PERBANDINGAN PENERAPAN METODE PENGAMANAN MOD SECURITY DAN MOD EVASIVE PADA *WEB SERVER* TERHADAP SERANGAN SLOW HEADERS

Panca Putra Pahlawan[1)], Faruk Ulum[2)]

Informatika, Fakultas Teknik dan Ilmu Komputer
Universitas Teknokrat Indonesia
Jl. ZA. Pagar Alam No.9 -11, Labuhan Ratu, Kec. Kedaton, Kota Bandar Lampung
Email : rectapanca@gmail.com[1)], faruk.ulum@teknokrat.ac.id[2)]

Abstrak

Perkembangan teknologi informasi semakin cepat dengan adanya internet, teknologi ini juga telah merambah ke dunia pendidikan, yaitu dengan adanya sistem pembelajaran daring (SPADA) atau pembelajaran jarak jauh (PJJ), yang diselenggarakan dan diatur dalam Permenristekdikti. Dalam penerapan teknologi tersebut dibutuhkan *Web Server* yang dapat berjalan dengan baik ketika pembelajaran sedang berlangsung. Menjaga keamanan *Web Server* dari berbagai gangguan atau serangan menjadi hal yang penting. Salah satu jenis serangan yang dapat terjadi adalah *DoS Attack* yaitu *Slow Headers.* Untuk mengatasi jenis serangan tersebut ada beberapa macam metode pengamanan yang dapat digunakan, diantaranya yaitu metode *Mod Security* dan metode *Mod Evasive*. Untuk mengetahui metode pengamanan terbaik dalam mengatasi serangan *Slow Headers* penulis melakukan penelitian mengenai pengujian dan analisa perbandingan dari penerapan metode *Mod Security* dan *Mod Evasive* terhadap serangan *Slow Headers* pada pengkondisian (*state*) sistem *Web Server* universitas xyz. Hasil dari penelitian yang telah dilakukan, mengenai penerapan pengamanan yang lebih baik dalam menangani serangan *Slow Headers,* dapat disimpulkan bahwa metode pengamanan yang lebih baik, yang telah diterapkan adalah *Mod Security*. *Mod Security* dapat melakukan pemblokiran permintaan walaupun permintaan yang dilakukan tidak utuh, dan dapat mempertahankan halaman situs pada *Web Server* agar tetap diakses oleh *Client*, sedangkan *Mod Evasive* tidak dapat melakukan hal tersebut.

Kata kunci: SPADA, *Web Server*, *Mod Security*, *Mod Evasive*, *Slow Headers.*

1. Pendahuluan

A. Latar Belakang

Saat ini teknologi internet dan *website* telah merambah ke dunia Pendidikan, yaitu dengan adanya sistem pembelajaran secara daring menggunakan teknologi *website* yang memiliki fitur untuk melakukan kegiatan belajar dan mengajar dengan mudah, karena dapat diakses dimana saja dan kapan saja. Untuk melakukan pemerataan Pendidikan dengan menerapkan teknologi ini, Kemenristek Dikti telah menyampaikan wacana mengenai penyelanggaraan sistem pembelajaran daring (SPADA) atau pembelajaran jarak jauh (PJJ) ini beberapa tahun lalu.

Dari aspek regulasinya, penyelenggaran SPADA atau PJJ ini sudah diatur dalam Permenristekdikti Nomor 51 Tahun 2018 dalam Bab VII tentang pasal 38 ayat 1, bahwa PJJ mempunyai karakteristik: terbuka, belajar mandiri, belajar dimana dan kapan saja, dan berbasis pada teknologi informasi dan komunikasi. (Sakasuti, 2019). Dalam penerapan teknologi tersebut dibutuhkan sebuah *Web Server* yang dapat berjalan dengan baik ketika pembelajaran sedang berlangsung.*Web Server* dirancang untuk dapat melayani beragam jenis data, baik text, hypertext, gambar maupun suara, tapi pada umumnya *Web Server* melayani data dalam bentuk *file* HTML (*Hypertext Mark Up Language*)(Rifqi, 2012).

