BAB III

LANDASAN TEORI

3.1 Definisi Jaringan Komputer

sebuah sistem yang terdiri atas komputer dan perangkat jaringan lainnya seperti: kabel, *switch*, i, *router*, dll yang bekerja bersama-sama untuk mencapai suatu tujuan yang sama. Tujuan dari jaringan komputer adalah:

- Membagi fungsi sumber daya seperti berbagi pemakaian printer, CPU, RAM, harddisk
- Komunikasi: contohnya surat elektronik, instant messaging, chatting - Akses informasi: contohnya web browsing.

Agar dapat mencapai tujuan yang sama, setiap bagian dari jaringan komputer meminta dan memberikan layanan (service). Pihak yang meminta layanan disebut klien (client) dan yang memberikan layanan disebut pelayan (server). Arsitektur ini disebut dengan sistem client-server, dan digunakan pada hampir seluruh aplikasi jaringan komputer.Adapun klasifikasi jaringan komputer berdasarkan skala antara lain:

- Personal Area Network (PAN)

- Campus Area Network (CAN)

- Local Area Network (LAN)

- Metropolitant Area Network (MAN)

- Wide Area Network (WAN)

- Global Area Network (GAN)

Berdasarkan fungsi : Pada dasarnya setiap jaringan komputer ada yang berfungsi sebagai client dan juga server. Tetapi ada jaringan yang memiliki komputer yang khusus didedikasikan sebagai server sedangkan yang lain sebagai client. Ada juga yang tidak memiliki komputer yang khusus berfungsi sebagai server saja. Karena itu berdasarkan fungsinya maka ada dua jenis jaringan komputer:

- Client-server

  Yaitu jaringan komputer dengan komputer yang didedikasikan khusus sebagai server. Sebuah service/layanan bisa diberikan oleh sebuah komputer atau lebih. Contohnya adalah sebuah domain seperti www.detik.com yang dilayani oleh banyak komputer web server. Atau bisa juga banyak service/layanan yang diberikan oleh satu komputer. Contohnya adalah server jtk.polban.ac.id yang merupakan satu komputer dengan multi service yaitu mail server, web server, file server, database server danlainnya.

- Peer-to-peer

  Yaitu **jaringan komputer** dimana setiap host dapat menjadi server dan juga menjadi client secara bersamaan. Contohnya dalam file sharing antar komputer

di Jaringan Windows Network Neighbourhood ada 5 komputer (kita beri nama A,B,C,D dan E) yang memberi hak akses terhadap file yang dimilikinya. Pada satu saat A mengakses file share dari B bernama data_nilai.xls dan juga memberi akses file soal_uas.doc kepada C. Saat A mengakses file dari B maka A berfungsi sebagai client dan saat A memberi akses file kepada C maka A berfungsi sebagai server. Kedua fungsi itu dilakukan oleh A secara bersamaan maka jaringan seperti ini dinamakan peer to peer.

3.2 Paket data

Paket jaringan atau **network packet** adalah satuan informasi dasar yang dapat ditransmisikan di atas jaringan atau melalui saluran komunikasidigital. Sebuah paket berisi *packet header* yang berisi informasi mengenai protokol tersebut (informasi mengenai jenis, sumber, tujuan, atau informasi lainnya), data yang hendak ditransmisikan yang disebut dengan *data payload*, dan *packet trailer* yang bersifat opsional. Sebuah paket memiliki struktur logis yang dibentuk oleh protokol yang digunakannya. Ukuran setiap paket juga dapat bervariasi, tergantung struktur yang dibentuk oleh arsitektur jaringan yang digunakan. Paket jaringan juga dapat disebut **datagram**, **frame**, atau **cell**.

Jika dilihat dari perspektif model tujuh lapis Open Systems Interconnection (OSI), istilah **packet** dan **frame** memiliki definisi yang jauh berbeda. Sebuah paket merupakan "amplop elektronik" yang mengandung informasi yang dibentuk pada lapisan 3 hingga lapisan 7 dari model tujuh lapis OSI tersebut; sementara sebuah *frame* adalah "amplop elektronik" yang mengandung informasi mengenai paket dan informasi lainnya dari semua lapisan dari tujuh lapisan OSI.

Paket jaringan dibuat saat transmisi dilakukan, dengan menggunakan proses enkapsulasi.

