS server has established a connection.
funkSbrTrapVarQueueName	The name of a queue in the RADIUS server. Implemented in Steel-Belted Radius version 5.3

Trap Definitions

Table 137 lists proprietary SNMP traps generated by Steel-Belted Radius. The columns in Table 137 consist of the following:

- OID Suffix – Identifies the OID suffix for the trap. To identify the OID number for an alarm, append the OID suffix to the Funk OID prefix (1.3.6.1.4.1.1411). For example, the ASN.1 number for the funkSbrTrapServiceStarted trapis 1.3.6.1.4.1.1411.100.
- Trap – Identifies the name of the proprietary trap.
- Description – Describes when the trap is generated.
- Type – Indicates whether the trap is informational, warning, or error
- Added to SBR – Identifies the version of Steel-Belted Radius in which the trap first appeared.

Note: Some Steel-Belted Radius traps are dilutable, which means that one trap message is generated after a specified number of events of that type occur.

Table 139: fnkradtr.mib Trap Definitions

OID Suffix	Trap Name	Description	Type	Added to SBR
100	funkSbrTrapServiceStarted	Sent when the RADIUS server is started. **Cause**: Trap indicates that the server itself has started. This does not mean that all of the various configured features have loaded successfully. If there is an issue with another component, traps specific to it will indicate so. This trap will show that a valid license is installed. It is now possible to interact with the server through the SBR Administrator or the LDAP Configuration Interface. **Severity**: If unexpected, this could be the result of a core in the radius process.	Info	1.15
101	unkSbrTrapServiceStopped	Sent when the RADIUS server is stopped. **Cause**: The server completed its shutdown operation and is no longer running. Server will not respond to any operation from the SBR Administrator, from the LCI, or from any inbound RADIUS data. **Severity**: If unexpected, this could be the result of a core in the radius process.	Info	
102	funkSbrTrapThreadsNormal	Sent when the number of available threads in the accounting or authentication server has risen above a specified threshold. **Cause**: The number of available threads on the system has risen above the threshold configured in the events.ini file.	Info	

OID Suffix	Trap Name	Description	Type	Added to SBR
		Severity: Could result in loss of packets or inoperablitity.		
103	funkSbrTrapFSNormal	Sent when the number of bytes available in the file system from which the server is running has risen above a specified threshold. **Cause**: The number of bytes available in the free disk space has increased above the threshold configured in the events.ini file. **Severity:** This can cause the system to become inoperable.	Info	
104	funkSbrTrapConcurrencyReconnect	Sent when RADIUS reconnects to the Service Level Manager server after it has sent a ConcurrencyFailure, ConcurrencyTimeout, or ConcurrencyLocalProxyFailuretrap. **Cause**: If a failure to communicate with the Concurrency Server has occurred, this trap is sent when communications have been re-established and the SLM server is responding again. **Severity**: Users may have either been rejected, or they may have been able to exceed their configured concurrent login policy, during the time interval when communications with the CS were down. This will depend on the settings in the forward.aut configuration file which resides on any of the SBR servers acting as clients to the CS. Check the RejectIfUnreachable setting if you are uncertain as to the expected behavior of SBR in the event that the CS is unreachable.	Info	1.15
105	funkSbrTrapSQLReconnect	Sent when Radius reconnects to the SQL database after it has sent a SQLConnectFail trap. **Cause**: If a failure to communicate with the SQL server database has occurred, this trap is sent when communications have been re-established and the SQL server is responding again. **Severity**: Users my have either been rejected,	Info	1.15

OID Suffix	Trap Name	Description	Type	Added to SBR
		or they may have been allowed onto the network without proper verification of credentials, during the time interval when the SQL server was unreachable. This will depend on the settings in the radsql.aut configuration file. Check the [Failure] section settings if you are uncertain as to the expected behavior of Steel-Belted Radius in the event that the SQL server is unreachable.		
106		Sent when Radius reconnects to the LDAP server after it has sent a LDAPConnectFail trap. **Cause**: If a failure to communicate with the LDAP database has occurred, this trap is sent when communications have been re-established and the LDAP server is responding again. **Severity**: Users my have either been rejected, or they may have been allowed onto the network without proper verification of credentials, during the time interval when the LDAP server was unreachable. This will depend on the settings in the ldapauth.aut configuration file. Check the [Failure] section settings if you are uncertain as to the expected behavior of SBR in the event that the LDAP database is unreachable.	Info	1.15
107	funkSbrTrapUserAccountLocked	Sent when a user's account becomes locked out due to an excessive number of rejected authentication attempts within a defined period of time. **Cause**: A user's account is locked, disallowing access to the network, after an excessive number of rejected authentication attempts. This functionality is configured in the lockout. ini file. **Severity**: The user will not be able to access the network until the account is unlocked.	Info	1.15
108	funkSbrTrapUserAccountReleased	Sent when a user's account, previously locked due to an excessive number of rejected	Info	1.15

