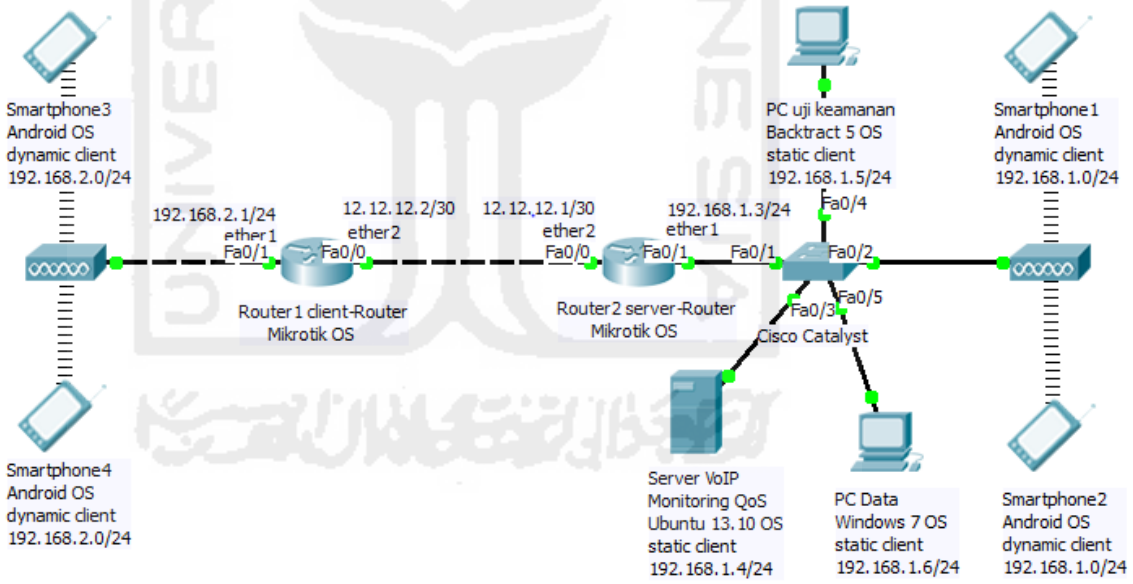
BAB III
METODOLOGI PENELITIAN

3.1. Gambaran Umum Sistem

Dalam pengerjaan tugas akhir ini, penulis menggunakan lima kondisi sistem, dari yang menggunakan *routing table* biasa, sampai yang menggunakan metode keamanan MPLS-VPN dan segmentasi VLAN, yang disertai dengan *firewall*. Berikut lima gambaran umum sistem, yang akan digunakan dalam pengerjaan tugas akhir ini.

1) Topologi VoIP pada jaringan *routing table* biasa.

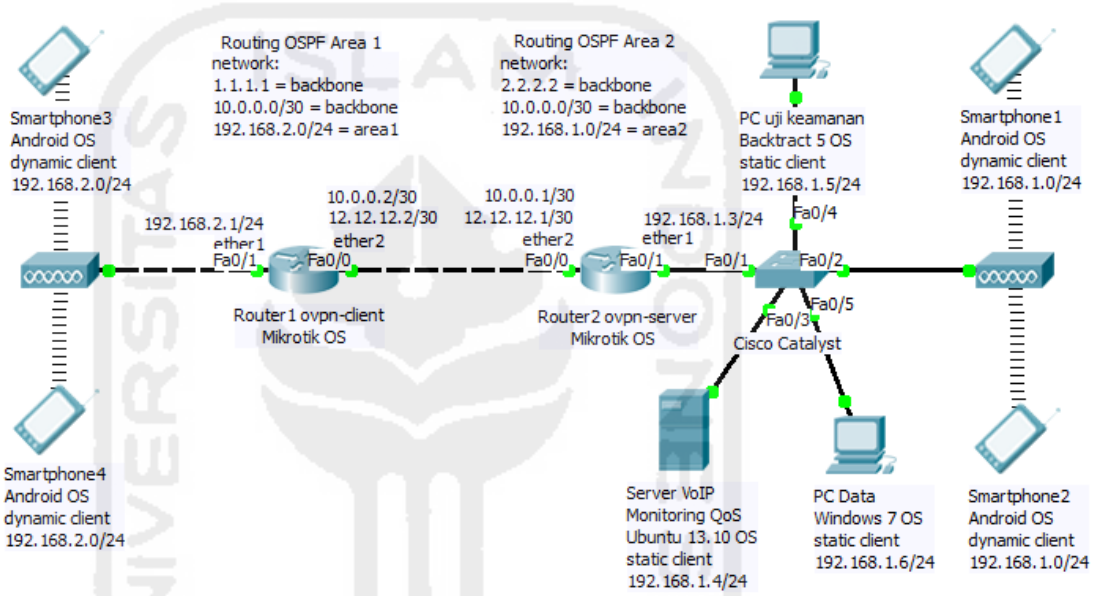
Topologi pada Gambar 3.1, menjelaskan bahwa sistem hanya menggunakan metode *routing table* biasa, sehingga belum ada metode pengamanan apapun di dalamnya.



Gambar 3.1 Topologi VoIP Menggunakan *Routing Table* Biasa

2) Topologi VoIP dengan menggunakan metode pengamanan MPLS-VPN.

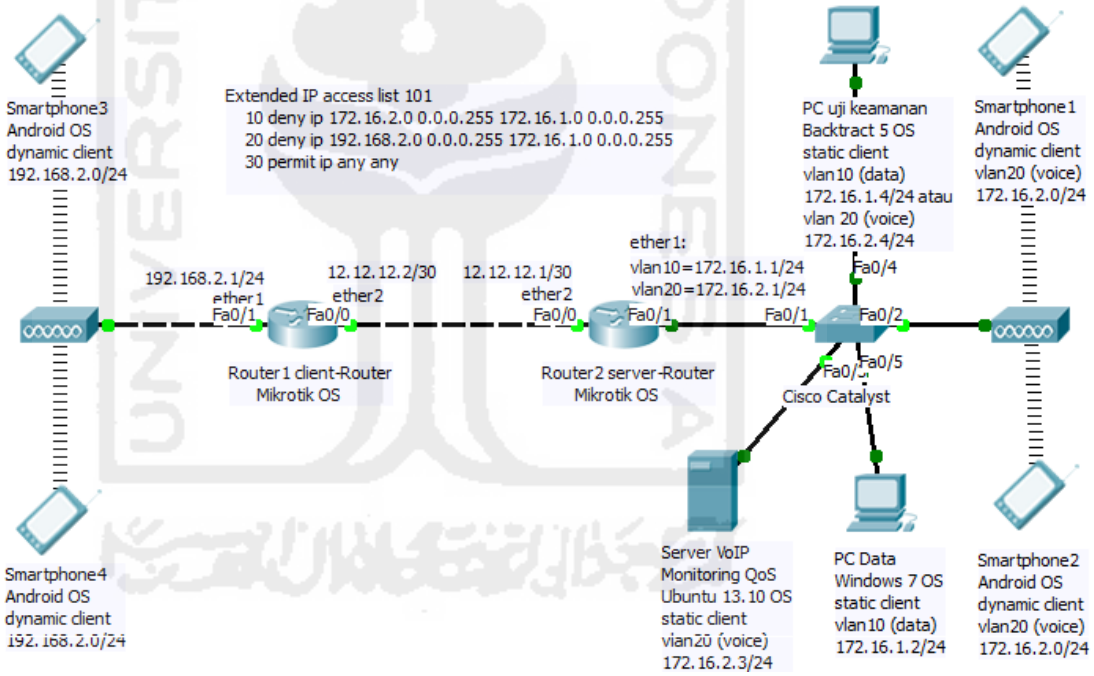
Topologi pada Gambar 3.2, menjelaskan bahwa sistem telah menggunakan metode pengamanan MPLS-VPN, dengan menambahkan pengaturan metode *routing* OSPF (*Open Shortest Path First*) pada jaringan VoIP, agar proses *routing* menjadi lebih cepat.



Gambar 3.2 Topologi VoIP Menggunakan Metode MPLS-VPN

3) Topologi VoIP dengan menggunakan metode segmentasi VLAN (disertai *firewall*).

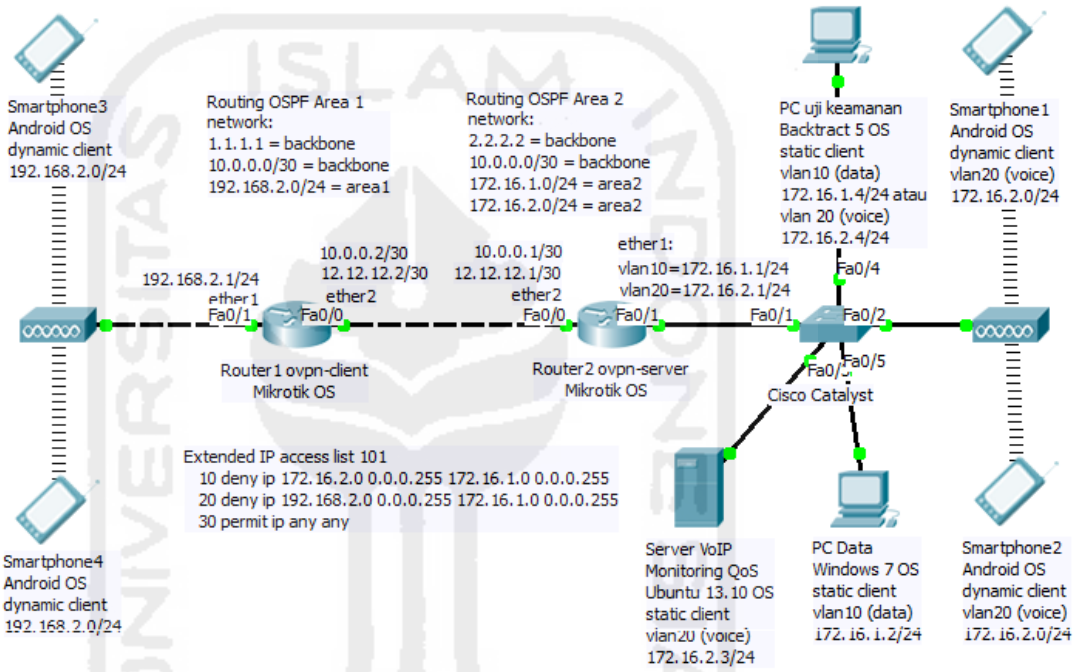
Topologi pada Gambar 3.3, menjelaskan bahwa sistem telah menerapkan metode keamanan segmentasi VLAN dan *firewall*, untuk memisahkan lalu lintas data dan suara pada jaringan VoIP. Penerapan aturan VLAN ini, ditandai dengan dipisahkannya jalur akses data dengan vlan 10 dan jalur akses suara dengan vlan 20. Kemudian untuk pengaturan *firewall*, ditandai dengan ditambahkannya perintah *extended* IP di dalam *Switch Cisco*, untuk mengatur aliran data agar kedua lalu lintas tidak dapat saling berkomunikasi, bahkan dengan menggunakan protokol ICMP.



Gambar 3.3 Topologi VoIP Menggunakan Metode Segmentasi VLAN (disertai *Firewall*)

4) Topologi VoIP pada jaringan MPLS-VPN, dengan menggunakan metode segmentasi VLAN (disertai *firewall*).

