

***IMPLEMENTASI SISTEM AUTENTIKASI TERPUSAT PADA WIFI
ACCES POINT DI KAMPUS UNHAS MENGGUNAKAN LDAP***



TUGAS AKHIR

*Disusun dalam rangka memenuhi salah satu persyaratan
Untuk menyelesaikan program Strata-1 Departemen Teknik Informatika*

Fakultas Teknik Universitas Hasanuddin

Makassar

Disusun Oleh:

<u>HANDRI</u> D42114021
--

DEPARTEMEN TEKNIK INFORMATIKA

FAKULTAS TEKNIK

UNIVERSITAS HASANUDDIN

2019

LEMBAR PENGESAHAN SKRIPSI

**“IMPLEMENTASI SISTEM AUTHENTIKASI TERPUSAT
PADA WIFI ACCESS POINT DI KAMPUS UNHAS
MENGUNAKAN LDAP”**

OLEH:

HANDRI

D42114021

Skripsi ini telah dipertahankan pada Ujian Akhir Sarjana tanggal 21 Januari 2020.
Diterima dan disahkan sebagai salah satu syarat memperoleh gelar Sarjana Teknik (ST.)
pada Program Studi S1 Teknik Informatika Departemen Teknik Informatika Fakultas
Teknik Universitas Hasanuddin.

Gowa, 27 Januari 2020

Disetujui oleh:

Pembimbing I,

Dr. Eng Muhammad Niswar, S.R., M.IT
NIP. 19730922 199903 1 001

Pembimbing II,

A. Ais Prayogi, S.T., M.Eng
NIP. 19830510 201404 1 001

Diterima dan disahkan oleh:
Ketua Departemen Teknik Informatika



Dr. Abu-Allah Ilham, S.T., M.IT.
NIP. 19731010 199802 1 001

ABSTRAK

Pada saat ini pengguna *wifi* dikampus Universitas Hasanuddin (UNHAS) sangatlah banyak mulai dari mahasiswa, staff hingga dosen. Dengan adanya *wifi* dapat menawarkan beragam kemudahan dalam berkomunikasi dan mencari informasi. Di kampus Universitas Hasanuddin (UNHAS) semua fakultas memiliki tersendiri server jaringannya sehingga cuman *client* di fakultas tersebut yang dapat menggunakan *wifi*.

Untuk meningkatkan kenyamanan dan efisiensi dari penggunaan *wifi* skala besar maka digunakan sistem autentikasi. Hal ini untuk meningkatkan keamanan dalam penggunaan *wifi* dalam meminimalisir resiko serangan, karena data yang terkirim saat melakukan *login* merupakan data *credential* pengguna.

Pada penelitian ini akan mengimplementasikan *autentikasi* terpusat menggunakan *Lightweight Directory Access Protocol (LDAP)*. Penelitian ini juga menyajikan kerangka pengukuran untuk mendukung berbagai tugas pengukuran system autentikasi yang dibuat, seperti *latency autentikasi dan performansi CPU* pada 50 unit PC. Maka hasil analisis dari implementasi sistem *autentikasi* pada *wifi* unhas adalah *latency autentikasi* selama 232,6 detik dan performansi *CPU* sebesar 25,3%. Sehingga penerapan system berfungsi dengan baik dan *freeradius* digunakan sebagai metode *autentikasi client*.

Kata kunci : *Wifi, autentikasi terpusat, Lightweight Directory Access Protocol (LDAP), FreeRADIUS*

KATA PENGANTAR

Puji dan syukur penulis panjatkan atas kehadiran TuhanYang Maha Esa karena berkat Rahmat dan Karunia-Nya sehingga Tugas Akhir yang berjudul **“IMPLEMENTASI SISTEM AUTENTIKASI TERPUSAT PADA WIFI ACCES POINT DI KAMPUS UNHAS MENGGUNAKAN LDAP”** ini dapat diselesaikan sebagai salah satu syarat dalam menyelesaikan jenjang Strata-1 pada Departemen Teknik Informatika Fakultas Teknik Universitas Hasanuddin.

Penyusunan penelitian ini disajikan hasil suatu penelitian yang menyangkut judul yang telah diangkat dan telah melalui proses pencarian dari berbagai sumber baik jurnal penelitian, *prosiding* pada seminar-seminar nasional/internasional, buku maupun dari situs-situs di internet.

Penulis menyadari bahwa dalam penyusunan dan penulisan skripsi ini tidak lepas dari bantuan, bimbingan serta dukungan dari berbagai pihak, dari masa perkuliahan sampai dengan masa penyusunan tugas akhir, sangatlah sulit untuk menyelesaikan tugas akhir ini. Oleh karena itu, penulis dengan senang hati menyampaikan terima kasih kepada:

- 1) Kedua Orang tua penulis, Bapak Lukas Sampe Lapu. dan Alm. Ibu Katrina Masita serta saudara-saudara penulis yang selalu memberikan dukungan, doa, dan semangat serta selalu sabar dalam mendidik penulis sejak kecil;
- 2) Bapak Dr.Eng. Muhammad Niswar, S.T., M.IT., selaku pembimbing I dan Bapak A. Ais Prayogi, ST., M.Eng, selaku pembimbing II yang selalu

menyediakan waktu, tenaga, pikiran dan perhatian yang luar biasa untuk mengarahkan penulis dalam penyusunan tugas akhir;

- 3) Bapak Dr. Amil Ahmad Ilham, S.T., M.IT., selaku Ketua Departemen Teknik Informatika Fakultas Teknik Universitas Hasanuddin atas bimbingannya selama masa perkuliahan penulis;
- 4) Bapak Dr. Amil Ahmad Ilham, S.T., M.IT., dan Ibu Elly Warni, S.T., M.T yang telah menyempatkan waktunya memberikan saran kepada penulis;
- 5) Bapak Ady Wahyudi Paundu, S.T., M.T. yang telah menyempatkan waktunya memberikan saran kepada penulis;
- 6) Pegawai PTIK, khususnya Bapak Lukman Hakim, Kak Ardyansah dan Kak Rendra yang telah membantu saya selama penelitian;
- 7) Keluarga Departemen Teknik Informatika FT-UH angkatan 2014 atas semua bantuan dan semangat yang diberikan selama ini;
- 8) Para teman-teman dan kakak-kakak lab CCIE, SEIS, CBS, IOT, AIMP yang telah memberikan begitu banyak bantuan selama penelitian, pengambilan data dan diskusi *progress* penyusunan Tugas Akhir;
- 9) Teman KMKT 2014 atas dukungan dan semangat yang diberikan selama ini;
- 10) Teman KMKO 2014 atas dukungan dan semangat yang diberikan selama ini;
- 11) Teman Alang Squad atas dukungan dan semangat yang diberikan selama ini;
- 12) Teman-teman Rectifier FT UH atas dukungan dan semangat yang diberikan selama ini;
- 13) Teman KKN 99 TMMD Kepulauan Selayar atas dukungan dan semangat yang diberikan selama ini;

14) Segenap Staf Departemen Teknik Informatika Fakultas Teknik Universitas

Hasanuddin yang telah membantu penulis.

15) Orang-orang berpengaruh lainnya yang tanpa sadar telah menolong dan menjadi inspirasi penulis.

Akhir kata, penulis berharap semoga Allah SWT. berkenan membalas segala kebaikan dari semua pihak yang telah banyak membantu. Semoga Tugas Akhir ini dapat memberikan manfaat bagi pengembangan ilmu. Aamiin.

Gowa, November 2019

Penulis

DAFTAR ISI

LEMBAR PENGESAHAN SKRIPSI	Error! Bookmark not defined.
ABSTRAK	i
KATA PENGANTAR.....	ii
DAFTAR ISI.....	v
DAFTAR GAMBAR.....	ix
DAFTAR TABEL.....	xi
BAB I.....	1
1.1. Latar Belakang	1
1.2. Rumusan Masalah	2
1.3. Tujuan Penelitian	3
1.4. Batasan Masalah.....	3
1.5. Manfaat Penelitian	3
1.6. Sistematika Penulisan.....	4
BAB II	5
2.1 Eduroam	5
2.2 Autentikasi Terpusat	6
2.3 <i>Wifi</i>	6
2.4 <i>Proxmox</i>	8
2.5 RADIUS (<i>Remote Access Dial-in User Service</i>).....	9

2.6.1. <i>FreeRadius</i>	10
2.6 LDAP (<i>Lightweight Directory Access Protocol</i>)	12
2.6.1. Fusion Directory.....	13
2.6.2. PhpLdapAdmin	14
2.7 <i>Web Service</i>	14
2.7.1 Apache.....	14
2.7.2 PHP	15
2.8 <i>Putty</i>	16
2.9 Keamanan.....	17
2.9.1. EAP	17
2.9.2. TLS/ SSL.....	17
2.9.3. EAP-TLS.....	18
2.9.4. EAP-TTLS	19
2.9.5. Protokol 802.1X	20
2.10. <i>Wireless LAN Controller (WLC)</i>	20
BAB III	22
3.1 Lokasi dan Waktu Penelitian	22
3.2 Instrumen Penelitian	22
3.3 Prosedur Penelitian.....	23
3.4 Perancangan Sistem	25

3.4.1	Instalasi FreeRadius	28
3.4.2	Instalasi OpenLdap.....	32
3.4.3	Setup Fusion Directory	35
3.4.4.	Testing.....	40
3.5	Skenario Pengujian.....	46
3.5.1	Pengujian Latency <i>Authentikasi</i>	46
3.5.2	Pengujian Penggunaan <i>CPU</i>	47
BAB IV	47
4.1	Hasil Penelitian	48
4.1.1	Hasil Pengujian Latency <i>Authentikasi</i>	48
4.1.2	Pengujian Penggunaan <i>CPU</i>	50
4.2	Pembahasan.....	53
4.2.1	Lantency <i>Authentikasi</i>	53
4.2.2	Performansi <i>CPU</i>	53
BAB V	55
5.1	Kesimpulan	55
5.2	Saran.....	55
DAFTAR PUSTAKA	57
LAMPIRAN	58

DAFTAR GAMBAR

Nomor	halaman
Gambar 2.1	11
Gambar 2.2	12
Gambar 2.3	14
Gambar 3.1	20
Gambar 3.2	22
Gambar 3.3	22
Gambar 3.4	23
Gambar 3.5	26
Gambar 3.6	26
Gambar 3.7	28
Gambar 3.8	28
Gambar 3.9	30
Gambar 3.10	31
Gambar 3.11	31
Gambar 3.12	32
Gambar 3.13	33
Gambar 3.14	33
Gambar 3.15	34
Gambar 3.16	35
Gambar 3.17	36
Gambar 3.18	37

Gambar 3.19	39
Gambar 3.20	39
Gambar 3.21	40
Gambar 3.22	40
Gambar 3.23	41
Gambar 3.24	42
Gambar 4.1	44
Gambar 4.2	45
Gambar 4.3	46
Gambar 4.4	47
Gambar 4.5	48

DAFTAR TABEL

Nomor	halaman
Tabel 4.1	45
Tabel 4.2	48

BAB I

PENDAHULUAN

1.1. Latar Belakang

Saat ini banyak cara dilakukan untuk memanfaatkan jaringan internet seperti *wifi* dan menjadi salah satu bagian terpenting dalam dunia kampus. Pada saat ini pengguna *wifi* dikampus Universitas Hasanuddin (UNHAS) sangatlah banyak mulai dari mahasiswa, staff hingga dosen. Dengan adanya *wifi* dapat menawarkan beragam kemudahan dalam berkomunikasi dan mencari informasi. Kebanyakan pengguna bila tiba dikampus mereka segera menyambungkan perangkat mereka dengan *wifi* kampus. Agar seluruh *client* di kampus Universitas Hasanuddin (UNHAS) dapat mengakses *wifi* maka digunakan access point, sebuah access point bisa menjangkau satu atau beberapa ruangan tergantung *radius* jangkauannya. Pada kampus Universitas Hasanuddin (UNHAS) tiap-tiap fakultas memiliki tersendiri server jaringannya sehingga cuman *client* di fakultas tersebut yang dapat menggunakannya. Masih ada beberapa kelemahan yang terjadi pada *wifi* dikampus yang dimana seseorang dapat melihat lalu lintas data pada jaringan, sehingga orang tersebut dapat menangkap data seperti *username* dan *password* dan menggunakannya.

Untuk mengatasi beberapa kelemahan diatas maka diterapkan autentikasi terpusat menggunakan protokol *Lightweight Directory Access Protocol (LDAP)*. *Lightweight Directory Access Protocol (LDAP)* adalah sebuah protocol *client-server* yang digunakan untuk mengakses suatu

directory service. LDAP menggunakan model client-server, dimana client mengirimkan identifier data kepada server menggunakan protokol TCP/IP dan server mencoba mencarinya pada DIT (Directory Information Tree) yang tersimpan di server. Bila ditemukan maka *client* diperbolehkan terhubung ke jaringan *wifi* dan apabila tidak ditemukan maka *client* diminta memasukan ulang identifier data. Jadi dengan menggunakan *Lightweight Directory Access Protocol (LDAP)* pada autentikasi terpusat *wifi access point* kampus maka keamanannya lebih aman dan kita dapat terhubung ke *wifi* selama dalam jangkauan *access point* walaupun di fakultas mana berada.

1.2. Rumusan Masalah

Berdasarkan latar belakang, maka rumusan masalah pada tugas akhir ini adalah:

1. Bagaimana menerapkan autentikasi terpusat pada akses *wifi access point* dengan menggunakan *lightweight directory access protocol (LDAP)* ?
2. Bagaimana kinerja dari sebuah *wifi access point* yang menggunakan *lightweight directory access protocol (LDAP)* ?
3. Apa keamanan yang diberikan apabila menggunakan *lightweight directory access protocol (LDAP)* ?

1.3. Tujuan Penelitian

1. Untuk mempelajari dan menerapkan sistem autentikasi terpusat pada *wifi access point* di kampus dengan menggunakan *lightweight directory access protocol (LDAP)*.
2. Untuk mengetahui kinerja *Lightweight Directory Access Protocol(LDAP)* pada *wifi access point* di kampus.
3. Untuk mengetahui keamanan apa yang di berikan menggunakan *lightweight directory access protocol (LDAP)*.

1.4. Batasan Masalah

Yang menjadi batasan masalah dalam tugas akhir ini adalah :

1. Membuat autentikasi pengguna *wifi* yang berbasis *lightweight directory access protocol (LDAP)*, yang mana Sistem ini akan diterapkan pada Sistem Operasi Linux Ubuntu 18.04.
2. Membuat *user* yang bisa *login* pada jaringan *wifi* yang dirancang.
3. Perangkat yang digunakan dalam membuat sistem adalah Laptop Asus dengan memori RAM 2Gb dan *Processor Intel Core i3 4030U CPU @1.90GH*, *Access Point* Cisco dan *Switch*.

1.5. Manfaat Penelitian

Manfaat dari tugas akhir ini adalah :

1.5.1. Bagi Peneliti

Penelitian ini diharapkan mampu menambah pengetahuan peneliti dalam mengimplementasikan sistem autentikasi terpusat menggunakan *lightweight directory access protocol (LDAP)*.

1.5.2. Bagi *Client*

Penelitian ini diharapkan memberikan kenyamanan dan keamanan bagi *Client* dalam menggunakan *wifi*.

1.5.3. Bagi Pendidikan

Penelitian ini diharapkan mampu menjadi acuan bagi penelitian selanjutnya terutama dalam keamanan *Wifi*.

1.6. Sistematika Penulisan

BAB I PENDAHULUAN

Bab ini berisi latar belakang masalah, rumusan masalah, tujuan penelitian, manfaat penelitian, batasan masalah, sistematika penulisan.

BAB II TINJAUAN PUSTAKA

Pada bab ini akan dijelaskan teori-teori yang menunjang percobaan yang dilakukan.

BAB III METODOLOGI PENELITIAN

Bab ini berisi analisis kebutuhan sistem, perancangan sistem, dan skenario pengujian.

BAB IV HASIL DAN PEMBAHASAN

Bab ini berisi hasil penelitian dan pembahasan penjabaran dari penelitian yang dilakukan.

BAB V PENUTUP

Bab ini berisi kesimpulan dari hasil penelitian dan saran.

BAB II

TINJAUAN PUSTAKA

2.1 Eduroam

Eduroam (education roaming) adalah layanan roaming WLAN antar lembaga akademis dan lembaga penelitian di seluruh dunia. Eduroam memberikan akses layanan internet yang aman kepada pengguna dari sebuah institusi yang berpartisipasi di dalam layanan eduroam ketika pengguna tersebut berkunjung ke institusi lain yang juga berpartisipasi di dalam layanan eduroam. Pengguna hanya perlu menggunakan kredensial dari institusi asalnya untuk proses *otentikasi*. Untuk proses otorisasi dilakukan oleh institusi tempat pengguna tersebut berkunjung. Setelah mendapatkan izin, maka pengguna tersebut akan mendapatkan ip dan dapat melakukan akses internet melalui firewall dan server proxy dari institusi tempat ia berkunjung (Ubaidillah).

