SIP Express Router RADIUS HOWTO

Jan Janak

SIP Express Router RADIUS HOWTO
by Jan Janak

Revision History
Revision $Revision: 1.5 $ $Date: 2003/09/23 18:56:40 $

Table of Contents

List of Examples

Chapter 1. Introduction

SIP Express Router can be configured to use RADIUS server for authentication, accounting, and group membership checking. Since configuration of RADIUS seems to be a common source of problems, we decided to put together this howto.

The howto covers installation and configuration of FreeRADIUS server only. There are also other RADIUS servers available and as long as they support digest authentication, they should work too. Any volunteers willing to describe setup of other RADIUS servers are encouraged to contact the author.

1.1. Prerequisities

To setup RADIUS support in SIP Express Router you will need the following:

- FreeRADIUS server, you can get it from FreeRADIUS website (http://www.freeradius.org). The howto describes installation and setup of release 0.9.1.
- Radiusclient library, you can get it from http://www.mcs.de/~lf/radius. The howto describes installation and setup of version 0.3.2.
- SIP Express Router, get it from http://iptel.org/ser
- You should also have some experience in configuring SIP Express Router. Before you enable RADIUS authentication or accounting make sure that the basic server is running and that you know how to customize it to your taste.
- If you want to use RADIUS accounting then you will have to compile SIP Express Router from sources so you should know how to do it.

Various unix/linux distributions might include binary packages of the mentioned applications. In that case you can safely use the packages, there shouldn’t be any problem. Location of some files may be different, though. We will describe how to install the software from sources only.

Warning

Configuration of FreeRADIUS server described in the document is in no way exhaustive. This document is a sort of quick-start-guide, it shows how to get things running, but you should definitely read FreeRADIUS documentation and configure the server properly ! You have been warned.

Chapter 2. Radiusclient Library

Untar the source tarball.

```
root@localhost:/usr/local/src# tar xvfz radiusclient-0.3.2.tar.gz
```

Compile and install the library.

```
root@localhost:/usr/local/src# cd radiusclient-0.3.2
root@localhost:/usr/local/src/radiusclient-0.3.2# ./configure
root@localhost:/usr/local/src/radiusclient-0.3.2# make
root@localhost:/usr/local/src/radiusclient-0.3.2# make install
```

By default all the configuration files of the radiusclient library will be in `/usr/local/etc/radiusclient` directory.

If you use binary packages then the configuration files will be probably in `/etc/radiusclient`.

2.1. File `radiusclient.conf`

The main configuration file of the library is `/usr/local/etc/radiusclient/radiusclient.conf`, open the file in your favourite text editor and find lines containing the following:

```
authserver      localhost
```

This is the hostname or IP address of the RADIUS server used for authentication. You will have to change this unless the server is running on the same host as your SIP proxy.

```
acctserver      localhost
```

This is the hostname or IP address of the RADIUS server used for accounting. You will have to change this unless the server is running on the same host as your SIP proxy.

2.2. File `servers`

RADIUS protocol uses simple access control mechanism based on shared secrets that allows RADIUS servers to limit access from RADIUS clients. A RADIUS server is configured with a secret string and only RADIUS clients that have the same secret will be accepted.

You need to configure a shared secret for each server you have configured in `radiusclient.conf` file in the previous step. The shared secrets are stored in `/usr/local/etc/radiusclient/servers` file.

Each line contains hostname of a RADIUS server and shared secret used in communication with that server. The two values are separated by whitespaces. Configure shared secrets for every RADIUS server you are going to use.

Warning

RADIUS servers and clients must be configured with the same shared secret, otherwise they will not accept RADIUS messages from each other and neither authentication nor accounting will work !

2.3. File `dictionary`

SIP Express Router uses some attributes that are not included in the dictionary of radiusclient library, therefore it is necesarry to add them. Unfortunatelly the dictionary file used by SIP Express Router is not included in the source tarball of SIP Express Router, so far it is in unstable branch of the CVS (http://iptel.org/ser/cvs) only, but it will be included in one of future releases. Meanwhile you can get it from the CVS web interface (http://cvs.berlios.de/cgi-bin/viewcvs.cgi/ser/sip_router/etc/dictionary.ser?rev=HEAD).

Download the file and put it into `/usr/local/etc/radiusclient` directory and then append it to the main dictionary file:

```
root@localhost:/usr/local/etc/radiusclient# cat dictionary.ser >> dictionary
```

That will append contents of `dictionary.ser` file to `dictionary` file which contains the main dictionary for the radiusclient library.

Chapter 3. FreeRADIUS Server

Untar, configure, build, and install the server:

```
root@localhost:/usr/local/src# tar xvfz freeradius-0.9.1.tar.gz
root@localhost:/usr/local/src# cd freeradius-0.9.1
root@localhost"/usr/local/src/freeradius-0.9.1# ./configure
root@localhost"/usr/local/src/freeradius-0.9.1# make
root@localhost"/usr/local/src/freeradius-0.9.1# make install
```

All the configuration files of FreeRADIUS server will be in `/usr/local/etc/raddb` directory. If you install a binary package then you will probably find them in `/etc/raddb`.