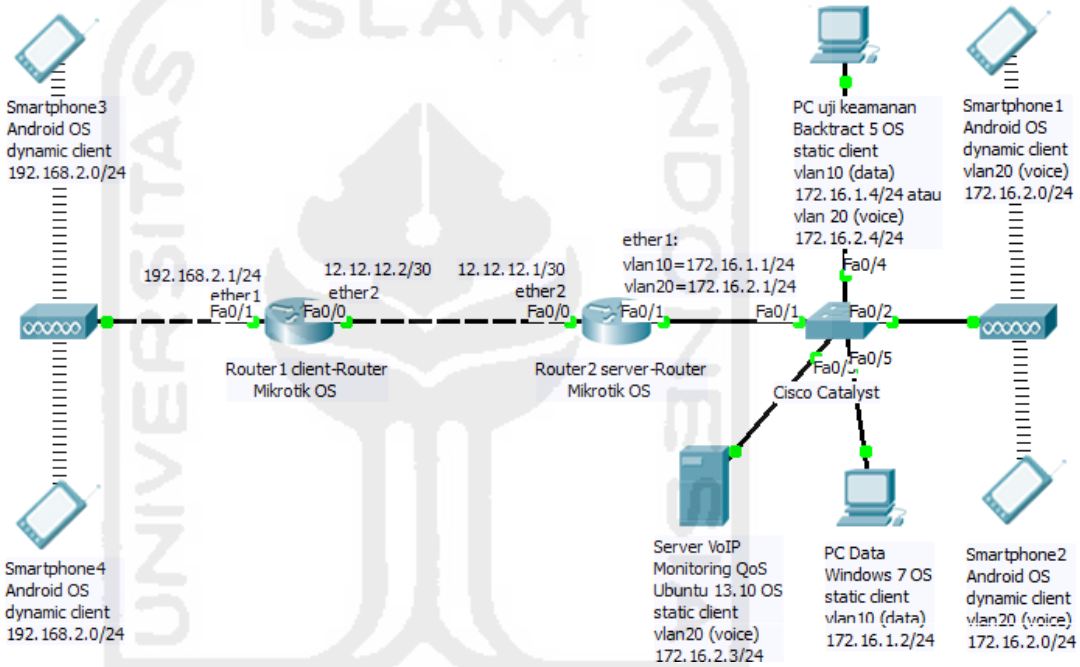
Topologi pada Gambar 3.4, menjelaskan bahwa sistem telah menggunakan kedua metode, yakni MPLS-VPN dan segmentasi VLAN yang disertai dengan pengaturan *firewall*, sehingga semua konfigurasi yang digunakan masing-masing metode, diterapkan dalam topologi ini.



Gambar 3.4 Topologi VoIP Menggunakan Metode MPLS-VPN dan Segmentasi VLAN (disertai *Firewall*)

5) Topologi VoIP dengan menggunakan metode segmentasi VLAN (tanpa *firewall*).

Topologi pada Gambar 3.5, menjelaskan bahwa sistem hanya menerapkan metode segmentasi VLAN tanpa disertai pengaturan *firewall*. Hal ini bertujuan untuk, mengetahui perbedaan hasil pengujian baik servis maupun keamanan VoIP, sesudah dan sebelum diterapkannya pengaturan *firewall* di dalam metode segmentasi VLAN.



Gambar 3.5 Topologi VoIP Menggunakan Metode Segmentasi VLAN (tanpa *Firewall*)

Dari perancangan di atas, maka dapat diketahui perangkat apa yang dibutuhkan untuk membangun jaringan VoIP. Dalam laporan ini, penulis hanya menunjukkan perangkat yang dibutuhkan, lengkapnya untuk topologi yang menggunakan metode pengamanan MPLS-VPN dan segmentasi VLAN yang disertai dengan pengaturan *firewall*. Perangkat tersebut antara lain

- Tiga *Personal Computer*, masing-masing digunakan sebagai *server* VoIP dan *monitoring* QoS, PC data (PC yang khusus digunakan di jaringan VLAN data), dan PC penguji yang digunakan di VLAN data dan VLAN suara secara bergantian,
- *Router*, digunakan untuk mengatur jaringan MPLS-VPN,
- *Switch*, digunakan untuk melakukan segmentasi VLAN dan pengaturan *firewall*,
- *Access Point*, digunakan untuk membantu *Router* dalam melakukan distribusi IP (DHCP *pool*), dan
- *Smartphone*, digunakan sebagai *client* VoIP.

Perangkat yang digunakan sengaja disusun demikian, agar dapat digunakan untuk menerapkan metode pengamanan VoIP, yakni MPLS-VPN dan segmentasi VLAN, yang disertai dengan pengaturan *firewall*. Dengan susunan tersebut, maka konfigurasi MPLS-VPN dapat dilakukan pada perangkat *Router*, sedangkan segmentasi VLAN serta *firewall*, dapat dilakukan pada perangkat *Switch*.

Dalam implementasinya, konfigurasi VPN diterapkan pada *Router* 1 dan *Router 2*. *Router* 1 berperan sebagai *client* VPN, sedangkan *Router* 2 berperan sebagai *server* VPN. Jenis VPN yang digunakan dalam tugas akhir ini adalah OpenVPN, yakni VPN yang memiliki kemampuan enkripsi lebih variatif daripada jenis VPN lainnya (IVPN, 2015). Kemudian untuk menunjang kinerja VPN, dibutuhkan konfigurasi MPLS yang disertai dengan pengaturan *firewall*. Konfigurasi tersebut dilakukan dengan, menambahkan *IP Address* sebagai sesi LDP (*Label Distribution Protocol*) pada *interface bridge.* Tujuan dari diterapkannya *interface bridge* adalah, untuk memastikan bahwa sesi LDP yang terletak di antara kedua *Router*, tidak akan mengalami masalah ketika terjadi perubahan *interface* atau *IP Address* pada kedua *Router*.

3.2. Kebutuhan Perangkat Keras

Dalam membangun sistem VoIP secara nyata, dibutuhkan beberapa perangkat keras yang mendukung jaringan tersebut terhubung. Perangkat keras yang dibutuhkan dalam pengerjaan tugas akhir ini, antara lain:

a. *RouterBoard* 493G, digunakan sebagai *server Router*
b. Laptop, digunakan untuk menjalankan semua sistem operasi yang dibutuhkan, yakni satu ISO *Mikrotik RouterOS* (sebagai *client Router*), satu ISO *Ubuntu 13.10* (sebagai *server* VoIP dan PC *monitoring* QoS), satu ISO *Backtrack* 5 (sebagai PC penguji keamanan jaringan) dan satu ISO *Windows* 7 (sebagai PC data).
c. *Cisco Catalyst Switches* 2960 *Series*, digunakan untuk menggabungkan banyak perangkat dalam satu jaringan, dan melakukan konfigurasi segmentasi VLAN data dan suara, serta *firewall*.
d. *Access Point TP Link*, digunakan untuk membantu *server Router* dan *client*, dalam melakukan distribusi IP (DHCP *pool*) ke perangkat klien.
e. *Smartphone Android*, digunakan sebagai perangkat klien.

f. Kabel LAN (*Local Area Network*) tipe *straight*, sebagai penghubung antar perangkat yang mengarah ke *Switch*.
g. Kabel LAN tipe *cross*, untuk menghubungkan laptop dengan *Access Point*, dan *server Router* dengan *client Router*.

3.3. Kebutuhan Perangkat Lunak

Berikut perangkat lunak yang dibutuhkan, untuk implementasi *server* VoIP.

a. *Cisco Packet Tracer*, digunakan untuk membuat topologi jaringan.
b. Sistem Operasi *Ubuntu* 13.10, digunakan sebagai dasar pembangunan *server* VoIP serta *monitoring* QoS.
c. ISO *MikroTik RouterOS* Level 6, digunakan sebagai *client Router*.
d. ISO *Windows* 7, digunakan sebagai PC data yang diletakkan khusus pada jaringan VLAN data.
e. ISO *Backtrack* 5, digunakan sebagai PC penguji keamanan jaringan, yang diletakkan pada jaringan VLAN data dan suara, secara bergantian.
f. *VirtualBox*, digunakan sebagai media untuk memudahkan pengguna dalam melakukan pengaturan jaringan, sebab dengan aplikasi ini, pengguna dapat menggunakan banyak sistem operasi dalam satu PC secara virtual.
g. *Wine*, adalah aplikasi yang digunakan untuk mendukung agar, aplikasi yang berjalan pada Sistem Operasi *Windows*, dapat berjalan pula pada Sistem Operasi *Unix*.
h. *Winbox*, adalah aplikasi yang digunakan untuk membantu *user* dalam melakukan konfigurasi *Router*, melalui tampilan grafis.
i. *Package* VoIP *Asterisk*, berisi sekumpulan *package* yang dibutuhkan untuk pembangunan *server* VoIP.
j. *FileZilla*, adalah aplikasi yang digunakan untuk mengirim *file* sertifikat dan kunci *server* OpenVPN, ke *server Router*.
k. *PuTTY*, adalah aplikasi yang digunakan untuk melakukan *remote* ke suatu *device*. Dalam hal ini, *putty* digunakan untuk melakukan *remote* ke *device Cisco Catalyst Switches*.

3.4. Pembangunan Sistem

Dalam tugas akhir ini, akan dijelaskan tahapan pembangunan *server* VoIP dari penerapan metode pengamanan VoIP, hingga ke penataan jaringan. Berikut beberapa tahapan yang dijalankan terlebih dahulu, untuk mendukung terbentuknya model pengamanan VoIP.

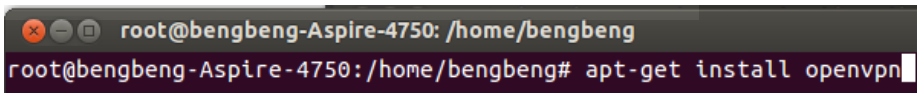
3.4.1 Konfigurasi OpenVPN pada *Mikrotik RouterOS*

Dalam pengerjaan tugas akhir ini, penulis menggunakan pengaturan VPN jenis *site to site*, dimana pengaturan VPN cukup dilakukan antar *Router*. Dengan cara tersebut, sekumpulan klien yang ada di bawah *Router*, dapat saling terhubung dalam jaringan yang berbeda, tanpa perlu melakukan pengaturan VPN kembali pada masing-masing klien, yang dikenal dengan istilah *remote* VPN. Cara *remote* VPN tersebut dirasa kurang efektif, sebab setiap klien harus memiliki masing-masing ID VPN, agar dapat terhubung ke *server Router*. Oleh karena itu, pengaturan jenis *site to site*, diharapkan dapat memudahkan perawatan jaringan VPN, dalam jangka waktu yang panjang.

Berikut tahapan konfigurasi OpenVPN pada *Router* 2, sebagai ***server Router***. Tahapan yang dilakukan pada setiap langkahnya, akan disertai dengan penjelasan.

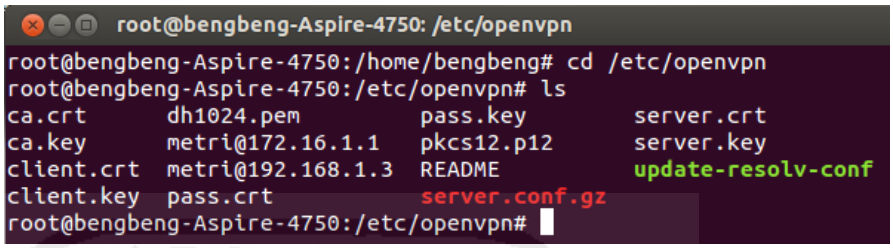
a) Agar konfigurasi VPN yang dilakukan dapat memperoleh hak akses penuh, maka perlu dimasukkan **sertifikat dan kunci *server* VPN** ke dalam *server Router*, melalui bantuan terminal *Ubuntu* dan aplikasi *FileZilla*.

1) Melakukan instalasi OpenVPN pada terminal *Ubuntu*.