OID Suffix	Trap Name	Description	Type	Added to SBR
		authentication attempts, becomes unlocked.		
109	funkSbrTrapProxySpoolReconnect	Sent when the proxy accounting spooler reconnects to the target realm after it has sent a ProxySpoolTimeout trap. **Cause**: Issues affecting transmission of spooled accounting proxy data to the configured downstream target(s) have been resolved (possibly a restoration of the network link, or the downstream proxy accounting server has become available again). **Severity**: If unexpected, then the accounting target system (possibly a billing server) was not receiving data from the AAA server for some time interval. During that time, data was written to the local disk for temporary storage until the accounting target became available again. There should be no data lost.	Info	1.15
110	funkSbrTrapIPAddrPoolNormal	Sent when the number of available IP addresses in any pool rises above a specified threshold. IP pool thresholds can be configured in events.ini file. **Severity**: Users could have been rejected if threshold warning trap 5027 was ignored.	Info	1.15
111	funkSbrTrapSQLConnect	Sent only once, when Radius initially connects to the SQL database.	Info	4.04
112	funkSbrTrapLDAPConnect	Sent only once, when Radius initially connects to the LDAP server.	Info	4.04
113	funkSbrTrapWatchdogStarted	Sent when the radiusd watchdog is started.	Info	4.04
114	funkSbrTrapWatchdogStopped	Sent when the radiusd watchdog is stopped.	Info	4.04
115	funkSbrTrapWatchdogRadiusStarted	Sent whenever the radiusd watchdog attempts to (re)start the RADIUS server.	Info	4.04
116	funkSbrTrapUserAccountRedirected	Sent when a user account has been redirected due to an excessive number of rejected authentication attempts.	Info	5.3
117	funkSbrTrapSS7CommunicationOK	Sent when SS7 communications are successful after a funkSbrTrapSS7CommunicationError trap has been sent.	Info	5.3

OID Suffix	Trap Name	Description	Type	Added to SBR
118	funkSbrTrapSS7CDRGenerationOK	Sent when CDR generation is successful after a funkSbrTrapSS7CDRGenerationError trap has been sent.	Info	5.3
119	funkSbrTrapSS7AuthDatabaseOK	Sent when access to the Authorization databases are successful after a funkSbrTrapSS7AuthDatabaseError trap has been sent.	Info	5.3
120	funkSbrTrapSS7ProvDatabaseOK	Sent when access to the SMS Provisioning database is successful after a funkSbrTrapSS7ProvDatabaseError trap has been sent	Info	5.3
5000	funkSbrTrapCmdArgBadPrivDir	Sent when an invalid private directory is specified on the command line used to launch the RADIUS server. The command line option is ignored.	Warning	1.15
5001	funkSbrTrapLowThreads	Sent when the count of threads available for the accounting or authentication server drops below a configurable threshold. An informational trap is sent when the count of available threads (at some future point) rises to an acceptable level.	Warning	1.15
5002	funkSbrTrapConcurrencyFailure	Sent when communications with the RADIUS concurrency server fails. Trap can be diluted	Warning	1.15
5003	funkSbrTrapConcurrencyTimeout	Sent when communications with the RADIUS concurrency server times out. Trap can be diluted.	Warning	1.15
5004	funkSbrTrapConcurrencyLocalProxy Failure	Sent when a local error prevents the RADIUS server from sending a proxy request to the RADIUS concurrency server. Trap can be diluted.	Warning	1.15
5005	funkSbrTrapStaticAcctProxyTimeout	Sent when the RADIUS server times out in an attempt to forward an accounting request to the location specified by the static proxy option. Trap can be diluted.	Warning	1.15
5006	funkSbrTrapStaticAcctProxyL	Sent when the RADIUS server encounters a	Warning	1.15

OID Suffix	Trap Name	Description	Type	Added to SBR
	ocal Failure	local failure in an attempt to forward an accounting request to the location specified by the static proxy option. Trap can be diluted.		
5007	funkSbrTrapLowFSSpace	Sent when the amount of space available in the file system in which the server's private directory resides falls below a configurable threshold. An informational trap is sent when the amount of available space (at some future point) rises to an acceptable level.	Warning	1.15
5008	funkSbrTrapSQLConnectFail	Sent when the connection to a SQL database has failed. Trap can be diluted.	Warning	1.15
5009	funkSbrTrapSQLDisconnect	Sent when a disconnect to a SQL database occurs. Trap can be diluted.	Warning	1.15
5010	funkSbrTrapSQLTimeou	Sent when a timeout occurs during an attempt to perform transactions to a SQL database. Trap can be diluted.	Warning	1.15
5011	funkSbrTrapAcctDbTimeout	Sent when the access to the accounting database has timed out. Trap can be diluted. No longer supported.	Warning	1.15
5012	funkSbrTrapAcctDbFailure	Sent when the access to the accounting database has failed. Trap can be diluted. No longer supported.	Warning	1.15
5013	funkSbrTrapVerifyServerTimeout	Sent when an attempt to communicate with the Verification Server has timed out. Trap can be diluted. No longer supported.	Warning	1.15
5014	funkSbrTrapVerifyServerFail	Sent when an attempt to communicate with the Verification Server has failed.	Warning	1.15

