

BAB IV

PEMBAHASAN

4.1 Identifikasi Masalah

Captive portal adalah salah satu sistem keamanan *area hotspot* yang cara kerjanya meminta calon pengguna layanan *internet* untuk memasukkan informasi akun miliknya kemudian sistem akan mencocokkan dengan informasi akun yang terdapat didalam *database* dari *server Captive portal*. Proses permintaan dan pengiriman akun calon pengguna dilakukan oleh *Coovachilli*. Jika informasi akun yang diinputkan oleh tamu sama dengan informasi akun yang ada di dalam *database*, maka user akan dapat menggunakan *internet* di *area hotspot* tersebut. Salah satu faktor penting yang diperlukan untuk membangun suatu sistem keamanan *Captive portal* adalah pembuatan maupun proses permintan dan pengiriman user id yang dilakukan oleh *Coovachilli* kemudian diimplementasikan pada suatu Jaringan Komputer. Dengan adanya *Coovachilli* dan Jaringan Komputer maka kita bisa membuat suatu layanan captive portal yang saling terhubung antara beberapa client dengan server pada suatu topologi jaringan

Dalam tahap pembahasan ini yang dilakukan yaitu membuat *Coovachilli* dan Jaringan Komputer, yang sesuai dengan kriteria sistem keamanan *Captive portal* yang akan diterapkan pada *area hotspot* milik PT PERTAMINA (PERSERO)

4.1.1 Kondisi Awal Area Hotspot

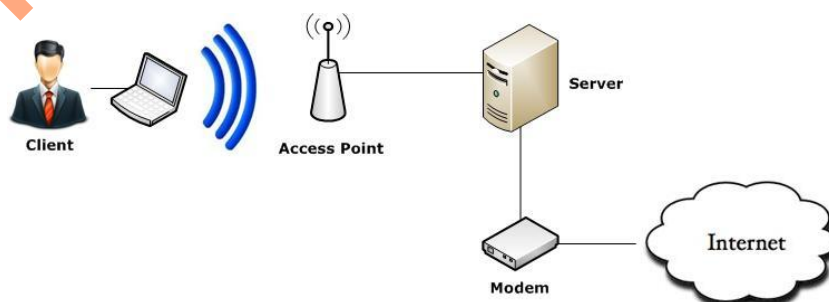
4.1.1.1 Sistem Keamanan Area Hotspot

Sistem keamanan yang digunakan pada *area hotspot* milik PT PERTAMINA (PERSERO) adalah WPA2 Enterprise. Sistem keamanan WPA2 Enterprise merupakan teknologi keamanan *hotspot* yang paling terbaru dan paling aman dibandingkan teknologi keamanan lainnya.

Namun, sistem keamanan ini memiliki kelemahan yaitu konfigurasi secara manual yang harus dilakukan oleh calon pengguna *internet* sebelum dapat menggunakan *internet* cukup rumit dan tidak praktis. Calon pengguna layanan *internet* harus membuat sebuah *network profile* dengan ketentuan – ketentuan yang sesuai dengan karakteristik keamanan WPA2 Enterprise dan proses pembuatan tersebut dilakukan calon pengguna secara manual.