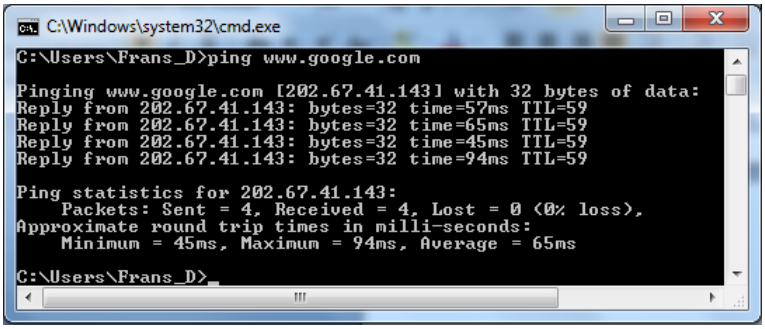
3.3 Perangkat Jaringan

Perangkat jaringan ada beberapa antara lain :

a. Server

Server adalah sebuah penyedia layanan yang dibutuhkan oleh pengguna jaringan komputer. Server biasanya berisi data atau layanan yang digunakan pada sebuah jaringan. Server dapat biasanya berupa PC yang memiliki performa tinggi sehingga dapat mengakomodimir kebutuhan user. Server dapat dibedakan menjadi beberapa jenis. Yaitu server *DNS(Domain Name Service) , WEB , Data Server , Proxy Server , DLl* . Secara fisik alat/ PC / *Tower* yang digunakan sama tetapi pada penggunaannya memiliki perbedaan pada fungsinya. Contoh dan penjelasan beberapa server yang umum digunakan. :

1. *DNS Server* : *DNS Server* adalah server yang bertugas untuk mentranslasikan alamat *website* menjadi alamat *IP* misal ketika kita mengetikan *www.google.com*dan menekan enter maka seketika itu juga alamat itu akan dikirimkan ke DNS server untuk diartikan menjadi alamat *IP* sehingga kita dapat mengakses website tersebut. *IP* dari *www.google.com*adalah :



Gambar 3.1Hasil Ping

Dari situ dapat kita lihat apabila kita me-*PING*www.google.com didapatkan alamat *IP* : 202.67.41.143.

2. *WEBServer* : *WEB server* adalah *server* yang menyediakan layanan *WEB* contohnya adalah www.yahoo.com menggunakan *web server* agar kita dapat melihat tampilannya.

3. *Data Server* : *Data server* adalah server yang bertugas untuk menampung data. Biasanya disebut juga *data center*. *Server* jenis ini biasanya mempunyai kapasitas penyimpanan yang sangat besar karena digunakan untuk menyimpan data maupun untuk backup data dari *server* lain. Ditandai dengan banyaknya harddisk atau media penyimpanan yang di *install* di *server* tersebut. *Data Server* biasanya tidak berdiri sendiri. Diperusahaan yang kecil *Data Server* digabungkan dengan *Web Server* untuk menghemat biaya.

Contoh *Data Server* :

Gambar 3.2*Data Server*

b. Switch

Switch adalah perangkat jaringan yang berfungsi untuk meng-*switch* paket data dari 1 port ke port lainnya yang berada dalam satu jaringan yang sama. 1 jaringan yang sama disini adalah jaringan yang memiliki *Net-Id* yang sama. Contoh 192.168.1.1/24 berada di satu jaringan dengan 192.168.1.129/24 karena *Net-Id* nya mereka berdua sama yaitu 192.168.1.0 yang didapat dari melakukan operasi *And* di kedua *IP* tersebut dengan *Net-Mask* mereka. contoh perangkat *Switch* :

Gambar 3.3 *Switch* Cisco

Switch di atas berisi 48 port yang dapat diisi dengan perangkat jaringan lain atau dengan komputer atau *notebook*. *Switch* tersebut juga memiliki 2 buah port *Gigabit-Ethernet* untuk access yang lebih cepat karena kecepatannya mencapai 10x lipat dari 48 port lainnya.

