ANALISA *VIRTUAL PRIVATE NETWORK* MENGGUNAKAN *OPENVPN DAN POINT TO POINT TUNNELING PROTOCOL*

ANALYSIS OF VIRTUAL PRIVATE NETWORK USING OPENVPN AND POINT TO POINT TUNNELING PROTOCOL

Prihatin Oktivasari & Andri Budhi Utomo
Politeknik Negeri Jakarta
Jl. Prof.Dr.G.A.Siwabessy, Kampus UI, Depok, 16425
ti2n_oktivasari@yahoo.com & andri.budhi8@gmail.com
(Diterima: 19/09/2016, Direvisi: 25/11/2016, Disetujui terbit: 30/11/2016)

Abstrak

Pengolahan data dapat dilakukan lebih efektif dengan sebuah server untuk dapat menghimpun seluruh data tersebut dan sebuah Virtual Private Network (VPN) server yang dapat dimanfaatkan oleh seluruh user. VPN dapat memberikan akses mudah dan cepat ke dalam sistem informasi kapan saja dan dimana saja melalui Internet dengan metode VPN Point to Point Tunneling Protocol (PPTP) dan OpenVPN. Kecepatan dan keamanan dari VPN dapat mencegah adanya kebocoran data. Implementasi dimulai dengan melakukan instalasi Operating System untuk VPN server dan instalasi VPN Server pada dua Virtual Machine, Operating System yang digunakan adalah Ubuntu Server 14.04. Pengujian yang dilakukan pada VPN Server meliputi pengujian *peforma*yaitu *packetloss*, *roundtrip* dan winSCP *transfer*, dan pengujiankeamanan yaitu *dial of service* dan *sniffing*. Hasil pengujian menunjukkan tidak adanya perbedaan yang signifikan dari sisi kehandalan dari kedua VPN Server tersebut, namun jika lebih diperhatikan OpenVPN lebih unggul dari PPTP begitu juga dari hasil pengujian keamanan.
Katakunci: data, PPTP, Open VPN, VPN server, VPN.

Abstract

Data processing can be done more effectively by a server to be able to collect all the data and a Virtual Private Network (VPN) server that can be used by all users. VPNs can provide quick and easy access to the information system anytime and anywhere via the Internet with VPN method Point to Point Tunneling Protocol (PPTP) and OpenVPN. Speed and security of a VPN can prevent data leakage. Implementation began with the installation of the Operating System for the VPN server and VPN Server installation on two Virtual Machine, Operating System used is Ubuntu Server 14:04. Tests conducted on the VPN Server includes the performa test, namely packetloss, *roundtrip*andwinSCP *transfer and the security test,namely dial of service*and*sniffing. The test results showed no significant difference in terms of the reliability of both the VPN server, but if more attention is superior to PPTP OpenVPN as well as from the results of safety testing.*
Keywords*: data, PPTP, OpenVPN, VPN server, VPN*

PENDAHULUAN

Pesatnya perkembangan teknologi memberikan fasilitas terhadap kita dalam segala hal, termasuk hal yang berkaitan dengan pekerjaan yang biasa dilakukan manusia. Perkembangan teknologi dalam bidang komputer membuat manusia menyadari akan pentingnya kebutuhan fasilitas yang disediakan oleh teknologi tersebut, khususnya dalam bidang pekerjaan. Dengan kemajuan zaman tersebut, membuat suatu instansi, baik pemerintah maupun swasta harus dapat melakukan proses pengolahan sistem informasi yang cepat, tepat dan akurat. Sebuah instansi harus dapat memanfaatkan kemajuan teknologi dalam bidang komputer dan jaringan untuk dapat menghemat tenaga, waktu, biaya dan lain-lain.

Untuk dapat mewujudkan hal tersebut, maka dukungan dari sisi infrastruktur jaringan pada sistem informasi dari instansi yang terkait, sangat diperlukan. Dengan memanfaatkan penggunaan jaringan internet, dapat membantu dalam mengatasi batasan jarak dan waktu. Kini seseorang dapat dengan mudah mengambil data atau mengolah data yang tersimpan di dalam jaringan lain, contohnya jaringan di dalam sebuah instansi seperti perusahaan baik negeri atau swasta juga dalam sebuah instansi yang bergerak dalam dunia pendidikan seperti sekolah dan perguruan tinggi, dari mana saja dan kapan saja. Hal tersebut dapat dilakukan jika jaringan tersebut terkoneksi dengan internet.

Cara atau sistem yang dapat digunakan untuk melakukan hal tersebut adalah menggunakan *Virtual Private Network* (VPN). Dengan VPN sebuah instansi dapat memperlebar akses dengan aman terhadap jaringan internalnya melalui jaringan internet dengan biaya yang relatif lebih murah. Seluruh aplikasi dan data yang penting pada jaringan tersebut, dapat diakses oleh pihak tertentu saja yang diberikan wewenang tanpa memperhatikan jarak dan tempat dimana diaksesnya. Oleh karena itu dalam penelitian ini akan diimplementasikan dan dianalisis sistem *Virtual Private Network* antara *OpenVPN* dan PPTP, dengan pengujian performa yaitu *packetloss, roundtrip* dan *winSCP transfer*, sedangkan untuk keamanan adalah pengujian *denial of service dan sniffing.*

LANDASAN TEORI

A. JaringanKomputer

Sebuah sistem yang terdiri dari berbagai komputer yang didesain untuk dapat mengakses informasi dengan sumberdaya yang ada, dapat dikatakan sebagai sebuah jaringan komputer. Jaringan komputer itu sendiri memiliki tujuan yaitu untuk mengirim data atau informasi dari pengirim kepada penerima secara cepat dan akurat. Jaringan komputer dapat diklasifikasikan berdasarkan skala dari jangkauannya menjadi *Local Area Network* (LAN), *Metropolitan Area Network* (MAN), dan *Wide Area Network* (WAN)Purbo, Onno W. (2015).

B. Internet

Internet adalah kumpulan dari jaringan-jaringan komputer di seluruh dunia yang terhubung dengan menggunakan *Transmission Control Protocol/Internet Protocol* (TCP/IP). TCP/IP berfungsi sebagai protokol dalam pertukaran data dan berbagi informasi dari seluruh dunia. Hal terpenting yang dibutuh dari TCP/IP adalah *Internet Protocol* (IP),

untuk menyediakan layanan pengiriman paket pada jaringan TCP/IP yang dibangun.

C. Virtual Private Network

Virtual Private Network (VPN) adalah sebuah teknologi komunikasi yang memungkinkan dapat terkoneksi ke jaringan publik dan menggunakannya untuk dapat bergabung dengan jaringan lokal. VPN merupakan koneksi virtual yang bersifat *private*, dikarenakan jaringan yang dibuat tidak nampak secara fisik hanya berupa jaringan virtual, dan jaringan tersebut tidak semua orang dapat mengaksesnya sehingga sifatnya *private*. Dengan cara tersebut maka akan didapatkan hak dan pengaturan yang sama seperti halnya berada di dalam kantor atau LAN itu sendiri, walaupun sebenarnya menggunakan jaringan milik publik.

Ardiyansyah (2008) teknologi VPN menyediakan beberapa fungsi utama untuk penggunaannya. Fungsi-fungsi utama tersebut antara lain sebagai berikut.

1. *Confidentially* (Kerahasiaan)

Dengan digunakannnya jaringan publik yang rawan pencurian data, maka teknologi VPN menggunakan sistem kerja dengan cara mengenkripsi semua data yang lewat melauinya. Dengan adanya teknologi enkripsi tersebut, maka kerahasiaan data dapat lebih terjaga. Walaupun ada pihak yang dapat menyadap data yang melewati internet bahkan jalur VPN itu sendiri, namun belum tentu dapat membaca data tersebut, karena data tersebut telah teracak. Dengan menerapkan sistem enkripsi ini, tidak ada satupun orang yang dapat mengakses dan membaca isi jaringan data dengan mudah.

