

BAB III

LANDASAN TEORI

3.1 Definisi Jaringan Komputer

Menurut Sutedjo (2006) Jaringan komputer adalah sekelompok komputer otonom yang saling menggunakan protokol komunikasi melalui media komunikasi sehingga dapat berbagi data, informasi, program aplikasi, dan perangkat keras seperti *printer*, *scanner*, *CD-drive* ataupun *hardisk*. Serta memungkinkan berkomunikasi secara elektronik. Sejumlah potensi jaringan komputer antara lain:

1. Mengintegrasikan dan berbagi pakai peralatan

Jaringan komputer memungkinkan penggunaan bersama peralatan komputer berbagai merek, yang semula tersebar di berbagai ruangan, unit, dan departemen sehingga meningkatkan efektivitas dari penggunaan sumber daya tersebut.

2. Komunikasi

Jaringan komputer memungkinkan terjadinya komunikasi antar pemakai komputer. Selain itu tersedia aplikasi *teleconference* yang memungkinkan dilakukannya rapat atau pertemuan tanpa harus meninggalkan meja kerja.

3. Mengintegrasikan data

Jaringan komputer diperlukan untuk mengintegrasikan data antar komputer-komputer *client* sehingga dapat diperoleh suatu jaringan relevan.

4. Perlindungan data dan informasi

Jaringan komputer memudahkan upaya perlindungan data yang terpusat pada server, melalui pengaturan hak akses dari para pemakai serta penerapan sistem *password*.

5. Sistem terdistribusi

Jaringan komputer dimanfaatkan pula untuk mendistribusikan proses dan aplikasi sehingga dapat mengurangi terjadinya *bottleneck* atau tumpukan pekerjaan pada suatu bagian.

6. Keterangan aliran informasi

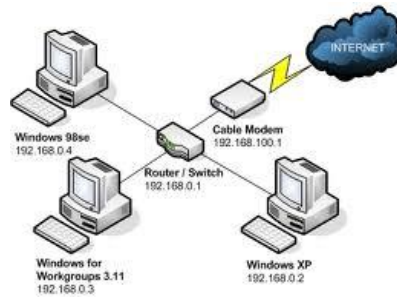
Jaringan komputer mampu mengalirkan data-data komputer *client* dengan cepat untuk di integrasikan dalam komputer server. Selain itu, jaringan mampu untuk mendistribusikan informasi secara kontinu keada pihak-pihak terkait yang membutuhkannya.

3.1.1 Jenis Jaringan

Jaringan komputer terdiri atas lima jenis yaitu *Local Area Network* (LAN), *Metropolitan Area Network* (MAN), *Wide Area Network* (WAN), *Wireless*. Berikut penjelasan dibawah ini:

1. Local Area Network (LAN)

Jaringan yang dibatasi oleh area yang relatif kecil, umumnya dibatasi oleh area lingkungan seperti kantor pada sebuah gedung atau tiap-tiap ruangan pada sebuah sekolah.



Sumber: www.networkingtipsblog.com

Gambar 3.1 *Local Area Network*

2. Metropolitan Area Network (MAN)

MAN, biasanya meliputi area yang lebih besar dari LAN, misalnya antar wilayah dalam satu propinsi. Dalam hal ini jaringan MAN menghubungkan beberapa buah jaringan-jaringan kecil ke dalam lingkungan area yang lebih besar, sebagai contoh jaringan kantor cabang sebuah bank di dalam sebuah kota besar dihubungkan antara satu dengan lainnya.



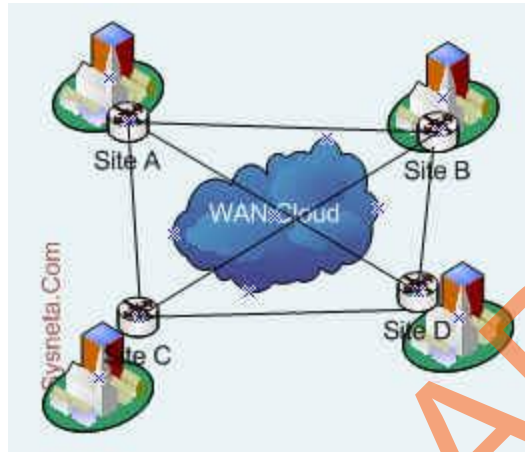
Sumber: www.cyf-kr.edu.pl/en/?a=man

Gambar 3.2 *Metropolitan Area Network*

3. Wide Area Network (WAN)

Sebuah WAN adalah jaringan yang ruang lingkungnya sudah menggunakan sarana satelit, *wireless*, ataupun kabel *fiber* optik WAN memiliki jangkauan yang

lebih luas hingga wilayah otoritas negara lain. Menggunakan sarana WAN, sebuah perusahaan yang ada di Indonesia bisa menghubungi kantor pusatnya yang ada di Jepang hanya dalam waktu beberapa menit.