4.1.1.2 Topologi Jaringan

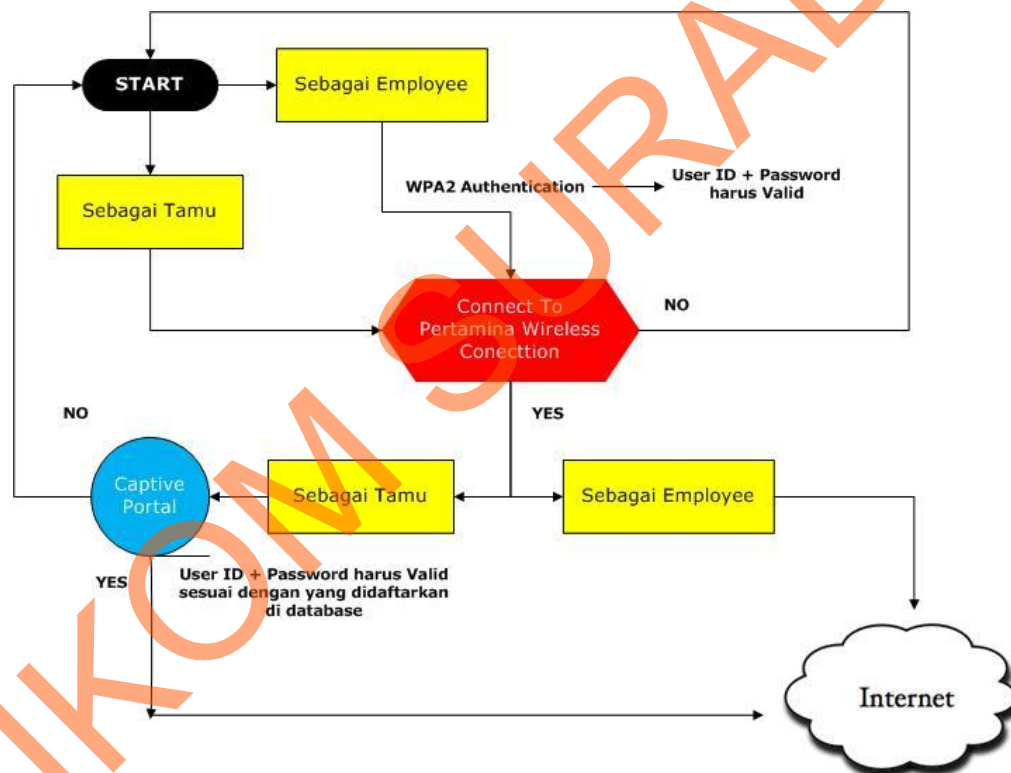
Untuk menerapkan sistem keamanan WPA2 Enterprise pada *area hotspot*, PT PERTAMINA (PERSERO) menggunakan topologi jaringan dengan susunan *access point* terhubung dengan *server* kemudian *server* terhubung langsung dengan modem seperti yang ditunjukkan pada Gambar 4.1.



Gambar 4.1 Topologi Area Hotspot

4.1.2 Perancangan Sistem Keamanan *Captive portal*

Untuk menyelesaikan permasalahan pada *area hotspot* milik PT PERTAMINA (PERSERO), maka sistem keamanan WPA2 Enterprise yang sudah diterapkan pada *area hotspot* tersebut akan diganti menggunakan sistem keamanan *Captive portal*. Namun, penggantian sistem keamanan ini hanya diterapkan untuk pengguna tamu saja. Untuk seluruh pekerja (*Employee*) di PT PERTAMINA (PERSERO) tetap menggunakan sistem keamanan WPA2 Enterprise. Penjelasan lebih lanjut ditampilkan pada Gambar 4.2.



Gambar 4.2 *Flowchart* Sistem Keamanan *Area Hotspot*

4.1.3 Kriteria Perangkat yang Diperlukan

Berdasarkan sistem keamanan *Captive portal* yang akan dibangun, maka untuk membangun *Coovachilli* dan Jaringan Komputer maka diperlukan spesifikasi komputer berikut ini :

Spesifikasi komputer

- Prosesor : Intel(R) Core(TM)2 Duo 64 bits
- RAM : 2 GB DDR3
- Harddisk : SATA Kapasitas 250 GB
- Ethernet card : LAN card on-board
- Sistem Operasi : Linux Ubuntu 13.04 Raring Ringtail

4.2 Pembahasan

Apabila semua kebutuhan yang mendukung pembuatan *captive portal* sudah terpenuhi, maka langkah selanjutnya adalah instalasi *packet*, konfigurasi *Coovachilli*, melakukan pengujian ip, dan terakhir menjalankan *Coovachilli*

Pada bagian instalasi nantinya akan melakukan instalasi terhadap beberapa *packet* yang dibutuhkan dalam pembuatan *captive portal*, misalkan *packet-packet* tersebut adalah *easyhotspot*, *freeradius*, *database*, dan juga *Coovachilli*.

