RADIUS Administrator's Guide

Livingston Enterprises, Inc.
6920 Koll Center Pkwy #220
Pleasanton, CA 94566
(510) 426-0770
(800) 458-9966

October 1996

950-1206A

Copyright and Trademarks

© 1996 Livingston Enterprises, Inc. All rights reserved.

The product names “ChoiceNet,” “ComOS,” “IRX,” “PortMaster,” “PMconsole,” “RADIUS,” and “True Digital” are trademarks belonging to Livingston Enterprises, Inc.

All other trademarks are the property of their respective owners.

Disclaimer

Livingston Enterprises, Inc. makes no express or implied representations or warranties with respect to the contents or use of this manual, and specifically disclaims any implied warranties of merchantability or fitness for a particular purpose. Livingston Enterprises, Inc. further reserves the right to revise this manual and to make changes to its content at any time, without obligation to notify any person or entity of such revisions or changes.

FCC Class A Notice - United States

Computing devices and peripherals manufactured by Livingston Enterprises, Inc. generate, use, and can radiate radio frequency energy, and if not installed and used in accordance with the instructions contained in this manual, may cause interference to radio communications. Such equipment has been tested and found to comply with the limits for a Class A computing device pursuant to Subpart J of Part 15 of the FCC Rules, which are designed to provide reasonable protection against radio interference when operated in a commercial environment. Operation of this equipment in a residential area is likely to cause interference, in which case the user — at his own expense — will be required to take whatever measures may be required to correct the interference.

Some components may not have been manufactured by Livingston Enterprises, Inc. If not, Livingston Enterprises has been advised by the manufacturer that the component has been tested and complies with the Class A computing device limits as described above.

Table of Contents

About this Guide vii

Preview of this Guide vii

Related Documentation viii

Document Conventions ix

Contacting Livingston Technical Support x

1. Overview 1-1

Introduction to RADIUS 1-1

How RADIUS Works 1-2

Getting Started 1-5

2. RADIUS Server Configuration 2-1

Introduction 2-1

RADIUS Installation 2-2

Installation with pminstall 2-2

Installation without pminstall 2-3

Configuring Client Information 2-6

3. RADIUS Client Configuration 3-1

Configuration Using Command Line Interface 3-1

Configuration Using PMconsole 3-3

4. Configuring User Information 4-1

Introduction 4-1

Username 4-2

Check Items 4-2

Passwords 4-2

Client Information 4-5
Prefixes and Suffixes 4-5
Reply Items 4-6
Service Type 4-6
Framed Protocol 4-7
Framed IP Address 4-8
Framed IP Netmask 4-8
Framed Route 4-8
Outbound-User 4-9
Callback Information 4-10
Routing 4-11
Packet Filters 4-12
Access Filters 4-12
Remote Host Information 4-13
MTU 4-14
Compression 4-14
IPX Network 4-15
RADIUS 2.0 Reply Items 4-16
Default User Entries 4-17
RADIUS DBM Database 4-19
User Entry Check and Reply Items: Complete Listing 4-20
Examples 4-25
PPP User Entry 4-25
Using Prefixes 4-26

5. RADIUS Menus **5-1**
Introduction 5-1
How Menus Work 5-1
Single-Level Menu 5-2
Nested Menus 5-3
6. SecurID Installation **6-1**
Introduction 6-1
SecurID Installation 6-3
Progress 6-3
ACE/Server 6-3
sdadmin 6-5
sdshell 6-6
RADIUS Configuration 6-8
New PIN Assignment Using RADIUS 6-8
Next Cardcode 6-10
Troubleshooting 6-11
sdadmin Cannot Find First Token 6-11
sdserv.bi and sdlog.bi Consume Too Much Disk Space 6-12
sdadmin Runs out of Memory 6-12
7. RADIUS Accounting **7-1**
Introduction 7-1
How RADIUS Accounting Works 7-1
Getting Started 7-2
Client Configuration 7-3
Server Configuration 7-3
RADIUS Accounting Flags 7-3
Accounting Attributes 7-4

Acct-Status-Type 7-4
Acct-Delay-Time 7-4
Acct-Session-Id 7-4
Acct-Authentic. 7-4
Acct-Session-Time 7-5
NAS-Port-Type 7-5
Acct-Input-Octets and Acct-Output-Octets 7-5
Called-Station-Id and Calling-Station-Id 7-5
Timestamp 7-5
Request-Authenticator 7-5
Acct-Terminate-Cause. 7-6
Examples 7-7
A. Troubleshooting A-1
Introduction A-1
Troubleshooting RADIUS Authentication. A-1
Checking radiusd Daemon A-1
Checking the PortMaster A-2
Checking /etc/raddb/users A-2
Host Unavailable. A-3
Invalid Login after 30 second wait A-3
Result of radiusd -x output A-4
Troubleshooting RADIUS Accounting. A-5

Figures

Figure 2-1 RADIUS Directory Structure 2-1
Figure 4-1 User Entry 4-1

Tables

Table 2-1 radiusd Flags 2-5

Table 4-1 Service-Type 4-6

Table 4-2 Framed-Routing Options 4-11

Table 4-3 Login-Service 4-13

Table 4-4 User Entry Check and Reply Items 4-20

Table 7-1 radiusd Accounting Daemon Flags 7-3

Table 7-2 Session Termination Causes. 7-6

Preface

About this Guide

This guide provides complete installation and configuration instructions for the Livingston Enterprises Remote Authentication Dial-In User Service (RADIUS™). This guide covers RADIUS server release 2.0.

RADIUS may be used in conjunction with the Livingston PortMaster™ family of products. To install and configure these products, see "Related Documentation" on page viii of the Preface.

This guide is designed to be used by qualified system administrators and network managers. Knowledge of UNIX and basic networking concepts is required to successfully install RADIUS.

Preview of this Guide

The RADIUS Administrator's Guide includes the following chapters:

Chapter 1, "Overview" gives an introduction to RADIUS.

Chapter 2, "RADIUS Server Configuration" provides step-by-step configuration instructions for RADIUS servers.

Chapter 3, "RADIUS Client Configuration" provides step-by-step configuration instructions for RADIUS clients.

Chapter 4, "Configuring User Information" describes how to configure user entries on the RADIUS server.

Chapter 5, "RADIUS Menus" describes the RADIUS menu feature.

Chapter 6, "SecurID Installation" provides a quick reference for Security Dynamics ACE/Server and ACE/Client installation.

Chapter 7, "RADIUS Accounting" describes how to log RADIUS security information.

Troubleshooting information is included in Appendix A.

Related Documentation

The following manuals are available from Livingston Enterprises. These manuals are included with most Livingston products; if these were not shipped with your unit, contact Livingston for ordering information.

- *Hardware Installation Guides*

  These guides contain complete hardware installation instructions. A *Hardware Installation Guide* is available for each PortMaster product line (IRX™, Office Router, Communications Server, and Integrated Access Server).

- *Configuration Guide for PortMaster Products*

  This guide provides a comprehensive overview of networking and configuration issues related to the PortMaster series of products.

- *Command Line Administrator's Guide*

  This guide provides the complete description and syntax of each command in the ComOS™ command set.

- *PMconsole for Windows Administrator's Guide*

  This guide covers PMconsole™ for Windows, a graphic configuration tool that may be used to configure the PortMaster. The majority of the material in this guide also applies to the UNIX version of PMconsole.

Document Conventions

The following conventions are used in this guide:

Convention	**Represents**	**Example**
Regular font	Normal text typeface	This is the default.
Bold font	Names of commands, parameters, and PMconsole menu options when used in body text.	Use **version** to display the version number.
Italic font	A command line variable to be replaced with a string or value	To set the IP address of the Ethernet interface: set *Ether0* address *Ipaddress*
[Regular font]	Optional arguments that may be used in commands.	set nameserver [2] *Ipaddress*
Vertical bar (|)	Separates two or more options in a command.	set debug isdn on|off

Contacting Livingston Technical Support

Free lifetime software support and software upgrades are available for every current Livingston product. The PortMaster comes with a one-year warranty.

To obtain technical support, contact Livingston Monday through Friday between the hours of 6 a.m. and 5 p.m. (GMT -8) Please record your ComOS version number and report it to the technical support staff.

- By voice, dial (800) 458-9966 within the USA (including Hawaii), Canada and the Caribbean or +1 (510) 426-0770 from outside these areas
- By FAX, dial +1 (510) 426-8951
- By electronic mail, send mail to **support@livingston.com**
- Using the World Wide Web, see **http://www.livingston.com/**

One-hour installation appointments may be scheduled in advance by calling the technical support phone number listed above.

New releases and upgrades of Livingston software are available via anonymous FTP from **ftp.livingston.com**.

Livingston maintains the following Internet mailing lists for PortMaster users:

- portmaster-users

  A discussion of general and specific PortMaster issues, including configuration and troubleshooting suggestions. To subscribe, send electronic mail to **majordomo@livingston.com** with **subscribe portmaster-users** in the body of the message.

  The mailing list is also available in a daily digest format. To receive the digest, send electronic mail to **majordomo@livingston.com** with **subscribe portmaster-users-digest** in the body of the message.

- portmaster-announce

  Announcements of new PortMaster products and software releases. To subscribe, send electronic mail to **majordomo@livingston.com** with **subscribe portmaster-announce** in the body of the message. All announcements to this list also go to the portmaster-users list. You do not need to subscribe to both lists.

Overview 1

Introduction to RADIUS

The Remote Authentication Dial-In User Service (RADIUS) is a client/server security protocol created by Livingston Enterprises. Security information is stored in a central location, known as the RADIUS server. RADIUS clients (such as a PortMaster communications server) communicate with the RADIUS server to authenticate users. Although the term **RADIUS** refers to the network protocol that the client and server use to communicate, it is often used to refer to the entire client/server system.

RADIUS offers the following advantages:

- Tight security

  In large networks, security information may be scattered throughout the network on different devices. RADIUS allows user information to be stored on one host, minimizing the risk of security loopholes. All authentication and access to network services is managed by the host functioning as the RADIUS server.

- Flexibility

  RADIUS server software is distributed in source code format to Livingston customers. Using modifiable “stubs,” RADIUS can be adapted to work with existing security systems and protocols. The RADIUS server may be adapted to your network, rather than adjusting your network to work with RADIUS.

  RADIUS may be used with any communications server that supports the RADIUS protocol. When new security technology becomes available or security needs increase, RADIUS may be expanded to offer new services.

- Simplified management

  Security information is stored in text files at a central location, the RADIUS server. Adding new users to the database or modifying existing user information can be easily accomplished by editing these text files.

- Extensive logging capabilities

  RADIUS provides extensive audit trail capabilities, referred to as RADIUS accounting. Information collected in a log file can be analyzed for security purposes, or used for billing.

The RADIUS server is available for the following operating systems:

- SunOS 4.1.4
- Solaris 2.5
- HP/UX 10.01
- Linux 1.2.13 (ELF)
- AIX 3.2.5
- SGI Irix 5.2
- DEC Alpha OSF/1 3.0
- BSD/OS 2.0

How RADIUS Works

The three main functions of RADIUS are authentication, authorization, and accounting.

- Authentication

  RADIUS is used to authenticate users for dial-in remote access. Authentication information may be stored in a local **users** file or accessed from external authentication mechanisms such as a UNIX password file or SecurID ACE/Server.

  For example, user bob may attempt to log into a PortMaster. The following authentication sequence would take place:

  1. The PortMaster asks bob for his username and password, then compares the username/password pair to the PortMaster User Table.
  2. If the username is not found in the User Table and security for the port is set to **on**, the PortMaster sends an **access-request** message to the RADIUS server, if one is defined. This message asks the RADIUS server to authenticate the user.

3. The RADIUS server checks its database to determine if user bob is present. In order for bob's login to be successful, a matching username and password must be found in the RADIUS database.
4. If a matching password is found in the RADIUS users file, the RADIUS server sends an **access-accept** message to the PortMaster, which lets the PortMaster know that bob has been successfully authenticated. It also sends authorization information about the services bob may access and configuration information about his connection.
5. If a matching password is not found in the RADIUS users file, the RADIUS server sends an **access-reject** packet, which lets the PortMaster know that the authentication attempt has failed. The PortMaster prevents bob's connection attempt.

- Authorization

  Authorization controls access to specific services on the network. Once a user is authenticated, RADIUS tells the PortMaster what a user is **authorized** (permitted) to access. For example, user bob may be authorized to use PPP for his connection, use IP address **192.168.200.4**, and use packet filter **std.ppp**.

