PENGEMBANGAN JARINGAN *WIRELESS* MENGGUNAKAN *USER AUTHENTICATION* BERBASIS RADIUS DALAM INDUSTRI 4.0 (Studi Kasus: Universitas Widyatama)

Muchamad Rusdan[1], Muhamad Sabar[2]
Program Studi Teknik Informatika, Sekolah Tinggi Teknologi Bandung
Email : rusdan@sttbandung.ac.id[1], sabar@sttbandung.ac.id[2]

ABSTRAK
Penelitian ini bertujuan mengembangkan jaringan wireless pada area hotspot yang masih mempergunakan Wi-Fi Protected Access 2-Pre-Shared Key (WPA2-PSK) akan dikembangkan menjadi berbasis Remote Access Dial In User Service (RADIUS) sebagai user authentication yang aman dan user-friendly yang mampu membedakan user yang diizinkan dan tidak diizinkan. Di Universitas Widyatama setiap mahasiswa, dosen, dan karyawan dapat menggunakan layanan jaringan wireless yang ada dengan cara memasukkan password pada laptop, smartphone, tablet, dan perangkat lainnya yang mendukung penggunaan WPA2-PSK untuk melakukan user authentication. Tujuan utama penelitian ini adalah untuk mengembangkan jaringan wireless dengan menggunakan user authentication dalam melakukan koneksi jaringan wireless, meningkatkan keamanan serta kemudahan dalam penggunaan jaringan wireless, dan dapat menghasilkan kebijakan dan prosedur penggunaan akses jaringan wireless. Pada penelitian ini metode penelitian yang digunakan metode Network Development Life Cycle (NDLC). Pengujian jaringan wireless dengan pengujian user authentication akan diuji melalui captive portal dan terminal server RADIUS dalam mewujudkan jaringan wireless yang aman dan user-friendly. Diperoleh kesimpulan bahwa user authentication berbasis RADIUS setelah dilakukan pengujian menggunakan perangkat smartphone dan notebook dapat disimpulkan bahwa user authentication berbasis RADIUS aman dan user-friendly.

Kata kunci: NDLC , RADIUS, User Authentication, User-friendly, Wireless Network

1. PENDAHULUAN

Pada era informasi saat ini, manusia memerlukan komunikasi untuk saling bertukar informasi dimana saja, kapan saja, dan dengan siapa saja. *Internet* telah merubah banyak hal, khususnya didalam pemenuhan kebutuhan akan informasi dan sistem jaringannya yang luas menciptakan banyak kemudahan akses informasi tidak lagi hanya terhubung ke *internet*, namun juga sudah mulai bergeser pada kemampuan mobilitas pengguna yang selalu terhubung dengan internet dengan cepat bahkan *realtime*. Untuk memenuhi sifat mobilitas dan selalu terhubung dibutuhkan solusi tanpa kabel atau wireless. Jaringan wireless adalah salah satu teknologi yang saat ini sudah digunakan secara luas diberbagai institusi, kantor, cafe, mall, bandara, hotel, bahkan di sekolah-sekolah maupun di kampus-kampus[1].

Jaringan *wireless* memberikan kemudahan dan fleksibilitas yang cukup tinggi serta nyaman untuk digunakan. Selama berada dalam area cakupan jaringan *wireless*, pengguna dapat mengakses *internet* setiap saat. Untuk membuat sebuah jaringan *wireless* terkoneksi ke *internet* dengan aman dan *user-friendly*, maka kita bisa membuat sebuah sistem *user authentication* yang berbasis *Remote Access Dial In User Service* (RADIUS) *Server* yang dapat digunakan untuk melakukan *Authentication*, *Authorization*, dan *Accounting* (AAA)[2].

Di Universitas Widyatama setiap mahasiswa, dosen, dan karyawan dapat menggunakan layanan jaringan *wireless* yang ada dengan cara memasukkan *password* pada *laptop*, *smartphone*, *tablet*, dan perangkat lainnya yang mendukung penggunaan *Wi-Fi Protected Access* (WPA) untuk melakukan *user authentication*. Sehingga sangat sulit untuk membedakan *user* atau pengguna yang diijinkan dan tidak diijinkan untuk menggunakan layanan jaringan *wireless* tersebut. Dengan adanya permasalahan tersebut maka alternatif solusinya perlu dikembangkan *user authentication* yang aman dan *user-friendly* yang mampu membedakan *user* atau pengguna yang diijinkan dan tidak diijinkan yang berbasis *Remote Access Dial In User Service* (RADIUS) pada jaringan *wireless* Universitas Widyatama [3].

Dari pemaparan latar belakang diatas maka, dapat dirumuskan masalah yaitu bagaimana mengembangkan *user authentication* yang aman dan *user-friendly* yang mampu membedakan *user* atau pengguna yang diijinkan dan tidak diijinkan untuk menggunakan layanan jaringan *wireless* berbasis RADIUS.

2. TINJAUAN PUSTAKA

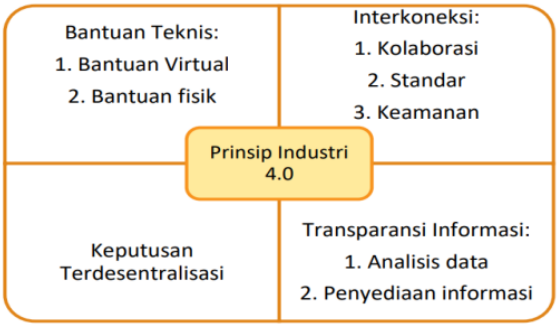
Jaringan *wireless* merupakan suatu jaringan komputer yang menggunakan frekuensi radio untuk komunikasi antara perangkat komputer dan akhirnya titik akses yang merupakan dasar dari komunikasi radio dua arah yang karakteristiknya bekerja pada frekuensi 2,4 GHz (802.11 b/g/n/ac) atau 5 GHz (802.11 a/n/ac). *Backbone* jaringan *wireless* biasanya menggunakan kabel, dengan satu atau lebih titik akses [4].

