

## **BAB IV**

### **IMPLENTASI DAN PENGUJIAN**

Asterisk merupakan paket aplikasi server yang dapat digunakan untuk membuat PC menjadi IP Private Branch Exchange (PBX). Sedangkan FreeRADIUS adalah paket aplikasi server yang digunakan sebagai *Authorization, Authentication dan Accounting* (AAA) terhadap *user* Asterisk. Pada bab ini akan dijelaskan tentang pembangunan *PBX Server Asterisk dan Server FreeRADIUS*, prosedur operasi dan pengujian yang mangacu pada desain perancangan pada BAB III. Dengan melakukan serangkaian uji coba maka akan diketahui apakah aplikasi ini bisa berjalan dengan baik.

Server Asterisk PBX dan Server FreeRADIUS bekerja pada sistem operasi Linux berbasis kernel 2.6.xx. Versi Linux yang akan di gunakan adalah Fedora 12 Constantine dan menggunakan protokol SIP. Untuk melakukan implementasi dengan baik, kita harus menyiapkan sebuah PC atau Notebook yang akan digunakan sebagai server dan beberapa telepon genggam sebagai *Mobile Client*.

#### **4.1. IMPLEMENTASI**

Pada tahapan implementasi terdapat empat cakupan yaitu konfigurasi pada sisi *server*, konfigurasi pada sisi *client*, pengujian pada sisi *server*, pengujian pada sisi *client* dan pengujian sistem.

Untuk membangun Sistem Keamanan Rumah ini, dibagi dua bagian yang penting yaitu yang pertama adalah Hardware dan yang kedua Software. berikut ini

adalah Spesifikasi Hardware dan Software yang penulis gunakan akan dijelaskan secara umum rancangan - rancangan yang diperlukan.

#### **4.1.1. Instalasi Server Asterisk PBX**

Pada instalasi Server ada dua hal yang harus dipersiapkan yaitu pada sisi *hardware* dan *software*. *Hardware* yang perlu dipersiapkan adalah sebuah PC atau *Notebook* dengan spesifikasi minimum PIII 500Mhz dengan kapasitas RAM 384 dan menggunakan *harddisk* 10Gb yang akan kita gunakan sebagai *PBX Server Asterisk*. Software yang perlu dipersiapkan adalah *Operating System* Linux Fedora 12 Constantine, Aplikasi PBX Server Asterisk dan FreeRADIUS Server. Pada kesempatan ini penulis tidak membahas tentang bagaimana cara instalasi *Operating System* Linux Fedora 12 Constantine tetapi lebih langsung membahas cara instalasi Aplikasi PBX Server Asterisk dan FreeRADIUS Server.

Unduh program asterisk dan modul pendukung di **situs <http://www.asterisk.org>**, lalu letakkan program asterisk dan program pendukung lainnya di **/usr/local/src/**

##### **a. Install asterisk-1.4.9**

Ekstrak paket asterisk-1.4.9 dengan utilitas tar

```
[root@localhost src]# tar xfvz asterisk-1.4.9.tar.gz
```

setelah diekstrak akan terbentuk direktori asterisk, lalu masuk ke direktori asterisk-1.4.9

```
[root@localhost src]# cd asterisk-1.4.9
```

Lakukan patching terhadap file radius-v1.0a.patch

```
[root@localhost asterisk-1.4.9]# patch -p1 < ../radius-
v1.0a.patch
```

Lakukan perintah configure

```
[root@localhost asterisk-1.4.9]# ./configure
```

Jalankan perintah make menuselect untuk memastikan rest\_auth terpilih pada paket asterisk yang akan diinstall.

```
[root@localhost asterisk-1.4.9]# make menuselect
```

lalu kompilasi asterisk dan install asterisk dengan perintah

```
[root@localhost asterisk-1.4.9]# make && make install
```

pasang contoh file konfigurasi dengan perintah

```
[root@localhost asterisk-1.4.9]# make samples.
```

#### **b. Install asterisk-sound-1.2.1**

Ekstrak paket asterisk-sound-1.2.1 dengan utilitas tar

```
[root@localhost src]# tar xvfz asterisk-sound-1.2.1.tar.gz
```

setelah diekstrak akan terbentuk direktori asterisk, lalu masuk ke direktori asterisk-sound-1.2.1

```
[root@localhost src]# cd asterisk-sound-1.2.1
```

instalasi paket yang dengan perintah

```
[root@localhost asterisk-sound-1.2.1]# make install
```

#### **c. Install mpg123-0.62**

Ekstrak paket asterisk dengan utilitas tar

```
[root@localhost src]# tar xvfz mpg123-0.62.tar.gz
```

setelah diekstrak akan terbentuk direktori mpg123-0.62, lalu masuk ke direktori mpg123-0.62

```
[root@localhost src]# cd mpg123-0.62
```

lalu konfigurasi file tersebut dengan perintah

```
[root@localhost mpg123-0.62]# ./configure
```

lalu kompilasi mpg123-0.62 dengan perintah

```
[root@localhost mpg123-0.62]# make
```

instalasi paket yang sudah dikompilasi dengan perintah

```
[root@localhost mpg123-0.62]# make install
```

#### 4.1.2. Instalasi Server FreeRADIUS

Sebelum melakukan instalasi server FreeRADIUS penulis harus melakukan instalasi RADIUS *client API* agar FreeRADIUS dapat berkomunikasi dengan Asterisk PBX. Pada penelitian ini penulis menggunakan *radiusclient-ng*. *Radiusclient-ng* dipilih karena mudah dalam instalasi dan berfungsi dengan baik.

Unduh RADIUS client API *radiusclient-ng-0.5.2.tar.gz* dari <http://developer.berlios.de/projects/radiusclient-ng/> kemudian letakan di direktori **/usr/local/src/** kemudian jalankan perintah sebagai berikut :

Extract *radiusclient-ng-0.5.2.tar.gz* dengan utilitas tar

```
[root@localhost src]# tar xvfz radiusclient-ng-0.5.2.tar.gz
```

Masuk kedirektori *radiusclient-ng-0.5.2*

```
[root@localhost src]# cd radiusclient-ng-0.5.2
```

Jalankan perintah *./configure*

```
[root@localhost radiusclient-ng-0.5.2]# ./configure
```

Kemudian lakukan kompilasi pada paket *radiusclient-ng-0.5.2*

```
[root@localhost radiusclient-ng-0.5.2]# make
```

Lakukan instalasi dengan perintah make install

```
[root@localhost radiusclient-ng-0.5.2]# make install
```

Radiusclient-ng sudah berhasil diinstal, semua file radiusclient-ng ada di direktori **/usr/local/etc/radiusclient-ng**.

Setelah *Radiusclient-ng* berhasil di instal berikutnya adalah instalasi Server FreeRADIUS. Penulis mengunduh file FreeRADIUS dari **www.freeradius.org** kemudian melakukan instalasi secara manual. Versi FreeRADIUS yang digunakan adalah FreeRADIUS-1.1.1 yang merupakan versi stabil dan bisa digabungkan dengan Server Asterisk PBX.

Unduh file freeradius-1.1.1.tar.gz letakan pada direktori **/usr/local/src/** kemudian jalankan perintah sebagai berikut :

Extract file freeradius-1.1.1.tar.gz dengan utilitas tar

```
[root@localhost src]# tar xvfz freeradius-1.1.1.tar.gz
```

Masuk kedalam direktori freeradius cd freeradius-1.1.1

```
[root@localhost src]# cd freeradius-1.1.1
```

Jalankan perintah ./configure terhadap file freeradius

```
[root@localhost freeradius-1.1.1]# ./configure
```

Lakukan kompilasi terhadap freeradius

```
[root@localhost freeradius-1.1.1]# make
```

Jalankan perintah instalasi freeradius

```
[root@localhost freeradius-1.1.1]# make install
```

Instalasi Server FreeRADIUS berhasil dilakukan, semua file konfigurasi FreeRADIUS ada di direktori **/usr/local/etc/raddb/**.