2. *Data Intergrity* (Keutuhan data)

Ketika melewati jaringan internet, sebenarnya data telah berjalan sangat jauh melintasi berbagai negara. Pada saat perjalanan tersebut, berbagai gangguan dapat terjadi terhadap isinya, baik hilang, rusak, ataupun dimanipulasi oleh orang yang tidak seharusnya. Pada VPN terdapat teknologi yang dapat menjaga keutuhan data mulai dari data dikirim hingga data sampai di tempat tujuan.

3. *Origin Authentication* (Autentikasi sumber)

Teknologi VPN memiliki kemampuan untuk melakukan autentikasi terhadap sumber-sumber pengirim data yang akan diterimanya. VPN akan melakukan pemeriksaan terhadap semua data yang masuk dan mengambil informasi dari sumber datanya. Kemudian, alamat sumber data tersebut akan disetujui apabila proses autentikasinya berhasil. Dengan demikian, VPN menjamin semua data yang dikirim dan diterima berasal dari sumber yang seharusnya. Tidak ada data yang dipalsukan atau dikirim oleh pihak-pihak lain.

4. *Non-repudiation*

Yaitu mencegah dua pihak dari menyangkal bahwa mereka telah mengirim atau menerima sebuah *file* mengakomodasi perubahan.

5. Kendali akses

Menentukan siapa yang diberikan akses ke sebuah sistem atau jaringan, sebagaimana informasi apa dan seberapa banyak seseorang dapat menerima.

Konsep VPN

Jaringan VPN menawarkan keamanan dan tidak terdeteksi dikarenakan IP yang digunakan adalah IP Public milik VPN *server*. Dengan adanya enkripsi dan dekripsi maka data yang lewat jaringan internet ini tidak dapat diakses oleh orang lain bahkan oleh *client* lain yang terhubung dengan *server* VPN. Kunci yang dibutuhkan untuk membuka enkripsi tersebut hanya diketahui oleh *server* VPN dan *client* yang terhubung dengannya. Dengan penggunaan enkripsi dan dekripsi itulah yang menyebabkan data yang lewat jaringan tidak dapat dimodifikasi dan dibaca sehingga keamanannya terjaminHendriana, Yana. (2012).

Teknologi *Tunneling*

Teknologi *tunneling* merupakan teknologi yang bertugas untuk manangani dan menyediakan koneksi *point-to-point* dari sumber ke tujuannya. Disebut *tunnel* (terowongan) karena koneksi *point-to-point* tersebut sebenarnya terbentuk dengan melintasi jaringan umum, namun koneksi tersebut tidak mempedulikan paket-paket data milik orang lain yang sama-sama melintasi jaringan umum tersebut, tetapi koneksi tersebut hanya melayani transportasi data dari pembuatnya. Hal ini sama dengan penggunaan jalur *busway* yang pada dasarnya menggunakan jalan raya, tetapi dia membuat jalur sendiri untuk dapat dilalui bus khusus. Koneksi *point-to-point* ini sesungguhnya tidak benar-benar ada, namun data yang dihantarkannya terlihat seperti benar-benar melewati koneksi pribadi yang bersifat *point-to-point*.

Teknologi *Enkripsi*

Teknologi enkripsi menjamin data yang melalui *tunnel* tidak dapat dibaca dengan mudah oleh orang lain yang bukan merupakan komputer tujuannya. Semakin banyak data yang lewat di dalam *tunnel* yang terbuka di jaringan publik, maka teknologi enkripsi ini semakin dibutuhkan. Enkripsi akan mengubah informasi yang ada dalam *tunnel* tersebut menjadi sebuah *ciphertext* (teks yang dikacaukan dan tidak ada artinya sama sekali apabila dibaca secara langsung). Untuk dapat membuatnya kembali memiliki arti atau dapat dibaca, maka dibutuhkan proses dekripsi. Proses dekripsi terjadi pada ujung dari hubungan *VPN*. Pada kedua ujung ini telah menyepakati sebuah algoritma yang akan digunakan untuk melakukan proses enkripsi dan dekripsinya. Dengan demikian, data yang dikirim aman sampai tempat tujuan, karena orang lain di luar *tunnel* tidak memiliki algoritma untuk membuka data tersebut.

Protokol dan Teknologi VPN

1. *Point-to-Point Tunneling Protocol* (PPTP)

PPTP merupakan protokol jaringan yang dikembangkan oleh Microsoft dan Cisco yang memungkinkan pengamanan transfer data dari *remoteclient* ke *server* pribadi instansi dengan membuat sebuah VPN melalui TCP/IP. Teknologi jaringan yang terdapat pada PPTP adalah pengembangan dari *remote access Point-to-Point Protocol* yang dikeluarkan oleh *Internet Engineering Task Force* (IETF). PPTP membungkus paket PPP menjadi IP *datagrams* agar dapat ditransmisikan melalui internet atau jaringan publik berbasis TCP/IP. PPTP juga dapat digunakan pada jaringan *privateLAN-to-LAN*.

2. *Layer* 2 *Tunneling* Protocol (L2TP)

L2TP merupakan *tunnelingprotocol* yang memadukan dua buah tunneling protokol yaitu *Layer* 2 *Forwarding* milik Cisco dan PPTP yang dimiliki Microsoft. L2TP umumnya digunakan untuk membuat Virtual Private Dial Network (VPDN) yang dapat membawa semua jenis protokol komunikasi di dalamnya dan biasanya menggunakan port 1702 dengan protokol UDP. Terdapat dua model tunnel yang dikenal, yaitu *compulsory* dan *voluntary*. Perbedaan utama keduanya terletak pada *endpoint tunnel*-nya. Pada *compulsorytunnel*, ujung *tunnel* berada pada ISP, sedangkan pada *voluntary* ujung *tunnel* berada pada *clientremote*.

3. *Internet Protocol Security* (IPsec)

IPSec adalah pengembangan dari protokol IP yang bertujuan untuk menyediakan keamanan pada suatu IP dan *layer* yang berada diatasnya. Pada dasarnya paket IP tidak memiliki keamanan, sehingga tidak ada jaminan bahwa paket yang diterima sama dengan paket ketika ditransmisikan oleh si pengirim paket. Paket IP yang tidak memiliki keamanan atau *security*, sangat mudah untuk diketahui isinya dan alamat IP itu sendiri. IPsec adalah metode yang bertujuan untuk menjaga keamanan IP datagram ketika paket diransmisikan pada *traffic*. Sehingga IPsec menjadi suatu mekanisme yang diimplementasikan pada VPN. *IPSec* berada pada *layer* tiga OSI yaitu *network layer* sehingga dapat mengamankan data dari *layer* yang berada atasnya. IPSec terdiri dari dua buah *security* protokol :

- *AH* (*Authentication Header*) melakukan autentikasi datagram untuk mengidentifikasi pengirim data tersebut.
- *ESP* (*Encapsulating Security Header*) melakukan enkripsi dan layanan autentifikasi.

Dua buah protokol tersebut dapat dikombinasikan atau berdiri sendiri dalam penyediaan kemanan. IPSec menggunakan dua buah protokol berbeda untuk menyediakan pengamanan data yaitu *AH* dan *ESP* keduanya dapat dikombinasikan ataupun berdiri sendiri. Dengan menggunakan *IPSec* maka suatu sistem dapat memilih protokol security apa yang akan digunakan, dikarenakan IPsec berada pada level IP.

4. *Secure Socket Layer* (SSL)

SSL merupakan suatu standar teknologi keamanan untuk menjamin data yang melalui *webserver* dan *webbrowser* dengan membuat koneksi yang dienkripsi, antara *server* atau situs dengan pengunjungnya. Tanpa SSL data akan mudah dilihat dan dirubah saat dikirim melalui internet. SSL bertindak sebagai protokol yang mengamankan komunikasi antara *client* dan *server*.

Protokol SSL mengotentikasi *server* kepada client menggunakan kriptografi kunci publik dan sertifikat digital. Protokol ini juga menyediakan otentikasi *client* ke *server*. Algoritma kunci publik yang digunakan adalah RSA, dan untuk algoritma kunci rahasia yang digunakan adalah IDEA, DES, 3DES, AES.

