BAB III
ANALISIS DAN PERANCANGAN SISTEM

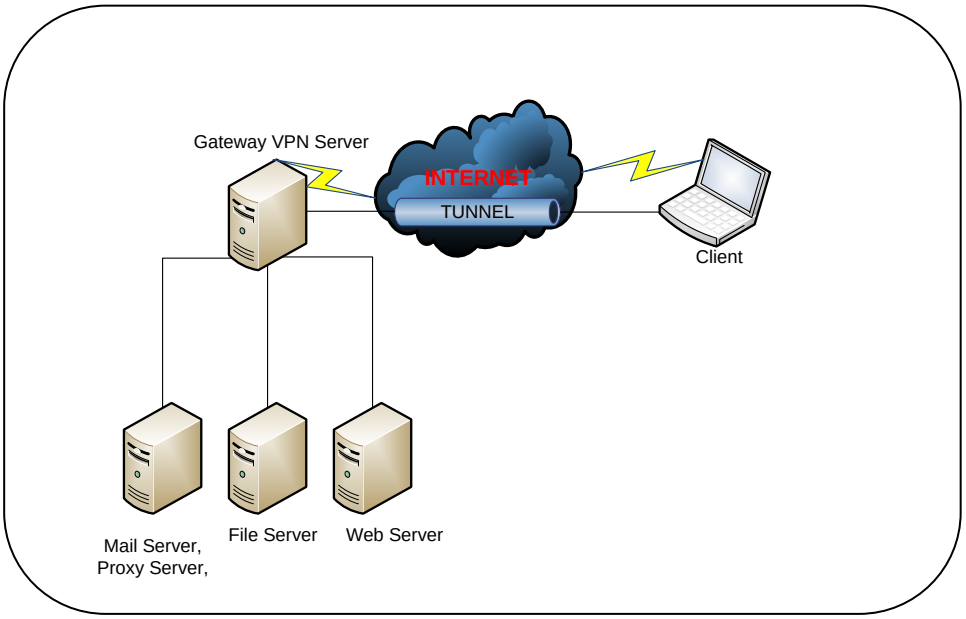
3.1 Analisis Sistem

Analisis Sistem adalah penguraian dari suatu masalah yang utuh ke dalam bagian-bagian komponennya dengan maksud untuk mengidentifikasikan dan mengevaluasi permasalahan-permasalahan, kesempatan-kesempatan, hambatan-hambatan yang terjadi dan kebutuhan-kebutuhan yang diharapkan sehingga dapat diusulkan perbaikan-perbaikannya. Bab ini akan menguraikan proses analisis sistem *remote access* VPN dengan metode *Secure Socket Layer* (SSL) dan perancangan sistem *remote access* VPN.

3.1.1 Analisis Masalah

Remote Access Virtual Private Network biasanya digunakan untuk menguhubungkan pengguna yang *mobile* dari berbagai lokasi yang jauh (*remote*). Biasanya jenis VPN ini digunakan oleh pegawai perusahaan yang ingin terhubung ke jaringan khusus perusahaannya dari berbagai lokasi yang jauh (*remote*) dari perusahaannya. Biasanya perusahaan yang ingin membuat jaringan VPN tipe ini akan bekerjasama dengan *enterprise service provider* (ESP). ESP akan memberikan suatu *network access server* (NAS) bagi perusahaan tersebut. ESP juga akan menyediakan *software client* untuk komputer-komputer yang digunakan pegawai perusahaan tersebut.

Untuk mengakses jaringan lokal perusahaan, pegawai tersebut harus terhubung ke NAS dengan men-*dial* nomor telepon yang sudah ditentukan. Kemudian dengan menggunakan *sotware client*, pegawai tersebut dapat terhubung ke jaringan lokal perusahaan.



Gambar 3.1. *Remote Access VPN secara umum*

Perusahaan yang memiliki pegawai yang ada di lapangan dalam jumlah besar dapat menggunakan *remote access* VPN untuk membangun WAN. VPN tipe ini akan memberikan keamanan, dengan mengenkripsi koneksi antara jaringan lokal perusahaan dengan pegawainya yang ada di lapangan. Akan tetapi kebutuhan akan tingkat keamanan data yang tinggi dan solusi membangun keamanan jaringan yang handal yang tidak mengeluarkan dana tidak begitu besar masih sangat mungkin dilakukan. Oleh karena itu bagaimana membangun suatu *Remote Access* VPN dengan tidak banyak mengeluarkan dana yang besar, maka

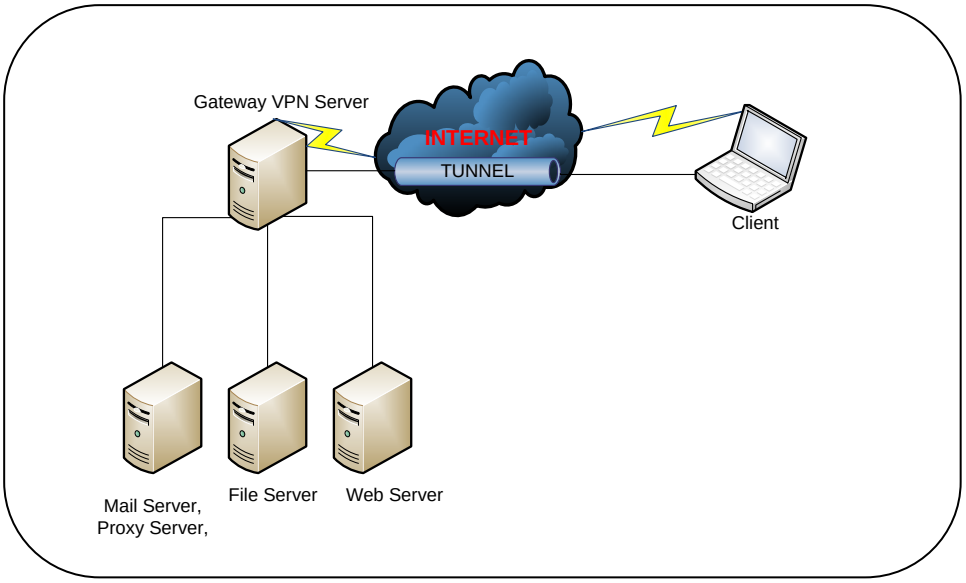
dapat dilakukan konfigurasi VPN pada tingkat sistem operasi, dengan tetap mempunyai kehandalan dalam keamanan.

