JURUSAN TEKNIK INFORMATIKA

FAKULTAS TEKNOLOGI INFORMASI

INSTITUT TEKNLOGI SEPULUH NOPEMBER

USULAN TUGAS AKHIR

1. IDENTITAS PENGUSUL

NAMA	**:**	**Fajri Rahmat**
NRP	**:**	**5110100123**
DOSEN WALI	**:**	**Bilqis Amaliah, S.Kom, M.Kom**
DOSEN PEMBIMBING	**:**	**1. Ary Mazharuddin Shiddiqi, S.Kom, M.Comp.Sc**
		2. Hudan Studiawan, S.Kom, M.Kom

2. JUDUL TUGAS AKHIR

"SISTEM PENDETEKSI DAN PENCEGAH PERETASAN TERHADAP APLIKASI BERBASIS *WEB* DENGAN MENGGUNAKAN *WEB APPLICATION FIREWALL(WAF)*."

3. LATAR BELAKANG

Aksi peretasan web pada saat sekarang ini sudah menjadi topik hangat. Pada waktu lalu, ketika konflik terjadi di Palestina. Simpatisan-simpatisan dari berbagai negara termasuk Indonesia melakukan aksi penyerangan terhadap situs *web* yang berdomain *.il* yang merupakan domain yang merujuk terhadap negara israel. Kasus lainnya yang juga terjadi pada beberap waktu yang lalu terjadi di Indonesia. Situs *web* presiden Republik Indonesia diretas oleh peretas asal jember. Tidak hanya terhenti disitu, situs-situs lainnya seperti situs kemenhan, kemenpora dan masih banyak situs lainnya, mengantri untuk diretas oleh para peretas lokal.

Dari berbagai kasus yang telah terjadi. Permasalahan soal keamanan informasi menjadi topik penting dalam dunia teknologi informasi. Berbagai produk untuk mendeteksi serangan sudah banyak yang dipublikasikan misalnya snort. Akan tetapi,

---

terkadang hal itu tidak bisa mendeteksi kesalahan pemrograman yang dilakukan oleh pemrogram dalam membangun aplikasi *web* mereka. Akibatnya, rata-rata kesalahan pemrograman seperti ini menjadi sebuah celah keamanan terhadap aplikasi *web* yang mereka bangun.

Pada tugas akhir ini akan dibangun suatu sistem sejenis *firewall* yang akan memblokir berbagai bentuk *request* yang diindikasikan sebagai serangan. Dan tentu saja akan melewati *request* normal. Proses penentuan apakah sebuah *request* adalah serangan atau bukan adalah dengan pencocokan *rules* yang sudah dikonfigurasi pada sistem yang akan dibangun nanti.

Hasil yang diharapkan nantinya setelah sistem ini dijalankan pada *server* yang melayani layanan *web* adalah serangan-serangan seperti *SQL Injection, Cross Site Scripting(XSS), Local File Inclusion(LFI)* dan/atau *Remote File Inclusion(RFI), Cross Site Request Forgery(CSRF)* dan berbagai bentuk serangan lainnya yang dapat membahayakan *web* dapat diblokir sebelum diproses oleh *webserver*. Sehingga aplikasi *web* lebih aman walaupun sebenarnya tidak ada keamanan yang mutlak itu. Tapi setidaknya dapat mengurangi usaha peretas dalam meretas *web*.

4. RUMUSAN MASALAH

Adapun rumusan masalah dalam tugas akhir ini adalah sebagai berikut:
1. Bagaimana cara kerja sistem dalam mendeteksi dan mencegah serangan*?*
2. Bagaimana sistem memonitor serangan apa saja yang sudah dan/atau sedang terjadi terhadap aplikasi *web*?

5. BATASAN MASALAH

Adapun batasan masalah dalam tugas akhir ini adalah sebagai berikut:
1. Tugas akhir hanya berlaku untuk aplikasi *web,* bukan untuk sistem keseluruhan. Jadi, serangan seperti *DDOS* tidak akan berlaku.
2. Berbagai bentuk kesalahan *input* seperti seperti memasukkan karakter petik(‘) pada *text input* walaupun itu bukan usaha dalam melakukan serangan akan tetap dicatat sistem pada *file log* sebagai sebuah serangan.

6. TUJUAN PEMBUATAN TUGAS AKHIR

Adapun tujuan dari tugas akhir ini adalah sebagai berikut:
1. Mendeteksi dan mencegah serangan terhadap aplikasi *web*.
2. Mencatat berbagai serangan terhadap aplikasi *web.*

7. MANFAAT TUGAS AKHIR

Adapun manfaat yang didapatkan dari tugas akhir ini adalah sebagai berikut:

1. Berbagai serangan terhadap aplikasi *web* dapat dideteksi dan dicegah sehingga dapat mempermudah kerja dari administrator.
2. Sistem mencatat berbagai bentuk serangan terhadap aplikasi *web* sehingga dapat menjadi bahan pertimbangan dari administrator kedepannya.

