BAB II

LANDASAN TEORI

2.1 *Network Protocol Analyzer*

Jaringan protokol analisis adalah proses untuk sebuah program atau perangkat untuk memecahkan kode *header* protokol jaringan dan trailer untuk memahami data dan informasi di dalam paket di enkapsulasi oleh protokol. Untuk melakukan analisis protokol, paket harus ditangkap pada *real time* untuk analisis jalur kecepatan atau analisis nanti. Program atau perangkat disebut *Analyzer Protocol*.

Dalam arsitektur jaringan yang khas, pendekatan berlapis yang digunakan untuk merancang protokol jaringan dan komunikasi. Jaringan yang paling populer arsitektur model referensi disebut model *OSI*. Protokol pada satu lapisan harus berkomunikasi dengan protokol pada lapisan yang sama. Fungsi utama dari *Network Protocol Analyzer* adalah untuk *decode* protokol di setiap lapisan. Protokol Informasi dari beberapa lapisan dapat digunakan oleh *Network Protocol Analyzer* untuk mengidentifikasi kemungkinan masalah dalam komunikasi jaringan, yang disebut Ahli Analisis. Analisa protokol dapat men-*decode* protokol lapisan ganda dan paket untuk kembali membangun paket tingkat yang lebih rendah (seperti tingkat *Link, IP* atau *TCP*) ke tingkat yang lebih tinggi (seperti tingkat aplikasi) pesan untuk pemahaman mendalam tentang lalu lintas jaringan dan aktivitas pengguna. Teknik ini digunakan dalam analisa protokol ketika lalu lintas jaringan pemantauan dan pengawasan pengguna adalah tujuan utama.

Network Protocol Analyzer dapat digunakan baik untuk manajemen jaringan yang sah atau untuk mencuri informasi dari jaringan. Jaringan operasi dan personil pemeliharaan menggunakan *Network Protocol Analyzer* untuk memonitor lalu lintas jaringan, menganalisis paket, menonton pemanfaatan sumber daya jaringan, melakukan analisis forensik dari pelanggaran keamanan jaringan dan memecahkan masalah jaringan. Analisa protokol yang tidak sah bisa sangat berbahaya bagi keamanan jaringan karena mereka hampir mustahil untuk mendeteksi dan dapat dimasukkan hampir di mana saja.

2.2 *Network Attack*

Network attacks sendiri jika dikategorikan menurut letak dapat dibagi menjadi dua yaitu *network attacks* yang berasal dari dalam *network* itu sendiri dan *network attacks* yang berasal dari luar network. Sedangkan bentuk *network attacks* dapat berasal dari sebuah host dan dapat juga berupa sebuah *device*/perangkat keras yang berhubungan dengan target, sebagai contoh kasus *wiretapping*. Yang menjadi sasaran atau target dari sebuah *attacks* dapat berupa *host* maupun *network* itu sendiri. Jika diasumsikan bahwa pengamanan terhadap infrastruktur dari sebuah *network* telah dilakukan, maka yang perlu diwaspadai adalah serangan dari luar *network*, dimana hanya proteksi saja yang dapat diandalkan untuk menghindari bahaya dari *network attacks* yang berasal dari luar. Untuk mengetahui bagaimana cara untuk memproteksi sebuah *network* dari *attacks* yang berasal dari luar *network* maka ada baiknya mengetahui apa yang menjadi motivasi adanya sebuah *attacks*.

2.3 *Security Attack Models*

*Interruptio*n: Perangkat sistem menjadi rusak atau tidak tersedia. Serangan ditujukan kepada ketersediaan *(availabilit*y) dari sistem.Contoh serangan adalah "*denial of service attack*".