Pada bagian konfigurasi nantinya akan dilakukan beberapa konfigurasi terhadap *Coovachilli*, konfigurasi tersebut meliputi mengaktifkan *Coovachilli*, merubah konfigurasi pada */etc/chilli/wwwsh*, */etc/chilli/up.sh*, */etc/chilli/default*, dan juga konfigurasi untuk menjalankan *Coovachilli*,

Pada bagian pengujian ip nantinya akan dicocokkan terhadap beberapa ip yang terhubung dengan sistem *captive portal*, ip tersebut nantinya akan melakukan proses *redirect* yang dilakukakn oleh *Coovachilli*, kesalahan yang sering terjadi ketika saat mensetting ip dari *modem* ke *server*, karena ip akan selalu berubah-ubah jika kita memasang / melepas *modem* dari *server*. jadi *admin* harus teliti dalam merubah ip tersebut.

Setelah kebutuhan untuk membuat *captive portal* sudah dijalankan, langkah selanjutnya ialah menjalankan *Coovachilli*. Apabila *Coovachilli* sudah aktif maka *Coovachilli* akan menjalankan tugasnya dengan meminta *user id* dan *password* dari *user* tersebut sebagai autentntikasi *user*, sehingga user harus memasukkan *user id* dan *password* sesuai dengan *user id* dan *password* yang berada di dalam database

4.2.1 Instalasi

Untuk membangun sebuah sistem keamanan *captive portal*, maka terlebih dahulu sudah terinstal dan sudah di konfigurasi beberapa paket ataupun kebutuhan yang nantinya diperlukan dalam membangun sebuah *captive portal*. Beberapa paket tersebut meliputi *easyhotspot*, *freeradius*, *database* maupun *LDAP Server*. karena nantinya data *user id* akan di kelola kedalam *database*.

Setalah semua paket terpasang kedalam *server* langkah selanjutnya yaitu membuat aplikasi *captive portal*, yang bertujuan untuk meminta *user id* dan *password* ketika ingin mengakses *internet*, kemudian mengirimkan data tersebut kedalam suatu *server* yang terdapat *database*, selanjutnya *database* akan

mencocokkan data dari *user* dengan data dari *database*. Apakah data tersebut sama atau tidak.

Pekerjaan selanjutnya adalah menginstall paket yang bernama *Coovachilli*. Namun paket ini sudah tidak tersedia di repo Ubuntu 13.04, jadi paket ini perlu di download secara manual. Untuk mendownloadnya, ikuti perintah – perintah di bawah ini :

1. wget http://ap.coova.org/chilli/coova-chilli_1.2.2_i386.deb
2. dpkg -i coova-chilli_1.2.2_i386.deb

4.2.2 Konfigurasi *Coovachilli*

Setelah semua paket sudah terinstall dan dikonfigurasi, maka langkah selanjutnya untuk membuat sistem *captive portal* yaitu dengan mengkonfigurasi *coovachilli*. *Coovachilli* nantinya akan difungsikan sebagai aplikasi yang akan mengirim user id dan password ke dalam *database*

1. Mengaktifkan *Coovachilli*

Secara default *Coovachilli* di setting dalam keadaan tidak aktif, maka untuk menjalankan *Coovachilli* harus mengaktifkannya terlebih dahulu dengan masuk kedalam direktori **/etc/default/chilli** dan merubah isi *file*

START_CHILLI=0 ubah menjadi **START_CHILLI=1**

2. Mendownload dan menginstall paket *Haserl*

Setelah paket *Coovachilli* sudah terinstall dan diaktifkan maka langkah selanjutnya untuk mengkonfigurasi *Coovachilli* yaitu menginstall

paket Haserl. Paket haserl berfungsi untuk konfigurasi www pada *Coovachilli*. Untuk mendownload dan menginstall paket haserl bisa menggunakan perintah

```
wget http://sourceforge.net/projects/haserl/files/haserl/0.8.0/haserl-0.8.0.tar.gz/download
tar -zxvf haserl-0.8.0.tar.gz
cd haserl-0.8.0;./configure;make;sudo make install
```