OID Suffix	Trap Name	Description	Type	Added to SBR
		Trap can be diluted. No longer supported.		
5015	funkSbrTrapLDAPConnectFailure	Sent when a connect failure to an LDAP server occurs.	Warning	1.15
5016	funkSbrTrapLDAPConnectFailures	Sent when an attempt to communicate with the LDAP Server has failed. Trap can be diluted.	Warning	1.15
5017	funkSbrTrapLDAPDisconnects	Sent when the LDAP Server has disconnected. Trap can be diluted.	Warning	1.15
5018	funkSbrTrapLDAPRequestTimeouts	Sent when a request sent to the LDAP Server has timed out. Trap can be diluted.	Warning	1.15
5019	funkSbrTrapLDAPDisconnect	Sent when a disconnect to a LDAP server occurs.	Warning	1.15
5020	funkSbrTrapLDAPRequestTimeout	Sent when a request sent to the LDAP Server has timed out.	Warning	1.15
5021	funkSbrTrapProxySpoolTimeout	Sent when a request forwarded by the proxy accounting spooler has timed out.	Warning	1.15
5022	funkSbrTrapProxySpoolTimeouts	Sent when a request forwarded by the proxy accounting spooler has timed out. Trap can be diluted.	Warning	1.15
5023	funkSbrTrapSoftLimitViolation	Sent when accepting a concurrency request exceeds a realm's soft limit. Trap can be diluted.	Warning	1.15
5024	funkSbrTrapHardLimitViolation	Sent when a concurrency request is rejected because a realm's hard limit has been reached. Trap can be diluted.	Warning	1.15
5025	funkSbrTrapConcurrencyS	Sent when a PAS realm has been misconfigured. All authentication requests to	Warning	1.15

OID Suffix	Trap Name	Description	Type	Added to SBR
	erver Misconfiguration	the named realm will be rejected.		
5026	funkSbrTrapACCTWriteFailure	Sent when the server is unable to commit accounting data to a persistent store such as the file system, database, etc. Trap can be diluted.	Warning	1.15
5027	funkSbrTrapIPAddrPoolLow	Sent when the number of available IP addresses in any pool falls below a configurable threshold. An informational trap is sent when the number of available IP addresses (at some future point) rises to an acceptable level.	Warning	1.15
5028	funkSbrTrapWatchdogRadiusTerm	Sent whenever the radiusd watchdog attempts to send a TERM signal to terminate the RADIUS server.	Warning	4.04
5029	funkSbrTrapWatchdogRadiusKill	Sent whenever the radiusd watchdog attempts to send a KILL signal to terminate the RADIUS server.	Warning	4.04
5030	funkSbrTrapFloodQueueOverflow	Sent whenever a flood queue drops a packet. Trap can be diluted.	Warning	5.3
5031	funkSbrTrapFailEAIni	Sent when an attempt to process the ea.ini file at server startup fails.	Warning	5.3
5032	funkSbrTrapPasError	Sent when rejecting a concurrency request for internal processing reasons. Trap can be diluted.	Warning	6.0
5033	funkSbrTrapMappingFailure	Sent when rejecting a concurrency request that fails to resolve a realm or region name Trap can be diluted.	Error	6.0
5034	funkSbrTrapSubscriberLimitViolation	Sent when rejecting a concurrency request that exceeds a subscriber limit. Trap can be diluted.	Error	6.0
10000	funkSbrTrapStartServiceError	Sent for a Windows version of RADIUS when the service control dispatcher returns an error. Not used for SNMP on Linux.	Error	1.15