Inisiatif eduroam awalnya dimulai pada tahun 2002. TFMobility Klaas Wierenga dari SURFnet yang merupakan bagian dari TERENA (Tran-European Research and Education Networking Association), yang saat ini berganti nama menjadi GÉANT, menyatukan sebuah infrastruktur berbasis RADIUS dengan protokol IEEE 802.1x untuk akses internet roaming antar institusi di Eropa. Tujuan dari berdirinya TERENA adalah untuk mempromosikan dan berpartisipasi dalam pengembangan infrastruktur informasi dan telekomunikasi internasional berkualitas tinggi untuk kepentingan penelitian dan pendidikan. Langkah apapun yang

diperlukan akan diambil untuk menunjukkan bahwa infrastruktur yang akan di kembangkan didasarkan pada standar yang terbuka dan menggunakan teknologi paling canggih yang tersedia. Selama masa pengembangannya, pada tahun 2003 sudah mulai banyak institusi di Eropa yang mulai bergabung. Hingga pada tahun 2004, Australia menjadi negara non-Eropa pertama yang terhubung di dalam layanan eduroam. Sampai saat ini, sudah terdapat 89 operator roaming yang tersebar di seluruh dunia yang memberikan akses layanan eduroam (Ubaidillah).

2.2 Autentikasi Terpusat

Autentikasi yaitu proses pengesahan identitas pengguna (*end user*) untuk mengakses jaringan. Proses ini diawali dengan pengiriman kode unik misalnya, *username, password, pin*, sidik jari oleh pengguna kepada server. Di sisi server, sistem akan menerima kode unik tersebut, selanjutnya membandingkan dengan kode unik yang disimpan dalam *database* server. Jika hasilnya sama, maka server akan mengirimkan hak akses kepada pengguna. Namun jika hasilnya tidak sama, maka server akan mengirimkan pesan kegagalan dan menolak hak akses pengguna (Yesi Novaria Kunang, 2008).

2.3 Wifi

Wifi (Wireless Fidelity) merupakan salah satu varian teknologi komunikasi dan informasi yang bekerja pada jaringan dan perangkat Wireless Local Area Network (WLAN). Sesuai dengan namanya, perangkat yang dibutuhkan untuk mengakses internet dengan layanan ini juga

nirkabel. Jika dibandingkan dengan internet lainnya, *Wifi* lebih mudah instalasinya. Namun, pastinya harus ada perangkat utama seperti wireless atau access point dan jaringan internet (Jori David Joseph,2016).

Layanan ini umumnya diperuntukan bagi tempat-tempat umum dengan aksesibilitas yang tinggi seperti pusat perbelanjaan, hotel, kafe, kampus, dan sebagainya. Layanan internet jenis ini dikenal pula dengan istilah hotspot. Untuk mengaksesnya diperlukan gadget yang memiliki fasilitas *Wifi* seperti laptop, netbook/notebook, PDA, atau ponsel (Jori David Joseph,2016).

Kecepatan akses internet wireless tergolong tinggi dan bisa mencapai 54 Mbps. Selain itu koneksi cenderung stabil sehingga proses pengaksesan layanan internet bisa dilakukan dengan lancar (Jori David Joseph,2016).

Jaringan *Wifi* memiliki standar yang digunakan pada jaringan lokal nirkabel *Wireless Local Area Network* dan *Metropolitan Area Network* (WLAN, MAN) yang didasari pada spesifikasi standar *Institute of Electrical and Electronic Engineers* (IEEE) 802.11, standarisasi jaringan *Wifi* menggunakan spesifikasi IEEE 802.11 a/b/g [2]. Terdapat tiga metode autentikasi keamanan jaringan *Wifi* antara lain *Wired Equivalent Privacy* (WEP), *Wireless Protected Access-Pre Shared Key* (WPA-PSK), dan WPA2-PSK (Jori David Joseph,2016).

Wifi memiliki kelebihan dan kekurangan. Kelebihan yang dimiliki oleh *wifi* antara lain adalah dengan menggunakan *wifi* kita lebih praktis

karena penggunaannya bisa berpindah-pindah tempat serta hampir semua perangkat teknologi sudah support dengan jaringan *wifi*. Kekurangannya adalah keamanan data yang harus lebih diperhatikan dan sekalipun anda bisa bebas bergerak namun perhatikan jangkauan *wifi*nya karena semakin jauh jangkauannya maka semakin lamban pula koneksinya (NH Sari,2018).

2.4 *Proxmox*

Proxmox adalah sebuah distro linux virtualisasi berbasis debian 64 bit yang mendukung Openvz dan KVM. Proxmox memungkinkan untuk melakukan manajemen terpusat dari banyak server fisik. Sebuah proxmox terdiri dari minimal satu master local dan beberapa node (Fadjrin,2013)

Proxmox merupakan software open source virtualization platform untuk menjalankan virtual appliance dan virtual machine. Proxmox VE (Virtual Environment) adalah distro khusus yang didekasikan secara khusus sebagai mesin virtualisasi yaitu KVM dan OpenVZ. (Suryono,2012).

Proxmox VE menggunakan Container Virtualization dan Full Virtualization:

- Container Virtualization (OpenVZ) merupakan teknologi yang disarankan untuk menjalankan server linux. OpenVZ membuat beberapa container yang secure dan terisolasi (disebut juga CT,VE atau VPS). Setiap Container melakukan dan mengeksekusi persis seperti layaknya sebuah *stand alone server*, sebuah container dapat di-reboot secara independen dan memiliki akses *super user ,IP*

address, memori, proses,file, aplikasi, system library dan konfigurasi tersendiri (Suryono,2012).

- Full Virtualization (KVM) merupakan singkatan dari (Kernel-based Virtual Machine) adalah solusi virtualisasi penuh untuk hardware berbasis x86 yang memiliki ekstensi virtualisasi (Intel VT atau AMDV CPU). Setiap virtual machine memiliki hardware pribadi yang virtual: network card,disk, adapter grafis,dll.KVM mirip dengan XEN akan tetapi KVM merupakan bagian dari Linux dan menggunakan system scheduler dan memory managemen regular dari Linux (Suryono,2012).

2.5 RADIUS (*Remote Access Dial-in User Service*)

RADIUS (*Remote Access Dial-in User Service*) adalah salah satu mekanisme untuk melakukan akses kontrol dalam mengecek dan melakukan autentikasi (*authentication*) *user* atau pengguna yang berdasarkan pada mekanisme autentikasi yang sebelumnya sudah banyak dilakukan, yaitu menggunakan metode *response/challenge*. *Remote Access Dial-in User Service* (RADIUS) dikembangkan pada pertengahan tahun 1990 oleh Livingstone Enterprise (sekarang Lucent Technologies). Pada awal perkembangannya RADIUS menggunakan *port* 1645, namun *port* tersebut bentrok dengan layanan *datametrics*. Sehingga *port* RADIUS diganti dan sekarang *port* yang digunakan oleh RADIUS adalah *port* 1812 dengan format standarnya ditetapkan pada RFC (*Request for Command*) 2138 (Muhammad Asfiandi,2014).

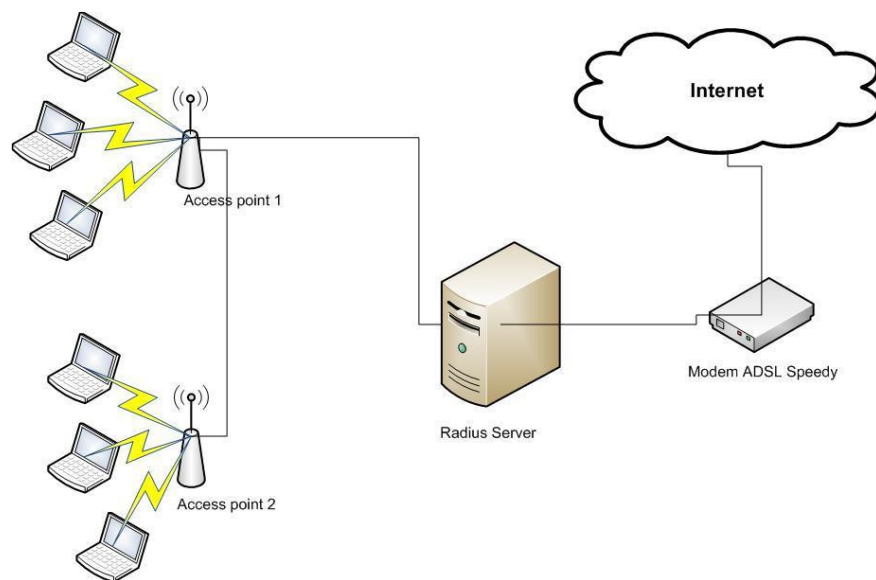
Server RADIUS memiliki mekanisme keamanan untuk melakukan autentikasi dan otorisasi kepada pengguna sebelum melakukan koneksi. Pada saat computer client ingin melakukan koneksi ke jaringan, maka server RADIUS akan melakukan autentikasi terlebih dahulu dengan meminta identitas pengguna yang merupakan *username* dan *password* untuk dicocokkan dengan data yang terdapat dalam database server RADIUS. Selanjutnya RADIUS akan menentukan apakah pengguna dapat menggunakan layanan jaringan komputer atau tidak. Jika proses autentikasi sukses dilakukan maka akan dilakukan proses pelaporan, yaitu mencatat semua aktivitas koneksi yang dilakukan oleh pengguna di jaringan, baik menghitung durasi waktu koneksi dan jumlah transfer data yang dilakukan oleh pengguna didalam jaringan. Proses pelaporan server RADIUS terhadap pengguna jaringan bias dalam bentuk waktu (detik, menit, jam) maupun dilakukan dalam bentuk besar transfer data (*Byte, Kbyte, Mbyte*) (Muhammad Asfiandi,2014).

2.6.1. FreeRadius

FreeRADIUS merupakan modular yang dikembangkan untuk dapat bekerja dengan performa tinggi dan didistribusikan dibawah lisensi GNU (*General Public License*). *FreeRADIUS* bisa diunduh dan digunakan secara gratis. Meskipun gratis, didalam *FreeRADIUS* sudah mengandung RADIUS server, PAM, module Apache dan banyak tambahan *tool* lainnya (Muhammad Asfiandi,2014).

FreeRADIUS adalah RADIUS server yang paling banyak digunakan didunia dan menjadi RADIUS server yang paling populer dikalangan Open Source. *FreeRADIUS* suport dengan semua protokol autentikasi dan dilengkapi dengan web administrasi pengguna berbasis PHP yang disebut *dialupadmin*. *FreeRADIUS* juga mengandung AAA yang dibutuhkan oleh banyak perusahaan seperti perusahaan telekomunikasi. *FreeRADIUS* juga merupakan server yang cepat dan memiliki banyak fitur (Muhammad Asfiandi,2014).

Alasan utama banyak yang memilih *FreeRADIUS* adalah mahalnya harga RADIUS server komersial. Sebagai contoh: harga *Interlink's Secure XS* mulai dari \$2375 untuk 250 pengguna, *Funk Odyssey* server \$2500, *VOP Radius Small Business* mulai dari \$995 untuk 100 pengguna. Harga RADIUS server komersial tersebut tidak akan terjangkau untuk pengguna hotspot, terutama untuk kampus (Muhammad Asfiandi,2014).



Gambar 2.1. Alur kerja Radius

Pada gambar 2.1 dimana *user* akan diminta identitas, seperti *username* dan *password*, untuk dapat dicocokkan dengan identitas yang telah dimasukan pada server Radius pada waktu sebelumnya, apabila identitas tersebut terdapat pada server Radius dan identitas yang dimasukan benar maka *user* langsung dapat mengakses internet dengan menggunakan jaringan *wireless (hotspot)* tersebut.

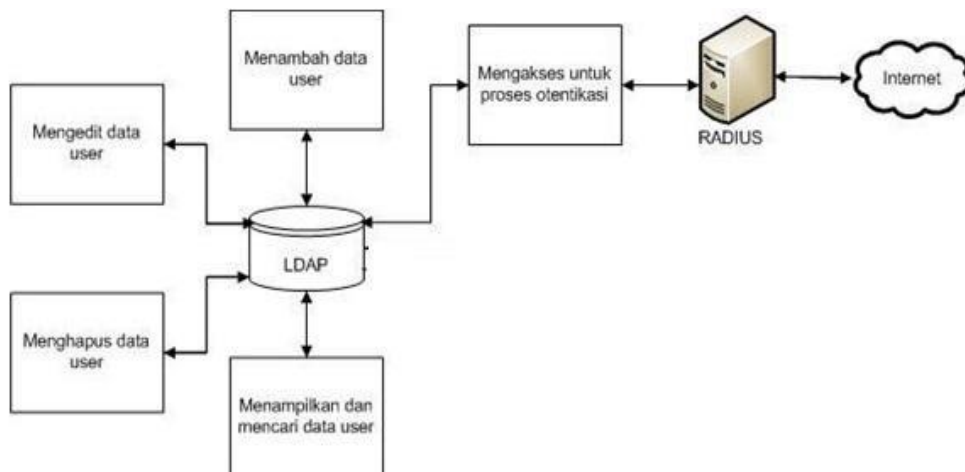
2.6 LDAP (*Lightweight Directory Access Protocol*)

LDAP adalah sebuah protokol yang mengatur mekanisme pengaksesan layanan direktori (Directory Service) yang dapat digunakan untuk mendeskripsikan banyak informasi seperti informasi tentang people, organizations, roles, services dan banyak entitas lainnya (Aprilia Ayu Mahardani, 2017).

LDAP menggunakan model client-server, dimana client mengirimkan identifier data kepada server menggunakan protokol TCP/IP dan server mencoba mencarinya pada DIT (Directory Information Tree) yang tersimpan di server. Bila di temukan maka hasilnya akan dikirimkan ke client tersebut namun bila tidak maka hasilnya berupa pointer ke server lain yang menyimpan data yang di cari (Aprilia Ayu Mahardani, 2017).

LDAP tersebut memiliki bentuk struktur yang berhirarki, bukannya berformat kolom dan baris, seperti halnya database normal, sehingga memudahkan untuk memasukkan sejumlah besar detail yang mirip dalam bentuk yang terorganisir. Awalnya, server LDAP merupakan sesuatu yang

terdapat diantara LDAP client dan sebuah server DAP (X 500), jadi untuk mengurangi resource yang dibutuhkan menjalankan client (Aprilia Ayu Mahardani, 2017).



Gambar 2.2. Alur kerja Ldap

Perancangan dari Gambar 2.2 dapat dilihat hubungan antara LDAP dengan RADIUS. Saat user akan mengakses perangkat, user harus memasukkan ID (username dan password) yang akan diproses oleh RADIUS. RADIUS akan mengambil data dari LDAP saat melakukan otentikasi ID user. Setelah ID terotentikasi, RADIUS akan melakukan proses otorisasi pemberian izin akses untuk user.

2.6.1. Fusion Directory

Fusion Directory adalah aplikasi web di bawah lisensi *GNU (General Public License)* dikembangkan di PHP untuk dengan mudah mengelola direktori *LDAP* dan semua layanan terkait (Samsyul Hdidayat, 2012).

2.6.2. PhpLdapAdmin

PhpLDAPAdmin adalah software opensource yang digunakan untuk mempermudah dalam pembuatan database di server LDAP. Dengan phpLDAPAdmin kita tidak perlu bersusah payah mondar-mandir ke terminal, sehingga cukup dengan membuka browser dan hanya sesekali melakukan cek ulang via terminal untuk memastikan apakah data yang kita masukkan benar atau salah (Samsyul Hdidayat, 2012).

2.7 Web Service

Web Service adalah aplikasi yang dibuat agar dapat dipanggil dan diakses oleh aplikasi lain melalui internet dengan menggunakan format pertukaran data sebagai format pengiriman pesan, menurut (Yusrizal, 2017).

Web services merupakan sebuah sistem terdistribusi memiliki komponen yang dapat di-deploy dan diakses menggunakan protokol HTTP (Hyper Text Transport Protocol) maupun HTTPS (HTTP Secure). Layanan web dapat di program dalam berbagai bahasa pemograman yang ada. Pada web services sekurang-kurangnya terdapat sebuah web server (jaringan penyedia layanan) dan sebuah klien. Klien meminta layanan yang ditawarkan oleh web server bisa melalui desktop/PC maupun mobile (Yusrizal, 2017).

2.7.1 Apache

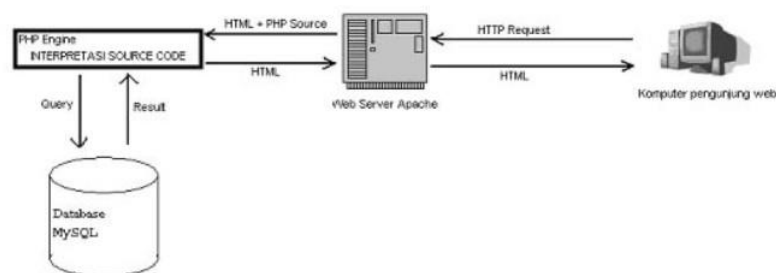
Apache merupakan salah satu server web yang paling banyak digunakan di dunia, beberapa keunggulan Apache dibandingkan dengan

web server yang lain seperti IIS (Internet Information Service) dari Microsoft adalah kemampuannya untuk mendukung berbagai bahasa script paling populer seperti PHP (Personal Home Page) dan JSP (Java Server Pages). Hal lain yang membuat Apache lebih diminati adalah sistem lisensinya yang gratis sehingga mengurangi biaya yang perlu dikeluarkan dalam membangun situs web dinamis (Andi Wahyu,2014).

Apache Web Server merupakan program untuk menjalankan web dalam sebuah komputer. Web service ini akan melayani setiap permintaan dari web browser dan mengirimkan data dalam bentuk html yang bisa dibaca oleh web browser dari pengguna computer (Andi Wahyu,2014).

