BAB II

TINJAUAN PUSTAKA DAN DASAR TEORI

2.1 Tinjauan Pustaka

Tinjauan pustaka mengacu pada beberapa penelitian yang terkait dengan keamanan jaringan, yaitu sebagai berikut :

Penelitian pertama adalah penelitian tentang keamanan jaringan menggunakan *Snort* dengan metode *rule-based detection* atau dikenal sebagai *signature-based detection* oleh Bayu Wicaksono. Penelitian ini menggunakan tambahan Iptables yang berfungsi untuk memblokir atau meneruskan paket pada jaringan.

Penelitian kedua adalah penelitian tentang analisis sistem keamanan web server dan database server menggunakan Suricata oleh Nazwita dan Siti Ramadhani. Penelitian ini berfokus pada penanganan ancaman brute force dan pencurian data *password* dari pengguna yang menjadi target.

Penelitian ketiga adalah penelitian tetang Pendeteksian serangan DDoS dengan metode *Intrusion Detection System(IDS)* menggunakan Suricata oleh Moch Fakhri Hidayatuloh. Penelitian ini mengimplementasikan cara mendeteksi serangan terhadap jaringan komputer seperti Ping Of Dead, SYN Attack.

Penelitian keempat dilakukan oleh Lutfi Nur Hakim tentang analisis perbandingan *Intrusion Detection System* Snort dan Suricata. Pada penelitian ini membahas cara kerja, optimasi dan kinerja sistem pendeteksian snort dan suricata dalam menangani ancaman yang terjadi pada infrastruktur jaringan dengan beberapa tipe serangan seperti pemanfaatan celah *port scanning* dan lain-lain.

Penelitian kelima adalah penelitian yang membahas Integritas Suricata Intrusion Detection System (IDS) dan Mikrotik Firewall untuk Keamanan Jaringan dengan studi kasus pada Fakultas Teknologi Informasi Universitas Kristen Satya Wacana oleh Oktriany Susanti Sundun. Penelitian ini melihat integritas antara suricata dengan mikrotik untuk mendeteksi serangan yang berbasis port dan protokol. Aplikasi yang dipakai untuk mendesain sistem ini adalah suricata. Suricata merupakan salah satu software selain snort yang mampu bekerja sebagai IDS, IPS dan monitoring. Suricata akan bekerja sebagai pihak ketiga yang membantu kinerja dari firewall mikrotik selain itu dibutuhkan juga barnyard yang akan membaca file output dari suricata yang bersifat binari dan mengirimkan log file tersebut ke dalam database. Suricata bekerja berdasarkan signature-base, setiap packet akan diperiksa berdasarkan rules yang ada pada suricata. Suricata akan mengeluarkan sebuah output binari yang disebut unified.alert. Barnyard akan bertugas menerjemahkan file output dan juga mengirimkan ke database, lalu mikrotik akan mengambil file dari database tersebut untuk ditindak lanjuti. Selain itu, desain pada penelitian ini membutuhkan penerapan firewall Demilitarized Zone (DMZ) hal ini dilakukan untuk melindungi jaringan LAN dari suatu serangan.

Penelitian keenam adalah penerapan analisis NSM berupa pengumpulan data, deteksi, dan analisis. Penelitian ini dilakukan oleh Chris Sanders dan Jason Smith dengan judul *Applied Network Security Monitoring: Collection, Detection, and Analysis*, membahas definisi keamanan jaringan dari terminologi inti dan asumsi yang akan digunakan oleh banyaknya administrator jaringan.

Penulis	Kasus	Metode	Software
Dwi Kuswanto	Pengamanan Jaringan Komputer	Rule-based detection	Snort
Nazwita dan Siti Ramadhani	Analisis sistem keamanan web server dan database server	Intrusion Detection System (IDS)	Suricata
Moch Fakhri Hidayatuloh	Pendeteksian serangan DDoS	Intrusion Detection System (IDS)	Suricata
Lutfi Nur Hakim	Analisis perbandingan *Intrusion Detection*	Intrusion Detection System (IDS)	Snort, Suricata

	System Snort dan Suricata		
Oktriany Susanti Sundun	Integritas Suricata Intrusion Detection System (IDS) dengan Mikrotik Firewall	Intrusion Detection System (IDS)	Suricata. barnyard
Chris Sanders , Jason Smith	Penerapan terminologi dan asumsi keamanan jaringan	Network Security Monitoring (NSM)	Snort, Suricata, Bro-IDS, SiLK, and Argus

2.2 Dasar Teori

2.2.1 Definisi Intrusion Prevention System (IPS)

Intrusion Prevention System (IPS) adalah sistem yang dapat mendeteksi aktivitas yang mencurigakan pada sistem atau jaringan. Perangkat lunak ini menganalisis semua lalu lintas di firewall mencari serangan dan anomali yang diketahui. *Intrusion Prevention System* dapat memberikan peringatan saat berhasil mendeteksi suatu aktivitas mencurigakan kepada administrator dan kemudian perangkat lunak ini akan melakukan pencegahan secara langsung.

