RSA RADIUS Server 6.1 Reference Guide

Powered by Steel-Belted Radius®

Contact Information

See our web site for regional Customer Support telephone and fax numbers.

RSA Security Inc.	RSA Security Ireland Limited
www.rsasecurity.com	www.rsasecurity.ie

Copyright

Trademarks

License agreement

Note on encryption technologies

This product may contain encryption technology. Many countries prohibit or restrict the use, import, or export of encryption technologies, and current use, import, and export regulations should be followed when exporting this product.

Distribution

Limit distribution of this document to trusted personnel.

RSA notice

The RC5™ Block Encryption Algorithm With Data-Dependent Rotations is protected by U.S. Patent #5,724,428 and #5,835,600.

First Printing: September 2005
Part Number: M05915REF

Contents

Preface

Before You Begin ix
Audience ix
What's In This Manual ix
Typographical Conventions x
Related Documentation xii

Chapter 1 Introduction

Guidelines for Editing Configuration Files 1

Chapter 2 Initialization (.ini) Files

account.ini File 6
- [Alias/name] Sections 6
- [Attributes] Section 7
- [Configuration] Section 8
- [Settings] Section 9
- [TypeNames] Section 12

certinfo.ini File 14
- Server Certificates 14
- Client Certificates 14

classmap.ini File 16
- [AttributeName] Section 16

eap.ini File 17

radius.ini File 19
- [Addresses] Section 19
- [Certificate] Section 19
- [Configuration] Section 20

[CurrentSessions] Section 25
[EmbedInClass] Section 26
[HiddenEAPIdentity] Section 26
[LDAP] Section 27
[LDAPAddresses] Section 28
[Ports] Section 29
[SecurID] Section 30
securid.ini File 31
[Configuration] Section 31
[Server_Settings] Section 32
SecurID Prompts Section 32
spi.ini File 37
[Keys] Section 37
[Hosts] Section 38
vendor.ini File 39
[Vendor-Product Identification] Section 39

Chapter 3 Dictionary Files

Overview 43
Dictionary File Location 44
Dictionary File Records 45
Editing Dictionary Files 45
Include Records 46
Master Dictionary File 46
ATTRIBUTE Records 47
Attribute Name and Identifier 48
Syntax Type Identifier 48
Compound Syntax Types 48
Flag Characters 49
VALUE Records 50
Macro Records 51
OPTION Records 52

Chapter 4 EAP Configuration Files

peapauth.aut File 54
[Bootstrap] Section 54
[Server_Settings] Section 55
[Session_Resumption] Section 57

ttlsauth.aut File 59
[Bootstrap] Section 59
[Server_Settings] Section 60
[Session_Resumption] Section 61
Sample ttlsauth.aut File 62

Index

Preface

The *RSA RADIUS Server 6.1 Reference Guide* describes the configuration options for the RSA RADIUS Server software.

Before You Begin

This manual assumes that you have installed the RSA RADIUS Server software and a local or remote copy of the RSA Authentication Manager (with embedded RSA RADIUS Administrator application). For more information, refer to the *RSA RADIUS Server 6.1 Administrator's 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 RSA RADIUS Server.

What's In This Manual

This manual contains the following chapters:

- Chapter 1, "Introduction," presents an overview of RSA RADIUS Server.
- Chapter 2, "Initialization (.ini) Files," describes the usage and settings for the initialization (`*.ini`) files used by RSA RADIUS Server.

- Chapter 3, "Dictionary Files," describes the usage and settings for the RSA RADIUS Server attribute processing and dictionary files.
- Chapter 4, "EAP Configuration Files," describes the EAP configuration files.

Typographical Conventions

This manual uses the following conventions to present special types of text.

Computer Text

Filenames, directory names, IP addresses, URLs, commands, and file listings appear in a plain fixed-width font:

For more information, go to `www.rsasecurity.com`

```
[Configuration]
AllowSystemPins = 0
```

In examples, text that you type literally is shown in a bold font.

```
C:\>cd \Program Files
```

Screen Interaction

Text related to the RSA RADIUS Administrator's user interface appears in **bold sans serif type**.

Click the **OK** button.

Enter your user name in the **Login** field.

Menu commands are presented as the name of the menu, followed by the **>** sign and the name of the command. If a menu item opens a submenu, the complete menu path is given.

Choose **Edit > Cut**.
Choose **Edit > Paste As… > Text**.

Variable Text

Variable text that you must replace with your own information appears in *italics*. For example, you would enter your name and password in place of ***YourName*** and ***YourPassword*** in the following interaction.

```
Enter your name: YourName
Password: YourPassword
```

File names and computer text can be displayed in italics to indicate that you should replace the values shown with values appropriate for your enterprise. For example, you would enter your own information in place of the italicized text in the following example:

```
[EventDilutions]
EventName=DilutionCount
...
```

Key Names

Names of keyboard keys appear in SMALL CAPS. When you need to press two or more keys simultaneously, the key names are joined by a + sign:

Press RETURN.

Press CTRL+ALT+DEL.

Syntax

- *radiusdir* represents the directory into which RSA RADIUS Server has been installed. By default, this is `C:\Program Files\RSA Security\RSA RADIUS` for Windows systems and `/opt/rsa/radius` on Linux and Solaris systems.
- 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 brackets) on the same line.

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

  In configuration files, brackets identify section headers:

  the [Configuration] section of `radius.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):

  ```
  [AttributeName]
  ```

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

Related Documentation

The following documents supplement the information in this manual.

RSA RADIUS Server Documentation

The *RSA RADIUS Server 6.1 Administrator's Guide* describes how to configure and administer the RSA RADIUS Server software.

Vendor Information

You can consult the online Vendor Information file for information about using RSA RADIUS Server with different remote access servers and firewalls. To access this file:

1. Start the RSA RADIUS Administrator application.
2. Choose **Web > NAS Vendor Information**.

   You can access the same information by clicking the **Web Info** button on the Add RADIUS Client or Edit RADIUS Client window.

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`.

- 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 2869, *RADIUS Extensions*. C. Rigney, W. Willats, P. Calhoun. June 2000.
- RFC 2882, *Network Access Servers Requirements: Extended RADIUS Practices*. D. Mitton. July 2000.

Third-Party Products

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

Getting Support and Service

RSA SecurCare Online	https://knowledge.rsasecurity.com
Customer Support Information	www.rsasecurity.com/support

Before You Call for Customer Support

Make sure you have direct access to the computer running the RSA Authentication Manager software.