Dikarenakan data/file yang terdapat pada *server* merupakan data perusahaan yang merupakan data rahasia perusahaan dan data kerja karyawan, maka dibutuhkan mekanisme keamanan komunikasi data. Mekanisme yang digunakan pada penelitian menggunakan konsep SSL VPN yang merupakan protokol komunikasi data yang mempunyai keunggulan untuk pertukaran *public key* (PKI) untuk mekanisme enkripsi dan autentikasi berbasis sertifikat dan menggunakan metode *asymmetric encryption* sebagai akses *remote* dari *client* ke *server* yang merupakan konsep kriptograpi kunci/sertifikat yang cocok digunakan dalam model jaringan *remote access* VPN

3.1.2 Analisis Topologi

Remote Access merupakan sebuah kemampuan untuk mengakses *resource-resource* pada suatu layanan maupun jaringan komputer dari lokasi manapun. Penerapan *remote access* vpn akan memberikan sebuah kemampuan untuk mengakses *resource-resource* pada suatu layanan maupun jaringan komputer melalui jalur *private* yang lebih aman.

Model topologi jaringan *Remote Access Virtual Private Network* yang umum digunakan biasanya terdiri dari sebuah komputer sebagai *server* VPN, *server* pendukung lain seperti web server, file server, dan *server-server* yang lain dan komputer *client* dengan bentuk topologi seperti pada gambar 3.2 berikut:



Gambar 3.2 *Topologi Remote Access VPN*

Pada skema diatas, *client* akan melakukan *remote access* terhadap layanan-layanan yang terdapat pada *server*. Dimana *client* memerlukan untuk mengakses file yang memerlukan koneksi langsung ke *server*. Sebelum *client* dapat mengakses file maupun layanan-layanan yang terdapat pada *server* jaringan lokal, *client* membuat koneksi dengan sebuah VPN *Server* sebagai mekanisme kemananan komunikasi data, untuk menjaga keamanan data pada saat koneksi dengan *server*, kemudian pada VPN *Server* akan melakukan *forwading* kepada layanan yang diinginkan oleh *client*, kemudian layanan yang diinginkan oleh *client* yang telah terkoneksi vpn akan me-*reply* layanan yang diinginkan tersebut. Dengan menggunakan koneksi VPN untuk mengakses layanan yang terdapat pada jaringan lokal, koneksi dari *client* ke *server* dapat dilakukan dengan aman.

Client dapat mengakses *server-server* seperti file server, web server, dan *server-server* yang lain setelah *client* melakukan koneksi melalui vpn, sehingga pengaksesan *server-server* tersebut dilakukan melalui *network interface* vpn, sehingga pengaksesan data dari *server* dapat dilakukan dengan aman.

3.1.3 Analisis Protokol

3.1.3.1 SSL VPN

Secure Socket Layer (SSL) adalah protokol yang digunakan untuk berkomunikasi lewat internet secara aman. Teknologi SSL menggunakan konsep teknologi kriptografi kunci publik untuk bisa mencapai komunikasi yang aman antar *server* dan *client*. Kedua pihak yang berkomunikasi ini (*server* dan *client*) saling mengirimkan data yang disamarkan dan untuk membacanya digunakan sandi dan kunci yang dimiliki kedua pihak yang berkomunikasi tersebut. SSL VPN menggunakan infrastruktur publik yang sudah ada di internet untuk melakukan pertukaran data (misal: antara kantor pusat sebuah perusahaan dan kantor cabangnya).

Karena dilewatkan pada jaringan internet publik, permasalahan konsep SSL VPN muncul sama seperti jaringan IP pada umumnya. Salah satu masalah jaringan internet (*IP public*) adalah tidak mempunyai dukungan yang baik terhadap keamanan. SSL VPN dapat digunakan untuk mengatasi persoalan keamanan tersebut. Dasar dari konsep SSL VPN ini adalah penggunaan infrastruktur IP untuk hubungan suatu *server* dengan *client*-nya dengan cara pengalamatan secara *private* dengan melakukan pengamanan terhadap transmisi paket data. Di dalam proses kerjanya, SSL terdiri dari beberapa fase, yaitu:

1. Negosiasi antar *peer* terhadap algoritma yang didukung oleh masing-masing *peer/endpoint* tersebut.
2. Pertukaran *Public Key* (*PKI*) untuk mekanisme enkripsi dan autentikasi berbasis sertifikat.
3. Enkripsi simetris maupun enkripsi asimetris.

SSL VPN memegang peranan penting untuk pengawasan transaksi pertukaran data/akses sehingga harus mempunyai mekanisme untuk memblokir akses ke jaringan privat *server* ketika jumlah akses bersamaan melebihi batas yang telah ditentukan. SSL VPN gateway harus mempunyai mekanisme sehingga setiap *client* memiliki administratornya sendiri yang mempunyai kewenangan seperti mengatur hak akses dari *client*, mengatur akses user pada sumber daya mereka. Perlu memeriksa sertifikat yang diterimanya agar lebih yakin bahwa *client* sedang berkomunikasi dengan *server* yang diinginkan. *Client* memeriksa sertifikat digital itu dengan membandingkan tanda tangan CA(*certificate of authority*) pada sertifikat digital itu dengan daftar OS yang dimiliki. Penyertaan sertifikat digital CA utama pada browser akan menghindarkan *client* dari pemalsuan sertifikat CA utama.