#### 4.1.3. Instalasi DHCP Server pada Server Asterisk PBX

Agar Server Asterisk PBX dapat melakukan alokasi alamat IP kepada *client* secara dinamis maka perlu dilakukan instalasi DHCP-server (*Dynamic Host Configuration Protocol*). DHCP-server yang digunakan adalah dengan menggunakan *DHCP-Server bind9* ke dalam Server Asterisk PBX dengan cara seperti gambar 4-1.

```
[root@localhost]# yum install bind9
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages were automatically installed and
are no longer required:
    libtonezone1
Use 'apt-get autoremove' to remove them.
The following extra packages will be installed:
    bind9-host bind9utils dnstools libbind9-50 libdns50
libdns53 libisc50
    libisccc50 libiscfg50 liblwres50
Suggested packages:
    bind9-doc resolvconf rblcheck
The following NEW packages will be installed:
    bind9 bind9utils libdns53
The following packages will be upgraded:
    bind9-host dnstools libbind9-50 libdns50 libisc50
```

```

libisccc50 libisccfg50

  liblwres50

8 upgraded, 3 newly installed, 0 to remove and 170 not
upgraded.

Need to get 1,502kB of archives.

After this operation, 1,221kB of additional disk space
will be used.

Do you want to continue [Y/n]? Y

Done.

Adding system user `bind' (UID 114) ...
Adding new user `bind' (UID 114) with group `bind' ...
Not creating home directory `/var/cache/bind'.
wrote key file "/etc/bind/rndc.key"

#

* Starting domain name service... bind9

[ OK ]

```

Gambar 4-1 Proses Instalasi DHCP-Server pada Server Asterisk PBX

Setelah proses instalasi DHCP server selesai, selanjutnya adalah melakukan konfigurasi server DHCP. Konfigurasi ini dilakukan agar setiap *client* yang terhubung dengan server mendapatkan alokasi alamat IP secara dinamis. Proses konfigurasi DHCP server sangat sederhana yaitu dengan cara di bawah ini :

- a. Sunting file `dhcp3-server` dengan cara *nano* `/etc/default/dhcp3-server` dan pilih *interface* mana yang

berfungsi untuk memberikan IP DHCP. Misal *interface eth1*.

Maka rubah interfacenya menjadi *INTERFACES=eth1*

- b. Tambahkan konfigurasi pada file *dhcpd.conf* dengan cara *nano /etc/dhcp3/dhcpd.conf*
- c. Misalkan kita ingin membuat IP DHCP dengan range 10.1.1.1 sampai dengan 10.1.1.200, dengan gateway NIC lain 10.1.1.1, dan DNS yaitu 192.168.1.200, maka konfigurasinya pada file *dhcpd.conf* adalah :

```

default-lease-time 600;
max-lease-time 7200;
option subnet-mask 255.255.255.0;
option broadcast-address 10.1.1.255;
option routers 10.1.1.1;
option domain-name-servers 192.168.1.200;
option domain-name domainname.com;
subnet 10.1.1.0 netmask 255.255.255.0 {
range 10.1.1.1 10.1.1.200;
}

```

- d. Lakukan hal di bawah ini pada sisi *client* (Misal interface di *client* adalah *eth1* maka konfigurasi di *interface* ini adalah dengan cara *nano /etc/network/interfaces*. Lakukan konfigurasi :

```

auto lo eth0
iface eth0 inet dhcp

```



iface lo inet loopback

- e. Setelah konfigurasi selesai maka cara untuk mengaktifkan DHCP-Server adalah pada CLI Linux menggunakan perintah `/etc/init.d/dhcp3-server start` dan akan tampil seperti pada gambar 4-2.

```
[root@localhost]# /etc/init.d/dhcp3-server restart
* Stopping DHCP server dhcpd3           [ OK ]
* Starting DHCP server dhcpd3           [ OK ]
```

Gambar 4-2 Mengaktifkan DHCP-Server pada Server Asterisk PBX

#### 4.1.4. Konfigurasi Server Asterisk PBX

Setelah semua aplikasi server terinstal dan kelengkapannya, langkah berikutnya adalah konfigurasi Server Asterisk PBX. Sunting file `sip.conf` menggunakan perintah `vi` seperti berikut:

```
[root@localhost]# vi /etc/asterisk/sip.conf
```

Kemudian aktifkan `codec gsm`, `g729` dan `ullaw` dengan cara menambahkan potongan script berikut kedalam file **sip.conf**.

```
allow=ullaw
```

```
allow=g729
```

```
allow=gsm
```

Langkah selanjutnya adalah isi real domain dengan `localhost` karena server Asterisk berada di `localhost`. Konfigurasi server Asterisk di lanjutkan dengan membuat `user` Asterisk dengan autentikasi server FreeRADIUS. Penulis membuat beberapa `user` Asterisk seperti terlihat pada contoh berikut :

```
[trimo]  
type=friend  
context=clientradius  
username=trimo  
fromdomain=localhost  
host=dynamic  
secret=radius:
```

```
[akiun]  
type=friend  
context=clientradius  
username=akiun  
fromdomain=localhost  
host=dynamic  
secret=radius :
```

Penjelasan:

- a. [trimo] merupakan *context user*, ini akan di pakai pada file *extension.conf* untuk pengaturan nomor *extension*.
- b. *Type = friend* merupakan salah satu tipe *client*.
- c. *Context = default* merupakan context jaringan. Ini juga akan di pakai pada file *extension.conf* untuk merencanakan dial plan, dan masing-masing *context = default* menunjukkan bahwa *context* yang ditunjukkan adalah *context* standar. Konfigurasi tiap *context* bisa dibuat berbeda-beda karena pada perancangan

*dial plan* hanya akan ada satu tipe *dial plan*, maka cukup di buat satu context saja yaitu *default*.

d. *Host = dynamic* menyatakan bahwa *user* dalam blok ini alamat IP nya dapat berubah-ubah / *dynamic*.

e. *Username = trimo*, merupakan username dari context [trimo].

f. *Secret=radius*: ini menunjukkan bahwa *user* akan melakukan otentikasi ke server FreeRADIUS.

Konfigurasi *dial plan* dilakukan dengan menyunting file */etc/asterisk/extension.conf*, konfigurasi di sini dilakukan agar tiap *client* yang terdaftar dapat melakukan proses berkomunikasi antar *client*. Konfigurasi *dial plan* sesuai dengan *user* yang terdapat pada file *sip.conf*.

```
[clientradius]
```

```
exten => 555,1,Playback(demo-echodone)
```

```
exten => 101,1,Dial(SIP/trimo)
```

```
exten => 102,1,Dial(SIP/akiun)
```

Penjelasan :

a. *Exten => 555* merupakan nomor panggilan untuk *demo-echodone* sesuai dengan bawaan server asterisk.

b. *Exten => 101* merupakan nomor panggilan untuk *user trimo*

c. *Exten => 102* merupakan nomor panggilan untuk *user akiun*.

#### 4.1.5. Konfigurasi Server FreeRADIUS

Konfigurasi server FreeRADIUS akan dilakukan penulis dengan merubah beberapa file yang merupakan *default setting* dari aplikasi FreeRADIUS yang telah diinstal pada sub bab 4.1.2. Berikut ini akan dijelaskan bagaimana cara melakukan konfigurasi tersebut.

Autentikasi yang akan di gunakan dalam server FreeRADIUS ini adalah dengan metode *digest*. Untuk mengaktifkan autentikasi dengan metode *digest*, sunting file */usr/local/etc/raddb/radiusd.conf* dan hilangkan tanda komentar (#) pada baris *digest* pada bagian *authenticate* dan *accounting*. Dengan melakukan hal diatas maka autentikasi dengan metode *digest* sudah aktif.

Langkah berikutnya adalah menyunting file */usr/local/etc/raddb/clients.conf* dan menyamakan nilai *secret* agar sesuai dengan nilai *secret* pada file *radiusclient-ng/server*. Setelah nilai *secret* pada kedua file sudah sama maka konfigurasi FreeRADIUS mendekati tahap akhir.

Sunting file */usr/local/etc/raddb/users* dan tambahkan *user* VoIP yang sebelumnya sudah penulis *definisikan* dalam file */etc/asterisk/sip.conf*. Metode autentikasi yang digunakan adalah *digest* yang merupakan kombinasi dari *user:realm:password*. Berikut contoh *user* dan *password* yang dibuat dalam penelitian ini.

```
trimo Digest-HA1 := "0f7f8f797b275cdb5f69fd17bf8c5b4b"
```

```
akiun Digest-HA1 := "429dc76f499369366300d9731744d625"
```

Penjelasan :

- a) *trimo* dan *akiun* merupakan *user* yang sebelumnya sudah di buat pada file */etc/asterisk/sip.conf*
- b) *Digest-HA1* merupakan metode autentikasi yang digunakan yaitu *digest*.
- c) *"0f7f8f797b275cdb5f69fd17bf8c5b4b"* merupakan contoh password yang dibentuk dengan metode enkripsi MD5. Kombinasi password ini di bentuk dengan menggunakan perintah *:r !echo -n 'trimo:localhost:tr1m0' |md5sum*.