Have the following information available when you call:

- Your RSA Security Customer/License ID. You can find this number on the license distribution medium or by running the Configuration Management application on Windows servers, or by issuing an `sdinfo` command on Linux or Solaris servers.
- RSA Authentication Manager software version number.
- The make and model of the machine on which the problem occurs.
- The name and version of the operating system under which the problem occurs.

Chapter 1
Introduction

RSA RADIUS Server is a complete implementation of the industry-standard RADIUS (Remote Authentication Dial-In User Service) protocols. It interfaces with a wide variety of network access equipment, and authenticates remote and WLAN users against numerous back-end databases, so that you can consolidate the administration of remote users and wireless local area network (WLAN) users.

RSA RADIUS Server delivers a complete RADIUS solution, designed to meet the access control and policy management requirements of enterprises. It interfaces with a wide variety of network access servers—including virtual private networks (VPNs), dial-in servers, and wireless LAN access points (APs)—and authenticates remote and WLAN users against your existing security infrastructure. This lets you control who can access your network and what resources are available to them, and requires little administration beyond your current management of LAN users. RSA RADIUS Server then logs all access usage, so you can track and document usage statistics.

Guidelines for Editing Configuration Files

When editing configuration files, please observe the following guidelines:

- Configuration files are ASCII text files that can be edited 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 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.

- Blank lines in configuration files are ignored during processing.
- Each setting in a configuration file must be on a separate line. When a setting takes the form *keyword = value*, you must put a space before and after the equal sign. Whitespace at the end of configuration setting line is ignored.
- 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 5/30/2005
  Session_Timeout = 0
  ```

  Incorrect:

  ```
  Session_Timeout = 0 ; Set to 0 on 5/30/2005
  ```

- The default configuration files provided with RSA RADIUS Server typically include section headers and settings that are commented out. In such cases, RSA RADIUS Server 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.

  If you want 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 may not be processed correctly.
- You can edit configuration files while RSA RADIUS Server is running. However, changes to settings do not take effect until you restart RSA RADIUS Server.
- Settings in RSA RADIUS Server configuration files are not copied as part of the replication process. If you change a setting in an RSA RADIUS Server

configuration file, you must copy the file manually to each server (Primary and Replica) in a realm to keep them synchronized.

Chapter 2
Initialization (.ini) Files

This chapter describes the usage and settings for the initialization (`*.ini`) files used by RSA RADIUS Server. Initialization files are loaded at startup time, and reside in the RSA RADIUS Server directory.

NOTE: *The initialization files for RSA RADIUS Server must remain in the installation directory. Do not move the files to other locations on your computer.*

File	Page
account.ini File	Page 6
certinfo.ini File	Page 14
classmap.ini File	Page 16
eap.ini File	Page 17
radius.ini File	Page 19
securid.ini File	Page 31
spi.ini File	Page 37
vendor.ini File	Page 39

account.ini File

The `account.ini` file contains information that controls how RADIUS accounting attributes are logged to a comma-delimited text file by RSA RADIUS Server. 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 RAS 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 RSA RADIUS Server 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]
VendorSpecificAttribute=
VendorSpecificAttribute=
.
.
.
```

Table 1. *account.ini [Alias|name] Syntax*

Parameter	Meaning
name	The preferred attribute name. The name attribute must be one that you are currently logging to a column in the RSA RADIUS Server 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 are 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 RSA 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 RSA RADIUS Server, 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:

```
[Attributes]
User-Name=
NAS-IP-Address=
NAS-Port=
Service-Type=
Framed-Protocol=
Framed-IP-Address=
.
.
.
```

The [Attributes] section lists one *AttributeName* on each line. An equal sign (=) must immediately follow 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 RSA 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 resequenced or deleted. Therefore, these attributes do not appear in the* `account.ini` *file [Attributes] section.*

[Configuration] Section

The [Configuration] section of the account.ini file (Table 2) specifies the location of the log directory for RSA RADIUS Server. You may need to add this section and parameter if it does not exist in your `account.ini` file.

***Table 2.** account.ini [Configuration] Syntax*

Parameter	Meaning
LogDir	If this setting is present, it overrides the default system location (*radiusdir*). You may need to add this section and field if it does not exist in your `account.ini` file.

[Settings] Section

RSA RADIUS Server writes all accounting data to the current accounting log file (`.act`) until that log file is closed. After closing the file, RSA RADIUS Server 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 3 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 3.** 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 following fields in the [Settings] section of the `account.ini` file control how entries are written to the accounting log file, and ensure the compatibility of these entries with a variety of database systems. The following rollover fields can be present in the [Settings] section.

***Table 4.** account.ini [Settings] Syntax*

Parameter	Meaning
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.

Table 4. *account.ini [Settings] Syntax (Continued)*

Parameter	Meaning
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.
MaxSize	The maximum size of an accounting log file, in bytes. Once the accounting log file reaches this limit, the log file is closed and a new file started. A value of 0 means unlimited size. Default value is 0.
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.
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.

Table 4. *account.ini [Settings] Syntax (Continued)*

Parameter	Meaning
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 often the current accounting log file is closed and a new file opened (rollover), up to one rollover per minute. Non-zero values indicate the number of minutes until the next rollover. A value of 0 causes a rollover once every 24 hours, at midnight local time. Default value is 0.
RolloverOnStartup	• If set to 1, RSA RADIUS Server closes the current accounting log file and opens a new one each time it is restarted. A sequence number *_nnnnn* is appended to the log file name, just as when MaxSize is reached. • If set to 0, RSA RADIUS Server appends entries to the previously open accounting log file each time it is restarted. Default value is 0.
Titles	• If set to 1, each time a new accounting 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.

[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 5. *account.ini [TypeNames] Syntax*

Parameter	Meaning
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 1–4, 5, 7–14 are defined in the [TypeNames] section of `account.ini`.

```
[TypeNames]
1 = Start
2 = Stop
3 = Interim
4 = Call-Start
5 = Call-Stop
7 = On
8 = Off
9 = Tunnel-Start
10 = Tunnel-Stop
11 = Tunnel-Reject
12 = Tunnel-Link-Start
13 = Tunnel-Link-Stop
14 = Tunnel-Link-Reject
```

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:

```
[TypeNames]
1 = Start
2 = Stop
.
.
.
639 = AscendType
28 = 3ComType
```

If no string is given for a particular Acct-Status-Type, RSA RADIUS Server uses the numeric value of the incoming Acct-Status-Type attribute, formatted as a string.

certinfo.ini File

The `certinfo.ini` (server certificate information) file is an ASCII file with a single section, [Certificate_Info]. This file allows the administrator to isolate the PCKS#12 file containing the server's certificate chain and private key in a portion of the file system that is accessible to the RSA RADIUS Server process but not to general users or operators of the system.