Adapun algoritma kriptografi yang saat ini umumnya digunakan adalah:

1. Untuk kriptografi *Public Key*: RSA, Diffie-Hellman, DSA atau Fortezza;
2. Untuk enkripsi simetris: RC2, RC4, IDEA, DES, Triple DES atau AES;

3. Untuk fungsi *hash* searah: MD5 atau SHA.

3.1.3.2 Fungsi SSL VPN

Beberapa Spesifikasi Fungsi yang umumnya ada dalam perangkat SSL VPN antara lain adalah:

1. Mode Operasi

Ada dua mode operasi yang biasanya ada pada perangkat SSL VPN yaitu:

a. **Static routes** : untuk mendukung pembentukan VPN dengan menggunakan rute statis yang di *assign* oleh admin secara permanen dengan software operasi.
b. **Unrestricted** : untuk mendukung pembentukan VPN dengan menggunakan mode berdasarkan kebutuhan *support* dari user, biasanya dilakukan secara dinamis dan *on demand*.

Pada penelitian yang dilakukan, mode operasi yang digunakan adalah *static routes* untuk pembentukan VPN yang telah ditentukan oleh admin.

2. Atribut SSL VPN

Beberapa atribut yang dimiliki secara umum oleh perangkat SSL VPN yang menunjukan kapabilitas sistem tersebut adalah:

a. Kapabilitas kreasi jumlah tunnel SSL VPN Tunnels
b. Kemampuan mendukung Versi SSL

c. Kemampuan melakukan enkripsi dan dukungannya terhadap metode enkripsi yang saat ini ada.

d. *Certificate* Sup*port*

3.1.3.3 *Asymmetric Encryption*

SSL/TLS menggunakan salah satu teknologi enkripsi yang sering digunakan pada SSL VPN, enkripsi asimetric merupakan suatu konsep untuk memastikan identitas dari mitra VPN. Kedua mitra enkripsi memiliki dua kunci masing-masing: satu publik dan satu privat. Kunci publik itu diserahkan kepada mitra komunikasi, untuk meningkatkan datanya. Oleh karena algoritma mathematical yang terpilih digunakan untuk menciptakan pasangan kunci *public/private*, hanya kunci pribadi penerima itu yang dapat mengurangi data yang disandikan oleh kunci publiknya. Pada penerapannya *asymmetric encryption* berbasiskan *key* dan *certificate*, yaitu berupa kunci-kunci dan sertifikat yang dibutuhkan oleh komputer untuk melakukan koneksi.

Kunci publik yang harus dimiliki oleh server maupun client yaitu berupa CA (*Certificate of Authority*), sertifikat ini harus dimiliki baik oleh *server* maupun *client*, dan kunci *private* yang harus dimiliki adalah *key server* dan *key client*.

3.1.4 Kebutuhan Pendukung Infrastruktur

Kebutuhan akan infrastruktur terbagi menjadi dua macam, yaitu perangkat keras dan perangkat lunak, yang keduanya saling mendukung satu sama lain.

3.1.4.1 Kebutuhan Perangkat Keras

Kebutuhan hardware pada skripsi ini terdiri dari beberapa perangkat keras meliputi PC sebagai VPN *server*, PC sebagai *server*, PC sebagai *client*, dan modem.

a. VPN *Server*

Komputer VPN *server* pada *Remote Access Virtual Private Network* yaitu bertindak sebagai *server* VPN, *server* VPN bertugas membentuk *tunnel*, melakukan validitas *certificate*, dan sebagai gateway antar vpn *client*. Komputer yang bertindak sebagai *server* VPN mempunyai spesifikasi sebagai berikut :

Tabel 3.1 Spesifikasi Perangkat Keras VPN Server

No	Komponen	Keterangan
1	Processor	Intel Core Solo 1.8 GHz
2	Memori	1.2 Gb
3	Harddisk	60 Gb
4	Operating System	Ubuntu Lucid Lynx

b. *Server*

Komputer *server* menyediakan layanan-layanan yang akan diakses oleh *client*, seperti web server dan file server. Adapun spesifikasinya adalah sebagai berikut:

Tabel 3.2 Spesifikasi Perangkat Keras Server

No	Komponen	Keterangan
1	Processor	Intel Pentium 4 2.0 GHz
2	Memori	1 Gb
3	Harddisk	80 Gb

4	Operating System	Ubuntu Lucid Lynx

c. *Client*

Client VPN terdiri dari beberapa PC spesifikasi dari komputer *client* adalah sebagai berikut :

Tabel 3.3 Spesifikasi Perangkat Keras Client

No	Komponen	Keterangan
1	Processor	Intel Atom 1.6 GHz
2	Memori	1 Gb
3	Harddisk	160 Gb
4	Operating System	Windows 7 / Ubuntu Lucid Lynx

3.1.4.2 Kebutuhan Perangkat Lunak

Software yang digunakan pada skripsi ini meliputi openvpn, tcpdump, nmap dan wireshark.

a. **Openvpn** adalah salah satu aplikasi VPN berbasiskan *open source* untuk membuat koneksi *encrypted tunnels* secara *virtual* dengan menggunakan autentikasi dengan yang lainnya menggunakan *pre-shared secret key*, *certificates*, atau *username* OpenVPN memiliki beberapa kelebihan untuk membuat VPN *server* seperti berbasiskan *open source*, keamanan, kestabilan, mekanisme *autentikasi* dan *enkripsi*.