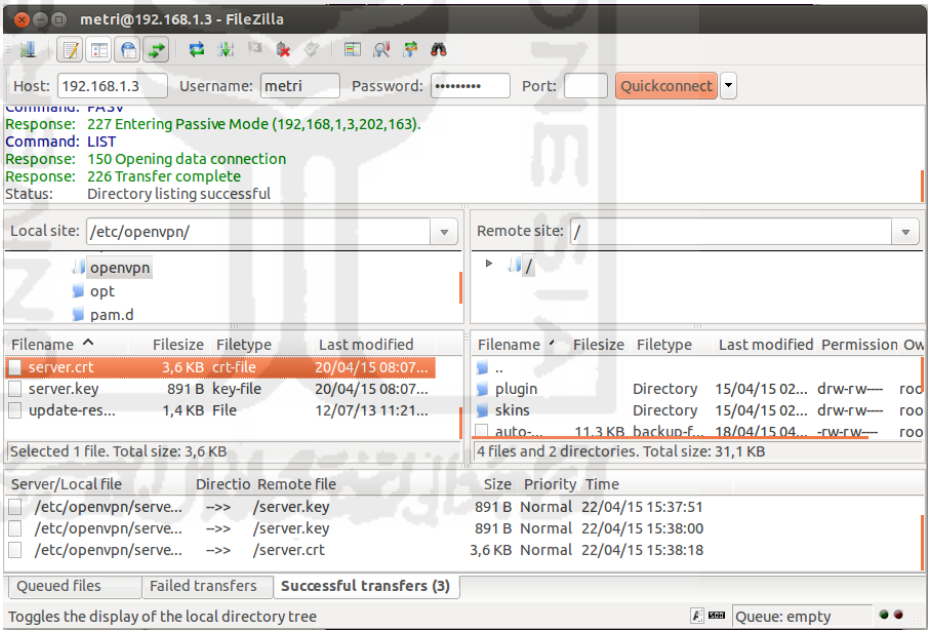
Gambar 3.6 *Install* OpenVPN

2) *File* hasil instalasi OpenVPN, dipindahkan ke dalam satu folder "*/etc/openvpn*", untuk memudahkan pengguna dalam menyaring sertifikat dan kunci VPN yang dibutuhkan.



Gambar 3.7 Sekumpulan *File* OpenVPN

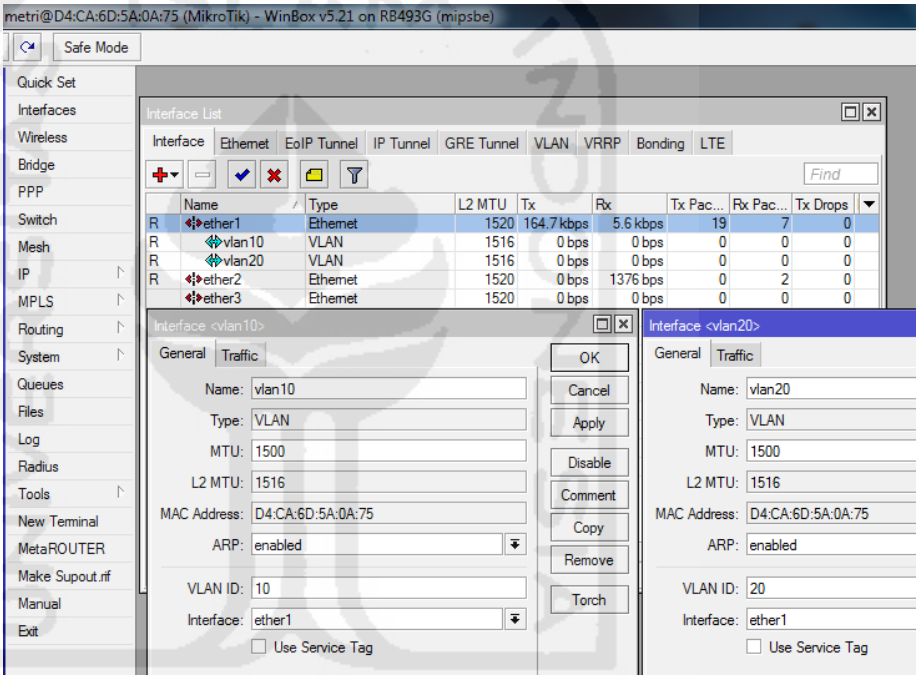
3) Mengirim *file server.key* dan *server.crt* ke *server Router* melalui aplikasi *FileZilla*. *File* dikirim, untuk membantu konfigurasi penuh pada *server* VPN.



Gambar 3.8 Pengiriman *File Server.key* dan *Server.crt* ke *Server Router*

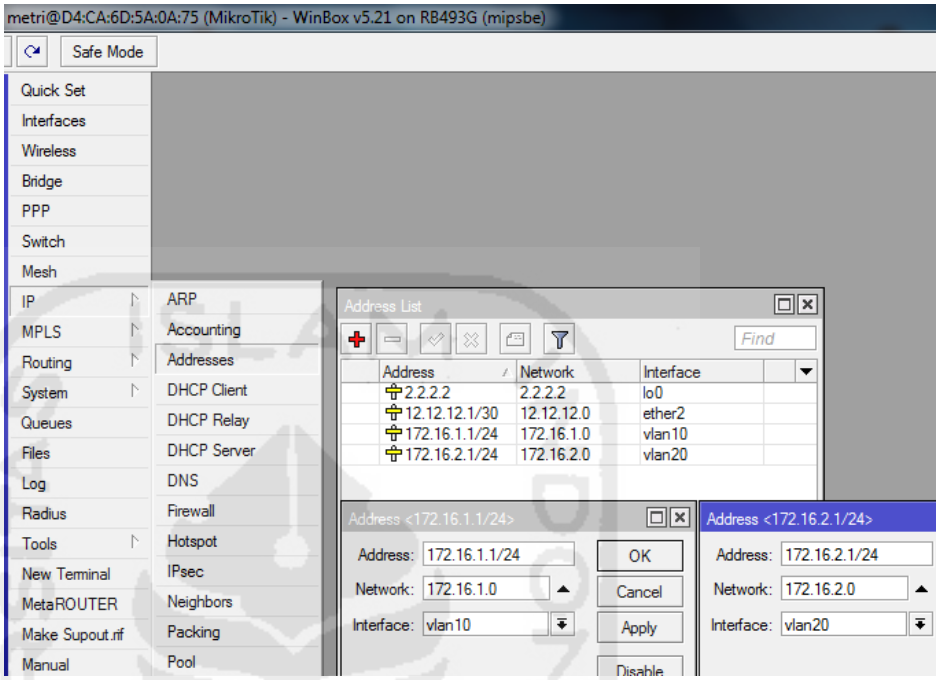
b) Tahap konfigurasi *server* VPN pada ***server Router***.

1) Menyediakan *interface vlan10* (VLAN data) dan *vlan20* (VLAN suara), di dalam *interface* yang akan terhubung ke klien. Jenis paket yang lewat pada kedua *interface* VLAN, sama-sama diperlakukan sebagai data, hanya jenis pelabelan dan pengaturan prioritasnya yang berbeda. Pembagian ini akan digunakan, untuk membantu proses segmentasi VLAN pada *layer* 2 (*data link*).



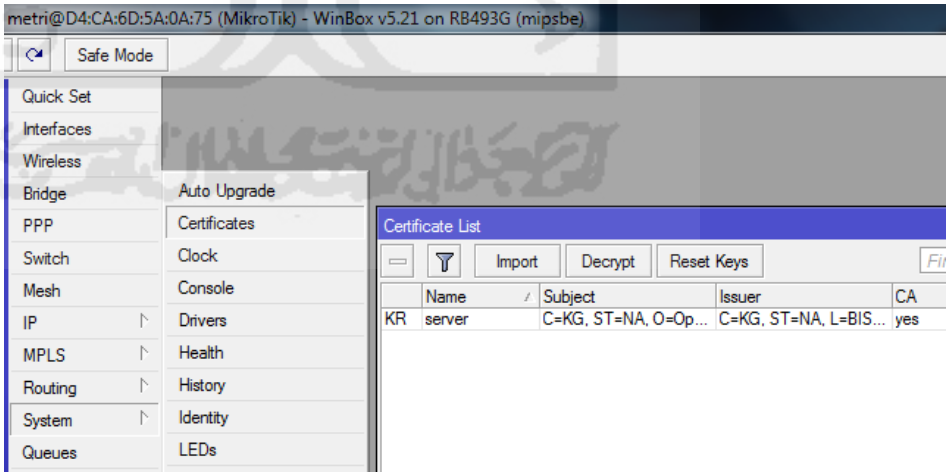
Gambar 3.9 *Setting Interface* VLAN Data dan Suara

2) Membuat alamat IP *static* pada *interface vlan10*, *vlan20*, dan *ether2*. Hal yang sama, juga dilakukan pada *client Router*.



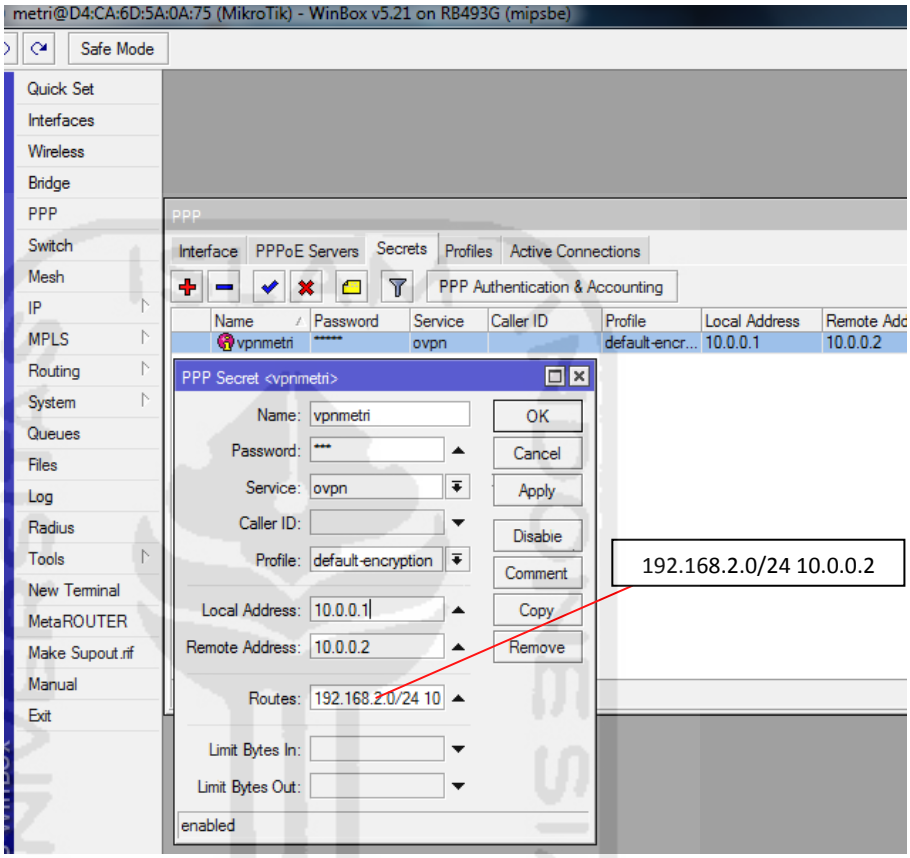
Gambar 3.10 *Setting* Alamat IP pada *Server Router*

3) Memasukkan serta mengaktifkan *file server.crt* dan *server.key*, ke dalam daftar sertifikat.