D. *Open*VPN

Sukaridhoto, Stritrusta. (2007) menyebutkan *Open*VPN adalah sebuah solusi VPN yang antar *platform* , aman dan

sangat mudah dikonfigurasikan dengan menggunakan antar muka virtual yang disediakan oleh *driver* jaringan universal TUN / TAP dan dijalankan sepenuhnya dengan pengguna yang merupakan perlindungan khusus pada sistem.

E. WinSCP

WinSCP adalah aplikasi open source untuk *client* Windows yang berfungsi untuk *transfer file* antara *Windows* dengan *Linux.WinSCP* mampu melakukan *transfer file* melalui protokol FTP, SFTP dan SCP.

F. *Sniffing*

Sniffing adalah aktivitas menyadap paket data yang sedang berjalan pada *traffic* sebuah jaringan. Paket data ini bisa berisi informasi mengenai apa saja, baik itu username, apa yang dilakukan pengguna melalui jaringan, termasuk mengidentifikasi komputer yang terinfeksi virus, sekaligus melihat apa yang membuat komputer menjadi lambat dalam jaringan. *Sniffing* juga dapat mengidentifikasi penyebab macet pada jaringan.

G. WireShark

WireShark adalah *network protocol analyzer* terkemuka di dunia. Ini memungkinkan pengguna dapat melihat apa yang sedang terjadi pada jaringan pengguna pada level *microscopic*. WireShark sudah diakui secara standar de facto dan de jure di banyak industri dan lembaga pendidikan. Aplikasi ini menangkap paket-paket jaringan dan berusaha untuk menampilkan semua informasi di paket tersebut sedetail mungkin. Hal ini dapat diibaratkan sebagai alat untuk memeriksa apa yang sebenarnya sedang terjadi di dalam jaringan.

H. DoS PingFlood

DoS PingFlood adalah aplikasi yang dioperasikan pada sistem operasi Windows. Pingflood menggunakan metode *ping* dan menggunakan protokol ICMP, tetapi dalam paket dengan jumlah yang sangat banyak serta dengan sangat cepat pengirimannya. Penggunaan pingflood harus pada OS Windows karena aplikasi ini berekstensi .exe. Korban dari pingflood biasanya adalah modem, Windows, Linux, router dan server. Semua sistem operasi dan mesin dijaringan komputer yang memiliki IP *address* bisa diserang tanpa terkecuali. Efek dari pingflood ini, yaitu akftivitas komputer korban yang meningkat serta *traffic* jaringan yang penuh Nandareynaldi. (2015).

METODE PENELITIAN

Perangkat yang digunakan berupa *server* yang terhubung langsung dengan *internet*, untuk membangun jaringan menggunakan *Virtual Private Network* maka *server* tersebut akan di-*install* PPTP dan OpenVPN. VPN tersebut terdapat pada VM (*virtualmachine*) yang terletak di dalam server, kemudian *server* itu sendiri terhubung dengan *routerboard* menggunakan metode NAT. Setelah itu *routerboard* akan terhubung dengan *routercloud* dengan media transmisi *wired* untuk dapat terhubung dengan *internet*.

Di dalam *server* terdapat VM (*virtualmachine*) yang digunakan sebagai VPN *server*. Sistem operasi yang digunakan pada VM tersebut adalah Ubuntu Server 14.04. Ubuntu termasuk sistem operasi yang *opensource*, stabil dan memiliki paket-paket *software server* yang cukup lengkap.

Untuk mengimplementasikan VPN pada jaringan *enterprise*, salah satu VM

pada *server* dipasangkan atau di-*install* PPTP. VPN menggunakan PPTP, mudah untuk digunakan dan sederhana. PPTP akan membuat terowongan atau *tunnel* dengan menggunakan *port* 1723.

Salah satu VM yang lain akan dipasang OpenVPN untuk membuat jaringan VPN. OpenVPN memiliki fungsi yang sama dengan PPTP, yaitu untuk membuat *tunnel* pada jaringan. OpenVPN bersifat *opensource* sama halnya dengan PPTP. OpenVPN dapat berjalan pada UDP (*User Datagram Protocol*) atau TCP (*Transmission Control Protocol*), dan menggunakan *port* 1194 untuk dapat terhubung dengan *client*.

Model Penelitian

Model penelitian menggunakan elaborasi Hardware dan software.

Perangkat yang digunakan berupa *server* yang terhubung langsung dengan *internet*, untuk membangun jaringan menggunakan *Virtual Private Network* maka *server* tersebut akan di-*install* PPTP dan OpenVPN. VPN tersebut terdapat pada VM (*virtualmachine*) yang terletak di dalam server, kemudian *server* itu sendiri terhubung dengan *routerboard* JTIK menggunakan metode NAT. Setelah itu *routerboard* JTIK akan terhubung dengan *routercloud* yang dimiliki oleh PUSDATIN dengan media transmisi *wired* untuk dapat terhubung dengan *internet*.

Di dalam *server* terdapat VM (*virtualmachine*) yang digunakan sebagai VPN *server*. Sistem operasi yang digunakan pada VM tersebut adalah Ubuntu Server 14.04. Ubuntu termasuk sistem operasi yang *opensource*, stabil dan memiliki paket-paket *softwareserver* yang cukup lengkap.

Untuk mengimplementasikan VPN pada jaringan *enterprise* jurusan Teknik Informatika dan Komputer, salah satu VM pada *server* dipasangkan atau di-*install* PPTP. VPN menggunakan PPTP, mudah untuk digunakan dan sederhana. PPTP akan membuat terowongan atau *tunnel* dengan menggunakan *port* 1723.

Salah satu VM yang lain akan dipasang OpenVPN untuk membuat jaringan VPN. OpenVPN memiliki fungsi yang sama dengan PPTP, yaitu untuk membuat *tunnel* pada jaringan. OpenVPN bersifat *opensource* sama halnya dengan PPTP. OpenVPN dapat berjalan pada UDP (*UserDatagramProtocol*) atau TCP (*TransmissionControlProtocol*), dan menggunakan *port* 1194 untuk dapat terhubung dengan *client*. Selain itu juga menggunakan WinSCP, OpenSSH, dan Putty.

Skema Alat

Gambar1 DiagramBlokVPN

Client yang akan mencoba terhubung melalui koneksi VPN *dengan* menggunakan VPN *client*. VPN client menggunakan media internet sebagai *tunnel* atau jaringan *private*. Kemudian diterima oleh VPN *server* yang terdapat di dalam *server*. Setelah itu VPN *client* dapat mengakses jaringan yang berada di belakang VPN *server*.

Kerja Aplikasi

Setelah VPN diimplementasikan pada *server*, dengan begitu *server* dapat diakses dari luar oleh *client*. Cara kerja untuk VPN *client* yang menggunakan PPTP adalah dengan mengontak VPN *server* untuk memverifikasi *username* dan

password melalui *internet*. Setelah verifikasi berhasil maka VPN *server* akan memberikan IP *address* baru untuk *client* sehingga terbentuklah sebuah koneksi atau biasa disebut *tunnel*. Dengan terhubungnya VPN *client* dengan VPN *server* makatej *client* dapat mengakses *resource* yang berada di belakang VPN *server* seperti mengambil data, memodifikasi data, mengirim data dll.

Pada VPN *client* yang menggunakan OpenVPN, ada beberapa hal yang harus dilakukan terlebih dahulu. *Client* harus memiliki *fileconfig* dari OpenVPN *client*, untuk dapat mengakses VPN *server*. Berbeda halnya dengan PPTP yang menggunakan *username* dan *password*, OpenVPN harus memverifikasi *fileconfig* pada *client* yang berisi sertifikat dan *key* yang sesuai dengan VPN *server*. Setelah berhasil barulah *client* mendapat IP *address* baru, selanjutnya *client* dapat mengakses jaringan atau *resource* yang berada dibelakang VPN *server*.