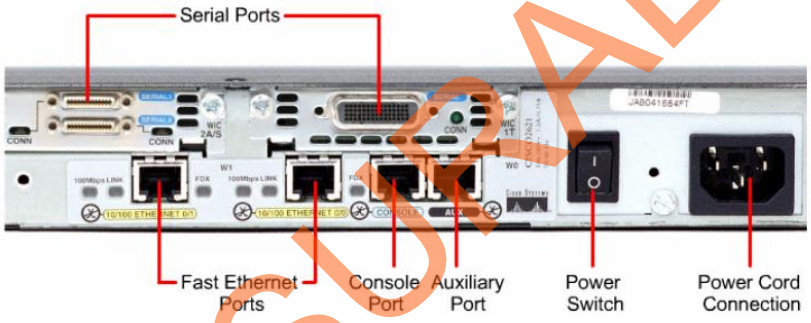
c. Router

Router adalah perangkat jaringan yang digunakan untuk meneruskan paket dari 1 jaringan ke jaringan lain yang berbeda. *Router* yang umum digunakan adalah merk Cisco meskipun ada *router* dengan merk lain.

Contoh *router* :

Gambar 3.4 Contoh Router

Gambar di atas adalah *Router* merk CISCO.



Gambar 3.5 Port yang ada di router

Ini adalah komponen dari *router* meliputi *port* yang ada. Penjelasan port :

- *Serial Port* : *Port* yang digunakan jika kita menggunakan koneksi *serial* dengan kabel *serial*.
- *Fast Ethernet* : *port* yang digunakan untuk menghubungkan router secara langsung dengan switch , PC , maupun *router* lain dengan kabel *RJ-45* (Kabel LAN).
- *Console Port* : *port* yang digunakan untuk menghubungkan router dengan PC dengan kabel *console* untuk kebutuhan setting *Router*.

- *Auxiliary port* : *port* yang digunakan untuk menyambungkan *router* dengan *Modem*.

3.4 Aplikasi yang Digunakan :

a. Nmap

Nmap ("Network Mapper") merupakan sebuah *tool open source* untuk eksplorasi dan audit keamanan jaringan. Ia dirancang untuk memeriksa jaringan besar secara cepat, meskipun ia 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.

Output Nmap adalah sebuah daftar target yang diperiksa, dengan informasi tambahannya tergantung pada opsi yang digunakan. Hal kunci di antara informasi itu adalah "tabel port menarik". Tabel tersebut berisi daftar angka port dan protokol, nama layanan, dan status. Statusnya adalah terbuka (*open*), difilter (*filtered*), tertutup (*closed*), atau tidak difilter (*unfiltered*). Terbuka berarti bahwa aplikasi pada mesin target sedang mendengarkan (*listening*) untuk koneksi/paket pada port tersebut. Difilter

berarti bahwa sebuah *firewall, filter*, atau penghalang jaringan lainnya memblokir *port* sehingga Nmap tidak dapat mengetahui apakah ia terbuka atau tertutup. Tertutup *port* tidak memiliki aplikasi yang sedang mendengarkan, meskipun mereka dapat terbuka kapanpun. Port digolongkan sebagai tidak difilter ketika mereka menanggapi probe Nmap, namun Nmap tidak dapat menentukan apakah mereka terbuka atau tertutup.

Nmap melaporkan kombinasi status *open|filtered* dan *closed|filtered* ketika ia tidak dapat menentukan status manakah yang menggambarkan sebuah port. Tabel *port* mungkin juga menyertakan detil versi software ketika diminta melakukan pemeriksaan versi. Ketika sebuah pemeriksaan protokol IP diminta (-sO), Nmap memberikan informasi pada protokol IP yang didukung alih-alih port-port yang mendengarkan.

Selain tabel port yang menarik, Nmap dapat pula memberikan informasi lebih lanjut tentang target, termasuk nama reverse DNS, prakiraan sistem operasi, jenis device, dan alamat MAC.

memeriksaan Nmap yang umum ditunjukkan dalam Example 1. Argumen yang digunakan pada contoh ini hanyalah -A, untuk memeriksa sistem operasi dan versi, pemeriksaan skrip, dan traceroute; -T4 untuk eksekusi yang lebih cepat; dan dua buah nama host target.

Example 1. Sebuah gambaran pemeriksaan Nmap

 # nmap -A -T4 scanme.nmap.org

Starting Nmap (http://nmap.org)

Interesting ports on scanme.nmap.org (64.13.134.52):

Not shown: 994 filtered ports

PORT STATE SERVICE VERSION

22/tcp open ssh OpenSSH 4.3 (protocol 2.0)

25/tcp closed smtp

53/tcp open domain ISC BIND 9.3.4

70/tcp closed gopher

80/tcp open http Apache httpd 2.2.2 ((Fedora))

|_ HTML title: Go ahead and ScanMe!