2.2.1.1 Tujuan Penggunaan IPS

IPS (Intrusion Prevention System) merupakan sebuah perangkat lunak yang beroperasi untuk mengidentifikasi dan memblokir ancaman terhadap jaringan dengan menilai setiap paket yang melintasi berdasarkan protokol jaringan pada aplikasi dan kemudian melakukan pelacakan ancaman keamanan jaringan. Sistem IPS sama dengan sistem setup IDS, IPS mampu mencegah beberapa ancaman yang datang dengan sedikit bantuan administrator atau bahkan tidak sama sekali. Serangan biasanya datang dalam bentuk input data berbahaya ke aplikasi target atau melalui layanan yang digunakan penyerang untuk mengganggu dan menguasai aplikasi atau jaringan target. Oleh karena itu IPS akan melakukan suatu tindakan untuk mencegah serangan sebelum terjadi eksekusi dalam memori dan IPS akan membandingkan file checksum yang tidak semestinya mendapatkan izin untuk dieksekusi dan juga menginterupsi sistem panggilan.

2.2.1.2 Jenis-Jenis Intrusion Prevention System

1. Host-based Intrusion Prevention System (HIPS)

Merupakan sebuah sistem pencegahan yang terdiri dari banyak lapisan, menggunakan paket filter, inspeksi status dan juga metode pencegahan yang bersifat real-time untuk menjaga host sistem berada dalam kondisi efisiensi performansi yang layak. Mekanisme kerja HISP yaitu dengan mencegah kode-kode berbahaya yang diinput untuk memasuki sistem agar tidak dieksekusi tanpa perlu mengecek threat signature.

2. Network-based Intrusion Prevention System (NIPS)

Network-based ini dapat menahan semua trafik jaringan dan memeriksa aktivitas dan kode yang mencurigakan. IPS jenis ini menggunakan in-line model, sehingga performansi tinggi merupakan sebuah elemen krusial dari perangkat IPS untuk mencegah kemacetan pada jaringan. NIPS biasanya didesain menggunakan 3 komponen untuk mengakselerasi performa transfer data. NIPS melakukan monitoring dan proteksi dalam satu jaringan secara global. NIPS menggabungkan fitur IPS dengan firewall dan kadang disebut sebagai *In-Line* IDS atau Gateway Intrusion Detection System (GIDS).

2.3 Suricata

Suricata merupakan mesin pendeteksi ancaman jaringan yang baik, cepat, dan tangguh. Mesin Suricata mampu mendeteksi intrusi *realtime* (IDS), pencegahan intrusi *inline* (IPS), pemantauan keamanan jaringan (NSM) dan pemrosesan pcap *offline*.

Suricata memeriksa lalu lintas jaringan menggunakan *rules* yang kuat dan ekstensif dengan kombinasi bahasa *signature*, dan memiliki dukungan *scripting* Lua yang kuat untuk mendeteksi ancaman yang kompleks.

Proyek dan kode Suricata didukung oleh Open Information Security Foundation (OISF), yayasan nirlaba yang berkomitmen untuk memastikan pengembangan dan kesuksesan Suricata sebagai proyek *open source*.

Berikut adalah fitur utama Suricata. Yakni, *Multi Threading, Performance Statistic, Automatic Protocol Detection, Gzip Decompression, Independent HTP Library, Standard Input Methods, Unified2 Output, Flow Variables, Fast IP Matching, HTTP Log module, Graphics Card Acceleration, IP Reputation dan Flowint*

2.4 Jenis Serangan

2.4.1 Denial of Service

Denial of service merupakan jenis serangan terhadap sebuah komputer atau *server* dengan cara menghabiskan *resources* (sumber) yang dimiliki oleh komputer tersebut sampai komputer tersebut tidak dapat menjalankan fungsinya dengan benar sehingga secara tidak langsung mencegah pengguna lain mendapatkan layanan dari *server*/komputer yang diserang tersebut.

Beberapa cara yang dilakukan oleh penyerang dalam melakukan *denial of service*, yakni sebagai berikut:

1. *Traffic flooding* merupakan teknik yang digunakan dengan membanjiri *traffic network* dengan data sehingga *traffic network* yang datang dari pengguna yang terdaftar, tidak dapat masuk ke dalam sistem jaringan.
2. *Request flooding* dilakukan dengan membanjiri jaringan dengan banyak *request* terhadap sebuah layanan jaringan yang disediakan oleh sebuah *host* sehingga *request* yang datang dari pengguna yang terdaftar tidak dapat dilayani oleh layanan tersebut.
3. Mengubah informasi konfigurasi sistem atau bahkan perusakan fisik terhadap komponen dan *server* yang dapat mengganggu komunikasi antara *host* dengan kliennya.