The following sections describe how to configure freeradius server. First we describe the common configuration that must be done in any case. Configuration specific for authentication, accounting, and group membership checking will be described in separate sections.

3.1. Common configuration

3.1.1. File `clients.conf`

File `/usr/local/etc/raddb/clients.conf` contains description of RADIUS clients that are allowed to use the server. For each of the clients you need to specify it’s hostname or IP address and also a shared secret. The shared secret must be the same string you configured in radiusclient library.

Suppose that your SIP server is running on host proxy.foo.bar and radiusclient library on that machine has been configure with “foobarsecret” as the shared secret. You need to put the following section into the file:

```
client proxy.foo.bar {
    secret = foobarsecret
    shortname = foo
}
```

This fragment allows access from RADIUS clients on proxy.foo.bar if they use “foobarsecret” as the shared secret.

> **Note:** The file already contains an entry for localhost (127.0.0.1), so if you are running the RADIUS server on the same host as your SIP server, then modify the existing entry instead. By default it contains shared secret “testing123”.

3.1.2. File `dictionary`

File `/usr/local/etc/raddb/dictionary` contains the dictionary of FreeRADIUS server. You have to add the same dictionary file (`dictionary.ser`), which you added to the dictionary of radiusclient library, also here. In this case you don't have to append the contents of the file, you can include it into the main file. Add the following line at the end of `/usr/local/etc/raddb/dictionary`:

```
$INCLUDE /usr/local/etc/radiusclient/dictionary.ser
```

That will include the same attribute definitions that are used in radiusclient library so the client and server will understand each other.

3.1.3. File `radiusd.conf`

Digest authentication is disabled by default and you must enable it in this file. There are two sections, "authorize" and "authenticate". Both sections contain line containing word "digest". Both of them are commented and you must un-comment them to enable digest authentication.

> **Note:** There is also another line containing word "digest" followed by curly braces and it is enabled by default. The section is supposed to contain digest module parameters but because digest module has no parameters, it is empty. This is not the line you are supposed to uncomment ! There are two more.

3.1.4. File `users`

This file contains authentication information for each user. For testing purposes we will create user "test". Put the following into the file:

```
test Auth-Type := Digest, User-Password == "test"
     Reply-Message = "Hello, test with digest"
```

The username and password is for testing only, you can safely remove the entry once your RADIUS server works and you are able to authenticate.

3.2. Test The Server

> **Note:** This step is optional.

The basic configuration of FreeRADIUS server is done it now we are going to test if it really works. Start the server with parameter -X. That will cause the server to stay in the foreground (it will not turn into daemon) and produce a lot of debuging information on the standard output:

```
root@/usr/local/src# radiusd -X
```

Create file `digest` and put the following into the file:

```
User-Name = "test", Digest-Response = "631d6d73147add2f9e437f59bbc3aeb7",
Digest-Realm = "testrealm", Digest-Nonce = "1234abcd" ,
Digest-Method = "INVITE", Digest-URI = "sip:5555551212@example.com",
Digest-Algorithm = "MD5", Digest-User-Name = "test"
```

All the attributes must be on a single line.

Run **radclient** to test the server:

```
root@/usr/local/src# radclient -f digest localhost auth <shared_secret>
```

> **Note:** I suppose that you run the test utility directly on the RADIUS server since it comes with the FreeRADIUS server package. That also means that you have to enable access from localhost in your `clients.conf` file. Don't forget to replace <shared_secret> with the shared secret configured for locahost clients in `clients.conf`.

If your server works properly then you should see the following response:

```
Received response ID 224, code 2, length = 45
        Reply-Message = "Hello, test with digest"
```

3.3. Authentication Configuration

To create user "joe" in domain "iptel.org" with password "heslo" put the following into file `/usr/local/etc/raddb/users`:

```
joe@iptel.org Auth-Type := Digest, User-Password == "heslo"
     Reply-Message = "Authenticated",
     Sip-Rpid = "1234"
```

Attribute "Sip-Rpid" is optional. The attribute contains a phone number associated to the user. SIP Express Router can be configured to put the phone number into Remote-Party-ID header field of the SIP message. The header field can be then used by PSTN gateways to display the number as the number of the caller on regular phones. You can omit the attribute if you don't need it.

3.4. Accounting Configuration

By default FreeRADIUS server will log all accounting requests into `/usr/local/var/log/radius/radacct` directory in form of plain text files. The server will create one file for each hostname in the directory. The following example shows how the log files look like.