Untuk mengkonfigurasi *Remote Access Virtual Private Network* melalui software openvpn, setidaknya harus mengkonfigurasi empat file konfigurasi, yang diantaranya adalah:

1. File openssl.cnf merupakan file konfigurasi yang digunakan oleh file konfigurasi yang lain sebagai parameternya. Openssl.cnf yang terdapat di direktori /usr/share/doc/openvpn/examples/easy-rsa/2.0/openssl.cnf

   Pada file openssl.cnf dibutuhkan konfigurasi untuk menentukan besar bits yang akan dienkripsi, masa validasi dan jenis enkripsi yang akan digunakan pada koneksi VPN.

   Pada file openssl.cnf untuk menentukan besar bits yang akan di enkripsi, masa validasi dan jenis enkripsi yang akan digunakan dari koneksi VPN terdapat pada bagian seperti yang ditunjukkan Tabel 3.4 di bawah ini:

Tabel 3.4 Parameter Openssl Yang Dibutuhkan

Parameter	**Keterangan**
Default_days	Untuk menentukan lama aktif dari sertifikat dan kunci yang digunakan
Default_crl_days	Untuk menentukan lama aktif dari crl yang digunakan
Default_md	Jenis enkripsi yang akan digunakan
preserve	

2. Vars merupakan file konfigurasi yang berisikan informasi umum yang akan digunakan dalam pembuatan sertifikat dan kunci. Parameter-parameter yang terdapat dalam vars diantaranya sebagai berikut:

Tabel 3.5 Parameter Vars

Parameter	Keterangan
Export OPENSSL	Informasi penggunaan OPENSSL
Export PKCS11TOOL	Informasi penggunaan tool PKCS
Export KEY_CONFIG	Informasi tempat penyimpanan file konfigurasi
Export KEY_DIR	Informasi tempat penyimpanan sertifikat dan kunci seteah proses pembuatan sertifikat dan kunci
Export KEY_SIZE	Informasi besarnya ukuran sertifikat dan kunci
Export KEY_COUNTRY	Informasi negara yang akan dimasukan ke sertifikat
Export KEY_PROVINCE	Informasi provinsi yang akan dimasukan ke sertifikat
Export KEY_CITY	Informasi kota yang akan dimasukan ke sertifikat
Export KEY_ORG	Informasi organisasi yang akan dimasukan ke serfifikat
Export KEY_EMAIL	Informasi email yang akan dimasukan ke sertifikat

3. File *Server*.conf merupakan file yang harus dikonfigurasi untuk menentukan bahwa komputer tersebut merupakan VPN *server*. *Server*.conf yang terdapat di direktori /usr/share/doc/openvpn/examples/sample-config-files/*server*.conf.

File *server*.conf digunakan sebagai konfigurasi untuk VPN *server*. Parameter-parameter yang digunakan diantaranya sebagai berikut:

Tabel 3.6 Parameter Konfigurasi VPN Server

Parameter	Keterangan
Server	Menunjukan sebagai *server*
Port 1194	*Port* yang akan dibuka untuk vpn (1194 merupakan *port* default untuk openvpn)
Proto tcp	Protokol yang digunakan adalah tcp
Dev tap	"dev tun" akan menciptakan tunnel routing IP. "dev tap" akan menciptakan tunnel Ethernet.
Ca /etc/openvpn/keys/ca.crt Cert /etc/openvpn/keys/*server*.crt Key /etc/openvpn/keys/*server*.key	Sertifikat dan kunci-kunci yang digunakan *server* serta tempat beradanya sertifikat dan kunci tersebut
Dh /etc/openvpn/keys/dh1024.pem	Parameter Diffie-Hellman dan tempat beradanya parameter Diffie-Hellman
Server 10.10.10.0 255.255.255.0	Mode konfigurasi *server* dan suplai dari subnet VPN. Dengan ditentukan parameter tersebut, maka IP *server* yaitu 10.10.10.1 dan sisanya adalah untuk *client*.
Ifconfig-pool-persist ipp.txt	Record perbaikan dari *client*.
Push "route 192.168.1.0 255.255.255.0"	Melakukan routing ke jaringan intranet dengan alamat broadcast 192.168.1.0/24
Client-to-*client*	Mengizinkan *client* agar dapat melihat satu sama lain
Keepalive 10 120	Menjaga agar pesan aktif antara pihak yang berhubungan dapat terjaga, dengan asumsi sistem akan ping selama 10 detik apabila pada jangka waktu 120

	detik tidak diterima ping.
Comp-lzo	Meng-*enable*-kan kompresi pada jalur VPN
Max-clients	Jumlah *client* yang diizinkan
User nobody Group nogroup	Untuk mengurangi hak istimewa daemon openvpn
Persist-key Persist-tun	Berguna untuk menghindari mengakses sumber daya tertentu pada restart yang mungkin tidak dapat diakses lagi karena *downgrade* hak istimewa
Status /var/log/opnvpn-status.log	Log dari status koneksi vpn
Verb 4	Untuk penggunaan umum
Mute 10	Tidak mengulangi pesan. Maksimal 10 pesan dari kategori pesan yang sama akan menjadi output untuk log
Plugin /usr/lib/openvpn/openvpn-authpam.so login	Menggunakan *plugin* untuk authentikasi *client*

4. File *Client*.conf digunakan sebagai konfigurasi untuk VPN menandakan bahwa komputer tersebut merupakan komputer VPN *client*. *Client*.conf yang terdapat di dalam direktori /usr/share/doc/openvpn/examples/sample-config-files/*client*.conf (berbasis Linux). Sedangkan yang berrbasis windows terdapat di dalam diretori C:/Program Files/OpenVPN/config/.