Interception: Pihak yang tidak berwenang berhasil mengakses asset atau informasi. Contoh dari serangan ini adalah penyadapan *(wiretapping).*

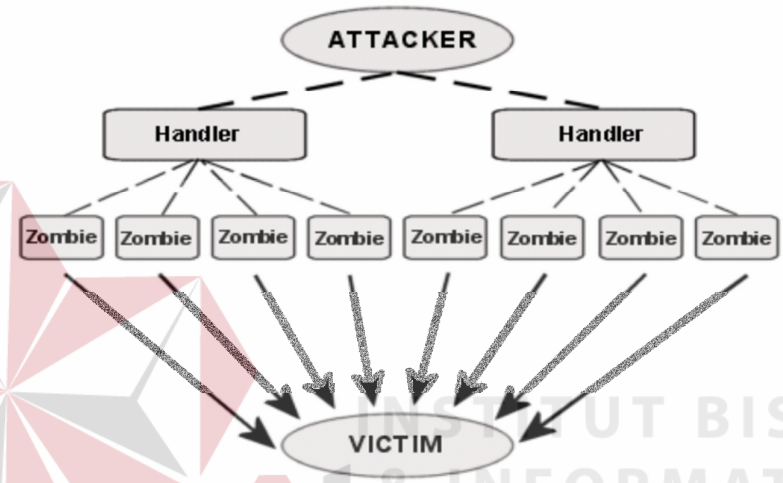
Modification: Pihak yang tidak berwenang tidak saja berhasil mengakses, akan tetapi dapat juga mengubah (tamper) aset. Contoh dari serangan ini antara lain adalah mengubah isi dari *website* dengan pesan-pesan yang merugikan pemilik *website.*

Fabrication: Pihak yang tidak berwenang menyisipkan objek palsu ke dalam sistem. Contoh dari serangan jenis ini adalah memasukkan pesan-pesan palsu seperti e-mail palsu ke dalam jaringan komputer.

2.4 Proses *Attack*

Serangan atau *attacks* pada sebuah *network* biasanya mempunyai proses atau tahap atau fase yang harus dilalui. Disini kami memberikan tiga buah fase yang dilalui oleh *attackers*. Fase pertama adalah fase persiapan. Dalam fase persiapan, attacker akan mengumpulkan informasi sebanyak mungkin mengenai target yang menjadi sasaran mereka. Fase kedua adalah fase eksekusi, fase ini merupakan *attack* yang sebenarnya dimana *attacker* melangsungkan *attack* pada sebuah sistem.Antara fase pertama dan fase kedua terkadang ditemui kasus dimana saat fase pertama berlangsung, berlangsung juga fase kedua. Contoh *scanning* untuk mendapatkan informasi pada sebuah *host* sama dengan *attack* pada *network* yang melingkupinya. Fase ketiga adalah fase akhir yang kami sebut

dengan fase *post-attack*. Fase ketiga merupakan fase akibat dari fase pertama dan fase kedua.Bisa jadi terjadinya kerusakan pada sebuah *network*, atau dikuasainya sebuah sistem *network* yang kemudian digunakan kembali oleh untuk melakukan serangan pada sistem *network* lainnya (Bayu Krisna). Untuk melihat proses serangan ini dapat dilihat pada Gambar 2.1.



Gambar 2.1. Attack Dalam Topologi Jaringan (Garg & Chawla, 2011)

2.5 *Denial of Service Attack*

Denial of Service atau yang mungkin lebih sering kita dengar dengan nama *DoS* merupakan suatu aktifitas yang menghambat laju kerja dari sebuah layanan atau malah mematikannya sehingga yang dapat menyebabkan pengguna yang asli/sah/memiliki hak akses tidak dapat menggunakan layanan. Dimana pada akhirnya, serangan ini mengakibatkan terhambatnya aktifitas yang akan dilakukan oleh korban yang akibatnya boleh dibilang sangat fatal.

DoS merupakan serangan yang cukup menakutkan di dunia internet karena akibat dari serangan ini *server* akan mati dan tidak dapat beroperasi lagi sehingga

otomatis tidak dapat meberikan pelayanan lagi. *DoS* memiliki beberapa jenis serangan, diantaranya adalah