2.7.2 PHP

PHP ialah bahasa skrip dalam server (*server-side embedded scripting language*). Artinya, PHP bekerja di dalam HTML dengan tugas membuat isi dokumen sesuai permintaan.



Gambar 2.3. Diagram Alur Kerja PHP

Pada gambar di atas tampak alur kerja *engine* PHP. Pertama-tama, sebuah komputer pengunjung web melakukan HTTP request terhadap halaman tertentu. Jika halaman yang *direquest* tersebut ialah halaman PHP, maka Web Server *Apache* akan meneruskan halaman PHP tersebut ke PHP *engine*. PHP *engine* akan melakukan interpretasi terhadap *source* PHP dalam halaman tersebut, dengan bantuan dari *database* jika perlu. Setelah interpretasi selesai, maka PHP akan mengembalikan hasilnya yang berupa HTML murni (tanpa adanya *source* PHP sama sekali) kepada Web Server *Apache*. Setelah itu Web Server *Apache* akan meneruskannya kepada komputer pengunjung web tadi dalam bentuk HTML yang dapat ditampilkan oleh internet browser (Ihsan Naskah, 2011).

2.8 *Putty*

Putty adalah sebuah program open source yang dapat Anda gunakan untuk melakukan protokol jaringan SSH, Telnet dan *Rlogin*. Aplikasi ini merupakan aplikasi portable sehingga tidak perlu di install. Protokol ini dapat digunakan untuk menjalankan sesi remote pada sebuah komputer melalui sebuah jaringan, baik itu LAN, maupun internet. Program ini banyak digunakan oleh para pengguna komputer tingkat menengah ke atas, yang biasanya digunakan untuk menyambungkan, mensimulasi, atau mencoba berbagai hal yang terkait dengan jaringan. Program ini juga dapat Anda gunakan sebagai tunnel di suatu jaringan (Andi,2010).

2.9 Keamanan

2.9.1. EAP

EAP adalah sebuah *framework* autentikasi hasil pengembangan IEEE yang berfungsi secara fleksibel. EAP sendiri bukanlah mekanisme autentikasi secara spesifik, EAP hanya menyediakan fungsi *transport* untuk membawa informasi autentikasi yang disediakan oleh *EAP method*. Dengan begitu, jika ada mekanisme autentikasi (*EAP method*) baru, tidak perlu melakukan *upgrade* pada semua peralatan jaringan. Saat ini terdapat lebih dari 40 *EAP method*, namun yang memenuhi standar untuk bekerja di jaringan nirkabel sebagaimana yang dijelaskan pada *Request For Comments* (RFC) 4017 dan telah disertifikasi oleh Wi-Fi Alliance hanya ada tujuh saja, termasuk di antaranya adalah EAP-TLS, EAPTTLS, dan EAP-PEAP (Azhari Harahap, 2011).

2.9.2. TLS/ SSL

TLS (*Transport Security Layer*) dan pendahulunya SSL (*Secure Socket Layer*) adalah protokol kriptografi yang menyediakan keamanan dalam berkomunikasi melalui jaringan. Dalam RFC 2246, dinyatakan bahwa TLS menyediakan keamanan dalam tiga hal :

1. *Mutual authentication* antara klien dan server dengan *public key cryptography* berdasarkan *digital signatures*. Dengan ini identitas klien/ server dapat dibuktikan dan pemalsuan pesan dapat dihindari. Algoritme kriptografi kunci publik yang sering digunakan adalah

RSA (Rivest, Shamir, Adleman) dan DSA (*Digital Signature Algorithm*).

2. Menjaga kerahasiaan data dengan fungsi kriptografi simetrik untuk melakukan enkripsi/ dekripsi data sehingga mencegah pihak ketiga untuk melakukan *eavesdropping*. Algoritme kunci simetrik yang sering digunakan adalah AES (*Advanced Encryption Standard*) dan 3DES (*Triple DES*).
3. Menghasilkan *Message Authentication Code* (MAC) melalui fungsi *hash* untuk mendeteksi adanya gangguan dan menjaga integritas data. Algoritme *hash* yang sering digunakan adalah MD5 (*Message-Digest Algorithm*) dan SHA-1 (*Secure Hash Algorithm*) (Azhari Harahap, 2011).

2.9.3. EAP-TLS

EAP-TLS merupakan standar EAP *method* pertama yang digunakan dalam jaringan nirkabel serta menyediakan banyak dukungan dari *vendor*. Menurut Sankar *et al* (2004), pada implementasinya EAP-TLS membutuhkan sertifikat digital dari sisi klien dan juga server. Perlunya sertifikat digital di sisi klien merupakan kelebihan sekaligus kekurangan dari EAP-TLS, karena setiap klien membutuhkan sertifikat digital sehingga manajemen dan distribusi dari sertifikat digital ke semua klien membutuhkan sumber daya administrasi tambahan. Di lain pihak, kelebihanannya adalah pada sisi keamanan dimana pihak ketiga

tidak akan bisa melakukan penyerangan tanpa adanya sertifikat digital (Azhari Harahap, 2011).

2.9.4. EAP-TTLS

Sama seperti EAP-TLS, EAP-TTLS (EAP*Tunneled* TLS) juga menggunakan protocol TLS. EAP-TTLS dikembangkan oleh Funk Software dan Certicom. Walaupun memiliki banyak dukungan di berbagai *platform* system operasi seperti Linux dan Mac OS, namun Microsoft Windows tidak memiliki dukungan secara *default*, dibutuhkan aplikasi *third party* agar EAP-TTLS bisa dijalankan. Pada EAP-TTLS, sertifikat digital di sisi klien bersifat *optional*, artinya tidak harus ada. Proses autentikasi dalam EAP-TTLS terbagi dalam dua langkah utama (Azhari Harahap, 2011):

1. Pembentukan *secure tunnel*, melalui sertifikat digital dari server, proses ini mirip dengan EAP-TLS.
2. Proses autentikasi, yang sering disebut juga *inner authentication*. Autentikasi menggunakan metode autentikasi lain seperti PAP (*Password Authentication Protocol*), CHAP (*Challenge-Handshake Authentication Protocol*), MS-CHAP (Microsoft-CHAP), atau bahkan EAP *method* lain. Data yang dikirim akan di enkripsi oleh *secure tunnel* yang dibentuk di langkah 1. EAP-TTLS mendukung penyembunyian identitas karena data dilewatkan melalui *secure tunnel* (Azhari Harahap, 2011).

2.9.5. Protokol 802.1X

Institute of Electrical and Electronics Engineers (IEEE) 802.1X merupakan standar yang digunakan untuk memberikan autentikasi dan otorisasi ke perangkat yang telah terhubung melalui port Local Area Network (LAN) secara fisik untuk menetapkan autentikasi point-to-point. Sehingga keamanan jaringan berbasis kabel dalam perusahaan atau organisasi dapat ditingkatkan (CISCO, 2011). Ada banyak implementasi yang telah dilakukan dengan metode autentikasi yang berbeda yang digunakan di standar IEEE 802.1X salah satunya menggunakan EAP-PEAP yang pengimplementasiannya cukup mudah dan memiliki tingkat keamanan yang cukup baik. Adanya implementasi autentikasi pengguna jaringan bertujuan untuk mencegah dan mengurangi adanya tindak kejahatan di jaringan

2.10. *Wireless LAN Controller (WLC)*

Wireless LAN Controller (WLC) adalah sebuah perangkat yang mengarahkan atau mengatur lalu lintas pada jaringan *wireless*. WLC biasanya digunakan pada jaringan yang berskala menengah ke besar. Tujuannya adalah untuk memudahkan pengontrolan banyak access point. Split MAC adalah kemampuan untuk membagi tugas antara AP dan WLC. AP bertugas untuk menyediakan layanan 802.11 secara realtime, misalnya menyebarkan beacon, ssid, dan merespon request. Tugas AP berikutnya adalah mengirim ACK kepada client ketika client mengirim data. Tugas

WLC pada split MAC adalah authentication, authorization, policy, dan sebagainya.

AP dan WLC berkomunikasi menggunakan protokol *CAPWAP*. *CAPWAP* adalah singkatan dari *Control and Provisioning of Wireless Access Point*. Ketika client mengirimkan data melalui wireless, frame 802.11 yang diterima AP dari client akan di enkapsulasi kedalam *CAPWAP* untuk diteruskan ke WLC. WLC juga memiliki kemampuan untuk merubah kekuatan sinyal dari AP untuk mengakomodasi coverage hole detection.

BAB III

METODOLOGI PENELITIAN

3.1 Lokasi dan Waktu Penelitian

Proses penelitian dilakukan sejak bulan Maret 2019. Dan lokasi penelitian dilakukan di Pusat Teknologi Informasi dan Komunikasi Universitas Hasanuddin dan LAB Cloud Computing.

3.2 Instrumen Penelitian

Instrumen yang digunakan pada penelitian ini terdiri dari Hardware dan Software.

1) *Hardware*

Hardware atau perangkat keras yang dibutuhkan untuk mengimplementasikan rancangan sistem adalah :

- a) Server
- b) WLC (Wireless LAN Controller)
- c) Access Point
- d) PC
- e) Laptop

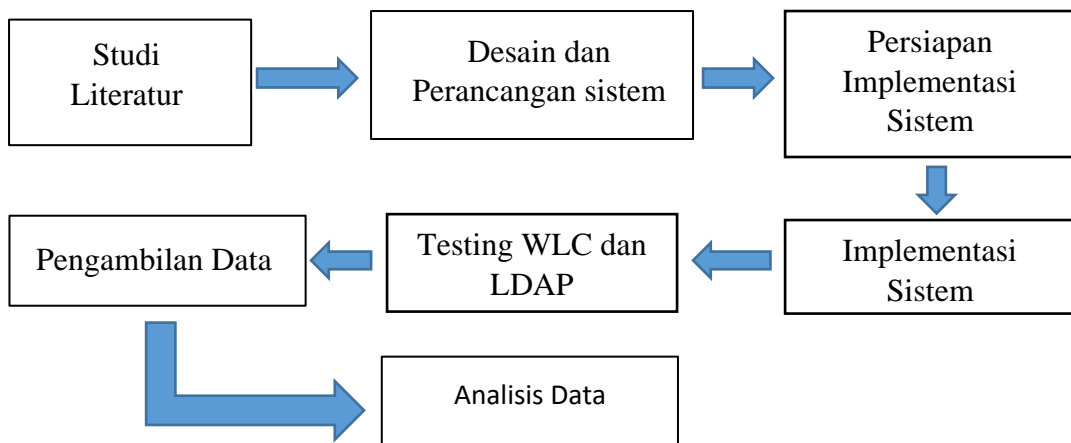
2) Software

- a) Proxmox
- b) Ubuntu Server 18.04
- c) Windows 10
- d) Freeradius

- e) LDAP (*Lightweight Directory Access Protocol*)
- f) Fusion Directory
- g) PhpLdapAdmin
- h) Web Service
- i) Apache
- j) Putty

3.3 Prosedur Penelitian

Tahapan pada penelitian ditunjukkan pada gambar 3.1



Gambar 3. 1. Diagram Tahapan Penelitian

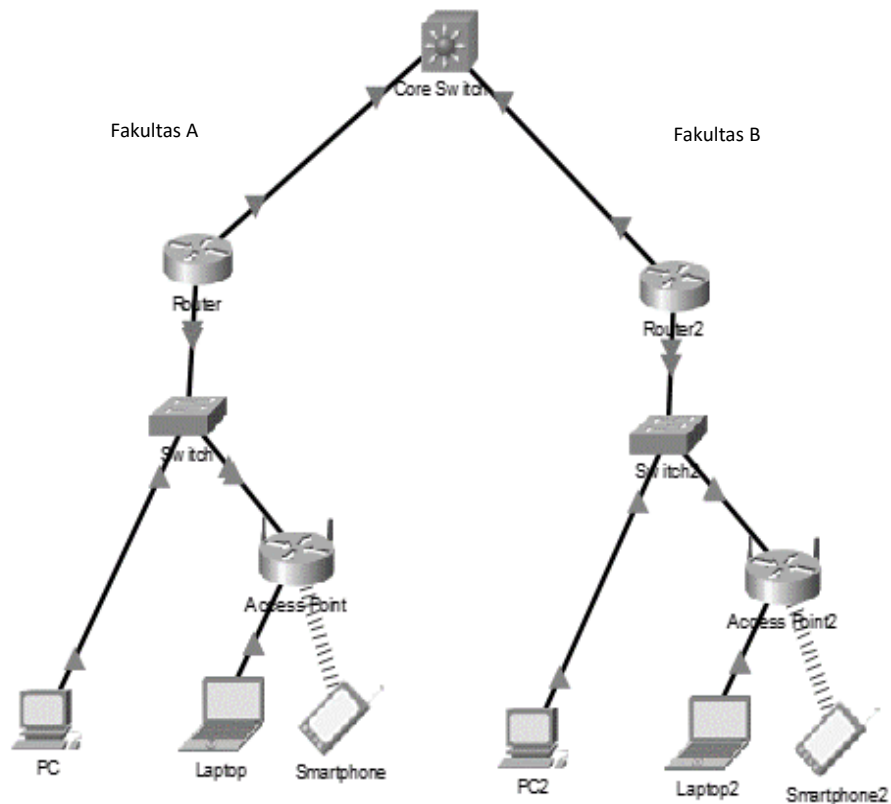
Tahapan secara garis besar dijelaskan sebagai berikut.:

1. Pada studi literatur dilakukan pencarian penelitian terkait dengan LDAP (*Lightweight Directory Access Protocol*) dan juga penelitian sebelumnya yang berkaitan dengan topik penelitian yang dapat menunjang kegiatan penelitian.
2. Pada tahap ini, dilakukan desain rancangan sistem yang akan dibuat mulai dari perancangan alur kerja sistem dan perancangan basis data

sistem.

3. Pada tahap ini, dilakukan persiapan berbagai jenis instrument yang dibutuhkan dalam penelitian, berupa software maupun hardware.
4. Pada tahap ini, desain sistem yang telah dirancang kemudian di implementasikan dan dibangun menggunakan instrument yang sudah disiapkan berupa server, *access point*, *WLC (Wireless LAN Controller)*, Proxmox , Ubuntu 18.04
5. Pada tahap ini, setelah sistem telah dibangun maka dilakukan testing *WLC (Wireless LAN Controller)* ke LDAP untuk mengetahui apakah konfigurasi yang dilakukan berhasil atau tidak.
6. Pada tahap ini setelah sistem telah dibangun maka dilakukan pengambilan data berupa *latency autentikasi* berapa lama selesai *autentikasi* dan performansi *CPU* untuk melihat kerja *CPU* untuk 50 unit PC.
7. Setelah dilakukan pengambilan data, maka data yang diperoleh akan dianalisis untuk mengetahui kinerja dari system yang diterapkan.

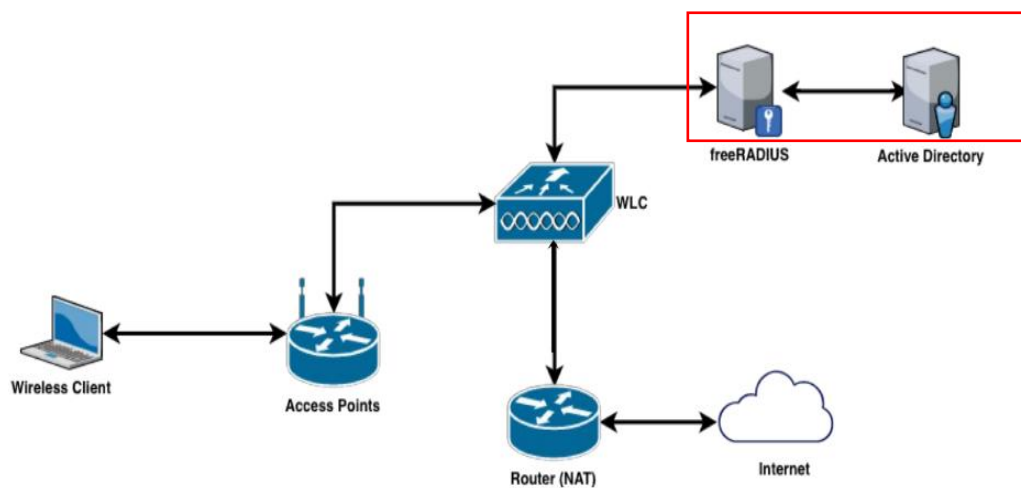
3.4 Perancangan Sistem



Gambar 3.2. Rancangan system yang saat ini digunakan

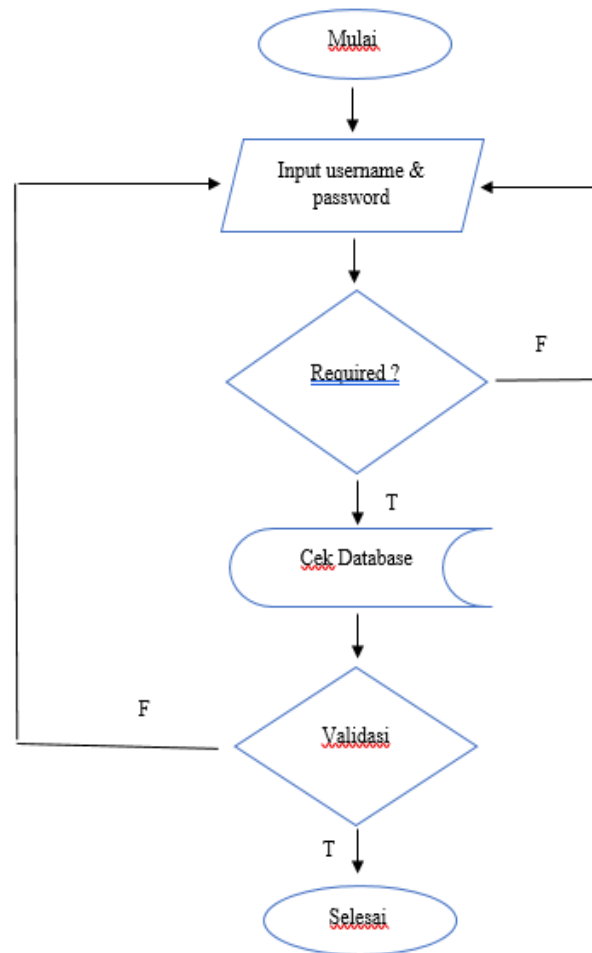
Gambar 3.2 adalah topologi jaringan internet yang ada di kampus UNHAS sekarang. Contoh pada gambar 3.2 terdapat dua fakultas yaitu fakultas A dan fakultas B. Bisa dilihat bahwa fakultas A dan fakultas B memiliki tersendiri routernya maka *login* pengguna hanya dapat dilakukan dimasing-masing fakultas. Pada router tiap fakultas disitu akan terjadi *authentikasi* terhadap pengguna *wifi* yang ingin *login*. Apabila pengguna dari fakultas A berkunjung ke fakultas B dan ingin menggunakan jaringan

wifinya maka pengguna fakultas A tersebut tidak bisa *login* dikarenakan pengguna dari fakultas A tidak terdaftar di fakultas B, begitupun sebaliknya. Pengguna yang dimaksud disini adalah dosen, pegawai, serta mahasiswa yang ada di fakultas tersebut. Sehingga pada penelitian ini diterapkan system *authentikasi* terpusat untuk mengatasi masalah diatas.