File *client*.conf berbasis sistem operasi linux, sedangkan untuk sistem operasi windows, format penamaan *client*.conf menjadi *client*.ovpn. Parameter-parameter yang digunakan adalah sebagai berikut:

Tabel 3.7 Parameter Konfigurasi Client

Parameter	Keterangan
Client	Menunjukan sebagai *client*
Dev tap	"dev tun" akan menciptakan tunnel routing IP. "dev tap" akan menciptakan tunnel Ethernet.
Proto tcp	Protokol yang digunakan adalah tcp
Remote IP Server 1194	IP Address *server* dan *port* yang dibuka untuk VPN
Resolve-retry infinite	Untuk mencoba menjaga koneksi dengan *server*.
Nobind	*Client* tidak perlu mengikat *port*
Persist-key Persist-tun	Berguna untuk menghindari mengakses sumber daya tertentu pada *restart* yang mungkin tidak dapat diakses lagi karena *downgrade* hak istimewa
Ca /etc/openvpn/keys/ca.crt Cert /etc/openvpn/keys/server.crt Key /etc/openvpn/keys/server.key	Sertifikat dan kunci-kunci yang digunakan *server* serta tempat beradanya sertifikat dan kunci tersebut
Comp-lzo	Meng-*enable*-kan kompresi pada jalur VPN
Verb 4	Untuk penggunaan umum
Auth-user-pass	Menggunakan authentikasi untuk *client* yang berupa *username* dan *password.*

b. TCPDUMP

TCPdump adalah tools yang berfungsi meng*capture*, membaca atau men*dump*ing paket yang sedang ditransmisikan melalui jalur TCP. TCPdump diciptakan untuk menolong programmer ataupun administrator dalam menganalisa dan *troubleshooting* aplikasi *networking*. Seperti pisau yang bermata dua, TCPdump bisa digunakan untuk bertahan dan juga bisa digunakan untuk menyerang.

TCPDump seringkali digunakan oleh para cracker untuk melaksanakan perkerjaannya, karena TCPdump bisa meng-*capture* atau men-*sniff* semua paket yang diterima oleh *network interface*. Sebagai contoh, digunakan *network* yang terdiri dari tiga komputer yang dihubungkan menggunakan *hub*. Komputer pertama, menggunakan sistem operasi Windows XP dengan IP *address* 192.0.0.1, sedang melakukan koneksi melalui telnet ke komputer kedua yang menggunakan Slackware 8.0 dengan IP *address* 192.168.0.2 dan *host* ketiga komputer Redhat 7.1 dengan IP *address* 192.168.0.3 yang menggunakan *utility* TCPdump. Alasan untuk membedakan *Operating System* yang digunakan adalah untuk menunjukkan bahwa TCP/IP dapat berkomunikasi dengan baik pada dua platform yang berbeda.

Untuk menjalankan TCPdump, ketik perintah tcpdump di *console* pada *host* 192.168.0.3 sebagai root. *Output* yang diperlihatkan di bawah ini adalah *output* yang bergulir *non-stop*, terus berganti baris tanpa henti hingga menekan Ctrl+C (^C) untuk menghentikan *utility* ini.

c. NMAP

Nmap (*Network Mapper*) adalah sebuah aplikasi atau *tool* yang berfungsi untuk melakukan *port scanning*, eksplorasi dan audit keamanan jaringan. Nmap dirancang untuk memeriksa jaringan besar secara cepat, meskipun dapat pula bekerja terhadap *host* tunggal. Nmap menggunakan paket IP raw dalam cara yang canggih untuk menentukan *host* mana saja yang tersedia pada jaringan, layanan (nama aplikasi dan versi) apa yang diberikan, sistem operasi (dan versinya) apa yang digunakan, apa jenis *firewall*/*filter* paket yang digunakan, dan sejumlah karakteristik lainnya. Meskipun Nmap umumnya digunakan untuk audit keamanan, namun banyak administrator sistem dan jaringan menganggapnya berguna untuk tugas rutin seperti inventori jaringan, mengelola jadwal *upgrade* layanan, dan melakukan *monitoring uptime host* atau layanan

d. Wireshark

Wireshark adalah program yang berfungsi untuk mengetahui kejadian yang terjadi pada saat kita melakukan interaksi pada suatu jaringan komputer. Wireshark merupakan salah satu *network analysis tool*, atau disebut juga dengan *protocol analysis tool* atau *packet sniffer*. Wireshark dapat digunakan untuk *troubleshooting* jaringan, analisis, pengembangan software dan protokol, serta untuk keperluan edukasi.

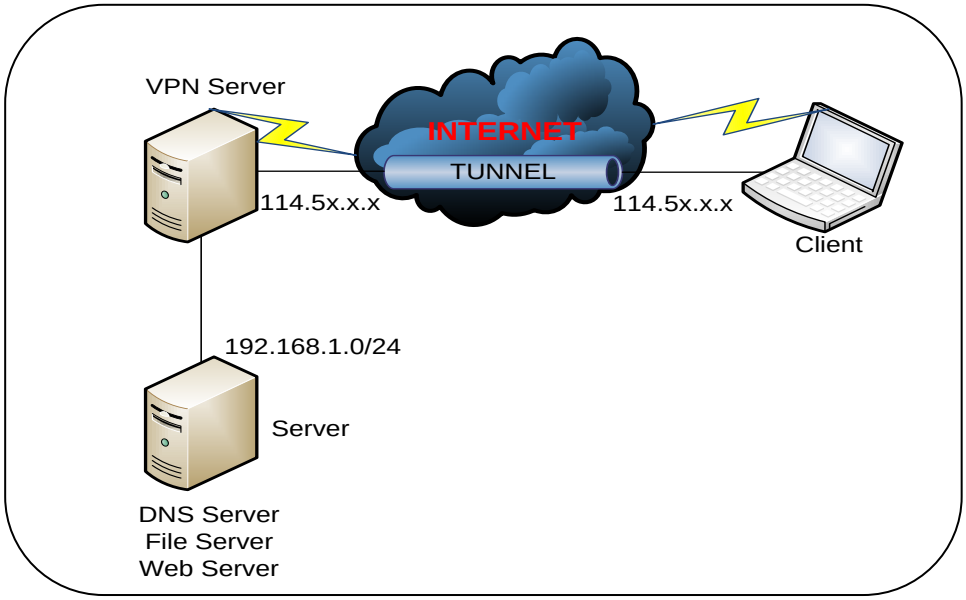
Wireshark mampu menangkap paket-paket data atau informasi yang berjalan dalam suatu jaringan yang dimonitor. Hampir semua jenis paket informasi dalam berbagai format protokol dapat ditangkap dan dianalisa.