1. *SYN Attack*

   Dalam proses pengiriman data yang melalui *TCP*, proses data yang terjadi adalah sebagai berikut : Hubungan *TCP* dimulai dengan mengirimkan paket *SYN-TCP* ke *host* yang dituju, pengiriman paket *SYN* adalah merupakan pembuka untuk membuka jalur koneksi antara dua *host* melalui protokol *TCP*. Apabila hubungan tersebut disetujui *host* tujuan akan mengirimkan paket *SYN-ACK* sebagai tanda bahwa jalur sudah terbentuk. Dan bagian terakhir adalah pengiriman paket *ACK* dari *host* awal ke host tujuan sebagai konfirmasi. Sedangkan *flood* SYN terjadi bila suatu *host* hanya mengirimkan paket *SYN TCP* saja secara kontinyu tanpa mengirimkan paket *ACK* sebagai konfirmasinya. Hal ini akan menyebabkan *host* tujuan akan terus menunggu paket tersebut dengan menyimpannya ke dalam *backlog*. Meskipun besar paket kecil, tetapi apabila pengiriman *SYN* tersebut terus menerus akan memperbesar *backlog*. Hal yang terjadi apabila *backlog* sudah besar akan mengakibatkan *host* tujuan akan otomatis menolak semua paket *SYN* yang datang, sehingga *host* tersebut tidak bisa dikoneksi oleh *host-host* yang lain (Pratama, 2010).

2. *PING Flood*

   PING Flooding adalah merupakan aktivitas serangan *DoS* sederhana, dilakukan oleh penyerang dengan *bandwidth* yang lebih baik dari korban, sehingga mesin korban tidak dapat mengirimkan paket data ke dalam jaringan (*network*), hal ini terjadi karena mesin korban dibanjiri.

3. *UDP Attack*

UDP adalah *protocol* yang tidak memerlukan koneksi terlebih dahulu dan tidak perlu melalui prosedur pengaturan koneksi untuk transfer data. *UDP Flood Attack* adalah *attacker*akan mengirimkan paket *UDP* ke *port* secara acak secara terus menerus sehingga akan mempengaruhi kerja system yang mengakibatkan sistem bisa hang (duraiswamy & palanivel, 2010).

Selain itu, agar komputer atau mesin yang diserang lumpuh total karena kehabisan *resource* dan pada akhirnya komputer akan menjadi hang, maka dibutuhkan *resource* yang cukup besar untuk seorang penyerang dalam melakukan aksi penyerangannya terhadap sasaran. Berikut ini merupakan beberapa *resource* yang dihabiskan :

1. ***SwapSpace*****,** *Swap spase* biasanya digunakan untuk mem-*forked child* proses.
2. ***Bandwidth*****,** Dalam serangan *DoS*, bukan hal yang aneh bila *bandwith* yang dipakai oleh korban akan dimakan habis.
3. ***Kernel Tables***, Serangan pada *kerneltables*, bisa berakibat sangat buruk pada sistem. Alokasi memori kepada *kernel* juga merupakan target serangan yang sensitif. *Kernel* memiliki *kernel map limit*, jika sistem mencapai posisi ini, maka sistem tidak bisa lagi mengalokasikan memori untuk *kernel* dan sistem harus di *re-boot*.
4. ***RAM*****,** Serangan *Denial of Service* banyak menghabiskan *RAM* sehingga sistem mau-tidak mau harus di *re-boot*.
5. ***Disk*****,** Serangan klasik banyak dilakukan dengan memenuhi *Disk*.

2.6 Statistika

Statistika adalah ilmu yang mempelajari bagaimana merencanakan, mengumpulkan, menganalisis, menginterpretasi, dan mempresentasikan data (Gunawan, 2012).

Penggunaan metode statistik banyak digunakan di dalam pembuatan, pengembangan produk makanan. Perangkat lunak komputer, farmasi dan berbagai bidang lain melibatkan pengumpulan informasi atau data ilmiah. Tentu saja pengumpulan data tersebut bukanlah hal yang baru, hal ini telah dikerjakan dengan selama lebih dari seribu tahun. Data dikumpulkan, dirangkum, dilaporkan dan disimpan untuk diteliti (Ronald E. Walpoe, 2000).

2.7 Distribusi Probabilitas

Kunci aplikasi probabilitas dalam statistik adalah memperkirakan terjadinya peluang/probabilitas yang dihubungkan dengan terjadinya peristiwa tersebut dalam beberapa keadaan.Jika kita mengetahui keseluruhan probabilitas dari kemungkinan outcome yang terjadi, seluruh probabilitas kejadian tersebut akan membentuk suatu distribusi probabilitas.