- Accounting

  RADIUS accounting permits system administrators to track dial-in use. This information is often used for billing purposes. See Chapter 7, "RADIUS Accounting" for more information.

RADIUS version 2.0 provides the following enhancements:

- Menus

  When RADIUS menus are used, users are presented with a list of login options after they are authenticated. The RADIUS administrator may customize menus, including "chaining" one menu to other menus. See Chapter 5, "RADIUS Menus" for more details.

- SecurID

  SecurID authentication is based on Security Dynamics' token technology, which authenticates users using a patented time-synchronization method.
  The RADIUS 2.0 server can forward some or all authentication requests to a SecurID ACE/Server running on the same host as the RADIUS server.

  For more information, see Chapter 2, "RADIUS Server Configuration" and Chapter 6, "SecurID Installation."

- builddbm utility

  RADIUS 2.0 includes a utility named builddbm, which increases the speed of user look-up by converting the users file to the UNIX DBM format. Livingston recommends the use of the builddbm utility when the users file contains more than 500 users. See "RADIUS DBM Database" on page 4-19 for more details.

- Prefix/Suffix

  Prefixes and Suffixes allow a user to access multiple accounts by prepending or appending a string of characters defined by the administrator to the username.

- Session-Timeout

  The Session-Timeout reply item specifies the time limit for a session. Session-Timeout is specified as a particular number of seconds, up to a maximum of 31536000 (1 year).

- Idle-Timeout

  The Idle-Timeout reply item controls the maximum time that a session may be idle before it is disconnected. Idle-Timeout is specified as a number of seconds between 120 (2 minutes) and 14400 (4 hours).

- Port-Limit

  The Port-Limit reply item controls the maximum number of ports available for a Multilink PPP or Multilink V.120 connection. Port-Limit only applies to ISDN connections; other connection types are not affected by this setting.

- NAS-Port-Type

  The NAS-Port-Type check item restricts the type of port. The user may use one of the following port types: asynchronous, synchronous, ISDN, ISDN-V120, or ISDN-V110.

Getting Started

Select a UNIX host to use as the RADIUS server. Choose a host with the following characteristics:

- Located in a secure physical location
- Root access is limited to the Security Officer or System Administrator
- Offers a limited number of user accounts, preferably none
- Offers basic memory and disk space

Livingston recommends the use of a secondary RADIUS server. The primary RADIUS server is always queried first; if the server does not respond, it is queried a second time, then both the primary and secondary servers are queried up to eight times at three-second intervals until one responds or 30 seconds elapses without a response.

The RADIUS accounting server may be located on the same host as the RADIUS server used for authentication, or on a separate host. A secondary accounting server can be defined; the secondary server serves as a backup in the event that the primary server cannot be contacted.

Each PortMaster using RADIUS and its RADIUS server(s) share an authentication key of up to 15 alphanumeric characters called the **shared secret**. The shared secret must be configured on each RADIUS server and the PortMaster. It is stored as clear text in the **clients** file on the RADIUS server and in the nonvolatile memory of the PortMaster. Each PortMaster may share a different secret with the RADIUS servers, or multiple PortMasters may share the same secret.

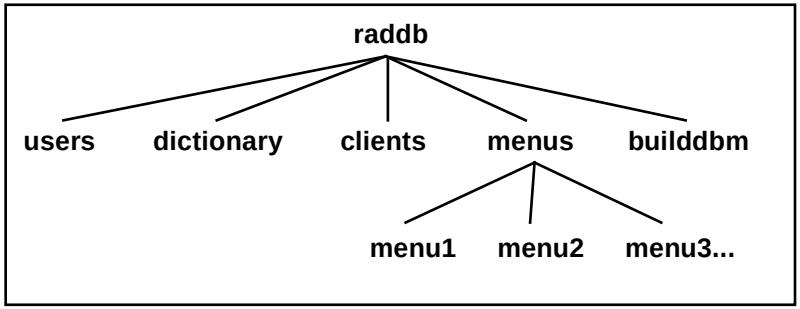
To configure the RADIUS server, continue to Chapter 2, "RADIUS Server Configuration." To configure a PortMaster to use RADIUS, see Chapter 3, "RADIUS Client Configuration." RADIUS accounting is described in Chapter 7.

RADIUS Server Configuration 2

Introduction

RADIUS server files are stored in the raddb (RADIUS database) directory, typically **/etc/raddb**. This directory contains files and subdirectories organized as shown in Figure 2-1.

Figure 2-1 RADIUS Directory Structure



The RADIUS server uses the UDP protocol, and listens for UDP packets on port 1645. When the **radiusd** daemon is executed to start the RADIUS server, the RADIUS accounting server is spawned as a child process; it listens for UDP packets on port 1646.

To configure RADIUS user information, see Chapter 4. To configure RADIUS accounting, see Chapter 7.

RADIUS Installation

RADIUS may be installed with or without the pminstall utility. pminstall is shipped on the PortMaster Software CD; it is designed to automate the installation of PortMaster software.

Installation with pminstall

To install RADIUS using pminstall, complete the following steps.

1. Log into the selected RADIUS server as root.
2. Mount the CD using the instructions inside the CD cover.
3. Install the PortMaster software by running **/cdrom/lei/unix/setup** (or by following the instructions inside the CD cover).
4. Enter the **pminstall** command at the UNIX prompt.

```
% /usr/portmaster/pminstall

1. PortMaster Internet Address Setup
2. Host Installation
3. PortMaster Upgrade
4. Host Upgrade
5. Install RADIUS
6. Exit

Please select an option from above:
```

5. Choose the **Install RADIUS** option to install all RADIUS files.
6. When prompted, enter the appropriate directories for each of the files. To place the files in the default directories, press the Return key as each option appears.

```
Database installation directory (/etc/raddb):
RADIUS accounting log directory (/usr/adm/radacct):
Directory to install radiusd in (/etc):
```

7. When RADIUS installation is complete, select the **Exit** option to quit pminstall.

8. Edit the clients and users files in **/etc/raddb**.
9. Enter the following command to start the RADIUS server:

```
/etc/radiusd
```

For a list of optional flags for the radiusd command, see Table 2-1 on page 2-5.

10. Continue to "Configuring Client Information" on page 2-6.

Note – radiusd is a stand-alone process; it cannot be run from /etc/inetd.conf.

Installation without pminstall

To install RADIUS without pminstall, complete the following steps.

1. If you are running NIS or NIS+, add the lines in Step 4 to the services NIS map on your NIS master and push the maps.

Note – Pushing the maps updates the database to include recently-entered information. Use the **make *mapname*** command on the NIS Master. For more details, consult your UNIX system documentation.

2. Log into the selected RADIUS server as root.
3. Mount the CD on **/cdrom** using the instructions inside the CD cover.
4. If you are not running NIS or NIS+, add the following lines to the **/etc/services** file:

```
radius      1645/udp      radiusd
radacct     1646/udp
```

5. As root, enter the following commands on the RADIUS server:

```
umask 022
mkdir /etc/raddb /usr/adm/radacct
chmod 700 /etc/raddb /usr/adm/radacct
```

The commands in this example (see page 2-3) create two directories, **raddb** and **radacct**. All RADIUS files (except the radiusd executable) are stored in the **/etc/raddb** directory. The radacct directory is used to store RADIUS accounting logs.

The **umask** and **chmod** commands affect the raddb and radacct directory permissions; root access is required for read, write, and execute privileges.

Caution – If you are upgrading from an existing installation of RADIUS 2.0, save the files in /etc/raddb before performing Step 6.

6. In RADIUS version 1.16, the **raddb** directory contains 3 files: **users**, **clients**, and **dictionary**. In RADIUS version 2.0, the raddb directory contains an additional directory named **menus**. Copy all files in /cdrom/lei/unix/radius/raddb to the /etc/raddb directory.

```
cp -r /cdrom/lei/unix/radius/raddb/* /etc/raddb
```

7. Copy the radiusd file to the **/etc** directory (or if you prefer, to another directory such as **/usr/sbin**). Copy the builddbm utility to **/etc/raddb/builddbm**. Replace *platform* with the name of your operating system, for example, **sun4_4.1**.

```
cp /cdrom/lei/unix/platform/radiusd /etc/radiusd
cp /cdrom/lei/unix/platform/builddbm /etc/raddb/builddbm
```

8. Use the **radiusd** command to start RADIUS. radiusd spawns the RADIUS accounting server as a child process. For more information about RADIUS accounting, see Chapter 7.

```
/etc/radiusd
```

Note – radiusd is a stand-alone process; it cannot be run from /etc/inetd.conf.

9. Continue to "Configuring Client Information" on page 2-6.

radiusd may be used with any of the following flags:

Table 2-1 radiusd Flags

Flag	Purpose
-a	Specifies an alternate directory for RADIUS accounting. The default directory is **/usr/adm/radacct**.
-b	Uses the DBM version of the users file. See "RADIUS DBM Database" on page 4-19 for more information.
-d	Specifies an alternate directory for RADIUS configuration files. The default directory is **/etc/raddb**.
-l	Specifies a RADIUS logfile to use instead of syslog.
-s	Runs RADIUS in single-threaded mode without spawning a child process to handle each authentication request.
-v	Displays the version of RADIUS without starting the radiusd daemon.
-x	Debug mode. To send debug output to syslog, use **-x -l syslog**.

10. To start the radiusd daemon each time the RADIUS server is booted, place radiusd in the **/etc/rc.local** file as shown in the example below. On some systems this may be **/etc/rc2.d/S99radiusd** or another file; consult your UNIX system documentation for more information.

```
#
# Start RADIUS
#
if [ -f /etc/radiusd ]; then
    echo “RADIUS”
    /etc/radiusd
fi
```

Note – radiusd does not need to be restarted each time the clients or users files are modified. This daemon only needs to be restarted when the dictionary file is modified.

11. Continue to "Configuring Client Information" on page 2-6.

Configuring Client Information

The **/etc/raddb/clients** file stores information about RADIUS clients, including each client's name or IP address and its shared secret.

clients is a flat text file; to add a client, enter the client's name or IP address and the shared secret. To add a comment, preface the desired line with the **#** sign.

Shared secrets must consist of 15 or fewer alphanumeric characters. There is no limit to the number of clients that may be added to this file.

Examples of client names and shared secrets are displayed below.

```
#Client Name        Key
#-------------------------------------
portmaster1         wP40cQ0
portmaster2         A3X445A
192.168.1.2         wer369st
```

As the clients file contains the shared secrets for the RADIUS clients, only root should have read and write access to the file.

```
-rw-------  1 root daemon 802 Jul 15 00:21 clients
```

Continue to Chapter 3 to configure the PortMaster as a RADIUS client.

RADIUS Client Configuration 3

This chapter covers configuration of the PortMaster as a RADIUS client. The following items must be configured on each PortMaster:

- IP address of the primary and optional alternate RADIUS servers
- IP address of the primary and optional alternate RADIUS accounting server, if accounting is to be performed
- RADIUS shared secret

There are two steps to configure a RADIUS client: adding the PortMaster and shared secret to the **clients** file on the RADIUS server (see page 2-6), and configuring the shared secret and address of the RADIUS server on the PortMaster.

RADIUS clients may be configured using the PortMaster command line interface (see the following section) or using PMconsole (see page 3-3).

Configuration Using Command Line Interface

To configure the PortMaster using the command line interface, complete the following steps.

1. Enable port security on all ports using the **set all security on** command. When port security is enabled, each user attempting to log into the port must be authenticated using the PortMaster User Table or RADIUS.

```
Command> set all security on
```

2. Enter the IP address of the primary RADIUS server using the following command:

```
Command> set authentic  192.168.200.23
```

3. Optionally, specify an alternate RADIUS server using the following command:

```
Command> set alternate 192.168.200.24
```

The primary RADIUS server is consulted first. If the server does not respond, it is queried a second time, then both servers are queried up to eight additional times at three-second intervals.

4. To log activity using RADIUS accounting, enter the IP address of the primary accounting server:

```
Command> set accounting 192.168.200.4
```

Optionally, specify an alternate accounting server:

```
Command> set accounting 2 192.168.200.5
```

5. Enter the secret shared by the PortMaster and RADIUS server using the **set secret** command. This is the same shared secret entered in the clients file on the RADIUS server (see page 2-6).

```
Command> set secret 3jk3l5d44vdpw89
```

The shared secret is a string of up to 15 alphanumeric printable ASCII characters. If a secret longer than 15 characters is specified, an error message is displayed.

6. Save your changes using the **save all** command, then reset all ports.

```
Command> save all
Command> reset all
```

Caution - Resetting all ports disconnects any user sessions in progress.