Langkah terakhir dalam konfigurasi server FreeRADIUS adalah menyamakan konfigurasi di server FreeRADIUS dengan *client* FreeRADIUS. Sunting file */usr/local/etc/radiusclient-ng/radiusclient.conf* kemudian aktifkan *authserver* dan *acctserver*. Karena server Asterisk dan server FreeRADIUS berada dalam satu PC maka penulis mengarahkan *authserver* dan *acctserver* ke arah *localhost*, seperti terlihat pada contoh berikut :

```
authserver      localhost
acctserver      localhost
```

Sunting file */usr/local/etc/radiusclient-ng/server* tambahkan baris *script* seperti contoh berikut :

```
#Server Name or Client/Server pair      Key
#-----
#portmaster.elemental.net              hardlyascret
#portmaster2.elemental.net             donttellanyone
localhost                               testing123
```

Nilai key yang dideskripsikan dalam contoh ini sama dengan nilai key pada file `/usr/local/etc/raddb/radiusd.conf`.

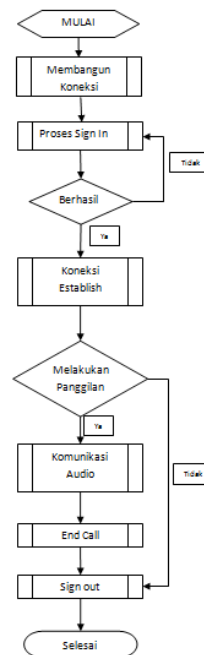
Tambahkan file yang sudah di sunting dan sesuai dengan konfigurasi yang diharapkan kepada `dictionary.sip` dengan cara :

```
[root@localhost asterisk-1.6.0.25]#cat >>dictionary
$INCLUDE dictionary.sip.
```

Sampai disini sudah selesai proses konfigurasi yang dilakukan terhadap server Asterisk maupun server FreeRADIUS. Langkah selanjutnya yaitu konfigurasi pada `client`.

#### **4.1.6. Implementasi dan Konfigurasi pada Sisi Client**

Aplikasi `client` digunakan untuk proses registrasi, konfigurasi jaringan, melakukan panggilan maupun menerima panggilan serta informasi-informasi komunikasi yang ada seperti informasi user, waktu panggilan dan informasi lainnya. Pada dasarnya `client` akan mengerjakan beberapa proses yang mendasar yaitu membangun koneksi, verifikasi dan pengiriman pesan. Proses kerja `client` secara sederhana dapat dilihat dari diagram alur pada gambar 4-3.



Gambar 4-3 Diagram Alur Sistem Aplikasi Client

#### a. *Instalasi Aplikasi pada Client*

Aplikasi yang digunakan pada sebuah mobile phone seperti Linphone, X-lite Twinkle, SJPhone, Siphone atau yang lainnya untuk dapat melakukan panggilan antar extension. Dalam hal ini terdapat dua jenis sistem operasi yang digunakan oleh Mobile client yaitu *Windows mobile 5* dan *Mac OS for mobile phone* dengan menggunakan *softphone x-lite dan Siphone*. Proses instalasi softphone pada Windows Mobile 5 relatif lebih mudah karena menggunakan aplikasi tambahan semacam *PC-Suite*, sedangkan untuk aplikasi *client* yang digunakan *Mac OS for mobile phone* sedikit lebih susah karena harus langsung diinstall dari *Mobile Phone* tersebut.

Cara melakukan instalasi Softphone di *Mac OS for mobile phone* adalah:

- a) Hidupkan iPhone dan masuk ke menu installer seperti pada gambar 4-4.



Gambar 4-4 Proses Instalasi Aplikasi Softphone Client

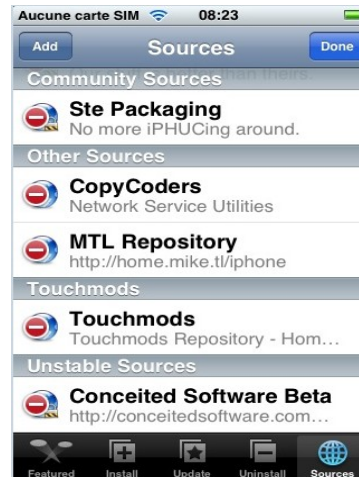
- b) Tambahkan sumber aplikasi dengan memilih menu edit seperti pada gambar 4-5.



Gambar 4-5 List Source Aplikasi Softphone Client

- c) Tambahkan source baru dengan memilih tombol add seperti pada gambar 4-6.





Gambar 4-6 Menambahkan Source Aplikasi Softphone Client

- d) Sisipkan <http://sip.free.fr/rep.xml> pada text box dan klik OK seperti pada gambar 4-7.



Gambar 4-7 Menambahkan Source Aplikasi Softphone Client

- e) Setelah ditambahkan sumber aplikasi siphone, kemudian ke menu installer untuk melakukan proses instalasi aplikasi siphone dan setelah itu keluar dari menu instalasi dengan menekan tombol *Home* dan reset mobile phone yang telah terinstall aplikasi softphone tersebut. Setelah melakukan reset mobile phone maka akan muncul aplikasi baru yang telah terinstall seperti pada gambar 4-8.

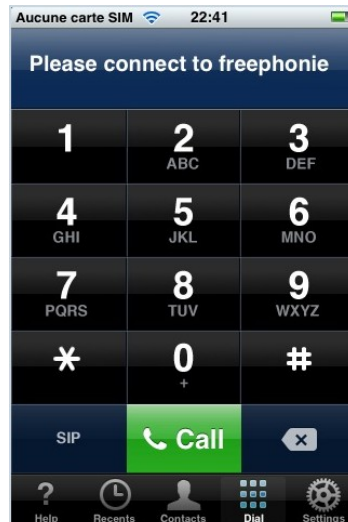


Gambar 4-8 Aplikasi Softphone Sudah Berhasil Diinstall

### b. Konfigurasi Aplikasi Client

Setelah melakukan instalasi aplikasi softphone pada perangkat mobile phone, langkah selanjutnya adalah melakukan konfigurasi pada perangkat tersebut agar bisa melakukan registrasi ke Server Asterisk PBX dan melakukan autentikasi ke Server FreeRADIUS. Langkah-langkah konfigurasi aplikasi softphone pada sisi client adalah sebagai berikut :

- a) Buka aplikasi softphone dan lakukan konfigurasi agar bisa melakukan registrasi ke Server Asterisk PBX dan melakukan autentikasi ke server FreeRADIUS seperti terlihat pada gambar 4-9.



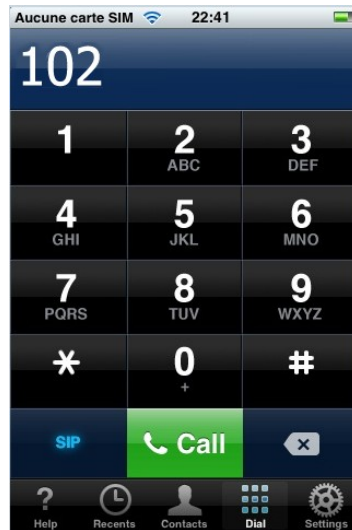
Gambar 4-9 Konfigurasi Aplikasi Softphone Client

- b) Dari aplikasi Siphone masuk ke konfigurasi Siphone dan tambahkan Username : trimo, Password : tr1m0, SIP Server: 10.1.1.5 kemudian konfigurasi direkam seperti gambar 4-10.



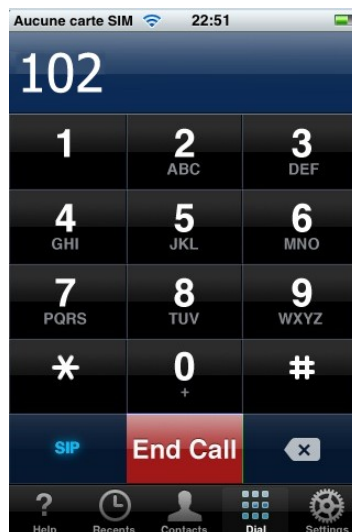
Gambar 4-10 Konfigurasi Aplikasi SoftphoneClient

- c) Setelah konfigurasi selesai maka Softphone Siphone siap untuk digunakan. Penulis melakukan percobaan test telepon ke no extension 102 atau user akiun seperti terlihat pada gambar 4-11.