Metode Analisis Data

Penelitian ini menggunakan pengujianpada konektivitas VPN. Pengujian akan dilakukan melalui komputer *client* terhadap jaringan VPN. Pengujian meliputi performa yaitu *packet loss, round trip,* winSCP transfer, dan keamanan yaitu *denial of service, dan sniffing* jaringan VPN dengan wireshark

HASIL PENELITIAN DAN PEMBAHASAN

Instalasi PPTP

Pada proses intalasi PPTP dilakukan pada VM 1, proses instalasi dimulai dengan melakukan *login* terlebih dahulu sebagai *root*. Setelah itu dilanjutkan dengan meng-install paket PPTP *server*.apt-get install pptpd. Setelah proses instalasi selesai dilanjutkan dengan merubah *file* pptpd.conf.nano /etc/pptpd.conf

Pada *file* pptpd.conf lakukan perubahan pada bagian bawah *file* dengan memasukkan perintah sebagai berikut:

localip 192.168.0.1

remoteip 192.168.0.100-200

Perubahan yang dilakukan adalah untuk menambahkan IP untuk *server* dan IP untuk *client* dari PPTP. Setelah itu keluar dan simpan perubahan dengan menekan kombinasi tombol "Ctrl+X" kemudian tekan tombol "Y" untuk menyimpan perubahan, lalu tekan tombol "Enter" untuk konfirmasi. Selanjutnya adalah melakukan perubahan pada *file* pptpd-options untuk menambahkan DNS.

nano /etc/ppp/pptpd-options

Pada *file* pptpd-options hilangkan *comment* atau tanda pagar pada baris ms-dns dan rubah menjadi DNS google atau DNS lainnya, menjadi seperti berikut:

ms-dns 8.8.8.8

ms-dns 8.8.4.4

Langkah selanjutnya adalah menambahkan *user* untuk VPN PPTP, dengan cara menambahkan *username* dan *password* pada *file* chap-secrets.

nano /etc/ppp/chap-secrets

Setelah *file* chap-secrets terbuka penulis menambahkan akun untuk VPN *user* pada baris terakhir. Penulis menambahkan akun tersebut *username* dan *password* secara berurutan seperti berikut:

pptpjtik *

vpnpptpjtik2015 *

Konfigurasi yang telah dilakukan belum langsung diterapkan, untuk menerapkannya dengan cara memasukkan perintah "/etc/init.d/pptpd *restart*". VPN PPTP sudah dapat mengakses sebatas pada sisi *server* untuk dapat mengakses sisi luar

server adalah dengan mengaktifkan IP *Forwarding*. Hal yang harus dilakukan adalah merubah isi *file* sysctl.conf.

nano /etc/sysctl.conf

Pada *file* sysctl.conf cari baris "net.ipv4.ip_forward=0", lalu hilangkan *comment* dan rubah parameternya dari 0 (*disabled*) menjadi 1 (*enabled*). Setelah itu masukkan perintah "sysctl -p" untuk menerapkan perubahan. VPN PPTP sudah siap untuk digunakan, langkah selanjutnya adalah konfigurasi dari sisi *client*.

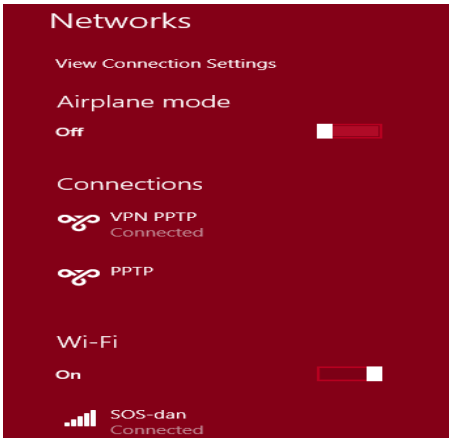
Konfigurasi *Client* PPTP

Setelah PPTP *server* telah dibuat, maka dilanjutkan konfigurasi pada sisi *client* agar dapat terhubung dengan VPN PPTP, sebelum itu pastikan komputer *client* terhubung dengan jaringan *internet*. Berikut adalah langkah-langkah untuk konfigurasi *client* pada Windows:

1. Pada *Network and Sharing Center* dilanjutkan dengan memilih *Set up new connection or network*
2. Memilih *connect to a workplace* untuk membuat VPN baru
3. Kemudian memilih *create a new connection*, lalu menekan tombol Next.
4. Setelah itu memilih *Use my internet connection* (VPN)
5. Selanjutnya mengisikan IP *address* 103.36.14.227 dan berikan nama pada jaringan yang dibuat.
6. Setelah koneksi terbuat dapat dilihat pada adapter jaringan, kemudian atur *security* dari adapter VPN yang telah dibuat menjadi PPTP
7. Untuk dapat masuk ke dalam jaringan VPN PPTP diperlukan otentikasi berupa *username* dan *password* yang telah diatur sebelumnya pada PPTP server, kemudian menekan tombol OK. Jika otentikasi berhasil maka akan tampak tulisan *connected* pada status koneksi seperti terlihat pada gambar 2 yang menandakan VPN siap untuk digunakan.

8. 

9. Gambar**2***Sign-in* VPN

10. 

11. **Gambar3** VPN PPTP *Connected*

Instalasi OpenVPN

Proses instalasi dapat dimulai dengan *installOpen*VPN dan *Easy*-RSA.

apt-*get installopen*vpn *easy*-rsa

Setelah proses *install* selesai dan seluruh *file* dari OpenVPN sudah tersimpan, hal yang dilakukan selanjutnya adalah mengekstrak *filesample* konfigurasi:

gunzip -c /usr/share/doc/openvpn/examples/sample-config-*files*/server.conf.gz > /etc/openvpn/server.conf

File yang akan dikonfigurasi adalah *server*.conf menggunakan teks editor "nano".

nano /etc/openvpn/server.conf

Pada *file* server.conf ada beberapa hal yang akan dilakukan perubahan. Hal pertama yang akan dirubah terdapat pada bagian "Diffie hellman parameters", dari parameter 1024 akan digandakan menjadi 2048, sehingga menjadi seperti berikut:

dh dh2048.pem

Perubahan tersebut akan menambah panjang RSA key menjadi *double* dari sebelumnya, saat *generatekey* untuk *server* dan *client*. Perubahan lain yang dilakukan adalah pada baris berikut:

;push "redirect-gateway def1 bypass-dhcp"

Pada baris tersebut "*uncomment*" atau hilangkan tanda *semicolon* (;) di depan baris tersebut untuk mengaktifkannya. Selanjutnya adalah menghilangkan *comment* pada dua baris berikut:

;push "dhcp-option DNS 208.67.222.222"

;push "dhcp-option DNS 208.67.220.220"

Bagian terakhir untuk dirubah adalah pada dua baris berikut dengan menghilangkan *comment*:

;*user nobody*

;group nogroup

Setelah itu simpan perubahan dengan menekan kombinasi tombol "Ctrl+X" kemudian tekan tombol "Y" untuk menyimpan perubahan, lalu tekan tombol "Enter" untuk konfirmasi.