Mengingat SPADA universitas xyz dapat diakses secara *online,* menjaga keamanan *Web Server* dari berbagai gangguan atau serangan menjadi hal yang penting. Maka dari itu dibuatlah suatu pengondisian sistem yang menyerupai *Web Server* (berikut dengan *Learning Management System* yang digunakan yaitu Moodle) PJJ atau SPADA dari universitas xyz, yang akan digunakan untuk penelitian nantinya.Salah satu jenis serangan adalah *Denial of Service Attack* (*DoS Attack*) yaitu *Slow Headers* (*Slowloris*). Ada berbagai macam metode pengamanan yang dapat digunakan untuk melindungi *Web Server* dari serangan tersebut, diantaranya yaitu metode *Mod Security* dan metode *Mod Evasive*(Suroto, 2018).

B. Rumusan Masalah

Berdasarkan latar belakang yang telah dipaparkan maka didapatkan rumusan masalah sebagai berikut:

1. Bagaimana cara mengetahui metode pengamanan yang terbaik antara Mod Security dan Mod Evasive dalam menangkal serangan *Slow Headers*?
2. Bagaimana hasil pengujian perbandingan dan analisa terhadap pengamanan Mod Security dan Mod Evasive yang akan diterapkan, manakah metode terbaik yang dapat menangkal atau meminimalisir serangan *Slow Headers*?

C. Tujuan Penelitian

Tujuan dari penelitian ini adalah:

1. Untuk melakukan pengujian dan analisa dari perbandingan penerapan metode pengamanan *Mod Security* dan *Mod Evasive* terhadap serangan *Slow HTTP (Slow Headers)* pada sebuah pengondisian sistem *Web Server*.
2. Untuk mengetahui metode terbaik (*Mod Security* dan *Mod Evasive)* dalam menangkal atau meminimalisir dampak serangan *Slow HTTP (Slow Headers)* pada sebuah pengondisian sistem *Web Server*.

D. Manfaat Penelitian

Dari penelitian yang dilakukan, diharapkan akan memberikan manfaat sebagai berikut:

1. Memberikan informasi tentang pengamanan *Web Server* terhadap serangan *Slow HTTP* (*Slow Headers*).
2. Memberikan informasi mengenai penerapan metode pengamanan *Mod Security* dan *Mod Evasive* pada *Web Server.*
3. Hasil penelitian dapat dijadikan salah satu pertimbangan dalam pemilihan metode pengamanan dan dalam melakukan tindak lanjut terkait pengamanan *Web Server* dari *website* SPADA universitas xyz.

2. Landasan Teori

A. Sistem Pembelajaran Daring (SPADA)

Sistem Pembelajaran Daring atau SPADA adalah salah satu program Direktorat Jenderal Pembelajaran dan Kemahasiswaan Kementrian Riset, Teknologi dan Pendidikan Tinggi untuk meningkatkan pemerataan akses terhadap pembelajaran yang bermutu di Perguruan Tinggi. Dengan sistem pembelajaran daringnya, SPADA Indonesia memberikan peluang bagi mahasiswa dari satu perguruan tinggi tertentu untuk dapat mengikuti suatu mata kuliah bermutu tertentu dari perguruan tinggi lain dan hasil belajarnya dapat diakui sama oleh perguruan tinggi dimana mahasiswa tersebut terdaftar(Nulismuh, 2016).

B. Ubuntu

Ubuntu adalah sistem operasi lengkap berbasis Linux, tersedia secara bebas dan mempunyai dukungan baik yang berasal dari komunitas maupun tenaga ahli profesional. Ubuntu cocok digunakan baik untuk *desktop* maupun *server*. Ubuntu menyertakan semua aplikasi standar untuk desktop mulai dari pengolah kata, aplikasi lembar sebar (*spreadsheet*) hingga aplikasi untuk mengakses internet, perangkat lunak untuk *Web Server*, peralatan untuk bahasa pemrograman dan tentu saja beragam permainan(Ubuntu-ID, 2014).

C. Web Server

Web Server atau lebih tepatnya *world wide Web Server* merupakan *server* Internet yang mampu melayani koneksi transfer data dalam protokol HTTP (*Hyper Text Transfer Protocol*) dan menunggu koneksi dari *port* tertentu. *Web Server* bertugas untuk menerima permintaan terhadap dokumen tertentu yang ditulis dalam format URL, kemudian mencari *file* yang sesuai dengan *file* pada sistem, membacanya kemudian mengirimkannya pada *Client* yang memintanya. *Web Server* dirancang untuk dapat melayani beragam jenis data, baik *text*, *hypertext*, gambar maupun suara, tapi pada umumnya *Web Server* melayani data dalam bentuk Bahasa HTML (*Hypertext Mark Up Language*)(Rifqi, 2012).