OID Suffix	Trap Name	Description	Type	Added to SBR
10001	funkSbrTrapSetStatusError	Sent for a Windows version of RADIUS when the attempt to inform the service control dispatcher of the status of the RADIUS server encounters an error. Not used for SNMP on Linux.	Error	1.15
10002	funkSbrTrapBadPrivDir	Sent for a Windows version of RADIUS when the attempt to inform the service control dispatcher of the status of the RADIUS server encounters an error. Not used for SNMP on Linux.	Error	1.15
10003	funkSbrTrapFailedThreadCreate	Sent when an attempt to create a thread at server startup encounters a failure. The server will fail to start.	Error	1.15
10004	funkSbrTrapFailedMutexCreate	Sent when an attempt to create a mutual exclusion lock (mutex) at server startup encounters a failure. A mutex prevents multiple threads from executing critical sections of code simultaneously. The server will fail to start.	Error	1.15
10005	funkSbrTrapFailedSignalInit	Sent when an attempt to initialize signal handling at server startup encounters a failure. The server will fail to start.	Error	1.15
10006	funkSbrTrapFailedEventInit	Sent for a Windows version of RADIUS when an attempt to initialize event processing at server startup encounters a failure. The server will fail to start. Not used for SNMP on Linux.	Error	1.15
10007	funkSbrTrapFailedLogFile	Sent when an attempt to open or create a log file at server startup encounters a failure. The server will fail to start.	Error	1.15
10008	funkSbrTrapFailedLDAPAdminInit	Sent when an attempt to initialize the LDAP administration interface at server startup encounters a failure. The server will fail to start.	Error	1.15
10009	funkSbrTrapFailedRPCInit	Sent when an attempt to initialize the RPC administration interface at server startup encounters a failure. The server will fail to start.	Error	1.15

OID Suffix	Trap Name	Description	Type	Added to SBR
10010	funkSbrTrapFailedIPInit	Sent when an attempt to initialize basic TCP/IP services at server startup encounters a failure. The server will fail to start.	Error	1.15
10011	funkSbrTrapFailedCurrentSessionsIn it	Sent when an attempt to initialize current sessions table processing at server startup encounters a failure. The server will fail to start.	Error	1.15
10012	funkSbrTrapFailedChallCacheInit	Sent when an attempt to initialize the RADIUS challenge continuation cache at server startup encounters a failure. The server will fail to start.	Error	1.15
10013	funkSbrTrapFailedActiveRASInit	Sent when an attempt to initialize the network access device activity monitor at server startup encounters a failure. The server will fail to start.	Error	1.15
10014	funkSbrTrapFailedDictionaryInit	Sent when an attempt to initialize the dictionary processing at server startup encounters a failure. The server will fail to start.	Error	1.15
10015	funkSbrTrapFailedVendorInit	Sent when an attempt to process the vendor.ini file at server startup encounters a failure. The server will fail to start.	Error	1.15
10016	funkSbrTrapFailedDBInit	Sent when an attempt to initialize the internal database at server startup encounters a failure. The server will fail to start.	Error	1.15
10017	funkSbrTrapFailedUnixUserInit	Sent when an attempt to initialize the Unix user browsing component at server startup encounters a failure. The server will fail to start.	Error	1.15
10018	funkSbrTrapFailedAdminRightsIni t	Sent when an attempt to initialize the administration user rights component at server startup encounters a failure. The server will fail to start.	Error	1.15
10019	funkSbrTrapFailedDbOpen	Sent when an attempt to open the internal database at server startup encounters a failure. The server will fail to start.	Error	1.15
10020	funkSbrTrapFailedDNISLookupIni t	Sent when an attempt to initialize the tunnel DNIS lookup component at server startup encounters a failure. The server will fail to start.	Error	1.15
10021	funkSbrTrapFailedConfigCacheIni	Sent when an attempt to initialize the	Error	1.15

OID Suffix	Trap Name	Description	Type	Added to SBR
	t	configuration caching component at server startup encounters a failure. The server will fail to start.		
10022	funkSbrTrapFailedDbCacheInit	Sent when an attempt to initialize the database caching component at server startup encounters a failure. The server will fail to start.	Error	1.15
10023	funkSbrTrapFailedLicenseInit	Sent when an attempt to initialize the licensing component at server startup encounters a failure. The server will fail to start.	Error	1.15
10024	funkSbrTrapFailedNDSTrusteeInit	Sent when an attempt to initialize NDS trustee processing on NetWare at server startup encounters a failure. The server will fail to start.	Error	1.15
10025	funkSbrTrapFailedHostLookupInit	Sent when an attempt to initialize host lookup processing on NetWare at server startup encounters a failure. The server will fail to start.	Error	1.15
10026	funkSbrTrapFailedAddrPoolInit	Sent when an attempt to initialize IP/IPX address pool resource management at server startup encounters a failure. The server will fail to start.	Error	1.15
10027	funkSbrTrapFailedLoginLimitInit	Sent when an attempt to initialize user login count tracking at server startup encounters a failure. The server will fail to start.	Error	1.15
10028	funkSbrTrapFailedPersistStoreCreate	Sent when an attempt to create the persistent store for current session list processing at server startup encounters a failure. The server will fail to start.	Error	1.15
10029	funkSbrTrapFailedPersistStoreInit	Sent when an attempt to initialize the persistent store for current session list processing at server startup encounters a failure. The server will fail to start.	Error	1.15
10030	funkSbrTrapFailedPerfMonInit	Sent for a Windows version of RADIUS when an attempt to initialize the Windows performance monitor interface at server startup encounters a failure. The server will fail to start. Not used for SNMP on Linux.	Error	1.15
10031	funkSbrTrapFailedLockInit	Sent when an attempt to initialize admin locking component at server startup encounters a failure.	Error	1.15