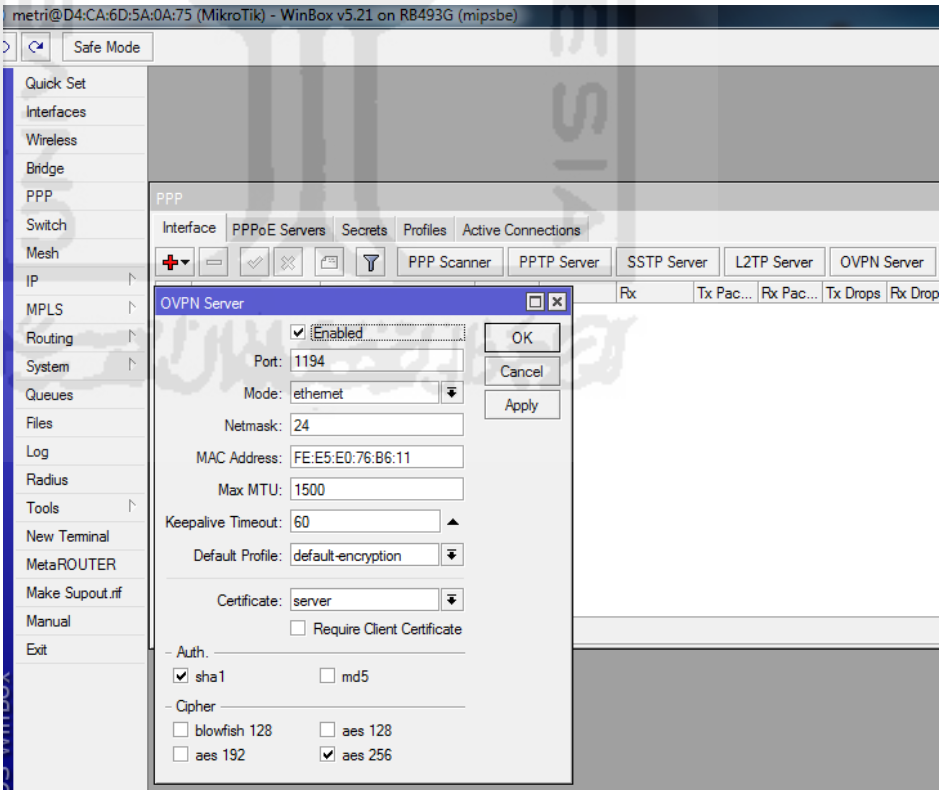
Gambar 3.11 *Input File Server* OpenVPN

4) Memberikan identitas pada *server* VPN, yang bertujuan sebagai tanda pengenal, agar klien dapat terdaftar dan terhubung ke jaringan *server*.



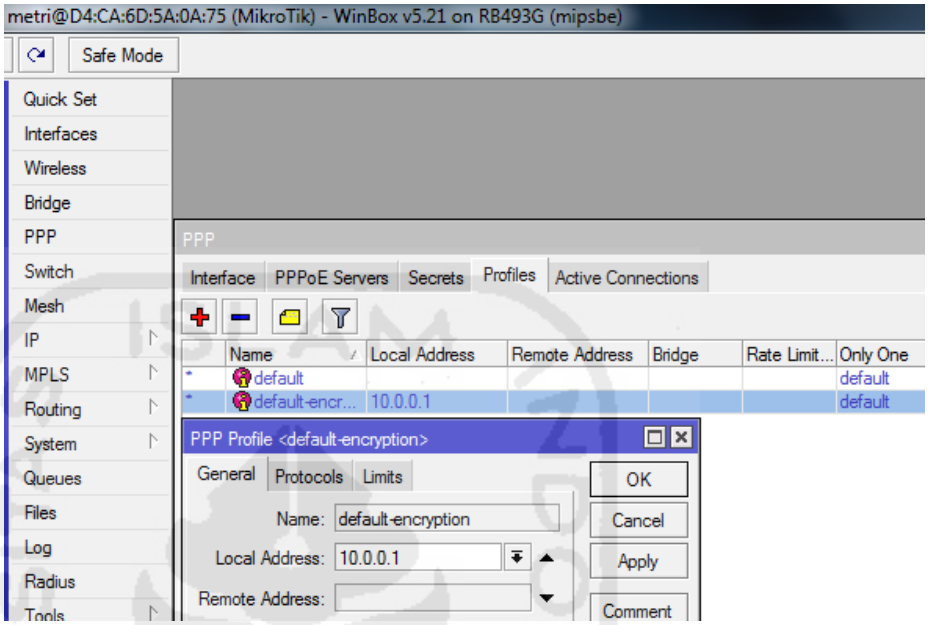
Gambar 3.12 *Setting* Identitas *Server* VPN

5) Dalam pengaturan ini, penulis memilih *SHA1* sebagai *authentication algorithm* (asimetris) dan *AES* 256 sebagai *cipher alghorithm* (simetris). Hal ini dikarenakan, keduanya memiliki keunggulan berupa tingkat keamanan yang lebih baik daripada algoritma yang sebelumnya, yakni *MD5* untuk *SHA1* dan *DES* untuk *AES* 256. Dalam salah satu artikel, disebutkan bahwa algoritma *AES* (*Advanced Encryption Standard*) memiliki peforma yang cukup baik, bahkan lebih baik daripada algoritma *Blowfish*, sebab algoritma AES dapat melakukan enkripsi dan dekripsi, jauh lebih cepat daripada algoritma *Blowfish* (Riftadi, 2009). Begitupun dengan pengamanan SHA1, yang memiliki kemampuan lebih baik daripada MD5, sebab *output* yang dihasilkan dari SHA1, lebih panjang 32 bit daripada MD5, sehingga membuatnya lebih tahan terhadap serangan *brute force* (Shaugi, 2012).



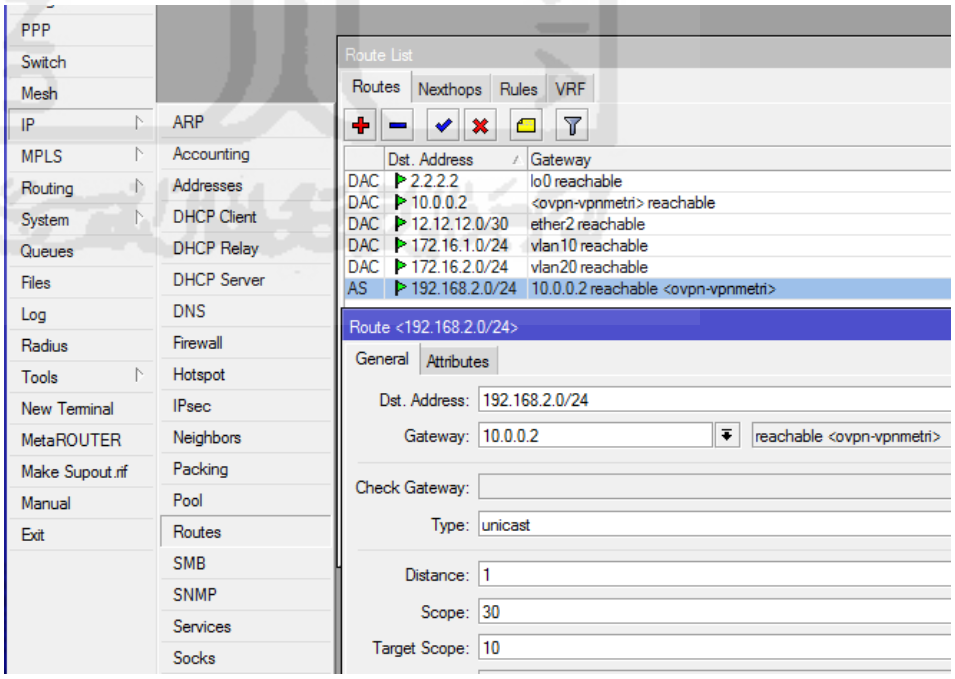
Gambar 3.13 Pengaturan Keamanan pada *Server* VPN

6) Mengatur profil *server* VPN, untuk memastikan alamat IP mana yang akan dijadikan sebagai *Local Address*.



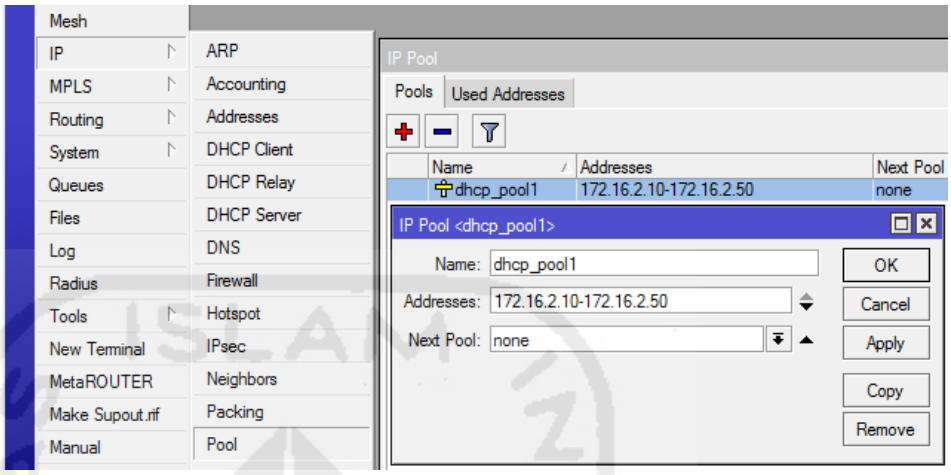
Gambar 3.14 *Setting* Profil *Server* VPN

7) Membuat tabel *routing* dengan mengarahkan jaringan ke *client Router*, melalui *interface* VPN.



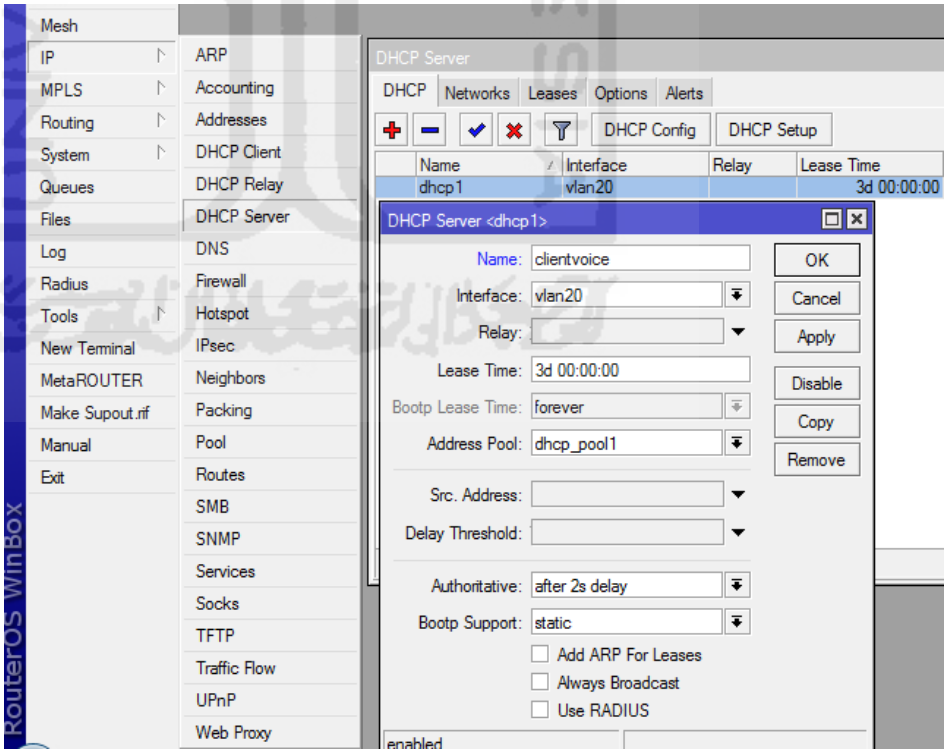
Gambar 3.15 *Routing Table Server* VPN

8) Melakukan distribusi alamat IP pada jaringan VLAN suara (*vlan20*), yakni dari alamat 172.16.2.10 s/d 172.16.2.50.