113/tcp closed auth

Device type: general purpose

Running: Linux 2.6.X

OS details: Linux 2.6.20-1 (Fedora Core 5)

TRACEROUTE (using port 80/tcp)

HOP RTT ADDRESS

[Cut first seven hops for brevity]

8 10.59 so-4-2-0.mpr3.pao1.us.above.net (64.125.28.142)

9 11.00 metro0.sv.svcolo.com (208.185.168.173)

10 9.93 scanme.nmap.org (64.13.134.52)

Nmap done: 1 IP address (1 host up) scanned in 17.00 seconds

b. PRTG

PRTG (Paessler Router Traphic Grapher) juga merupakan software untuk *monitoring resource network* yang dapat memanfaatkan SNMP (Simple Network Management Protocol), *Packet Sniffing*, *WMI (Windows Management Instrumentation)*, ataupun NetFlow.

Secara garis besar, PRTG dapat digunakan untuk melakukan hal-hal sbb:

- Mengawasi terhadap koneksi *resource-resource* pada jaringan
- Mengawasi dan mengukur penggunaan *bandwith* pada *device-device* jaringan
- Mencari dan menemukan serta mengakses *device-device* yang ada pada jaringan
- Mendeteksi aktifitas yang tidak seharusnya (*suspicious and malicious*) baik dari user ataupun *device* yang ada dalam jaringan
- Mengawasi penggunaan terhadap *resource* sistem, seperti konsumsi **CPU**, penggunaan **memory**, sisa kapasitas **drive** yang tersedia, dll.

- mengelompokkan paket-paket yang lewat pada traffic berdasarkan sumber (*source*) dan tujuannya (*destination*)

Beda dengan **MRTG** yang sifatnya **OpenSource**, PRTG lahir dengan 3 versi, yaitu: **freeware**, **trial version**, dan **enterprise level** (*commercial license*). Untuk mendapatkan *software*-nya, silahkan kunjungi situs PRTG Paessler. Anda juga bisa mendapatkan *serial number* untuk *trial version* selama 30 hari. Perbedaan antara versi-versi yang ada kurang lebihnya sebagai berikut:

- untuk *freeware*, Anda dapat menggunakannya dengan bebas termasuk untuk keperluan *commercial*, tapi hanya sebatas penggunaan untuk 10 sensor, dan interval *monitoring* paling pendek adalah 60 detik (1 menit) untuk update report dari tiap-tiap probe, serta penggunaan sensor hanya terbatas untuk tipe *SNMP, WMI, dan Packet Sniffing* (tidak mendukung *NetFlow*).

- untuk trial version, Anda diberi waktu selama 30 hari untuk menggunakan hingga 500 sensor lebih, dan interval *monitoring* paling pendek 1 detik untuk update report dari tiap-tiap *probe*. Tapi bisa mendukung penggunaan sensor untuk tipe *SNMP, WMI, packet sniffing*, hingga *NetFlow*. Setelah 30 hari,secara otomatis Anda akan diminta untuk memasukkan serial number dari *software* yang dapat Anda peroleh setelah Anda melakukan pembayaran. Jika Anda belum melakukan pembayaran, maka secara otomatis

versi software PRTG Anda akan dialihkan ke mode default, yaitu ***freeware***.

- untuk *commercial edition*, Anda dapat menggunakan semau Anda tentunya mulai dari 100 sensor hingga lebih (mencapai ribuan), tergantung pada versi yang Anda pilih. Juga didukung dengan tipe sensor *SNMP*, *WMI*, *Packet Sniffing*, dan *NetFlow* tentunya.Selain perbedaan dalam hal versi yang tersedia, saat saya mempelajari PRTG ini, PRTG hanya tersedia untuk lingkungan Windows Operating System. Jika Anda tetap ingin menjalankan PRTG ini dalam lingkungan keluarga Unix, seperti Linux misalnya, Anda membutuhkan Wine (Windows Emulator) untuk menjalankannya. **PRTG Network Monitor** dapat dijalankan pada lingkungan Windows XP,2000,2003,2008 Server, dan Vista baik untuk lingkungan 32 bit ataupun 64 bit.Untuk menjalankan interface aplikasi yang berbasis web, dibutuhkan Internet Explorer versi 7.x atau Mozilla Firefox 2/3. Untuk **PRTG System Tray** (yang berjalan sebagai *windows service*), dapat dijalankan hampir disemua lingkungan sistem operasi Windows.