Server Certificates

Server certificates must meet the following requirements:

- The certificate must be in PKCS#12 `.pfx` file format with a private key and all of the certificates necessary to establish the chain to the issuing certificate authority (CA).
- The certificates cannot be self-signed.
- The `certinfo.ini` file in the `\Radius\service` directory on the RSA RADIUS Server host must point to the `.pfx` file.
- Key usage must be set in one of the following ways:
  - Key usage unspecified.
  - `digitalSignature` or `keyEncipherment` is set with `no extended key usage`, `serverAuth`, `msSGC`, or `nsSGC`.

Client Certificates

Client certificates must meet the following requirements:

- Key usage must be set in one of the following ways:
  - Key usage unspecified.
  - `digitalSignature` is set with `no extended key usage` or `clientAuth` is specified.
- If the certificate is a Netscape certificate, the client bit must be set.
- If the certificate is a CA certificate only, select only extended key usage.

Table 6. *certinfo.ini [Certificate_Info] Syntax*

Parameter	Meaning
Certificate_And_Private_Key_File	Identifies the location of the PKCS#12 file containing the server's certificate chain and private key, as well as all the certificates needed to establish a chain to the CA that issued the server's certificate. This should be specified as an absolute file path to remove any ambiguity regarding the file location.
Password	Specifies the password required to retrieve the server's private key, which is included in the PKCS#12 file.

Example

```
[Certificate_Info]
; Location of the PKCS#12 file containing the certificate
; and private key of the server and all certificates necessary to
; establish a chain to the Certificate Authority that issued
; the certificate.
Certificate_And_Private_Key_File = C:\Program Files\RSA Security\RSA
    RADIUS\service\test_svr.pfx

; Password with which the private key contained in the PKCS#12
; file mentioned above was encrypted.
Password = tryme
```

classmap.ini File

The `classmap.ini` initialization file specifies what RSA RADIUS Server does with RADIUS attributes encoded in one or more Class attributes included in accounting requests.

[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 7.** classmap.ini [Attributename] Syntax*

Parameter	Meaning
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	Name of the attribute that should be added to the accounting request, which contains the original value of the attribute identified by *AttributeName*.
[,Attribute]	Name of the attribute in the accounting request that should contain the value of the attribute displaced when *AttributeName's* value replaced the existing *Attribute* value. Valid only when `replace` keyword is used.

In the following example, the encapsulated `User-Name` attribute would replace the existing `User-Name` in the accounting request.

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

eap.ini File

The `eap.ini` configuration file allows you to configure what EAP authentication types are tried when authenticating users through the different RSA RADIUS Server authentication methods.

Each authentication method against which you want EAP authentication to be performed 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:

- [EAP-TTLS]
- [EAP-PEAP]
- [SecurID]

Table 8 lists the fields contained in each section.

Table 8. *eap.ini Syntax*

Parameter	Meaning
EAP-Only	Specifies the type of credentials used in the inner authentication of authentication methods expected to handle EAP. • If set to 0, the authentication method accepts all types of user credentials. • If set to 1, the authentication method accepts only EAP credentials or acts only as a back-end server to an automatic EAP protocol method. If you are using RSA 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 RSA SecurID processing to be omitted when using EAP/Generic Token, ultimately leading to the user being rejected. Default value is 1 for [EAP-TTLS] and [EAP-PEAP]. Default value is 0 for [SecurID User].

Table 8. *eap.ini Syntax (Continued)*

Parameter	Meaning
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 are `TTLS`, `PEAP`, `Generic-Token`, `EAP-15` (RSA Security EAP), and `EAP-32` (EAP Protected One-Time Password). Leave this 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 value is 0.
Available-EAP-Types	A comma-separated or pipe-separated (|) list of the EAP protocols that can be selected when configuring the RSA RADIUS Server. Valid values are `TTLS`, `PEAP`, `Generic-Token`, `EAP-15` (RSA Security EAP), and `EAP-32` (EAP Protected One-Time Password).

NOTE: *RSA RADIUS Server is configured with an* `eap.ini` *file that works for all but the most complex or unusual environments. You should edit the* `eap.ini` *settings only when the default* `eap.ini` *file does not meet your needs.*

radius.ini File

The `radius.ini` initialization file is the main configuration file that determines the operation of the RSA RADIUS Server.

Warning: *Use caution when editing `radius.ini`, so that values pertaining to one feature are not overwritten or lost while you configure another feature. Make a backup copy of `radius.ini` before you make changes.*

[Addresses] Section

By default, RSA RADIUS Server attempts to bind all of the system's IP addresses, as reported by name services, so that it can listen for incoming RADIUS packets on all available network interfaces. On a multihomed system, some IP addresses may not be reported by name services. To ensure that RSA RADIUS Server binds one or more network addresses, edit the [Addresses] section in `radius.ini` to list the IP addresses you want RSA RADIUS Server to use. When you list one or more addresses in the [Addresses] section, addresses not listed are ignored.

The following example specifies that RSA RADIUS Server listens only on the network interfaces at 192.168.20.30 and 192.168.10.44:

```
[Addresses]
192.168.20.30
192.168.10.44
```

[Certificate] Section

The [Certificate] section of `radius.ini` (Table 9) specifies the location of the server's `certinfo.ini` file (described on "certinfo.ini File" on page 14), which is required by the EAP-TTLS and EAP-PEAP plug-ins.

Table 9. *radius.ini [Certificate] syntax*

Parameter	Meaning
Server_Certificate_Info_File	The full path of the file that contains information about the server's certificate. This is not the location of the PKCS#12 file that contains the certificate, but rather the file that contains information about it.

The following example illustrates the [Certificate] section for a Linux or Solaris host:

```
[Certificate]
Server_Certificate_Info_File =
   /opt/local/radius/certInfo.ini
```

Similarly, the following example illustrates the [Certificate] section for a Windows host:

```
[Certificate]
Server_Certificate_Info_File = C:\Program Files\RSA
   Security\RSA RADIUS\Service\certInfo.ini
```

[Configuration] Section

The [Configuration] section of `radius.ini` (Table 10) contains parameters that control the most basic behavior of the RSA RADIUS Server. The following fields may be present:

Table 10. *radius.ini [Configuration] Syntax*

Parameter	Meaning
Apply-Login-Limits	• If set to `yes`, the maximum number of concurrent connections for each user is enforced, and connection attempts that exceed the limit are rejected. • If set to `no`, connections above the limit are allowed, but an event is noted in the RADIUS server log file. Default value is `yes`.
AuthenticateOnly	• If set to 0, the normal response attributes are included in the response. • If set to 1, no response attributes are included in the response packet to an AuthenticateOnly (Service-Type 8) request. Default value is 1.

