Chapter 12

Integrating Merit RADIUS

Use the information in this chapter to integrate Merit AAA with JUNOSe routers. Refer to the *SRC-PE Release Notes* for information about compatibility of this SRC release with Merit AAA 4.22E releases. The SRC software does not support the use of RADIUS with JUNOS routing platforms.

This chapter contains the following sections:

Information about the simpler case of integrating the Merit AAA 4.22E server with the JUNOSe router (without using the SRC software) is provided.

The SRC software can take advantage of a RADIUS server to authenticate against an LDAP server, which is used to store subscriber and service information, among other items.

NOTE: The Merit AAA 4.22E product is provided with the SRC software; all others are third-party products. We recommend that the Merit AAA 4.22E RADIUS solution be used for trial purposes only.

System Requirements for the Merit AAA Server

The following are the system requirements:

- Operating system—Solaris 8 or higher
- RAM—At least 64 MB of working memory
- Disk—Depends on external database support and storage time of the accounting log files; at least 40 MB of hard-disk space

Installing Merit AAA

The Merit AAA 4.22E server package is part of the SRC software distribution and is called UMCradius. The installation procedure for the Merit AAA 4.22E for a Solaris host is described in the *SRC-PE Getting Started Guide, Chapter 28, Installing the SRC Software on a Solaris Platform*.

If the Merit AAA 4.22E server is installed from the SRC software distribution, the dictionary file is already extended by the JUNOSe-specific attributes.

LDAP Features for the Merit AAA Server

The Merit AAA server package is composed of functional building blocks called authentication/authorization transfer vectors (AATVs). These AATVs perform a specific function, such as UNIX password checking or authentication against an LDAP directory.

LDAP authentication allows all user configurations to be done and stored in the LDAP directory, eliminating the need to edit the server's configuration files to change user information. In addition to being a policy repository, the LDAP directory also replaces the user's file or the UNIX password file as the place to store a user ID and password; performance is higher when one is dealing with a large number of users.

The ProLDAP AATV is an authentication AATV that performs two functions. First, it checks the validity of the user's ID and password. Second, if authentication is successful, the AATV loads attribute value pairs into the aaaCheck-list, aaaDeny-list, and aaaReply-list in the authentication request. The ProLDAP AATV uses a set of asynchronous LDAP API functions that allow the ProLDAP to be a direct-type AATV. Using a poll interface inside the AAA server engine, the main process is not blocked by the ProLDAP AATV; therefore, the ProLDAP process does not create and run a child process. The asynchronous LDAP API functions allow an LDAP search, for example, to be sent out to a directory server without waiting for the search result to come back. Later on, the owner of the search may poll the LDAP client to find out if any result is available from the search.

The ProLDAP AATV is designed to work with different LDAP directory configurations. The directory may be configured to either allow or not allow the user password to be returned to the AAA server in an LDAP search. The ProLDAP AATV may be configured to first try searching for the user in the directory. If the password is returned, the ProLDAP AATV makes a password comparison to authenticate the user. Otherwise, the ProLDAP AATV will try to bind the user to the directory with the given password. ProLDAP may be configured to do a bind or search operation, but only if the directories are known to support those configurations.

Configuration of the LDAP search operations based on realms is described in *Configuring LDAP Authentication for the Merit AAA Server* on page 148.

Configuring UDP Ports for the Merit AAA Server

The transaction-based RADIUS protocol uses two UDP ports: one for authentication packets and one for accounting packets. The ports must be configured on two sides: the Merit AAA server and the RADIUS clients (SRC software and JUNOSe router).

The officially assigned UDP port numbers are:

- 1812 for authentication
- 1813 for accounting

Early deployments of RADIUS used 1645/udp for authentication packets and 1646/udp for accounting packets.

The Merit AAA RADIUS server uses the latter ports by default, whereas the JUNOSe router uses the official ports by default.

There are two ways to change these settings:

- Edit the */etc/services* file to contain two entries for RADIUS authentication and accounting service that specify the ports you wish to use:

  radius 1812/udp # RADIUS Authentication
  radacct 1813/udp # RADIUS Accounting

- Override all default and configured values at server startup with the **radiusd -p** and **radiusd -q** command line options. The SRC software installs the Merit AAA server with a start script, called *rad*, which uses ports 1812 and 1813 for authentication and accounting (see next section).

Starting the Merit AAA Server

We include a script for starting the RADIUS server. The filename of the script is *rad*; it is installed in the directory */opt/UMC/radius*.

To start the Merit AAA server:

1. Log in as `root`.
2. Change the directory to */opt/UMC/radius*, and start the program by typing:

```
cd /opt/UMC/radius
./rad start
```

During startup, the RADIUS server binds to the LDAP server. This process requires that the LDAP server be running before the RADIUS server is started.