D. Apache

Apache adalah perangkat lunak *Web Server open-source* yang mendukung sekitar 46% situs di seluruh dunia (Budiman, Sucipto and Dian, 2021). Nama resmi adalah Apache *HTTP Server*, dan dikelola dan dikembangkan oleh Apache *Software Foundation*. Ini memungkinkan pemilik situs web untuk menyajikan konten di *web* - karenanya dinamai "*Web Server*". Ini adalah salah satu *Web Server* tertua dan paling dapat diandalkan, dengan versi pertama dirilis lebih dari 20 tahun yang lalu, pada tahun 1995. Ketika seseorang ingin mengunjungi situs web, mereka memasukkan nama domain ke bilah alamat *browser* mereka. Kemudian, *Web Server* mengirimkan *file* yang diminta dengan bertindak sebagai pengirim pengiriman virtual(Gediminas, 2019).

E. Moodle

Moodle merupakan salah satu paket perangkat lunak untuk membuat suatu pelatihan – pelatihan (pembelajaran) berbasis *web* dan internet yang biasa disebut sebagai *Learning Management System* (LMS). Moodle disediakan secara gratis dan bebas digunakan karena merupakan *software open source* (dibawah lisensi GNU Public)(Ambarita, 2016).

F. DoS

Serangan *denial of service* adalah ancaman keamanan di mana penyerang mengirimkan sejumlah besar permintaan palsu ke *host* atau *server*, sehingga host target menolak akses dari pengguna yang berwenang. Layanan dari host

menjadi tidak tersedia. Oleh karena itu, serangan itu mengganggu ketersediaan sistem. Serangan *denial of service* (DoS) yang bertujuan untuk pengguna yang sah, klien, pelanggan dari berhasil mengakses internet telah menimbulkan tantangan serius bagi keamanan jaringan. Jika serangan *denial of service* diluncurkan dari beberapa komputer, sering disebut Serangan *Distributed Denial of Service* (DDoS)(Suroto, 2018).

G. Slow Headers

Salah satu jenis serangan *DoS* adalah serangan *Slow HTTP* (*Slow Headers*). *Slow HTTP* (*Slow Headers*) mengeksploitasi metode kerja protokol HTTP (terutama pada bagian *Header)*, yang mengharuskan setiap permintaan dari klien diterima sepenuhnya oleh *server* sebelum diproses. Jika permintaan HTTP (*Header)* tidak lengkap, atau jika kecepatan transfer sangat rendah, *server* tetap sibuk menunggu sisa data. Jika *server* menyimpan terlalu banyak sumber daya sibuk, maka ini menciptakan penolakan layanan(Suroto, 2018).

H. Mod Security

Mod Security adalah *firewall* aplikasi *web open-source*. *Mod Security* memungkinkan pemantauan otomatis pada pengamanan aplikasi *web* secara *real-time*. Rangkaian aturan perlindungan memungkinkan *admin* untuk memeriksa lalu lintas HTTP dan secara andal memblokir lalu lintas yang tidak diinginkan (Impe, 2015).

I. Mod Evasive

Mod Evasive adalah modul untuk Apache untuk memberikan tindakan perlindungan jika terjadi serangan HTTP DoS atau DDoS. Ini juga dirancang untuk menjadi alat deteksi dan manajemen jaringan, dan dapat dengan mudah dikonfigurasikan untuk berkomunikasi dengan *ipchains, firewall, router*, dan sebagainya(Mishra and Sourav, 2012).

J. Wireshark

Wireshark merupakan *tool* yang digunakan untuk melakukan analisa paket data jaringan. Wireshark disebut juga *Network packet analyzer* yang berfungsi menangkap paket-paket jaringan dan berusaha untuk menampilkan semua informasi dipaket tersebut sedetail mungkin. Sebenarnya *network packet analyzer* sebagai alat untuk memeriksa apa yang sebenarnya terjadi di dalam jaringan baik kabel maupun wireless (Adriant and Mardianto, 2015).