Kebutuhan Hardware pada dasarnya sangat bergantung pada tipe sensor yang nantinya digunakan. Tapi secara umum, berikut penjelasan global untuk kebutuhan hardware untuk menjalankan **PRTG Network Monitoring** :

- ***CPU***, kebanyakan *CPU* saat ini sudah bisa digunakan untuk menjalankan PRTG dengan 1000 sensor, tapi itupun juga tergantung pada jenis/tipe sensor yang digunakan.
- ***Memory***, rata-rata dibutuhkan 150 KB dari memori untuk tiap sensor.
- ***Disk Space***, rata-rata dibutuhkan 200 KB dari sisa ruang kosong pada disk untuk tiap sensor per hari (untuk *monitoring* dengan interval waktu 60 detik/1 menit).

Untuk device yang di-*monitor*, memerlukan hal-hal sebagai berikut:

- Untuk sensor jenis SNMP, maka pada probe/device yang di-*monitor* harus dilengkapi dengan software yang *compatible* dengan *SNMP*, dan **PRTG Core Server** harus bisa mengakses *SNMP* dari i yang dimaksud.
- Untuk sensor jenis ***WMI*** (Windows Management Instrumentation) dibutuhkan arsitektur jaringan Windows.
- Untuk sensor jenis ***NetFlow***, maka *probe* yang bersangkutan harus dikonfigurasi sedemikian rupa untuk mampu mengirimkan paket data NetFlow (NetFlow versi 5) ke **Core Server**.

Arsitektur PRTG terdiri atas 2 bagian, yaitu :

PRTG Core Server

PRTG *Core Server* merupakan bagian utama dari PRTG. PRTG *Core Server* berisi antara lain:

- Konfigurasi objek/probe yang dimonitor
- *Data storage* untuk menyimpan hasil dari monitoring
- *Report engine* dan *scheduler*
- *Mail server* untuk notifikasi via *e-mail*
- *Web Server* yang mendukung http (default port:80) maupun https (SSL, default port:443)

PRTG Core Server ini akan mengelola satu atau lebih Probe yang terkoneksi dengannya.

PRTG Probe Component

Jika kita mengacu pada arsitektur *SNMP*, maka **Probe** ini bisa diibaratkan seperti **Agent**, sementara ***Core Server*** dapat diibaratkan seperti **Manager** (*managed station*).Secara garis besarnya, ada 2 jenis **PRTG Probe**, yaitu *local* probe dan *remote* probe.

Local probe secara otomatis akan dibuatkan saat kita meng-install PRTG. Sedangkan untuk remote probe, kita harus menambahkannyasendiri.

Pada PRTG semua proses *monitoring* akan dijalankan oleh yang namanya sensor. Sensor ini akan dijalankan pada tiap-tiap probe secara independent. **Probe** akan mengambil konfigurasi untuknya dari **PRTG Core Server** untuk kemudian melakukan proses *monitoring* sesuai dengan konfigurasi yang didapat secara independent. Secara independent, maksudnya, jika suatu saat koneksi antara **Probe** dengan **Core Server** terputus, probe tetap dapat bekerja sendiri. Dan jika nantinya koneksi tersedia kembali, maka probe akan mengirimkan hasil *monitoring*-nya ke **Core Server** untuk kemudian **Core Server** melakukan proses *update* terhadap *data storage* **Probe** yang bersangkutan. Ke-2 komponen ini (**Core Server** dan **Probe**) akan bekerja sebagai service (*daemon*) pada **Windows Operating System**. **PRTG** secara otomatis akan melakukan *monitoring* terhadap kesehatan system dan resource jaringan. Untuk keperluan tersebut, **PRTG** secara *default* akan membuatkan beberapa sensor. Dari beberapa sensor tersebut yang sangat penting dan perlu kita perhatikan adalah **"Probe Health"**. Sensor ini merupakan semacam *summary* dari beberapa sensor yang ada. Sebisa mungkin dipertahankan agar nilai dari sensor **Probe Health** ini selalu berada pada nilai 100%.

Gambar 3.6 Local Probe