The RADIUS process is automatically started whenever the Solaris host is started.

If you are using a Merit AAA server that is not supplied by Juniper Networks, you can start the Merit server by launching the RADIUS process.

The syntax is as follows:

```
radiusd -d <conf directory> -da <aaatv directory> -dl <log directory>
-A <acct directory> -n -p <auth port> -q <acct port> -f <fsm file> -pp
<auth relay port> -qq <acct relay port> -g {'syslog' | 'logfile' | 'stderr'} -l
<log format> -t <timeout> -v -z -h
```

where:

- -d—Directory of users, clients, authfile, dictionary, configuration files
- -da—Directory in which the binary AATVs reside
- -dl—Directory into which the log files should go
- -A—Directory in which to put accounting records
- -n—New session table at start for local authorization service (LAS)
- -p—Port number on which to listen for authentication requests
- -q—Port number on which to listen for accounting requests
- -f—Allows the user to specify an alternate finite state machine (FSM) table file instead of the default *radius.fsm* file
- -pp—Port number on which to relay authentication requests

- -qq—Port number on which to relay accounting requests
- -g—Type of logging; select logfile, syslog, or stderr logging
- -t—Inactivity timeout value (minutes)
- -v—Displays RADIUS version
- -h—Displays this help syntax

Stopping the Merit AAA Server

We include a script for stopping the RADIUS server. The filename of the script is *rad*; it is installed in the directory */opt/UMC/radius*.

To stop the RADIUS server:

1. Log in as `root`.
2. Change the directory to */opt/UMC/radius* and stop the program by typing:

   cd /opt/UMC/radius
   ./rad stop

Displaying the Status of the Merit AAA Server

We include a script for displaying the status of the RADIUS server. The filename of the script is *rad*; it is installed in the directory */opt/UMC/radius*.

To check the status of the Merit AAA server:

1. Log in as `root`.
2. Change the directory to */opt/UMC/radius* and display the status by typing:

   cd /opt/UMC/radius
   ./rad status

Extending Dictionary Files with JUNOSe Parameters for the Merit AAA Server

In addition to supporting standard RADIUS attributes, the JUNOSe router supports JUNOSe-specific attributes. These attributes must be introduced to the Merit AAA server. You must use the RADIUS attributes for both Merit AAA server–JUNOSe router integration and Merit AAA server–JUNOSe router–SRC integration. See the *JUNOSe Broadband Access Configuration Guide* for more information about the RADIUS attributes supported by the JUNOSe router.

If you use the Merit AAA server package that we supply, you do not need to extend the dictionary files, and you can proceed to the next section. If, however, you use another version of the Merit AAA server, you must extend the dictionary file.

In such a case, move to the configuration directory of the Merit AAA installation, and edit the dictionary file. Append the JUNOSe-specific attributes to the dictionary file in the following way:

1. Access the directory in which you installed the Merit AAA installation.

   cd /opt/UMC/radius

2. Open the *radius.dct* file.

3. At the end of the file, add the JUNOSe attributes in the ERX RADIUS Dictionary file. To locate the ERX RADIUS Dictionary file, see the JUNOSe software documentation for the supported release on the Juniper Networks Web site at

   http://www.juniper.net/techpubs/software/

The next step defines the JUNOSe router as the network access server (NAS) to be recognized by the Merit AAA server. This involves the extension of the vendor file, which is located in */opt/UMC/radius/etc*.

The vendor file contains a list of zero or more vendor entries. Each vendor entry contains a vendor name and a vendor number. Each entry optionally contains an interim way of mapping external (with respect to the RADIUS server) attribute numbers to internal (with respect to the RADIUS server) vendor-specific attributes. This optional mapping is used on RADIUS requests and responses. The following lines must be added, where every line starting with the character "#" indicates a comment:

```
# Juniper Networks Inc. extensions
ERX-VSA.attr ERX-VSA.value 4874 Juniper
```

Configuring LDAP Authentication for the Merit AAA Server

The SRC software assumes that all RADIUS authentications are performed against the SDX LDAP directory. This section also applies to Merit AAA server integration with a JUNOSe router if Merit AAA authenticates against an LDAP directory. Integration of the JUNOSe-specific attributes, such as primary Domain Name System (DNS), virtual router, and others, must be performed, which is outlined in this section.