Table 10. *radius.ini [Configuration] Syntax (Continued)*

Parameter	Meaning
CheckMessageAuthenticator	Validation of Message-Authenticator can occur on receipt of an Access-Request from a RAS device. • If set to 0, incoming Message-Authenticator attributes are not validated. • If set to 1, incoming Message-Authenticator attributes are validated. Default value is 0.
ClassAttributeStyle	• If set to 1, RSA RADIUS Server uses unencrypted Class attributes with multiple ASCII keys in Access-Accept packets. • If set to 2, RSA RADIUS Server uses enhanced/encrypted Class attributes in Access-Accept packets. Default value is 2.
DisableSecondaryMakeModelSelection	If set to 0, RSA RADIUS Server: 1 Looks up the RAS 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 RAS 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 RAS device by name. If the name is found, override the make/model information defined in Step 1 or Step 2. If set to 1, RSA RADIUS Server looks up the RAS devices entry by using the source address of the request and sets the make/model according to the information specified for the client. Default value is 0.

Table 10. *radius.ini [Configuration] Syntax (Continued)*

Parameter	Meaning
LogAccept	• If set to 1, specifies that messages associated with Accepts that meet the current LogLevel should be recorded in the log file. • If set to 0, the LogAccept setting is read whenever the server receives a `HUP` signal. Default value is 1.
LogfileMaxMBytes	• If set to 0, the RADIUS 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 RADIUS server log file is closed when it reaches the specified number of megabytes (1024 x 1024 bytes), and a new RADIUS server log file using the date and time it was opened as its filename (*`YYYYMMDD_HHMM`*`.log`) is opened. Default value is 0. ***NOTE:*** *If* `LogFileMaxMBytes` *is configured for a small non-zero number, the log file may roll over within a one-minute period. If this occurs, the log file size is ignored until the minute precision clock changes to ensure that log files have unique file names.*
LogLevel	Sets the rate at which RSA RADIUS Server writes entries to the RADIUS server log file (`.LOG`). The LogLevel may be the number 0, 1, or 2: • 0 – Production logging level • 1 – Informational logging level • 2 – Debug logging level Default value is 0.

Table 10. *radius.ini [Configuration] Syntax (Continued)*

Parameter	Meaning
LogReject	• If set to 0, messages associated with Rejects that meet the current LogLevel are ignored. • If set to 1, messages associated with Rejects that meet the current LogLevel are recorded in the log file. Default value is 1.
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	Name of the location of the RSA RADIUS Server directory, which contains the database and dictionary files. If not specified, default value is the directory in which the RSA RADIUS Server service/daemon resides.

Table 10. *radius.ini [Configuration] Syntax (Continued)*

Parameter	Meaning
SendOnlyOneClassAttribute	When a user's identity information is encrypted during authentication, RSA RADIUS Server uses a special Class attribute to pass the user's encrypted identity to an accounting server. Because this typically requires more than one Class attribute to be included in the Accept response, and because some Access Points do not support echoing more than one Class attribute, you can use the SendOnlyOneClassAttribute parameter to specify how RSA RADIUS Server forwards encrypted user identity information. • If set to 1, RSA RADIUS Server creates a Class attribute containing a Class attribute flag, a server identifier, and a transaction identifier. The user identification data that would normally be stored in the Class attribute(s) is stored in the current sessions table. When RSA RADIUS Server receives an accounting request, it looks up the Class information in the current sessions table and uses it as if it had arrived in the accounting request packet. • If set to 0, RSA RADIUS Server creates one or more Class attributes to return a user's encrypted identity to the Access Point, with the assumption that the AP will forward the Class attribute(s) containing the encrypted user identification information to the accounting server. Default value is 0. ***NOTE:*** *This feature works only if accounting requests go to the same server that performs authentication. Accounting requests that go to servers other than the authenticating server are unable to use this feature.*

Table 10. *radius.ini [Configuration] Syntax (Continued)*

Parameter	Meaning
TraceLevel	Specifies the RADIUS packet tracing level: • 0 – No packet tracing • 1 – Parsed content of packets is logged • 2 – raw content and parsed content of the packet is logged Packet traces are written to the log file and can be a useful tool for troubleshooting interoperability problems. Default value is 0.

[CurrentSessions] Section

The [CurrentSessions] section of `radius.ini` (Table 11) controls the Current Sessions List.

Table 11. *radius.ini [CurrentSessions] Syntax*

Parameter	Meaning
Enable	• If set to 1, user sessions are tracked in the Current Sessions table. • If set to 0, user sessions are not tracked in the Current Sessions table. Default value is 1.
CaseSensitiveUsernameCompare	• If set to 1, the server uses case-sensitive lookups when it searches its Current Sessions List for sessions that have the same user name. • If set to 0, the server ignores case. Default value is 1.

[EmbedInClass] Section

The [EmbedInClass] section of `radius.ini` (Table 12) identifies attributes that are available during authentication processing that need to be made available in accounting requests. This feature allows billing information to be embedded in a Class attribute returned to RSA RADIUS Server by a RAS device. When RSA RADIUS Server 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 (described on page 16).

The syntax for embedding attributes is as follows:

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

Table 12. *radius.ini [EmbedInClass] Syntax*

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

[HiddenEAPIdentity] Section

The [HiddenEAPIdentity] section of `radius.ini` allows the known inner identity of EAP/TTLS 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 13. *radius.ini [HiddenEAPIdentity] Syntax*

Parameter	Meaning
IncludeInAcceptResponse	• If set to 0, inclusion of the inner identity in Access-Accept responses is disabled. • If set to 1, RSA RADIUS Server 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.

[LDAP] Section

The [LDAP] section of `radius.ini` sets the TCP port number that you want to use for communication between the LDAP server portion of RSA RADIUS Server and any LDAP clients.

The syntax is as follows:

```
[LDAP]
Enable = 1
TCPPort = 667
CachePasscodesMin = 3
```

Table 14. *radius.ini [LDAP] Syntax*

Parameter	Meaning
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:*** *The LDAP Configuration Interface (LCI) is an optional add-on for your edition of RSA RADIUS Server. You must license the LCI before you can configure or use it. After you enter the LCI license key, you must set this parameter to 1. Refer to the RSA RADIUS Server Administrator's Guide for information on how to enter the LCI license key.*
TCPPort	The TCP port number that you want to use. If this setting is not present, RSA RADIUS Server uses TCP port 389. The default version of the `radius.ini` file specifies TCP port 667.
CachePasscodesMin	Number of minutes the RSA RADIUS Server caches user passcodes during authentication. Default value is three minutes.

