BAB III
METODE PENELITIAN

3.1 ALUR PENELITIAN

Tahapan kerja yang akan dilakukan oleh penulis dijelaskan dalam subbab Alur penelitian. Alur penelitian ini digunakan sebagai pedoman selama melakukan penelitian agar hasil yang diperoleh tidak menyimpang dari tujuan. Tahapan-tahapan kerja tersebut disusun dalam bentuk *flowchart* seperti pada gambar 3.1.

Gambar 3.1 Diagram Alur Penelitian

Gambar 3.1 menjelaskan bahwa penelitian ini diawali dengan studi literatur yaitu mencari berbagai referensi terkait penelitian. Studi literatur dilakukan mengingat pentingnya sebuah studi literatur yang dapat menunjang ilmu baik teori maupun praktiknya dalam penerapan terhadap simulasi yang akan dilakukan. Studi literatur yang dilakukan ini berkaitan dengan tema yang dibuat dan literatur yang berasal dari penelitian sebelumnya dengan tema yang sama, buku-buku, jurnal, artikel, dan informasi dari internet yang memiliki kaitan dengan tema. Hal ini dilakukan untuk membandingkan seberapa beda antara penelitian sebelumnya dengan yang dirancang pada tugas akhir ini.

Pada blok menyiapkan perangkat merupakan tahap untuk mempersiapkan peralatan yang dibutuhkan baik itu perangkat keras (*hardware*) maupun perangkat lunak (*software*). Selanjutnya melakukan tahapan konfigurasi sistem yang terdiri dari beberapa tahapan mulai dari konfigurasi jaringan, server IPS hingga *tools* penyerang. Kemudian dilanjutkan dengan melakukan pengujian berupa penyerangan ke webserver. Pengujian serangan dilakukan dengan 2 skenario. Jika pengujian tidak berhasil maka akan kembali pada proses konfigurasi sistem, dan jika berhasil maka akan dilanjutkan untuk pengambilan hasil data seperti log Suricata serta melakukan analisis juga kesimpulan.

3.2 PERANGKAT YANG DIGUNAKAN

Pada tahap menyiapkan perangkat ini perlu adanya sebuah sistem penunjang alat dan bahan yang sesuai dengan kebutuhan dalam proses melakukan implementasi meliputi *software* dan *hardware*. Berikut adalah spesifikasi perangkat yang akan digunakan:

a. Perangkat Keras (*Hardware*)

   1. *Personal Computer* (PC)

      Pada penelitian ini *personal computer* (pc) digunakan untuk menjalankan router dan server NIPS, melakukan serangan SQL *Injection* serta untuk menjalankan webserver. Spesifikasi PC yang digunakan untuk penelitian dapat dilihat pada tabel 3.1 berikut.

Tabel 3.1 Spesifikasi PC

No.	Jenis	Spesifikasi		
		Operating System (OS)	*Processor*	*RAM*
1.	*Router & Server NIPS*	Linux Ubuntu 20.04	Intel® Core™ i7-7700 CPU @3.60GHz × 8	8GB
2.	*Attacker*	Windows 10	Intel® Core™ i7-7700 CPU @3.60GHz × 8	8GB
3.	*Normal user*	Windows 10	Intel® Core™ i7-7700 CPU @3.60GHz × 8	8GB
4.	*Webserver*	Linux Ubuntu 20.04	Intel® Core™ i3-7020U CPU @ 2.30GHz	4GB

2. Switch

Switch bekerja pada layer 2 yang berfungsi untuk menghubungkan beberapa perangkat komputer agar dapat melakukan pertukaran paket dan meneruskan paket tersebut ke perangkat tujuan.

b. Perangkat Lunak (*Software*)

1. *Suricata*

Pada sistem ini Suricata versi 6.0.4 digunakan sebagai *tool* IPS yang difungsikan dalam mode *inline* (NIPS), dimana semua *traffic* jaringan akan melakukan pantauan dan deteksi secara menyeluruh dengan menggunakan metode pendeteksian *signature-based*. Metode ini bekerja dengan melakukan deteksi serangan kemudian mencocokannya dengan *signature* yang telah ditetapkan.

2. *IPTables*

IPTables dalam sistem ini berfungsi sebagai firewall yang mengatur keluar masuknya paket. Ketika terdapat paket yang masuk akan diteruskan ke

Suricata. Pada Suricata akan dilakukan pencocokan dengan signature yang terdapat dalam file *rules* serta melakukan eksekusi terhadap paket tersebut. Jika terdapat kecocokan antara paket dengan aturan Suricata, maka *IPTables* akan memblokir paket tersebut.