Configuring the Merit AAA Server

The Merit AAA server configuration for the ProLDAP AATV is done through the *authfile* file, which is stored in the configuration directory */opt/UMC/radius/etc*. You must configure these tasks:

- How the Merit AAA server performs authentication
- Which external database is used for authentication, based on the realm name

Administrators must create a table in the *authfile* file for each realm name. Merit AAA supports up to four LDAP directories, which could be used for authentication for each realm.

```
realm PROLDAP description
{
Filter-Type bin | cis
Directory directory-1
{
Host dir1.host.com
Port port-number
Administrator directory-manager-dn
[Password directory-manager-password]
SearchBase realm-search-base-in-directory
Authenticate Auto | Bind | Search
}
...
}
```

where

- realm—Identifies the realm name that is used during PPP login (username@realm). The special value NULL specifies treatment of any incoming access request, where no realm name is submitted during the PPP login.
- PROLDAP—Identifies that this table is valid for the ProLDAP AATV.
- Filter-Type—Identifies treatment of the user ID. Valid values are either case sensitive (bin) or not case sensitive (cis).
- Directory—Identifies the start of the directory section. Up to four directory sections are supported per realm. If the value contains spaces or tabs, it must be enclosed by either the double-quote or the single-quote character. Merit AAA uses the round-robin method for those identified directories.
- Host—The value (fully qualified DNS name or IP address) identifies the LDAP directory.
- Port—Identifies the port the LDAP server listens to.
- Administrator—DN that specifies the user entry AAA uses to log in against the LDAP directory. The DN must be specified if Authenticate is set to search.
- SearchBase—DN that represents the start point of the LDAP search operation for that realm.
- Authenticate—Identifies how Merit AAA authenticates incoming access requests. Valid values are:
  - Auto—AAA performs a search as the configured administrator (searches anonymously if no configured administrator), anticipating that the password is in the result. It binds as the user if the password is not available.
  - Bind—AAA tries to bind with the user ID and password specified during the PPP login.

- Search—AAA binds and performs search operation. LDAP returns the user password, which is compared with the password submitted during the PPP login.

NOTE: The SRC software uses the search option.

The following *authfile* example depicts the treatment of PPP logins without any realms and with the realm name *isp1.com*:

```
# This is a realm entry for an LDAP Server with PROLDAP with NO Realm
#
NULL  PROLDAP Default-Setting
{
Filter-Type BIN
Directory SDX
{
Host 123.45.3.1
Port 389
Administrator "cn=radius,ou=components,o=operators,o=umc"
Password      "radius"
SearchBase    "retailerName=default, o=users, o=umc"
Authenticate  search
}
}
# This is a realm entry for two LDAP Server with PROLDAP with Realm isp1.com
#
virneo.com PROLDAP Virneo-Setting
{
Filter-Type BIN
Directory virneo
{
Host 245.3.4.5
Port 389
Administrator "cn=radius,ou=components,o=operators,o=umc"
Password      "radius"
SearchBase    "retailerName=SP,o=users,o=umc"
Authenticate  search
}
    Directory virneo-backup
{
Host 245.3.4.6
Port 389
Administrator "cn=radius,ou=components,o=operators,o=umc"
Password      "radius"
SearchBase    "retailerName=SP,o=users,o=umc"
Authenticate  search
```

After the installation of Merit AAA from the SRC software distribution, the NULL realm is enabled by default.

Configuring RADIUS Profiles with the LDAP Directory

RADIUS servers search objects from the type umcRadiusPerson to authenticate incoming PPP sessions. If RADIUS and JUNOSe-specific attributes must be returned to the JUNOSe router during the authentication process, Merit AAA expects some special AAA attributes:

- aaaReply—A response sent back from the server (for example, a session time limit)
- aaaCheck—An attribute that must be present in the user entry for the entry to evaluate as True
- aaaDeny—An attribute that must NOT be present in the user entry for the entry to evaluate as True

These attributes are multivalued attributes containing the RADIUS attribute value pairs to be processed by the Merit AAA server.

The following depicts a umcRadiusPerson object that returns the RADIUS attribute values for Session-Timeout, Idle-Timeout, and Class, and the JUNOSe-specific attribute for the virtual router to be used on the JUNOSe router. This entry is shown in Lightweight Data Interchange Format (LDIF) notation:

```
dn:serviceName=bras,uniqueID=jane,ou=local,retailerName=isp1,
o=Users,o=umc
objectClass: umcRadiusPerson
objectClass: umcServiceProfile
objectClass: top
uid: jane
userPassword: secret
serviceName: bras1
usedService: serviceName=bras,o=Services,o=umc
aaaReply: Virtual-Router-Name=Default
aaaReply: Class=1,uid,bras
aaaReply: Idle-Timeout=2700
aaaReply: Session-Timeout=10800
```

Example: Merit AAA Accounting Log File Format

The following is an example of an accounting log file generated by Merit AAA with:

- Some accounting activity coming from the JUNOSe RADIUS client (tracking the activity of a PPP session)
- Some accounting activity coming from the SDX RADIUS client (a video service being activated, then deactivated)