8. TINJAUAN PUSTAKA

8.1. Apache HTTP Server

Apache HTTP Server Project, merupakan sebuah proyek yang bertujuan untuk menyediakan sebuah pengamanan, efisiensi dan *extensible server* yang menyediakan layanan HTTP yang sesuai dengan standar HTTP yang ada sekarang dan merupakan sebuah *open source HTTP server* [1].

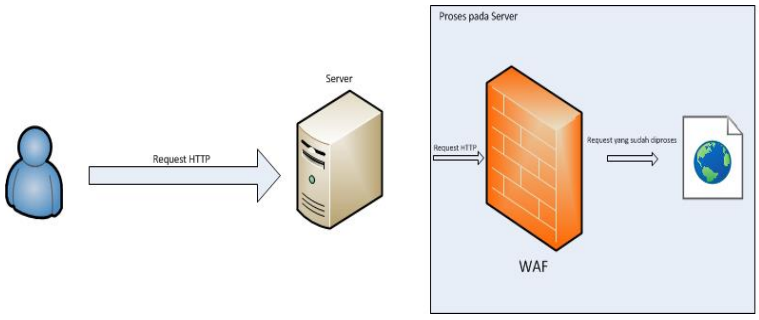
8.2. Ironbee

Ironbee adalah sebuah *framework* dan pustaka untuk inspeksi keamanan aplikasi berbasis *web* atau bisa juga didenifisikan sebagai sebuah pustaka untuk membangun sebuah *WAF*. Berikut adalah komponen-komponen yang menyusun *Ironbee framework:*

1. *Ironbee Library :* merupakan komponen utama dari mesin inspeksi.
2. *Ironbee Modules :* Ekstensi to mesin inspeksi dalam bentuk dari *loadable shared libraries* yang didefinisikan oleh konfigurasi *Ironbee.*
3. Komponen *server* : *Server* yang merupakan *executable* yang menggerakkan mesin inspeksi.
4. Konfigurasi [2].

9. RINGKASAN ISI TUGAS AKHIR

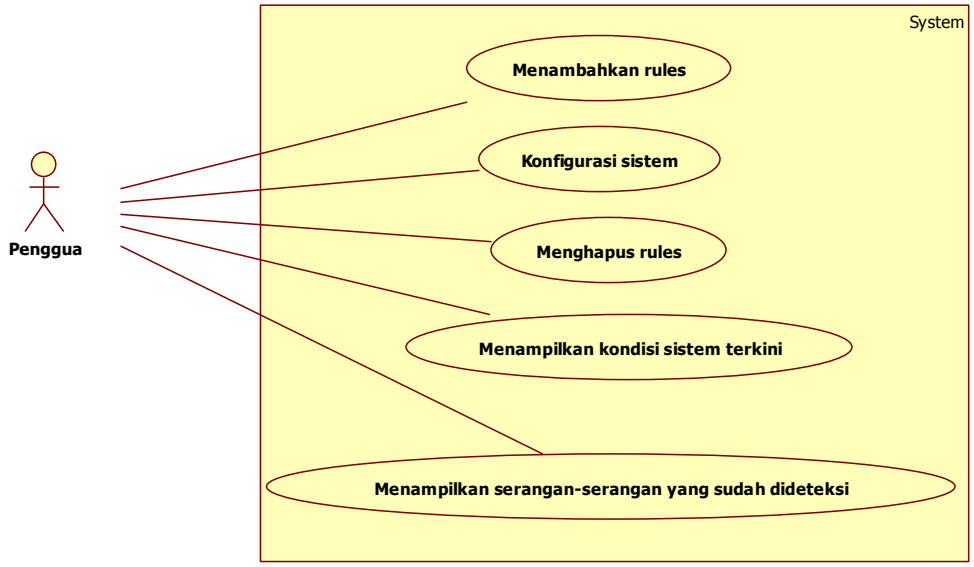
Tugas akhir yang akan dikerjakan nantinya adalah suatu sistem yang menyerupai *firewall,* akan tetapi hanya untuk aplikasi *web*. Gambar 1. Arsitektur sistem WAF



Gambar 1. Arsitektur sistem WAF

merupakan gambaran dari arsitektur sistem pada tugas akhir ini. Pada *WAF* ini, ketika ada pengguna yang melakukan *request* , dan sebelum *webserver* menjawab *request* tersebut maka *request* tadi akan diproses terlebih dahulu oleh *WAF*. Setelah itu, jika pada *request* tadi tidak ada indikasi sebuah serangan seperti tanda petik(‘) maka halaman *request* tadi akan diproses tapi jika terindikasi ada serangan maka *request* tadi akan diblokir.

Sedangkan untuk mempermudah dalam mengkonfigurasi *WAF* ini nantinya, maka akan dibuatkan suatu panel kontrol berbasis *web*. Gambar 2. Use case diagram panel kontrol adalah *use case diagram* dari panel kontrol. Pengguna pada panel kontrol dapat menambahkan dan menghapus *rules* melalui *web* tanpa harus melalui terminal. Begitu juga dengan konfigurasi-konfigurasi lainnya dapat dilakukan dengan menggunakan panel kontrol. Begitu juga pada panel kontrolnya juga terdapat kondisi dari *server* dan *log* dari serangan-serangan yang diambil dari *log file* yang sudah diolah sehingga mudah untuk dibaca dan dipahami.