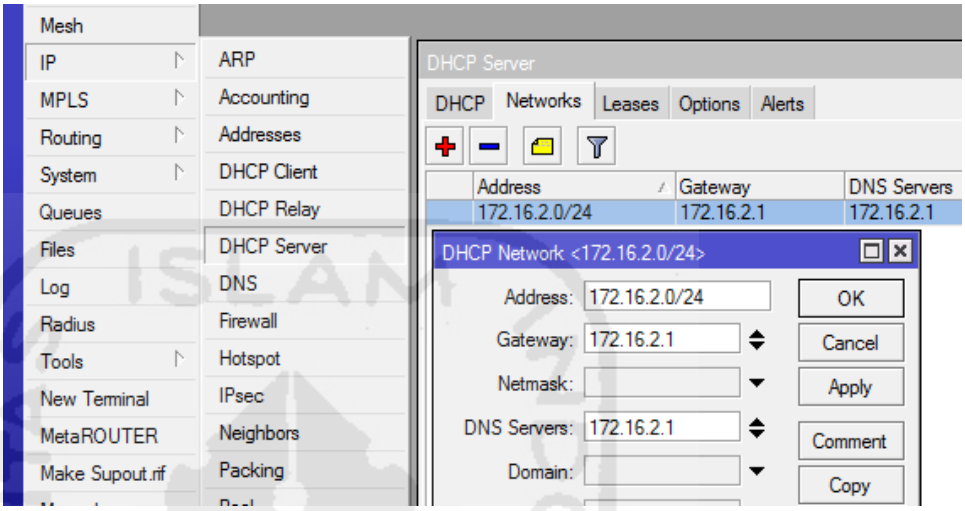
Gambar 3.16 *Setting Range* Alamat IP pada Jaringan VLAN Suara

9) Mengatur jalur distribusi alamat IP, agar melewati *interface* yang digunakan *client* VoIP (*vlan20*).



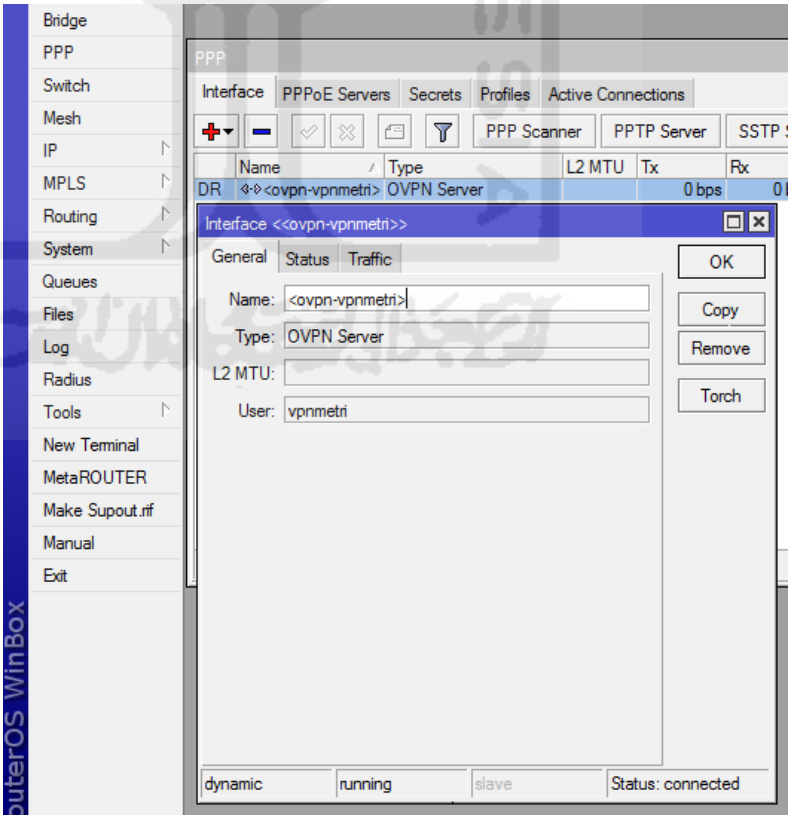
Gambar 3.17 *Setting Interface* DHCP *Server*

10) Mengatur lalu lintas jaringan yang digunakan *client* VoIP, dengan menambahkan *gateway* dan DNS *servers* yang telah diatur pada pengaturan alamat IP sebelumnya.



Gambar 3.18 *Setting Network* DHCP *Server*

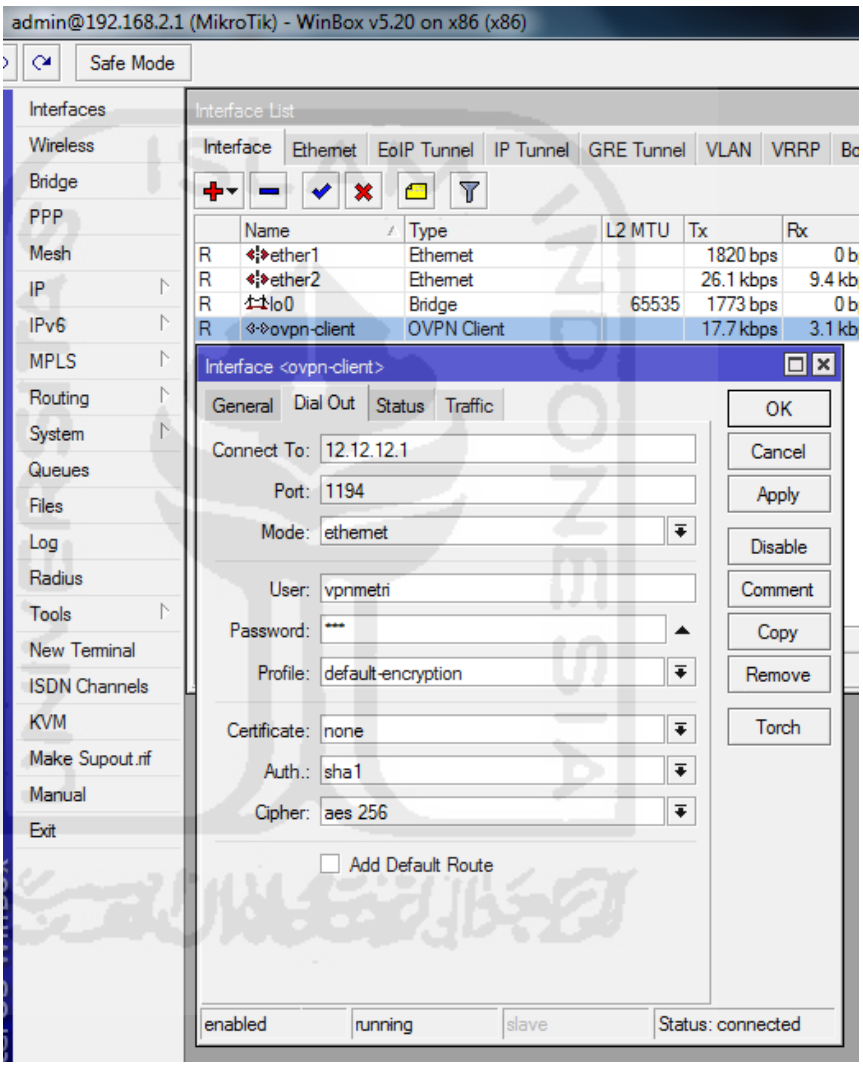
11) Kondisi jika *server* VPN, telah saling terhubung dengan *client* VPN (status *connected*).



Gambar 3.19 Status *Server* VPN "*Connected*"

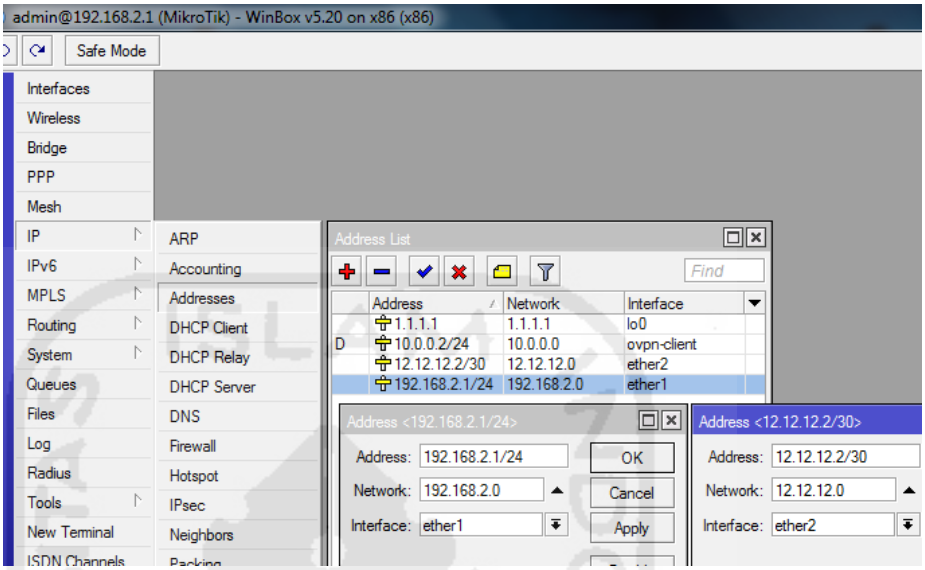
Setelah melakukan pengaturan *server* VPN pada *server Router*, tahap berikutnya yakni melakukan konfigurasi *client* VPN pada ***client Router***.

1) Memasukkan alamat IP *server Router* melalui "*Dial Out*". Alamat IP tersebut yang digunakan klien, agar dapat terhubung ke jaringan *server*.



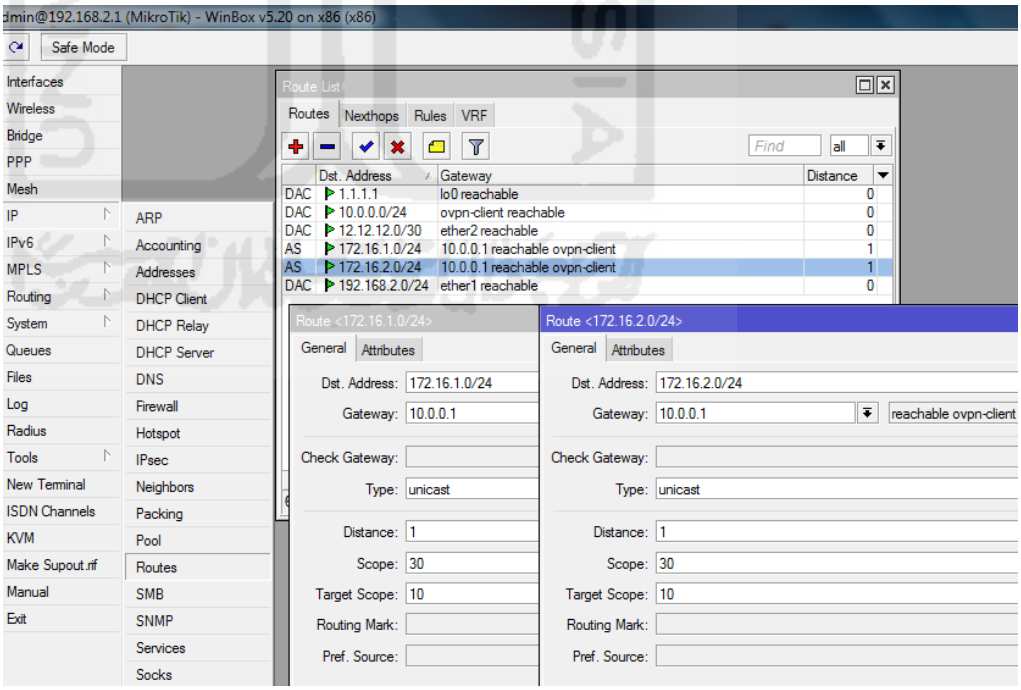
Gambar 3.20 *Client* VPN *Connect to Server* VPN

2) Menambahkan alamat IP *client Router*, dalam *interface ether1* dan *ether2*.