3. SQLMap

SQLMap memungkinkan untuk mendeteksi dan melakukan serangan SQL *Injection* secara otomatis. SQLMap yang dijalankan pada sistem operasi Windows akan berjalan dengan menggunakan *command prompt*.

4. *Httperf*

Httperf merupakan sebuah aplikasi untuk mengukur kinerja webserver. Httperf bekerja dengan menggunakan protokol HTTP baik HTTP/1.0 dan HTTP/1.1. Proses paling dasar yang dilakukan oleh httperf yaitu untuk menghasilkan permintaan HTTP GET yang tetap dan untuk mengukur berapa banyak tanggapan (respon) dari server. Pada penelitian ini httperf digunakan untuk melihat *response time* dari server.

5. *Apache*

Apache merupakan *software* web server *open source* yang berfungsi untuk meningkatkan *user experience* ketika *user* mengakses website. Penelitian ini menggunakan Apache2.4.41 untuk menjadi perantara antara server dengan sisi client.

6. *MySQL*

Pada penelitian ini menggunakan MySQL sebagai sistem pengelola database yang berbasis perintah SQL *(Structured Query Language)*.

7. *DVWA*

Damn Vulnerability Web Application (DVWA) digunakan sebagai objek penyerangan oleh penulis. DVWA merupakan sebuah web server berbasis php/mysql yang dirancang dengan memiliki banyak celah keamanan salah satunya SQL *Injection*.

8. *Web browser*

Web browser digunakan oleh *web client* untuk mengakses halaman web yang telah dibuat. Pada penelitian ini menggunakan *web browser Firefox*.

3.3 KONFIGURASI SISTEM

Pada tahap ini, penulis melakukan konfigurasi komponen yang dibutuhkan untuk melakukan penelitian. Adapun tahapan konfigurasi yang diperlukan antara lain:

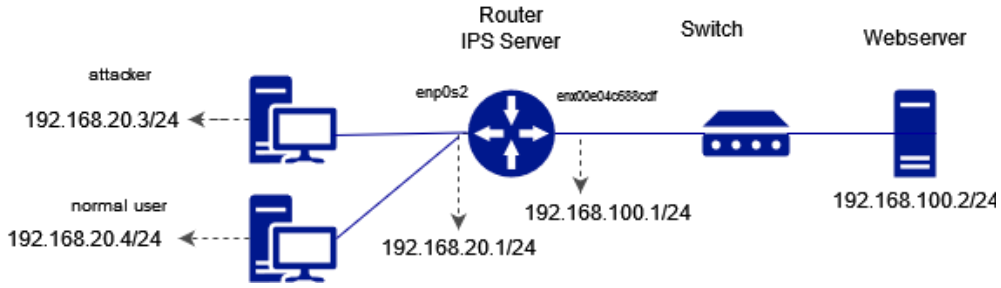
a. Konfigurasi Server IPS
   1. Melakukan instalasi Suricata
   2. Melakukan konfigurasi Suricata
   3. Melakukan konfigurasi *rules*
   4. Konfigurasi *IPTables*
   5. Melakukan setting jaringan

b. Konfigurasi Komponen Pendukung
   1. Konfigurasi *tool* SQLMap pada PC yang berperan sebagai *attacker*.
   2. Konfigurasi webserver untuk pengujian sistem.

3.4 TAHAPAN KONFIGURASI DAN PENGUJIAN

3.4.1 Konfigurasi Jaringan

Gambar 3.3 menjelaskan bahwa penelitian ini menggunakan jenis topologi LAN dengan alamat IP di–*setting static* menggunakan *network address* 192.168.100.0/24. Konfigurasi jaringan dilakukan pada sebuah PC yang difungsikan sebagai *router* dengan ketentuan server IPS menggunakan alamat IP 192.168.100.1/24, dan 192.168.100.2/24 sebagai IP *address* dari webserver.



Gambar 3.2 Konfigurasi Jaringan

3.4.2 Konfigurasi Server IPS

Penulis melakukan konfigurasi Suricata yang berfungsi sebagai IPS meliputi beberapa tahapan yaitu tahap instalasi, konfigurasi pada file `suricata.yaml`, konfigurasi *rules* dan mengaktifkan Suricata.