Gambar 4-11 Melakukan Panggilan dengan Siphone

- d) Untuk mengakhiri panggilan klik tombol end call seperti pada gambar 4-12.



Gambar 4-12 Mengakhiri Panggilan pada Siphone

## 4.2. PENGUJIAN

Untuk membuktikan bahwa sistem yang telah terinstal Server Asterisk PBX dan Server FreeRADIUS berjalan dengan baik maka perlu dilakukan serangkaian pengujian. Berikut ini pengujian yang penulis lakukan pada sistem yang sudah dibangun.

### 4.2.1. Pengujian Server Asterisk PBX

Cara menjalankan Server Asterisk PBX yang sudah ada dalam sistem, masuk ke terminal Linux kemudian ketikkan perintah :

```
[root@localhost asterisk]# safe_asterisk
```

maka server asterisk sudah berjalan dengan baik. Untuk melihat Server Asterisk PBX dalam mode debug jalankan perintah :

```
[root@localhost asterisk]# asterisk -r
```

Mode debug Server Asterisk PBX terlihat seperti pada gambar 4-13.

```

File Edit View Terminal Help
e causes include: the remote application did not send a reply, the message bus s
ecurity policy blocked the reply, the reply timeout expired, or the network conn
ection was broken.)
GConf Error: Failed to contact configuration server; some possible causes are th
at you need to enable TCP/IP networking for ORBit, or you have stale NFS locks d
ue to a system crash. See http://projects.gnome.org/gconf/ for information. (Det
ails - 1: Failed to get connection to session: Did not receive a reply. Possibl
e causes include: the remote application did not send a reply, the message bus
security policy blocked the reply, the reply timeout expired, or the network conn
ection was broken.)
[root@localhost asterisk]# safe_asterisk
[root@localhost asterisk]# asterisk -r
Asterisk 1.4.9, Copyright (C) 1999 - 2007 Digium, Inc. and others.
Created by Mark Spencer <markster@digium.com>
Asterisk comes with ABSOLUTELY NO WARRANTY; type 'core show warranty' for detail
s.
This is free software, with components licensed under the GNU General Public
License version 2 and other licenses; you are welcome to redistribute it under
certain conditions. Type 'core show license' for details.
=====
Connected to Asterisk 1.4.9 currently running on localhost (pid = 16877)
Verbosity is at least 3
-- Remote UNIX connection
localhost*CLI>

```

Gambar 4-13 Server Asterisk PBX dalam Mode Debug

#### 4.2.2. Pengujian Server FreeRADIUS

Setelah Server Asterisk PBX dipastikan berjalan dengan baik, pengujian berikutnya adalah menjalankan Server FreeRADIUS dengan perintah berikut ini.

```
[root@localhost raddb]# radiusd start
```

Untuk melihat Server FreeRADIUS dalam mode debug jalankan perintah

```
[root@localhost asterisk]# radiusd -X
```

Server FreeRADIUS siap untuk menerima perintah (*Ready to process requests*) terlihat seperti pada gambar 4-14.

```

Module: Instantiated files (files)
Module: Loaded Acct-Unique-Session-Id
  acct_unique: key = "User-Name, Acct-Session-Id, NAS-IP-Address, Client-IP-Address, NAS-Port"
Module: Instantiated acct_unique (acct_unique)
Module: Loaded detail
  detail: detailfile = "/usr/local/var/log/radius/radacct/{Client-IP-Address}/detail-%Y%m%d"
  detail: detailperm = 384
  detail: dirperm = 493
  detail: locking = no
Module: Instantiated detail (detail)
Module: Loaded radutmp
  radutmp: filename = "/usr/local/var/log/radius/radutmp"
  radutmp: username = "%{User-Name}"
  radutmp: case_sensitive = yes
  radutmp: check_with_nas = yes
  radutmp: perm = 384
  radutmp: callerid = yes
Module: Instantiated radutmp (radutmp)
Listening on authentication *:1812
Listening on accounting *:1813
Ready to process requests.

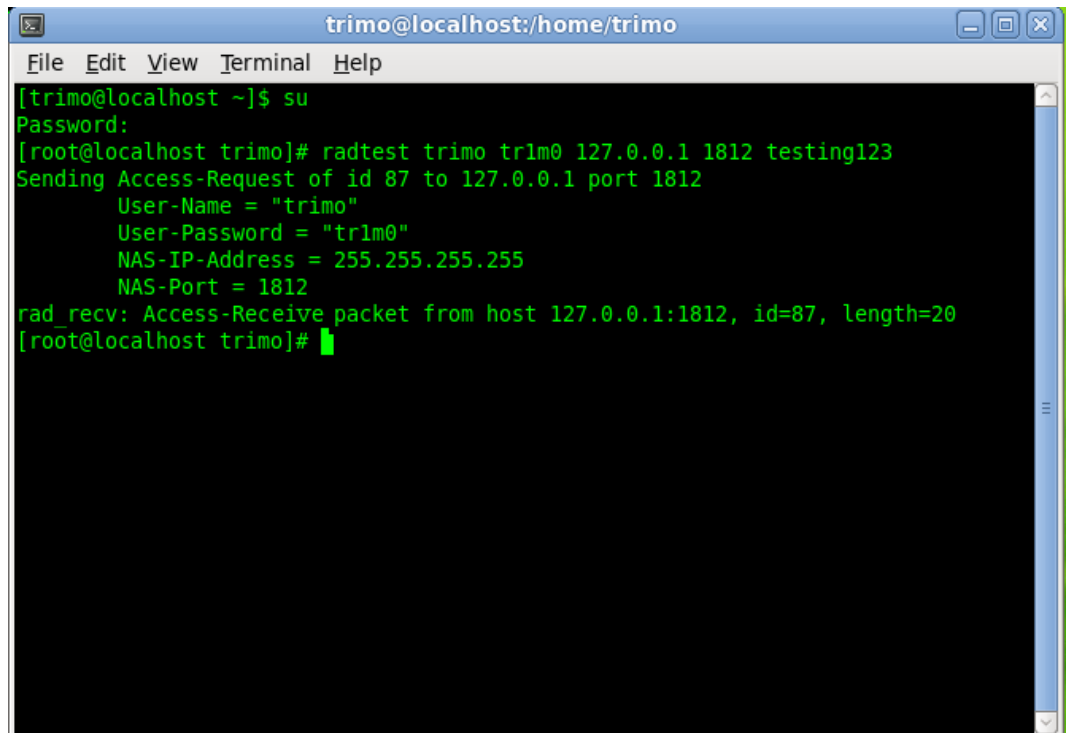
```

Gambar 4-14 Server FreeRADIUS dalam Mode Debug

Untuk memastikan Server FreeRADIUS bisa menerima perintah dari user penulis akan melakukan permintaan dari konsole Linux dengan perintah di bawah ini :

```
[root@localhost trimo]# radtest trimo tr1m0 127.0.0.1 1812
testing123
```

Server FreeRADIUS akan menerima permintaan dari user seperti terlihat pada gambar 4-15.