Untuk mengaktifkan paket *forwarding* sama dengan yang ada pada PPTP, menjadi seperti berikut:

net.ipv4.ip_*forward*=1

Tahap selanjutnya adalah konfigurasi pada *UncomplicatedFirewall* (ufw), ufw adalah sebuah *front-end* untuk *iptables*. Untuk konfigurasinya tidaklah sulit. Hal yang pertama dilakukan adalah memasukkan perintah untuk mengizinkan ssh pada ufw, seperti berikut:

ufw *allow* ssh

Setelah itu adalah mengizinkan port *Open*VPN yaitu 1194 dan dapat berjalan di atas protokol TCP.

ufw *allow* 1194/tcp

Kebijakan *forwarding* pada ufw perlu diatur juga. Penulis akan melakukan konfigurasi pada *file* konfigurasi utama ufw.

nano /etc/default/ufw

Di dalam *file* ufw lihat pada baris "*DEFAULT_FORWARD_POLICY="DROP*"". Baris ini harus dirubah pada bagian **DROP** menjadi **ACCEPT**. Setelah selesai akan terlihat seperti ini:

DEFAULT_FORWARD_POLICY="ACCEPT"

Kemudian simpan dan keluar. Selanjutnya penulis akan menambahkan aturan tambahan pada ufw untuk menerjemahkan alamat dan menyamarkan IP dari *client* yang terhubung.

nano /etc/ufw/before.rules

Pada bagian atas dari *before.rules*, akan tampak seperti dibawah ini, bagian yang dicetak merah adalah bagian yang ditambahkan:

 # ufw-before-output

 # ufw-before-forward

#

 # START OPENVPN RULES

 # NAT table rules

**nat*

:POSTROUTING ACCEPT [0:0]

 # Allow traffic from OpenVPN client to eth0

-A POSTROUTING -s 10.8.0.0/8 -o eth0 -j MASQUERADECOMMIT

 # END OPENVPN RULES

 # Don't delete these required lines, otherwise there will be errors

**filter*

Setelah penambahan aturan pada ufw, selanjutnya adalah mengaktifkan ufw tersebut.

ufw enable

Dengan mengaktifkan ufw makan akan disertai peringatan sebagai berikut:

Command may disrupt existing ssh connections. Proceed with operation (y|n)?

Kemudian dilanjutkan dengan menekan tombol "y", maka akan tampil keluaran seperti berikut:*Firewall is active and enabled on system startup*.

Setelah selesai konfigurasi untuk meilihat status dan aturan yang sudah dimasukkan pada *firewall* dapat dengan memasukkan perintah "*ufw status*.

Langkah selanjutnya adalah konfigurasi dan membuat *Certificate Authority* (CA). Setiap *client* VPN akan mengotentikasi *servercertificate* dan *server* harus mengotentikasi *clientcertificate* sebelum terhubung. OpenVPN menggunakan Easy RSA sebagai *generatecertificate*. Hal pertama yang harus dilakukan adalah menyalin easy-rsa *generationscripts*.

cp -r /usr/share/easy-rsa/ /etc/openvpn

Lalu membuat direktori untuk penyimpanan *key*.

mkdir /etc/openvpn/easy-rsa/keys

Easy-RSA mempunyai variabel *file*, kita dapat merubah untuk membuat sertifikat ekslusif kepada orang-orang kita, bisnis, atau entitas apa pun yang kita pilih. Informasi ini akan disalin ke *certificates* dan *keys*, dan akan membantu mengidentifikasi *keys* dikemudian.

vim /etc/openvpn/easy-rsa/vars

Variabel dibawah yang dicetak merah disesuaikan dengan kebutuhan penulis.

export KEY_COUNTRY="ID"
export KEY_PROVINCE="JKT"
export KEY_CITY="Jakarta"
export KEY_ORG="PNJ"
export KEY_EMAIL=" "
export KEY_OU="JTIK"

Selanjutnya adalah menentukan nama *server*, jika merubah nama *server* maka perlu memperbarui *file* konfigurasi *Open*VPN yang merujuk ke *server.key* dan *server.crt*.

export KEY_NAME="openvpn"

Tahap selanjutnya adalah *generate* Diffie-Hellman parameter, hal ini akan membutuhkan waktu beberapa menit.

openssl dhparam -out /etc/openvpn/dh2048.pem 2048

Langkah selanjutnya adalah melakukan konfigurasi pada direktori easy-rsa, untuk melanjutkan *file* yang sebelumnya telah dipindahkan.

cd /etc/openvpn/easy-rsa

Pada direktori easy-rsa hal yang pertama yang dilakukan adalah menginisialisasi PKI (*Public Key Infrastructure*), dengan memasukkan perintah sebagai berikut:

. ./vars

Dari perintah diatas menghasilkan keluaran berupa peringatan, berhubung belum ada yang di-*generate* maka tidak perlu diperhatikan. Selanjutnya adalah membersihkan direktori kerja dari kemungkinan *file* lama.

./clean-all

Perintah selanjutnya adalah membangun *Certificate Authority* (CA) dengan menerapkan perintah OpenSSL interaktif. Output-nya akan meminta konfirmasi variabel yang sebelumnya telah dimasukkan kedalam variabel Easy-RSA (*countryname*, *organization*, etc.).

./build-ca

Saat proses dimulai cukup menekan tombol "Enter" untuk melewati setiap *prompt*. Jika terdapat perubahan

maka dapat langsung memasukkannya dari dalam *prompt*.

Masih di dalam /etc/openvpn/easy-rsa, sekarang masukkan perintah untuk membangun *serverkey*. Langkah selanjutnya akan menggunakan variabel pada Easy-RSA.

./build-key-server openvpn

Keluaran dari perintah tersebut mirip dengan perintah *./build-ca*, tetapi dengan dua tambahan yaitu:

Please enter the following 'extra' attributes

to be sent with your certificate request

A challenge password []:

An optional company name []:

Keduanya harus dibiarkan kosong, jadi dilanjutkan hanya dengan menekan tombol "Enter" untuk melewati masing-masing. Kemudian dilanjutkan dengan pertanyaan tambahan dan memerlukan respon positif dengan menekan tombol "y".

Sign the certificate? [y/n]

1 out of 1 certificate requests certified, commit? [y/n]

Selanjutnya akan menampilkan keluaran ketika selesai seperti berikut:

Write out database with 1 new entries

Data Base Updated

Tahap selanjutnya adalah memindahkan *servercertificates* dan *keys*. Hal yang harus dilakukan adalah menyalin ke lokasi yang tepat.

cp/etc/openvpn/easy-rsa/keys/{server.crt,server.key,ca.crt} /etc/openvpn

Pada poin ini, OpenVPN *server* siap untuk digunakan dan dapat dilihat statusnya dengan dua perintah berikut:

service openvpn start

service openvpn status

Pada perintah status seharusnya menghasilkan keluaran seperti berikut:

VPN 'server' is running

Hal ini menandakan OpenVPN server telah berjalan dan siap untuk digunakan. OpenVPN dapat digunakan jika *user* atau *client* juga harus mempunyai CA. Tetap di dalam direktori /etc/openvpn/easy-rsa, CA untuk *client* juga dibuat di dalamnya.

./build-key client1

Bagian berwarna merah dapat disesuaikan dengan nama yang lain. Proses pembuatannya mirip dengan pembuatan *serverkey*, maka hal yang sama juga dilakukan pada pembuatan *client*. Proses ini akan menghasilkan tiga *file* dengan ekstensi .crt, .csr, dan .key. *File* yang dibutuhkan adalah *file* .crt dan .key saja, untuk membuat *fileclient*. Hal tersebut dapat dilakukan berkali-kali sesuai dengan jumlah *client* yang dibutuhkan. OpenVPN telah menyediakan sampel *fileclient*, untuk dapat digunakan *file* tersebut akan dirubah dari .conf menjadi .ovpn, karena *fileclient* berekstensi .ovpn.

cp/usr/share/doc/openvpn/examples/sample-config-files/client.conf /etc/openvpn/easy-rsa/keys/client.ovpn

Selanjutnya adalah mengkonfigurasi semua *file* tersebut yang telah dibuat sebelumnya yaitu client1.crt, client1.key, client.ovpn dan ca.crt. *File* tersebut akan diolah di luar dari OpenVPN server, untuk kita dapat menggunakan *file* transfer untuk mengunduh *file-file* tersebut. *File* transfer yang digunakan adalah WinSCP, berikut adalah letak *file* dengan direktorinya:

/etc/openvpn/easy-rsa/keys/client1.crt

/etc/openvpn/easy-rsa/keys/client1.key

/etc/openvpn/easy-rsa/keys/client.ovpn

/etc/openvpn/ca.crt

Setelah *file-file* tersebut diunduh, langkah selanjutnya adalah menyatukan *file-file* tersebut kedalam client.ovpn. Client.ovpn akan dapat dirubah menggunakan *text editor* apa saja. Hal-hal yang harus diperhatikan ketika di dalam client.ovpn, yang pertama adalah memasukkan IP *server* JTIK pada baris berikut:

remote my-server-1 1194

Menjadi seperti berikut

remote 103.36.14.227 1194

Kemudian hal lain yang harus diperhatikan adalah menghilangkan *comment* pada dua baris berikut:

user nobody

group nogroup

Hal terakhir adalah memastikan bahwa tiga baris berikut ini dalam keadaan comment, karena kita akan menyatukan *file-file* tersebut.