[LDAPAddresses] Section

The [LDAPAddresses] section of `radius.ini` specifies the interfaces on which RSA RADIUS Server listens for 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:

```
[LDAPAddresses]
199.198.197.196
196.197.198.199
```

If the [LDAPAddresses] section is omitted, RSA RADIUS Server listens for LCI requests on all bound IP interfaces.

[Ports] Section

The [Ports] section of `radius.ini` provides a method for setting the UDP ports used by RSA RADIUS Server.

- 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 Solaris or 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 in `/etc/services` 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: *Any failure to bind to one of the selected UDP ports causes the affected subcomponent (authentication or accounting) to fail to initialize.*

Table 15. *radius.ini [Ports] Syntax*

Parameter	Meaning
UDPAuthPort	The UDP port(s) used for authentication (one line per port assignment). Default values are 1645 and 1812.
UDPAcctPort	The UDP port(s) used for accounting (one line per port assignment). Default values are 1646 and 1813.

Example

```
[Ports]
SecureTcpAdminPort = 1813
UDPAuthPort = 1645
UDPAuthPort = 1812
UDPAcctPort = 1646
UDPAcctPort = 1813
```

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.

[SecurID] Section

The [SecurID] section of `radius.ini` contains items specific to RSA SecurID authentication for ISDN users. It provides information that allows RSA RADIUS Server to cache the user's credentials after the user is authenticated. This technique is necessary to permit a second ISDN B-channel to be authenticated during the user's session. RSA RADIUS Server uses the cached token to authenticate the second channel.

***NOTE:** If this feature is not enabled, users who use EAP-GTC or PAP to authenticate against an RSA Authentication Manager database over an ISDN connection that bonds both B-channels will fail to authenticate because of an RSA SecurID security violation. ISDN users running only one B-channel are not affected. This feature is not available for authentication methods other than EAP-GTC and PAP.*

***Table 16.** radius.ini [SecurID] Syntax*

Parameter	Meaning
CachePasscodes	A value of `yes` means passcode caching is enabled. A value of `no` means passcode caching is disabled. Default value is `no`.
SecondsToCachePasscodes	The number of seconds to retain the cached RSA SecurID credentials (PIN and token code). Default value is 60 seconds.

securid.ini File

The `securid.ini` file allows you to specify user access settings and to 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 through RSA SecurID and the default prompt strings are too long for the screen on the authentication device.

[Configuration] Section

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

```
[Configuration]
AllowSystemPins = 0
CheckUserAllowedByClient = 1
DefaultProfile = DEFAULT
```

Table 17. *securid.ini [Configuration] Syntax*

Parameter	Meaning
`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 RAS. • If set to 0, the RADIUS server does not verify the user is allowed to connect through the RAS. Default value is 1. ***NOTE:*** *If this parameter is set to 1, RAS 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 RSA Security EAP (EAP-15) and Protected One-Time Password (EAP-32) authentication.

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

***Table 18.** securid.ini [Configuration] Syntax*

Parameter	Meaning
`Greeting`	A string of as many as 80 characters returned to a RAS after a user is authenticated. For example, “Welcome to RSA Security Software.”
`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 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.

SecurID Prompts Section

If the `securid.ini` file is present in the RSA RADIUS Server directory, RSA RADIUS Server 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, RSA RADIUS Server 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.

- Strings that require one `%s` placeholder have names with a `_S` suffix.
- Strings that require no `%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 19 lists other formatting conventions for the `securid.ini` file.

***Table 19.** 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 RSA RADIUS Server. 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 displays the default prompt strings, which may be too long for some RSA SecurID displays. Although text lines in this display appear to wrap to a second line, text wrapping is not supported in `securid.ini` entries.

```
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
; BEGINNING OF the original prompts. These will also appear by default if
; none of the samples in this file are enabled (leading ';' removed)
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;[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
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
; END OF the default prompts
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
```

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      = ARE YOU PREPARED TO HAVE THE SYSTEM\r\nGENERATE A 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\nthen 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
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
; END OF 2 lines by 40 characters prompts
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
```

Figure 2 *2 x 40 Display Substitution Strings*

Example 3: Terse Substitution Strings

This example 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       = Reenter pin
;InputReenterPin_1     = Reenter pin 1
;OutputReject          = Reject
;OutputChange          = Change
;OutputAccepted        = Accepted
;OutputDenied          = Denied
;OutputNoPassReqd      = No pass reqd
;OutputDeniedFinal     = Denied final
;Characters            = chars
;Digits                = digits
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
; END OF extremely terse prompts
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
```

Figure 3 *Terse Substitution Strings*

spi.ini File

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

All RSA RADIUS Servers that may receive authentication and accounting requests from a common RAS or AP must be configured with similar `spi.ini` files, which must list the IP addresses of all the servers in that "realm." 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 realm.

[Keys] Section

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

```
[Keys]
CurrentKey = n
1 = value
2 = value
.
.
.
```

Table 20. *spi.ini [Keys] Syntax*

Parameter	Meaning
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 RSA RADIUS Server does not exchange encrypted Class attributes with other servers. • *n* – Use the key specified below to encrypt Class attributes. Default value is 0.
n = *value*	Specifies the number of the current encryption key.

In the following example, the RSA RADIUS Server generates a unique random key to encrypt Class attributes.

```
[Keys]
CurrentKey = 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 realm.

```
[Keys]
CurrentKey = 2
1 = firstkey
2 = swordfish
3 = mypassword
```

[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 realm.