Configuration Using PMconsole

To configure the PortMaster using PMconsole, complete the following steps:

1. Choose **RADIUS** from the Edit menu.
2. In the dialog box that appears, enter the IP address of the primary and optional alternate RADIUS servers.
3. To log activity using RADIUS accounting, enter the IP address of the primary and optional alternate accounting servers.
4. Enter the secret shared by the RADIUS client and RADIUS server. For security reasons, the secret is not displayed in the dialog box.

   The shared secret is case-sensitive, and must consist of 15 characters or less. Control characters may not be used.

Note – Do not press the Return key when the cursor is in the RADIUS Secret field of the dialog box. Pressing the Return key at this point will erase the secret when the Save button is pressed.

5. To save the RADIUS settings, click the **Save** button.
6. To leave the window, click the **Done** button.
7. On each port, turn Security **on**, then click the **Save** button to save the port setting to the PortMaster's non-volatile memory. Click the **Remote Reset** button, then click the **Done** button to close the dialog box.

   When port security is enabled, each user attempting to log into the port must be authenticated using the PortMaster User Table or RADIUS.

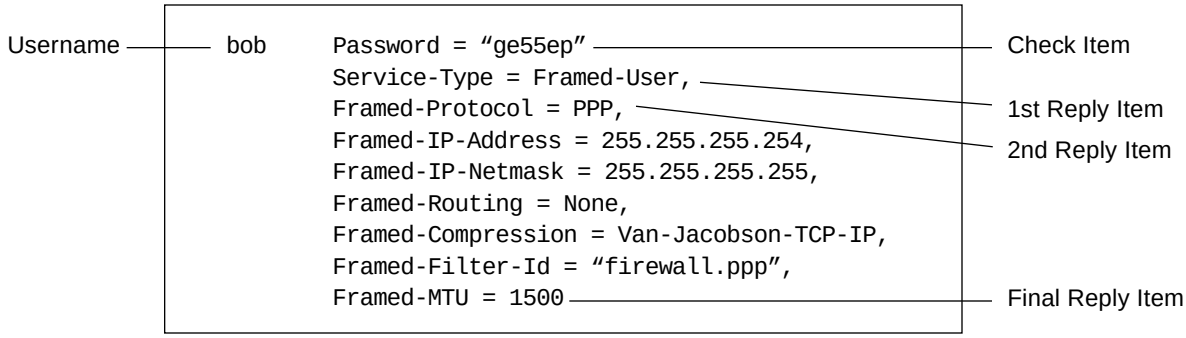
Note – Some older versions of PMconsole may display the **Pass-Thru Login** option instead of the Security option in this dialog box. In this case, ensure that Pass-Thru Login is **disabled**; this has the same effect as turning Security on.

Configuring User Information 4

Introduction

The RADIUS users file on the RADIUS server stores authentication and authorization information for all users authenticated with RADIUS. Each user has an **entry** which consists of three parts: the **username**, a list of **check items**, and a list of **reply items**. Figure 4-1 displays an example.

Figure 4-1 User Entry



- Username

  The Username is the first part of each user entry. Usernames consist of up to 63 non-whitespace printable ASCII characters. If SecurID or a System password file is used for authentication, the username must conform to the UNIX username limitation, typically 8 characters or less.

- Check items

  Check items are listed on the first line of a user entry, separated by commas. For an access-request (see "How RADIUS Works" on page 1-2) to succeed, all check items in the user entry must be matched in the access-request.

  In Figure 4-1, bob’s password is the only check item. To successfully authenticate bob, the RADIUS server must receive this password in bob’s access-request.

- Reply items

  Reply items give the PortMaster information about the user's connection, for example, whether PPP or SLIP is used or whether the user's IP address is negotiated. In Figure 4-1, Framed-Protocol is a reply item. The value of Framed-Protocol is PPP, indicating that bob uses PPP for his connection.

  If all check items in the user entry are satisfied by the access-request, the RADIUS server sends the reply items to the PortMaster to configure the connection.

Note – Several common user entries are listed in "Examples" on page 4-25. All check items and reply items are listed in Table 4-4 on page 4-20.

Username

Each user entry must have a username. As stated in the previous section, a username must consist of up to 63 non-whitespace printable ASCII characters.

Check Items

Check items may consist of any of the following: password information, client information, prefixes, or suffixes.

Passwords

Two different password characteristics may be specified in a user entry; the password's location, and its expiration date.

Password Locations

The **Auth-Type** check item is used to specify the type of authentication to use for a particular user. Auth-Type may be set to one of the following: Local, System, or SecurID. If this check item is omitted from the user entry, Local is assumed.

- Local

  To indicate that a user's password is stored in the RADIUS users file, use the **Local** Auth-Type. To set the user's password, use the Password check item. An example is displayed below.

```
bob     Auth-Type = Local, Password = "ge55ep"
```

Note – When a user's password is stored locally, the Auth-Type check item may be omitted; only the Password check item is required.

- System

  To indicate that a user's password is stored in a system password file, use the **System** Auth-Type. System can be a password file in UNIX such as /etc/passwd, /etc/shadow, or a password map in NIS or NIS+. When the RADIUS server receives a username/password pair from the client, it queries the operating system to determine if there is a matching username/password pair.

```
bob     Auth-Type = System
```

  The System Auth-Type is equivalent to the RADIUS 1.16 **Password = "UNIX"**, which is also permitted in RADIUS 2.0 for backwards compatibility.

```
bob     Password = "UNIX"
```

- SecurID

  The SecurID Auth-Type indicates that the user's password should be authenticated by a SecurID ACE/Server.

```
bob     Auth-Type = SecurID
```

  To receive a passcode from SecurID, the ACE/Server software must be running on the same UNIX host as the RADIUS server. In this case, the RADIUS server serves as an ACE/Server Master. If the ACE/Server Master is installed on a different host, the RADIUS server must be configured as an ACE/Server Slave. See Chapter 5 for instructions.

Password Expiration Date

To disable logins after a particular date, complete the following steps.

1. Specify the date of expiration using the Expiration check item. The date must be specified in ***"Mmm dd yyyy"*** format; an example is shown below.

```
bob     Password = "ge55gep", Expiration = "Dec 04 1996"
```

2. Edit the Password-Expiration and Password-Warning values in /etc/raddb/dictionary to meet your security needs.

```
VALUE     Server-Config     Password-Expiration     30
VALUE     Server-Config     Password-Warning        5
```

The first parameter, **Password-Expiration**, updates the Expiration date in the users file when a user changes his password. In this example, Password-Expiration is set to 30. If user bob changes his password on January 1, 1997, his Expiration date in the users file changes to **Jan 31, 1997**.

Password-Warning controls when users are notified that their accounts are about to expire. In the example above, users receive warning messages 5 days before their password expiration date.

Note – A mechanism to permit users to change their passwords is outside the scope of RADIUS. RADIUS 2.0 supports **radpass** for backwards compatibility with sites supporting that feature, however, use of radpass is not recommended and support will be removed in a future release of RADIUS.

3. If you modified the dictionary file, kill and restart the radiusd daemon.

Client Information

The **NAS-IP-Address** check item specifies the IP address of a particular PortMaster. When this setting is used as a check item in a user entry, the user must be attempting to start a connection on the specified PortMaster in order for the connection to succeed.

The **NAS-Port** check item may be used to specify a particular PortMaster port. To be successfully authenticated, the user must be attempting to log into this port.

The **NAS-Port-Type** check item may be used to specify the type of port. Options for the NAS-Port-Type are as follows: Async, Sync, ISDN, ISDN-V120, or ISDN-V110. The PortMaster must run ComOS release 3.3.1 or later to support NAS-Port-Type.

The following example displays a user entry containing the NAS-IP-Address and NAS-Port-Type settings.

```
bob     Password = "ge55gep", NAS-IP-Address = 192.168.1.54, NAS-Port-Type = ISDN
        Service-Type = Framed-User,
        Framed-Protocol = PPP
```

Prefixes and Suffixes

The **Prefix** and **Suffix** check items allow a user to access multiple services by prepending or appending a series of characters to his username.

Prefixes and Suffixes are most useful when defined in the DEFAULT user entry (see the example below), however, they may also be used with individual user entries. Prefix and Suffix strings must consist of 16 or fewer alphanumeric printable ASCII characters.

```
Pbob    Auth-Type = System, Prefix = "P"
        Framed-Protocol = PPP,
```

In the above example, bob's username and password are stored in a system password file. In order for bob to use this particular account, he must specify a username of **Pbob** when attempting to connect to the PortMaster.

The RADIUS server strips any prefixes and suffixes and looks up the username. Using the previous example, the RADIUS server would strip the **P** and check the system password for bob.

```
DEFAULT     Auth-Type = System, Suffix = "%slip"
            Framed-Protocol = SLIP,
```

If bob specified a username of **bob%slip**, the RADIUS server would configure bob's connection using the settings in the DEFAULT entry.

Reply Items

Service Type

The type of service provided to the user, called the **Service-Type**, must be specified in each user entry. Service-Type must be set to one of the values shown in Table 4-1.

Table 4-1 Service-Type

Service-Type	Explanation
Login-User	User connects via Telnet, Rlogin, pmd, or TCP-Clear.
Framed-User	User uses PPP or SLIP for the connection.
Outbound-User	User uses Telnet for outbound connections.
Callback-Login-User	The PortMaster verifies the user's identity by disconnecting the port and dialing the user back at a specified number. The user's identity must be verified before the connection is permitted.
Callback-Framed-User	The Portmaster verifies the user's identity by disconnecting the port and dialing the user back using a specified Location Table entry. When the user's identity is verified, PPP or SLIP is used for the connection.

Note – To configure the callback number or location, see "Callback Information" on page 4-10.

In the following example, user bob's Service-Type is **Framed-User**.

```
bob     Auth-Type = System
        Service-Type = Framed-User
```

Framed Protocol

When the Service-Type is Framed-User, the **Framed-Protocol** reply item should be included in the user entry to indicate whether PPP or SLIP is used. For example, user bob is a PPP user. His user entry includes the following lines:

```
bob     Auth-Type = System
        Service-Type = Framed-User,
        Framed-Protocol = PPP
```

Framed-Protocol can also be used as a reply item requiring PPP autodetection by the PortMaster.

```
bob     Auth-Type = System, Framed-Protocol = PPP
        Service-Type = Framed-User,
        Framed-Protocol = PPP
```

To authenticate a user using PAP, set the Auth-Type to any of the following: **Local**, **System**, or **SecurID**. To authenticate a user using CHAP, the Auth-Type must be **Local** and PAP must be turned off using the following command on the PortMaster:

```
set pap off
```

Framed IP Address

The **Framed-IP-Address** reply item is used to specify the user's IP address.

When Framed-IP-Address is set to 255.255.255.255, the PortMaster negotiates the address with the end-node (dial-in user). When it is set to 255.255.255.254 (or omitted), the PortMaster assigns an IP address to the dial-in user from the Assigned Address Pool.

Note – To create an Assigned Address Pool for the PortMaster, see the *Configuration Guide for PortMaster Products.*

Framed IP Netmask

A netmask may be specified for a user using the **Framed-IP-Netmask** reply item. If this reply item is omitted, the default subnet mask of 255.255.255.255 is used.

Framed Route

The **Framed-Route** reply item adds a route to the PortMaster's routing table when service to the user begins. Three pieces of information are required; the destination IP address, gateway IP address, and metric. An example is shown below.

```
bob     Auth-Type = System
        Service-Type = Framed-User,
        Framed-Protocol = PPP,
        Framed-IP-Address = 150.128.1.1
        Framed-Route = "150.128.1.0 150.128.1.1 1"
```

In this example, 150.128.1.0 is the IP address of a destination network. 150.128.1.1 is the IP address of the gateway for this network, and 1 is the metric (hop count).

If 0.0.0.0 is specified as the gateway IP address, the user's IP address is substituted for the gateway.

Outbound-User

The **Outbound-User** setting allows a user to gain outbound access to network device ports using Telnet. This feature is supported in ComOS version 3.3.2 or later and RADIUS 2.0. In order to use this feature, the port must be set to **device /dev/network** or **twoway /dev/network**.

To restrict users to outbound access, **Service-Type = Outbound-User** must be a check item in the user entry. The Login-TCP-Port setting may be used to specify the TCP port for the connection; the port must be between 10000 and 10100. An example is displayed below.

```
bob     Password = “ge55gep”, Service-Type = Outbound-User
        Service-Type = Outbound-User,
        Login-Service = Telnet,
        Login-TCP-Port = 10000
```

Using the above example, when user bob is attempting an outbound connection, the PortMaster client checks its local User Table for an entry for bob. If bob is not found in the table, the PortMaster sends an access-request to the RADIUS server indicating that bob is an Outbound-User.