Gambar 3.3. Rancangan system yang akan diterapkan

Gambar 3.3 adalah rancangan system yang akan dibuat. Pada kotak yang berwarna merah adalah system yang akan *diinstalasi* pada penelitian ini. *Instalasinya* dapat dilihat pada poin 3.4.1 – 3.4.3. Setelah system *diinstalasi* maka selanjutnya pada *WLC* *dikonfigurasi* agar system yang *diinstalasi* tadi dapat digunakan dan siap dipancarkan oleh *access point* sehingga pengguna dapat menggunakannya. Konfigurasi tersebut dapat dilihat pada poin 3.4.4. Apabila system sudah rampung maka jumlah *access point* diperbanyak sesuai dengan kebutuhan yang akan digunakan.



Gambar 3.4 Alur proses *login*

Gambar 3.4 adalah penggambaran dari alur proses terhadap system yang dibuat. Mulai dari pengguna memasukan username dan password. Kemudian dicek apakah kolom username dan password sudah terisi atau tidak. Jika system mendekteksi bahwa salah satu dari kolom username dan password tidak terisi maka akan kembali pada menu inputan. Dan jika kolom inputan telah terisi system akan melanjutkan pada proses pengecekan kedatabase dimana dalam proses ini terjadi proses validasi atas username dan password yang dimasukkan, yang mana jika salah akan

kembali lagi pada menu inputan awal. Tetapi jika benar maka alur proses *login* berhasil.

Perancangan sistem dilakukan sesuai desain rancangan yang telah ditentukan. Dengan beberapa hal yang perlu disiapkan untuk menerapkan sistem, antara lain mempersiapkan perangkat keras yang akan digunakan dan juga instalasi perangkat lunak diperlukan.

Implementasi *LDAP* dan *FreeRadius* pada jaringan *Wifi* kampus UNHAS yang akan dimulai dengan pemasangan server pada pada rak server. Spesifikasi dari server tersebut adalah *CPU* Intel Xeon 8 core, RAM 32GB dan hardisk 2TB. Jika sudah dilakukan pemasangan maka dilanjutkan dengan membuat virtualisasi *LDAP* dan *FreeRadius* pada server. Kemudian install Ubuntu 18.04 pada virtualisasi *LDAP* dan *FreeRadius*.

3.4.1 Instalasi FreeRadius

FreeRadius digunakan untuk menangani *otentikasi* dan otorisasi koneksi yang dilakukan user. Pada saat komputer client akan menghubungkan diri dengan jaringan maka server *Radius* akan meminta identitas user (*username* dan *password*) untuk kemudian dicocokkan dengan data yang ada dalam database server untuk kemudian ditentukan apakah user diijinkan untuk menggunakan layanan dalam jaringan komputer. Jika proses *otentikasi* dan otorisasi berhasil maka proses pelaporan dilakukan, yakni dengan mencatat semua aktifitas koneksi user, menghitung durasi waktu dan jumlah transfer data dilakukan oleh user.

Install Ubuntu server 18.04 pada *virtualisasi FreeRadius* yang ada pada server, lalu *install FreeRadius* pada ubuntu server yang telah terinstall tadi dengan perintah :

a) Instalasi FreeRadius dengan perintah :

```
sudo apt-get install freeradius freeradius-ldap freeradius-  
config freeradius-mysql freeradius-common freeradius-utils  
nmap
```

Fungsi dari perintah diatas untuk menginstall freeradius yang akan digunakan sebagai otentikasi

b) Membuat akun dan sandi test dengan perintah :

```
nano /etc/freeradius/3.0/users
```

Fungsi dari perintah diatas untuk membuat akun yang akan dikoneksikan antara freeradius dengan ldap

c) Mengedit konfigurasi ldap dengan perintah :

```
nano /etc/freeradius/3.0/mods-available/ldap  
  
# isi server dengan alamat ip ke ldap  
  
server = 10.1.2.36  
  
identity = 'cn=admin,dc=unhas,dc=ac,dc=id'  
  
password = *****  
  
base_dn = 'dc=unhas,dc=ac,dc=id'
```

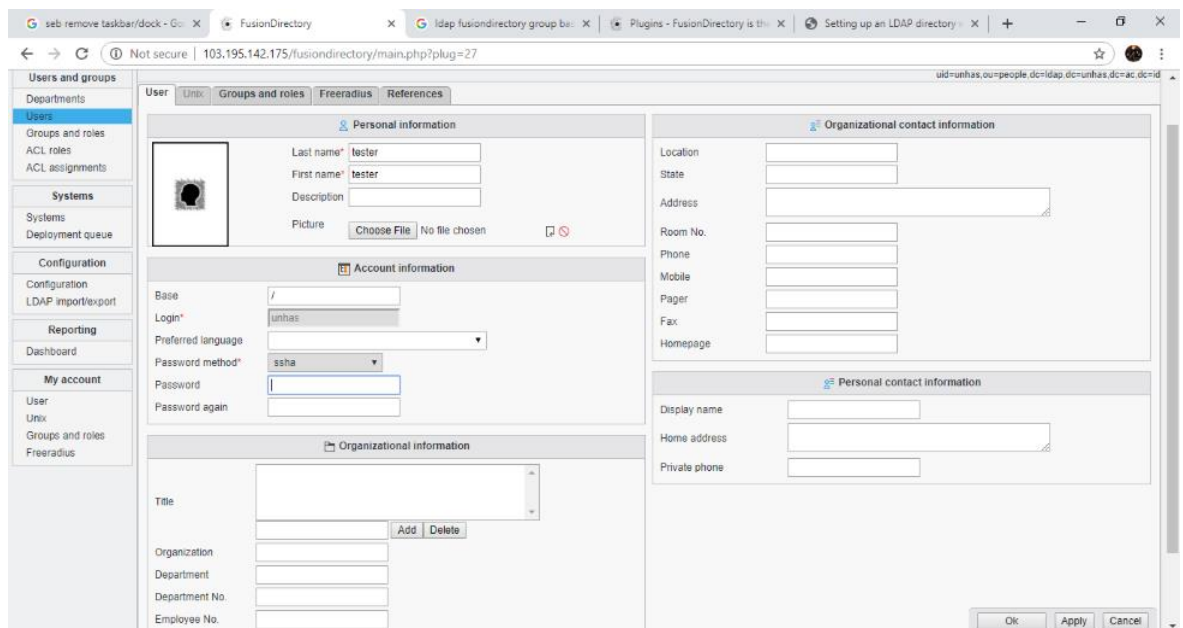
Fungsi perintah diatas untuk membuat identitas ldap

d) Mengcopy file ke mods-enabled dengan perintah :

```
ln -s /etc/freeradius/3.0/mods-available/ldap
/etc/freeradius/3.0/mods-enabled/ldap
```

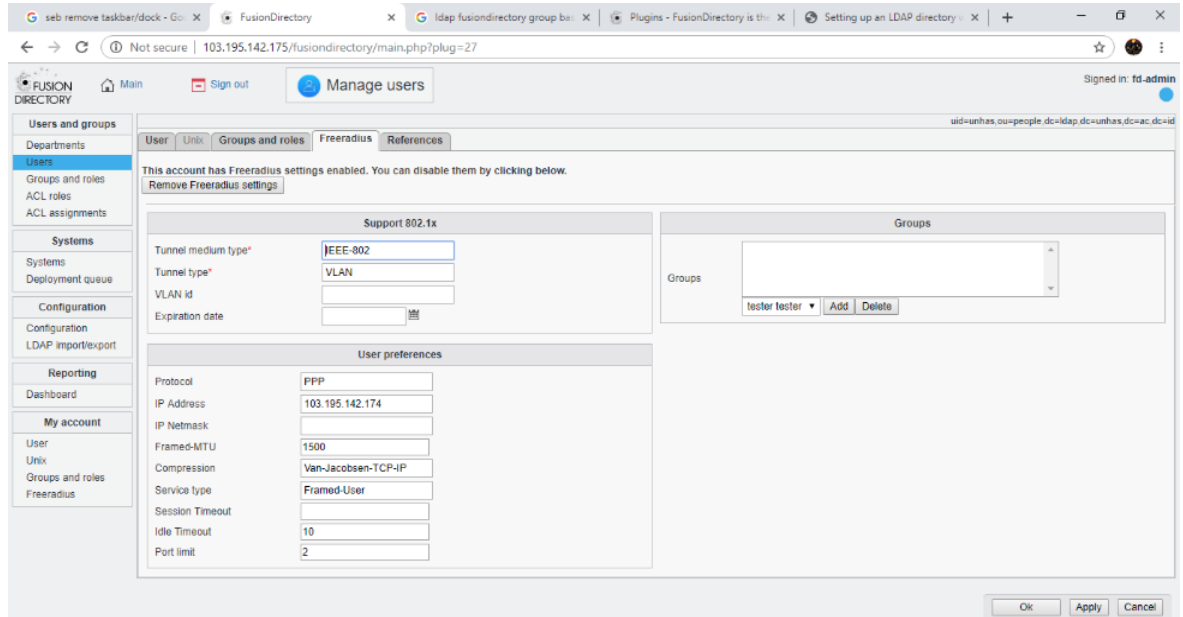
Fungsi diatas untuk menyalin konfigurasi

e) Membuat user baru di FusionDirectory



Gambar 3. 5. Membuat user baru di Fusion Directory

f) Masukkan settingan freeradius



Gambar 3. 6. Masukan settingan FreeRadius

g) Mengetest koneksi radius ke ldap.

- 1) Melihat radius bekerja tanpa masalah dengan perintah :

```
freeradius -X
```

- 2) Test koneksi radius ke ldap dengan perintah :

```
radtest 'user' 'pass' 'ip' '18120 testing123
```

h) Freeradius Terhubung dengan LDAP

```
root@srv-ldap:/# radtest unhas ***** 127.0.0.1 18120
testing123 Sent Access-Request Id 146 from 0.0.0.0:47194 to
127.0.0.1:1812 length 75

User-Name = "unhas"

User-Password = "*****"

NAS-IP-Address = 10.1.2.9
```

```
NAS-Port = 18120

Message-Authenticator = 0x00

Cleartext-Password = "*****"

Received Access-Accept Id 146 from 127.0.0.1:1812 to
0.0.0.0:0 length 20
```

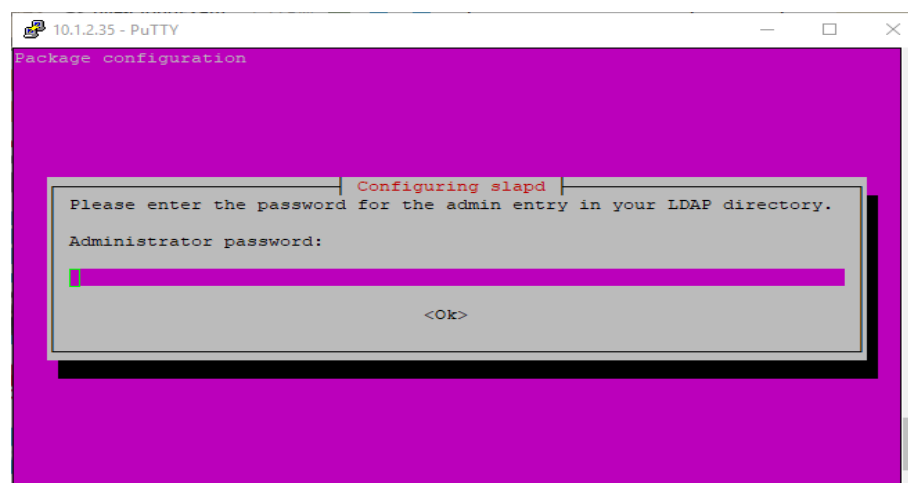
Fungsi diatas untuk *testing* dari *freeradius* ke *ldap* apakah sudah bisa terhubung atau belum.

3.4.2 Instalasi OpenLdap

OpenLdap digunakan oleh program email dan beberapa aplikasi lain untuk mencari dan mengambil informasi dari sebuah direktori yang disimpan pada sebuah server.

Install Ubuntu server 18.04 pada *virtualisasi LDAP* yang ada pada server, lalu *install openldap* pada ubuntu server yang telah terinstall tadi dengan perintah :

- a) `sudo apt-get install -y slapd ldap-utils libarchive-zip-perl`



Gambar 3.7. Instalasi openldap

- b) Memastikan port Ldap ada dan berjalan dengan perintah :
- nmap localhost

```

10.1.2.35 - PuTTY
e /usr/lib/x86_64-linux-gnu/libblas.so.3 (libblas.so.3-x86_64-linux-gnu) in auto
mode
Processing triggers for libc-bin (2.27-3ubuntu1) ...
Processing triggers for man-db (2.8.3-2) ...
Setting up liblinear3:amd64 (2.1.0+dfsg-2) ...
Setting up liblua5.3-0:amd64 (5.3.3-1ubuntu0.18.04.1) ...
Setting up nmap (7.60-1ubuntu5) ...
Processing triggers for libc-bin (2.27-3ubuntu1) ...
$ nmap localhost
-sh: 5: nmap: not found
$ nmap localhost

Starting Nmap 7.60 ( https://nmap.org ) at 2019-04-30 06:27 UTC
Nmap scan report for localhost (127.0.0.1)
Host is up (0.00023s latency).
Other addresses for localhost (not scanned): ::1
rDNS record for 127.0.0.1: localhost.localdomain
Not shown: 998 closed ports
PORT      STATE SERVICE
22/tcp    open  ssh
389/tcp   open  ldap
Nmap done: 1 IP address (1 host up) scanned in 0.13 seconds
$

```

Gambar 3. 8. Memastikan port ldap berjalan

- c) Membuat base_domain Top level domain =unhas.ac.id dengan perintah:

```

my_base_dn=$(echo "unhas.co.id" | sed -e 's/\./,dc=/g;s/^/dc=/' )
$ echo "$my_base_dn" dc=unhas,dc=co,dc=id

```

- d) Memasukkan file settingnan

- 1)

```

root@srv-ldap: sed -ie "s/olcSuffix:*/olcSuffix:
$my_base_dn/"
/etc/ldap/slapd.d/cn\=config/olcDatabase\={1\}mdb.ldif

```
- 2)

```

root@srv-ldap: sed -ie "s/olcRootDN:*/olcRootDN:
cn=admin,$my_base_dn/"
/etc/ldap/slapd.d/cn\=config/olcDatabase\={1\}mdb.ldif

```

```
3) root@srv-ldap: tail -n +3 /etc/ldap/slapd.d/cn=config/olcDatabase={1}mdb.ldif > /tmp/db.ldif
```

```
4) root@srv-ldap: cat /tmp/db.ldif >> /etc/ldap/slapd.d/cn=config/olcDatabase={1}mdb.ldif
```

- e) Install Fusion Directory dan plugin-pluginnya seperti : php-mbstring, fusiondirectory, fusiondirectory-plugin-freeradius, fusiondirectory-plugin, freeradius-schema, fusiondirectory-plugin-ldapmanager, fusiondirectory-plugin-systems, fusiondirectory-plugin-systems-schema dengan perintah sebagai berikut :

```
apt-get install -y php-mbstring fusiondirectory fusiondirectory-plugin-freeradius fusiondirectory-plugin-freeradius-schema fusiondirectory-plugin-ldapmanager fusiondirectory-plugin-systems fusiondirectory-plugin-systems-schema
```

- f) Menginstal skema direktori fusion dengan perintah :

```
sudo fusiondirectory-insert-schema
```