a. Instalasi suricata

Instalasi Suricata diawali dengan mengunduh file Suricata pada web resmi dan kemudian di-*install* dengan menggunakan perintah `apt-get install suricata -y`. Suricata pada penelitian ini akan difungsikan sebagai IPS, maka diperlukan beberapa paket tambahan agar IPS dapat berjalan dengan baik. Paket tambahan tersebut dapat di-*install* dengan menggunakan perintah seperti berikut.

```
apt-get install libnetfilter-queue-dev libnetfilter-queue1 libnetfilter-log-dev libnetfilter-log1 libnfnetlink-dev libnfnetlink0 -y
```

b. Konfigurasi file `suricata.yaml`

Suricata menggunakan format .yaml sebagai konfigurasi yang terletak pada direktori `/etc/suricata/`. File `suricata.yaml` merupakan file konfigurasi utama dimana didalam file tersebut terdapat pengaturan seperti menentukan *network address* yang akan digunakan, menentukan *interfaces* yang akan dipindai oleh Suricata, letak file *rules* yang akan diterapkan, dan menentukan *output file log*.

c. Konfigurasi file *rules*

File konfigurasi *rules* berisikan *rules* yang akan diterapkan pada Suricata. Penyimpanan file ini memiliki direktori *default* yang berada di `/etc/suricata/rules/`. Sebuah *rules* pada Suricata terdiri dari tiga bagian utama yaitu *action, header* dan *rule option*. Bagian *action* merupakan bagian yang berfungsi untuk menentukan tindakan yang diperlukan ketika *signature* cocok dengan paket. Beberapa *action* yang dapat digunakan yaitu *alert, pass, drop,* dan *reject*. Selanjutnya bagian *header*, pada bagian ini dibagi menjadi beberapa bagian yaitu *protocol*, *source and destination*, *ports (source and destination),* dan *direction*. Protokol yang mendeklarasikan protokol apa yang menjadi fokus Suricata. Protokol dasar terdiri dari tcp, udp, icmp, dan ip, sedangkan untuk protokol pada layer 7 dapat menggunakan http, ftp, smb, dns, ssh, dan lain sebagainya. Setelah protokol terdapat *source and destination* yang

mendeklarasikan tentang sumber dan tujuan paket. Bagian *ports (source and destination)* berarti *port* yang akan menjadi perhatian Suricata. Kemudian pada bagian *direction* merupakan bagian yang menentukan ke arah mana *signature* dicocokkan. Pada umumnya setiap *signature* memiliki arah ke kanan (->), namun terdapat kemungkinan untuk memiliki aturan yang cocok dua arah (< >). Bagian *rule* yang terakhir yaitu *rules option* yang diapit oleh tanda kurung () dan dipisahkan oleh *semicolons* (;). Pada bagian ini umumnya berisi seperti pesan (*msg*), *keyword*, sig, dan rev.

d. Konfigurasi file *output*

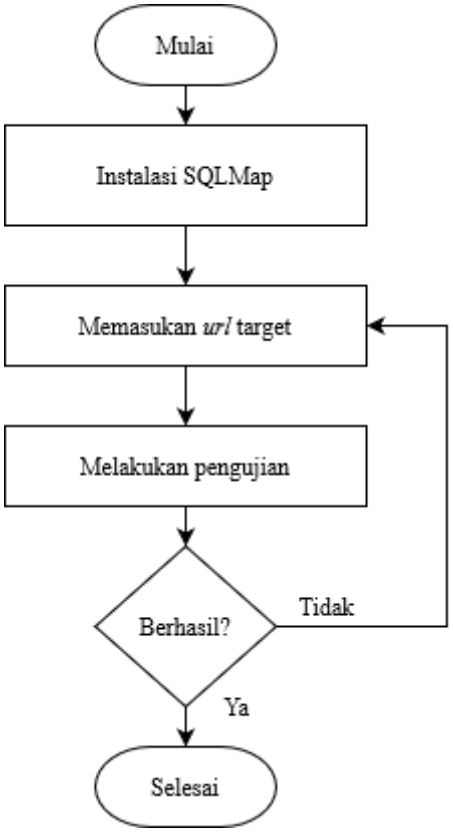
File *output* (*alert* dan *events*) Suricata secara *default* tersimpan pada direktori `/var/log/suricata/`. Direktori penyimpanan file *output* dapat diubah dengan menggunakan *command* `-l` atau juga dapat mengubah konfigurasi didalam file `suricata.yaml`.Namun pada penelitian ini, penulis menggunakan direktori *default* sebagai direktori penyimpanan file *output*.

e. Konfigurasi *IPTables*

IPTables pada penelitian ini menggunakan sebuah *tool* yaitu NFQUEUE, dimana *tool* tersebut nantinya yang akan mengirim paket ke Suricata. *IPTables* bertugas sebagai firewall akan melakukan penyaringan paket *(packet filtering)* yang masuk ke *interface* `enxd03745243baf` (LAN) kemudian paket tersebut diarahkan ke NFQUEUE dan dikirimkan ke Suricata untuk dicocokan dengan *signature rules* yang telah dibuat. Ketika paket tersebut sesuai dengan *signature* maka akan di-*drop*.