OID Suffix	Trap Name	Description	Type	Added to SBR
		The server will fail to start.		
10032	funkSbrTrapFailedPlugInInit	Sent when an attempt to initialize the plug-in support component at server startup encounters a failure. The server will fail to start.	Error	1.15
10033	funkSbrTrapFailedPacketCacheInit	Sent when an attempt to initialize duplicate packet request cache at server startup encounters a failure. The server will fail to start.	Error	1.15
10034	funkSbrTrapFailedNameMangleInit	Sent when an attempt to initialize name mangling support at server startup encounters a failure. The server will fail to start.	Error	1.15
10035	funkSbrTrapFailedNameStripInit	Sent when an attempt to initialize name stripping support at server startup encounters a failure. The server will fail to start.	Error	1.15
10036	funkSbrTrapFailedFSSpaceChecking	Sent when an attempt to determine the amount of free file system space fails. File system space checking will be disabled until the server is restarted.	Error	1.15
10037	funkSbrTrapFailedNameValidateInit	Sent when an attempt to initialize name validation support at server startup encounters a failure. The server will fail to start.	Error	1.15
10038	funkSbrTrapFailedResourceCheckInit	Sent when an attempt to initialize system resource checking at server startup encounters a failure. The server will fail to start.	Error	1.15
10039	funkSbrTrapFailedSystemStatsInit	Sent when an attempt to initialize statistic collection at server startup encounters a failure. The server will fail to start.	Error	1.15
10040	funkSbrTrapSQLConnectFailure	Sent when a connection attempt from the SQL authentication or accounting plug-in to the specified system has failed.	Error	1.15
10041	funkSbrTrapSQLDiscon	Sent when a disconnect from a SQL database has occurred.	Error	1.15
10042		individual SQL timeout (not sent as a trap)	Error	1.15

OID Suffix	Trap Name	Description	Type	Added to SBR
10043	funkSbrTrapFailedReserveMemoryAll oc	Sent when an attempt to allocate reserved memory based on a setting in the radius.ini file fails. The server will start without reserved memory, but will be unable to warn of low memory conditions.	Error	1.15
10044	funkSbrTrapReserveMemoryFreed	Sent when an attempt to allocate memory during runtime fails and the block of memory reserved at system startup is freed in an attempt to alleviate the low memory condition.	Error	1.15
10045	funkSbrTrapMemoryAllocFail	Sent when an attempt to allocate memory has failed. Trap can be diluted.	Error	1.15
10046		individual accounting database connect failure (not sent as a trap)	Error	1.15
10047		individual verification server connect failure (not sent as a trap)	Error	1.15
10048	funkSbrTrapFailedMibInfoCollectInit	Sent when an attempt to initialize MIB information collection at server startup encounters a failure. The server will fail to start.	Error	1.15
10049	funkSbrTrapFailedMibInfoAccessInit	Sent when an attempt to initialize MIB access at server startup encounters a failure. The server will fail to start.	Error	1.15
10050	funkSbrTrapFailedCommonIPInit	Sent when an attempt to initialize common IP services at server startup encounters a failure. The server will fail to start.	Error	1.15
10051	funkSbrTrapWatchdogAborted	Sent whenever the radiusd watchdog aborts. This is typically due to a prolonged inability to control or communicate with the RADIUS server, or some fatal error that has occurred within the watchdog itself.	Error	4.04
10052	funkSbrTrapWatchdogFailedInit	Sent whenever the radiusd watchdog is unable to initialize itself. This is typically due to insufficient or invalid command line parameters given to the watchdog itself.	Error	4.04
10053	funkSbrTrapAdminAuthFailedInit	Sent whenever the server is unable to initialize administrative authentication and authorization.	Error	5.3