 # SSL/TLS parms.

 # . . .

#ca ca.crt

#cert client.crt

#key client.key

Terakhir adalah menyatukan *file-file* ca.crt, *client*1.crt, dan *client*1.key, pada bagian akhir *client*.ovpn, dengan menggunakan basic XML, dengan bentuk seperti berikut:

<ca>
(isi *file* ca.crt)
</ca>
<cert>
(isi *fileclient*.crt)
</cert>
<key>
(isi *fileclient*.key)
</key>

Setelah memasukkan isi *file-file* tersebut kedalam bagian-bagian yang telah disediakan, selanjutnya adalah menyimpan *file* tersebut, maka *fileclient*.ovpn sudah dapat digunakan. Hal ini dapat dilakukan pada *client* lain jika ingin membuat *client* lain, hal yang perlu diganti adalah pada isi bagian *client*.crt dan *client.key* saja, selebihnya sama untuk setiap *client*.

Konfigurasi *ClientOpen*VPN

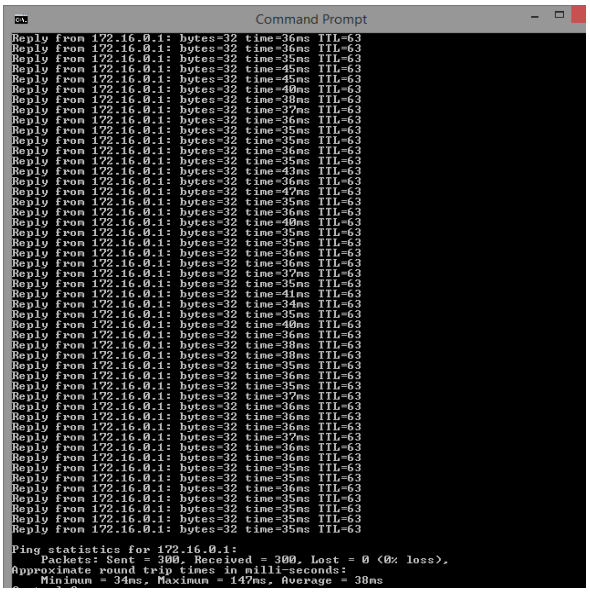
Pada konfigurasi *client* OpenVPN yang dibutuhkan adalah *file client* dari tiga *file* yang telah disatukan sebelumnya dan aplikasi tambahan yaitu *Open*VPN GUI. Setelah aplikasi tersebut diinstall,langkah selanjutnya adalah menaruh *file client* pada direktori instalasi *Open*VPN GUI pada *folder config*, setelah itu *Open*VPN dapat digunakan.

Pengujian

Prosedur pengujian akan menjelaskan konektivitas VPN, yang dilakukan pada jaringan *internet* menggunakan *provider* Telkom dengan kecepatan *download* 2.57 Mbps dan *upload* 0.60 Mbps. Pengujian performa meliputi*PacketLoss*, *RoundTrip* dan WinSCP Transfer, sedangkan untuk keamanan adalah pengujian *Denial of Service* dan *Sniffing.*

1. *PacketLoss*

Pada pengujian ini bertujuan untuk memantau rata-rata, minimum dan maksimum *packetloss* yang melalui *tunnel* VPN. Pengujian dilakukan dengan cara *ping* tanpa beban dalam waktu lima menit sebanyak tiga kali menggunakan *Command Prompt* Windows (cmd). Hal ini dilakukan pada kedua VPN dengan destinasi IP pada *gateway server* 172.16.0.1, terlihatpada gambar 4.



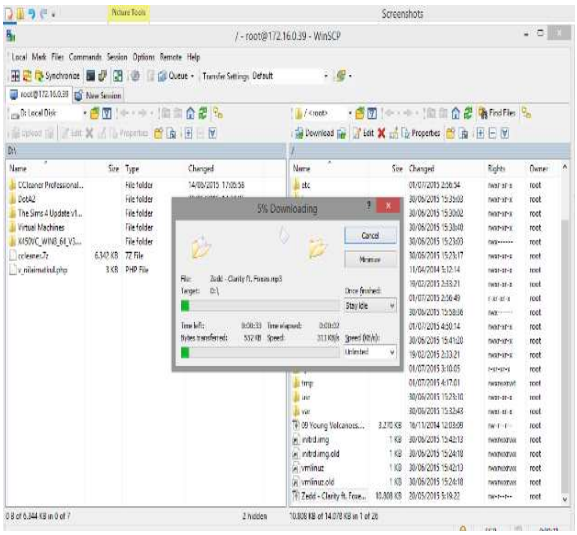
Gambar 4 *Ping* OpenVPN

2. *Round Trip*

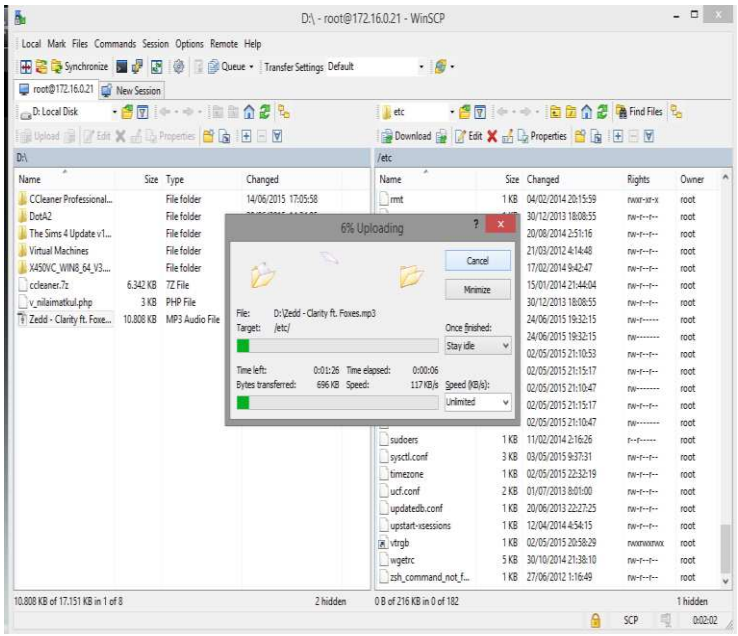
Pengujian *Round Trip* bertujuan menghitung rata-rata dan maksimum waktu *round trip* pada *tunnel* VPN. Sebenernya waktu *round trip* terdapat pada hasil *ping*, sehingga pengujian ini dilakukan bersamaan dengan *packet loss*. Round trip adalah perjalanan paket *ping* dari komputer yang digunakan untuk melakukan *ping*, kemudian ke IP *gateway* kembali lagi ke komputer *client*, atau dapat dikatakan pulang pergi.

3. WinSCP Transfer

WinSCP adalah *tools* atau aplikasi *file* transfer untuk Linux yang berfungsi memindahkan atau merubah data baik itu dari *client* ke *host* atau sebaliknya. Pengujian ini bertujuan untuk mengetahui waktu yang dibutuhkan untuk transfer *file* melalui tunnel VPN, baik itu *download* maupun *upload* dan dilakukan pada setiap koneksi VPN. Pada saat transfer *file* menggunakan sebuah *file* yang berukuran 10.808 KB, *file* tersebut akan dipindah-pindahkan menggunakan WinSCP.