User Authentication adalah proses usaha pengecekan identitas seorang pengguna sistem komunikasi pada proses *login* ke dalam sebuah sistem. Pengguna yang telah lolos pengecekan identitas adalah pengguna resmi pada sistem, orang yang memiliki otoritas atas sistem, atau mungkin aplikasi yang berjalan pada sistem. Penggunaan sistem autentikasi diharapkan dapat membentuk sebuah sistem khusus, yang hanya dapat dipergunakan oleh orang-orang yang memiliki hak guna [5].

Remote Access Dial In User Service (RADIUS) merupakan *network protocol* keamanan komputer yang digunakan untuk membuat manajemen akses secara terkontrol pada sebuah jaringan yang besar[6]. RADIUS dapat didefinisikan di dalam *Request For Comments* (RFC) 2865 dan RFC 2866. RADIUS biasa digunakan oleh perusahaan untuk mengatur hak akses ke *internet* bagi *user* [7].

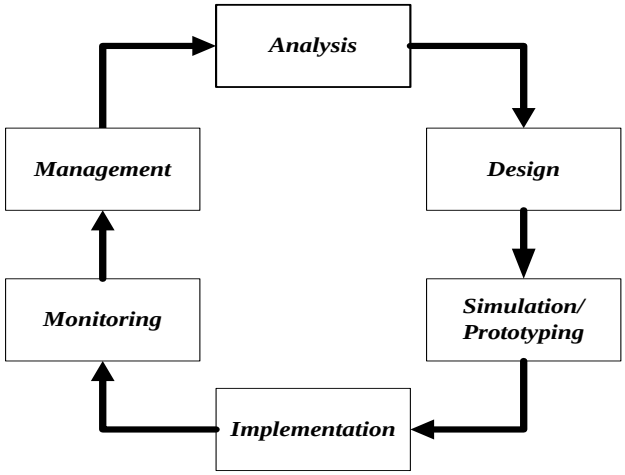
Industri 4.0 merupakan suatu istilah yang berasal dari sebuah proyek yang dipelopori oleh pemerintah Jerman untuk memperkenalkan komputerisasi pada industri manufaktur[8]. Empat desain prinsip industri 4.0, yaitu[8]:

1. Interkoneksi (sambungan)
2. Transparansi informasi
3. Bantuan teknis
4. Keputusan terdesentralisasi



Gambar 1 Desain Prinsip Industri 4.0
(Sumber: Herman **ET AL, 2016)**

Metode penelitian *Network Development Life Cycle* (NDLC) yang tahapannya dapat dilihat pada gambar 2, metode tersebut yang didalamnya meliputi:



Gambar 2 Metode NDLC

Pada gambar 2 dapat diuraikan sebagai berikut:

a. *Analysis* (Analisis)

Tahap awal ini dilakukan analisis kebutuhan, analisis permasalahan yang muncul, analisis keinginan user, dan analisis sistem yang sudah ada saat ini menggunakan teknik observasi dan teknik studi literatur.

b. *Design* (Desain)

Tahap desain ini akan menghasilkan desain topologi jaringan *wireless* yang akan dibangun dan/atau dikembangkan, diharapkan dengan desain akan memberikan gambaran seutuhnya dari kebutuhan yang ada.

c. *Simulation/Prototyping* (Prototipe/Simulasi)

Tahap ini yang akan diterapkan adalah pembuatan prototyping atau prototype bukan pembuatan simulation (simulasi). *Prototyping* merupakan proses perulangan dalam pengembangan sistem dimana kebutuhan diubah ke dalam sistem yang bekerja *(working system)* yang secara terus menerus diperbaiki melalui kerjasama antara user dan analis.

d. *Implementation* (Implementasi)

Tahap implementasi akan menerapkan semua yang telah direncanakan dan di desain sebelumnya. Implementasi merupakan tahapan yang sangat menentukan berhasil atau gagalnya suatu *project* yang akan dibangun dan/atau dikembangkan.

e. *Monitoring*

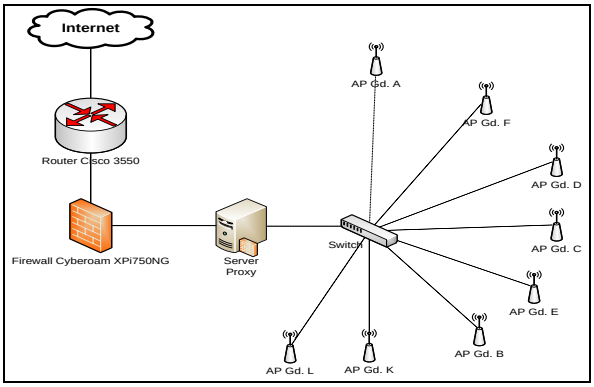
Tahap *monitoring* merupakan tahapan yang penting, agar jaringan komputer dan komunikasi dapat berjalan sesuai dengan keinginan dan tujuan awal dari *user* pada tahap awal analisis, maka pada penelitian ini perlu dilakukan kegiatan *monitoring* atau pengawasan terhadap sistem yang telah dibuat.

f. *Management* (Manajemen)

Tahap *management* atau pengaturan, salah satu yang menjadi perhatian yang khusus adalah masalah *policy* (kebijakan). Kebijakan perlu dibuat untuk mengatur agar sistem yang telah dibangun dan/atau dikembangkan dapat berjalan dengan baik serta dapat berlangsung lama dan unsur *reliability* terjaga.