```
[Hosts]
192.168.15.21
192.168.23.121
192.168.23.205
```

vendor.ini File

The `vendor.ini` initialization file contains information that allows RSA RADIUS Server to work with the products of other vendors.

[Vendor-Product Identification] Section

The [Vendor-Product Identification] section of `vendor.ini` identifies and provides information about the network access servers that can be used with RSA RADIUS Server. Table 21 lists the fields that may be present for each make/model of a vendor product.

Table 21. *vendor.ini [Vendor-Product Identification] Syntax*

Parameter	Meaning
vendor-product	This required field 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 window. This list is used when adding a new RADIUS client or when selecting a vendor-specific attribute.
dictionary	This required field specifies the dictionary file to use for this product. The dictionary file must be located in the same directory as the RSA RADIUS Server daemon or service. You do not need to specify an extension on the dictionary name; RSA RADIUS Server automatically attaches an extension of `.DCT` to the dictionary names listed in this field.
call-filter-attribute	Specifies the attributed used for call filter functions. Used only by Ascend/Lucent RAS equipment.
challenge-response-attribute	Specifies the attribute number in which a RAS 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 RAS equipment.
help-id	Help context for the vendor's product in the vendor information help file.

Table 21. *vendor.ini [Vendor-Product Identification] Syntax (Continued)*

Parameter	Meaning
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	This field determines whether RSA RADIUS Server may infer that one user has logged off if the port assigned to that user is now being used by another user. • If set to `No`, such an inference is made and the previous user is removed from the Active Users list. • If set to `Yes`, no such inference is made and both users are deemed active. Default value is `No`.
max-eap-fragment	You can specify the size of the maximum EAP-Message in the `ttlsauth.aut`, `peapauth.aut`, and `tlsauth.eap` files. Default for the maximum fragment length is 1020. This is inefficient, however, as the fragment length must be set to a number low enough to work with all of a customer's Access Points. This setting allows specifying 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.
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.

Table 21. *vendor.ini [Vendor-Product Identification] Syntax (Continued)*

Parameter	Meaning
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 do not handle this 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 RAS on how long to 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`.

Chapter 3
Dictionary Files

This chapter describes the usage and settings for the RSA RADIUS Server attribute processing and dictionary files, which specify RADIUS attributes.

***NOTE:** The dictionary files for RSA RADIUS Server must remain in the installation directory. Do not move the files to other locations on your computer.*

Overview

For each product listed in the `vendor.ini` file (described on page 39), RSA RADIUS Server provides a dictionary (`.dct`) file. Dictionary files enable RSA RADIUS Server to exchange attributes with RADIUS clients. Like initialization files, dictionary files are loaded at startup time, and reside in the RSA RADIUS Server directory:

```
*.dct
dictiona.dcm
```

- Dictionary files identify the attributes RSA RADIUS Server should expect when receiving RADIUS requests from a specific type of device.
- Dictionary files identify the attributes RSA RADIUS Server should include when sending a RADIUS response to a specific type of device.

Figure 4 illustrates a 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
ATTRIBUTE Juniper-Allow-Configuration   Juniper-VSA(4, string) r
ATTRIBUTE Juniper-Deny-Configuration    Juniper-VSA(5, string) r

##################################################################
# Juniper.dct - Juniper Networks dictionary
##################################################################
```

Figure 4 *Sample Dictionary File*

Dictionary File Location

Windows: Dictionary files must be placed in the same directory as the RSA RADIUS Server service. While starting up, RSA RADIUS Server 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.

Solaris/Linux: Dictionary files must be placed in the same directory as the RSA RADIUS Server daemon. During initialization, RSA RADIUS Server 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 22.

Table 22. *Dictionary File Keywords*

Keyword	Meaning
@	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 RSA RADIUS Server reflect specific vendors' implementations of RADIUS clients. Therefore, you do not usually need to modify the dictionary files shipped with RSA RADIUS Server. However, if you are in communication with your RAS vendor about a new product, a new attribute, or a new value for an attribute, you can add this information to your existing RSA RADIUS Server configuration by editing dictionary files.

Before you edit an existing dictionary file or create a new one to integrate your changes into RSA RADIUS Server, you must do the following:

See "vendor.ini File" on page 39.

1. Add a new vendor-product entry to `vendor.ini` so that you can reference the new dictionary while configuring RSA RADIUS Server.
2. Place your dictionary file in the same directory as the RSA RADIUS Server 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 23. *ATTRIBUTE Record Syntax*

Parameter	Meaning
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. See "Flag Characters" on page 49 for information on flag characters.

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 RSA RADIUS Server 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 information:

- 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 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 RSA RADIUS Server database.

See "Flag Characters" on page 49.

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.

Syntax Type Identifier

Standard `syntax_type` identifiers are listed in Table 24.

***Table 24.** Syntax Type Identifiers*

Syntax Type	Meaning
hexadecimal	Hexadecimal string
hex4	4-byte hexadecimal number
int1, int4, integer	1- or 4-byte decimal number (integer is equivalent to int4)
ipaddr	IP address or IP netmask attribute
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)

Compound Syntax Types

In addition to the standard `syntax_type` identifiers listed in Table 24, 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 Table 25 can be combined inside square brackets to form a compound syntax type.

***Table 25.** Compound Syntax Types*

Option	Meaning
vid=*nnn*	The device manufacturer's SMI Network Management Private Enterprise code (assigned by ISO) in decimal form.
type*N*=*nnn*	Type 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.

Table 25. Compound Syntax Types (Continued)

Option	Meaning
len*N*=*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).
data=*syntax_type*	The actual data to be included in the attribute; the syntax can be any of the standard syntax types.
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 in order for the attribute to include a tag field. A value of 0 indicates that the field should be present but ignored.

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` field consists of the concatenation of one or more characters from the list in Table 26.

Table 26. Flag Characters

Flag Character	Meaning
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 a single time within a user or profile checklist.
C	Attribute can appear multiple times within a user or profile checklist.
r	Attribute can appear a single time within a user or profile return list.
R	Attribute can appear multiple times within a user or profile return list.
o *or* O	Attribute is orderable; the administrator can control the order in which such attributes are stored in the RSA RADIUS Server 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_name value_name integer_value`

Table 27. *VALUE Records*

Parameter	Meaning
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       Framed-Protocol     SLIP          2
```

Using these dictionary records, the administrator does not need to 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 RSA RADIUS Server 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 28. *MACRO Records*

Parameter	Meaning
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:

```
MACRO Cisco-VSA(t, s) 26 [vid=9 type1=%t% len1=+2
   data=%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 RSA RADIUS Server 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 29.

Table 29. *OPTION Records*

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

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

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

Chapter 4
EAP Configuration Files

This chapter describes the EAP configuration files. These files are loaded at startup time and reside in the RSA RADIUS Server directory.

NOTE: *The EAP configuration files for RSA RADIUS Server must remain in the installation directory. Do not move the files to other locations on your computer.*

File	Page
peapauth.aut File	page 54
ttlsauth.aut File	page 59

peapauth.aut File

The EAP-PEAP plug-in is configured through the `peapauth.aut` file. This configuration file is read each time the RSA RADIUS Server restarts or receives a HUP signal.