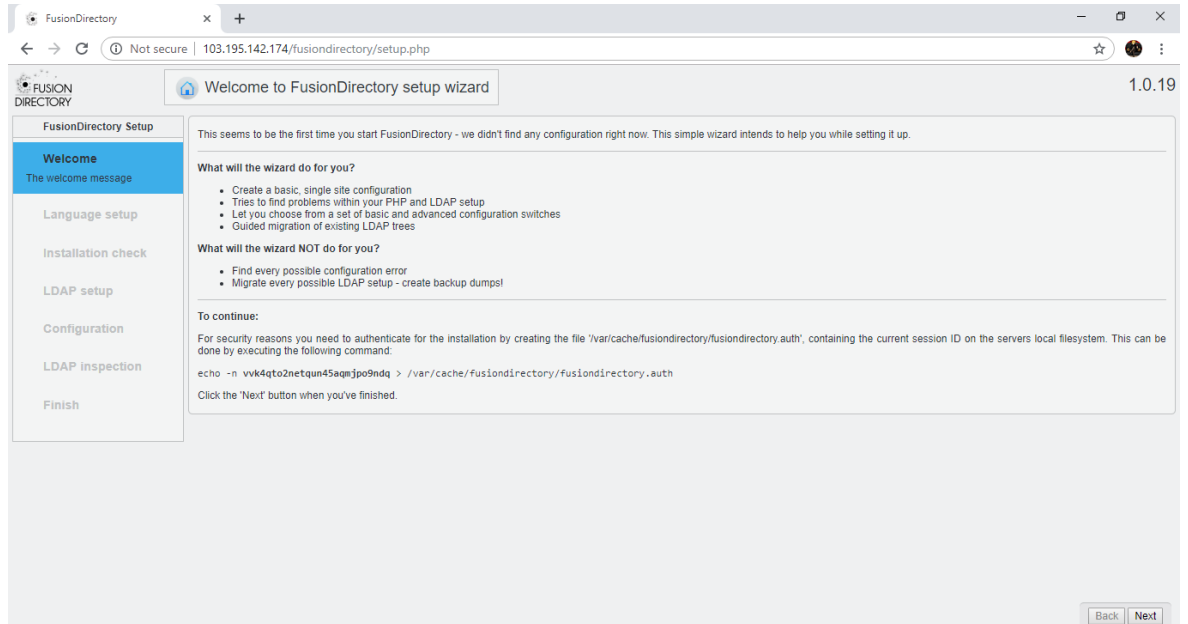
- g) Mengubah skema untuk menentukan jenis objek tambahan dalam LDAP dan konfigurasi FreeRadius dengan perintah :

```
sudo fusiondirectory-insert-schema -i -c -y /etc/ldap/schema/fusiondirectory/*
```

3.4.3 Setup Fusion Directory

a) Di browser masukkan, `http://<your VMs IP address>/fusiondirectory`.

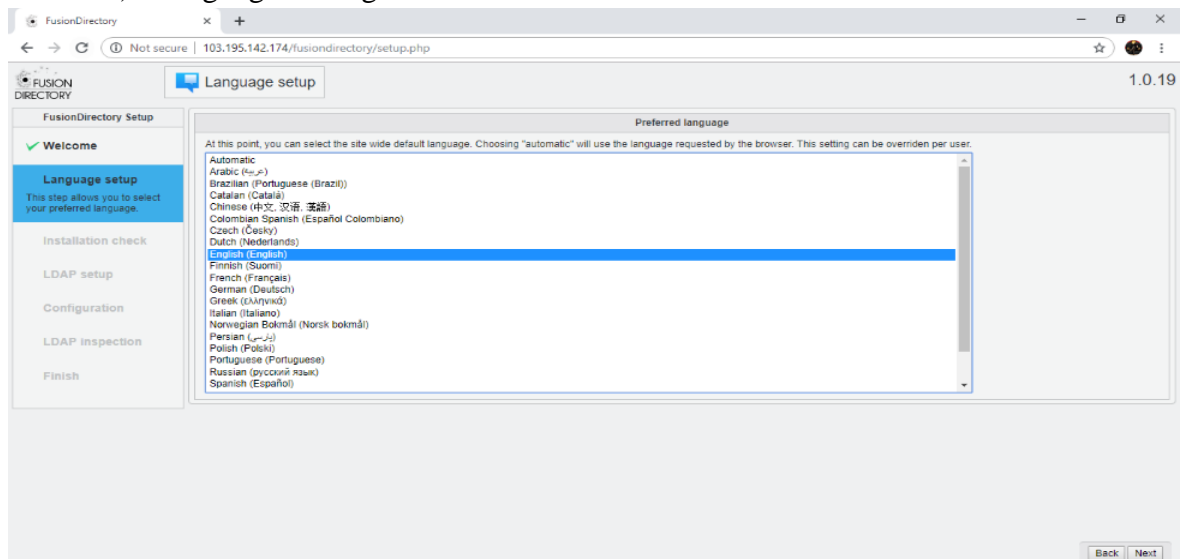
Disini IP dari VMs yaitu `http://10.1.2.36/fusiondirectory/`



Gambar 3.9. Tampilan *fusion directory*

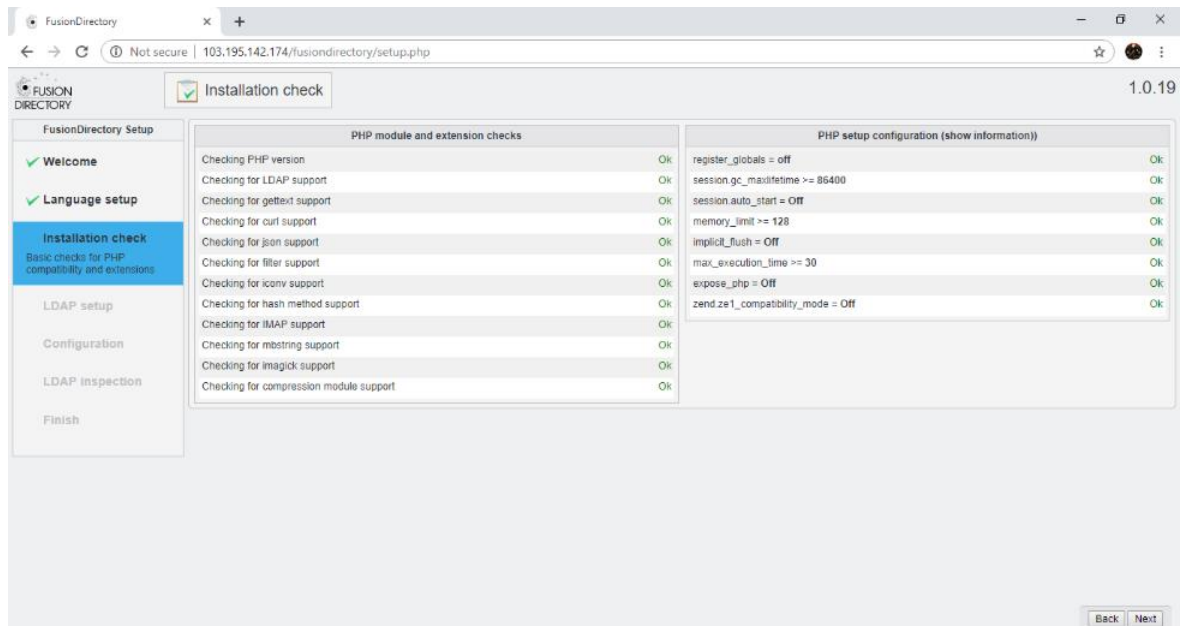
Kemudian jalankan echo command dari header To Continue dan klik next

b) Language Setting



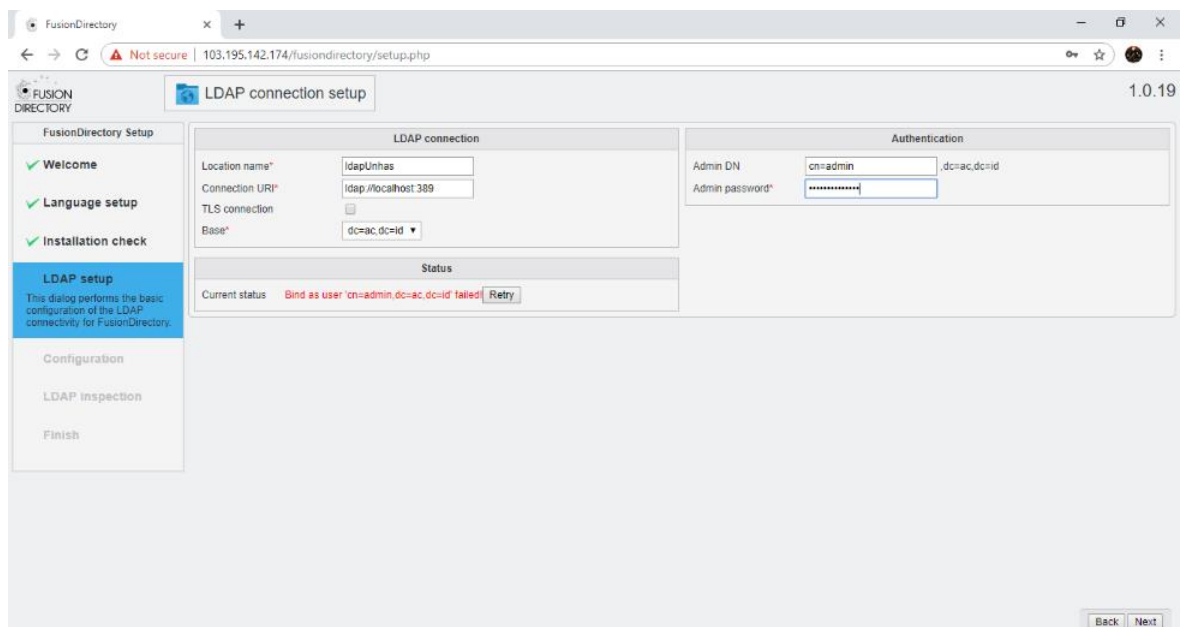
Gambar 3.10. Pengaturan bahasa

c) Installation Check



Gambar 3.11. Installation check

d) Ldap Setup



Gambar 3.12. Setting ldap

*Ldap Connection

Location name : ldapUnhas

Connection URI : ldap://localhost:389

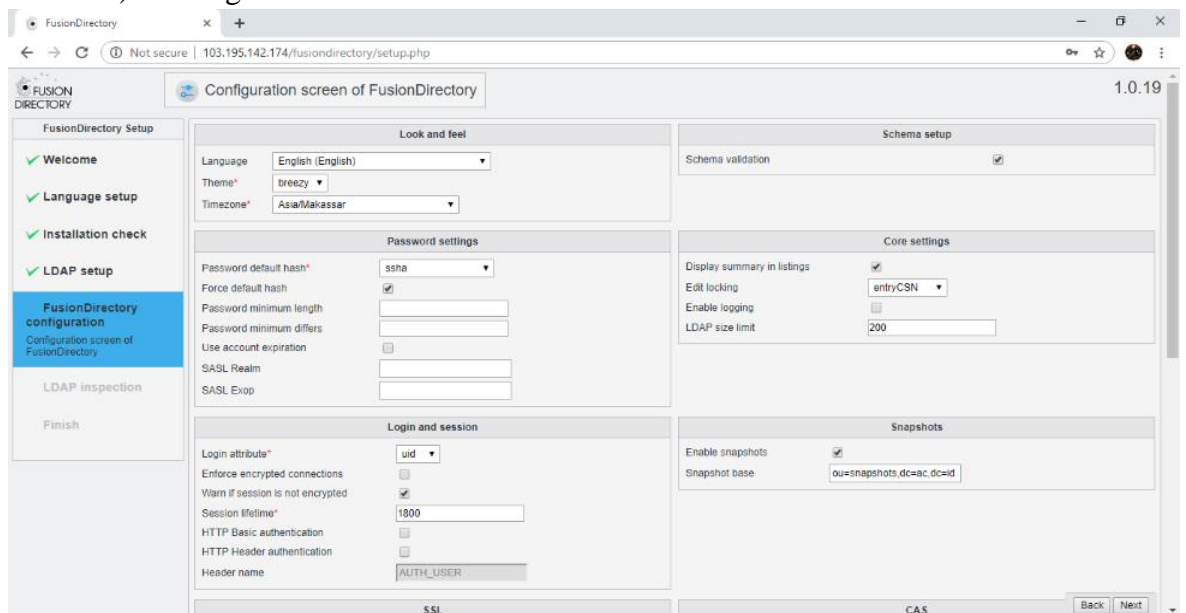
Base : dc=unhas,dc=ac,dc=id

*Authentication

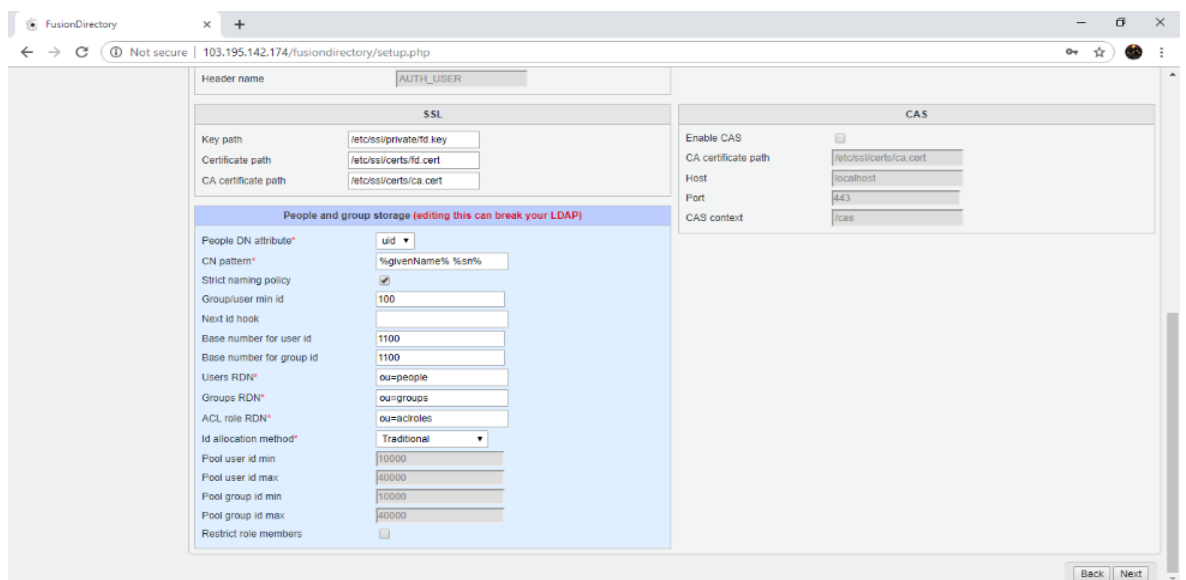
Admin DN : cn=admin

Admin Password : *****

e) Configuration



Gambar 3. 13. Configuration screen of FusionDirectory



Gambar 3. 14. Configuration screen of FusionDirectory

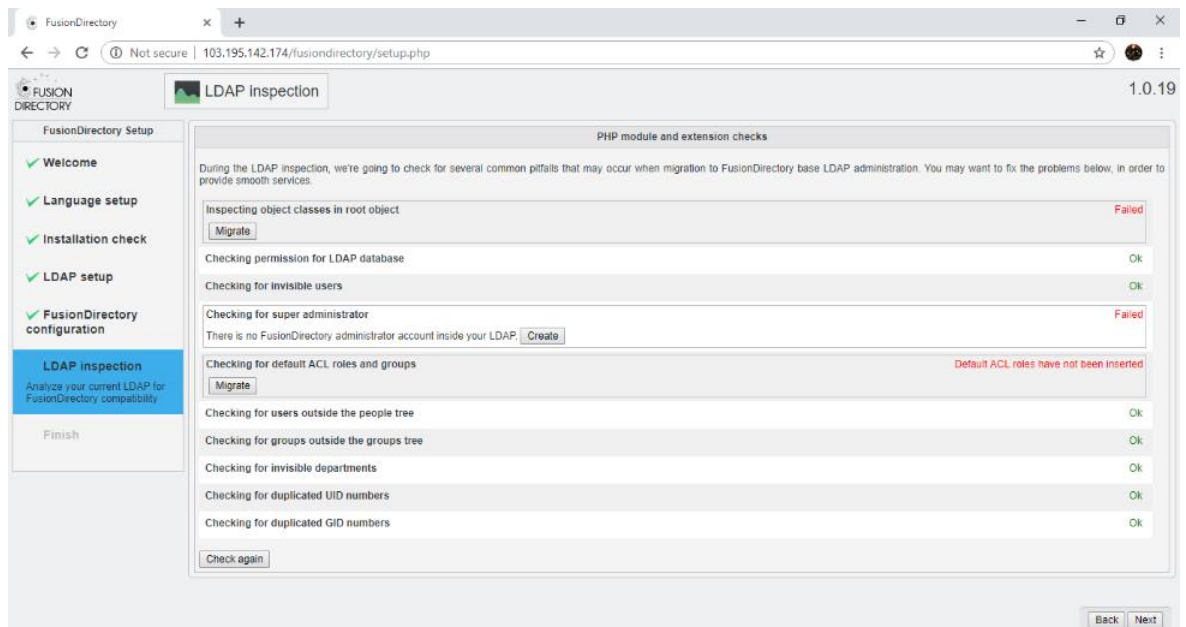
Timezone : Asia/Makassar

Password Default hash : ssha

Force default hash : centang ya

f) Ldap Inspection

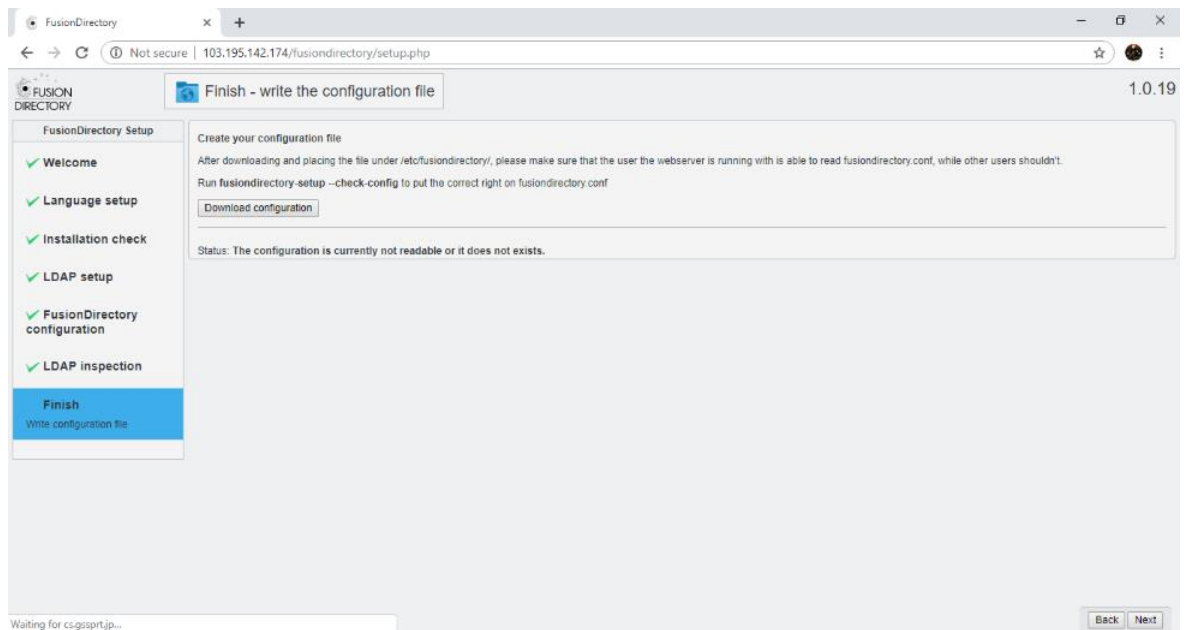
"Inspecting object classis in root object" pilih "Migrate"