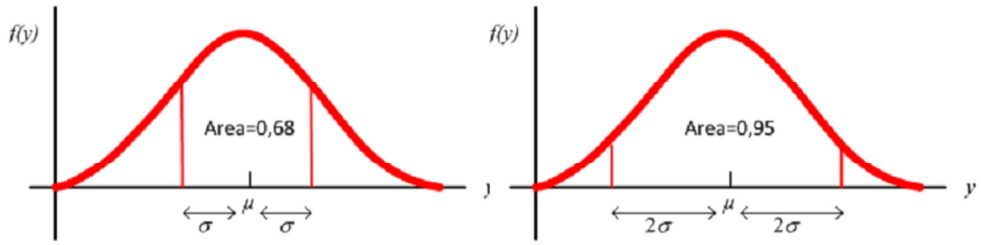
Adapun macam distribusi probabilitas diantaranya adalah :

1. Distribusi Normal (*Gaussian*)

Distribusi normal seringkali disebut sebagai distribusi *Gaussian*. Sifat dari distribusi ini yaitu :

1. Total area di bawah kurva normal jika di jumlahkan adalah 1.
2. Kurva normal adalah simetris terhadap rata-rata (μ).

3. Karena kurva ini berbentuk seperti bel simetris, maka total area yang dibatasi oleh standar *deviasi* (σ) disekitar rata-rata selalu sama. Seperti yang di tunjukkan pada gambar 2.2 :



Gambar 2.2. Kurva Variabel Acak Kontinyu Normal (Harinaldi, 2005)

Rumus *Gaussian* untuk distribusi Normal :

$$\int(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{1}{2\sigma^2}(x-\mu)^2}$$

$-\infty < x < \infty$ $\quad \sigma^2 = 0$

$-\infty < \mu < \infty$ $\quad \pi = 3,14 \quad e = 2,71828$

Ciri khas kurva simetris, seperti lonceng dengan titik belok $\mu \pm \sigma$ serta luas di bawah kurva = *probability* = 1.

Kurva Normal Umum, Untuk dapat menentukan probabilitas di dalam kurva normal umum (untuk suatu sampel yang cukup besar, terutama untuk gejala alam seperti berat badan dan tinggi badan), nilai yang akan dicari ditransformasikan dulu ke nilai kurva normal standar melalui transformasi Z (*deviasi* relatif).

dimana:

$$Z = \frac{x - \mu}{\sigma} \qquad Z = \frac{X - \overline{X}}{S}$$

Kurva normal standar N (μ = 0, σ = 1)

Kurva normal umum N (μ, σ)

Rumus untuk distribusi eksponensial adalah f(x)= $\propto e^{-\propto x}$

4. Variabel Acak Diskrit *Geometri*

Variabel acak diskrit geometri ini dapat digunakan untuk memodelkan jumlah paket dalam antrian sebuah router dengan jumlah memori yang tidak terbatas. Dan untuk nilai $0 \leq p < 1$, terdapat dua macam pmf (*probability mass function*) atau fungsi probabilitas massa dari variable acak diskrit, yaitu :

$$P_{y(k)} = (1-p)p^{k-1}$$

atau

$$P_{y(k)} = (1-p)\,p^{k}$$

5. Distribusi *Gamma*

Distribusi *Gamma* adalah distribusi fungsi padat yang terkenal luas dalam bidang matematika.Fungsi *Gamma*, penamaan *Gamma* (Γ) pada distribusi *Gamma* berasal dari fungsi :