NOTE: The Merit AAA server that we supply supports interim accounting by default.

```
Tue May  1 10:58:42 2001
    Acct-Status-Type = Start
    User-Name = "user1@isp1"
    Event-Time = "May  1 2001"
    Acct-Delay-Time = 0
```

```
    NAS-Identifier = "OBIWAN"
    Acct-Session-Id = "erx fastEthernet 3/1::0000022073"
    NAS-IP-Address = 10.227.9.145
    Service-Type = Framed
    Framed-Protocol = PPP
    Framed-IP-Address = 10.227.9.150
    Framed-IP-Netmask = 255.255.255.255
    Framed-Compression = None
    NAS-Port-Type = 15
    NAS-Port = 822083584
    NAS-Port-Id = "fastEthernet 3/1:"
    Ingress-Policy-Name = "unlim"
    Acct-Authentic = RADIUS
    User-Id = "user1"
    User-Realm = "isp1"

Tue May  1 10:59:49 2001
    Acct-Status-Type = Start
    Acct-Delay-Time = 0
    User-Name = "user1@isp1"
    Acct-Session-Id = "sspServiceVideoG:user1:e634da23b6"
    NAS-Identifier = "SSP.lion"
    User-Id = "user1"
    User-Realm = "isp1"

Tue May  1 11:07:25 2001
    Acct-Status-Type = Stop
    Acct-Delay-Time = 0
    User-Name = "user1"
    Acct-Session-Id = "sspServiceVideoG:user1:e634da23b6"
    Acct-Input-Octets = 10681
    Acct-Input-Gigawords = 0
    Acct-Input-Packets = 94
    Acct-Output-Octets = 0
    Acct-Output-Gigawords = 0
    Acct-Output-Packets = 0
    Acct-Session-Time = 456
    NAS-Identifier = "SSP"
    User-Id = "user1"
    User-Realm = ""
    LAS-Start-Time = 988729189
    LAS-Code = LAS-Notlocal
    LAS-Duration = 456
```

Configuring the Merit AAA Server and RADIUS Clients

For the Merit AAA server and RADIUS clients (JUNOSe router and the SAE software) to communicate, you must configure both the client and the server.

Configuring the Merit AAA Server

The RADIUS server must be able to communicate with the RADIUS clients. The following information about all RADIUS clients connected to the RADIUS server must be known to the RADIUS server:

- IP address of the RADIUS client
- RADIUS shared secret to be exchanged between Merit AAA and the client
- Model (vendor) of the RADIUS client

Configure this information by editing the */opt/UMC/radius/etc/clients* file. The client file should look like the following:

```
#Client Name         Key             [type]       [version]        [prefix]
#---------------- -------------- --------------- -------          --------
# SSP Client         192.23.3.10    secret       type=Juniper:NAS    v1
# Juniper ERX node (Enable the Juniper extensions)
                     192.23.3.1     secret       type=Juniper:NAS    v1
```

Configuring RADIUS Clients

Each RADIUS client must be able to contact its RADIUS server. The following information is required for client/server communication:

- IP address of the RADIUS server
- RADIUS shared secret to be exchanged between the Merit AAA server and the client
- UDP ports on which the client sends and receives RADIUS authentication and accounting packets. They must match the server configuration.

The RADIUS client configuration of the JUNOSe router is described in the *JUNOSe Broadband Access Configuration Guide*.

The RADIUS client configuration of the SAE is described in the *SRC-PE Getting Started Guide*.

Testing the Merit AAA Server

The Merit AAA installation from the SRC packages provides a script called *tstrad* for testing the RADIUS setup. This script uses the Merit test tool **radpwtst**. The test script is located in the directory */opt/UMC/radius*. To test the Merit AAA configuration, change to the directory */opt/UMC/radius,* and start the program by typing:

./tstrad <username> <userPassword>

where

< username > —Specifies the string identifying the user during the PPP login; for example, jane@isp1.com

< userPassword > —Specifies the user password submitted during the PPP login; for example, ./tstrad jane@isp1.com secret

If the customer did not install the Merit software from the SRC package, use the **radpwst** tool for testing the Merit AAA configuration by typing:

radpwtst -d <conf directory> -p <auth port> -s <server name> -u <auth type> -x -w < userPassword > <username>

where

- -d—Directory of users, clients, authfile, dictionary, etc.
- -p—Port number on which to listen for authentication requests
- -s—IP address or fully qualified DNS name of the server hosting Merit AAA
- -u—Authentication type; always use ppp
- -x—Allows the user to turn on debugging output
- -w—Allows the user to provide a password on the command line and not be prompted

The following example accomplishes the same as the **tstrad** script:

```
radpwst -d /opt/UMC/radius/etc -p 1812 -s 'hostname' -u ppp -x  -w secret
jane@virneo.com
```