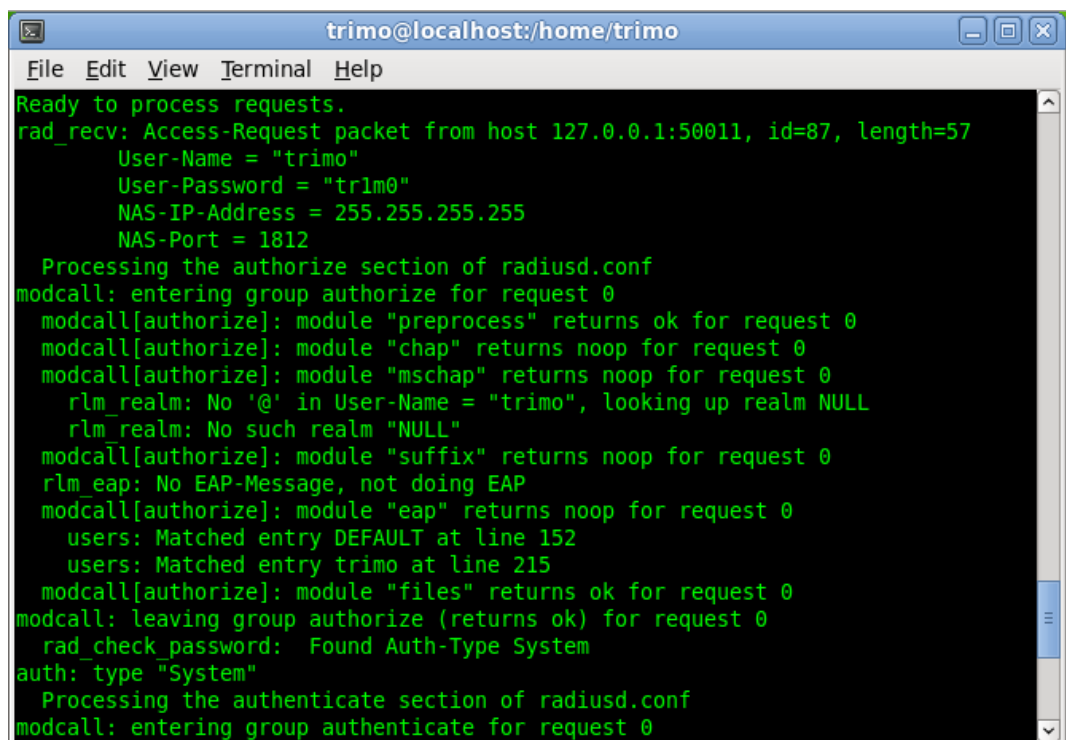
```

trimo@localhost:/home/trimo
File Edit View Terminal Help
[trimo@localhost ~]$ su
Password:
[root@localhost trimo]# radtest trimo tr1m0 127.0.0.1 1812 testing123
Sending Access-Request of id 87 to 127.0.0.1 port 1812
    User-Name = "trimo"
    User-Password = "tr1m0"
    NAS-IP-Address = 255.255.255.255
    NAS-Port = 1812
rad_recv: Access-Receive packet from host 127.0.0.1:1812, id=87, length=20
[root@localhost trimo]#

```

Gambar 4-15 Server FreeRADIUS Menerima Permintaan dari User

Pada saat melakukan permintaan diatas pada Server FreeRADIUS dalam mode debug akan terlihat seperti gambar 4-16.



```

trimo@localhost:/home/trimo
File Edit View Terminal Help
Ready to process requests.
rad_recv: Access-Request packet from host 127.0.0.1:50011, id=87, length=57
    User-Name = "trimo"
    User-Password = "tr1m0"
    NAS-IP-Address = 255.255.255.255
    NAS-Port = 1812
Processing the authorize section of radiusd.conf
modcall: entering group authorize for request 0
modcall[authorize]: module "preprocess" returns ok for request 0
modcall[authorize]: module "chap" returns noop for request 0
modcall[authorize]: module "mschap" returns noop for request 0
    rlm_realm: No '@' in User-Name = "trimo", looking up realm NULL
    rlm_realm: No such realm "NULL"
modcall[authorize]: module "suffix" returns noop for request 0
    rlm_eap: No EAP-Message, not doing EAP
modcall[authorize]: module "eap" returns noop for request 0
    users: Matched entry DEFAULT at line 152
    users: Matched entry trimo at line 215
modcall[authorize]: module "files" returns ok for request 0
modcall: leaving group authorize (returns ok) for request 0
rad_check_password: Found Auth-Type System
auth: type "System"
Processing the authenticate section of radiusd.conf
modcall: entering group authenticate for request 0

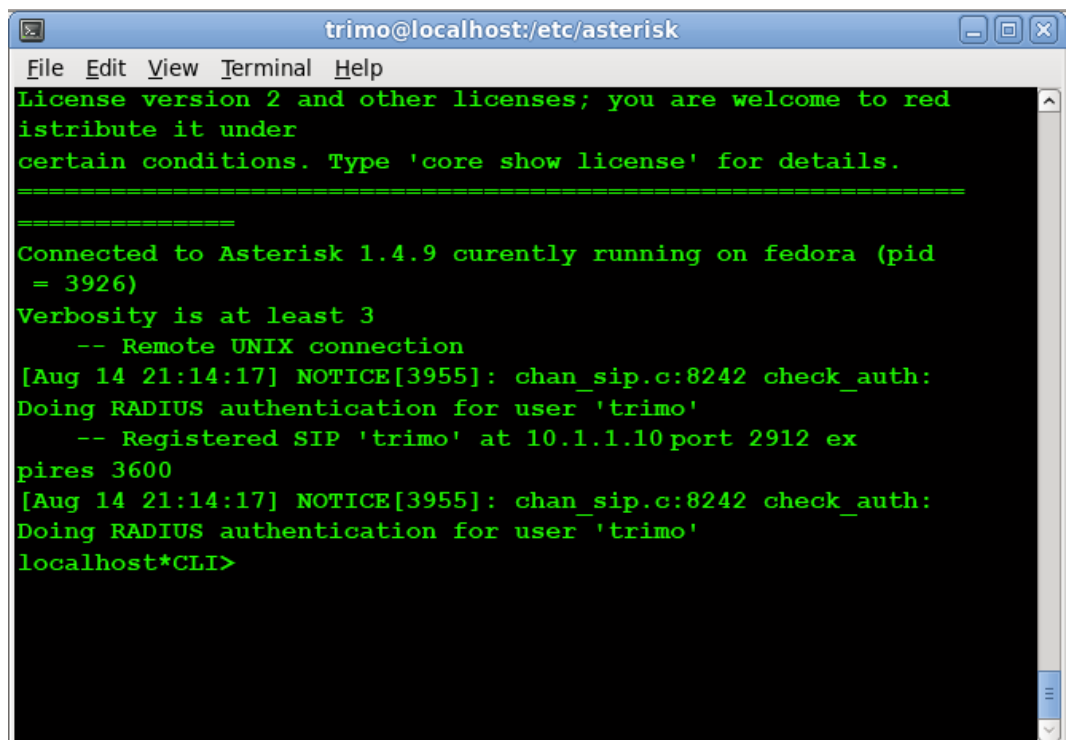
```

Gambar 4-16 Server FreeRADIUS dalam Mode Debug Menerima Permintaan User

### 4.2.3. Pengujian Server Asterisk PBX dengan Server FreeRADIUS

Server Asterisk PBX dan Server FreeRADIUS berada dalam satu mesin. Konfigurasi agar kedua Server dapat berkomunikasi sudah dijelaskan pada Sub Bab sebelumnya. Pada Sub Bab ini penulis akan fokus pada pengujiannya.

Pada saat client melakukan registrasi ke Server Asterisk PBX dan autentikasi ke Server FreeRADIUS pada mode debug pada kedua server akan terlihat seperti gambar 4-17.