3. Merubah konfigurasi pada **/etc/chilli/wwwsh**

Mengkonfigurasi letak paket haserl yang sudah terinstal, dan paket tersebut diletakkan pada **/usr/local/bin**

```
haserl=$(which haserl 2>/dev/null)
ubah menjadi :
haserl=/usr/local/bin/haserl
```

4. Merubah konfigurasi pada **/etc/chilli/up.sh**

Langkah selanjutnya yaitu menambahkan beberapa baris berikut pada file **/etc/chilli/up.sh**

```
may not have been populated the first time; run again
[ -e "/var/run/chilli.iptables" ] && sh /var/run/chilli.iptables
2>/dev/null
# force-add the final rule necessary to fix routing tables
iptables -I POSTROUTING -t nat -o $HS_WANIF -j
MASQUERADE
```

5. Konfigurasi file `/etc/chilli/default`

Kemudian buka file `/etc/chilli/default` dan sesuaikan parameter sesuai setting-setting dari beberapa paket yang sudah terkonfigurasi, paket-paket tersebut yaitu *database*, *freeradius* maupun *easyhotspot*. berikut adalah setting `/etc/chilli/default` :

```
Coova-Chilli Default Configurations.

# To customize, copy this file to /etc/chilli/config
# and edit to your liking. This is included in shell scripts
# that configure chilli and related programs before file 'config'.
###

# Local Network Configurations
#
# HS_WANIF=eth0 # WAN Interface toward the Internet
HS_LANIF=eth1 # Subscriber Interface for client devices
HS_NETWORK=10.1.0.0 # HotSpot Network (must include
HS_UAMLISTEN)
HS_NETMASK=255.255.255.0 # HotSpot Network Netmask
HS_UAMLISTEN=10.1.0.1 # HotSpot IP Address (on subscriber
network)
HS_UAMPORT=3990 # HotSpot UAM Port (on subscriber
network)
HS_UAMUIPORT=4990 # HotSpot UAM "UI" Port (on
subscriber network, for embedded portal)

# HS_DYNIP=
```



```
# HS_DYNIP_MASK=255.255.255.0

# HS_STATIP=

# HS_STATIP_MASK=255.255.255.0

# HS_DNS_DOMAIN=

# DNS Servers
HS_DNS1=192.168.1.254

###

# HotSpot settings for simple Captive Portal

#

HS_NASID=nas01

HS_RADIUS=localhost

HS_RADIUS2=localhost

HS_UAMALLOW=10.1.0.1,192.168.1.254,lestari-net.co.cc

HS_RADSECRET=easyhotspot # Set to be your RADIUS shared
secret//sesuai yang kita isikan di /etc/freeradius/clients.conf

HS_UAMSECRET=easyhotspot # Set to be your UAM secret

HS_UAMALIASNAME=chilli

# Configure RADIUS proxy support (for 802.1x + captive portal
support)

# HS_RADPROXY=on

# HS_RADPROXY_LISTEN=127.0.0.1

# HS_RADPROXY_CLIENT=127.0.0.1

# HS_RADPROXY_PORT=1645

# HS_RADPROXY_SECRET=$HS_RADSECRET
```

```
# Example OpenWrt /etc/config/wireless entry for hostapd

# option encryption wpa2

# option server $HS_RADPROXY_LISTEN

# option port $HS_RADPROXY_PORT

# option key $HS_RADPROXY_SECRET

# To alternatively use a HTTP URL for AAA instead of
RADIUS:

# HS_UAMAAAURL=http://my-site/script.php

# Put entire domains in the walled-garden with DNS inspection
# HS_UAMDOMAINS=".paypal.com,.paypalobjects.com"

# Optional initial redirect and RADIUS settings

# HS_SSID=<ssid> # To send to the captive portal

# HS_NASMAC=<mac address> # To explicitly set Called-
Station-Id

# HS_NASIP=<ip address> # To explicitly set NAS-IP-Address

# The server to be used in combination with HS_UAMFORMAT
to

# create the final chilli 'uamserver' url configuration.

HS_UAMSERVER=$HS_UAMLISTEN