3.4.3 Konfigurasi *Attacker*

Pada tahap ini, penulis melakukan konfigurasi *tools* yang akan digunakan untuk melakukan penyerangan. Penyerangan dilakukan dengan menggunakan *tools* yaitu menggunakan SQLMap yang di-*install* pada sistem operasi Windows 10. Ketika melakukan penyerangan *SQL Injection* dengan SQLMap memerlukan alamat *url* target dan parameter pada url tersebut. Pada penelitian ini *url* berbentuk alamat IP dari *webserver*. Diagram alur konfigurasi penyerang ditunjukkan pada gambar 3.4.



Gambar 3.3 Konfigurasi *attacker*

3.4.4 Skenario Pengujian

Skenario pengujian pada penelitian ini berkaitan dengan ketersediaan *database* saat akan diakses oleh *user*. Skenario pengujian dibagi menjadi dua. Skenario pertama adalah kondisi dimana belum diterapkannya Suricata ke dalam sistem dan pada skenario kedua merupakan kondisi dimana sistem Suricata sudah diterapkan. Kedua skenario ini akan menggunakan dua *user,* dimana satu *user* sebagai *attacker* dan *user* lainnya sebagai *user* biasa. *Tool attacker* yang digunakan untuk dua skenario adalah SQLMap. Pengujian kinerja Suricata berdasarkan parameter *response time* untuk dilakukan ketika Suricata sedang aktif. Pengujian ini untuk melihat waktu yang diperlukan oleh Suricata untuk menangani sebuah serangan yang masuk. Pengujian dilakukan sebanyak 30 kali. Penulis menggunakan *httperf* sebagai *tool* pengujian untuk memperoleh nilai *response time*.

Pengujian *rules* dilakukan pada kondisi saat terjadi penyerangan dan ketika Suricata aktif. Pengujian ini bertujuan untuk mengetahui efektivitas dari *rules* yang diterapkan dalam menangani serangan SQL *Injection*. Efektivitas *rules* diukur dari

bagaimana *rules* tersebut bekerja dalam mendeteksi dan melakukan *blocking* terhadap serangan SQL *Injection*.

3.5 PARAMETER KEBERHASILAN SISTEM

Penelitian ini akan mengujikan jenis serangan SQL *Injection* yang termasuk kedalam parameter *Confidentiality*. Termasuk *Confidentiality* karena berkaitan dengan keamanan informasi agar tidak dapat dilihat oleh pihak lain, ketika serangan SQL *Injection* terjadi menyebabkan seorang *attacker* dapat melihat informasi yang terdapat dalam sebuah *database* dari sebuah *website* seperti jenis database sampai *username* dan *password*.

Tabel 3.2 Jenis Serangan

Jenis Serangan	**Parameter**	***Testing tool***
SQL Injection	*Confidentiality*	SQLMap

Tingkat keberhasilan pada sistem ini akan dinilai berdasarkan efisiensi *rules*, ini berarti apakah *rules* yang telah dibuat mampu menangkap dan mencegah ancaman serangan dengan tepat. Keberhasilan sistem juga akan diukur dari parameter *response time* dari sisi *user*. Tabel 3.3 dan 3.4 berikut ini merupakan tabel dari skenario pengujian.

Tabel 3.3 Pengujian *Rules* IPS

Rule	***Blind sqli***	***Error sqli***	***Union sqli***
1	Y/N	Y/N	Y/N
2	Y/N	Y/N	Y/N
3	Y/N	Y/N	Y/N

Tabel 3.4 Pengujian *Response time* saat terjadi penyerangan

Pengujian ke-	***Response time (ms)*** **Suricata aktif**
Pengujian 1	
Pengujian 2	
Pengujian 3	
Pengujian 4	
Pengujian 5	

Pengujian ke-	***Response time (ms)*** **Suricata aktif**
Pengujian 6	
Pengujian 7	
Pengujian 8	
Pengujian 9	
Pengujian 10	
Pengujian 11	
Pengujian 12	
Pengujian 13	
Pengujian 14	
Pengujian 15	
Pengujian 16	
Pengujian 17	
Pengujian 18	
Pengujian 19	
Pengujian 20	
Pengujian 21	
Pengujian 22	
Pengujian 23	
Pengujian 24	
Pengujian 25	
Pengujian 26	
Pengujian 27	
Pengujian 28	
Pengujian 29	
Pengujian 30	