K. Switchblade

OWASP Switchblade merupakan *Tool* yang digunakan untuk pengujian *Denial of Service* meliputi ketersediaan, kinerja dan perencanaan aplikasi *web* agar lebih proaktif terkait resiko serangan ini (OWASP, 2018).

3. Metode Penelitian

A. Kerangka Penelitian

Untuk tercapainya tujuan akhir dari penelitian ini, penulis melakukan beberapa tahapan umum. Kerangka penelitian dapat dilihat pada gambar 1:

Gambar 1 Kerangka Penelitian

Keterangan:

1. Masalah

Tahapan penelitian ini diawali dengan penentuan masalah penelitian, yaitu mengenai berbagai jenis serangan DDoS yang masih sering terjadi, hal tersebut mendasari mengenai perbandingan metode pengamanan.

2. Pendekatan

Pendekatan yang dimaksud penulis adalah penggunaan metode pengamanan. Metode pengamanan yang dipilih pada penelitian ini, yang akan digunakan untuk menangkal atau meninimalisir serangan DDoS (*Slow Headers*) adalah *Mod Security dan Mod Evasive*.

3. Perancangan

Tahap ini dilakukan perancangan Pengkondisian *(state) Web Server* menggunakan apache pada ubuntu *virtualbox.*

Implementasi

Implementasi yang dilakukan adalah menerapkan pengamanan *Mod Security* dan *Mod Evasive* pada *Web Server* yang telah dirancang dan ditentukan.

4. Pengujian

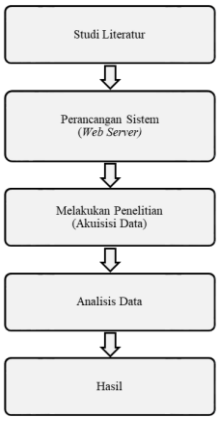
Pengujian dilakukan pada *Web Server* yang telah diterapkan metode pengamanan *Mod Security* dan *Mod Evasive*, mengunakan *tool Switchblade* untuk

mengetahui tingkat ketahanan *Web Server* terhadap serangan *Slow Headers*.

5. Hasil
Hasil dari penelitian ini adalah sebuah informasi yang dapat menjelaskan metode pengamanan terbaik, yang dapat menangkal atau meminimalisir dari serangan *Slow Headers* yang telah dilakukan.

B. Tahapan Penelitian

Tahapan penelitian digunakan sebagai pedoman dalam pelaksanaan penelitian agar hasil yang dicapai tidak menyimpang dari tujuan yang telah dilakukan sebelumnya. Tahapan pada penelitian ini dapat dilihat pada gambar 2 berikut.



Gambar 2 Tahapan Penelitian

Keterangan:

1. Studi Literatur
Pada tahap ini dilakukan kajian literatur dari jurnal, skripsi, dan makalah sebagai bahan pendukung untuk melakukan tahapan penelitian mengenai pengujian dan pengamanan *web server* dari salah satu jenis serangan *Slow HTTP (Slow Headers)* menggunakan *Mod Security* dan *Mod Evasive*.

2. Perancangan Sistem *Web Server*
Pada tahap ini dilakukan perancangan dengan melihat kebutuhan sistem yang digunakan dan juga pembangunan sistem *web server* yang nantinya akan dilakukan pengujian menggunakan serangan *Slow Headers*.

3. Melakukan Penelitian (Akuisisi Data)
Pada tahapan ini dilakukan penelitian dengan cara melakukan pengujian dan pencatatan dari setiap uji coba serangan yang dilakukan dengan kombinasi serangan yang berbeda pada *web server*, yang tidak menerapkan metode pengamanan dan yang menerapkan metode pengamanan (*Mod security* dan *Mod Evasive*).

4. Analisis Data
Data dari hasil pencatatan akan dianalisis secara objektif, dengan cara membandingkan data dari setiap uji coba serangan terhadap ketahanan *web server*, yang tidak menerapkan pengamanan dan yang telah menerapkan pengamanan.