2.1 ANALISIS DAN PERANCANGAN

A. Analisis Sistem yang Sedang Berjalan



Gambar 3 Kondisi *Existing*

Jaringan *wireless* di Universitas Widyatama pada saat ini untuk keamanannya belum mempergunakan *user authentication* (*user* dan *password*) namun hanya mempergunakan *Wi-Fi Protected Access 2-Pre-Shared Key* (WPA2-PSK) sebagai *user authentication*. Gambar 3 memperlihatkan bahwa belum memiliki server *user authentication* yang berbasis *Remote Access Dial In User Service* (RADIUS), namun masih menggunakan WPA2-PSK sebagai pengganti *user authentication.*

B. Analisis Kebutuhan Sistem

Analisis kebutuhan perangkat keras dan perangkat lunak jaringan *wireless* dengan menggunakan *user authentication* berbasis RADIUS, yaitu:

Tabel 1 Kebutuhan Perangkat Keras

No	Nama	Qty	Keterangan
1	Komputer *Server* I	1	*Captive Portal + RADIUS client + Firewall*
2	Komputer *Server* II	1	*Data Base* RADIUS

Adapun alasan pemilihan perangkat keras pada Tabel 1 selain karena merupakan perangkat keras tersebut dibutuhkan dalam menunjang pengembangan jaringan *wireless* dengan *user authentication* berbasis *Remote Access Dial In User Service* (RADIUS), alasan lainnya penggunaan 2 komputer *server* bertujuan supaya aplikasi *user authentication* dan *database* terpisah guna menjamin keamanan data dan proses tidak memberatkan kerja dari komputer *server*. Pemilihan perangkat keras pun disesuaikan dengan kebutuhan dari perangkat lunak yang akan digunakan dalam pengembangan jaringan *wireless* menggunakan *user authentication* berbasis RADIUS.

Tabel 2 Kebutuhan Perangkat Lunak

No	Nama	Keterangan
1	Sistem Operasi *Linux Ubuntu*	*Server Captive Portal + RADIUS Client + Firewall* dan *Server* DB RADIUS
2	*CoovaChilli*	Aplikasi *Captive Portal* dan DHCP
3	*Apache 2.0*	Aplikasi *Web Server*
4	*MySQL*	Aplikasi *Database* RADIUS
6	PHP	Aplikasi Bahasa Pemrograman
7	FreeRADIUS	Aplikasi RADIUS berbasis *Open Source*

Alasan pemilihan perangkat lunak pada Tabel 2 selain karena merupakan perangkat lunak *Open Source* yang bisa didapatkan secara mudah, alasan lainnya bebas dapat diunduh dari *internet* secara gratis dan bebas untuk dimodifikasi sesuai dengan kebutuhan, dukungan yang berkualitas tersedia secara gratis di Internet, memiliki fitur keamanan yang lebih unggul, sangat rendah terinfeksi oleh *virus*, *trojan*, *worm*, *spyware*, dan *malware*, memiliki fleksibilitas tinggi untuk dikonfigurasi, dan proses pemasangan serta konfigurasi lebih mudah untuk dilakukan.

C. Analisis Pengguna

Pengguna yang menggunakan layanan jaringan *wireless* di Universitas Widyatama meliputi:

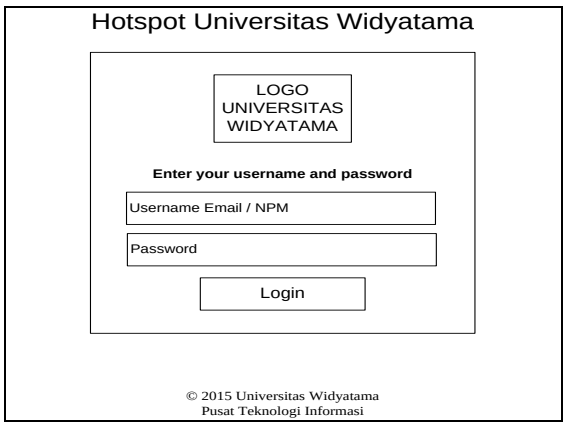
a. Mahasiswa, dengan jumlah mahasiswa yang aktif sekitar 8000 mahasiswa.
b. Dosen, dengan jumlah dosen tetap dan tidak tetap sekitar 700 dosen.
c. Karyawan, dengan jumlah karyawan Yayasan Widyatama dan Universitas Widyatama sekitar 300 karyawan.

Pengguna jaringan *wireless* tersebar dibeberapa titik lokasi yaitu gedung A, gedung B, gedung K, gedung C, gedung D, gedung F, dan gedung E, yang mana dari hasil observasi didapatkan bahwa titik lokasi yang paling banyak mengakses atau menggunakan jaringan *wireless* yaitu gedung K (Perpustakaan), gedung B (*Foodcourt*), dan gedung D (Taman), dikarenakan lokasi tersebut merupakan tempat berkumpul dan berinteraksinya mahasiswa, dosen, dan karyawan.