OID Suffix	Trap Name	Description	Type	Added to SBR
		The server will fail to start.		
10054	funkSbrTrapServiceFailedInit	Sent when the server has failed to start. This trap is sent in addition to a specific failure trap.	Error	5.3
10055	funkSbrTrapSS7MapGatewayFailed In it	Sent when the SS7 MAP Gateway has failed to initialize. The server will start but SS7 functions will not be available.	Error	5.3
10056	funkSbrTrapSS7CommunicationE rror	Sent when SS7 communication has failed.	Error	5.3
10057	funkSbrTrapSS7CDRGenerationE rror	Sent when CDRs cannot be written.	Error	5.3
10058	funkSbrTrapSS7AuthDatabaseErr or	Sent when access to an Authorization database has failed.	Error	5.3
10059	funkSbrTrapSS7ProvDatabaseErr or	Sent when access to the Provisioning database has failed.	Error	5.3

Server Rate Statistics

This section presents an overview of the SNMP accessible rate statistics maintained by Steel-Belted Radius. The fnkrate.mib MIB maintains rate statistics for the Steel-Belted Radius server. Steel-Belted Radius maintains three types of values for these types of statistics:

- The current-rate statistics specify the rate measured over the most recent rate interval.
- The average-rate statistics specify the rate measured since startup, or the most recent statistics reset command.
- The peak-rate statistics specify the highest rate observed since startup, or the most recent statistics reset command.

Table 140: Server Rate Statistics

OID Suffix	Statistic	Function
1	funkSbrRatesSecondsPerInterval	Specifies the duration (in seconds) of the interval over which the rate statistics are gathered.
2	funkSbrRatesAuthRequestCurrentRate	AuthRequest Current Rate
3	funkSbrRatesAuthRequestAverageRate	AuthRequest Average Rate
4	funkSbrRatesAuthRequestPeakRate	AuthRequest Peak Rate
5	funkSbrRatesAuthAcceptCurrentRate	AuthAccept Current Rate
6	funkSbrRatesAuthAcceptAverageRate	AuthAccept Average Rate
7	funkSbrRatesAuthAcceptPeakRate	AuthAccept Peak Rate
8	funkSbrRatesAuthRejectCurrentRate	AuthReject Current Rate
9	funkSbrRatesAuthRejectAverageRate	AuthReject Average Rate
10	funkSbrRatesAuthRejectPeakRate	AuthReject Peak Rate
11	funkSbrRatesAcctStartCurrentRate	AcctStart Current Rate
12	funkSbrRatesAcctStartAverageRate	AcctStart Average Rate
13	funkSbrRatesAcctStartPeakRate	AcctStart Peak Rate
14	funkSbrRatesAcctStopCurrentRate	AcctStop Current Rate
15	funkSbrRatesAcctStopAverageRate	AcctStop Average Rate
16	funkSbrRatesAcctStopPeakRate	AcctStop Peak Rate
17	funkSbrRatesProxyAuthRequestCurrentRate	ProxyAuthRequest Current Rate
18	funkSbrRatesProxyAuthRequestAverageRate	ProxyAuthRequest Average Rate
19	funkSbrRatesProxyAuthRequestPeakRate	ProxyAuthRequest Peak Rate

OID Suffix	Statistic	Function
20	funkSbrRatesProxyAcctRequestCurrentRate	ProxyAcctRequest Current Rate
21	funkSbrRatesProxyAcctRequestAverageRate	ProxyAcctRequest Average Rate
22	funkSbrRatesProxyAcctRequestPeakRate	ProxyAcctRequest Peak Rate
23	funkSbrRatesProxyFailTimeoutCurrentRate	ProxyFailTimeout Current Rate
24	funkSbrRatesProxyFailTimeoutAverageRate	ProxyFailTimeout Average Rate
25	funkSbrRatesProxyFailTimeoutPeakRate	ProxyFailTimeout Peak Rate
26	funkSbrRatesProxyFailBadrespCurrentRate	ProxyFailBadresp Current Rate
27	funkSbrRatesProxyFailBadrespAverageRate	ProxyFailBadresp Average Rate
28	funkSbrRatesProxyFailBadrespPeakRate	ProxyFailBadresp Peak Rate
29	funkSbrRatesProxyFailBadsecretCurrentRate	ProxyFailBadsecret Current Rate
30	funkSbrRatesProxyFailBadsecretAverageRate	ProxyFailBadsecret Average Rate
31	funkSbrRatesProxyFailBadsecretPeakRate	ProxyFailBadsecret Peak Rate
32	funkSbrRatesProxyFailMissingresrCurrentRate	ProxyFailMissingresr Current Rate
33	funkSbrRatesProxyFailMissingresrAverageRate	ProxyFailMissingresr Average Rate
34	funkSbrRatesProxyFailMissingresrPeakRate	ProxyFailMissingresr Peak Rate
35	funkSbrRatesProxyRetriesCurrentRate	ProxyRetries Current Rate
36	funkSbrRatesProxyRetriesAverageRate	ProxyRetries Average Rate
37	funkSbrRatesProxyRetriesPeakRate	ProxyRetries Peak Rate
38	funkSbrRatesProxyAuthRejProxyCurrentRate	ProxyAuthRejProxy Current Rate