Gambar 5*DownloadFile* (PPTP)



Gambar 6*Upload File* (OpenVPN)

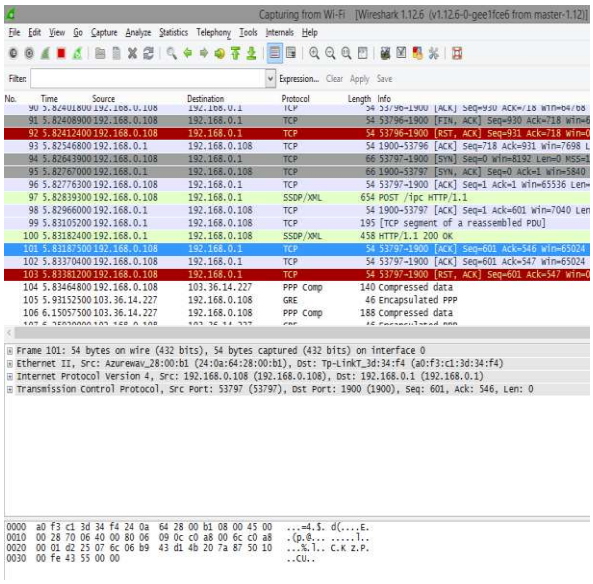
4. *Denial of Service*

Denial of Service adalah pengujian untuk melakukan *attack* pada VPN *server*. Pengujian ini bertujuan menghentikan atau mematikan *service* VPN pada VPN *server*. Prosedur pengujiannya menggunakan aplikasi pingflood.exe dengan IP tujuan masing-masing VPN *server* dan dilakukan pada *Command Prom*pt Windows (cmd). Format pengujiannya "pingflood ip_vpn_server –s 65000 –n 100000". Untuk mengetahui apakah sudah VPN terganggu atau tidaknya dilakukan secara

bersamaan *ping* 172.16.0.1, jika sudah terputus atau berhenti maka ping akan menampilkan *Request Timed Out* (RTO). Jika VPN sudah terganggu maka dapat diketahui saat dipaket keberapakah VPN tersebut terbebani.

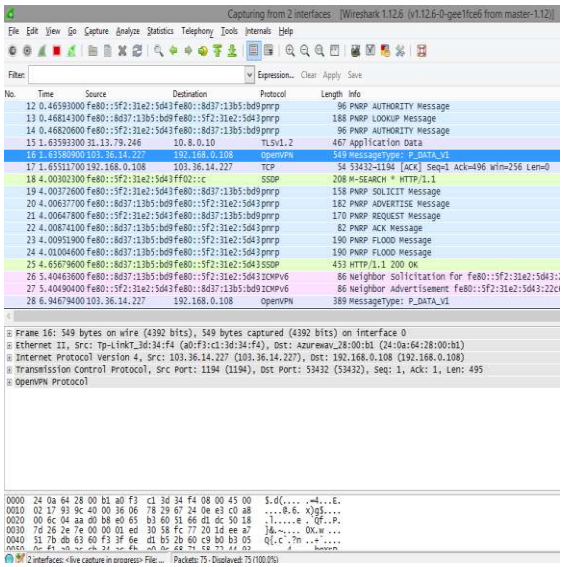
5. *Sniffing* Jaringan VPN

Pada tahap ini mencoba melakukan *sniffing* pada kedua jaringan VPN dari saat mencoba untuk tersambung dengan jaringan hingga tersambung dengan tunnel, menggunakan aplikasi Wireshark.



Gambar 7*Sniffing* PPTP

Gambar 7 adalah gambar proses *Sniffing* pada VPN PPTP saat mencoba hubungan antara *Client* dengan *Server*.



Gambar**8***Sniffing* OpenVPN

Pada gambar 8 adalah proses *sniffing* yang dilakukan pada OpenVPN *Client* yang mencoba untuk terhubung dengan VPN *Server*.

Data Hasil Pengujian

Pengujian yang telah dilakukan menghasilkan data-data yang dapat diperoleh dari setiap VPN.

1. *Packet Loss*

Pada pengujian *packetloss* untuk mengukur rata-rata, minimum dan maksimum *packetloss* saat *ping* pada VPN dapat dilihat pada tabel 1.

Tabel 1 Hasil *Packet Loss*

Jaringan VPN	IP Sumber	IP Tujuan	Min. Packet Loss	Max. Packet Loss	Rata-rata Packet Loss
PPTP	192.168.0.100	172.16.0.1	0	0	0
Open VPN	10.8.0.10	172.16.0.1	0	0	0

2. *RoundTrip*

Pada tabel 2 menunjukkan rata-rata, minimum dan maksimum waktu yang dibutuhkan untuk *roundtrip*.

Tabel 2 Hasil *Round Trip*

Jaringan VPN	IP Sumber	IP Tujuan	Min. Waktu (ms)	Max. Waktu (ms)	Rata-rata Waktu (ms)
PPTP	192.16	172.1	4	45	4
	8.0.100	6.0.1	5	0	1
Open	10.8.0.	172.1	3	45	4
VPN	10	6.0.1	4	8	3

3. WinSCP Transfer

Berikut adalah hasil dari pengujian *download* dan *upload file* pada kedua VPN.

Tabel 3 Hasil Transfer *File*

Jaringan VPN	IP Sumber	IP Tujuan	Waktu Download	Waktu Upload
PPTP	192.168	172.16	00:00:3	00:03:
	.0.100	.0.39	7:47	08:75
Open	10.8.0.	172.16	00:00:3	00:02:
VPN		.0.21	7:66	33:79

4. *Dial of Service*

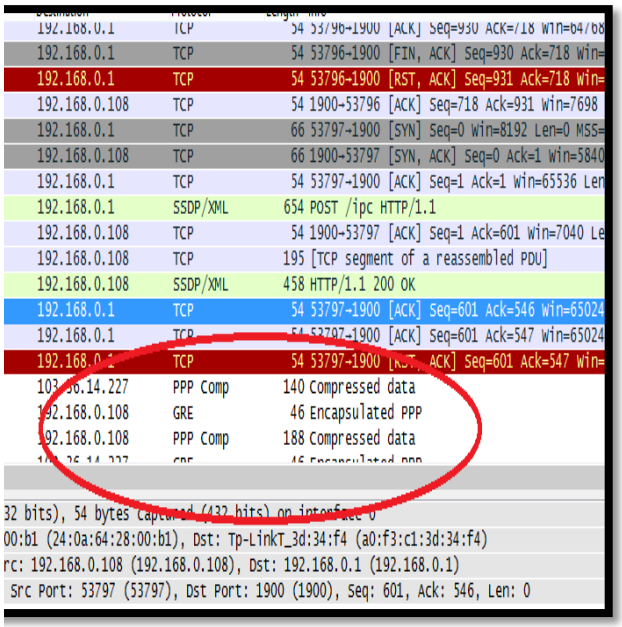
Pengujian *Dial of Service* menggunakan *size* 65000 Bytes dengan jumlah paket yang akan dikirim 100000, memperoleh hasil sebagai berikut:

a. Hasil pengujian DOS PPTP menunjukkan pada paket 341 kemudian pada pengujian lain menunjukkan paket 73944, VPN menggunakan PPTP mengalami gangguan sehingga terputus (*disconnect*) dari VPN *server*, dan perlu *connect* kembali jika ingin menggunakan VPN PPTP kembali.

b. Hasil pengujian DOS *Open*VPN menunjukkan paket 885 sedangkan pada pengujian pada komputer lain menunjukkan paket 951222 , untuk mengalami gangguan. Koneksi pada VPN *Open*VPN mengalami RTO tetapi tidak terputus dari VPN server.