3. HASIL DAN PEMBAHASAN

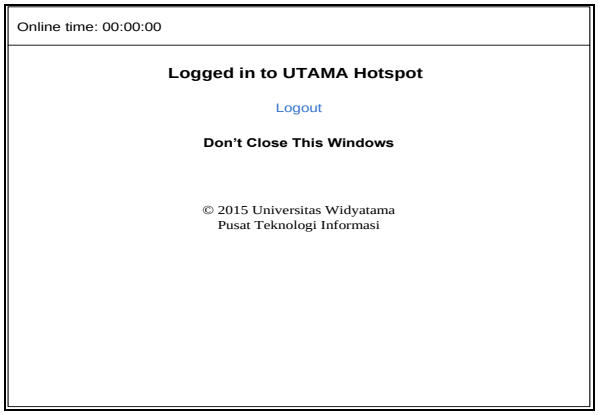
3.1 Desain *User Interface*

Desain *user interface* menu *login* aplikasi *user authentication* berbasis RADIUS merupakan hasil pengembangan dari *user interface* menu *login* yang dimiliki oleh *CoovaChilli*, yang disesuaikan dengan kebutuhan dari sistem yang diharapkan oleh *user*.



Gambar 4 Desain *User Interface Login*

Gambar 4 merupakan rancangan dari *form login* dari aplikasi *user authentication* berbasis RADIUS yang didalamnya memiliki beberapa bagian nama *header*, *logo* Universitas Widyatama, kolom input *username*, kolom input *password*, dan tombol *Login*.



Gambar 5 Desain *User Interface Logout*

Gambar 5 merupakan rancangan tampilan *user logout* yang berfungsi untuk mengakhiri dan/atau keluar dari layanan jaringan *wireless* di Universitas Widyatama.

A. Skenario Pengujian Sistem

Pada pengujian sistem ini, akan dilakukan pembuktian terhadap keamanan dan *user-friendly* dari aplikasi *user authentication* berbasis *Remote Access Dial In User Service* (RADIUS). Adapun skenario yang akan dilakukan yaitu:

a. Pengujian *authentication* pada sisi RADIUS *server*.
b. Pengujian *authentication* pada sisi *client* jaringan *wireless* melalui aplikasi *user authentication* atau *captive portal.*

3.2 Implementasi Aplikasi *User Auhentication* Berbasis RADIUS

User authentication merupakan sistem keamanan pada jaringan *wireless* yang mekanismenya meminta pengguna layanan jaringan *wireless* untuk memasukkan *username* dan *password* yang cocok dengan yang tersimpan pada *database*, pada penelitian ini *database* yang menyimpan informasi *username* dan *password* yaitu *database* RADIUS, pada penelitian ini menggunakan RADIUS yang berbasis *opensource* yaitu FreeRADIUS. Proses permintaan dan pengiriman *username* dan *password* pengguna dilakukan melalui *captive portal*, pada penelitian ini menggunakan *captive portal opensource* yaitu *CoovaChilli* yang telah dimodifikasi sesuai dengan kebutuhan. Proses implementasi *user authentication* berbasis RADIUS sebagai berikut:

1) Instalasi dan Konfigurasi MySQL

MySQL diperlukan untuk menyimpan *database* pengguna (Mahasiswa, Dosen, dan Pegawai) yang tersinkronisasi dengan *database* akun e-mail. Untuk menginstalasi *MySQL* pada sistem operasi *Ubuntu 12.04* dengan menggunakan perintah:

 # sudo apt-get update
 # sude apt-get install –f mysql-server-5.0

Setelah proses instalasi *MySQL* selesai, maka selanjutnya adalah proses konfigurasi pada file *my.cnf*, dengan cara sebagai berikut:

 # sudo nano /etc/mysql/my.cnf

Kemudian cari baris yang terdapat *bind-address* = 127.0.0.1 dan berikan tanda # pada awal baris, seperti di bawah ini:

bind-address = 127.0.0.1

menjadi

#bind-address = 127.0.0.1

Konfigurasi di atas dimaksudkan supaya *server MySQL* dapat diakses bukan hanya dari *localhost* saja. Kemudian lakukan restart pada *server MySQL* dengan perintah:

 # sudo /etc/init.d/mysql restart

Setelah semua langkah instalasi dan konfigurasi dilaksanakan maka *server MySQL* siap untuk dipergunakan.

2) Instalasi dan Konfigurasi *Apache* dan PHP

Penelitian ini menggunakan model *authentication CoovaChilli* yang menggunakan *Universal Access Method* (UAM), maka dengan demikian *server* membutuhkan *Apache* sebagai *web server* yang dilengkapi dengan modul SSL dan PHP. Cara menginstalasi Apache dan PHP dapat dilakukan dengan perintah:

 # sudo apt-get install –f apache2 php5

Setelah proses instalasi *Apache* dan PHP maka dilanjutkan dengan proses instalasi modul SSL pada *Apache* yang digunakan untuk mengenkripsi data antara *browser* pengguna dengan *web server Apache*. Cara instalasi modul SSL pada *Apache* dibutuhkan *OpenSSL* dan konfigurasi *ssl-cert*, dengan perintah sebagai berikut:

 # sudo apt-get install –f openssl ssl-cert

Setelah instalasi modul SSL tidak langsung aktif secara *default* dengan demikian untuk mengaktifkannya dibutuhkan sertifikat SSL dan konfigurasi *site* SSL, cara membuatnya sebagai berikut:

 # sudo mkdir /etc/apache2/ssl

 # sudo make-ssl-cert /usr/share/ ssl-cert/ssleay.cnf /etc/apache2/ssl/apache.pem

Pada tahap pengaktifan di atas, *user* akan diberikan beberapa pertanyaan mengenai organisasi yang akan menggunakan sertifikat SSL tersebut. Modul SSL juga perlu diaktifkan setelah proses instalasi selesai dengan perintah:

ptiutama@wifi:~# sudo a2enmod ssl

Langkah selanjutnya adalah membuat situs khusus untuk akses HTTPS, dengan perintah sebagai berikut:

 # sudo cp /etc/apache2/sites-available/default /etc/apache2/sites-available/ssl

Setelah proses pembuatan situs khusus untuk akses HTTPS, langkah selanjutnya adalah mengedit *file* /etc/apache2/sites-available/ssl dengan mengubah 3 baris teratas menjadi seperti dibawah ini:

NameVirtualHost *:443
<VirtualHost *:443>
ServerAdmin itc@widyatama.ac.id
SSLEngine On
SSLCertificateFile /etc/Apache2/ssl/apache.pem

Langkah selanjutnya *enable* situs SSL yang telah dibuat kemudian ubah situs *default* supaya tidak berbentrokan dengan situs SSL, dengan perintah sebagai berikut:

 # sudo a2ensite ssl
 # sudo nano /etc/apache2/sites-available/default

Langkah selanjutnya mengubah 2 baris isi *file /etc/apache2/sites-available/default* menjadi seperti dibawah ini:

.NameVirtualHost *:80
<VirtualHost *:80>

Langkah selanjutnya ubah berkas */etc/apache2/ports.conf* menjadi seperti dibawah ini:

Listen 80
Listen 443

Setelah diubah simpan berkas tersebut kemudian reload kembali apache:

 # sudo /etc/init.d/apache2 force-reload

3) Instalasi RADIUS *Server* dengan FreeRADIUS

Setelah proses instalasi *MySQL*, *Apache*, dan PHP selesai, maka langkah selanjutnya adalah instalasi RADIUS *Server* dengan menggunakan perangkat lunak berbasis *opensource FreeRADIUS*. Pemakaian *FreeRADIUS* bertujuan untuk menghubungkan basis data *MySQL* dan *captive portal CoovaChilli*. Cara instalasi *FreeRADIUS* pada sistem operasi *Ubuntu 12.04* dengan menggunakan perintah sebagai berikut:

 # sudo apt-get install –f freeradius freeradius-mysql

Langkah selanjutnya adalah menyiapkan dan/atau membuat *database MySQL* yang akan digunakan oleh *FreeRADIUS*. *FreeRADIUS* dalam paket instalasinya telah menyediakan skema *database* yang dapat diambil pada direktori */usr/share/doc/freeradius/examples/*. Tahap-tahap dalam membuat *database MySQL FreeRADIUS* sebagai berikut:

 # sudo mysql -u root -p
Enter password: rusdan123
mysql> CREATE DATABASE radius_wifi;
quit;

langkah selanjutnya adalah memasukkan skema *database FreeRADIUS* ke dalam *database radius_wifi*, perintahnya sebagai berikut:

 # sudo mysql -u root –prusdan123 radius_wifi < /etc/freeradius/sql/mysql/schema.sql
 # sudo mysql -u root –prusdan123 radius_wifi < /etc/freeradius/sql/mysql/nas.sql

Langkah selanjutnya supaya *FreeRADIUS* dapat mengakses *database* maka, perlu dilakukan konfigurasi *file /etc/freeradius/sql.conf*, yang caranya sebagai berikut:

 # sudo nano /etc/freeradius/sql.conf
server = "localhost"
login = "root"
password = "rusdan123"
radius_db = "radius_wifi"
exit;

langkah selanjutnya melakukan set *FreeRADIUS client password*, sebagai berikut:

 # sudo nano /etc/freeradius/clients.conf
client 127.0.0.1 {
secret = tesing123
nastype = other
shortname = ptiutama
}

Langkah selanjutnya melakukan pengetesan terhadap *FreeRADIUS* yang telah diinstalasi dan dikonfigurasi dengan perintah sebagai berikut:

 # sudo /etc/init.d/freeradius stop
 # sudo freeradius -XXX

Bilamana *FreeRADIUS* telah berjalan dengan baik, maka akan tampil pesan seperti dibawah ini:

Debug: Ready to process requests.

Kemudian tekan tombol Ctrl+C untuk *exit*. Kemudian jalankan *FreeRADIUS* dengan perintah sebagai berikut:

 # sudo /etc/init.d/freeradius start

Langkah selanjutnya yaitu pengetesan *user* yang telah ditersimpan di *database FreeRADIUS*, dengan perintah sebagai berikut:

 # sudo radtest rusdan rusdan123 127.0.0.1 0 testing123

Bilamana semua berjalan dengan baik, maka akan memperoleh jawaban dari *FreeRADIUS* sebagai berikut:

Sending Access-Request of id 182 to 127.0.0.1 port 1812
User-Name = "rusdan"
User-Password = "rusdan123"
NAS-IP-Address = 127.0.1.1
NAS-Port = 0
rad_recv: Access-Accept packet from host 127.0.0.1 port 1812, id=182, length=37

4) Instalasi dan Konfigurasi *CoovaChilli*

Setelah proses instalasi dan konfigurasi *FreeRADIUS* maka dilanjutkan dengan instalasi dan konfigurasi *CoovaChilli*. Cara instalasi dan konfigurasi seperti di bawah ini:

 # sudo wget http://ap.coova.org/chilli/coova-chilli_1.0.13-1_i386.deb

Kemudian lakukan proses instalasi dengan perintah sebagai berikut:

 # sudo dpkg -i coova-chilli_1.0.13-1_i386.deb

Kemudian *copy file* konfigurasi *default* dan konfigurasi *Apache* dengan perintah sebagai berikut:

 # sudo cp /etc/chilli/defaults /etc/chilli/config
 # sudo mkdir /var/www/hotspot
 # sudo cd /var/www/hotspot
 # sudo cp /etc/chilli/www/* /var/www/hotspot
 # sudo mkdir /var/www/hotspot/images
 # sudo cp /var/www/hotspot/coova.jpg /var/www/hotspot/images/
 # sudo mkdir /var/www/hotspot/uam
 # sudo cd /var/www/hotspot/uam
 # sudo wget http://ap.coova.org/uam/
 # sudo wget http://ap.coova.org/js/chilli.js