OID Suffix	Statistic	Function
39	funkSbrRatesProxyAuthRejProxyAverageRate	ProxyAuthRejProxy Average Rate
40	funkSbrRatesProxyAuthRejProxyPeakRate	ProxyAuthRejProxy Peak Rate
41	funkSbrRatesProxyAcctFailProxCurrentRate	ProxyAcctFailProx Current Rate
42	funkSbrRatesProxyAcctFailProxAverageRate	ProxyAcctFailProx Average Rate
43	funkSbrRatesProxyAcctFailProxPeakRate	ProxyAcctFailProx Peak Rate
44	funkSbrRatesProxyAuthRejProxyErrorCurrentRate	ProxyAuthRejProxyError Current Rate
45	funkSbrRatesProxyAuthRejProxyErrorAverageRate	ProxyAuthRejProxyError Average Rate
46	funkSbrRatesProxyAuthRejProxyErrorPeakRate	ProxyAuthRejProxyError Peak Rate

Appendix D

Windows Events

Steel-Belted Radius generates a variety of Windows events. Regardless of severity, each event is attributed to one of the following three Windows services: the "core" Steel-Belted Radius service, the authentication service, or the accounting service.

Table 140 lists service identifiers.

Table 141: Windows Events

ID	Symbolic Name	Text
1	RADCAT_CORE	Core
2	RADCAT_AUTH	Authentication
3	RADCAT_ACCT	Accounting

Informational Events

Table 141 lists events generated for informational purposes only. Informational events do not require operator intervention.

Note: Some informational events indicate that a previous warning event has been cleared.

Table 142: Informational Events

ID	Informational Event	Comment
100	The Steel-Belted Radius service has started.	
101	The Steel-Belted Radius service has stopped.	
102	Count of available threads has risen to acceptable threshold of nnnn.	You can set the threshold value for nnnn in the [Thresholds] section of the events.ini file.
103	You can set the threshold value for nnnn in the [Thresholds] section of the events.ini file.	You can set the threshold value for nnnnnnnn in the [Thresholds] section of the events.ini file.
104	Steel-Belted Radius has reconnected to the Service Level Manager server after a	

ID	Informational Event	Comment
	ConcurrencyFailure.	
105	Steel-Belted Radius has reconnected to the SQL database after a SQLConnectFail.	
106	Steel-Belted Radius has reconnected to the LDAP database after an LDAPConnectFail.	
107	A user's account has been locked due to excessive authentication attempts within a specified period.	
108	A user account, previously locked due to an excessive amount of rejected authentication attempts, has been unlocked.	
109	The target server for proxy spooling has reconnected.	

Warning Events

Table 142 lists warning events that may require operator intervention.

Some warning events can be diluted, meaning that a warning message is generate every nnnn times an event occurs. Warning message dilution is configured in the [EventDilutions] section of the events.ini file.

Table 143: Warning Events

ID	Warning Event	Dilutable?
5001	Count of available threads has dropped to the minimum threshold of nnnn. Indicates that a low thread count available condition has been detected. This event can be issued in the authentication or accounting category to indicate a shortage of authentication or accounting threads.	No
5002	Concurrency server returned failure indication. This event represents nnnn failures. Indicates that a reject was returned from the Service Level Manager server (concurrency server) in response to a proxied authentication request. The reject was for a reason other than exceeded port limit.	Yes
5003	Timed out in proxy attempt to concurrency server. This event represents nnnn requests timing out. Indicates that a time-out was encountered when proxy-forwarding an authentication request to the Service Level Manager server (concurrency server).	Yes
	Local failure encountered in attempt to proxy to concurrency server. This event represents	

ID	Warning Event	Dilutable?
5004	nnnn requests timing out. Indicates that a local processing failure was encountered when trying to proxy-forward an authentication request to the Service Level Manager server (concurrency server).	Yes
5005	Timed out in static accounting proxy attempts. This event represents nnnn failures. Indicates that a time-out was encountered when proxy-forwarding an accounting request to the Service Level Manager server (concurrency	Yes
5006	Local failure encountered in attempt to proxy for static accounting. This event represents nnnn requests timing out. Indicates that a local processing failure was encountered when trying to proxy-forward an accounting request to the Service Level Manager server (concurrency server). You can set the threshold value for nnnn using the [EventDilutions] section of events.ini.	Yes
5007	Amount of free file system space has dropped below minimum threshold. Free byte count is nnnnnnnn. You can set the threshold value for nnnnnnnn in the [Thresholds] section of events. ini.	No
5008	nnnn attempts to connect to SQL server failed.	Yes
5009	nnnn disconnects from SQL server due to error.	Yes
5010	nnnn timeouts on SQL requests.	Yes
5011	Access to accounting server database has timed out. This event represents nnnn timeouts.	Yes
5012	Access to accounting server database has failed. This event represents nnnn failures.	Yes
5013	Verification Server has timed out. This event represents nnnn Verification Server timeouts.	Yes
5014	Verification Server requests have failed. This event represents nnnn Verification Server failures.	Yes
5015	The connection to an LDAP server has failed.	No
5016	Communication with an LDAP server has failed. This event represents nnnn connection failures.	Yes
5017	The LDAP server has disconnected. This event represents nnnn connection failures.	Yes
5018	A request to the LDAP server has timed out. This event represents nnnn request timeouts.	Yes
5019	The LDAP server has disconnected	No
5020	A request to the LDAP server has timed out.	No