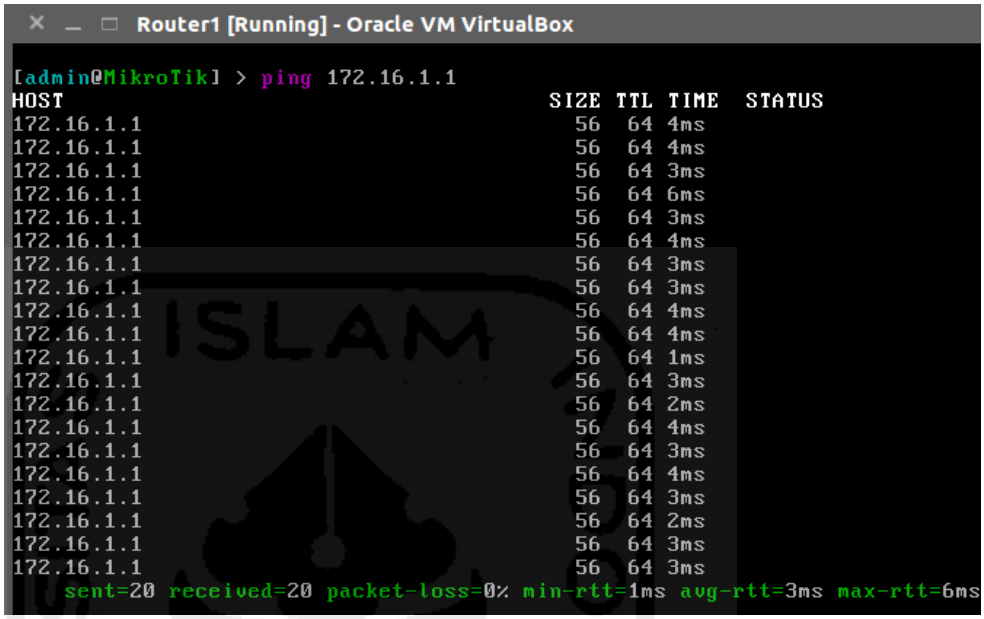
Gambar 3.21 Alamat IP *Client* VPN

3) Menambahkan tabel *routing*, dengan tujuan ke jaringan *vlan10* dan *vlan20* pada *server Router*, melalui *interface* VPN.

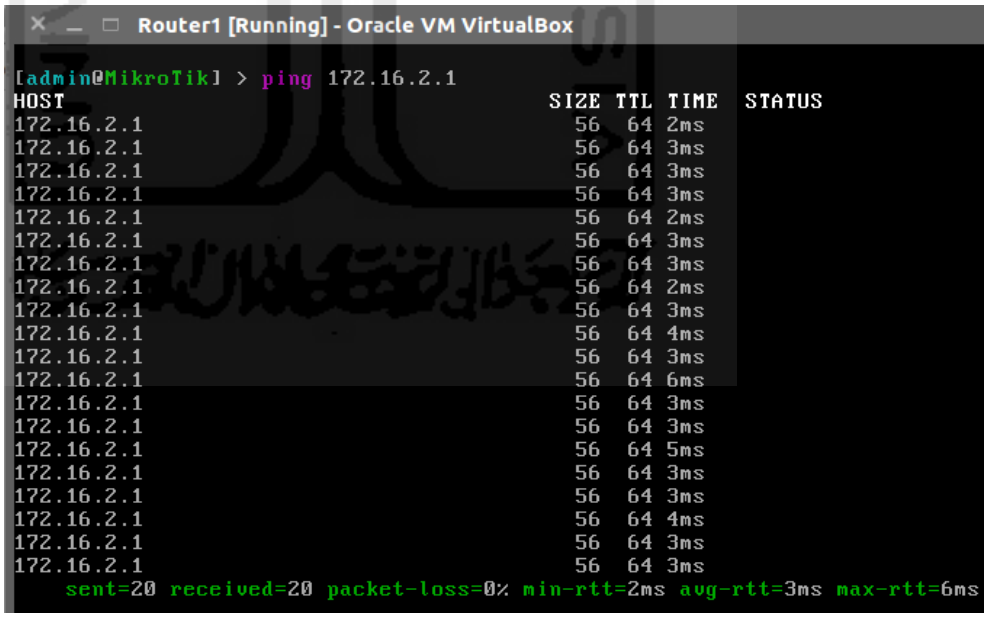


Gambar 3.22 *Routing Table Client* VPN

4) Melakukan pengetesan jaringan, dari *client Router* ke *server Router* (*vlan10* dan *vlan20*), seperti pada Gambar 3.23 dan Gambar 3.24.



Gambar 3.23 Pengetesan Jaringan dari *Client Router* ke Jaringan *Vlan10*



Gambar 3.24 Pengetesan Jaringan dari *Client Router* ke Jaringan *Vlan20*

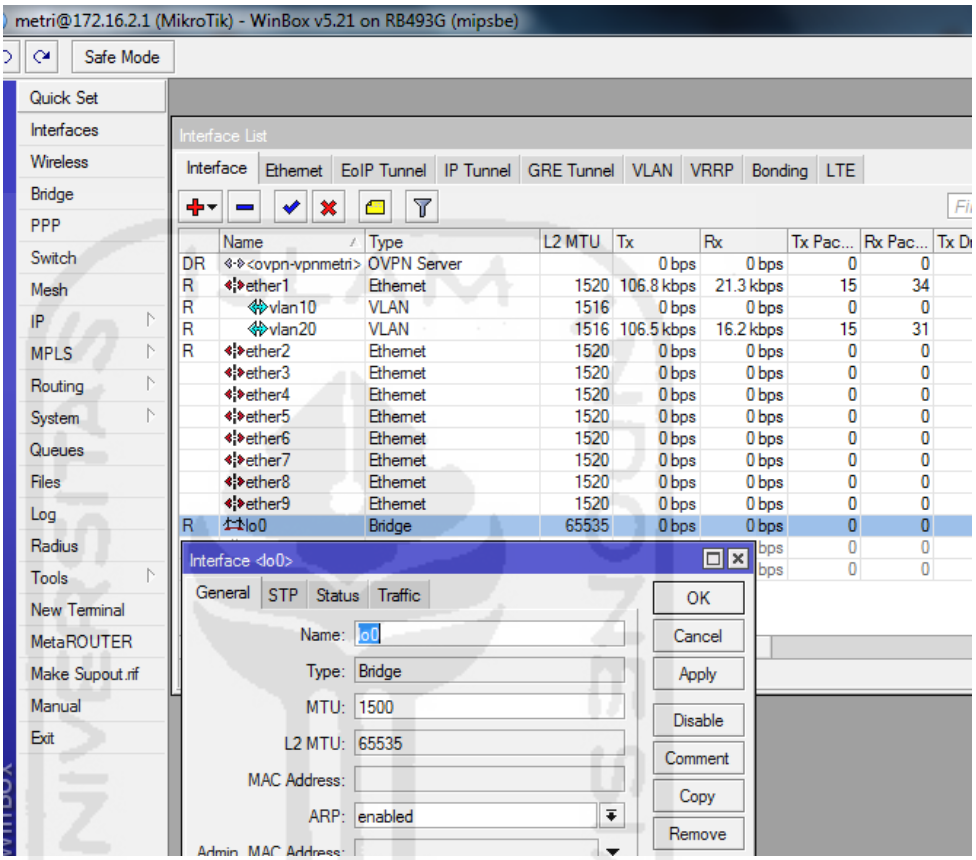
3.4.2 Konfigurasi MPLS pada *Mikrotik RouterOS*

Konfigurasi MPLS (*Multiprotocol Label Switching*) digunakan untuk menunjang kinerja VPN, sehingga jaringan menjadi lebih *reliable*, aman, dan terkendali. Hal ini dikarenakan, adanya pemisahan jaringan *Router* ke dalam area yang berbeda, dengan membuat alamat IP baru sebagai identitas. Pembuatan identitas tersebut, dilakukan dengan bantuan LSR (*Label Switching Router),* yakni komponen MPLS yang berfungsi meneruskan paket ke *layer* 3 (lapisan *network*). Dengan rancangan yang lebih terstruktur tersebut, yang membuat MPLS menjadi lebih *reliable* dalam menghemat *bandwidth* (Bacharudin, 2011).

Penggunaan MPLS ini menjadi penting, karena adanya metode *routing* OSPF (*Open Shortest Path First*) dalam konfigurasinya, sehingga membuat proses *routing* menjadi lebih cepat. Hal ini dikarenakan, OSPF bekerja dengan menggunakan protokol *routing link state*, yakni protokol yang bergerak dengan mengumpulkan semua informasi terlebih dahulu, untuk menghitung jarak terpendek ke setiap node. Algoritma yang digunakan pada metode ini adalah SPF (*Sort Path First*), dengan model perhitungan ($\boldsymbol{cost} = \frac{\mathbf{100.000\ Kbps}*}{\boldsymbol{link\ speed}}$) (Muhammad dan Suhardi, 2011).

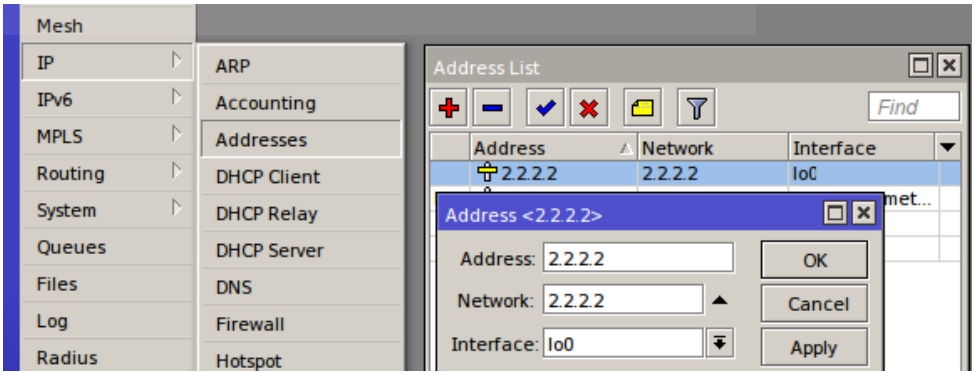
Berikut tahapan konfigurasi MPLS pada ***server Router***.

1) Membuat *interface loopback (bridge). Interface* ini yang akan digunakan untuk konfigurasi MPLS.



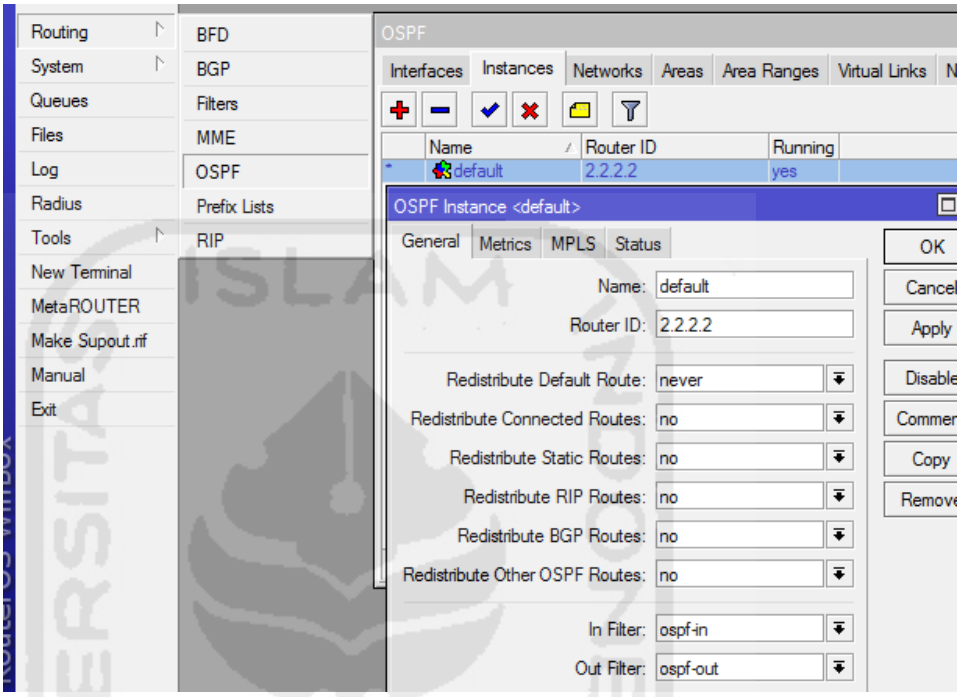
Gambar 3.25 *Interface Loopback* pada *Server Router*

2) Membuat alamat IP *loopback*, yang akan digunakan sebagai identitas untuk membedakan area jaringan.