Note that you must configure the [Certificate] section of `radius.ini` to specify the path to the file that describes the server's certificate. For more information, refer to "[Certificate] Section" on page 19.

[Bootstrap] Section

The [Bootstrap] section of the `peapauth.aut` file (Table 30) specifies information that RSA RADIUS Server uses to load the EAP-PEAP authentication method.

Table 30. *peapauth.aut [Bootstrap] Syntax*

[Bootstrap] Field	Meaning
LibraryName	Specifies the name of the EAP-PEAP module. Default value is `peapauth.dll` for Windows and `peapauth.so` for Solaris and Linux. Do not change this unless you are advised to do so by RSA Security Customer Support.
Enable	Specifies whether the EAP-PEAP authentication module is enabled. • 0 – EAP-PEAP is disabled. • 1 – EAP-PEAP is enabled. Default value is 1.
InitializationString	Specifies the name of the authentication method . 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 31) allows you to configure the basic operation of the EAP-PEAP plug-in.

Table 31. *peapauth.aut [Server_Settings] Syntax*

[Server_Settings] field	Meaning
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. Default value is 1.
DH_Prime_Bits	This attribute selects the size prime that the module uses for Diffie-Hellman modular exponentiation. The larger the prime, the less susceptible the system is to certain types of attacks. The smaller the prime, the cheaper (in CPU terms) the Diffie-Hellman key agreement operation. Supported values are 512, 1024, 1536, 2048, 3072 and 4096. Default value is 1024.

Table 31. *peapauth.aut [Server_Settings] Syntax (Continued)*

[Server_Settings] field	Meaning
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,” and other TLS-related RFCs and draft RFCs. Default value is: `0x16, 0x13, 0x66, 0x15, 0x12, 0x0a, 0x05, 0x04, 0x07, 0x09`.
PEAP_Min_Version	Specifies the minimum version of the PEAP protocol that the server should negotiate: 0 – Negotiate version 0, which is compatible with Microsoft's initial PEAP implementation (shipped in Microsoft Windows XP Service Pack 1). 1 – Negotiate version 1, which is compatible with Cisco Systems initial PEAP implementation (shipped in Cisco Systems ACU). Default value is 0. The value entered in this field must be less than or equal to the value entered for PEAP_Max_Version.
PEAP_Max_Version	Specifies the maximum version of the PEAP protocol that the server should negotiate: 0 – Negotiate version 0, which is compatible with Microsoft's initial PEAP implementation (shipped in Microsoft Windows XP Service Pack 1). 1 – Negotiate version 1, which is compatible with Cisco Systems initial PEAP implementation (shipped in Cisco Systems ACU). Default value is 1. The value entered in this field must be equal to or greater than the value entered for PEAP_Min_Version.

[Session_Resumption] Section

The [Session_Resumption] section (Table 32) allows you to specify whether session resumption is permitted and under what conditions session resumption is performed.

Table 32. *peapauth.aut [Session_Resumption] Syntax*

[Session_Resumption] field	Meaning
Session_Timeout	Set this attribute to the maximum number of seconds you want the client to remain connected to the RAS or AP before having to reauthenticate. If not set to 0, the lesser of this value and the remaining resumption limit (see the following description) is sent in a Session-Limit attribute to the RAS or AP 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. Setting of a value such as 600 (10 minutes) does not necessarily cause a full reauthentication to occur every 10 minutes. The resumption limit can be configured to make most reauthentications fast and computationally cheap.
Termination_Action	Set this attribute to the integer value that you want returned in a Termination-Action attribute. This is a standard attribute supported by most Access Points and determines what happens when the session timeout is reached. If you do not specify a value for this attribute, the plug-in does not generate such an attribute. This does not prevent the authentication methods performing secondary authorization from providing a value for this attribute. Default is to not send this attribute.

Table 32. *peapauth.aut [Session_Resumption] Syntax (Continued)*

[Session_Resumption] field	Meaning
Resumption_Limit	Set this attribute to the maximum number of seconds you want the client to be able to reauthenticate using the TLS session resumption feature. This type of reauthentication 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.

ttlsauth.aut File

The EAP-TTLS plug-in is configured by modifying the settings in the `ttlsauth.aut` file.

Note that you must configure the [Certificate] section of `radius.ini` to specify the path to the file that describes the server's certificate. For more information, refer to "[Certificate] Section" on page 19.

[Bootstrap] Section

The [Bootstrap] section of the `ttlsauth.aut` file (Table 33) specifies information that RSA RADIUS Server uses to load the EAP-PEAP authentication method.

***Table 33.** ttlsauth.aut [Bootstrap] Syntax*

[Bootstrap] Field	Meaning
LibraryName	Specifies the name of the EAP-PEAP module. Default value is `ttlsauth.dll` for Windows and `ttlsauth.so` for Solaris and Linux. Do not change this unless you are advised to do so by RSA Security Customer Support.
Enable	Specifies whether the EAP-TTLS authentication module is enabled. • 0 – EAP-TTLS is disabled. • 1 – EAP-TTLS is enabled. Default value is 1.
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-TTLS.

[Server_Settings] Section

The [Server_Settings] section (Table 34) allows you to configure the basic operation of the plug-in.

Table 34. *ttlsauth.aut [Server_Settings] Syntax*

[Server_Settings] field	Meaning
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. Default value is 1.
DH_Prime_Bits	This attribute selects the size prime that the module uses for Diffie-Hellman modular exponentiation. The larger the prime, the less susceptible the system is to certain types of attacks. The smaller the prime, the cheaper (in CPU terms) the Diffie-Hellman key agreement operation. Supported values are 768, 1024, 1536, 2048, 3072 and 4096. Default value is 1024.

Table 34. ttlsauth.aut [Server_Settings] Syntax (Continued)

[Server_Settings] field	Meaning
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," and other TLS-related RFCs and draft RFCs. Default value is: `0x16, 0x13, 0x66, 0x15, 0x12, 0x0a, 0x05, 0x04, 0x07, 0x09.`

[Session_Resumption] Section

The [Session_Resumption] section allows you to specify whether session resumption is permitted and under what conditions session resumption is performed.

Table 35. ttlsauth.aut [Session_Resumption] Syntax

[Session_Resumption] field	Meaning
Session_Timeout	Set this attribute to the maximum number of seconds you want the client to remain connected to the RAS or AP before having to reauthenticate. If not set to 0, the lesser of this value and the remaining resumption limit (see the following description) is sent in a Session-Limit attribute to the RAS or AP 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. Setting of a value such as 600 (10 minutes) does not necessarily cause a full reauthentication to occur every 10 minutes. The resumption limit can be configured to make most reauthentications fast and computationally cheap.