5. Hasil
Hasilnnya adalah kesimpulan akhir yang telah didapatkan dari tahapan analisis data. Data ini berupa grafik yang menjelaskan mengenai metode pengamanan terbaik dalam menangani serangan *Slow Headers.*

C. Implementasi Rules Mod Security

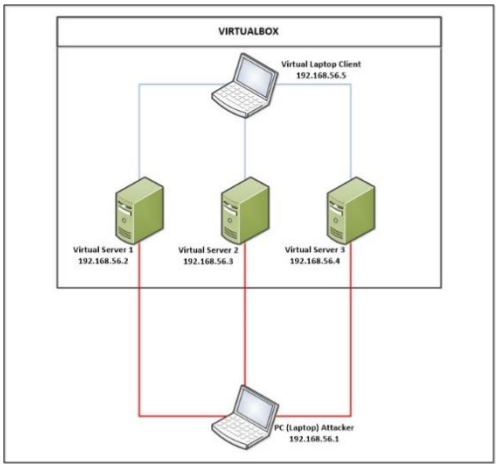
```
SecAction
phase:1,id:122,nolog,pass,initcol:ip=%{REMOTE_A
DDR} SecRule RESPONSE_STATUS "@streq 408"
"phase:5,t:none,nolog,pass,
setvar:ip.slow_dos_counter=+1,
expirevar:ip.slow_dos_counter=15, id:'123'" SecRule
IP:SLOW_DOS_COUNTER "@gt 1"
"phase:1,t:none,log,drop,msg:'Client connection
dropped due to suspected as  an slow http attack',
id:'124'" (Suroto, 2018)
```

D. Implementasi Rules Mod Evasive

```
<IfModule Mod evasive20.c>
   DOSHashTableSize 3097
   DOSPageCount                2
   DOSSiteCount                20
   DOSPageInterval        1
   DOSSiteInterval         1
   DOSBlockingPeriod      60
   DOSLogDir              "/var/log/Mod evasive"
</IfModule>
```

E. Skenario Pengujian

Skenario pengujian akan dilakukan menggunakan 3 *virtual server* seperti yang dijelaskan sebelumnya. Pengujian metode pengamanan pada *Web Server* akan menggunakan aplikasi *denial of service* yaitu switchblade. Untuk topologi pengujian yang akan dilakukan pada *virtual server* dapat dilihat pada gambar 3 berikut.



Gambar 3 Topologi Pengujian Pada Virtual Server (*Web Server*)

Pengujian akan dilakukan dengan melihat dampak dari sisi *virtual Client*, dengan beberapa kombinasi pengaturan serangan yang dilakukan (Park *et al.*, 2014), kombinasi pengaturan serangan pada switchblade dapat dilihat pada tabel 1 berikut.

Tabel 1 Kombinasi Serangan

Slow Headers Attack		
Connections	*Connection Rate*	*Timeout(s)*
300	50	100
300	50	200
600	50	100
300	50	10

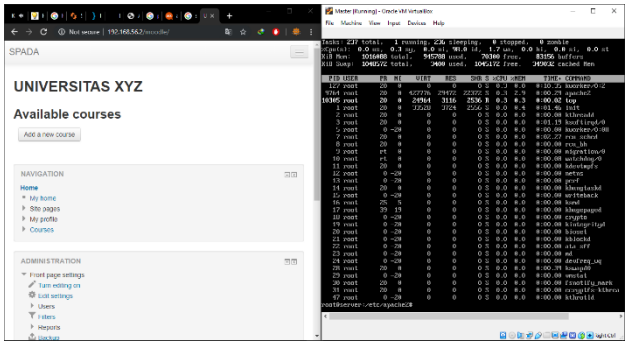
4. Hasil Penelitian dan Pembahasan

A. Hasil Penelitian

Hasil sistem *Web Server* yang telah berhasil dirancang dan dapat berjalan sesuai dengan tujuan dari penulis tanpa ada masalah atau *error* yang terjadi, karena untuk melakukan pengujian yang baik, harus dirancang juga sistem yang dapat berjalan dengan baik pula. Agar penelitian yang dilakukan nantinya tidak mengalami kendala saat melakukan tahap pengujian dan juga tahap pengumpulan data dari hasil pengujian yang telah dilakukan.