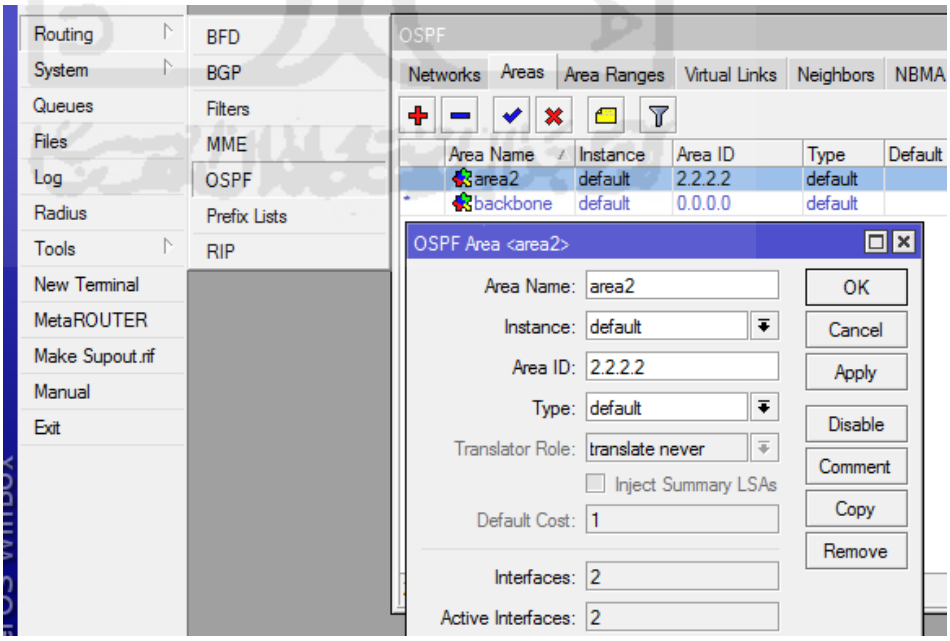
Gambar 3.26 Membuat Alamat IP *Loopback* pada *Server Router*

3) Tahap menjadikan alamat IP *loopback*, sebagai ID *server Router*. ID *Router* tersebut diatur, agar dapat melakukan *routing table*, dengan menggunakan metode OSPF.



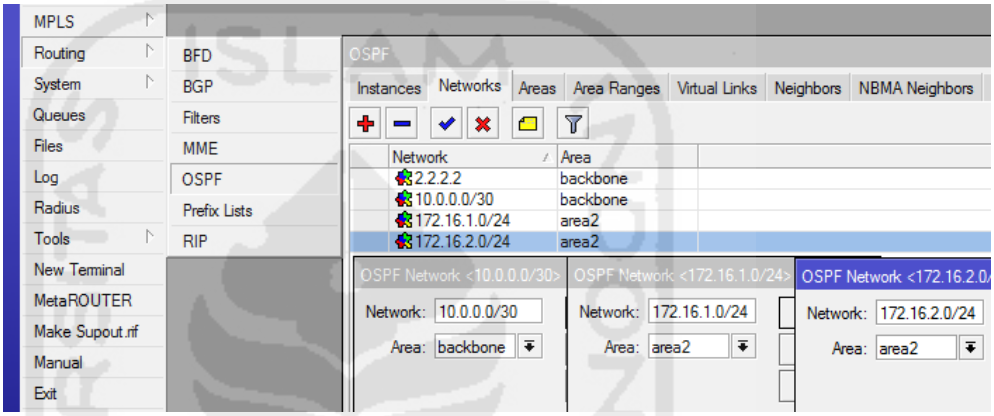
Gambar 3.27 Pengaturan Alamat IP *Loopback* sebagai ID *Server Router*

4) Memberikan penamaan area terhadap alamat IP *server loopback*.



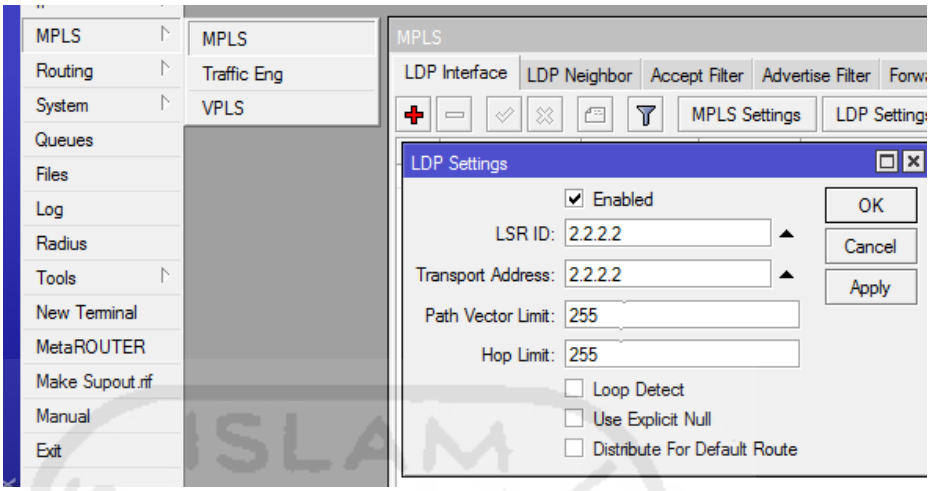
Gambar 3.28 Pengaturan Area *Server Router*

5) Penamaan area pada alamat IP *loopback*, digunakan untuk mengumpulkan *client network* (172.16.1.0/24 dan 172.16.2.0/24) ke dalam satu area. Kemudian untuk *network* lainnya, menggunakan tipe area *backbone* (jaringan utama), agar *network* tersebut dapat terkoneksi dengan kecepatan tinggi. Langkah inilah yang dilakukan untuk menerapkan metode *routing* OSPF, yang bertujuan meningkatkan kecepatan jaringan.



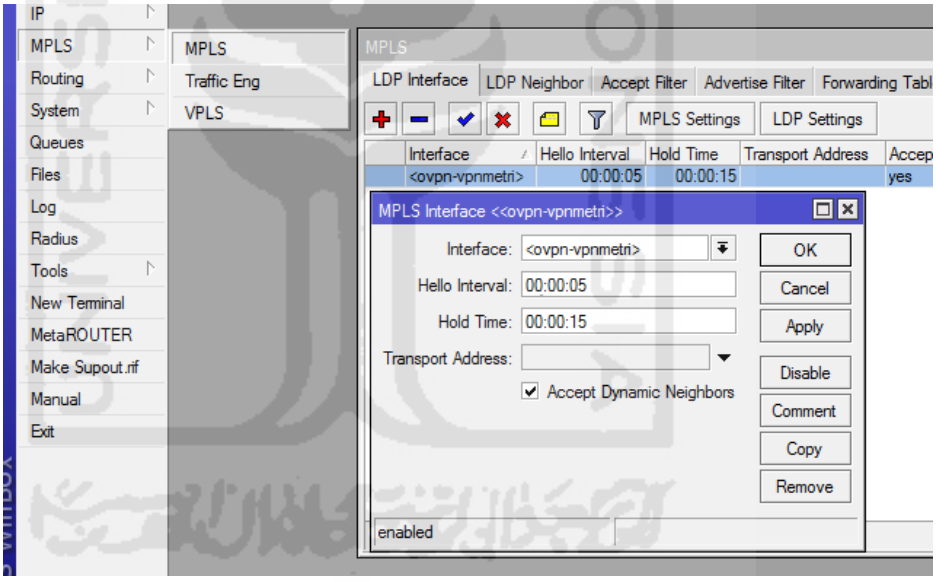
Gambar 3.29 *Setting* Area *Network* pada *Server Router*

6) Mengaktifkan *Label Distribution Protocol* pada "*LDP Settings"*, dengan tujuan untuk mendistribusikan informasi ke setiap LSR (*Label Switching Router*). LSR sendiri merupakan bagian dari *Router* yang berada di ujung jaringan MPLS, yang berfungsi mengatur lalu lintas data yang lewat, dengan melakukan pengecekan sekali. Pengaturan tersebut dilakukan dengan menetapkan LSP (*Label Switched Path*), yang digunakan sebagai jalur untuk meneruskan paket data, yang dibawa *label swapping* dari satu *node* MPLS ke *node* MPLS lainnya.



Gambar 3.30 *Enabled "LDP Settings"* pada *Server Router*

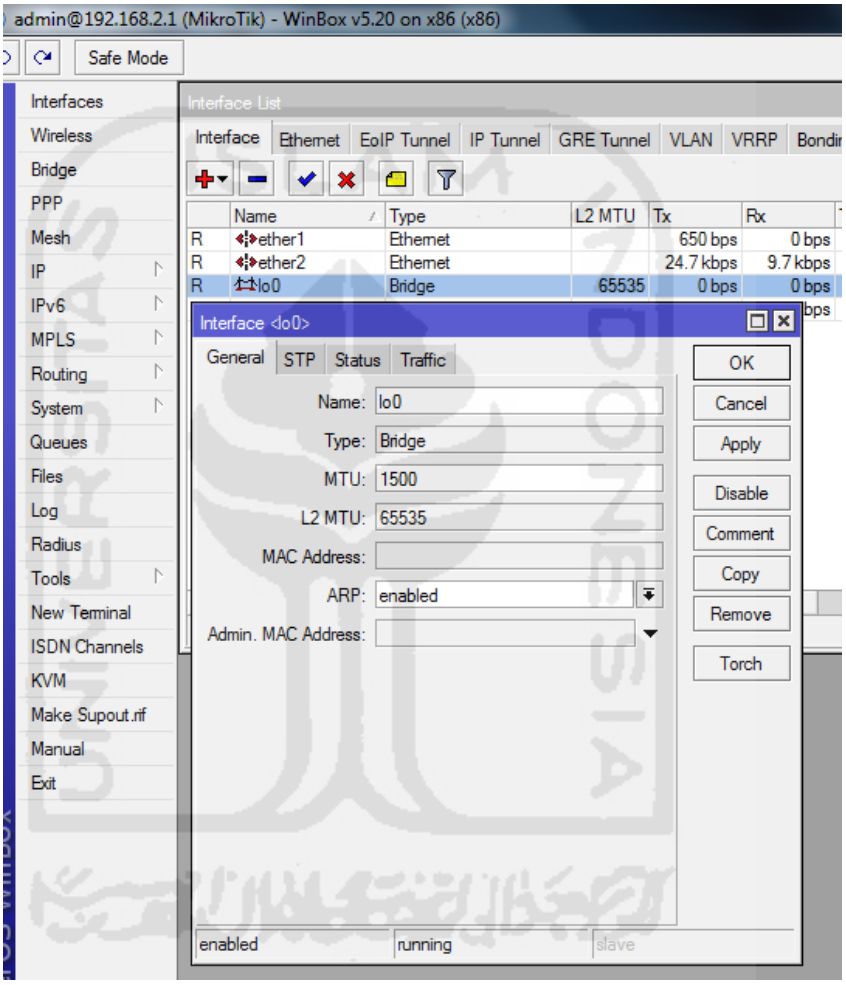
7) Mengaktifkan *interface* MPLS dengan *interface Router* VPN.