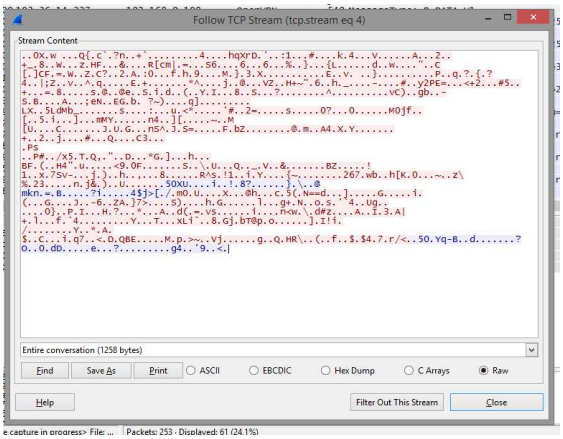
5. *Sniffing* Jaringan VPN

Proses *sniffing* menggunakan Wireshark pada kedua jaringan VPN diperoleh hasil yang berbeda diantara keduanya. *Sniffing* yang dilakukan pada VPN PPTP hanya menunjukkan aktivitas tunnel VPN hal itu ditunjukkan pada kolom info yaitu *Encapsulated* PPP dan *Compressed* Data, sehingga tidak dapat menunjukkan tahap selanjutnya yaitu "*follow tcp stream*".



Gambar 9 *Sniffing* PPTP

Pada hasil *sniffing* jaringan OpenVPN menggunakan Wireshark terlihat dengan jelas kegiatan OpenVPN dengan ditunjukkan pada bagian *protocol* bertuliskan OpenVPN. Saat melakukan tahap selanjutnya yaitu *follow* tcp *stream* diperoleh hasil yang tidak jelas atau acak seperti yang terlihat pada gambar 9.Gambar 10 menunjukkan data random atau acak dari hasil *sniffing*.



Gambar 10 TCP *Stream* OpenVPN

Analisis Data/Evaluasi

Berdasarkan hasil data yang diperoleh dapat terlihat perbedaan teknologi diantara kedua VPN *server* tersebut. PPTP dan OpenVPN memiliki metode yang berbeda dalam hal teknik membuat VPN.

Dari hasil pengujian performa pada VPN, dapat dilihat bahwa perbedaan performa pada kedua VPN tidak terlalu signifikan. Hal ini dapat terlihat pada tabel *PacketLoss* antara kedua VPN, kedua VPN tersebut mampu menyelesaikan *ping* tanpa ada paket yang hilang atau *loss*. Hasil pengujian *RoundTrip* menunjukkan bahwa waktu yang dimiliki antara kedua VPN memanglah tidak sama, baik itu dalam waktu minimum, maksimum, maupun rata-rata waktu tetapi perbedaan tersebut tidaklah begitu jauh mengingat satuan waktu yang digunakan adalah *milisecond*. Waktu paling minimum dimiliki oleh OpenVPN, ini menunjukkan OpenVPN dapat sedikit lebih cepat dari PPTP, untuk waktu paling maksimum dimiliki oleh OpenVPN juga, menunjukkan OpenVPN berjalan sedikit lebih lama dari PPTP, sedangkan untuk rata-rata waktu kedua VPN hampir sama hanya berbeda sangat tipis sekali. Pengujian untuk WCP Transfer yaitu pengujian untuk mengukur kecepatan transfer pada kedua VPN menghasilkan data yang berbeda antara kedua VPN dalam hal *download* dan *upload*. Pada pengujian *download* kedua VPN menghasilkan waktu yang sama, hanya berbeda *milisecond*, sedangkan untuk *upload* terdapat perbedaan yang sedikit lebih jelas waktu yang dimiliki oleh OpenVPN lebih baik dari PPTP seperti yang terlihat pada tabel. Dari hasil pengujian performa tersebut, performa yang dimiliki antara kedua VPN hampir sama, namun apa bila lebih diperhatikan OpenVPN memiliki performa lebih baik dari pada PPTP.VPN memiliki waktu lebih cepat untuk transfer file dan lebih banyak jumlah paket yang diterima oleh OpenVPN saat dilakukan serangan sebelum akhirnya mengalami gangguan pada *service* VPN.

OpenVPN memang memiliki tingkat enkripsi yang baik dengan otentikasi menggunakan sertifikat tetapi OpenVPN juga memiliki kehandalan yang lebih unggul dari PPTP pada pengujian transfer *file* dikarenakan OpenVPN mempunyai kemampuan dalam menggunakan protokol dan dapat berjalan cepat walaupun di dalam koneksi yang tinggi akan *latency* dan jarak yang jauh, serta disebabkan OpenVPN memiliki *default* MTU (*Maximum Transmission Unit*) yang lebih besar dari PPTP yaitu 1500 sedangkan PPTP hanya 1460. MTU memberikan maksimum dari panjang paket yang dapat dikirim dalam sebuah *frame* pada lapisan *Network Interface*.

Pada pengujian keamanan VPN memanglah hanya pengujian yang tidak terlalu komplikasi atau rumit. Untuk pengujian *Dial of Service* menunjukkan perbedaan yang jelas antara kedua VPN, apabila dilihat hasil pengujian, PPTP *server* mengalami gangguan pada paket 341, sedangkan OpenVPN pada paket 885

dari total 100000 paket. Hal ini menunjukkan bahwa OpenVPN memiliki ketahanan terhadap serangan lebih baik dari pada PPTP. Untuk pengujian sniffing PPTP memang tidak dapat dilacak dengan aplikasi Wireshark, sedangkan OpenVPN dapat terlihat dengan jelas aktivitasnya, namun hasil dari *follow tcp stream* menghasilkan data yang tidak jelas ini menunjukkan bahwa data tersebut telah dienkripsi oleh OpenVPN *server*.

PENUTUP

Kesimpulan

Implementasi VPN dengan menggunakan metode PPTP dan *Open*VPN bertujuan untuk mengetahui efisiensi dari metode tersebut. Hasil yang diperoleh dari implementasi dan pengujian metode VPN tersebut, dapat disimpulkan sebagai berikut:

a. VPN menggunakan PPTP dan *Open*VPN dapat diimplementasikan pada jaringan *server* sehingga *user* atau *client* dapat mengakses dimana saja dan kapan saja melalui jaringan internet.
b. Pengujian yang dilakukan pada performa menghasilkan perbedaan yang tidak begitu signifikan, tetapi apabila diamati *Open*VPN lebih unggul dari PPTP, hal ini ditunjukkan saat pengujian transfer *file Open*VPN memiliki waktu lebih cepat. Sedangkan pada pengujian keamanan *Open*VPN lebih unggul dari PPTP, hal ini dapat dilihat dari lebih banyaknya jumlah paket yang diterima oleh *Open*VPN saat dilakukan serangan sebelum akhirnya mengalami gangguan pada *service* VPN.

UCAPAN TERIMAKASIH

Penulis mengucapkan terima kasih atas bantuan sarana dan prasarana dari Laboratorium Komputer dan Jaringan Teknik Informatika dan Komputer Politeknik Negeri Jakarta dalam mendukung penelitian ini.

DAFTAR PUSTAKA

Afrianto, Irawan and Setiawan, Eko Budi. (2014). Majalah Ilmiah Unikom: Kajian Virtual Private Network (VPN) Sebagai Sistem Pengamanan Data pada Jaringan Komputer (Studi Kasus Jaringan Komputer Unikom). UNIKOM, 43-50.

Fajrin, Akbar, dkk. (2011). *Pengaman Sistem Jaringan*. Universitas Pembangunan Nasional.

Hendriana, Yana. (2012). *Evaluasi Implementasi Keamanan Jaringan Virtual Private Network (VPN) (Studi Kasus Pada C.V. Pangestu Jaya)*. Yogyakarta:Universitas Ahmad Dahlan.

Nandareynaldi. (2015). *Serangan DoS dengan Pingflood*. http://www.binushacker.net/serangan-dos-dengan-pingflood.html. [1 Juli 2015]

Purbo, Onno W. (2015). *Virtual Private Network (VPN) sebagai alternatif Komunikasi Data Pada Jaringan Skala Luas (WAN)*. http://kambing.ui.ac.id/onnopurbo/ library/library-ref-ind/ref-ind-3/network/VPN_jurnal.pdf.

Sakiwan. (2010). *Kajian Virtual Private Network (VPN) LAPAN dan Pemanfaatannya Dalam Mendukung Pengembangan E-Goverment*. *Berita Dirgantara*, 11, 145-152.

Sukaridhoto, Stritrusta. (2007). *OpenVPN*. Politeknik Elektronika Negeri Surabaya.