The RADIUS server examines bob’s entry in the users file; if Outbound-User is included as a reply item, the PortMaster is notified to permit the connection.

The PortMaster should be configured as shown in the example below. This example configures port **s1**, however, multiple ports may be configured to listen at different TCP port numbers or at the same TCP port number to create a pool of devices.

```
set s1 device /dev/network
set s1 service_device telnet 10000
set s1 modem off
```

Callback Information

In order for a user to be authenticated using callback, a phone number or location must be specified in the user's entry.

Callback-Login-User

When a user's Service-Type is **Callback-Login-User**, a phone number must be specified using the **Callback-Number** reply item.

```
bob     Password = "ge55gep"
        Service-Type = Callback-Login-User,
        Callback-Number = "9,1-800-555-1212"
```

After the RADIUS verifies the password for user bob, it sends an access-accept message including the Callback-Number to the PortMaster. The PortMaster calls the user back at the specified number; if the user is reached successfully, the PortMaster asks the user to re-enter his password and then sets up the connection.

Callback-Framed-User

When a user's service type is **Callback-Framed-User**, a location must be specified using the **Callback-Id** setting. An example is displayed below.

```
bob     Password = "ge55gep"
        Service-Type = Callback-Framed-User,
        Callback-Id = "bobhome"
```

After the RADIUS server verifies the password for user bob, it sends an access-accept message including the Callback-Id to the PortMaster. The PortMaster checks its local Location Table; if there is a matching location name, it makes the connection using that location's settings.

Note – Creating Location Table entries is covered in "Configuring Dial-Out Locations" in the *Configuration Guide for PortMaster Products.*

Routing

The **Framed-Routing** reply item controls how RIP is used on the user's interface. RIP options include:

Table 4-2 Framed-Routing Options

Option	Explanation
None	Disables RIP on the interface.
Broadcast	The interface sends RIP updates.
Listen	The interface listens for RIP updates.
Broadcast-Listen	The interface sends and listens for RIP updates.

The following example displays user bob's user entry. Framed-Routing is set to **None**; bob's interface neither sends nor listens for RIP updates.

```
bob     Password = "ge55gep"
        Service-Type = Framed-User,
        Framed-Protocol = PPP,
        Framed-Routing = None,
```

Typically, Framed-Routing is set to **Broadcast-Listen** for connections to other routers, and set to **None** for user connections.

Packet Filters

Each PPP or SLIP user authenticated with RADIUS may be associated with packet filters using the **Filter-Id** reply item. In the following example, the **firewall** filter is used during bob's connection.

```
bob     Password = "ge55gep"
        Service-Type = Framed-User,
        Framed-Protocol = PPP,
        Filter-Id = "firewall"
```

Filters must be defined on each PortMaster the user accesses. To control whether the filter restricts incoming or outgoing traffic, the filter defined on the PortMaster must have an **.in** or **.out** suffix attached to its name. In the above example, the filter **firewall.in** is used as a filter for packets entering the PortMaster via the interface, and **firewall.out** is used as an output filter for packets leaving the PortMaster via the interface.

The .in and .out suffixes do not need to be specified in the user entry. When a user dials in to the PortMaster, the .in or .out suffix is automatically appended to the filter name provided by RADIUS.

Note – To configure filters on a PortMaster, see the "Configuring Filters" chapter of the *Configuration Guide for PortMaster Products*.

Access Filters

Each host prompt login user authenticated with RADIUS may be associated with an access filter using the Filter-Id reply item. In the following example, the **gnric** filter is used to restrict the hosts that bob may access during a connection:

```
bob     Password = "ge55gep"
        Service-Type = Login-User,
        Login-IP-Host = 255.255.255.255,
        Login-Service = Telnet,
        Login-TCP-Port = 23,
        Filter-Id = "gnric"
```

Access filters must be defined on each PortMaster the user accesses, using the same name as the Filter-Id. The access filter name defined in the user record must be exactly the same as the filter name defined on the PortMaster. The PortMaster does not append anything to the name of an access filter, unlike packet filters.

Remote Host Information

When a user's Service-Type is Login-User or Callback-Login-User, two pieces of information may be supplied: the service used to connect to the host, and the name or IP address of the remote host. A TCP port number may optionally be supplied.

To specify the login service, use the **Login-Service** reply item. All Login-Service values are described in Table 4-3.

Table 4-3 Login-Service

Login-Service	**Description**
Telnet	Establishes a Telnet connection to the remote host.
Rlogin	Establishes an Rlogin connection to the remote host.
TCP-Clear	Establishes a TCP clear connection to the remote host. 8-bit data is passed through this connection without interpretation. This option is the equivalent of the **netdata** login service on the PortMaster.
PortMaster	Establishes a connection to the remote host using the PortMaster login service. To use this setting, the **in.pmd** daemon must be installed on the remote host.

The name or IP address of the remote host is specified using the **Login-IP-Host** reply item. If the user is to log into a particular TCP port on the remote host, the port number may be specified using the **Login-TCP-Port** reply item.

An example is displayed below. In this entry, user bob is authenticated, then called back at the Callback-Number. If successfully authenticated, a Telnet connection to port 23 on host 192.168.1.76 is established.

```
bob     Password = "ge55gep"
        Service-Type = Callback-Login-User,
        Login-IP-Host = 192.168.1.76,
        Login-Service = Telnet,
        Login-TCP-Port = 23,
        Callback-Number = "9,1-800-555-1234"
```

If Login-IP-Host is set to 0.0.0.0 or omitted, the host defined for the port is used. If Login-IP-Host is set to 255.255.255.255, the user is presented with a **Host:** prompt; the user enters the hostname or the host's IP address at this prompt.

MTU

The **Framed-MTU** reply item configures the Maximum Transmission Unit for a user's connection.

```
Framed-MTU = 1500
```

Framed-MTU is only used for PPP and SLIP connections. For PPP connections, the Framed-MTU may be between 100 and 1520 bytes. SLIP connections may have an MTU between 100 and 1006 bytes. On IPX networks, Framed-MTU should be set to at least 600 bytes.

Note – If PPP negotiates an MTU for the connection, the Framed-MTU setting is ignored.

Compression

Van Jacobson TCP/IP Header Compression is enabled by default. To disable compression, set the Framed-Compression setting to **None**.

```
Framed-Compression = None
```

IPX Network

When an IPX network is used for a particular user's connection, the **Framed-IPX-Network** reply item must appear in the user entry. The PortMaster supports IPX over PPP.

Framed-IPX-Network must be specified in dotted quad format (xx.xx.xx.xx). For example, the hexadecimal network number 123456 must be expressed as 0.18.52.86.

```
bob     Password = "testing"
        Service-Type = Framed-User,
        Framed-Protocol = PPP
        Framed-IPX-Network = 0.18.52.86
```

To convert an IPX hexadecimal network number to dotted quad format, use the following PERL script:

```
#!/usr/local/bin/perl
# hex   - convert ip addresses to hexadecimal and vice versa
for (@ARGV) {
        if (/\./) {             # convert . to hex
                @octets = split(/\./,$_);
                for $octet (@octets) {
                        printf "%02X",$octet;
                }
                print "\n";
        } else {                # convert hex to .
                $buf = '';
                while (s/\w\w//) {
                        $buf .= hex($&).'.';
                }
                $buf =~ s/\.$/\n/;
                print $buf;
        }
}
```

RADIUS 2.0 Reply Items

This section describes the reply items introduced in RADIUS 2.0. To use RADIUS 2.0, all PortMaster clients must be using ComOS version 3.3.1 or later. ComOS version 3.3.3 or later is recommended.

Session-Timeout

Session-Timeout specifies the time limit for a session. When this reply item appears in a user entry, the user is disconnected when the time limit is reached.
Session-Timeout is specified as a particular number of seconds, up to a maximum of 31536000 (1 year).

```
bob     Password = "ge55gep"
        Service-Type = Framed-User,
        Framed-Protocol = PPP,
        Session-Timeout = 7200
```

In the above example, user bob is automatically disconnected after 7200 seconds (2 hours).

Idle-Timeout

The **Idle-Timeout** specifies the time a session may be idle before it is disconnected. Idle-Timeout is specified as a number of seconds between 120 (2 minutes) and 14400 (4 hours), and is rounded down to a multiple of 60.

```
bob     Password = "ge55gep"
        Service-Type = Framed-User,
        Framed-Protocol = PPP,
        Idle-Timeout = 600
```

In the above example, if the session is inactive longer than 600 seconds (10 minutes), user bob is disconnected.

Note – Idle-Timeout and Session-Timeout values are specified in seconds in the RADIUS users file. If these timeout values are set using the PortMaster command line interface or PMconsole, they are specified in minutes.

Port-Limit

The **Port-Limit** reply item controls the maximum number of ports available for a Multilink PPP or Multilink V.120 connection. Port-Limit only applies to ISDN connections; other connection types are not affected.

The Port-Limit value may be as high as the maximum number of B channels available for the ISDN ports. For example, if a PortMaster has 15 ISDN BRI ports, the Port-Limit value may be as high as 30.

```
bob     Password = "ge55gep", NAS-Port-Type = ISDN
        Service-Type = Framed-User,
        Framed-Protocol = PPP,
        Port-Limit = 1
```

In the above example, user bob's connection may use only one B channel.

Default User Entries

When the RADIUS server receives a username/password pair from a PortMaster, the RADIUS server scans the users file for a match, starting from the top of the file. If a match is located, the user is authenticated using the information in that user entry. If a matching user entry is not found during the scan, but a matching DEFAULT entry is located, that entry is used.

The DEFAULT entry is typically used with Auth-Type System or SecurID. These entries should appear at the end of the users file; the RADIUS server stops scanning entries when a matching DEFAULT entry is found.

```
DEFAULT     Auth-Type = System
            Service-Type = Framed-User,
            Framed-Protocol = PPP,
            Framed-IP-Address = 255.255.255.254,
            Framed-Routing = None,
            Filter-Id = "firewall",
            Framed-MTU = 1500
```

For example, user bob's password is stored in a UNIX password file. When he attempts to connect to the network, the RADIUS server scans the users file to determine if there is a matching user entry. If a matching entry is not found before the DEFAULT entry is found, the DEFAULT entry is used. Since the DEFAULT entry includes **Framed-Protocol = PPP** as a reply item, PPP is used for bob's connection.

Note – In RADIUS version 1.16, only one DEFAULT entry was permitted.

RADIUS 2.0 permits multiple DEFAULT user entries. To distinguish between DEFAULT entries, the **Prefix** and **Suffix** settings are used. When users prepend or append the Prefix or Suffix to their username, the RADIUS server matches them to the corresponding DEFAULT entry.

```
DEFAULT     Auth-Type = System, Prefix = "P"
            Service-Type = Framed-User,
            Framed-Protocol = PPP,
            Framed-IP-Address = 255.255.255.254,
            Framed-Routing = None,
            Framed-MTU = 1500

DEFAULT     Auth-Type = System, Suffix = "%C"
            Service-Type = Framed-User,
            Framed-Protocol = SLIP,
            Framed-IP-Address = 255.255.255.254,
            Framed-MTU = 1006

DEFAULT     Auth-Type = System, Prefix = "S"
            Service-Type = Framed-User,
            Framed-Protocol = SLIP,
            Framed-IP-Address = 255.255.255.254,
            Framed-Compression = None,
            Framed-MTU = 1006
```

In the above example, assume that user bob's password is stored in a UNIX password file and that there is not a matching entry in the RADIUS users file. If bob uses **Pbob** as his username, the first DEFAULT entry is used, and bob is authenticated as a PPP user. If bob logs in as **bob%C**, the second DEFAULT entry is used and he is authenticated as a CSLIP user.

DEFAULT entries may be named simply **DEFAULT**, or they may have a number appended to the end of the entry name, for example, **DEFAULT1**, **DEFAULT2**, etc. An example is shown below.

```
DEFAULT1   Auth-Type = System, Prefix = "P"
           Service-Type = Framed-User,
           Framed-Protocol = PPP,
           Framed-IP-Address = 255.255.255.254,
           Framed-Routing = None,
           Framed-MTU = 1500

DEFAULT2   Auth-Type = System, Suffix = "%C"
           Service-Type = Framed-User,
           Framed-Protocol = SLIP,
           Framed-IP-Address = 255.255.255.254,
           Framed-MTU = 1006

DEFAULT3   Auth-Type = System, Prefix = "S"
           Service-Type = Framed-User,
           Framed-Protocol = SLIP,
           Framed-IP-Address = 255.255.255.254,
           Framed-Compression = None,
           Framed-MTU = 1006
```

RADIUS DBM Database

The **builddbm** utility included with RADIUS converts the users text file to the UNIX DBM format, which increases the speed of user lookups. Livingston recommends the use of the builddbm utility when the users file contains more than 500 users.