# Use HS_UAMFORMAT to define the actual captive portal url.

# Shell variable replacement takes place when evaluated, so here

# HS_UAMSERVER is escaped and later replaced by the pre-
defined
```

```
# HS_UAMSERVER to form the actual “-uamserver” option in
chilli.

HS_UAMFORMAT=http://\${HS_UAMLISTEN:\$HS_UAMUIPO
RT/www/login.chi

# Same principal goes for HS_UAMHOMEPAGE.

HS_UAMHOMEPAGE=http://\${HS_UAMLISTEN:\$HS_UAMP
ORT/www/coova.html

# This option will be configured to be the WISPr LoginURL as
well

# as provide “uamService” to the ChilliController. The UAM
Service is

# described in: http://www.coova.org/CoovaChilli/UAMService

#

# HS_UAMSERVICE=

###

# Features not activated per-default (default to off)

#

# HS_RADCONF=off      # Get some configurations from
RADIUS or a URL (‘on’ and ‘url’ respectively)

#

# HS_ANYIP=on        # Allow any IP address on subscriber LAN

#

# HS_MACAUTH=on      # To turn on MAC Authentication

#
```

```
# HS_MACAUTHDENY=on # Put client in 'drop' state on
MAC Auth Access-Reject

#

# HS_MACAUTHMODE=local # To allow MAC
Authentication based on macallowed, not RADIUS

#

# HS_MACALLOW="..." # List of MAC addresses to
authenticate (comma seperated)

#

# HS_USELOCALUSERS=on # To use the
/etc/chilli/localusers file

#

# HS_OPENIDAUTH=on # To inform the RADIUS server to
allow OpenID Auth

#

# HS_WPAGUESTS=on # To inform the RADIUS server to
allow WPA Guests

#

# HS_DNSPARANOIA=on # To drop DNS packets containing
something other

# # than A, CNAME, SOA, or MX records

#

# HS_OPENIDAUTH=on # To inform the RADIUS server to
allow OpenID Auth
```

```
#           # Will also configure the embedded login forms for
OpenID

#

# HS_USE_MAP=on      # Short hand for allowing the required
google

#           # sites to use Google maps (adds many google sites!)

#

###

# Other feature settings and their defaults

#

# HS_DEFSESSIONTIMEOUT=0 # Default session-timeout if
not defined by RADIUS (0 for unlimited)

#

# HS_DEFIDLETIMEOUT=0   # Default idle-timeout if not
defined by RADIUS (0 for unlimited)

#

# HS_DEFBANDWIDTHMAXDOWN=0 # Default WISPr-
Bandwidth-Max-Down if not defined by RADIUS (0 for
unlimited)

#

# HS_DEFBANDWIDTHMAXUP=0      # Default WISPr-
Bandwidth-Max-Up if not defined by RADIUS (0 for unlimited)

###

# Centralized configuration options examples
```

```

#
# HS_RADCONF=url      # requires curl
# HS_RADCONF_URL=https://coova.org/app/ap/config
# HS_RADCONF=on      # gather the ChilliSpot-Config
attributes in
#           # Administrative-User login
# HS_RADCONF_SERVER=rad01.coova.org      # RADIUS
Server
# HS_RADCONF_SECRET=coova-anonymous      # RADIUS
Shared Secret
# HS_RADCONF_AUTHPORT=1812      # Auth port
# HS_RADCONF_USER=chillispot      # Username
# HS_RADCONF_PWD=chillispot      # Password
###
# Firewall issues
#
# Uncomment the following to add ports to the allowed local ports
list
# The up.sh script will allow these local ports to be used, while the
default
# is to block all unwanted traffic to the tun/tap.
#
HS_TCP_PORTS="22 80 10000" //saya membuka port 22 untuk
ssh, 80 untuk webinterface EHS dan 10000 untuk webmin.