Langkah selanjutnya adalah meng-*copy file /etc/chilli/defaults* ke *file* baru dengan nama *config* pada *directory* yang sama. Kemudian lakukan konfigurasi pada *file /etc/chilli/config* dengan perintah sebagai berikut:

 # sudo nano /etc/chilli/config

Isi dari *file /etc/chilli/config*:

```
# -*- /bin/sh -*-
#
#   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'.
# HS_WANIF=eth0               # WAN Interface toward the Internet
HS_LANIF=eth1           # Subscriber Interface for client devices
HS_NETWORK=172.16.0.0       # HotSpot Network (must include HS_UAMLISTEN)
HS_NETMASK=255.255.0.0  # HotSpot Network Netmask
HS_UAMLISTEN=172.16.1.1 # HotSpot IP Address (on subscriber network)
HS_UAMPORT=3990         # HotSpot Port (on subscriber network)
# Allow some additional local ports (used in the up.sh script when
# setting the firewall for the created tun/tap)
HS_TCP_PORTS="80 443"
# HS_DYNIP=
# HS_DYNIP_MASK=255.255.255.0
# HS_STATIP=
# HS_STATIP_MASK=255.255.255.0
# HS_DNS_DOMAIN=
# if your interface eth0 for example has the ip 192.168.5.2
# and your router where your internet connection is established has the address
# 192.168.5.1 than you are allowed to access the router from your wlan network 192.168.2.0/24
# so you have don't have to define the dns servers below
#
HS_DNS2=62.72.64.237
HS_DNS1=192.168.2.1
###
#   HotSpot settings for simple Captive Portal
#
HS_NASID=nas01
HS_UAMSECRET=testing123
HS_RADIUS=127.0.0.1
HS_RADIUS2=127.0.0.1
HS_RADSECRET=radiussecret
# please provide here the address for your router too.
# From the example above it has the address 192.168.0.1 (comma separated)
HS_UAMALLOW=172.16.1.1
#    Put entire domains in the walled-garden with DNS inspection
# HS_UAMDOMAINS="www.widyatama.ac.id"
#   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=172.16.1.1
#   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=https://\$HS_UAMSERVER/uam/
#   Same principal goes for HS_UAMHOMEPAGE.
HS_UAMHOMEPAGE=http://\$HS_UAMLISTEN:\$HS_UAMPORT/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://coova.org/wiki/index.php/CoovaChilli/UAMService
#
HS_UAMSERVICE=https://10.1.0.1/cgi-bin/hotspotlogin.cgi
```

5) Instalasi dan Konfigurasi Firewall

Pembuat *CoovaChilli* sudah membuat aturan firewall untuk *IPTables*, tapi *script* yang disediakan masih membutuhkan sedikit bantuan untuk dapat berjalan dengan baik. Konfigurasi *IPTables* di *CoovaChilli* dilakukan melalui script /etc/chilli/up.sh yang di jalankan sesudah *interface* tun0 beroperasi. *File* /etc/chilli/up.sh dijalankan dengan perintah sebagai berikut:

 # sudo sh /etc/chilli/up.sh

Setelah *script firewall CoovaChilli* dijalankan maka aturan *firewall IPTables* telah terpasang.

6) Sinkronisasi *Database* RADIUS dengan *Database* E-Mail

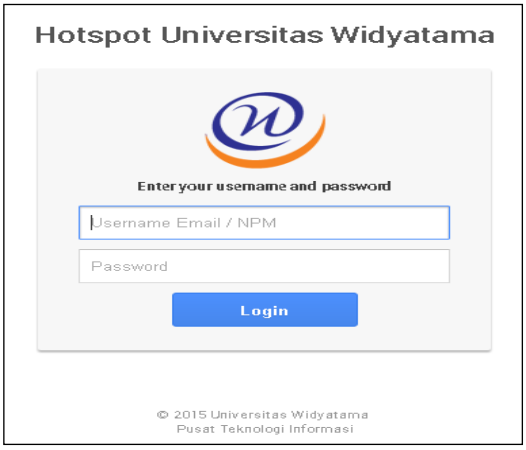
Setelah proses instalasi selesai, maka langkah selanjutnya adalah bagaimana mensinkronisasikan *database* RADIUS dengan *database* E-Mail, dikarenakan *username* dan *password* yang digunakan untuk *login* ke jaringan *wireless* Universitas Widyatama akan menggunakan NPM, *username*, dan *password* e-mail.