3.2 Perancangan Sistem

Pada perancangan sistem akan dibahas mengenai perancangan sistem *remote access* VPN, yang meliputi perancangan infrastruktur VPN atau arsitektur simulasi dan instalasi perangkat lunak.

3.2.1 Arsitektur Simulasi

Perancangan Jaringan yang akan digunakan pada skripsi ini terdiri dari sebuah komputer sebagai *server* VPN, *server* dan komputer *client* dengan bentuk topologi seperti yang ditunjukan pada gambar 3.3:



Gambar 3.3 *Simulasi Jaringan Remote Access VPN*

Pada VPN *server* merupakan Dedicated *Server* VPN dan sebagai internet gateway, sedangkan *server-server* lain seperti web server, file server, yang dapat diakses oleh VPN *client*, terdapat pada *server* tersendiri.

Pada komputer *client* terkoneksi dengan internet menggunakan modem HSDPA dengan kecepatan akses maksimal 256 KBps dan mendapatkan IP Publik dinamis (114.5x.x.x).

Pada komputer *client* terkoneksi dengan internet menggunakan modem HSDPA dengan kecepatan akses maksimal 256 KBps dan mendapatkan IP Publik dinamis (114.5x.x.x). VPN *Server* dihubungkan dengan sebuah jaringan lokal yang merupakan komputer *server* penyedia layanan-layanan seperti file *server*, web server dan *server* lainya. Komputer tersebut mempuyai alamat 192.168.1.0/24.

Skenario yang akan dilakukan yaitu *client* yang terkoneksi dengan internet akan mengakses layanan pada *server*, terlebih dahulu *client* melakukan koneksi terhadap VPN *server*, yang kemudian VPN *server* akan melakukan *forwading* menuju komputer *server* lain terhadap request yang dilakukan oleh *client*.

VPN *server* yang berfungsi sebagai infrastruktur komunikasi yang mempunyai mekanisme protokol komunikasi data SSL memberikan IP Address kepada *client* yang terkoneksi dengan VPN *server*, sehingga pada *client* akan mempunyai IP address vpn yang telah diberikan oleh *server*, sehingga apabila *client* melakukan *remote* terhadap *server* melalui koneksi internet, maka *network* yang digunakan untuk berkomunikasi dengan *server* yaitu menggunakan *network* vpn.

Pada VPN *server* akan diberikan alamat IP 10.10.10.1 dengan subnet 24 bit(255.255.255.0), sedangkan *client* akan dialokasikan alamat IP dengan range

10.10.10.2 – 10.10.10.254, sehingga akan menghasilkan alamat IP lengkap sebagai berikut:

Tabel 3.8. Alokasi IP Address

Device	IP Address	Publik/Private	Interface
VPN *Server*	114.5x.x.x.	IP Publik	Modem HSDPA
	192.168.1.1	IP Private	Eth0
	10.10.10.1	IP VPN	Tap0
Server	192.168.1.100	IP Private	Eth0
Client	114.5x.x.x	IP Publik	Modem HSDPA
	10.10.10.x	IP VPN	Tap0

3.2.2 Perancangan Perangkat Lunak

3.2.2.1 Instalasi Openvpn

Dalam instalasi openvpn pada sistem operasi untuk VPN *server* yang digunakan adalah distro Ubuntu 10.04 Lucid Lynx dengan kernel 2.6.32-16. Dan terdapat beberapa prasyarat yang harus diperlukan untuk sistem operasi, diantanya yaitu:

1. **Sistem Operasi Harus Mendukung untuk Driver Universal TUN/TAP:**

   Kernel yang lebih baru dari versi 2.4 dari hampir semua distribusi modern Linux mendukung untuk perangkat TUN/TAP. Apabila menggunakan suatu distribusi lama atau jika telah membangun kernel sendiri, harus menambahkan support ini ke konfigurasi.

2. **Library OpenSSL Harus Telah Diinstal Pada System**
3. **Library Lempel-Ziv-Oberhumer (LZO) Compression Harus Diinstal**

LZO adalah suatu *library* kompresi *real-time* yang digunakan oleh Openvpn untuk memampatkan data sebelum pengiriman (paket library lzo bernama liblzo-2).