Sumber: www.sysneta.com/koneksi-wan

Gambar 3.3 *Wide Area Network*

4. *Wireless* (Jaringan tanpa kabel)

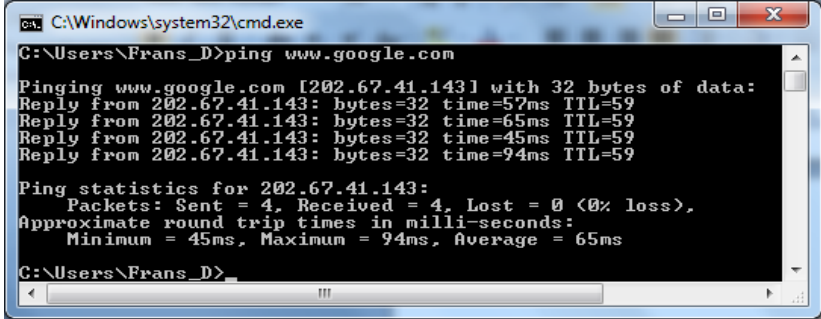
Jaringan tanpa kabel merupakan suatu solusi terhadap komunikasi yang tidak bisa dilakukan dengan jaringan yang menggunakan kabel. Misalnya orang yang ingin mendapat informasi atau melakukan komunikasi walaupun sedang berada di atas mobil atau pesawat terbang, maka mutlak jaringan tanpa kabel diperlukan karena koneksi kabel tidaklah mungkin dibuat di dalam mobil atau pesawat. Saat ini jaringan tanpa kabel sudah marak digunakan dengan memanfaatkan jasa satelit dan mampu memberikan kecepatan akses yang lebih cepat dibandingkan dengan jaringan yang menggunakan kabel.

3.2 Perangkat Jaringan Komputer

3.2.1 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 mengakomodir kebutuhan user. Server dapat dibedakan menjadi beberapa jenis. Yaitu server *DNS(Domain Name Service)* , *WEB* , *Data Server* , *Proxy Server* , *DLI* . Secara fisik alat / PC / *Tower* yang digunakan sama tetapi pada penggunaannya memiliki perbedaan pada fungsinya. Contoh dan penjelasan beberapa server yang umum digunakan. :

- a. *DNS Server* : *DNS Server* adalah server yang bertugas untuk mentranslasikan alamat *website* menjadi alamat *IP* misal ketika kita mengetikkan *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



```

C:\Windows\system32\cmd.exe
C:\Users\Frans_D>ping www.google.com
Pinging www.google.com [202.67.41.143] with 32 bytes of data:
Reply from 202.67.41.143: bytes=32 time=57ms TTL=59
Reply from 202.67.41.143: bytes=32 time=65ms TTL=59
Reply from 202.67.41.143: bytes=32 time=45ms TTL=59
Reply from 202.67.41.143: bytes=32 time=94ms TTL=59

Ping statistics for 202.67.41.143:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 45ms, Maximum = 94ms, Average = 65ms

C:\Users\Frans_D>

```

Gambar 3.4 Hasil Ping

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

- b. *WEB Server* : *WEB server* adalah server yang menyediakan layanan *WEB* contohnya adalah *www.yahoo.com* menggunakan *web server* agar kita dapat melihat tampilannya.
- c. *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. Di perusahaan yang kecil *Data Server* digabungkan dengan *Web Server* untuk menghemat biaya.

Contoh *Data Server* :



Sumber : <http://www.inhabitat.com>

Gambar 3.5 Server

3.2.2 Hub Dan Switch (Konsentrator)

Konsentrator adalah perangkat lunak untuk kabel-kabel jaringan dari tiap *workstation*, server, atau perangkat lainnya. Konsentrator biasa dipakai pada topologi *star*. *Hub* dan *switch* umumnya mempunyai *port* RJ-45 sebagai tempat menghubungkan komputer.