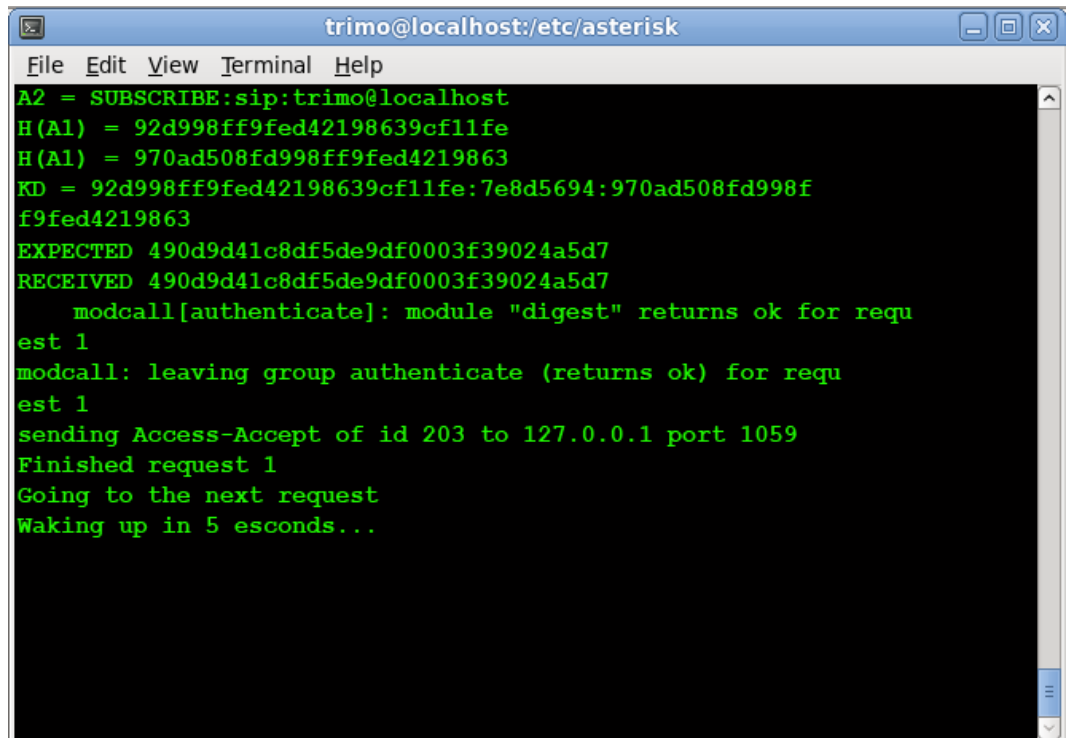
A screenshot of a terminal window titled 'trimo@localhost:/etc/asterisk'. The terminal displays the Asterisk CLI interface. It shows a license notice at the top, followed by connection information: 'Connected to Asterisk 1.4.9 currently running on fedora (pid = 3926)'. The verbosity is set to 3. The terminal shows two consecutive log messages: '[Aug 14 21:14:17] NOTICE[3955]: chan\_sip.c:8242 check\_auth: Doing RADIUS authentication for user 'trimo'' and '-- Registered SIP 'trimo' at 10.1.1.10 port 2912 expires 3600'. The prompt 'localhost\*CLI>' is visible at the bottom.

```
trimo@localhost:/etc/asterisk
File Edit View Terminal Help
License version 2 and other licenses; you are welcome to red
istribute it under
certain conditions. Type 'core show license' for details.
=====
Connected to Asterisk 1.4.9 curently running on fedora (pid
= 3926)
Verbosity is at least 3
-- Remote UNIX connection
[Aug 14 21:14:17] NOTICE[3955]: chan_sip.c:8242 check_auth:
Doing RADIUS authentication for user 'trimo'
-- Registered SIP 'trimo' at 10.1.1.10 port 2912 ex
pires 3600
[Aug 14 21:14:17] NOTICE[3955]: chan_sip.c:8242 check_auth:
Doing RADIUS authentication for user 'trimo'
localhost*CLI>
```

Gambar 4-17 Registrasi User pada Server Asterisk PBX

Pada mode debug FreeRADIUS akan terlihat seperti gambar 4-18 saat user VoIP melakukan autentikasi ke Server FreeRADIUS.





```

trimo@localhost:/etc/asterisk
File Edit View Terminal Help
A2 = SUBSCRIBE:sip:trimo@localhost
H(A1) = 92d998ff9fed42198639cf11fe
H(A1) = 970ad508fd998ff9fed4219863
KD = 92d998ff9fed42198639cf11fe:7e8d5694:970ad508fd998f
f9fed4219863
EXPECTED 490d9d41c8df5de9df0003f39024a5d7
RECEIVED 490d9d41c8df5de9df0003f39024a5d7
  modcall[authenticate]: module "digest" returns ok for requ
est 1
modcall: leaving group authenticate (returns ok) for requ
est 1
sending Access-Accept of id 203 to 127.0.0.1 port 1059
Finished request 1
Going to the next request
Waking up in 5 esconds...

```

Gambar 4-18 Autentikasi User VoIP pada Server FreeRADIUS

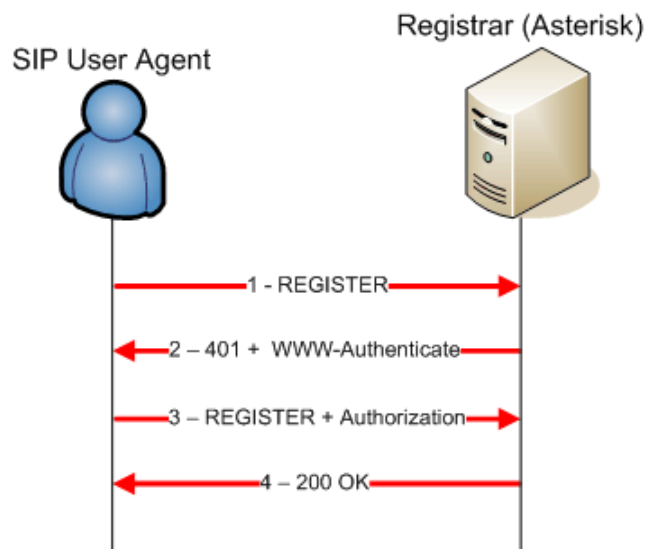
Pada pengujian Server Asterisk PBX dan Server FreeRADIUS terlihat bahwa autentikasi user VoIP di Server FreeRADIUS berhasil dilakukan.

#### 4.2.4. Pengujian Client

Pengujian Client terbagi menjadi dua bagian yaitu pengujian client pada Server Asterisk PBX tanpa autentikasi Server FreeRADIUS dan pengujian client pada Server Asterisk PBX dengan autentikasi Server FreeRADIUS.

**a. Pengujian Client Server Asterisk PBX tanpa Autentikasi Server FreeRADIUS**

Pengujian ini dilakukan terhadap Server Asterisk PBX tanpa Autentikasi Server FreeRADIUS. Pada pengujian ini user VoIP akan melakukan registrasi kepada Server Asterisk seperti terlihat pada gambar 4-19.



Gambar 4-19 Registrasi User VoIP ke Server Asterisk PBX

Penjelasan :

1. SIP User Agent melakukan registrasi ke Server Asterisk
2. Server Asterisk akan mengembalikan permintaan dengan kode 401 atau membutuhkan autentikasi.
3. SIP User akan melakukan registrasi lagi berikut autentikasi ke Server Asterisk.
4. Server Asterisk menerima permintaan dan mengembalikan dengan kode 200 atau berhasil.

Pada Server Asterisk PBX pada saat ada user melakukan registrasi akan terlihat seperti pada gambar 4-20.

```

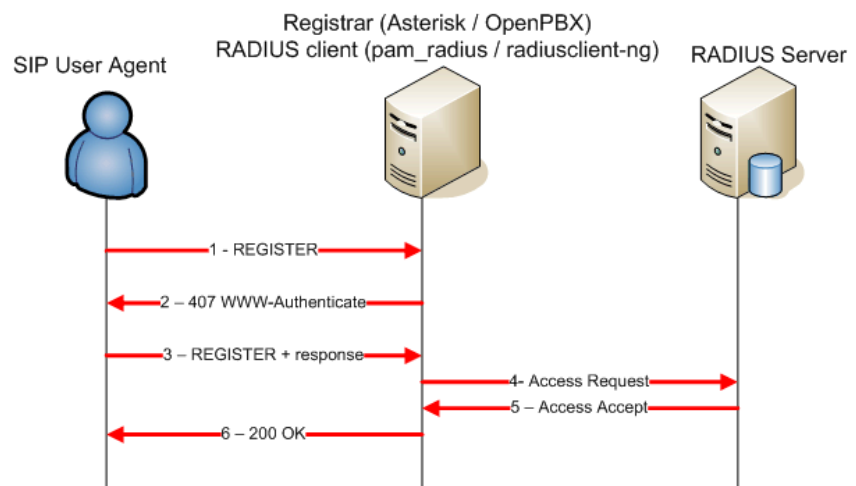
-- Registered SIP 'trimo' at 10.1.1.10 port 2842
-- Registered SIP 'akiun' at 10.1.1.10 port 24308
localhost*CLI>

```

Gambar 4-20 Registrasi User ke Server Asterisk PBX

## b. Pengujian Client Server Asterisk PBX dengan Autentikasi Server FreeRADIUS

Pengujian ini dilakukan terhadap Server Asterisk PBX dengan autentikasi Server FreeRADIUS atau seperti terlihat pada gambar 4-21.