To run builddbm, use the following commands:

```
cd /etc/raddb
./builddbm
```

To run the radiusd daemon after the users file is converted to DBM, execute radiusd with the **-b** option.

```
/etc/radiusd -b
```

builddbm generates the **users.dir** and **users.pag** files, which are used by the radiusd daemon. On some versions of UNIX a **users.db** file is created instead.

Note – After the users file has been converted to the DBM format, builddbm must be run again if any changes are made to the user entries.

User Entry Check and Reply Items: Complete Listing

Table 4-4 lists all user entry check and reply items.

Table 4-4 User Entry Check and Reply Items

Item	Options	Explanation	May be Used as Check item?	May be Used as Reply item?
User-Name	User's name. May be up to 63 characters.		N/A	No
Password	User's password		Yes	No
Auth-Type	Local	User's password is stored in the RADIUS users file. Default.	Yes	No
	System	User's password is stored in a system password file.	Yes	No
	SecurID	User is authenticated via SecurID.	Yes	No

Item	Options	Explanation	May be Used as Check item?	May be Used as Reply item?
Expiration	Must be specified in "*Mmm dd yyyy*" format	Date that user's password expires.	Yes	No
Prefix	String of characters in double quotes	Prepended to username to match a user to a particular user entry. Used primarily for DEFAULT entries.	Yes	No
Suffix	String of characters in double quotes	Appended to username to match a user to a particular user entry. Used primarily for DEFAULT entries.	Yes	No
NAS-IP-Address	IP address	PortMaster's IP address.	Yes	No
NAS-Port	Number	The PortMaster port number that the user is dialed into (for example, 2 = S2)	Yes	No
NAS-Port-Type	ISDN	ISDN Port	Yes	No
	Async	Asynchronous Port	Yes	No
	Sync	Synchronous Port	Yes	No
	ISDN-V120	ISDN in V.120 mode	Yes	No
	ISDN-V110	ISDN in V.110 mode	Yes	No
Service-Type	Login-User	User connects via Telnet, Rlogin, PortMaster, or TCP-Clear login service.	No	Yes
	Framed-User	User uses PPP or SLIP for the connection.	Yes	Yes

Item	Options	Explanation	May be Used as Check item?	May be Used as Reply item?
Service-Type	Outbound-User	User uses Telnet for outbound connections.	Yes	Yes
	Callback-Login-User	Calls user back and connects via Telnet, Rlogin, PortMaster, or TCP-Clear login service.	No	Yes
	Callback-Framed-User	Calls user back and establishes a Framed connection (PPP or SLIP).	No	Yes
Login-Service	Telnet	Establishes a Telnet connection to the remote host.	No	Yes
	Rlogin	Establishes an Rlogin connection to the remote host.	No	Yes
	TCP-Clear	Establishes a TCP clear connection to the remote host.	No	Yes
	PortMaster	Establishes a connection to the remote host using the PortMaster login service.	No	Yes
Login-IP-Host	IP address	Address of the remote host.	No	Yes
Login-TCP-Port	TCP port number	TCP port number of the Login-Service	No	Yes
Framed-Protocol	PPP	PPP is used for the connection.	Yes	Yes
	SLIP	SLIP is used for the connection.	No	Yes

Item	Options	Explanation	May be Used as Check item?	May be Used as Reply item?
Framed-IP-Address	IP Address	The user's IP address.	No	Yes
Framed-IP-Netmask	Netmask	The user's netmask.	No	Yes
Framed-Routing	None	Disables RIP on the interface.	No	Yes
	Broadcast	The interface sends RIP updates.	No	Yes
	Listen	The interface listens to RIP updates.	No	Yes
	Broadcast-Listen	The interface sends and listens to RIP updates.	No	Yes
Filter-Id	Filter name	Filter name to be used for packet or access filtering on the interface.	No	Yes
Framed-MTU	Number	Number of bytes in Maximum Transmission Unit	No	Yes
Framed-Compression	None	If this reply item is omitted, Van Jacobson TCP/IP header compression is used.	No	Yes
	Van-Jacobson-TCP-IP	Van Jacobson TCP/IP header compression is used for the connection. Default.	No	Yes
Reply-Message	Text message in double quotes (" ")	Display a message to user after authentication	No	Yes

Item	Options	Explanation	May be Used as Check item?	May be Used as Reply item?
Callback-Number	Phone number in double quotes (“ “)	Specify only for Service-Type = Callback-Login-User	No	Yes
Callback-Id	Location name in double quotes (“ “)	Specify only for Service-Type = Callback-Framed-User	No	Yes
Framed-IPX-Network	Dotted quad IPX network number	IPX network number	No	Yes
Port-Limit	Number of B channels for ISDN MP or multilink V.120	Specify the number of B channels a user might have	No	Yes
Session-Timeout	In seconds	Specify the time limit for a session	No	Yes
Idle-Timeout	In seconds	Specify the idle time limit for a session	No	Yes
Menu	Menu name in double quotes (“ “)	Define a menu in a user record	No	Yes
Termination-Menu	Menu name in double quotes (“ “)	Menu to display after service is terminated. This item can only be set in a menu.	No	Yes

Examples

User entries may be configured in a number of ways to fit network security requirements. The following examples illustrate a series of typical RADIUS user entries.

PPP User Entry

This example illustrates a typical RADIUS entry for a PPP user.

```
bob     Password = "ge55gep"
        Service-Type = Framed-User,
        Framed-Protocol = PPP,
        Framed-IP-Address = 255.255.255.254,
        Framed-Routing = None,
        Framed-Compression = Van-Jacobson-TCP-IP,
        Framed-MTU = 1500,
        Filter-Id = "firewall"
```

In this example, user bob has password **ge55gep**. He is a Framed-User, which indicates that he uses SLIP or PPP for his connections. The following line, **Framed-Protocol**, specifies PPP.

An IP address of 255.255.255.254 is specified, indicating that an IP address is assigned to bob from the PortMaster Assigned Address Pool.

Note – To create an Assigned Address Pool, see the *Configuration Guide for PortMaster Products*.

Framed-Routing is set to **None**, which disables RIP for bob's interface. RIP packets are not sent or listened for. Van Jacobson TCP/IP compression is used for the connection, and the MTU (Maximum Transmission Unit) is set to 1500.

The Filter-Id identifies the packet filter used for the connection; if they exist on the PortMaster, **firewall.in** is used as an input filter and **firewall.out** is used as an output filter.

Using Prefixes

Creating multiple DEFAULT entries can eliminate the time required to create multiple accounts for users. Users prepend or append the prefix or suffix to their username when they attempt to log into the PortMaster; the RADIUS server uses these prefixes and suffixes to match the user to the corresponding DEFAULT entry.

In the following example, the user file contains four DEFAULT entries; one entry for PPP, SLIP, CSLIP, and Telnet users.

```
DEFAULT1   Auth-Type = System, Prefix = "P"
           Service-Type = Framed-User,
           Framed-Protocol = PPP,
           Framed-IP-Address = 255.255.255.254,
           Framed-Routing = None,
           Filter-Id = "firewall",
           Framed-MTU = 1500

DEFAULT2   Auth-Type = System, Prefix = "S"
           Service-Type = Framed-User,
           Framed-Protocol = SLIP,
           Framed-IP-Address = 255.255.255.254,
           Framed-Compression = None

DEFAULT3   Auth-Type = System, Prefix = "C"
           Service-Type = Framed-User,
           Framed-Protocol = SLIP,
           Framed-IP-Address = 255.255.255.254,
           Framed-Compression = Van-Jacobson-TCP-IP

DEFAULT4   Auth-Type = System
           Service-Type = Login-User,
           Login-IP-Host = 172.16.1.4,
           Login-Service = Telnet
```

If user bob enters **Pbob** as his username, he is authenticated as a PPP user. If he enters **bob%T** as a username, he is authenticated as a Telnet user. If he enters **Sbob** as a username, he is authenticated as a SLIP user.

RADIUS Menus 5

Introduction

RADIUS menus allow a user to select different login options after being authenticated. For a user with several different account types, menus allow the user to select different options without re-connecting.

How Menus Work

RADIUS menus are implemented as text files located in the **/etc/raddb/menus** directory on the RADIUS server. The number of menu files under the menus directory is unlimited. Menu files contain the **menu** and **end** keywords to indicate the start and end of the text displayed to the user. Text between the menu and end keywords can be any printable ASCII characters. The text in the menu file is case-sensitive.

A menu file can accommodate up to 2K bytes of data. A menu can refer to other menus or may be a single-level menu.

A menu may be referenced by any user entry in the **users** file, including the DEFAULT entry. The Menu reply item is the only reply item in the user entry when a menu is referenced.

```
DEFAULT     Auth-Type = System
            Menu = “menu1”
```

Using the above example, after user bob is authenticated, the **menu1** menu is displayed and he is prompted to make a selection. When bob selects a menu option, the corresponding service is provided.

The menu filename must be created under the **/etc/raddb/menus** directory of the RADIUS server. Refer to "Single-Level Menu" on page 5-2 and "Nested Menus" on page 5-3 for menu file examples. Menu names can be up to 120 alphanumeric printable ASCII characters and must be enclosed in double quotes (“ “).

Single-Level Menu

A single-level menu does not reference other menus. An example is displayed below; this menu would be **/etc/raddb/menus/menu1**.

```
menu
        *** Welcome to EDU OnLine ***
Please select an option:

        1. Start CSLIP session
        2. Start PPP Session
        3. Quit

        Option:
end
1
        Service-Type = Framed-User,
        Framed-Protocol = SLIP,
        Framed-IP-Address = 255.255.255.254,
        Framed-Routing = None,
        Framed-MTU = 1006,
        Termination-Menu = “menu1”
#
2
        Service-Type = Framed-User,
        Framed-Protocol = PPP,
        Framed-IP-Address = 255.255.255.254,
        Framed-Routing = None,
        Termination-Menu = “menu1”
#
3
        Menu = “EXIT”
#
DEFAULT
        Menu = “menu1”
```

In this example, after RADIUS authenticates the user, **menu1** is displayed and the user is prompted to select a service from this menu. Once the user has finished the SLIP or PPP session, the termination menu is displayed and the user is prompted to select a new service. If a Termination-Menu is not included in the reply items, the user is disconnected immediately after the SLIP or PPP session.

Nested Menus

Nested menus refer to other menus. In the example below, the menu has an **other** option; if a user chooses this option, a second menu is displayed.

```
menu
*** Welcome to the Internet Service ***
Please enter an option:
        ppp - Start PPP session
        telnet - Begin login session with a host
        other - Display a second menu
Option:
end
ppp
        Service-Type = Framed-User
        Framed-Protocol = PPP,
        Framed-IP-Address = 255.255.255.254,
        Framed-Routing = None,
        Framed-MTU = 1500
#
telnet
        Service-Type = Login-User,
        Login-IP-Host = 172.16.1.81,
        Login-Service = Telnet,
        Login-TCP-Port = 23
#
other
        Menu = “menu3”
#
DEFAULT
        Menu = “menu2”
```

SecurID Installation 6

Introduction

This chapter is an overview of the installation and configuration of SecurID when used with RADIUS 2.0. It serves as a quick reference guide for the ACE/Server and ACE/Client software. Refer to the Security Dynamics manual set for future ACE/Server software releases and detailed features of SecurID.

Note – Livingston Technical Support does not provide support for the ACE/Server and ACE/Client installation and configuration. Please contact Security Dynamics Technical Support at (617) 547-7820. Livingston Technical Support provides support for RADIUS when used in conjunction with SecurID after the sdshell utility has verified that the ACE/Server is working properly.

The ACE/Server and ACE/Client software version 2.1.1 is supported on the following platforms:

- SunOS version 4.1.4 on a Sun SPARCstation
- Sun Solaris version 2.5 on a Sun SPARCstation
- HP/UX version 10.01 on a Hewlett-Packard HP 9000 Series 7xx or 8xx
- AIX version 3.2.5 on an IBM RISC System/6000

The Security Dynamics authentication system (generally referred to as SecurID) consists of the following components:

- ACE/Server authentication server

  Stores usernames and serial numbers of tokens and performs calculations to verify the identity of users.