Perbedaannya, *switch* merupakan konsentrator yang memiliki kemampuan manajemen trafik data lebih baik dibandingkan *hub*.



Sumber: nuchaini.blogspot.com/2011/05/peralatan-jaringan.html

Gambar 3.6 Konsentrator

3.2.3 Bridge

Fungsi dari perangkat ini hampir sama dengan fungsi *repeater*, tetapi *bridge* mampu menghubungkan antar jaringan yang menggunakan transmisi berbeda. Misalnya, jaringan *ethernet baseband* dengan *Ethernet broadband*.

Bridge dapat pula menghubungkan jaringan yang menggunakan tipe kabel yang berbeda ataupun topologi yang berbeda.

3.2.4 Router

Router merupakan perangkat yang dikhususkan untuk menangani koneksi antara dua atau lebih jaringan yang terhubung melalui packet *switching*. *Router* bekerja dengan melihat alamat asal dan alamat tujuan yang melewatinya dan memutuskan *route* yang akan dilewati paket tersebut untuk sampai ke tujuan. *Router* mengetahui alamat masing-masing komputer di lingkungan jaringan lokalnya, mengetahui alamat *bridge*, dan *router* lainnya.



Sumber: ajiswild.blogspot.com/2012/05

Gambar 3.7 Router

3.2.5 IP Address

IP address dibentuk oleh sekumpulan bilangan biner sepanjang 32 *bit*, yang dibagi atas 4 bagian. Setiap bagian panjangnya 8 *bit*. *IP address* merupakan identifikasi setiap *host* pada jaringan internet. Artinya tidak boleh ada *host* lain (yang tergabung ke internet) menggunakan *IP address* yang sama, seperti 193.160.5.1.

Network ID			Host ID
193	160	5	1

Tabel 3.1 Tabel bagian dari IP Address

Bentuk penulisan *IP address* diatas dikenal dengan notasi “*dotted decimal*”. Dalam praktiknya, *IP address* bentuk desimal inilah yang digunakan sebagai alamat *host*.

Saat ini alokasi *IP address* versi 4 sudah semakin berkurang. IPv4 sudah digunakan hampir 20 tahun. Untuk mengatasinya, telah dikembangkan *IP address* versi 6. Ipv4 menggunakan 32 bit. Sehingga kurang lebih terdapat 4 milyar komputer yang dapat terhubung ke internet menggunakan IPv4.

Kemudian, *IP address* yang berjumlah 4 milyar ini tidak semuanya dapat digunakan sebagai *IP address* untuk *host*. Ada yang digunakan untuk keperluan khusus. Seperti untuk keperluan alamat network, alamat broadcast, alamat *localhost*, LAN dan sebagainya. *IP address* yang digunakan untuk keperluan LAN/intranet disebut sebagai *IP address private*. Sedangkan *IP address* yang digunakan untuk keperluan internet disebut *IP address public*.

KELAS	IP ADDRESS
A	10.0.0.0 – 10.255.255.255
B	172.16.0.0 – 172.255.255.255
C	192.168.0.0 – 192.168.255.255

Tabel 3.2 *IP Address Private*

Secara umum, *IP address* dapat dibagi menjadi 5 buah kelas. Kelas A, B, C, D dan E. Namun dalam praktiknya hanya kelas A, B, dan C yang dipakai untuk keperluan umum. Ketiga kelas *IP address* ini disebut *IP address unicast*. *IP address* kelas D dan E digunakan untuk keperluan khusus. *IP address* kelas D disebut juga *IP address multicast*. Sedangkan *IP address* kelas E digunakan untuk keperluan riset.

3.3 Paket data

Paket jaringan atau *network packet* adalah satuan informasi dasar yang dapat ditransmisikan di atas jaringan atau melalui saluran komunikasi digital. 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 *packet* 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 *packet* dan informasi lainnya dari semua lapisan dari tujuh lapisan OSI.

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

3.4 Aplikasi yang Digunakan :

3.4.1 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 detail 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.

memeriksa 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.pa01.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

```

3.4.2 PRTG

PRTG (Paessler Router Traffic 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 semua 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 menambahkannya sendiri.

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 membuat 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%.