```

```
###  
  
# Standard configurations  
  
#  
  
HS_MODE=hotspot  
HS_TYPE=chillispot  
  
# HS_RADAUTH=1812  
  
# HS_RADACCT=1813  
  
# HS_ADMUSR=chillispot  
  
# HS_ADMPWD=chillispot  
  
###  
  
# Post-Auth proxy settings  
  
#  
  
# HS_POSTAUTH_PROXY=<host or ip>  
  
# HS_POSTAUTH_PROXYPORT=<port>  
  
# Directory specifying where internal web pages can be served  
# by chilli with url /www/<file name>. Only extentions like .html  
# .jpg, .gif, .png, .js are allowed. See below for using .chi as a  
# CGI extension.  
  
HS_WWWDIR=/etc/chilli/www  
  
# Using this option assumes 'haserl' is installed per-default  
# but, and CGI type program can ran from wwsh to process  
requests  
  
# to chilli with url /www/filename.chi  
  
HS_WWWBIN=/etc/chilli/wwsh
```

```

# Some configurations used in certain user interfaces

#

HS_PROVIDER=Lestari-Net

HS_PROVIDER_LINK=http://lestari-net.co.cc/

###

# WISPr RADIUS Attribute support

#

HS_LOC_NAME="Lestari-Net HotSpot" # WISPr Location
Name and used in portal

# WISPr settings (to form a proper WISPr-Location-Id)

# HS_LOC_NETWORK="My Network" # Network name

# HS_LOC_AC=408 # Phone area code

# HS_LOC_CC=1 # Phone country code

# HS_LOC_ISOCC=US # ISO Country code

```

Lakukan konfigurasi sesuai dengan kebutuhan, misalkan dari ip address, DNS maupun dari *radius proxynya*. Jika sudah dikonfigurasi lalu save

6. Menjalankan *Coovachilli*

Jika semua sudah terkonfigurasi langkah selanjutnya ialah menjalankan *Coovachilli*, masukkan perintah dibawah ini :

```
/etc/init.d/chilli start
```


7. Melakukan pengecekan

Seharusnya saat ini *server billing hotspot* sudah berjalan dengan baik, begitu juga dengan *easyhotspotnya*. lakukan pengecekan dengan perintah

```
ifconfig
```