Gambar 2. Use case diagram panel kontrol

10. METODOLOGI

a. Penyusunan proposal tugas akhir

Proposal tugas akhir berisi tentang tugas akhir yang akan dibuat nanti yaitu suatu sistem pendeteksi dan pencegah peretasan terhadap aplikasi berbasis *web* dengan menggunakan *web application firewall(WAF)*. Pada bagian latar belakang, dijelaskan hal-hal yang melatarbelakangi pembuatan *waf* ini seperti meningkatnya kasus peretasan aplikasi *web* yang terjadi akhir-akhir ini.

Pada bagian rumusan masalah, berisi tentang rumusan-rumusan apa saja yang akan dipecahkan pada tugas akhir ini. Dan pada bagian batasan masalah berisi tentang hal-hal yang menjadi batasan pada proyek tugas akhir ini. Dilanjutkan dengan manfaat dan tujuan dari tugas akhir.

Pada tinjauan pustaka, berisi tentang penjelasan singkat tentang isi pustaka yang akan menjadi literature yang akan digunakan dalam pengerjaan tugas akhir nanti.

Pada ringkasan isi tugas akhir, berisi tentang ringkasan analisa struktur sistem yang akan dibuat nanti. Dan juga terdapat tentang *use case diagram* dari panel kontrol pengguna yang nantinya akan mempermudah siapa saja yang akan menggunakan dan memanfaatkan sistem dalam mengkonfigurasinya.

b. Studi literatur

Adapun studi literatur yang akan dipelajari untuk pengerjaan tugas akhir ini adalah *Ironbee framework.* Karena untuk pembuatan tugas akhir ini *framework* yang akan digunakan adalah *Ironbee.* Selain itu juga akan mendalami bahasa pemrograman **lua** karena modul-modul pada *Ironbee* ini dibuat dengan bahasa pemrograman lua**.**

c. Analisis dan desain

Pada tahap ini akan dilakukan tahap-tahap menganalisa kebutuhan dari sistem seperti proses kerjanya, spesifikasi perangkat keras yang dibutuhkan serta sistem operasi yang digunakan. Setelah kebutuhan untuk pengerjaan tugas akhir ini sudah ditentukan, tahap selanjutnya merancang desain dan infrastruktur dari sistem yang akan dibuat.

d. Implementasi

Untuk mengimplementasi tugas akhir ini, ada beberapa kakas bantu yang dibutuhkan sebagai berikut:

1. Vim, *text editor* dan juga kakas bantu dalam memprogram nantinya.
2. Bahasa pemrograman C/C++
3. Bahasa pemrograman LUA
4. Bahasa pemrograman PHP
5. Ironbee

e. Pengujian dan evaluasi

Untuk menguji hasil dari tugas akhir ini, pada *server* yang sama akan dipasang aplikasi *web* yang memiliki beberapa kerentanan. Lalu akan dilakukan penyerangan terhadap aplikasi *web* tadi. Dengan begitu akan dilihat, apakah

serangan tersebut dapat dideteksi atau tidak. Dan juga akan dilakukan memasukkan karakter-karakter biasa pada *input text*. Lalu akan dilihat bagaimana hasilnya.

Dari hasil yang didapatkan, akan dilakukan beberapa evaluasi sehingga kedepannya dapat mengurangi terjadi hasil yang tidak diinginkan seperti *false positive*.

f. Penyusunan Buku Tugas Akhir

Pada tahap ini dilakukan penyusunan laporan yang menjelaskan dasar teori dan metode yang digunakan dalam tugas akhir ini serta hasil dari implementasi aplikasi perangkat lunak yang telah dibuat. Sistematika penulisan buku tugas akhir secara garis besar antara lain:

1. Pendahuluan
    a. Latar Belakang
    b. Rumusan Masalah
    c. Batasan Tugas Akhir
    d. Tujuan
    e. Metodologi
    f. Sistematika Penulisan
2. Tinjauan Pustaka
3. Desain dan Implementasi
4. Pengujian dan Evaluasi
5. Kesimpulan dan Saran
6. Daftar Pustaka

11. JADWAL KEGIATAN

Tahapan	2013												2014								
	Oktober				November				Desember				Januari					Februari			
Penyusunan Proposal																					
Studi Literatur																					
Perancangan sistem																					
Implementasi																					
Pengujian dan evaluasi																					
Penyusunan buku																					

12. DAFTAR PUSTAKA

[1] S. F. Apache, "Apache HTTP Server Projecy," 22 Juli 2013. [Online]. Available: http://httpd.apache.org/.

[2] B. Rectanus and I. Ristic, Ironbee Reference Manual, Redwood: Qualys, Inc, 2013.

[3] R. C. Barnett, Web Application Defender's Cookbook, Indianapolis: Wiley Publishing, Inc., 2013.