- ACE/Server client

  Machine generating the SecurID authentication attempt.

- Token

  A small, handheld device that generates a random number. A new number is generated and displayed every 60 seconds.

  There are three types of tokens supported in SecurID: the standard SecurID card, the SecurID Key Fob, and the SecurID PINPAD.

- PASSCODE

  A two-part password, consisting of a memorized personal identification number (PIN) followed by the current number displayed on the token.

Note – In order to use RADIUS with SecurID, the ACE/Server software must be running on the same UNIX host as the RADIUS server. If the ACE/Server software is installed on a different machine, then the RADIUS server must be an ACE/Server slave.

When SecurID is used with RADIUS, a connection proceeds as follows:

1. A remote user initiates a connection by dialing into the PortMaster.
2. The PortMaster prompts for the user's username and password.
3. The user enters a username. At the password prompt, the user enters a PASSCODE (PIN followed by the currently displayed number on the token).
4. The PortMaster forwards this information to the RADIUS server for authentication.
5. The RADIUS server examines the user file, scanning for the appropriate username. When the entry is located, it is examined to determine the user's authentication method.
6. When the RADIUS server discovers that the authentication method is SecurID, it forwards the username and PASSCODE to the ACE/Server for authentication.
7. The ACE/Server examines its database for the username and serial number of the user's token. It uses the serial number to verify the PASSCODE entered by the user. It also verifies that the time on the token is synchronized with the ACE/Server.
8. The ACE/Server sends the result of the database lookup (identity verified or not verified) to the RADIUS server.
9. If the user's identity was verified by the ACE/Server, the RADIUS server sends an access-accept message to the PortMaster along with the additional information from the RADIUS user entry. If the ACE/Server rejects the user's PASSCODE, the RADIUS server sends an access-reject message to the PortMaster.

SecurID Installation

The SecurID software package consists of a number of applications and utilities. This section covers the installation and use of two components, Progress and ACE/Server, and two utilities, sdshell and sdadmin.

SecurID software is not shipped with the PortMaster. This software must be ordered directly from Security Dynamics at (617) 547-7820.

Progress

Progress is an application development environment; this software must be installed before any additional SecurID software may be installed. In order to run Progress software with ACE/Server version 2.1.1, the Progress software version must be V7.3C01 or later.

Progress requires serial and control numbers for installation. Have these numbers available before beginning the installation.

To install Progress, follow the instructions in the Progress Installation Notes shipped with the Progress software. Note that Progress installs its software using the **proinst** utility, which must be run in an xterm window. To display an xterm on SunOS or Solaris, use the following command:

```
/usr/openwin/bin/xterm &
```

ACE/Server

The RADIUS 2.0 server is compatible with ACE/Server version 1.3 or higher. To install ACE/Server and the ACE/Server client software, complete the following steps:

1. Log in as root.
2. Read the ACE/Server tape into the ***ace_install*** directory of the ACE/Server machine.
3. ACE/Server installs its software using the **sdsetup** utility. If you are installing ACE/Server 2.0.1 on SunOS 4.1.4 or Solaris 2.5, the **check_os_version** subroutine of sdsetup must be modified to add the 4.1.4 or 2.5 string. If the appropriate string is not added, sdsetup aborts and displays an “unsupported OS” message.

Change the check_os_version subroutine of sdsetup to contain the following lines:

```
case "$SUN_OS" in
'4.1.3' | '4.1.4' ) VALID_OS=TRUE;;
      *      ) VALID_OS=FALSE;;

case "$SOL_OS" in
'5.3' | '5.4' | '5.5' ) VALID_OS=TRUE;;
      *      ) VALID_OS=FALSE;;
```

4. Run sdsetup to install ACE/Server.

   sdsetup cannot be run while the **sdconnect** process or **aceserver** daemon are running. Stop these processes before attempting to run sdsetup.

```
ace_install/sdsetup
```

   The ACE/Server software is typically installed on the same machine as the RADIUS server. To run ACE/Server on a different machine, the RADIUS server must be configured as an ACE/Server slave. See the *ACE/Server Installation and Configuration Guide* from Security Dynamics for instructions on configuring the ACE/Server Slave.

5. The sdsetup utility stops during the installation; at this point, add the SecurID UDP port number to the **/etc/services** file as follows:

```
securid       5500/udp                    #ACE/Server
securidprop   5100/udp                    #ACE/Server Slave
```

   To configure a slave server in addition to a master server, add the **securidprop** entry. If you are using NIS or NIS+, add these entries to the services NIS map on your NIS master and push the maps.

Note – Pushing the maps updates the database to include recently-entered information. Use the **make services** command on the NIS Master. For more details, consult your UNIX system documentation.

6. Continue sdsetup to install the ACE/Server client software. Complete instructions are given in Part 2 of the *ACE/Server Installation and Configuration Guide*.

sdadmin

sdadmin is an ACE/Server administration utility. Using sdadmin, a system administrator can add and delete users, assign PINs and tokens, and monitor network activity. sdadmin may be run in GUI (the default) or character mode.

To use sdadmin, complete the following steps:

1. Ensure that you are in the directory that contains the ACE/Server files. By default, ACE/Server software is installed in the **/usr/ace** directory.
2. Start the database broker (**sdconnect**) as root.

   ```
   /usr/ace/sdconnect start
   ```

   To stop the database broker, use the **sdconnect stop** command.

3. Start the ACE/Server daemon using the following command:

   ```
   /usr/ace/aceserver start
   ```

   To stop ACE/Server, use the **aceserver stop** command.

4. To automatically start the ACE/Server processes (sdconnect and aceserver) after the host is rebooted, add the following lines to **/etc/rc.local** or equivalent boot file of your UNIX system:

   ```
   if [ -x /usr/ace/aceserver ]; then
           /usr/ace/aceserver stop
           /usr/ace/sdconnect stop
           /usr/ace/sdconnect start
           /usr/ace/aceserver start
   else
           echo "Cannot start aceserver"
   fi
   ```

5. Launch sdadmin in GUI or character mode. Character mode requires the use of the **-c** switch, shown below.

```
/usr/ace/sdadmin &
or
/usr/ace/sdadmin -c &
```

   To run sdadmin in GUI mode, the host's window environment must be an implementation of X11R5 or later. If you are running SunOS on a SPARCstation, Sun OpenWindows is an X11R4 implementation, therefore, the GUI sdadmin utility cannot be displayed. To use the GUI sdadmin, the X11R5 kit (shipped with the ACE/Server software) must be installed. See Part 1 of the *ACE/Server Installation and Configuration Guide* for instructions.

6. Using the instructions in the *ACE/Server Administration Manual*, add users to the database, activate users on the client, and assign tokens to the users.
7. Choose a method of PIN assignment using the instructions in the "Pin Administration" chapter of the *ACE/Server Administration Manual*. Note that PINs may be assigned using RADIUS.

sdshell

sdshell is an ACE/Server client utility used to assign new PINs to users. It can also be used as a troubleshooting method to verify ACE/Server client/server communication before configuring RADIUS.

To execute sdshell, the sdconnect and aceserver daemons must be running.

To use sdshell, assign tokens to each user (see the previous section) and instruct a user to log into his or her account and run sdshell. sdshell runs through a PIN assignment sequence, as displayed in the example on the next page.

Instruct the user to enter a new PIN or press Return to have a PIN automatically generated. The user-generated PIN or system-generated PIN must be configured for the user when adding the user to the ACE/Server database.

```
% sdshell
Enter PASSCODE:

Enter your new PIN, containing 4 to 8 digits,
        or
Return to generate a new PIN and display it on the screen,
        or
Ctrl d to cancel the new PIN procedure:

Please re-enter new PIN:

Wait for the code on your token to change, then log in with the new PIN

Enter PASSCODE:
PASSCODE Accepted
```

The PIN options in sdshell (user-selected or system-generated) may vary, depending on how the PIN mode is configured. See the “Pin Administration” chapter of the *ACE/Server Administration Manual* for configuration instructions.

If the user’s new PASSCODE is accepted, communication between the ACE/Server client and server is successful. Proceed to the next section, “RADIUS Configuration.”

Note – Livingston Technical Support does not provide support for the ACE/Server and ACE/client installation and configuration problems. Please contact Security Dynamics Technical Support at (617) 547-7820. Livingston Technical Support provides support for RADIUS when used in conjunction with SecurID after the sdshell utility has verified that the ACE/Server is working properly.

RADIUS Configuration

Each SecurID user must have an entry in the RADIUS **users** file or must use a DEFAULT entry. In the entry, the Auth-Type check item must be **SecurID**, as shown in the following example:

```
DEFAULT     Auth-Type = SecurID
            Service-Type = Framed-User,
            Framed-Protocol = PPP,
            Framed-Address = 255.255.255.254,
            Framed-Routing = None,
            Framed-MTU = 1500
```

Users authenticated using this DEFAULT entry must be activated and assigned a token card using the ACE/Server **sdadmin** utility, as discussed on page 6-5.

When user bob dials into the PortMaster, the following prompts are displayed:

```
login: <enter username>
Password: <enter PIN number followed by a token code>
```

New PIN Assignment Using RADIUS

When a new user is added to the ACE/Server database, a token card is assigned to the user. If the token card does not have a PIN number, the user is put in a New PIN mode by the ACE/Server during the first connection attempt. To be authenticated in this mode, the user must select a PIN number.

Users may be forced into New PIN mode by the ACE administrator if the user has forgotten the PIN number or an attacker has learned the PIN number.

A New PIN mode user can assign the PIN number using RADIUS when he is dialing into the network. Refer to the “Pin Administration” chapter of the *ACE/Server Administration Manual* for more information on New PIN mode.

User-Generated PIN

When a user in New PIN mode is forced to create a PIN number via RADIUS, the "New PIN required" prompt appears to instruct the user to enter a PIN number.

```
login: bob
Password: <token code>
New PIN required: 1234
```

In the above example, when user bob dials into the network, he enters his login name at the login prompt. At the Password prompt, he enters the token code number and the PortMaster sends an access-request to the RADIUS server. The ACE/Server looks in its database and recognizes that user bob is a new PIN mode user. It sends an access-challenge to the PortMaster, and the New PIN required prompt is displayed prompting bob to enter a PIN number.

After bob enters his PIN number, the RADIUS server responds with the following message:

```
New PIN Accepted:  Wait for the next card code to login
Password:
```

In the subsequent login, at the Password prompt, bob's password would be a PIN number followed by a token code.

System-Generated PIN

The ACE/Server provides a system-generated PIN using the **sdshell** utility as described on page 6-6. sdshell displays the number on the screen for the user to memorize.

Note – sdshell displays the system-generated PIN for only 10 seconds. After the PIN number disappears, it cannot be viewed again.

When dialing into the network, the user enters his system-generated PIN at the "New PIN required:" prompt.

Next Cardcode

If a user enters a valid PIN and an invalid token code, the Next Cardcode prompt is displayed. This prompt also appears if the user's token card is not synchronized with the ACE/Server.

If an authorized user's token card is not synchronized with the ACE/Server, the user must wait until the token code changes, then enter the new token code number at the Next Cardcode prompt. After the system verifies the second token code, the user is authenticated.

If an unauthorized user enters a stolen PIN followed by a guessed token code, he is given three opportunities to enter the correct token code. If three invalid token codes are entered, the unauthorized user is disconnected.

```
login: bob
Password: <PIN number followed by invalid token code>
Next Cardcode:
```

In the above example, bob has entered a valid PIN number followed by an invalid token code. The Next Cardcode prompt appears, indicating that bob's token card is not synchronized with the ACE/Server. Bob must wait for 60 seconds for a new token code, then enter this code at the Next Cardcode prompt.