Jika dalam *result ifconfig* telah muncul *eth1* dan *tun0* maka *server* anda sudah bisa digunakan. Hubungkan sebuah PC dengan kabel Lan ke *eth1*, dan pastikan PC tersebut menggunakan setting IP otomatis. dan lihat jika PC sudah mendapatkan IP DHCP dari *coovachilli* dengan maka *coovachilli* sudah bekerja dengan baik. Anda tinggal membuka *browser* dan buka alamat sembarang, disini harusnya anda *redirect* ke halaman *login* (dalam beberapa kali ujicoba sering kali *coova* merespon lambat sehingga user tidak di *redirect* kehalaman *login*). Untuk lebih jelasnya bisa dilihat pada Gambar 4.3.

```
root@pertamina-System-Product-Name: /home/pertamina
root@pertamina-System-Product-Name: /home/pertamina# ifconfig
eth4  Link encap:Ethernet  HWaddr 08:25:64:b0:26:22
      inet addr:192.168.1.2  Bcast:192.168.1.255  Mask:255.255.255.0
      inet6 addr: fe80::225:64ff:feb0:2622/64 Scope:Link
      UP BROADCAST RUNNING PROMISC  MTU:1500  Metric:1
      RX packets:13110 errors:0 dropped:0 overruns:0 frame:0
      TX packets:4299 errors:0 dropped:0 overruns:0 carrier:0
      collisions:0 txqueuelen:1000
      RX bytes:1265837 (1.2 MB)  TX bytes:1315190 (1.3 MB)
      Interrupt:21 Memory:fe6e0000-fe700000

Lo    Link encap:Local Loopback
      inet addr:127.0.0.1  Mask:255.0.0.0
      inet6 addr: ::1/128 Scope:Host
      UP LOOPBACK RUNNING  MTU:65536  Metric:1
      RX packets:2320 errors:0 dropped:0 overruns:0 frame:0
      TX packets:2320 errors:0 dropped:0 overruns:0 carrier:0
      collisions:0 txqueuelen:0
      RX bytes:485481 (485.4 KB)  TX bytes:485481 (485.4 KB)

tun0  Link encap:UNSPEC  HWaddr 00-00-00-00-00-00-00-00-00-00-00-00-00-00-00-00
-00
      inet addr:192.168.1.1  P-t-P:192.168.1.1  Mask:255.255.255.0
      UP POINTOPOINT RUNNING  MTU:1500  Metric:1
      RX packets:3504 errors:0 dropped:0 overruns:0 frame:0
      TX packets:3280 errors:0 dropped:0 overruns:0 carrier:0
      collisions:0 txqueuelen:100
      RX bytes:352342 (352.3 KB)  TX bytes:1176808 (1.1 MB)

usb0  Link encap:Ethernet  HWaddr 76:89:06:19:68:b0
      inet addr:192.168.42.23  Bcast:192.168.42.255  Mask:255.255.255.0
      inet6 addr: fe80::7489:6ff:fe19:68b0/64 Scope:Link
      UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
      RX packets:7775 errors:1 dropped:0 overruns:0 frame:1
      TX packets:7107 errors:0 dropped:0 overruns:0 carrier:0
      collisions:0 txqueuelen:1000
      RX bytes:6844972 (6.8 MB)  TX bytes:1239035 (1.2 MB)

root@pertamina-System-Product-Name: /home/pertamina#
```

Gambar 4.3 Perintah ifconfig untuk melihat jaringan yang terhubung

8. Membuat *symlink*

Langkah selanjutnya yang harus dilakukan adalah membuat *symlink* untuk *easyhotspot* agar bisa dibuka di *browser*, dengan melakukan perintah dibawah ini

```
In -s /opt/easyhotspot/htdocs /var/www/easyhotspot
```

Setelah itu silahkan buka **<http://ip-server-anda/easyhotspot>** untuk memulai manajemen *hotspot* harus mat, dan sebelumnya harus memastikan tiga *file config easyhotspot* sudah terisi dengan benar yaitu **[/opt/easyhotspot/htdocs/system/application/config/config.php](#)**, **[/opt/easyhotspot/htdocs/system/application/config/database.php](#)** **[/opt/easyhotspot/htdocs/system/application/config/easyhotspot.php](#)**

9. Menginstall halaman *login Coovachilli*

Untuk menginstallnya anda bisa melakukan perintah dibawah :

```
wget http://ndra.gmib26.net/files/www.zip
$cp www.zip /etc/chilli
$cd /etc/chilli
$cp -r /etc/chilli/www /etc/chilli/www.bak
$rm -rf /etc/chilli/www
$unzip -a www.zip
```

Untuk melakukan perubahan terhadap halaman *login* dari *hotspot* bisa dilakukan dengan mengedit file-file yang ada di **[/etc/chilli/www](#)** dan jika ingin mudah menggunakan atau ingin mendesain halaman *login*

yang sudah dibuat, bisa mencari *file* berekstensi **.tmpl** dan merubahnya sesuai dengan keinginan masing – masing.