ID	Warning Event	Dilutable?
5021	The target server of proxy spooling fails to respond (non-dilutable).	No
5022	The target server of proxy spooling fails to respond (dilutable).	Yes
5023	An SLM soft limit has been reached for realm xxxx. This represents a total of nnnn hard limit violations for all servers.	Yes
5024	An SLM hard limit has been reached for realm xxxx. This represents a total of nnnn hard limit violations for all servers.	Yes
5026	Failure to commit accounting data to a persistent store such as the file system or database.	Yes
5032	An SLM error has occured.	Yes
5033	An SLM subscriber has been reached for realm xxxx. This represents a total of nnnn subscriber limits for all servers.	Yes

Error Events

Error events usually require some form of operator intervention.

Most Steel-Belted Radius error events are generated at startup, as the service initializes its components. If a component fails at startup, the start operation is aborted and the system generates an error event. The text of the error event message identifies what Steel-Belted Radius was doing when it failed. In some cases, the operator can take direct action in response to an error event. For example, if the system is unable to open a log file, the system disk might be full, leaving no room to create additional files.

The event text identifies the problem area in the software. You should escalate the problem to your next level of support. When you do, be sure to indicate the ID, name, and text of the event.

Table 144: Error Events

ID	Error Event
10000	StartServiceCtrlDispatcher failed with error nnnn.
10001	SetServiceStatus failed with error nnnn.
10002	Invalid private directory 'directory' specified.
10003	Unable to create thread.
10004	Unable to create mutex.

ID	Error Event
10005	Unable to initialize signal handling.
10006	Unable to configure event processing.
10007	Unable to create or open log file.
10008	Unable to initialize LDAP administration interface.
10009	Unable to initialize RPC administration interface.
10010	Unable to initialize base IP interface.
10011	Unable to initialize current user list processing.
10012	Unable to initialize challenge continuation cache.
10013	Unable to initialize network access device activity monitor.
10014	Unable to initialize dictionary processing.
10015	Unable to process vendor.ini file.
10018	Unable to initialize admin user rights component.
10020	Unable to initialize tunnel DNIS lookup component.
10021	Unable to initialize configuration caching component.
10022	Unable to initialize database caching component.
10023	Unable to initialize license processing.
10024	Unable to initialize NDS trustee processing.
10025	Unable to initialize NetWare host lookup processing.
10026	Unable to initialize IP/IPX pool resource management.

ID	Error Event
10027	Unable to initialize user login count tracking.
10028	Unable to create persistent store for Sessions list.
10029	Unable to initialize persistent store for Sessions list.
10030	Unable to initialize performance monitor interface component.
10031	Unable to initialize admin locking component.
10032	Unable to initialize plug-in support component.
10033	Unable to initialize duplicate request cache.
10034	Unable to initialize name mangling support.
10035	Unable to initialize name stripping support.
10036	Error nnnn returned from call to GetDiskFreeSpaceEx. File system space checking disabled.
10037	Unable to initialize name validation support. Service start aborted.
10038	Unable to initialize system resource checking. Service start aborted.
10039	Unable to initialize statistic collection. Service start aborted.
10040	Attempt to connect to SQL server xxxxxxxx failed.
10041	Disconnect from SQL server xxxxxxxx due to error.
10042	Timeout on SQL request.
10043	Unable to allocate reserved memory specified by ReserveMemoryKB. You can set the ReserveMemoryKB value in the [Thresholds] section of the events.ini configuration file.
10044	Memory allocation failure encountered. Reserved memory released as last resort.
10045	nnnn memory allocation failures have occurred. You can set the threshold value for nnnn using the [EventDilutions] section of events.ini.

ID	Error Event
10046	The connection to the Accounting Server has failed.
10047	The connection to the Verification Server has failed.
10050	The initialization of common IP services at server startup has failed.

Symbols