Database (db) RADIUS dan *database* e-mail perlu disinkronisasikan karena data NPM, *username*, dan *password* yang ada pada db RADIUS harus sama isinya dengan db e-mail, selain itu dikarenakan struktur tabel db RADIUS dengan db e-mail sangat berbeda maka proses sinkronisasi pun tidak bisa dilakukan secara langsung, maka diperlukan pembuatan *script* untuk melakukan sinkronisasi antara db RADIUS dengan db e-mail. Proses sinkronisasi db RADIUS dengan db e-mail dilakukan dengan membuat *script* PHP yang berjalan secara otomatis dan/atau dieksekusi setiap jam. Adapun *script* PHP yang dibuat untuk melakukan sinkronisasi db RADIUS dan db e-mail sebagai berikut:

```
<?php
        $username = "user";
        $password = "user123";
        $hostname = "192.168.1.1";
        $dbmaster  = "dbemail";
        $tbmaster = "newaccountuser";
        $username1 = "user1";
        $password1 = "user1123";
        $hostname1 = "localhost";
        $dbmaster1  = "radius_wifi";
        $tbmaster1 = "radcheck";
        /*
        //connection to the database
        $dbhandle = mysql_connect($hostname, $username,
$password)
          or die("Unable to connect to MySQL");
        echo "Connected to MySQL<br>";
        //select a database to work with
        $selected = mysql_select_db($dbmaster,$dbhandle)
        or die("Could not select examples");
        $result = mysql_query("SELECT B.user_username,
        A.user_password                    FROM
        dbemailutama.td_akun_pwd A INNER JOIN
        dbemailutama.tm_ak$
        mysql_close($dbhandle);
        $dbhandle1 = mysql_connect($hostname1,
$username1, $password1)
                or die("Unable to connect to MySQL
Local");
        echo "Connected to MySQL Local<br>";
        $selected1                                  =
mysql_select_db($dbmaster1,$dbhandle1)
                or die("Could not select examples");
        while ($row = mysql_fetch_array($result)) {
        $result1 = mysql_query("SELECT COUNT(*) dup
        FROM radcheck WHERE UserName =
        '".$row{'user_username'}."'");
        $row1 = mysql_fetch_array($result1);
         if($row1{'dup'} == '0'){
         $result1 = mysql_query("INSERT INTO radcheck
        (UserName, Attribute, Value) VALUES
        ('".$row{'user_username'}."','Passwo$
          }else{
        $result1 = mysql_query("UPDATE radcheck SET
        Value = '".$row{'user_password'}."' WHERE
        UserName = '".$row{'user_userna$
          }
        }
        mysql_close($dbhandle1);
*/
?>
```

4 Pengujian *User Authentication*

Tahap pengujian aplikasi *user authentication* berbasis RADIUS dilakukan menggunakan *Smartphone* Samsung Galaxy Mini dan *Notebook* Toshiba M55 dengan mencoba terhubung ke jaringan *wireless* Universitas Widyatama yang telah selesai dikembangkan dengan *Service Set Identifier* (SSID): @UTAMA. Pengujian dilakukan dengan menggunakan *web browser Google Chrome* baik pada *smartphone* dan *notebook*. Setelah *user* berhasil terhubung ke jaringan *wireless* Universitas Widyatama dengan mendapatkan *Internet Protocol* (IP) *Address Dynamic Host Configuration Protocol* (DHCP) dari *server*, maka selanjutnya *user* akan dipaksa untuk melakukan *authentication* dengan mengunjungi *captive portal*. Seperti dapat dilihat pada gambar 6.



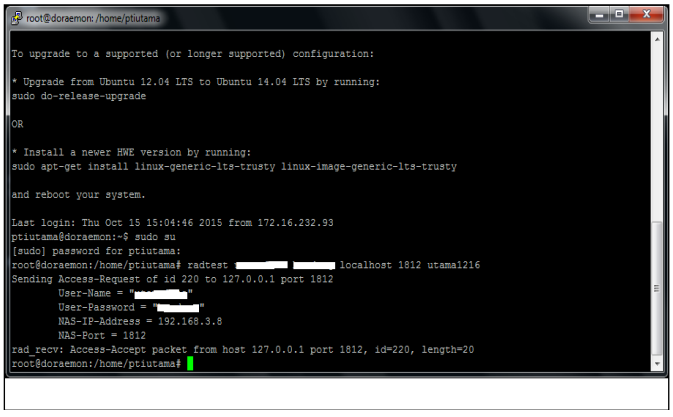
Gambar 6 User Interface Form Login

Pengujian aplikasi *user authentication* berbasis RADIUS dilakukan dengan dua cara. Cara pertama dilakukan dengan melakukan proses pengujian *authentication* pada sisi server RADIUS dan cara kedua dilakukan pengujian *authentication* pada sisi *user* jaringan *wireless* melalui *captive portal*. *Username* dan *password* yang digunakan untuk melakukan *login* ke jaringan *wireless* pengguna dibagi menjadi 3 bagian, yaitu:

a. Mahasiswa, *login* dengan menggunakan Nomor Pokok Mahasiswa (NPM) sebagai *username* dan *password portal* sebagai *password*-nya.
b. Dosen dan karyawan, login menggunakan username dan password e-mail Universitas Widyatama.

Pengujian *authentication* pada RADIUS *server* dilakukan untuk memastikan RADIUS *server* dapat berjalan dengan baik dalam melakukan proses *authentication* terhadap data *user* yang terekam dalam *database MySQL*. Perintah untuk melakukan pengujian *authentication* RADIUS *server* sebagai berikut:

 # sudo radtest rusdan rusdan123 localhost 1812 testing123



Gambar 7 Keterangan Reply Message *Access-Accept*

Hasil pengujian *authentication* RADIUS *server* secara umum *server* dapat berjalan dengan baik dalam menangani permintaan *authentication request.* Hal ini dapat diketahui melalui status dari proses *authentication* yang benilai *Access-Accept.*

Langkah selanjutnya, pengujian *authentication* melalui *captive portal* yang dilakukan untuk menguji proses *authentication* pada sisi *client* jaringan *wireless* yang dapat berjalan dengan baik. Hasil pengujian *authentication* yang dilakukan pada sisi *client* jaringan *wireless* melalui *captive portal* sebagai berikut:

Tabel 3 Hasil Pengujian User Authentication Melalui Captive Portal

Kondisi	Username	Password	Proses Login	
			Sukses	Gagal
Database kosong	-	-		√
Data baru	rusdan	rusdan123	√	
Ubah password	rusdan	rusdan123		√
Password Baru	rusdan	rusdan1234	√	
Case sensitive	rusdan	Rusdan123		√
Case sensitive	rusdan	RUSDAN123		√
Case sensitive	Rusdan	rusdan123		√
Data dihapus	-	-	-	√