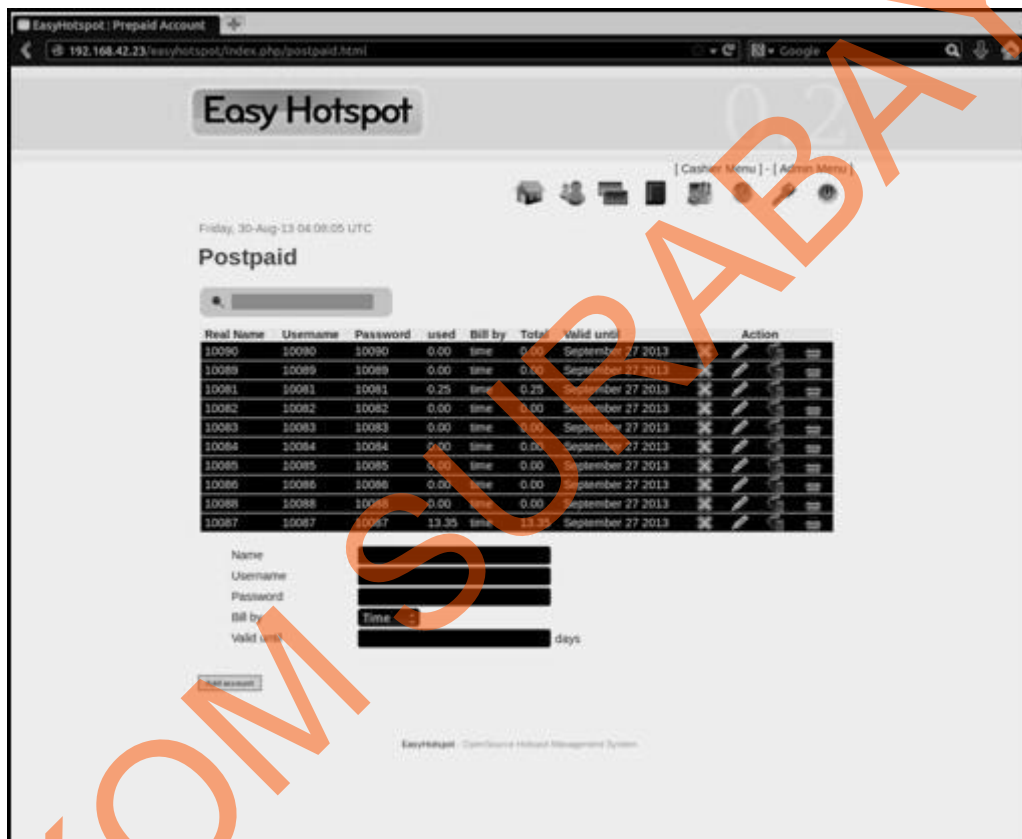
10. Pengujian *Coovachilli*

Ketika *user* tamu ingin mengakses *internet* maka secara otomatis user langsung ke *redirect* ke halaman *captive portal* yang sudah dibuat. Pada halaman tersebut *Coovachilli* akan meminta *user id* dan *password* dari *user* tersebut sebagai autentikasi *user*, sehingga user harus memasukkan *user id* dan *password* sesuai dengan *user id* dan *password* yang berada di dalam database. Berikut tampilan *Coovachilli* ketika meminta *user id* dan *password* sebagai autentikasi *user*. Untuk melihat tampilan awal *Coovachilli* yang mendirect *web* bisa dilihat pada Gambar 4.4.



Gambar 4.4 Tampilan *Coovachilli* ketika meminta autentikasi *user*

Apabila user sudah memasukkan *user id* dan *password* kemudian *Coovachilli* akan mengirimkan data tersebut ke dalam *database* yang sudah dibuat, disini tugas dari *database* untuk mencocokkan data dari *user* dengan data dari *database*, seperti yang ditunjukkan pada Gambar 4.5.



Gambar 4.5 Tampilan Easyhotspot

Apabila cocok maka *Coovachilli* akan langsung memberikan layanan kepada *user*. Tetapi sebaliknya apabila data tidak cocok maka *Coovachilli* langsung mendrop *packet* tersebut sehingga *user* tidak bisa menggunakan *internet*.