Gambar 3. 15. Ldap Inpection

- Dihalaman selanjutnya pilih "Migrate" lagi
- "Checking for super administrator" pilih "Create"
- Akun super user untuk fusiondirectory
User ID : fd-admin
Password : *****
- Kemudian "Apply"
- "Checking for default ACL roles and groups" pilih 'Migrate'
- Next

g) Finish



Gambar 3. 16. Finish setup FusionDirectory

Pilih "Download configuration"

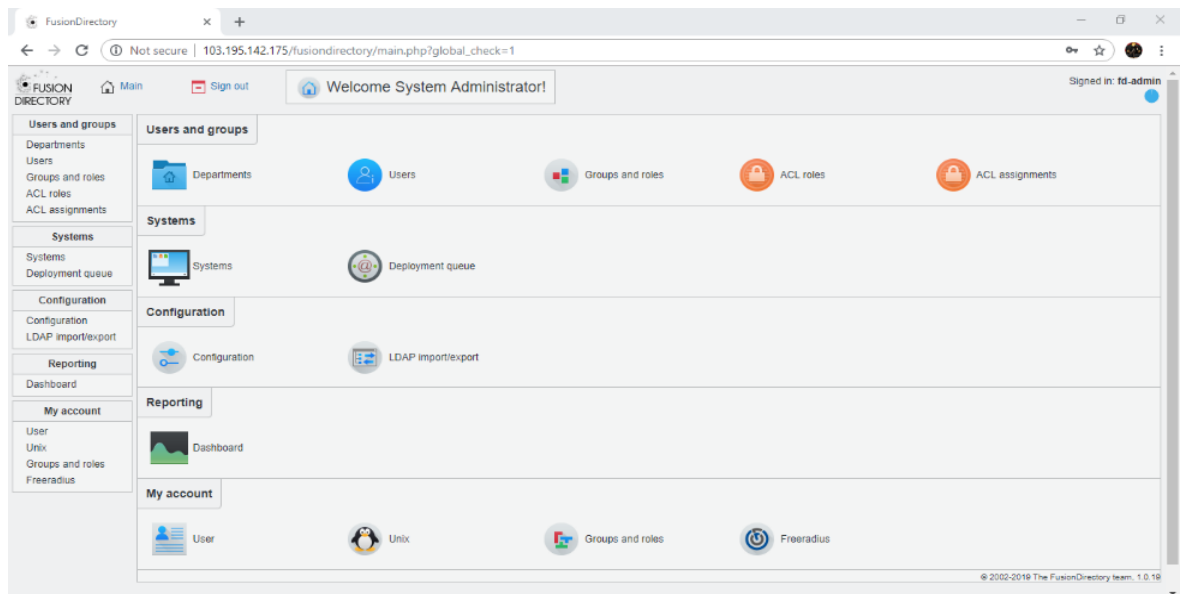
scp file /etc/fusiondirectory di VM(you may need to scp it to your home directory, and then copy it over)

jalankan "fusiondirectory-setup --check-config"

h) Kemudian *Login* ke FusionDirectory

<VM ip>/fusiondirectory> = http://10.1.2.36/fusiondirectory/

Masukkan user dan password yang dibuat sebelumnya

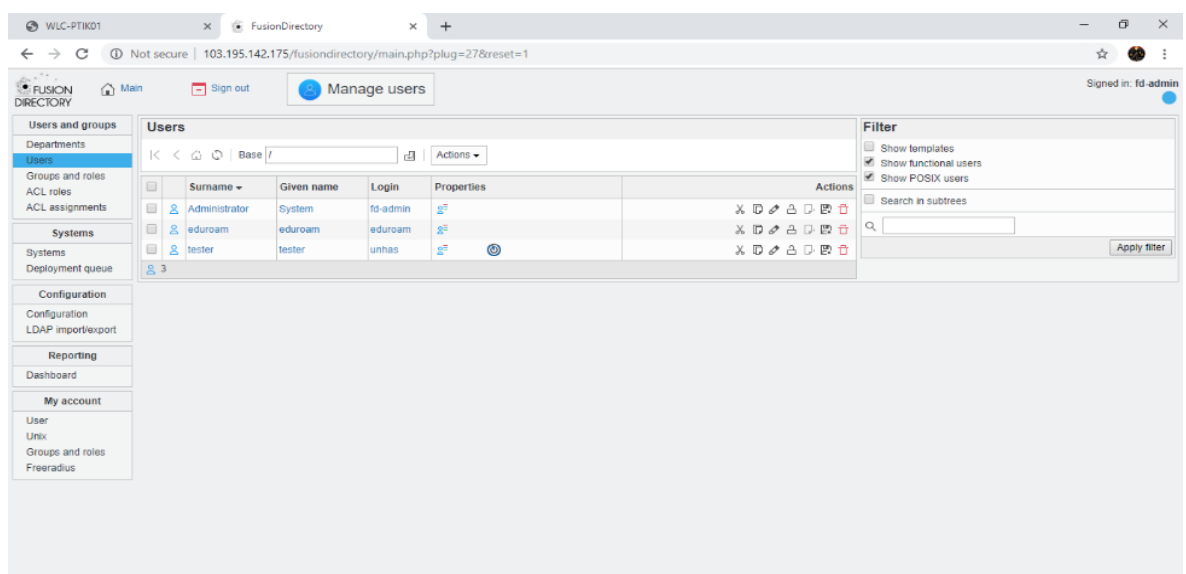


Gambar 3. 17. Tampilan *login* setelah setting FusionDirectory

3.4.4. Testing

Pada tahap ini kita melakukan pengujian terhadap LDAP dengan WLC untuk mengetahui system yang dirancang dapat terhubung dan siap digunakan. Adapun langkah-langkah sebagai berikut :

- a) Membuat User baru di fusiondirectory



Gambar 3. 18. Membuat *user*

User : handri

Pass : *****

b) Koneksi Radius Dengan WLC

```
nano /etc/freeradius/3.0/clients.conf
```

```
#koneksi ke radius
```

```
client 10.1.2.35 {
```

```
    ipaddr = 10.1.2.35
```

```
    secret =*****
```

```
}
```

```
#koneksi ke ldap
```

```
client 10.1.2.36 {
```

```
    ipaddr = 10.1.2.36
```

```
    secret = *****
```

```
}
```

```
#koneksi ke wlc
```

```
client 10.0.1.23 {
```

```
    ipaddr = 10.0.1.23
```

```
    secret = *****
```

```
}
```

Fungsi diatas untuk memasukkan alamat ip dan sandi pada setiap perangkat yang digunakan dalam penerapan system sehingga perangkat yang digunakan saling terhubung.

c) Memasukkan tipe otentikasi

```
vi /etc/freeradius/3.0/mods-enabled/eap

eap {

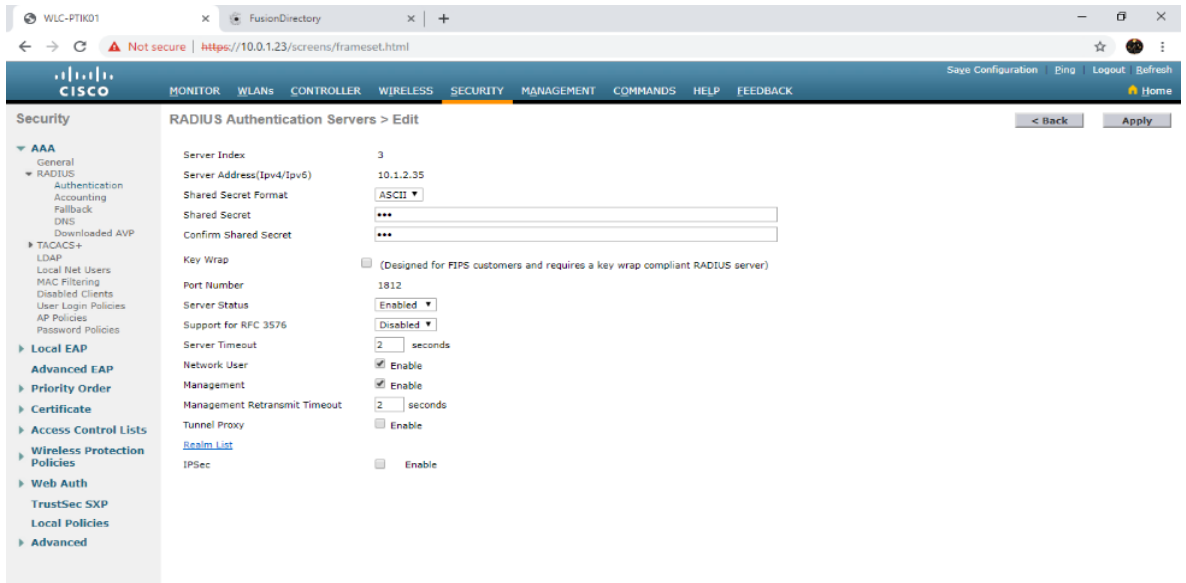
default_eap_type = eap_ttls

}
```

Fungsi diatas untuk memasukkan tipe otentikasi yang akan digunakan. Pada langkah ini kita akan mengatur keamanan otentikasi terhadap system yang telah dibuat. Bisa dilihat bahwa keamanan yang digunakan yaitu eap terlebih khusus eap_ttls. Proses otentikasi menggunakan eap-ttls terbagi dalam dua langkah utama, yaitu:

1. Pembentukan *secure tunnel* melalui sertifikat server.
2. Data yang dikirim akan di enkripsi oleh *secure tunnel* yang dibentuk di langkah 1. EAP-TTLS mendukung penyembunyian identitas karena data dilewatkan melalui *secure tunnel*. Sehingga data identitas *client* sangat aman. Data tersebut hanya diketahui oleh *client* dan *server*.

d) Manajemen WLC Security Radius

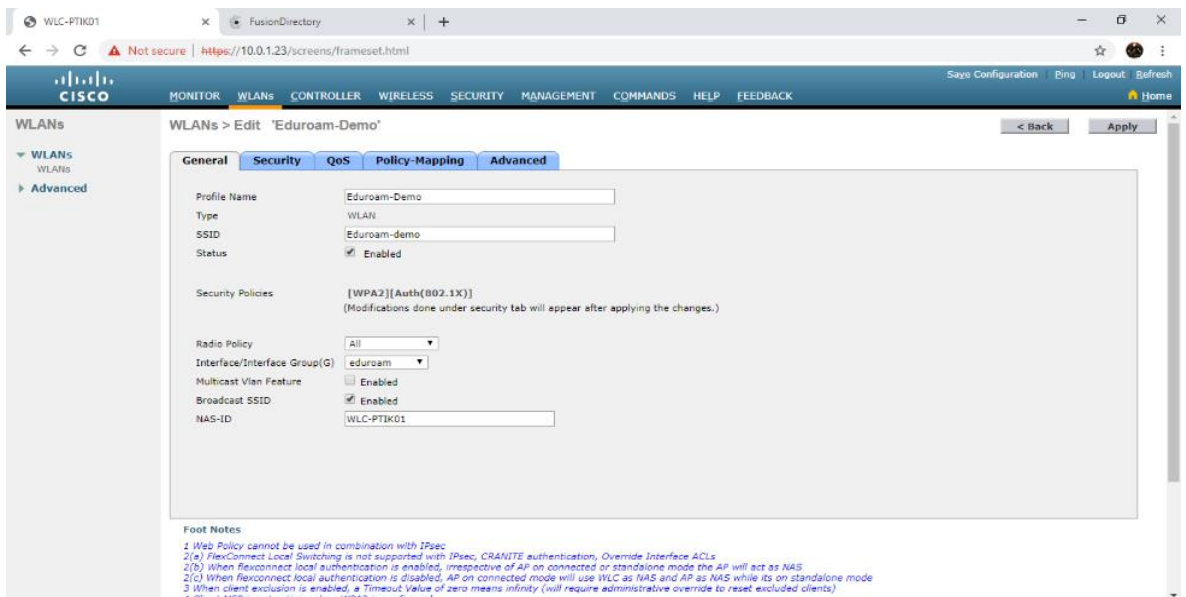


Gambar 3. 19. Manajemen WLC

Ip : 10.1.2.35 //ip radius srv

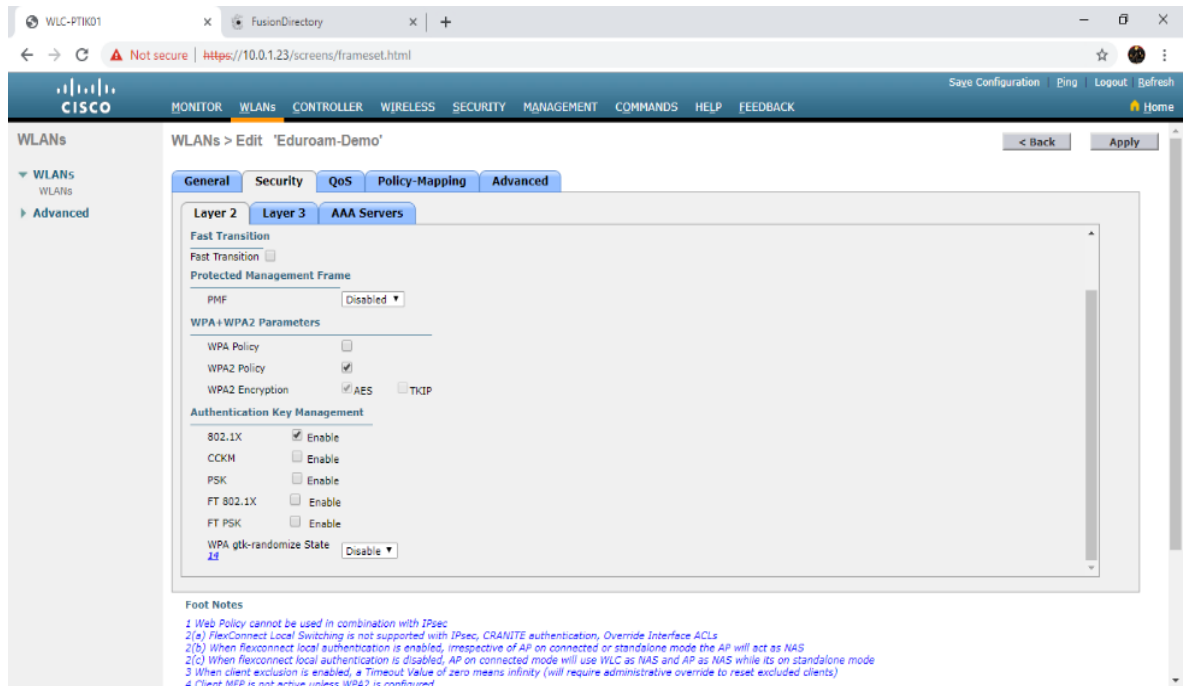
SharedSecret : ***** // /etc/freeradius/3.0/clients.conf

e) Membuat SSID Eduroam-demo



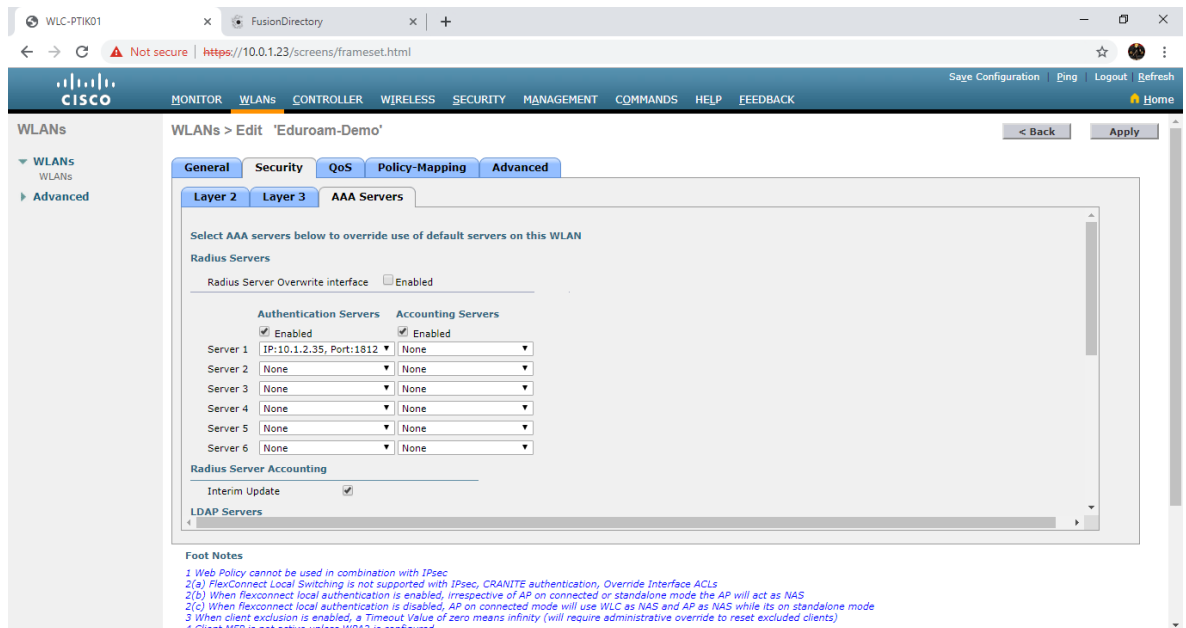
Gambar 3. 20. Membuat SSID

f) Mengatur Security WAP+WAP2, dan mengaktifkan 802.1x



Gambar 3. 21. Mengatur security

g) Memilih radius server yang telah dibuat.