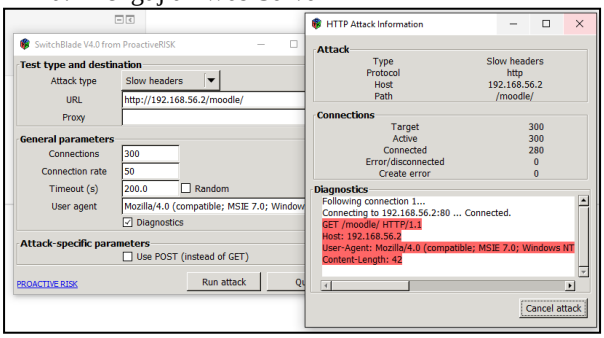
1. Hasil Penelitian *Web Server*

Hasil dari rancangan sistem *Web Server* yang telah dibangun dapat dilihat pada gambar 4 berikut.

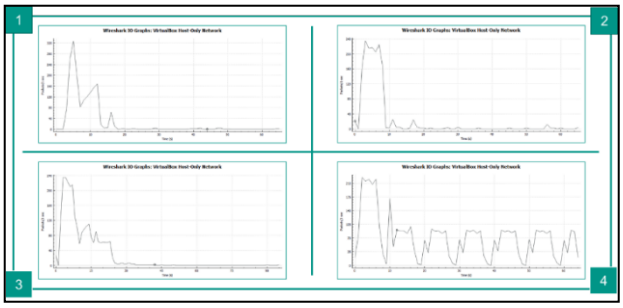


Gambar 4 Hasil Rancangan Sistem *Web Server*

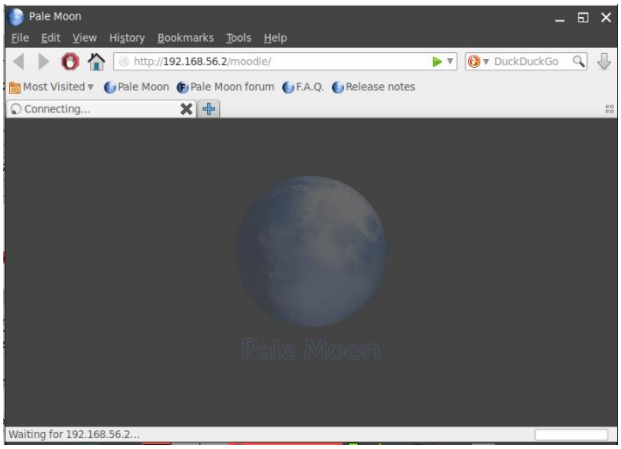
2. Pengujian Sistem *Web Server*
   a. Pengujian *Web Server* 1



Gambar 5 Proses Penyerangan Dengan *Switchblade* (*Web Server 1*)