3.2.2.1.1 Instalasi Openvpn Di Linux

Pada skripsi ini digunakan Openvpn versi 2.1.1. Kernel linux yang digunakan adalah kernel versi 2.6.32-16 Ubuntu 10.04 Lucid Lynx. untuk menginstall Openvpm dapat digunakan perintah:

root@herwan:~#apt-get install openvpn

Setelah proses instalasi, maka file hasil instalasi akan tersimpan pada lokasi-lokasi yang ditunjukan pada tabel 3.8:

Tabel 3.8 Tempat dan File yang Terinstal oleh Openvpn

Lokasi alamat dan File Openvpn	Fungsi
/etc/openvpn	Direktori yang berisi file konfigurasi
/etc/network/if-up.d/openvpn /etc/network/if-down.d /etc/network/if-down.d/openvpn	Men-start/stop openvpn ketika jaringan menjadi up/down
/etc/init.d/openvpn start /etc/init.d/openvpn stop	Men-start/stop service openvpn
/sbin/openvpn	File binary
/usr/share/doc/openvpn	File-file dokumentasi
/usr/share/man/man/openvpn.gz	Halaman manual openvpn
/usr/share/doc/openvpn/examples/sample-config-files	Contoh file konfigurasi
/usr/share/doc/openvpn/examples/sample-keys	Contoh file kunci(*keys*)
/usr/share/doc/openvpn/examples/easy-	Kumpulan script utnuk

rsa	pembuatan tunnel
/usr/share/doc/openvpn/change.log.debian.gz /usr/share/doc/openvpn/change.log.gz	History version
/usr/share/openvpn/verify-cn	Verifikasi fungsi –cn (perintah revoke)
/usr/lib/openvpn/openvpn/openvpn-auth-pam.so /usr/lib/openvpn/openvpn-down-root.so	Chroot mode dan library untuk PAM autentifikasi

3.2.2.1.2 Instalasi Openvpn Di Windows

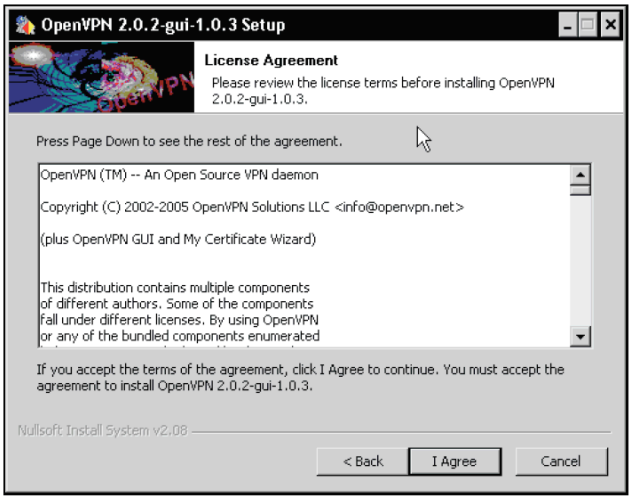
OpenVPN dapat bekerja sebagai suatu *service* pada sistem operasi Windows, yang berarti OpenVPN dimulai secara otomatis pada *startup*. Openvpn dapat dikonfigurasi untuk *enable tunnel* secara otomatis.

Download versi terbaru dari OpenVPN GUI dari http://openvpn.se/ ke drive lokal. Log in sebagai administrator atau *privileged user* dan *double-klik* pada *download* file untuk mulai *setup wizard*. Apabila menggunakan *firewall* akan di-*prompt* untuk mengijinkan OpenVPN diinstal dan dikoneksikan ke internet. Pada Gambar 3.4 sampai Gambar 3.9 menunjukan proses instalasi openvpn pada sistem operasi windows:



Gambar 3.4 *Tampilan Awal Instalasi Openvpn*

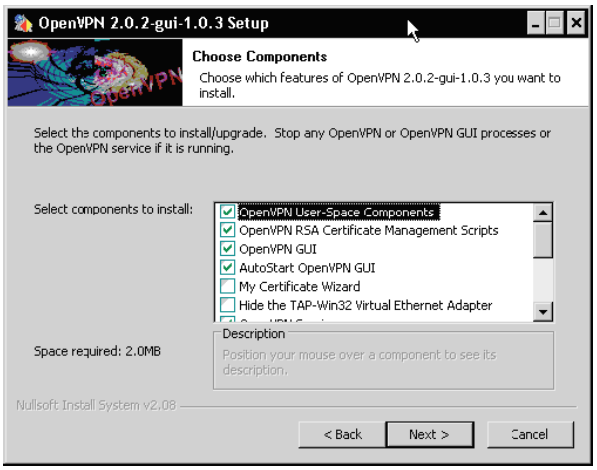
Instalasi wizard OpenVPN GUI. Klik Next untuk memproses.



Gambar 3.5 *Tampilan Agreement dari Openvpn*

Meski OpenVPN dan OpenVPN GUI tersedia lengkap di bawah open source General Public License (GPL), tetap harus menerima *license agreement*. apabila membaca *license*-nya untuk memastikan bahwa rencana untuk menggunakan OpenVPN cocok dengannya. Klik pada I Agree untuk melanjutkan.

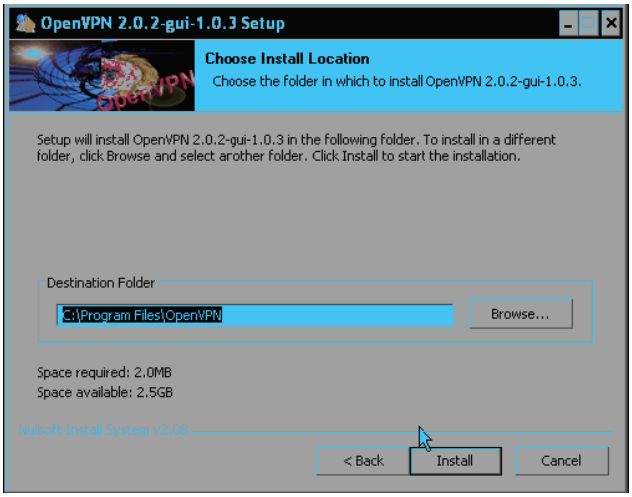
Jendela dialog yang berikutnya menawarkan suatu pilihan di komponen OpenVPN yang dapat diinstal.