Troubleshooting

Progress version V7.3C01 has some known bugs that may cause problems during SecurID installation. This section covers the three bugs that you are most likely to encounter and suggests solutions for them. If you still have problems after trying these solutions, contact Security Dynamics Technical Support at (617) 547-7820.

sdadmin Cannot Find First Token

When sdadmin is launched for the first time, the error message "cannot find first token, database may be empty" may appear. To correct this problem, complete the following steps:

1. Log in as root.
2. Execute **sdnewdb**, located in the **/usr/ace** directory:

```
/usr/ace/sdnewdb
```

3. Choose the **Select All** option to create a new server and log databases.
4. Each batch of token cards from Security Dynamics is accompanied by a file. The file name consists of a 6-digit number and the **.asc** suffix. Run the **sdimport** utility to read the serial numbers of the token cards into the database.

```
/usr/ace/sdimport filename.asc
```

5. Re-launch **sdadmin** using either of the following commands:

```
/usr/ace/sdadmin &
or
/usr/ace/sdadmin -c &
```

sdserv.bi and sdlog.bi Consume Too Much Disk Space

The **sdserv.bi** and **sdlog.bi** files (located in the **/usr/ace** directory) occasionally need to be truncated. If they are not truncated, they may consume too much disk space and cause problems for the ACE/Server database. To truncate these files, use the following commands:

```
/usr/dlc/bin/_proutil -c truncate sdserv.bi
/usr/dlc/bin/_proutil -c truncate sdlog.bi
```

sdadmin Runs out of Memory

When sdadmin is executed on Solaris 2.4 or HP/UX 9.03 hosts, an “out of memory” message is displayed. To correct this problem, complete the following steps:

1. Add the kernel parameters shown in the following example to the **/etc/system** file on the ACE/Server host.

```
set semsys:seminfo_semmni=64
set semsys:seminfo_semmns=200
set semsys:seminfo_semmnu=100
set semsys:seminfo_semmsl=50

set shmsys:shminfo_shmmax=16777216
set shmsys:shminfo_shmmni=100
set shmsys:shminfo_shmseg=16
```

2. Reboot the host using the following command:

```
reboot -rv
```

RADIUS Accounting 7

Introduction

RADIUS accounting logs information about dial-in connections. This information is often used for billing purposes. RADIUS accounting consists of a client/server format; as transactions occur, they are recorded in a file named **/usr/adm/radacct/*portmastername*/detail** on the RADIUS accounting server.

How RADIUS Accounting Works

RADIUS accounting consists of an accounting server and accounting clients (PortMasters). The radiusd daemon for accounting is a child process of the radiusd authentication daemon; it starts automatically when radiusd is executed.

The RADIUS accounting server uses the UDP protocol, and listens for UDP packets at port 1646.

RADIUS accounting consists of the following steps:

1. The PortMaster (accounting client) sends an **accounting-request** packet containing the record of an event to the accounting server.
2. The accounting server sends an **accounting-response** packet back to the PortMaster to acknowledge receipt of the request.
3. If the PortMaster does not receive a response, it continues to send accounting-requests until it receives a response.

   A backoff algorithm is used to determine the delay between accounting-requests if an accounting-response is not received.

4. The PortMaster records the number of seconds that have passed between the event and the current attempt to send the record; this number is the **Acct-Delay-Time** value. As additional time passes before an accounting-response is received, the Acct-Delay-Time is updated.

5. When the user is connected, a Start accounting record is recorded in a file called **/usr/adm/radacct/*portmastername*/detail** on the accounting server.

   The Start record typically contains the Session-Id, the User-Name, Service-Type, Login-Service, Login-IP-Host, Acct-Delay-Time, and other relevant information from a user's entry in the users file.

Note – When the user is disconnected, a Stop record is generated. This record contains the same information as the Start record, however, it also includes **Acct-Session-Time**, which records the time (in seconds) of a user's session.

Getting Started

Select a UNIX host to use as the RADIUS accounting server. This host may be the same host as the RADIUS server used for authentication or a separate host.

Choose a host with the following characteristics:

- Located in a secure physical location
- Root access is limited to the Security Officer or System Administrator
- Offers a limited number of user accounts, preferably none
- Offers basic memory
- Offers enough disk space to store the RADIUS accounting **detail** files

  For typical installations, allocate 50 MB per 1000 users if the logs are rotated monthly. Keep in mind that it is much better to allocate too much space than too little; your usage may vary.

  For example, if you have 1000 users, one port for every 10 users, an average connect time (per user) of one hour, and all ports are in use around the clock, one month of logs would require 50 MB of disk space:

700 bytes / session * 1000 users * 1 port / 10 users * 1 session / hour * 24 hours / day * 30 days / month

Livingston recommends the use of a secondary RADIUS accounting server. The primary accounting server is always used first; if this server is unavailable, the secondary server is used.

Client Configuration

To configure RADIUS accounting information on a PortMaster, see Chapter 3, "RADIUS Client Configuration."

Server Configuration

To install the RADIUS accounting server, log into the selected accounting server as root. Create a **radacct** directory within the **/usr/adm** directory.

```
mkdir /usr/adm/radacct
chmod 700 /usr/adm/radacct
```

RADIUS accounting automatically creates subdirectories within the **/usr/adm/radacct** directory for each PortMaster serving as a RADIUS accounting client and logs the accounting start and stop records to the **detail** file in the directory.

RADIUS Accounting Flags

The flags associated with the parent **radiusd** are described in Chapter 2, "RADIUS Server Configuration."

The radiusd accounting daemon may also be used with the flags shown in Table 7-1.

Table 7-1 radiusd Accounting Daemon Flags

Flag	Purpose
-a	Specifies an alternate directory for RADIUS accounting logs. The default directory is **/usr/adm/radacct**.
-v	Displays the RADIUS version number without starting the radiusd daemon. This flag also applies to the RADIUS authentication server; the RADIUS authentication and accounting servers have the same version number.

Accounting Attributes

In order for RADIUS accounting to function, a series of accounting attributes (listed below) are defined in the **/etc/raddb/dictionary** file on the RADIUS server.

Acct-Status-Type

Acct-Status-Type has two values: Start and Stop. A Start record is created when a user session begins. A Stop record is recorded when the session ends.

Acct-Delay-Time

The PortMaster records the number of seconds that have passed between the event and the current attempt to send the record; this number is the **Acct-Delay-Time** value.

The approximate time of an event can be determined by subtracting the Acct-Delay-Time from the time of the record's arrival on the RADIUS accounting server.

Acct-Session-Id

Acct-Session-Id is a unique number assigned to each Start and Stop record to make it easy to match the Start and Stop records in a detail file, and to eliminate duplicate records.

The Acct-Session-Id is a string consisting of 8 uppercase hexadecimal digits. The first two digits increment each time the PortMaster is rebooted. The next 6 digits begin at 0 (for the first user login after a reboot) and increment up to approximately 16 million logins. This is equal to one user logging into each port of a 30-port unit every minute for an entire year.

Acct-Authentic

Acct-Authentic records whether the user was authenticated via RADIUS or by the PortMaster User Table. Accounting records are not generated for passthrough users, as those users are authenticated by the destination host.

Acct-Session-Time

The **Acct-Session-Time** records the user's connection time in seconds. This information is only included in Stop records.

NAS-Port-Type

NAS-Port-Type records the type of port used in the connection. The port type may be any of the following: Async, Sync, ISDN, ISDN-V120, or ISDN-V110.

Acct-Input-Octets and Acct-Output-Octets

Records the number of bytes received (**Acct-Input-Octets**) and sent (**Acct-Output-Octets**) during a session. These values only appear in Stop records.

Called-Station-Id and Calling-Station-Id

Called-Station-Id and **Calling-Station-Id** record the called and calling numbers. This information is recorded when the NAS-Port-Type is ISDN, ISDN-V120, or ISDN-V110 where supported by the local Telco.

Timestamp

Timestamp records the time of arrival on the RADIUS Accounting host measured in seconds since the epoch (00:00 January 1, 1970).

This attribute provides a machine-friendly version of the logging time at the beginning of the accounting record. To find the actual time of the event, subtract Acct-Delay-Time from Timestamp.

Request-Authenticator

The **Request-Authenticator** attribute only appears in an accounting record when the RADIUS 2.0 server notices a problem with the accounting request's digital signature.

A Request-Authenticator of **None** means that the accounting request was not digitally signed, and was probably sent by a PortMaster running a version of ComOS that did not sign accounting packets. If the Request-Authenticator value is **Unverified**, the accounting request signature did not match the expected value. Ensure that the shared secret on the PortMaster matches the shared secret in the **/etc/raddb/clients** file.

Acct-Terminate-Cause

The **Acct-Terminate-Cause**, shown in Table 7-2, indicates the cause of a session's termination. This information only appears in Stop records.

Table 7-2 Session Termination Causes

Termination Cause	Meaning
Admin-Reset	Port was reset by an administrator.
Host-Request	Session was disconnected or logged out by the Login-IP-Host. This can indicate normal termination of a login session or that the remote host has crashed or become unreachable.
Idle-Timeout	Idle timer expired for user or port.
Lost-Carrier	Session terminated when the modem dropped DCD. This can indicate any of the following: the user or his modem hung up the phone from their end (in which case there is no problem), the line was dropped, the line took a noise hit too severe for the modem to recover from, or the local modem dropped DCD for some other reason.
Port-Error	PortMaster had to reset the port. Most commonly occurs when a device attached to the port caused too many interrupts.
Session-Timeout	Session timer expired for user.
User-Error	PortMaster received a PPP Configuration Request or ACK when a session was already established, so it terminated the session. This is caused by a PPP implementation error in the dial-in client.
User-Request	Dial-in PPP client requested that we terminate the connection. This message is expected from a proper PPP client termination.

Examples

The following example displays two accounting records in a PortMaster's detail file.

```
Tue Jul 30 14:48:18 1996
        Acct-Session-Id = "35000004"
        User-Name = "bob"
        NAS-IP-Address = 172.16.64.91
        NAS-Port = 1
        NAS-Port-Type = Async
        Acct-Status-Type = Start
        Acct-Authentic = RADIUS
        Service-Type = Login-User
        Login-Service = Telnet
        Login-IP-Host = 172.16.64.25
        Acct-Delay-Time = 0
        Timestamp = 838763298

Tue Jul 30 14:48:39 1996
        Acct-Session-Id = "35000004"
        User-Name = "bob"
        NAS-IP-Address = 172.16.64.91
        NAS-Port = 1
        NAS-Port-Type = Async
        Acct-Status-Type = Stop
        Acct-Session-Time = 21
        Acct-Authentic = RADIUS
        Acct-Input-Octets = 22
        Acct-Output-Octets = 187
        Acct-Terminate-Cause = Host-Request
        Service-Type = Login-User
        Login-Service = Telnet
        Login-IP-Host = 172.16.64.25
        Acct-Delay-Time = 0
        Timestamp = 838763319
```

The Acct-Status-Type attribute in the record indicates whether the record was sent when the connection began (Start) or when it ended (Stop). In the Start record above, the Acct-Session-Id is listed at the beginning of the record. Note that this value matches the Acct-Session-Id of the Stop record, indicating that these records correspond to the same session.

User-Name specifies the username, in this case, **bob**. NAS-IP-Address specifies the IP address of the PortMaster. NAS-Port-Type specifies that this is an asynchronous connection. Acct-Authentic specifies that bob is authenticated via RADIUS. Service-Type and Login-Service specify that bob is a login user using Telnet. Login-IP-Host specifies the host that user bob logged into.

In the Stop accounting record, Acct-Session-Time specifies that bob's connection lasted 21 seconds. Acct-Input-Octets indicates that 22 bytes of incoming traffic was received; Acct-Output-Octets indicates that 187 bytes of outgoing traffic was sent.

The Acct-Terminate-Cause indicates that a Host-Request terminated the session, meaning that bob logged out of the host or that the host logged him out. The Acct-Delay-Time is 0 seconds, indicating that the RADIUS accounting server received the accounting-request on the first try.

For more information on accounting attributes, see page 7-4 and page 4-16.

The following example displays Start and Stop accounting records for an ISDN PPP connection.

```
Wed May  8 10:51:12 1996
        Acct-Session-Id = “2400020E”
        User-Name = “Pbob”
        NAS-IP-Address = 172.16.1.21
        NAS-Port = 12
        NAS-Port-Type = ISDN
        Acct-Status-Type = Start
        Acct-Authentic = RADIUS
        Called-Station-Id = “5551111”
        Calling-Station-Id = “5105552222”
        Service-Type = Framed-User
        Framed-Protocol = PPP
        Framed-Address = 172.16.93.1
        Acct-Delay-Time = 0
        Timestamp = 838763356

Wed May  8 12:50:49 1996
        Acct-Session-Id = “2400020E”
        User-Name = “Pbob”
        NAS-IP-Address = 172.16.1.21
        NAS-Port = 12
        NAS-Port-Type = ISDN
        Acct-Status-Type = Stop
        Acct-Session-Time = 7177
        Acct-Authentic = RADIUS
        Acct-Input-Octets = 14994
        Acct-Output-Octets = 90862
        Called-Station-Id = “5551111”
        Calling-Station-Id = “5105552222”
        Service-Type = Framed-User
        Framed-Protocol = PPP
        Framed-Address = 172.16.93.1
        Acct-Delay-Time = 0
        Timestamp = 838763378
```

In the Start record of the example above, the NAS-Port-Type specifies that the user Pbob is using ISDN for his connection. Called-Station-Id and Calling-Station-Id specify the source and destination of the ISDN call. Service-Type and Framed-Protocol indicate that user Pbob is a framed user using PPP to establish the connection.