Table 35. *ttlsauth.aut [Session_Resumption] Syntax (Continued)*

[Session_Resumption] field	Meaning
Termination_Action	Set this attribute to the integer value that you want returned in a Termination-Action attribute. This is a standard attribute supported by most Access Points and determines what happens when the session timeout is reached. If you do not specify a value for this attribute, the plug-in does not generate such an attribute. This does not prevent the authentication methods performing secondary authorization from providing a value for this attribute. Default value is to not send this attribute.
Resumption_Limit	Set this attribute to the maximum number of seconds you want the client to be able to reauthenticate using the TLS session resumption feature. This type of reauthentication 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 ttlsauth.aut File

```
[Bootstrap]
LibraryName=ttlsauth.dll
Enable=1
InitializationString=EAP-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_Keys = 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 = 0x16, 0x13, 0x66, 0x15, 0x12, 0x0a, 0x05,
   0x04, 0x07, 0x09

[Session_Resumption]
; Maximum length of time (in seconds) the NAS/AP will
   allow
; the session to persist before the client is asked
; to reauthenticate.
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
```

For this to work, you must also provide the following settings in the [ttlsauth] section of the `eap.ini` file:

```
   First-Handle-Via-Auto-EAP = 0
   EAP-Type = TTLS
; 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
```

For this to work, you must also provide the following settings in the [ttlsauth] section of the `eap.ini` file:

```
   First-Handle-Via-Auto-EAP = 0
   EAP-Type = TTLS
```

Index

Symbols

/etc/services 29

A

Accept 22
account.ini 6
  Alias/name 6
  Attributes 7
  Configuration 8
  Settings 9
  TypeNames 12
Acct-Status-Type 12
Acct-Status-Type attribute 12
Alias/name section
  in account.ini 6
AllowSystemPins 31
Apply-Login-Limits 20
attributeName 27
Attributename section
  in classmap.ini 16
Attributes section
  in account.ini 7
AuthenticateOnly 20
Available-EAP-Types 18

B

Bootstrap section
  in peapauth.aut 54, 59
BufferSize 9

C

CachePasscodes 30
CachePasscodesMin 28
call-filter-attribute 39
Call-Start (Acct-Status-Type) 12
Call-Stop (Acct-Status-Type) 12
Carryover 9
CaseSensitiveUsernameCompare 25
certificate chain 14
Certificate section
  in radius.ini 19
certinfo.ini 14, 19
challenge-response- attribute 39
CheckMessageAuthenticator 21
CheckUserAllowed ByClient 31
Cipher_Suites 56, 61
Class attributes 16
ClassAttributeStyle 21
classmap.ini 16, 26
  AttributeName 16
cluster, for class attribute encryption 37
Configuration section
  in account.ini 8
  in radius.ini 20
CurrentKey 37
CurrentSessions section
  in radius.ini 25

D

data-filter-attribute 39
DefaultProfile 31
DH_Prime_Bits 55, 60
Dictionary 39
Diffie-Hellman 55, 60
DisableSecondaryMakeModelSelection 21

E

eap.ini 17, 18, 63
EAP-15 18
EAP-32 18
EAP-Only 17

EAP-PEAP 17, 19
EAP-TTLS 17, 19
EAP-Type 18
EmbedInClass section
in radius.ini 26
Enable 10, 54, 59
encryption key 37
exponentiation 55, 60

F

First-Handle-Via-Auto-EAP 18

G

Greeting 32

H

help-id 39
HiddenEAPIdentity section
in radius.ini 26
Hosts section
in spi.ini 38

I

ignore-acct-ss 40
ignore-ports 40
IncludeInAcceptResponse 27
InitializationString 54, 59
Interim (Acct-Status-Type) 12

K

Keys section
in spi.ini 37

L

LDAP section
in radius.ini 27
LDAPAddresses section
in radius.ini 28
LibraryName 54, 59
LineSize 10
LogAccept 22
LogDir 8
LogFileMaxMBytes 22
LogLevel 22
LogReject 23

M

macro records 51
Max-EAP-Fragment 40
MaxSize 10
modular exponentiation 55, 60

N

name 6
New PIN mode 31

O

Off (Acct-Status-Type) 12
On (Acct-Status-Type) 12

P

PCKS#12 file 14
PEAP_Max_Version 56
PEAP_Min_Version 56
peapauth.aut 40, 54
PhantomTimeout 23
PIN 31
PIN, system-generated 31
port-number-usage 40
Ports section
in radius.ini 29
POTP
private key 14
PrivateDir 23
product-scan-acct 40
product-scan-auth 41
Protected One-Time Password, see POTP

Q

QuoteBinary 10
QuoteInteger 10
QuoteIPAddress 10
QuoteText 11
QuoteTime 11

R

radius.ini 19
  Certificates 19
  Configuration 20
  CurrentSessions 25
  EmbedInClass 26
  HiddenEAPIdentity 26
  LDAP 27
  LDAPAddresses 28
  Ports 29
  SecurID 30
radiusdir xi
Reject 23
Resumption_Limit 58, 62
Return_MPPE_Keys 32, 55, 60
Rollover 11
RolloverOnStartup 11

S

SecondsToCachePasscodes 30
SecurID section
  in radius.ini 30
SecurID User 17
securid.ini 31
send-class-attribute 41
SendOnlyOneClassAttribute 24
send-session-timeout-on-challenge 41
Server_Certificate_Info_File 19
Session_Timeout 57, 61
Settings section
  in account.ini 9
size limit in RADIUS system RADIUS system log
  size limit 22
spi.ini 37
Start (Acct-Status-Type) 12
Stop (Acct-Status-Type) 12
system-generated PIN 31

T

TCPPort 28
Termination_Action 57, 62
Titles 11
TLS_Message_Fragment_Length 55, 60
tlsauth.eap 40
TraceLevel 25
ttlsauth.aut 40, 59
Tunnel-Link-Reject (Acct-Status-Type) 12
Tunnel-Link-Start (Acct-Status-Type) 12
Tunnel-Link-Stop (Acct-Status-Type) 12
Tunnel-Reject (Acct-Status-Type) 12
Tunnel-Start (Acct-Status-Type) 12
Tunnel-Stop (Acct-Status-Type) 12
TypeNames section
  in account.ini 12

U

UDP port 29
UDPAcctPort 29
UDPAuthPort 29
UTC 11

V

vendor.ini 39
Vendor-Product 39
VendorSpecificAttribute 6