Gambar 3.31 *Interface* MPLS pada *Server Router*

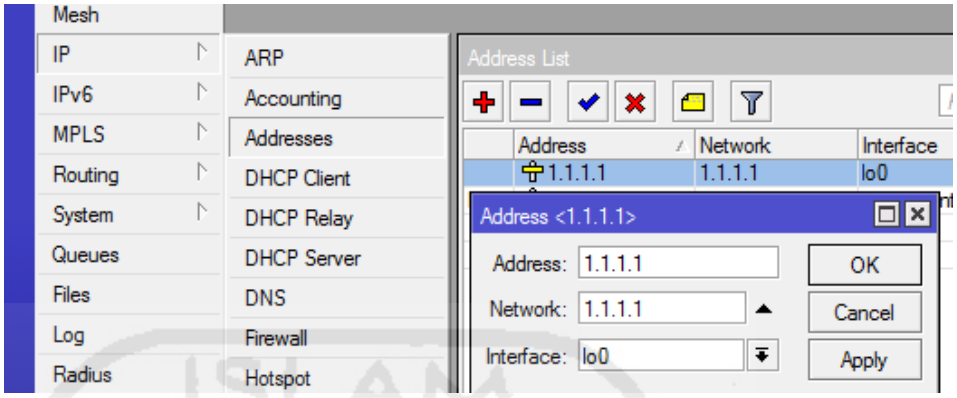
Tahapan selanjutnya, yakni dengan melakukan konfigurasi MPLS pada ***client Router***. Langkah konfigurasi MPLS pada *Router* ini, sama dengan langkah sebelumnya, hanya menyesuaikan dengan pengaturan IP yang telah dilakukan sebelumnya.

1) Membuat *interface loopback.*



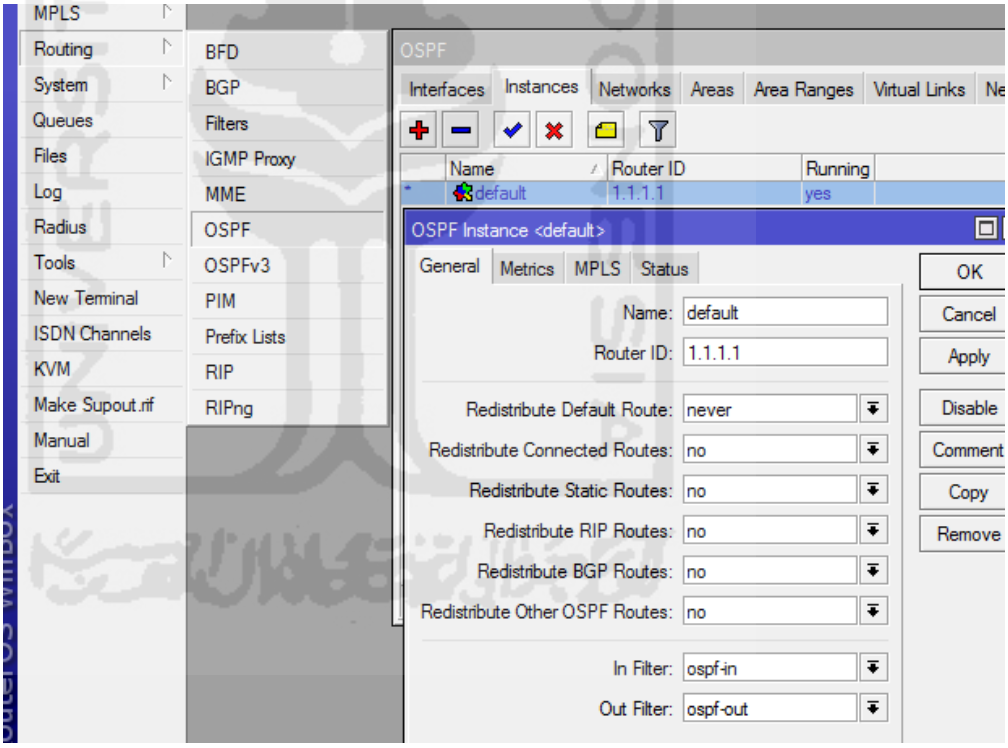
Gambar 3.32 *Interface Loopback* pada *Client Router*

2) Mengatur alamat IP *loopback*.



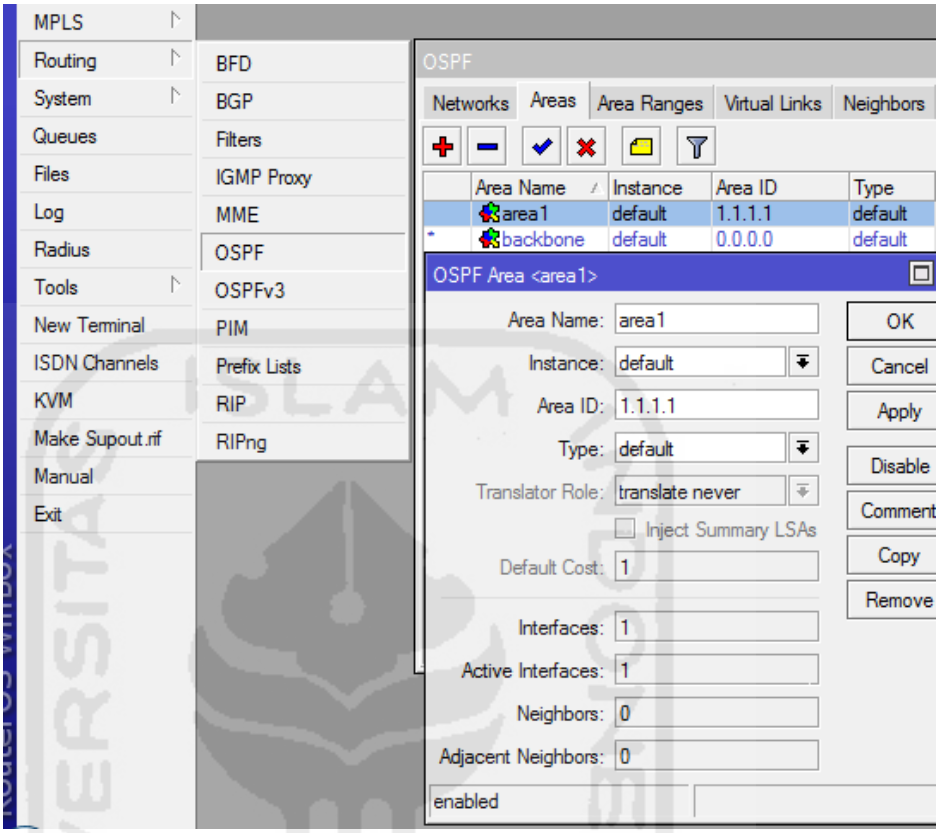
Gambar 3.33 Alamat IP *Loopback Client Router*

3) Mengatur alamat IP *loopback* sebagai ID *client Router*.



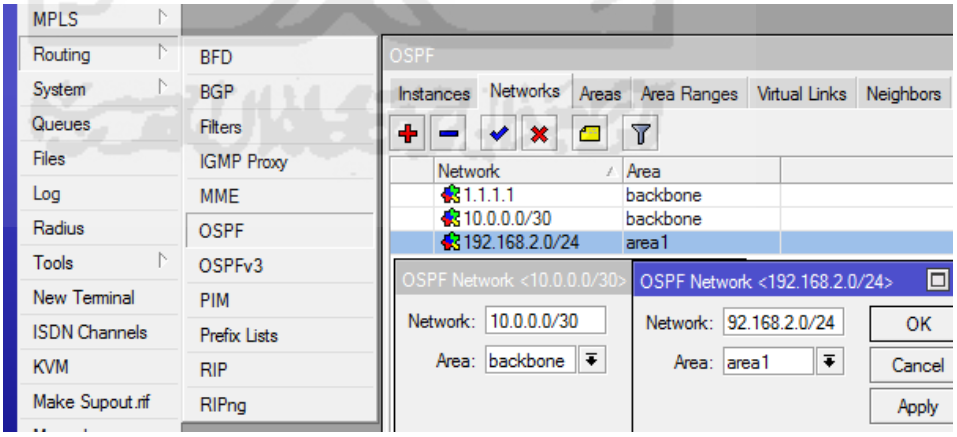
Gambar 3.34 Pengaturan Alamat IP *Loopback* sebagai ID *Client Router*

4) Penamaan area alamat IP *loopback* pada *client Router*.



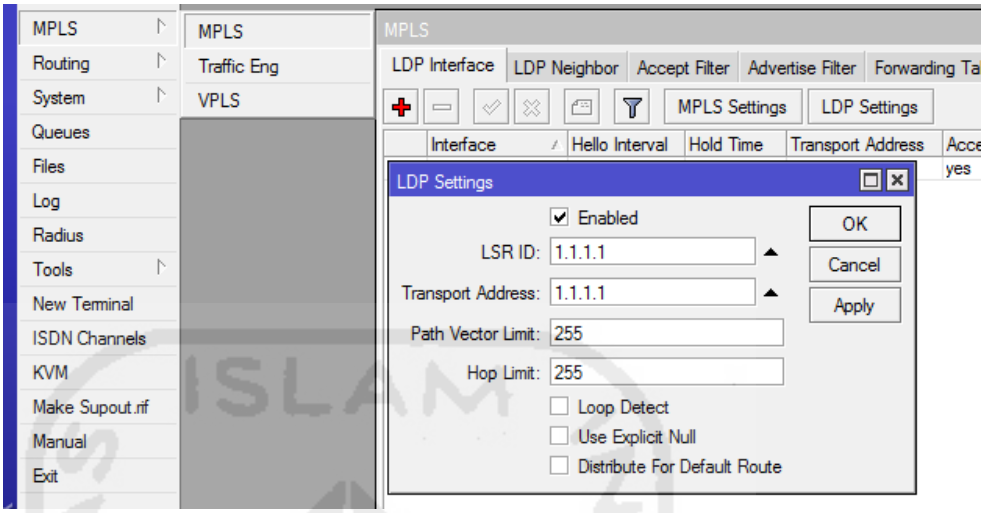
Gambar 3.35 Pengaturan Area *Client Router*

5) Pengaturan *network area*.



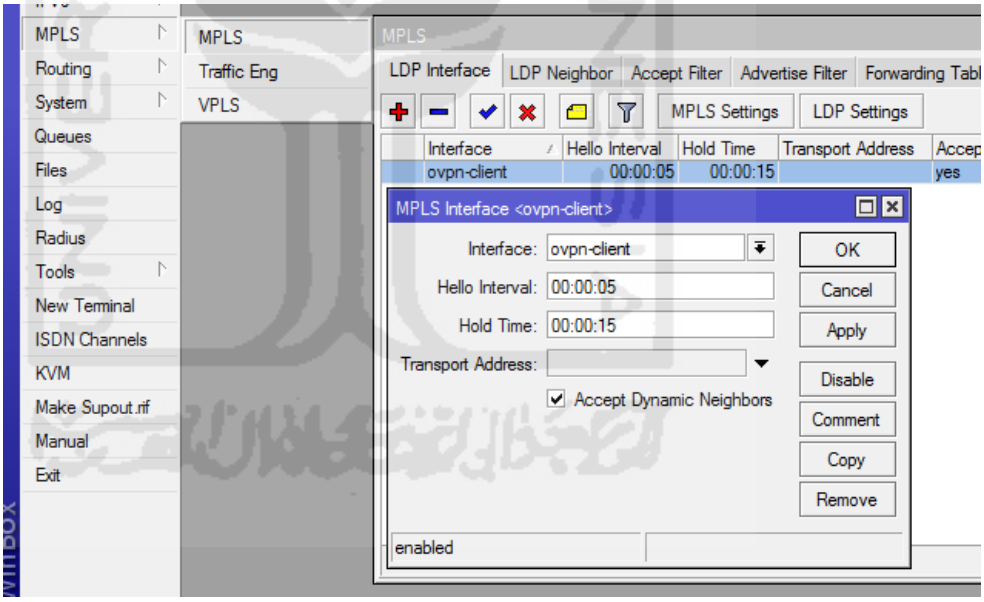
Gambar 3.36 *Setting Network Area* pada *Client Router*

6) Mengaktifkan "*LDP Settings*".



Gambar 3.37 *LDP Settings Client Router*

7) Mengaktifkan *interface* MPLS dengan *interface Router* VPN.



Gambar 3.38 *Interface* MPLS pada *Client Router*