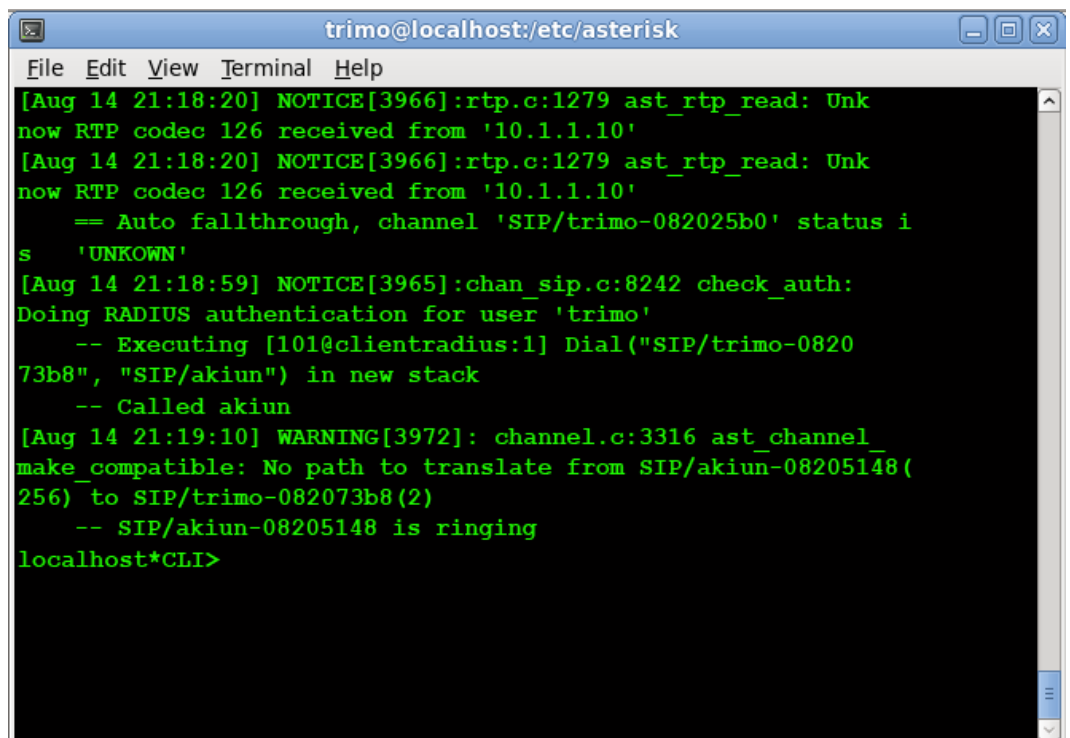
Gambar 4-21 Registrasi User VoIP dengan Autentikasi Server FreeRADIUS

Penjelasan :

1. SIP User Agent melakukan registrasi ke server Asterisk
2. Server Asterisk mengembalikan permintaan 407 yang artinya membutuhkan autentikasi
3. SIP User Agent melakukan registrasi dengan autentikasi
4. Client API Radius melakukan permintaan ke Server FreeRADIUS
5. Server FreeRADIUS menerima permintaan

6. Server Asterisk menerima permintaan dan mengembalikan ke User Agent kode 200

Proses registrasi user Asterisk PBX dan Server FreeRADIUS bisa di lihat pada **Sub Bab 4.2.3**. Proses pengujian selanjutnya adalah user trimo akan melakukan panggilan pada user akiun. Berikut adalah gambar hasil pengujian yang terlihat pada Server Asterisk PBX.



```

trimo@localhost:/etc/asterisk
File Edit View Terminal Help
[Aug 14 21:18:20] NOTICE[3966]:rtp.c:1279 ast_rtp_read: Unk
now RTP codec 126 received from '10.1.1.10'
[Aug 14 21:18:20] NOTICE[3966]:rtp.c:1279 ast_rtp_read: Unk
now RTP codec 126 received from '10.1.1.10'
    == Auto fallthrough, channel 'SIP/trimo-082025b0' status is
s 'UNKOWN'
[Aug 14 21:18:59] NOTICE[3965]:chan_sip.c:8242 check_auth:
Doing RADIUS authentication for user 'trimo'
    -- Executing [101@clientradius:1] Dial("SIP/trimo-0820
73b8", "SIP/akiun") in new stack
    -- Called akiun
[Aug 14 21:19:10] WARNING[3972]: channel.c:3316 ast_channel_
make_compatible: No path to translate from SIP/akiun-08205148 (
256) to SIP/trimo-082073b8 (2)
    -- SIP/akiun-08205148 is ringing
localhost*CLI>

```

Gambar 4-22 User Trimo Melakukan Panggilan ke User Akiun di Server Asterisk PBX

Pada gambar di atas terlihat setiap aktifitas yang ada pada Server Asterisk PBX akan selalu meminta autentikasi kepada Server FreeRADIUS.

Gambar 4-23 adalah hasil pengujian yang terlihat pada Server FreeRADIUS.

```

trimo@localhost:/etc/asterisk
File Edit View Terminal Help
H(A1) = 92d998ff9fed42198639cf11fe
H(A1) = 970ad508fd998ff9fed4219863
KD = 92d998ff9fed42198639cf11fe:7e8d5694:970ad508fd998f
f9fed4219863
EXPECTED 490d9d41c8df5de9df0003f39024a5d7
RECEIVED 490d9d41c8df5de9df0003f39024a5d7
  modcall[authenticate]: module "digest" returns ok for requ
est 3
  modcall: leaving group authenticate (returns ok) for requ
est 3
Sending Access-Accept of id 205 to 127.0.0.1 port 1061
Finished request 1
Going to the next request
--- Walking the entire request list ---
Waking up in 5 esconds...

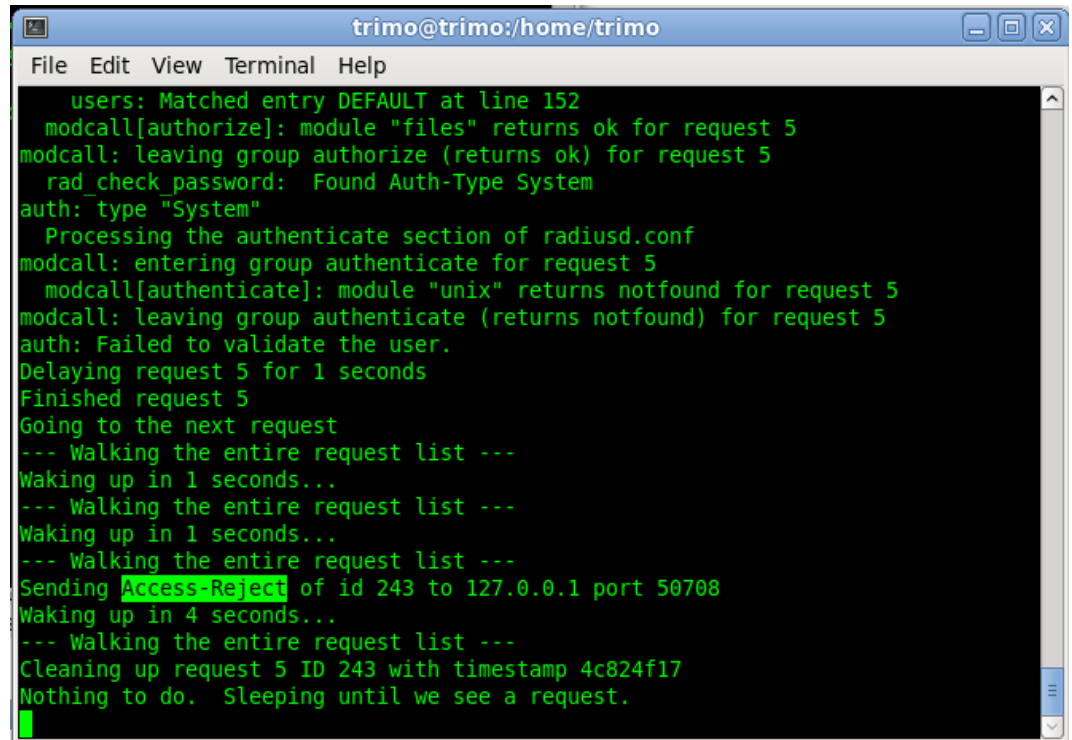
```

Gambar 4-23 User Trimo Melakukan Panggilan ke User Akiun di Server FreeRADIUS

Pada gambar di atas terlihat Server FreeRADIUS memberikan autentikasi kepada user trimo yang akan melakukan panggilan ke user akiun. Setelah kedua user VoIP saling berhubungan maka Server FreeRADIUS akan kembali pada posisi idle dan siap menerima permintaan selanjutnya.