Example 3-1. Example of Accounting Report

```
Tue Jun 24 00:20:55 2003
        Acct-Status-Type = Start
        Service-Type = 15
        Sip-Response-Code = 200
        Sip-Method = 1
        User-Name = "gh@192.168.2.16"
        Calling-Station-Id = "sip:gh@192.168.2.16"
        Called-Station-Id = "sip:jiri@192.168.2.16"
        Sip-Translated-Request-URI = "sip:jiri@192.168.2.36"
        Acct-Session-Id = "b9a2ffaa-0458-42e1-b5fd-59656b795d29@192.168.2.32"
        Sip-To-Tag = "cb2cfe2e-3659-28c7-a8cc-ab0b8cbd3012"
        Sip-From-Tag = "a783bd2f-bb8d-46fd-84a9-00a9833f189e"
        Sip-CSeq = "1"
        NAS-IP-Address = 192.168.2.16
        NAS-Port = 5060
        Acct-Delay-Time = 0
        Client-IP-Address = 127.0.0.1
        Acct-Unique-Session-Id = "9b323e6b2f5b0f33"
        Timestamp = 1056406855

Tue Jun 24 00:20:56 2003
        Acct-Status-Type = Stop
        Service-Type = 15
        Sip-Response-Code = 200
        Sip-Method = 8
        User-Name = "jiri@192.168.2.16"
        Calling-Station-Id = "sip:jiri@192.168.2.16"
        Called-Station-Id = "sip:gh@192.168.2.16"
        Sip-Translated-Request-URI = "sip:192.168.2.32:9576"
        Acct-Session-Id = "b9a2ffaa-0458-42e1-b5fd-59656b795d29@192.168.2.32"
        Sip-To-Tag = "a783bd2f-bb8d-46fd-84a9-00a9833f189e"
        Sip-From-Tag = "cb2cfe2e-3659-28c7-a8cc-ab0b8cbd3012"
        Sip-CSeq = "4580"
        NAS-IP-Address = 192.168.2.16
        NAS-Port = 5060
        Acct-Delay-Time = 0
        Client-IP-Address = 127.0.0.1
        Acct-Unique-Session-Id = "b2c2479a07b17c95"
        Timestamp = 1056406856
```

3.5. Group Checking Configuration

If you want to make user “joe” in domain “iptel.org” member of group “pstn” then add the following to your `/usr/local/etc/raddb/users` file:

```
joe@iptel.org Sip-Group == "pstn", Auth-Type := Accept
        Reply-Message = "Authorized"
```

Chapter 4. SIP Express Router Configuration

We will describe installation from sources here. If you use binary packages then there is an additional package containg RADIUS related modules. You will need to install the package.

> **Warning**
>
> Due to a mistake the binary packages for RADIUS do not include RADIUS-enabled version of acc (accounting) module. The packages contain modules for RADIUS authentication and group membership checking only.
>
> If you need accounting over RADIUS then you will have to compile RADIUS-enabled version of acc module from the sources. This will be fixed in one of future releases, we apologize for any incovenience.

RADIUS-related modules are not compiled by default. To compile them, edit `Makefile`, find variable `exclude_modules` and you should see “auth_radius”, “group_radius”, and “uri_radius” among excluded modules. Simply remove the three modules from the list.

If you need RADIUS accounting then edit also sip_router/modules/acc/Makefile and uncomment lines containing:

```
DEFS+=-DRAD_ACC
LIBS=-L$(LOCALBASE)/lib -lradiusclient
```

Then recompile and re-install SIP Express Router:

```
root@localhost:/usr/local/src/sip_router# make proper
root@localhost:/usr/local/src/sip_router# make all
root@localhost:/usr/local/src/sip_router# make install
```

4.1. Authentication Configuration

Edit configuration file of SIP Express Router and instead of `auth_db.so` load `auth_radius.so` and also replace `www_authorize` with `radius_www_authorize`.

> **Note:** `radius_www_authorize` takes just one parameter (as opposed to `www_authorize` which takes 2).

4.2. Accounting Configuration

To enable RADIUS accounting simply use `radius_log_flag` and `radius_log_missed_flag` parameters instead of `log_flag` and `log_missed_flag`. Mark transactions that should be logged with flags configured in the parameters.

4.3. Group Membership Checking

Instead of `group.so` load `group_radius.so`. The module exports the same functions as `group.so`, the only difference is that all the function names exported by `group_radius.so` have “radius_” prefix.

Chapter 5. Frequently Asked Questions

1. I compiled SIP Express Router RADIUS modules and installed radiusclient library, but when I try to start ser I get the following error message:

```
libradiusclient.so.0: cannot open shared object file: No such file or directory
```

Make sure that the directory which contains the library (usually `/usr/local/lib`) is listed in `/etc/ld.so.conf` and run **ldconfig -v** (as root).

2. I configured everything as described in this HOWTO, but I get the following message from radiusclient librarary “ check_radius_reply: received invalid reply digest from RADIUS server”. What does that mean ?

That means that radiusclient library was unable to verify digest of the RADIUS message (it is not related to SIP digest) because shared secret of the client and server do not match.

> **Note:** FreeRADIUS server has two files that can contain definitions of clients and corresponding shared secrets--`clients` and `clients.conf`.
>
> If you have proper shared secret in one file and you still get the mentioned error message then check also the other file. This can easily happen to clients running on the same host (127.0.0.1 or localhost), because `clients.conf` contains definition for localhost by default with secret “testing123”.