Gambar 3.6 *Tampilan Pilihan Komponen Openvpn yang Dapat Diinstal*

Di dalam dialog ini, terdapat beberapa opsi untuk dipilih. Meskipun jika secara normal tidak perlu untuk merubah di sini, tabel yang berikut memberi satu ikhtisar dari masukan-masukan dan ketika perlu menginstal fitur. *Client* Install adalah suatu sistem yang berkoneksi ke sistem OpenVPN lain, sedangkan *Server* Install itu adalah satu sistem OpenVPN yang mengizinkan koneksi-koneksi berikutnya.

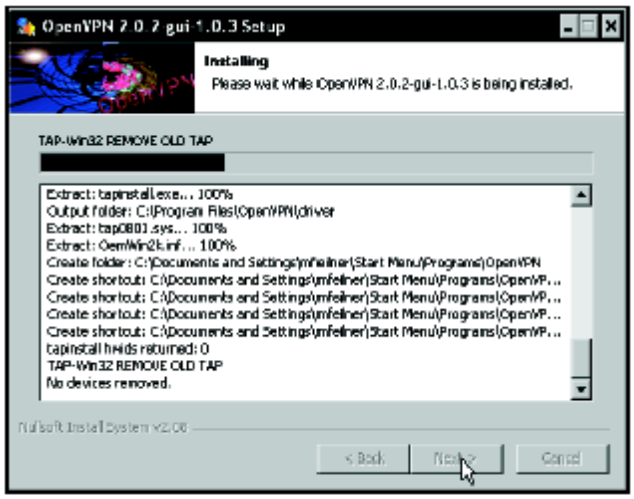
Tekan NEXT untuk melanjutkan instalasi.



Gambar 3.7 *Lokasi Untuk Instalasi Openvpn*

Sekarang pilih satu direktori instalasi untuk OpenVPN. Alur standar instalasi dari OpenVPN pada sistem operasi Windows adalah C:\Program Files\OpenVPN. Setelah meng-klik Install, proses instalas dimulai.

Sementara OpenVPN sedang diinstal, anda dapat membaca outputnya di dalam window instalasi dan mengikuti pembuatan folder, file, dan shortcut dan instalasi dari driver (TAP) untuk networking.



Gambar 3.8 *Proses Instalasi Openvpn*

Proses instalasi telah selesai.



Gambar 3.9 *Proses Instalasi Openvpn Selesai*

3.2.2.2 Instalasi TCPDump

Untuk menginstall TCPdump pada komputer dapat dilakukan dengan mendownload paket TCPdump di http://www.tcpdump.org. TCPdump membutuhkan libpcap (*packet capture library*) yang juga tersedia di situs TCPdump. Libpcap harus diinstal terlebih dahulu dalam komputer. Atau dengan cara menggunakan perintah *aptitude* di *console* untuk mendapatkan paket TCPdump, dengan cara:

root@herwan:~#apt-get install tcpdump

Untuk menjalankan TCPdump, ketik perintah tcpdump di *console* pada *host* 192.168.0.3 sebagai root. *Output* yang diperlihatkan di bawah ini adalah *output* yang bergulir *non-stop*, terus berganti baris tanpa henti hingga menekan Ctrl+C (^C) untuk menghentikan *utility* ini.

```
# tcpdump
tcpdump: listening on eth0
```

```
05:22:27.216338 burner.ssh > prime.1035:
P3797249897:3797249949(52) ack 2183278948 win 8576 (DF)
[tos 0x10]
```

C*ommand line* berikut ini akan memulai TCPdump dan memperlihatkan frame (*Protocol Data Unit* (PDU) yang memindahkan data dari pengirim ke tujuan di dalam *network*, menurut OSI model, PDU yang berada di *datalink layer* disebut sebagai frame, PDU yang berada di *network layer* disebut paket, dan PDU yang berada di *transport layer* disebut segmen) yang mengandung IP *address* 192.168.0.2

```
# tcpdump host 192.168.0.2
tcpdump: listening on eth0
19:16:04.817889 arp who-has tssoss tell prime
19:16:04.818025 arp reply tssoss is-at 0:a0:c9:20:5b:fe
19:16:04.818182 prime.1219 > tssoss.telnet:
S2506660519:2506660519(0) win 16384 <mss
1460,nop,nop,sackOK> (DF)
```

Perintah berikut ini hanya akan mengeluarkan *output* pada frame yang mengandung spesifik IP *address* dan spesifik *port*. Opsi **-nn** akan membuat *disable* translasi nama dan *port*.

```
# tcpdump -nn host 192.168.0.2 and port 23
tcpdump: listening on eth0
19:20:00.804501 192.168.0.1.1221 > 192.168.0.2.23:
S2565655403:2565655403(0) win 16384 <mss
1460,nop,nop,sackOK> (DF)
```

Jika pada perintah di atas ditambahkan opsi **–t**, *display output* akan ditampilkan tanpa keterangan waktu. Sedangkan penambahan opsi **-e** akan menampilkan **layer** kedua dari informasi *datalink*. Keterangan MAC *address* baik milik mesin pengirim maupun mesin tujuan adalah salah satu contoh dari informasi *datalink*.

```
# tcpdump -nne host 192.168.0.2 and port 23
tcpdump: listening on eth0
```

```
19:30:13.024247 0:5:5d:f4:9e:1f 0:a0:c9:20:5b:fe 0800 62:
192.168.0.1.1223 > 192.168.0.2.23:
S2718633695:2718633695(0) win 16384 <mss
1460,nop,nop,sackOK> (DF)
```