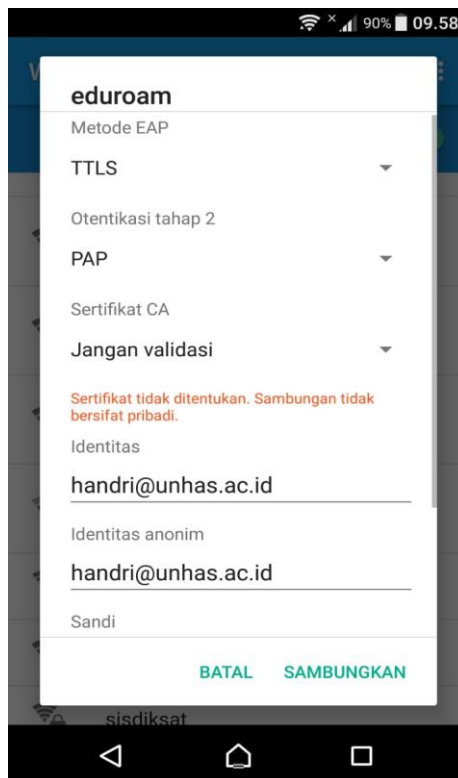
Gambar 3. 22. Memilih radius server yang telah dibuat

h) Menguji coba koneksi

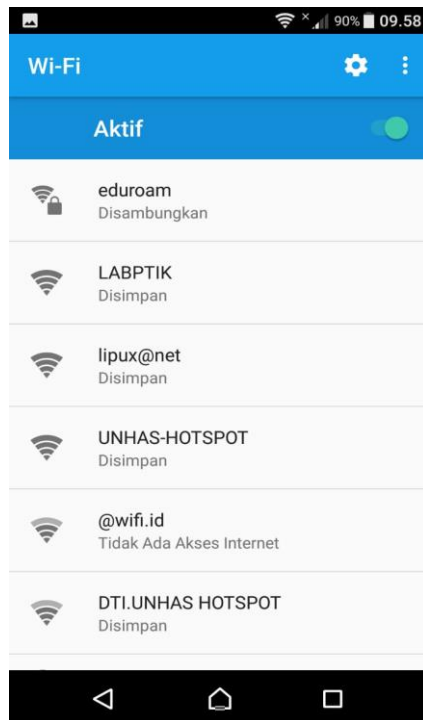
```
radtest handri@uhas.ac.id ***** 10.1.2.35 18120 testing123
```

Fungsi diatas untuk menguji koneksi dari *client* ke *server*

i) Koneksi Perangkat Mobile dengan LDAP



Gambar 3. 23. Menghubungkan *client* menggunakan perangkat *mobile*



Gambar 3. 24. Perangkat *mobile* sudah terhubung

3.5 Skenario Pengujian

Pengujian dilakukan dengan menggunakan parameter sebagai berikut:

3.5.1 Pengujian Latency *Authentikasi*

1. Menyiapkan 50 unit PC digunakan sebagai *client*
2. Menghubungkan 1 unit PC ke jaringan wifi yang telah dibuat untuk melihat berapa lama waktu otentikasinya.
3. Setelah data waktu otentikasinya didapat, catat data tersebut untuk dianalisis.
4. Ulangi langkah 2 dan 3 dengan jumlah user bertambah satu demi satu sampai dengan 50 unit PC yang harus dikoneksikan secara bersamaan.

5. Data yang diperoleh kemudian dianalisis untuk mengetahui waktu yang dibutuhkan setiap PC yang akan terhubung ke jaringan wifi dibuat.

3.5.2 Pengujian Penggunaan CPU

Cara pengujian ini sama dengan langkah pengujian latency otentikasi.

1. Menyiapkan 50 unit PC digunakan sebagai *client*
2. Menghubungkan 1 unit PC ke jaringan wifi yang telah dibuat untuk melihat berapa persen penggunaan CPU.
3. Setelah data persenan penggunaan CPU didapat, catat data tersebut untuk dianalisis.
4. Ulangi langkah 2 dan 3 dengan jumlah user bertambah satu demi satu sampai dengan 50 unit PC yang harus dikoneksikan secara bersamaan.
5. Data yang diperoleh kemudian dianalisis untuk mengetahui waktu yang dibutuhkan setiap PC yang akan terhubung ke jaringan wifi dibuat.

BAB IV

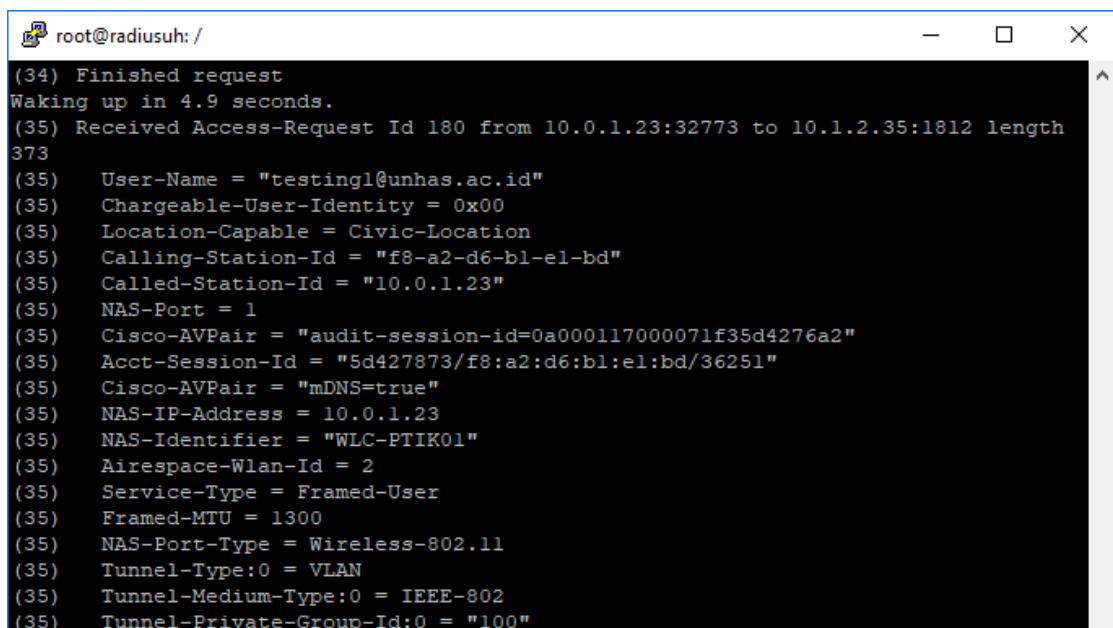
HASIL DAN PEMBAHASAN

4.1 Hasil Penelitian

Bab ini menjelaskan keseluruhan hasil pengujian dan analisa dari penerapan sistem yang telah dibuat, dengan demikian akan diketahui performansi dari *latemcy autentikasi* dan penggunaan *CPU*.

4.1.1 Hasil Pengujian Latency *Authentikasi*

Pada pengujian ini dilakukan menggunakan 50 unit PC yang akan dihubungkan ke jaringan yang telah dibuat. Dibutuhkan 1 unit PC sebagai *monitoring* terhadap 50 unit PC yang akan terhubung. Untuk melihat lantency *Authentikasi* yang diinginkan maka kita mengetikan perintah *freeradius -X* di *Command Line Interface (CLI)* pada putty.



```
root@radiusuh: /
(34) Finished request
Waking up in 4.9 seconds.
(35) Received Access-Request Id 180 from 10.0.1.23:32773 to 10.1.2.35:1812 length
373
(35)  User-Name = "testing1@unhas.ac.id"
(35)  Chargeable-User-Identity = 0x00
(35)  Location-Capable = Civic-Location
(35)  Calling-Station-Id = "f8-a2-d6-b1-el-bd"
(35)  Called-Station-Id = "10.0.1.23"
(35)  NAS-Port = 1
(35)  Cisco-AVPair = "audit-session-id=0a000117000071f35d4276a2"
(35)  Acct-Session-Id = "5d427873/f8:a2:d6:b1:el:bd/36251"
(35)  Cisco-AVPair = "mDNS=true"
(35)  NAS-IP-Address = 10.0.1.23
(35)  NAS-Identifier = "WLC-PTIK01"
(35)  Airespace-Wlan-Id = 2
(35)  Service-Type = Framed-User
(35)  Framed-MTU = 1300
(35)  NAS-Port-Type = Wireless-802.11
(35)  Tunnel-Type:0 = VLAN
(35)  Tunnel-Medium-Type:0 = IEEE-802
(35)  Tunnel-Private-Group-Id:0 = "100"
```

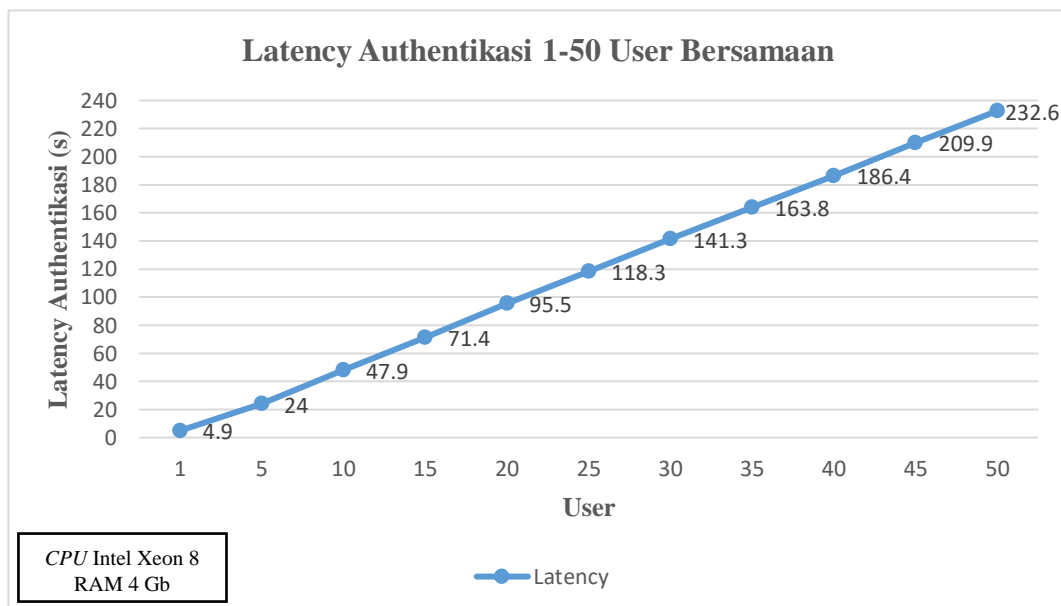
Gambar 4.1. User yang terhubung

Fungsi perintah *freeradius -X* adalah untuk menjalankan *authentikasi freeradius* dan sekaligus dapat melihat detail terhadap user yang terhubung, dapat dilihat pada gambar 4.1.

Tabel 4.1. Hasil pengujian latency pada 1-50 unit PC

User	Latency <i>Authentikasi</i> (s)
1	4,9
5	24
10	47,9
15	71,4
20	95,5
25	118,3
30	141,3
35	163,8
40	186,4
45	209,9
50	232,6

Hasil pengujian *latency authentikasi* dapat dilihat pada table 4.1 dan untuk lebih jelasnya dapat dilihat pada grafik gambar 4.1.



Gambar 4.2. Grafik *Latency Authentikasi* pada 1-50 unit PC

Berdasarkan gambar 4.2., grafik yang diperoleh dari masing-masing percobaan mulai dari 1 hingga 50 unit PC yang dikoneksikan secara bersamaan maka menunjukkan bahwa latency *otentikasi* yang dihasilkan semakin besar. Hal ini karena user yang dikoneksikan terus bertambah sehingga waktu yang dibutuhkan juga akan terus bertambah.

4.1.2 Pengujian Penggunaan CPU

Pada pengujian ini dilakukan menggunakan 50 unit PC yang akan dihubungkan ke jaringan yang telah dibuat. Dibutuhkan 1 unit PC sebagai *monitoring* terhadap 50 unit PC yang akan terhubung. Untuk memperoleh penggunaan *CPU* secara *realtime* dapat mengetikkan perintah “`top -b -n (...) -p (ID) | grep (command) > nama_file.txt`” di *Command Line Interface (CLI)* pada *putty*. Fungsi *top* yaitu fungsi yang dapat melihat semua fungsi yang sedang berjalan di system, dapat dilihat pada gambar 4.3

```

root@radiusuh: /
top - 13:46:47 up 1 day, 4:33, 2 users, load average: 0.07, 0.03, 0.00
Tasks: 111 total, 2 running, 69 sleeping, 0 stopped, 0 zombie
%Cpu(s): 0.0 us, 0.2 sy, 0.0 ni, 99.8 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st
KiB Mem : 4039432 total, 2826812 free, 140976 used, 1071644 buff/cache
KiB Swap: 4038652 total, 4038652 free, 0 used. 3643668 avail Mem

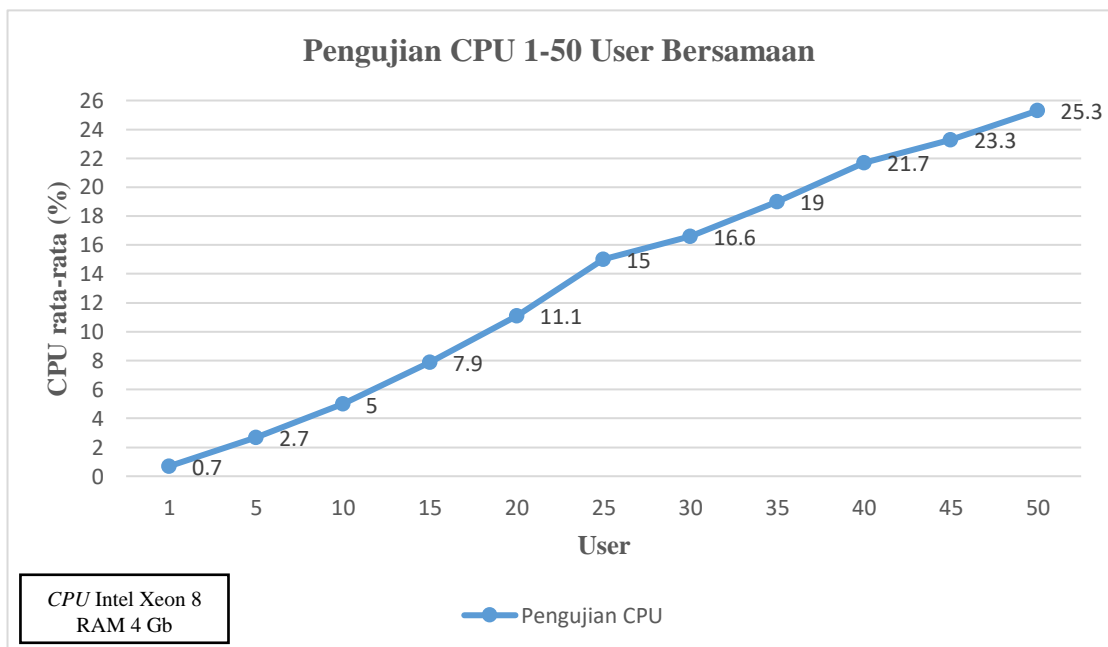
  PID USER      PR  NI   VIRT   RES   SHR  S  %CPU  %MEM    TIME+  COMMAND
 17544 www-data  20   0 526532 17832 10652 S   0.0   0.4   0:00.00 apache2
 17545 www-data  20   0 526532 17756 10588 S   0.0   0.4   0:00.00 apache2
 17546 www-data  20   0 526968 18072 10700 S   0.0   0.4   0:00.00 apache2
 17547 www-data  20   0 526532 17832 10652 S   0.0   0.4   0:00.00 apache2
 18174 www-data  20   0 526532 16772  9640 S   0.0   0.4   0:00.00 apache2
 23892 root       20   0 0        0      0 I   0.0   0.0   0:00.25 kworker/0:0
 24001 root       20   0 0        0      0 I   0.0   0.0   0:00.14 kworker/u4+
 24206 root       20   0 0        0      0 I   0.0   0.0   0:00.05 kworker/u4+
 24314 root       20   0 0        0      0 I   0.0   0.0   0:00.00 kworker/l:1
 24363 root       20   0 105684   6948   5960 S   0.0   0.2   0:00.03 sshd
 24365 handri   20   0 76628    7500   6464 S   0.0   0.2   0:00.04 systemd
 24366 handri   20   0 259388   2720     60 S   0.0   0.1   0:00.00 (sd-pam)
 24484 handri   20   0 107984   5680   4676 S   0.0   0.1   0:00.03 sshd
 24485 handri   20   0 4628     776    712 S   0.0   0.0   0:00.00 sh
 24494 root       20   0 66696   4420   3792 S   0.0   0.1   0:00.01 sudo
 24496 root       20   0 61756   3664   3216 S   0.0   0.1   0:00.00 su
 24497 root       20   0 20312   3884   3332 S   0.0   0.1   0:00.01 bash
  
```

Gambar 4.3. Tampilan perintah *top*

Tabel 4.2. Hasil pengujian performansi *CPU* pada 1-50 unit PC

User	Performasi <i>CPU</i> (%)
1	0,7
5	2,7
10	5
15	7,9
20	11,1
25	15
30	16,6
35	19
40	21,7
45	23,3
50	25,3

Hasil pengujian *performansi CPU* dapat dilihat pada table 4.2 dan untuk lebih jelasnya dapat dilihat pada grafik gambar 4.5.