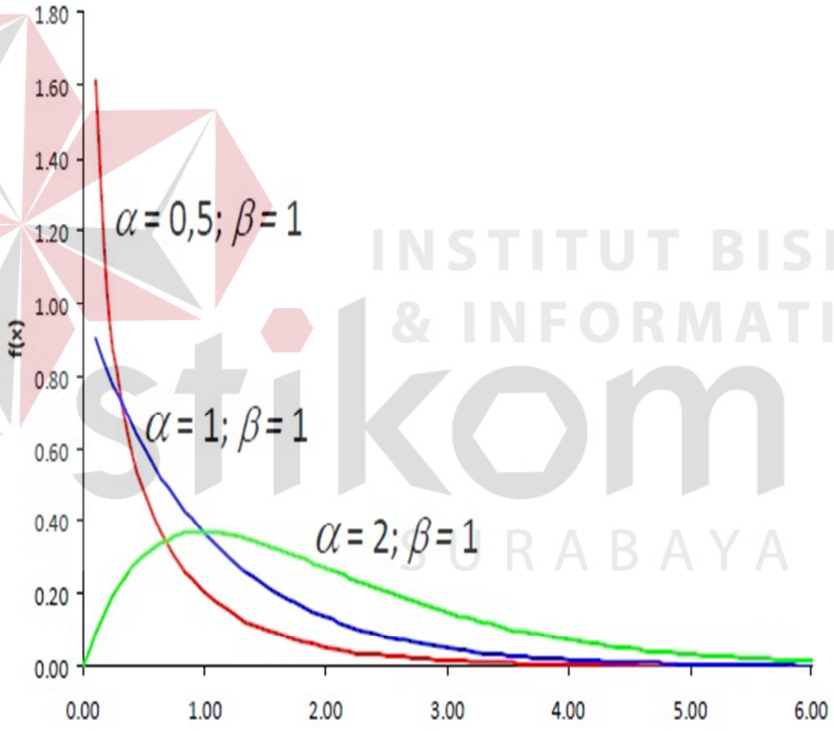
$$\Gamma(\alpha) = \int_0^\infty y^{\alpha-1} e^{-y}\, dy.$$

Fungsi *Gamma* ini adalah fungsi rekursif di mana Γ(n) = (n-1). Perubah acak kontinu X berdistribusi *Gamma* dengan parameter dan jika fungsi padatnya berbentuk:

$$f(x) = \begin{cases} \dfrac{1}{\beta^{\alpha}\Gamma(\alpha)} x^{\alpha-1} e^{-\frac{x}{\beta}} & ; \quad x > 0 \\ 0 & ; x \text{ yanglain} \end{cases} \quad \alpha > 0, \beta > 0$$

X = data

Kurva distribusi *Gamma* dengan parameter α dan β dapat dilihat pada Gambar 2.3.



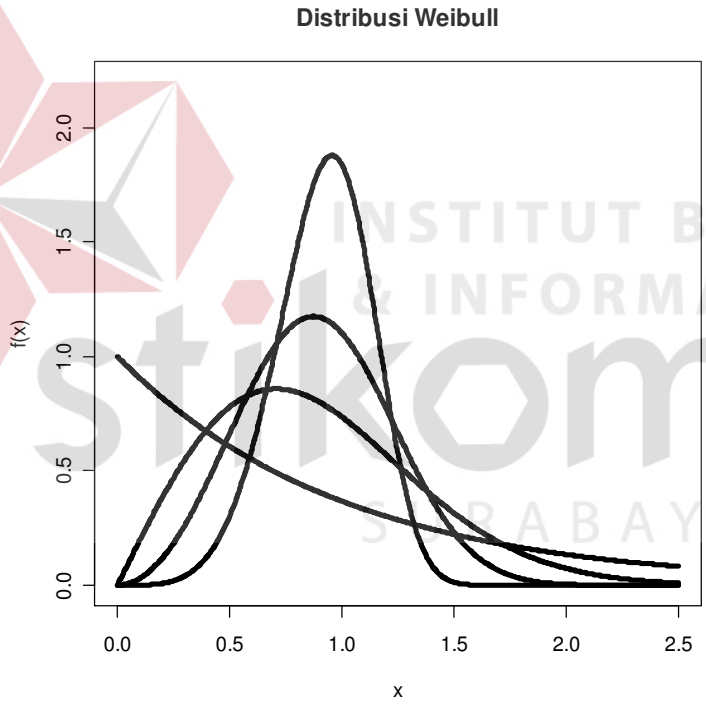
Gambar 2.3. Bentuk Kurva Distribusi *Gamma* (Suprayogi, 2006)

6. Distribusi *Weibull*

Perubah acak kontinu X terdistribusi *Weibull* dengan parameter, jika fungsi padatnya berbentuk:

$$f_W(x;\alpha) = \begin{cases} \dfrac{\alpha}{\beta^{\alpha}} x^{\alpha-1} e^{-(x/\beta)^{\alpha}} & x \geq 0 \\ 0 & yang\ lain \end{cases}$$

Untuk distribusi *Weibull* dapat dilihat pada Gambar 2.4.