2.4.2 Scanning

Scanning merupakan aktivitas yang dilakukan untuk mendapatkan informasi target. Adapun informasi yang ditemukan oleh penyerang antara lain *IP address*, sistem operasi, arsitektur sistem, *service* running di tiap komputer. *Scanning* dapat dibagi menjadi tiga jenis yaitu:

1. *Port Scanning* merupakan *scanning* yang bertujuan untuk menemukan *port-port* yang terbuka dari suatu *host*.
2. *Network Scanning* merupakan *scanning* yang bertujuan untuk menemukan *host* atau komputer yang aktif pada suatu jaringan.
3. *Vulnerability Scanning* merupakan *scanning* yang bertujuan menemukan kelemahan dari suatu sistem.

Terdapat metodologi ataupun langkah-langkah yang dilakukan dalam melakukan *scanning* antara lain:

a. Discover/ reconnaissance

Reconnaissance dikenal juga dengan sebutan *footprinting*, yang bertujuan untuk mendapatkan informasi awal, seperti alamat IP, DNS *server*, *domain* , tabel *routing*, sistem operasi, dsb. Intinya adalah mendapatkan informasi detail sebanyak-banyaknya sebagai persiapan untuk melakukan langkah selanjutnya. Seluruh informasi tersebut tidak selalu diambil secara diam-diam.

b. Scanning

Setelah mengenali sistem secara keseluruhan, penyerang mulai mencari jalur penyusupan yang lebih spesifik. Jalur tersebut dapat berupa *port*. *Port* yang umum digunakan sistem antara lain *port* 80 untuk HTTP, *port* 21 untuk FTP, *port* 1433 untuk Microsoft SQL *Server*, *port* 3389 untuk terminal *service*, dsb.

c. Enumeration

Langkah selanjutnya yang dilakukan untuk mengambil informasi yang lebih detail. Informasi tersebut dapat berupa *user-user*, *sharing-folder*, *service* yang berjalan termasuk versinya.

d. Penetration

Pada tahap ini, penyerang mengambil alih sistem setelah memperoleh informasi-informasi yang dibutuhkan. Bisa jadi penyerang masuk tidak dengan hak *administrator*, tetapi mampu menyerang *resource* sehingga akhirnya mendapatkan hak akses *administrator*. Dapat dikatakan, jika penyerang sampai masuk ke tahap ini, berarti telah melewati pintu terpenting pertahanan sistem. Terkadang jebolnya pintu keamanan ini diakibatkan oleh kelalaian sistem itu sendiri. Sebagai contohnya adalah penggunaan *password* yang lemah dan mudah ditebak, kesalahan pemrograman yang mengakibatkan terbukanya serangan dari luar. Karena itu, selain melakukan konfigurasi sistem dan jaringan yang baik, pengamanan dari sisi pemrograman juga sangat vital.

e. Elevation

Setelah mampu mengakses sistem, maka penyerang mengubah status *privilege*nya setara dengan *user* yang memiliki hak penuh terhadap sistem.

f. Pilfer

Dengan memperoleh control penuh dari sistem, penyerang leluasa untuk melakukan apa yang dikehendakinya, seperti mengambil data yang baik dalam bentuk *text file*, *database*, dokumen, *e-mail*, dsb.

g. Expansion

Tidak hanya menyusup pada suatu sistem, penyerang dapat memperluas penyusupannya dengan memasuki sistem atau jaringan yang lain. Dalam tahap ini, seorang penyerang melakukan kembali proses *reconnaissance, scanning*, dan *enumeration* dengan target sistem yang lain.

h. Housekeeping

Dengan melakukan proses yang sering disebut dengan *covering track*, penyerang berusaha menghapus jejaknya dengan bersih.

2.4.3 Exploit

Exploit adalah sebuah kode yang digunakan untuk menyerang keamanan komputer secara spesifik. *Exploit* banyak digunakan untuk penentrasi secara legal ataupun ilegal untuk mencari celah kelemahan (*vulnerability*) sistem yang sebagai

target. Bisa juga dikatakan sebuah perangkat lunak yang menyerang kerapuhan keamanan (*security vulnerability*) yang spesifik namun tidak selalu bertujuan untuk melancarkan aksi yang tidak diinginkan. Banyak peneliti keamanan komputer menggunakan *exploit* untuk mendemonstrasikan bahwa suatu sistem memiliki kerapuhan.

Memang ada badan peneliti yang bekerja sama dengan produsen perangkat lunak. Peneliti itu bertugas mencari kerapuhan dari sebuah perangkat lunak dan kalau mereka menemukannya, mereka melaporkan hasil temuan ke produsen agar produsen dapat mengambil tindakan. Meskipun demikian, *exploit* kadang menjadi bagian dari suatu malware yang bertugas menyerang kerapuhan keamanan.