Berdasarkan Tabel 3 dari hasil pengujian dapat diuraikan sebagai berikut:

a. *Database* kosong, proses *login username* dan *password* dikosongkan, maka proses *user authentication* akan gagal dilakukan.
b. Data baru, proses *login username* dan *password* dibuat dan tersimpan di *database* RADIUS yaitu *username*: rusdan dan *password*: rusdan123, maka proses *user authentication* akan sukses dilakukan.
c. Ubah *password*, ketika *username* benar namun *password* yang dimasukkan bukan password yang baru yaitu rusdan1234, maka proses *user authentication* akan gagal dilakukan.
d. Password baru, ketika proses *login* menggunakan *username* dan *password* menggunakan password yang baru yaitu rusdan1234, maka proses *user authentication* akan sukses dilakukan.
e. *Case sensitive*, ketika proses *login* menggunakan *username* namun dengan *password* yang salah satu karakternya salah yaitu Rusdan123 yang seharusnya

rusdan123, maka proses *user authentication* akan gagal dilakukan.

f. *Case sensitive*, ketika proses *login* menggunakan *username* namun dengan *password* yang karakternya salah yaitu RUSDAN123 yang seharusnya rusdan123, maka proses *user authentication* akan gagal dilakukan.
g. *Case sensitive*, ketika proses *login* menggunakan *username* yang karakternya salah yaitu Rusdan yang seharusnya rusdan namun dengan *password* benar yaitu rusdan123, maka proses *user authentication* akan gagal dilakukan.
h. Data dihapus, proses *login username* dan *password* dengan menggunakan yaitu *username*: rusdan dan *password*: rusdan123 namun data telah dihapus, maka proses *user authentication* akan gagal dilakukan.

4. KESIMPULAN

Dari penelitian yang telah dilakukan, maka diperoleh kesimpulan bahwa pengembangan *user authentication* berbasis *Remote Access Dial In User Service* (RADIUS) juga menjadi permasalahan yang dihadapi, setelah dilakukan pengujian menggunakan perangkat *smartphone* dan *notebook* dapat disimpulkan bahwa *user authentication* berbasis RADIUS aman dan *user-friendly* yang mampu membedakan *user* atau pengguna yang diizinkan dan tidak diizinkan untuk menggunakan layanan jaringan *wireless*. Kesimpulan tersebut didapatkan dari hasil pengujian yang telah dilakukan pengujian aplikasi *user authentication* berbasis RADIUS dilakukan dengan dua cara pengujian RADIUS *server* dan cara kedua dilakukan pengujian pada sisi *client* melalui *captive portal*. Pengujian *user authentication* pada RADIUS *server* secara umum dapat berjalan dengan baik dalam menangani permintaan *authentication request*, hal ini dapat diketahui melalui status dari proses *authentication* yang benilai *Access-Accept*, sedangkan pengujian *user authentication* melalui *captive portal* yang dilakukan untuk menguji proses *authentication* pada sisi *client* menggunakan perangkat yang mendukung jaringan *wireless*. Hasil pengujian *user authentication* yang dilakukan pada sisi *client* jaringan *wireless* melalui *captive portal* yaitu, ketika *username* dan *password* yang digunakan sama dengan yang tersimpan pada *Database* (DB) RADIUS, maka proses *login* berhasil dilakukan, namun ketika *username* dan *password* salah dan tidak tersimpan pada DB RADIUS, maka proses *login* akan gagal dilakukan atau karakter yang dimasukkan berbeda dengan yang tersimpan pada DB RADIUS, maka proses login akan gagal dilakukan..

REFERENSI

A. Julianto, Migunani, and R. Efendi, "Otentikasi Penggunaan Layanan Wireless LAN Dengan FreeRADIUS dan Chillispot," J. Teknol. Inf. dan Komun., vol. 4, no. 2, pp. 1–10, 2013.

Patrick, Andrew. *Authentication Technologies & Identity Theft.* Canada: Information Security Group, Institute for Information Technology & Department of Psychology, Carleton University. 2007.

D. C. Pramudita, A. Pinandito, and E. S. Pramukantoro, "Otentikasi dan Manajemen Pengguna Hotspot Router Mikrotik Menggunakan RADIUS dan PHP-MySQL," J. Mhs. PTIIK UB, vol. 1, no. 1, p. 11, 2013.

I. Riadi and Wahyu Prio Wicaksono, "Implementasi Quality of Service Menggunakan Metode Hierarchical Token Bucket," JUSI, vol. 1, no. 2, pp. 93–104, 2011.

H. Setiawan, "Rancang Bangun captive Portal Untuk Jaringan Wireless Berbasis Open Source Pada CV. Gempar Production Palembang," STMIK PalComTech, pp. 1–9, 2013.

A. Tenggono, "Desain Dan Implementasi User Authentication Untuk Fasilitas Hotspot STMIK Palcomtech," J. Teknol. DAN Inform., vol. 1, no. 3, pp. 184–201, 2011.

R. P. Tenggario and J. Lukas, "Manajemen Jaringan Wireless Menggunakan Server Radius," J. Tek. Komput., vol. 19, no. 9, pp. 80–87, 2011.

M. Yahya, "Era Industri 4.0: Tantangan Dan Peluang Perkembangan Pendidikan Kejuruan Indonesia," *Orasi Ilm. Profr. Bid. Ilmu Pendidik. Kejuru. Univ. Negeri Makassar*, pp. 1–25, 2018.