#### 4.2.5. Pengujian Keamanan Proses Registrasi

Pengujian keamanan dilakukan terhadap Server Asterisk dan Server FreeRADIUS. Pengujian keamanan disini adalah penulis mencoba melakukan registrasi terhadap Server Asterisk dan Server FreeRADIUS dengan *user* yang tidak sah. Dari Proses pengujian ini akan diketahui sistem yang aman tidak akan menerima permintaan dari *user* yang tidak sah. Gambar 4-24 menunjukkan Server FreeRADIUS menolak permintaan dari *user* yang tidak sah.



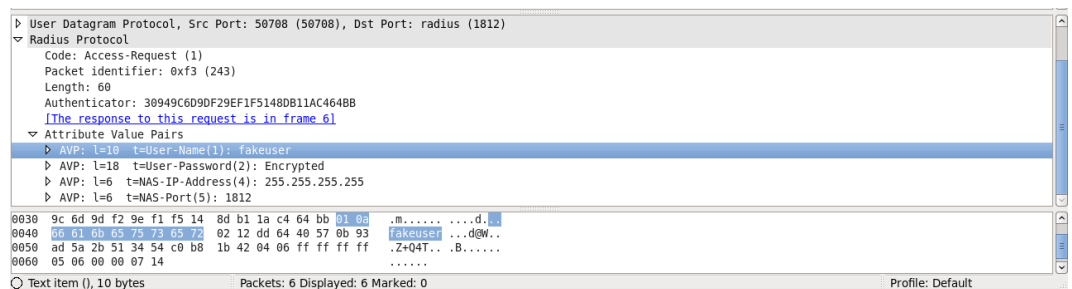
```

trimo@trimo:/home/trimo
File Edit View Terminal Help
users: Matched entry DEFAULT at line 152
modcall[authorize]: module "files" returns ok for request 5
modcall: leaving group authorize (returns ok) for request 5
rad_check_password: Found Auth-Type System
auth: type "System"
Processing the authenticate section of radiusd.conf
modcall: entering group authenticate for request 5
modcall[authenticate]: module "unix" returns notfound for request 5
modcall: leaving group authenticate (returns notfound) for request 5
auth: Failed to validate the user.
Delaying request 5 for 1 seconds
Finished request 5
Going to the next request
--- Walking the entire request list ---
Waking up in 1 seconds...
--- Walking the entire request list ---
Waking up in 1 seconds...
--- Walking the entire request list ---
Sending Access-Reject of id 243 to 127.0.0.1 port 50708
Waking up in 4 seconds...
--- Walking the entire request list ---
Cleaning up request 5 ID 243 with timestamp 4c824f17
Nothing to do. Sleeping until we see a request.

```

Gambar 4-24 FreeRADIUS Menolak Permintaan

Penulis menggunakan wireshark untuk melihat proses yang terjadi pada port 1812 atau protokol RADIUS. Gambar 4-25 adalah hasil *capture* pada wireshark, pada gambar berikut terlihat *fakeuser* melakukan permintaan terhadap Server FreeRADIUS.



```

User Datagram Protocol, Src Port: 50708 (50708), Dst Port: radius (1812)
RADIUS Protocol
Code: Access-Request (1)
Packet identifier: 0xf3 (243)
Length: 60
Authenticator: 30949C6D9DF29EF1F5148DB11AC464BB
[The response to this request is in frame 61]
Attribute Value Pairs
AVP: l=10 t=User-Name(1): fakeuser
AVP: l=18 t=User-Password(2): Encrypted
AVP: l=6 t=NAS-IP-Address(4): 255.255.255.255
AVP: l=6 t=NAS-Port(5): 1812
0030 9c 6d 9d f2 9e f1 f5 14 8d b1 1a c4 64 bb 01 0a .m.....d..
0040 66 61 6b 65 75 73 65 72 02 12 dd 64 40 57 0b 93 fakeuser...d@W..
0050 ad 5a 2b 51 34 54 c0 b8 1b 42 04 06 ff ff ff ff .Z+Q4T...B.....
0060 05 06 00 00 07 14 .....

```

Gambar 4-25 User Tidak Sah Melakukan Permintaan terhadap FreeRADIUS

Gambar 4-26 adalah hasil *capture* wireshark, pada gambar terlihat Server FreeRADIUS menolak permintaan dari *fakeuser*. Server FreeRADIUS hanya memberikan *authorization* dan *authentication* kepada *user* yang sudah terdaftar pada file `/usr/local/etc/raddb/users`.

```

Destination port: 50708 (50708)
Length: 28
Checksum: 0xfe2f [validation disabled]
  [Good Checksum: False]
  [Bad Checksum: False]
RADIUS Protocol
  Code: Access-Reject (3)
    Packet Identifier: 0xf3 (243)
    Length: 28
    Authenticator: 522470BBA3E610FE6264C14820688691
    [This is a response to a request in frame 51]
    [Time from request: 2.003088000 seconds]
0010 00 30 00 00 40 00 40 11 3c bb 7f 00 00 01 7f 00 .0..@.@.<.....
0020 00 01 07 14 c6 14 00 1c fe 2f 33 f3 00 14 52 24 ...../|...R$
0030 7d bb a3 e6 10 fe 62 64 c1 48 2d 68 86 91 }.....bd .H-h..

```

Gambar 4-26 Server FreeRADIUS Menolak Permintaan dari Fakeuser

#### 4.2.6. Analisa dan Perbandingan Server Asterisk PBX Tanpa dan Setelah Menggunakan Server FreeRADIUS

Dari serangkaian pengujian terhadap Server Asterisk PBX dan Server FreeRADIUS yang sudah penulis lakukan maka penulis bisa membandingkan kinerja Server Asterisk PBX tanpa dan setelah menggunakan Server FreeRADIUS. Perbandingan yang akan dibahas dalam Sub Bab ini adalah registrasi user, saat melakukan panggilan dan penyimpanan password.

##### a. Registrasi User

Dalam hal ini saat user melakukan registrasi ke Server Asterisk PBX sebelum menggunakan autentikasi Server FreeRADIUS dilakukan secara langsung tanpa melalui proses autentikasi. Proses registrasi tanpa autentikasi ini berpengaruh terhadap keamanan sistem VoIP itu sendiri. Bisa saja user yang tidak memiliki hak bisa melakukan register padahal Server Asterisk PBX tidak memberikan akses kepada user tersebut. Saat Asterisk menggunakan autentikasi Server FreeRADIUS setiap user yang akan menggunakan layanan Server Asterisk PBX harus melalui autentikasi FreeRADIUS. Jadi setiap user yang melakukan register adalah user yang sah.

b. ***Proses Komunikasi antar Client VoIP***

Saat user melakukan komunikasi dengan user lain pada Server Asterisk PBX sebelum menggunakan autentikasi Server FreeRADIUS ada kemungkinan proses komunikasi bisa dilihat oleh user lain yang tidak mempunyai hak. Saat menggunakan Server FreeRADIUS fungsi otorisasi berfungsi user yang tidak mempunyai otorisasi terhadap proses komunikasi itu tidak bisa melihat proses komunikasi.

c. **Penyimpanan Password**

User VoIP pada Server Asterisk PBX disimpan pada file */etc/asterisk/sip.conf* user dan password disimpan dalam bentuk *plaintext* sehingga sangat besar kemungkinan terjadi sniffing terhadap user VoIP karena user dan password tidak dienkripsi. Saat Server Asterisk PBX menggunakan autentikasi Server FreeRADIUS user VoIP di simpan dalam file */usr/local/etc/raddb/users* user dan password disimpan dalam bentuk enkripsi MD5. Metode autentikasi *digest* dalam format *user:realm:password* akan membuat sistem lebih aman karena user dan password sudah dienkripsi. Selain itu Server FreeRADIUS mengetahui dari domain mana saja user yang melakukan *request* terhadap Server FreeRADIUS.

Setelah melihat perbandingan registrasi user, saat melakukan panggilan dan penyimpanan password penulis dapat menarik kesimpulan bahwa menggunakan Server FreeRADIUS dapat meningkatkan keamanan



sistem VoIP yang sudah penulis bangun. Setiap user yang mendaftar dan melakukan komunikasi dalam sistem harus melalui proses autentikasi dan otorisasi sehingga user yang menggunakan service Asterisk PBX sudah pasti user yang sah. Cara penyimpanan password yang sudah dienkripsi meningkatkan keamanan pada sistem yang sudah dibangun karena metode enkripsi MD5 merupakan salah satu metode enkripsi yang terkenal keamanannya.