The Stop record in this example indicates that the login time for user bob was 7177 seconds or 1 hour, 59 minutes, and 37 seconds. The Acct-Input-Octets and Acct-Output-Octets indicate that the incoming traffic for this session was 14994 bytes, and outgoing traffic was 90862 bytes.

Note – Examples of PERL scripts to process the RADIUS accounting logs are available at Livingston's FTP site at **ftp://ftp.livingston.com/pub/le/radius/**.

Troubleshooting A

Introduction

This appendix provides hints and tips for troubleshooting the RADIUS authentication server and the RADIUS accounting server.

Troubleshooting RADIUS Authentication

Most problems are caused by skipping a step during installation. Carefully check the installation instructions in Chapter 2, "RADIUS Server Configuration" and Chapter 3, "RADIUS Client Configuration" to ensure that the authentication server was properly installed.

If the problem is not solved after reviewing the instructions in Chapter 2 and Chapter 3, read the troubleshooting suggestions in this section.

Checking radiusd Daemon

1. Use **radiusd -v** to display the version number.
2. Make sure **/etc/radiusd** is running.
3. Make sure in the **/etc/raddb** directory (or wherever you specify with the **-d** flag) that you have the following files: dictionary, users, and clients. If you are using RADIUS menus, check the **menus** subdirectory.
4. Use **radiusd -x** to view incoming and outgoing packets from RADIUS.

Checking the PortMaster

1. Make sure that the RADIUS server is reachable from the PortMaster.
2. Make sure that security is **on** for each port:

```
Command> set all security on
Command> save all
Command> reset all
```

   When security is on, the show port command displays security in parentheses as follows: **(Security)**.

3. Use the **show global** command to ensure that the RADIUS server IP address is set on the PortMaster.
4. Make sure the secret set on the PortMaster using the **set secret *password*** command matches the secret in the **/etc/raddb/clients** file on the RADIUS server.

   The PortMaster will not display the shared secret, however, you may set it again if you are not sure that it is set properly. If you update the shared secret, make sure to use the **save all** command to save the shared secret in the PortMaster's nonvolatile memory.

Checking /etc/raddb/users

1. Items in the user entries are case sensitive. Verify the spelling and case of each line of the users file. Compare keywords against the **/etc/raddb/dictionary** file to ensure that they are the same.
2. Verify that the user can authenticate with a clear text password before authenticating with **Auth-Type = System** or **Auth-Type = SecurID**.

Host Unavailable

If a **Host Unavailable** message is displayed after a username is entered at the login prompt, security for the port is not enabled and **rlogind** and **in.pmd** are not running on the host configured for that port. The PortMaster is attempting to do a pass-through login to a host that is not prepared to accept it.

To verify that this is the problem, enter the following command. Replace s1 with the port that you are using.

```
Command> show s1
```

Security should be displayed in parentheses, **(Security)**. If it is not displayed, enter the following commands:

```
Command> set s1 security on
Command> reset s1
Command> save all
```

Invalid Login after 30 second wait

The PortMaster sends 10 access-requests at 3-second intervals and then displays an **Invalid Login** message. This message may indicate one of the following problems:

1. RADIUS is not running on the server. Check to ensure that **/etc/radiusd** is running.
2. The RADIUS server is not defined correctly on the PortMaster. Check the RADIUS server information using the following commands:

```
Command> show global
Command> show netcon
```

3. There is no entry for the PortMaster in the **/etc/raddb/clients** file. This may be verified by running **radiusd -x**. If the output of **radiusd -x** produces 10 access-requests with the same ID, but does not produce a corresponding access-accept or access-reject message, the PortMaster hostname is probably missing or not defined correctly in the /etc/raddb/clients file.
4. radiusd responses are not getting back to the PortMaster. Examine the routing table on the RADIUS server host and ping the PortMaster from this host.

5. The PortMaster is ignoring radiusd responses. This is a relatively rare occurrence, usually caused by one of the following:
   - Multiple IP addresses for a single Ethernet interface on the RADIUS server host
   - Multiple Ethernet interfaces, and the RADIUS server is replying to a request from the PortMaster on a different interface than the interface that received the request
   - The source of the access-accept or access-reject packet does not match the destination of the access-request packet

Result of radiusd -x output

If **radiusd -x** shows more than one access-reject sent for the same ID, check the following:

1. Check the route back to the PortMaster; ensure that replies are getting to the PortMaster.
2. Check to see if the RADIUS server host has more than one Ethernet port or multiple IP addresses assigned to the same Ethernet interface.
3. Check for packet filters between the RADIUS server host and the PortMaster filtering out the RADIUS return packets.
4. On the PortMaster, use **ptrace** to show packets returning from the host running radiusd:

```
Command> add filter r
Command> set filter r 1 permit udp src eq 1645
Command> set filter r 2 permit icmp
Command> ptrace r
```

Note – ptrace on a PortMaster does not show UDP or ICMP packets generated on the PortMaster itself. Outgoing RADIUS access requests are not shown, however, returning packets are displayed. To turn off tracing, use the ptrace command with no arguments.

Check the source address of a packet during tracing. A multihomed RADIUS host may be using the wrong source address when replying to access-request packets.

If radiusd -x shows an access-reject right away, check the following:

1. Check the spelling of the username and password. The case must match exactly.
2. Check syslog for errors from radiusd.
3. Use the **show table user** command to verify that the user is not in the PortMaster User Table; the local User Table is always checked first during authentication attempts.
4. If **Auth-Type = System** is not working, attempt to use a cleartext password in the user entry.
5. If **Auth-Type = System** is specified on a system that has shadow passwords, ensure that radiusd is run as root in order to access the shadow passwords.
6. Verify the spelling, case, and syntax of the **/etc/raddb/users** file. If radiusd finds any errors in the user entry, it sends an access-reject message and logs an error to syslog.
7. Check that the shared secret in **/etc/raddb/clients** matches the one set on the PortMaster using the **set secret** command
8. If using PMconsole, ensure that the Return key was not pressed when the cursor was in the RADIUS Secret field of the dialog box. Pressing the Return key at this point erases the secret when the Save button is clicked.

Troubleshooting RADIUS Accounting

Most problems are caused by skipping a step during installation. Carefully check the installation instructions in Chapter 2 and Chapter 3 to ensure that the accounting server was properly installed.

If the problem is not solved after reviewing these instructions, check the following:

1. Make sure the **/usr/adm/radacct** directory exists and that the account used to execute radiusd has write permission to this directory.
2. Run radiusd using the **-v** flag to ensure that radiusd is version 1.16 or 2.0.
3. Make sure that you do not have any other process bound to UDP port 1646. Kill radiusd and use the **netstat -a** command; there should not be anything bound to UDP ports 1645 or 1646. Start radiusd and use the netstat -a command again; radiusd should now be listening on both 1645 and 1646.

   Some operating systems display the sockets symbolically as **.radius** and **.radacct** rather than .1645 and .1646.

4. Use the **show global** command to verify that the IP address of the accounting host has been configured on the PortMaster. If it has not been configured, set it using the **set accounting *IPaddress*** on the PortMaster, where *IPaddress* is the IP address of the host running radiusd.
5. Check syslog (auth.warning) for error messages from radiusd. During normal use, very few error messages should appear.
6. Ping the PortMaster from the RADIUS server to check connectivity.
7. If the previous suggestions do not solve the problem, run **radiusd -x** on the RADIUS server host and check to determine if accounting records are displayed.

Index

A

accounting
- flags 7-3
- logged information 7-4
- overview of 7-1
- server configuration 7-3
- troubleshooting A-5

Acct-Terminate-Cause 7-6

ACE/Server 6-3

advantages of RADIUS 1-1

Auth-Type
- Local 4-3
- SecurID 4-3
- System 4-3

B

builddbm 4-19

C

callback 4-10
- Callback-Framed-User 4-6, 4-10
- Callback-Login-User 4-6, 4-10

Called-Station-Id 7-5

Calling-Station-Id 7-5

check items 4-2
- Auth-Type 4-2
- examples 4-25
- Expiration 4-4
- Filter-Id 4-12
- Framed-IP-Address 4-8
- Framed-IP-Netmask 4-8
- Framed-MTU 4-14
- Framed-Protocol 4-7
- Framed-Route 4-8
- NAS-IP-Address 4-5
- NAS-Port 4-5
- NAS-Port-Type 4-5
- Outbound-User 4-9
- Prefixes 4-5
- Suffixes 4-5

clients
- clients file 2-6
- configuring client information 2-6
- configuring PortMaster 3-1
- NAS-IP-Address 4-5
- NAS-Port 4-5
- NAS-Port-Type 4-5

compression, TCP/IP 4-14

conventions in this manual ix

converting IPX decimal to dotted quad 4-15

D

DBM database 4-19

DEFAULT user entry 4-17

directory structure, RADIUS server 2-1

disconnecting users 4-16

document conventions ix

documentation, related viii

E

examples
- user entries 4-25

F

filters 4-12
flags, radiusd 2-5, 7-3
Framed-Compression 4-14
Framed-IP-Address 4-8
Framed-IP-Netmask 4-8
Framed-IPX-Network 4-15
Framed-MTU 4-14
Framed-Protocol 4-7
Framed-Route 4-8
Framed-Routing 4-11

H

host unavailable message A-3

I

idle timeouts 4-16
in.pmd 4-13
installation
    accounting 7-3
    RADIUS 2-2
    SecurID 6-1
invalid login message A-3
IPX
    setting network information 4-15

L

Local Auth-Type 4-3
Login-IP-Host 4-13
Login-Service 4-13
Login-User 4-6

M

menus
    nested 5-3
    overview 5-1
    single-level 5-2
MTU, setting 4-14

N

NAS information
    NAS-IP-Address 4-5
    NAS-Port 4-5
    NAS-Port-Type 4-5, 7-5
nested menus 5-3

O

operating systems, supported 1-2
Outbound-User 4-6, 4-9
overview of RADIUS 1-1

P

packet filters 4-12
Pass-Thru Login option 3-3
passwords
    expiration date 4-4
    location of 4-2
PERL script 4-15
PIN assignment, SecurID 6-8
PMconsole configuration 3-3
pminstall 2-2
Port-Limit 4-17
Prefixes 4-5
Progress software 6-3

R

RADIUS
- 2.0 reply items 4-16
- accounting 7-1
- directory structure 2-1
- installation 2-2
- menus 5-1
- overview of 1-1
- server, primary 3-1
- server, secondary 3-1
- troubleshooting A-1
- users file 4-1

radiusd flags 2-5, 7-3

related documentation viii

remote host information 4-13

reply items 4-6
- Callback-Framed-User 4-10
- Callback-Login-User 4-10
- examples 4-25
- Framed-Compression 4-14
- Framed-IPX-Network 4-15
- Framed-Routing 4-11
- Idle-Timeout 4-16
- Login-IP-Host 4-13
- Login-Service 4-13
- Port-Limit 4-17
- Service-Type 4-6
- Session-Timeout 4-16

RIP configuration 4-11

S

sdadmin 6-5

sdsetup 6-4

sdshell 6-6

SecurID
- ACE/Server 6-3
- Auth-Type 4-3
- entry in users file 6-8
- installation 6-1
- PIN assignment 6-8
- Progress 6-3
- sdadmin 6-5
- sdshell 6-6
- technical support 6-1
- troubleshooting 6-11

Security option 3-3

servers, selecting 1-5, 7-2

Service-Type 4-6
- Callback-Framed-User 4-6
- Callback-Login-User 4-6
- Login-User 4-6
- Outbound-User 4-6

session termination, reasons for 7-6

Session-Timeout 4-16

shared secret 2-6

single-level menus 5-2

Start and Stop records 7-4

Suffixes 4-5

support, technical x

System Auth-Type 4-3

T

TCP/IP header compression 4-14

technical support
- Livingston x
- Security Dynamics 6-1

troubleshooting
- accounting A-5
- authentication A-1
- SecurID 6-11

U

user entries
- complete list of options 4-20
- DEFAULT 4-17
- examples 4-25

username, restrictions 4-1

users file 4-1

users, disconnecting 4-16