Gambar 4.5. Grafik rata-rata *CPU* pada 1-50 unit PC

Berdasarkan gambar 4.5, grafik yang diperoleh dari percobaan 1-50 PC yang dikoneksikan secara bersamaan maka menunjukkan bahwa

performansi *CPU* memiliki presentase yang terus naik. Hal ini disebabkan karena permintaan dari user yang ingin terhubung terus bertambah sehingga kinerja *CPU* juga akan bertambah. Sehingga menunjukkan bahwa semakin banyak user yang ingin terhubung maka semakin besar juga jumlah performansi *CPU* yang dihasilkan.

4.2 Pembahasan

4.2.1 Lantency *Authentikasi*

Berdasarkan tabel 4.1, besar waktu yang dibutuhkan dari 1 – 50 unit PC agar terhubung ke jaringan adalah 4,8 detik – 232,6 detik. *Latency autentikasi* dipengaruhi oleh berapa banyak pengguna yang akan terhubung secara bersamaan. Dengan banyaknya pengguna yang akan terhubung secara bersamaan maka semakin besar waktu yang dibutuhkan. Pada uji coba 50 unit PC secara bersamaan system akan memvalidasi satu per satu pengguna dari setiap 50 unit PC sehingga membutuhkan waktu yang cukup lama untuk menyelesaikan semuanya. Total waktu yang dibutuhkan untuk menyelesaikan proses validasi dari 50 unit PC yaitu 232,6 detik. Sehingga rata-rata untuk menyelesaikan per unit PC dari 50 unit PC yaitu 4,6 detik.

4.2.2 Performansi *CPU*

Berdasarkan tabel 4.2, besar penggunaan *CPU* yang dibutuhkan saat 1 – 50 unit PC terhubung ke jaringan adalah 0,7% -25,3%. Penggunaan *CPU* yang meningkat dipengaruhi oleh berapa banyak pengguna yang akan terhubung secara bersamaan. Dengan banyaknya

pengguna yang akan terhubung secara bersamaan maka semakin meningkat *CPU* yang digunakan. Pada uji coba 50 unit PC yang dihubungkan secara bersamaan performansi *CPU* meningkat sampai dengan 25,3%. Jika dirata-ratakan per unit PC membutuhkan 0,5% penggunaan *CPU*.

BAB V

PENUTUP

5.1 Kesimpulan

Dari hasil analisis yang telah dilakukan dalam pengujian sistem *otentikasi* terpusat menggunakan LDAP maka dapat disimpulkan bahwa:

1. Penerapan system *otentikasi* terpusat menggunakan LDAP dapat mempermudah client dalam mengakses *wifi* dilingkup kampus UNHAS.
2. Semakin banyak pengguna yang terhubung secara bersamaan maka semakin banyak pula waktu yang diperlukan untuk *otentikasi* sehingga semakin lama juga pengguna menunggu.
3. Penggunaan LDAP sangat meningkatkan keamanan dimana paket yang dikirim terenkripsi sehingga sulit untuk mengetahuinya.

5.2 Saran

Sehubungan dengan selesainya proses pembuatan skripsi ini, penulis bermaksud menyampaikan beberapa saran kepada para pembaca yakni:

1. Sebelum penerapan system yang akan dibuat alangkah baiknya lebih dulu pahami system yang dibuat.
2. Perlu adanya pengembangan terhadap otoritas akses sehingga pengguna mempunyai hak akses lebih.

3. Untuk pengembangan berikutnya untuk proses *autentikasi* dengan berbasis freeradius ini jangan hanya diterapkan pada jaringan *wireless*, melainkan juga pada jaringan LAN.

DAFTAR PUSTAKA

- Ubaidillah. "Analisis Eduroam Indonesia"
- Yesi Novaria Kunang, 2008. "Autentikasi pengguna wireless LAN berbasis Radius server".
- Jori David Joseph, 2016. "Analisis Trafik Jaringan Wireless Fidelity (*Wifi*) menggunakan Network Protocol Analyzer pada *Authentikasi WEP*".
- Fadjrin, Akbar, Jahnsen Gultom. 2013. "Cloud Computing Server Menggunakan Proxmox Pada CV. Cipta Solusi Sejahtera. STMIK PalComTech Palembang."
- Suryono, 2012. "Pembuatan Prototype Virtual Server Menggunakan Proxmox Ve Untuk Optimalisasi Resource Hardware Di Noc FKIP UNS".
- Muhammad Asfiandi, 2014. "Konfigurasi Freeradius sebagai Radius Server Menggunakan Centos 6.5 di PT. Telekomunikasi Seluler Department IT Operation Regional Sumbagut".
- Aprilia Ayu Mahardani, 2017. "Implementasi Openvpn Menggunakan Ldap Sebagai Manajemen User Pada Sistem Operasi Ubuntu".
- Yusrizal, 2017. "Rancang Bangun Layanan Web (Web Service) Untuk Aplikasi Rekam Medis Praktik Pribadi".
- Azhari Harahap, 2011. "Perbandingan Kinerja Eap-Tls, Eap-Ttls Dan Eap-Peap Sebagai Protokol Autentikasi Pada Jaringan Nirkabel"
- Andre Rizal Sinaga dkk, 2018. "Implementasi Autentikasi Mode Multi-Auth Pada Jaringan Local Area Network Berbasis Kabel Menggunakan Protokol IEEE 802.1X Dan Radius Server"

LAMPIRAN

LAMPIRAN 1

- Hasil proses masukan settingan pada perintah,

```
cat /tmp/db.ldif >> /etc/ldap/slapd.d
```

```
/cn\=config/olcDatabase\={1}\mdb.ldif :
```

```
# AUTO-GENERATED FILE - DO NOT EDIT!! Use ldapmodify.
```

```
# CRC32 d2700c21
```

```
dn: olcDatabase={1}mdb
```

```
objectClass: olcDatabaseConfig
```

```
objectClass: olcMdbConfig
```

```
olcDatabase: {1}mdb
```

```
olcDbDirectory: /var/lib/ldap
```

```
olcSuffix: ,dc=unhas,dc=ac,dc=id
```

```
olcAccess: {0}to attrs=userPassword by self write by anonymous auth by
```

```
    * none
```

```
olcAccess: {1}to attrs=shadowLastChange by self write by * read
```

```
olcAccess: {2}to * by * read
```

```
olcLastMod: TRUE
```

```
olcRootDN: cn=admin,dc=unhas,dc=ac,dc=id
```

```
olcRootPW::
```

```
e1NTSEF9NUZCaHVkWGREMndHWWg1OXVCbTVxUnkyc0VW
```

```
NlhoOTE=
```

olcDbCheckpoint: 512 30
olcDbIndex: objectClass eq
olcDbIndex: cn,uid eq
olcDbIndex: uidNumber,gidNumber eq
olcDbIndex: member,memberUid eq
olcDbMaxSize: 1073741824
structuralObjectClass: olcMdbConfig
entryUUID: aba9ac0a-ff5c-1038-94e4-5be379cd044a
creatorsName: cn=config
createTimestamp: 20190430062645Z
entryCSN: 20190430062645.289435Z#000000#000#000000
modifiersName: cn=config
modifyTimestamp: 20190430062645Z
dn: olcDatabase={1}mdb
objectClass: olcDatabaseConfig
objectClass: olcMdbConfig
olcDatabase: {1}mdb
olcDbDirectory: /var/lib/ldap
olcSuffix: dc=unhas,dc=ac,dc=id
olcAccess: {0}to attrs=userPassword by self write by anonymous auth by
* none
olcAccess: {1}to attrs=shadowLastChange by self write by * read
olcAccess: {2}to * by * read

olcLastMod: TRUE

olcRootDN: cn=admin,dc=unhas,dc=ac,dc=id

olcRootPW::

e1NTSEF9NUZCaHVkWGREMndHWWg1OXVCbTVxUnkyc0VW

NlhoOTE=

olcDbCheckpoint: 512 30

olcDbIndex: objectClass eq

LAMPIRAN 2

- Hasil proses pada perintah, sudo fusiondirectory-insert-schema :

```
SASL/EXTERNAL authentication started
```

```
SASL username:
```

```
gidNumber=0+uidNumber=0,cn=peercred,cn=external,cn=auth
```

```
SASL SSF: 0
```

```
executing 'ldapadd -Y EXTERNAL -H ldapi:/// -f
```

```
/etc/ldap/schema/fusiondirectory/core-fd.ldif'
```

```
SASL/EXTERNAL authentication started
```

```
SASL username:
```

```
gidNumber=0+uidNumber=0,cn=peercred,cn=external,cn=auth
```

```
SASL SSF: 0
```

```
adding new entry "cn=core-fd,cn=schema,cn=config"
```

```
SASL/EXTERNAL authentication started
```

```
SASL username:
```

```
gidNumber=0+uidNumber=0,cn=peercred,cn=external,cn=auth
```


SASL SSF: 0

executing 'ldapadd -Y EXTERNAL -H ldapi:/// -f

/etc/ldap/schema/fusiondirectory/core-fd-conf.ldif'

SASL/EXTERNAL authentication started

SASL username:

gidNumber=0+uidNumber=0,cn=peercred,cn=external,cn=auth

SASL SSF: 0

adding new entry "cn=core-fd-conf,cn=schema,cn=config"

SASL/EXTERNAL authentication started

SASL username:

gidNumber=0+uidNumber=0,cn=peercred,cn=external,cn=auth

SASL SSF: 0

executing 'ldapadd -Y EXTERNAL -H ldapi:/// -f

/etc/ldap/schema/fusiondirectory/ldapns.ldif'

SASL/EXTERNAL authentication started

SASL username:

gidNumber=0+uidNumber=0,cn=peercred,cn=external,cn=auth

SASL SSF: 0

adding new entry "cn=ldapns,cn=schema,cn=config"

SASL/EXTERNAL authentication started

SASL username:

gidNumber=0+uidNumber=0,cn=peercred,cn=external,cn=auth

SASL SSF: 0

```
executing 'ldapadd -Y EXTERNAL -H ldapi:/// -f
/etc/ldap/schema/fusiondirectory/template-fd.ldif'
SASL/EXTERNAL authentication started
SASL username:
gidNumber=0+uidNumber=0,cn=peercred,cn=external,cn=auth
SASL SSF: 0
adding new entry "cn=template-fd,cn=schema,cn=config"
```

LAMPIRAN 3

- Hasil proses pada perintah, sudo fusiondirectory-insert-schema -i -c -y

```
/etc/ldap/schema/fusiondirectory/* :
SASL SSF: 0
executing 'ldapadd -Y EXTERNAL -H ldapi:/// -f
/etc/ldap/schema/fusiondirectory/systems-fd.ldif'
SASL/EXTERNAL authentication started
SASL username:
gidNumber=0+uidNumber=0,cn=peercred,cn=external,cn=auth
SASL SSF: 0
adding new entry "cn=systems-fd,cn=schema,cn=config"
SASL/EXTERNAL authentication started
SASL username:
gidNumber=0+uidNumber=0,cn=peercred,cn=external,cn=auth
SASL SSF: 0
template-fd already exists in the LDAP, skipping...
```

LAMPIRAN 4

- Hasil proses menguji coba koneksi pada perintah,

radtest handri@uhas.ac.id ***** 10.1.2.35 18120 testing123

(0) Received Access-Request Id 59 from 10.1.2.35:49891 to

10.1.2.35:1812 length 77

(0) User-Name = "handri@uhas.ac.id"

(0) User-Password = "*****"

(0) NAS-IP-Address = 10.1.23

(0) NAS-Port = 18120

(0) Sent Access-Accept Id 59 from 10.1.2.35:1812 to 10.1.2.35:49891

length 0

(0) Finished request

Waking up in 4.9 seconds.

(0) Cleaning up request packet ID 59 with timestamp +15

LAMPIRAN 5

- Hasil dari menguji koneksi Perangkat Mobile dengan LDAP

(8) Received Access-Request Id 29 from 10.0.1.23:32771 to

10.1.2.35:1812 length 313

(8) User-Name = "handri@unhas.ac.id"

- (8) Chargeable-User-Identity = 0x00
- (8) Location-Capable = Civic-Location
- (8) Calling-Station-Id = "0c-98-38-c2-f2-b3"
- (8) Called-Station-Id = "10.0.1.23"
- (8) NAS-Port = 1
- (8) Cisco-AVPair = "audit-session-id=0a000117000ad77b5ce765ea"
- (8) Acct-Session-Id = "5ce765ea/0c:98:38:c2:f2:b3/6286"
- (8) Cisco-AVPair = "mDNS=true"
- (8) NAS-IP-Address = 10.0.1.23
- (8) NAS-Identifier = "WLC-PTIK01"
- (10) Finished request

Waking up in 4.9 seconds.

- (1) Cleaning up request packet ID 22 with timestamp +180
- (2) Cleaning up request packet ID 23 with timestamp +180
- (3) Cleaning up request packet ID 24 with timestamp +180
- (4) Cleaning up request packet ID 25 with timestamp +180
- (5) Cleaning up request packet ID 26 with timestamp +180
- (6) Cleaning up request packet ID 27 with timestamp +180

(7) Cleaning up request packet ID 28 with timestamp +180

(8) Cleaning up request packet ID 29 with timestamp +180

(9) Cleaning up request packet ID 30 with timestamp +180

(10) Cleaning up request packet ID 31 with timestamp +180

LAMPIRAN 6

Tabel 4.1. Hasil pengujian latency pada 1-50 unit PC

User	Latency <i>Authentikasi</i> (s)
1	4,8
2	9,7
3	14,5
4	19,2
5	24
6	28,8
7	33,5
8	38,3
9	43,1
10	47,9
11	52,8
12	57,1
13	62
14	66,6
15	71,4
16	76,2
17	81,1
18	85,7

19	90,6
20	95,5
21	100,3
22	104,6
23	109,2
24	113,6
25	118,3
26	123
27	127,4
28	132
29	136,5
30	141,3
31	145,6
32	150,3
33	154,7
34	159,3
35	163,8
36	168,1
37	172,8
38	177,7
39	182,1
40	186,4
41	191
42	195,9
43	200,6
44	205,1
45	209,9
46	214,3
47	219
48	223,6

49	228,2
50	232,6

LAMPIRAN 7

Tabel 4.2. Hasil pengujian *CPU* 1-50 unit PC

User	Kinerja <i>CPU</i> (%)
1	0,7
2	1,4
3	1,7
4	2
5	2,7
6	3,4
7	3,7
8	4,3
9	4,7
10	5
11	5,7
12	6,4
13	6,7
14	7
15	7,9
16	8,3
17	9,2
18	10
19	10,6
20	11,1
21	12,4
22	14,3
23	14,4

24	14,9
25	15
26	15,3
27	15,7
28	16
29	16,3
30	16,6
31	17
32	17,7
33	18,7
34	18,6
35	19
36	19,5
37	20
38	20,6
39	21,1
40	21,7
41	22
42	22,3
43	22,5
44	23
45	23,3
46	23,9
47	24,4
48	24,7
49	24,9
50	25,3