Gambar 2.4. Bentuk Kurva Distribusi *Weibull* (Walpole & Myers,1995)

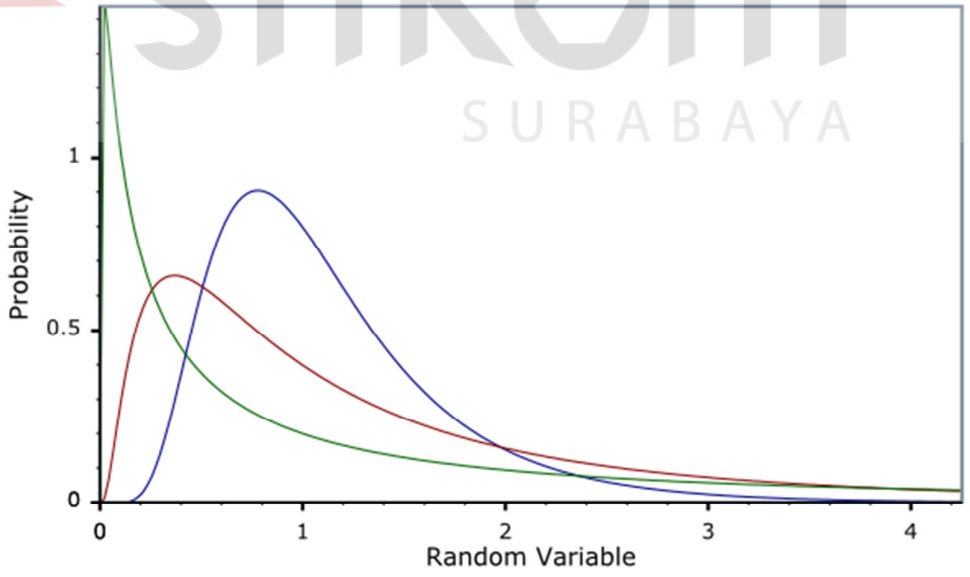
7. Distribusi *Lognormal*

Sebuah variable acak non-negatif X dikatakan memiliki distribusi *Lognormal* jika ln(X) memiliki distribusi normal. Fungsi kepadatan probabilitas dari sebuah variable acak yang memenuhi distribusi *Lognormal* jika ln(x) terdistribusi normal dengan parameter mu dan sigma. Dibawah ini adalah rumus pdf *Lognormal* :

$$y = f(x|\mu, \sigma) = \frac{1}{x\sigma\sqrt{2\pi}} e^{\frac{-(\ln x - \mu)^2}{2\sigma^2}}$$

$x = \text{data}, \mu = \text{rataan}, \sigma = \text{standar deviasi}$

Untuk melihat karakteristik dari distribusi *Lognormal* dapat ditunjukan dalam Gambar 2.5.



Gambar 2.5. *PDF* Distribusi *Lognormal* (Harinaldi, 2005)

2.8 Metode *Sturgess*

Metode *Sturgess* adalah metode untuk menentukan banyaknya kelas interval (Frids, 2002) dengan rumus:

K = 1 + 3, 322 log n

Jangkauan range = Nilai max – nilai min

Jumlah kelas = 1+3.322*log(n)

Interval kelas = Jangkauan range / jumlah kelas

Dengan:

K : Jumlah Kelas

n : Jumlah Data

2.9 Distribusi Frekuensi Relatif

Distribusi frekuensi relatif menyatakan proporsi data yang berada pada suatu kelas interval. Distribusi frekuensi relative pada suatu kelas didapatkan dengan cara membagi frekuensi dengan total data

Sedangkan distribusi frekuensi kumulatif relative dapat didapatkan dengan dua cara. Pertama, kita menjumlahkan frekuensi relatif dari kelas interval pertama sampai kelas interval tersebut. Atau kita bisa mendapatkannya dengan membagi frekuensi kumulatif dengan total data.

Distribusi frekuensi relatif = frekuensi / total data