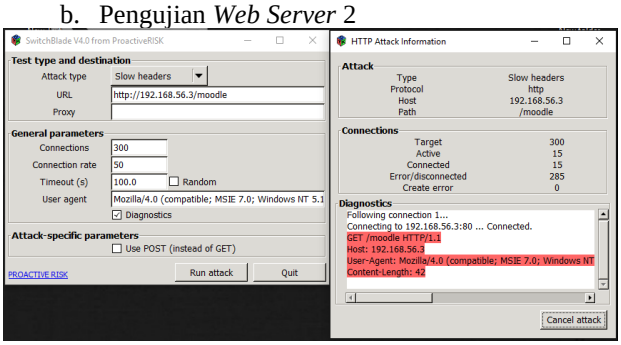


Gambar 6 Pengujian *Web Server* 1

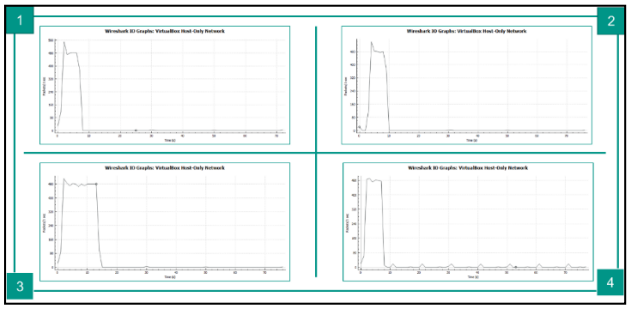


Gambar 7 Dampak Pengujian *Web Server* 1 Pada *Client*

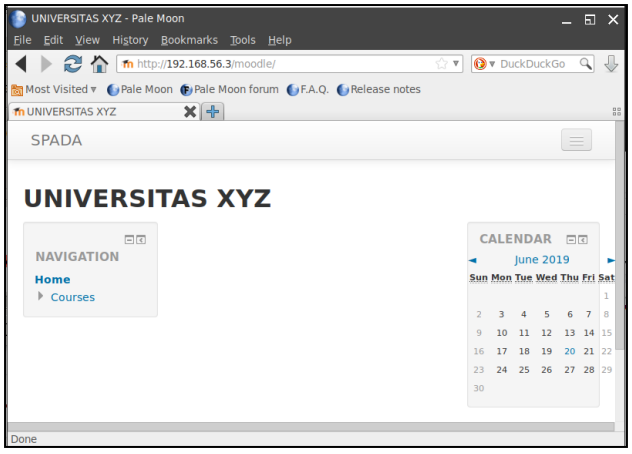
b. Pengujian *Web Server* 2



Gambar 8 Proses Penyerangan Dengan *Switchblade* (*Web Server 2*)

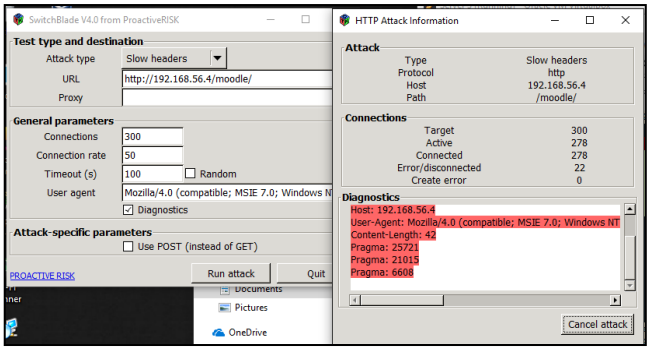


Gambar 9 Pengujian *Web Server* 2

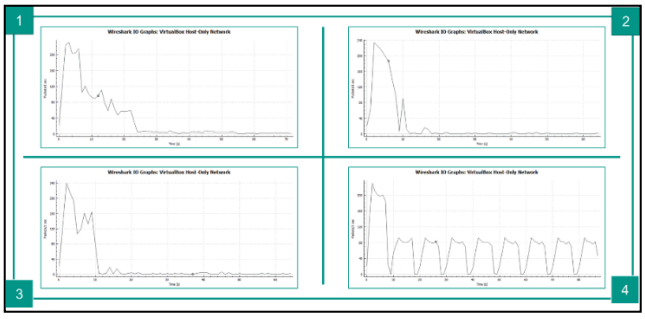


Gambar 10 Dampak Pengujian *Web Server* 2 Pada *Client*

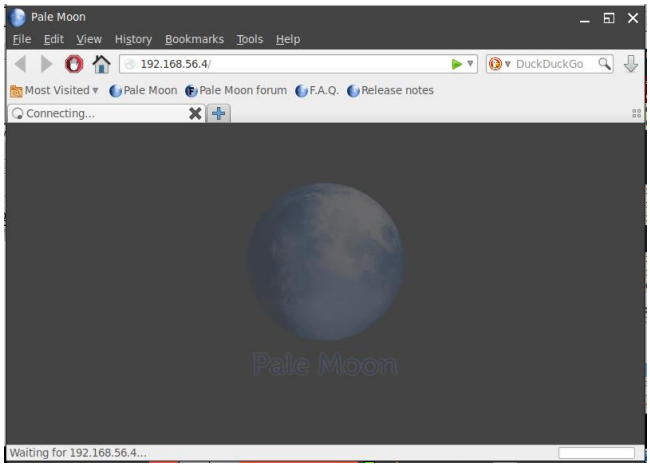
c. Pengujian *Web Server* 3



Gambar 11 Proses Penyerangan Dengan *Switchblade* (*Web Server 3*)



Gambar 12 Pengujian *Web Server* 3



Gambar 13 Dampak Pengujian *Web Server* 3 Pada *Client*

B. Pembahasan

Pada bagian ini akan dilakukan pembahasan terkait penelitian yang telah dilakukan, pembahasan meliputi hasil rancangan sistem *Web Server* dan juga penjelalasan mengenai pengujian setiap *Web Server*, terhadap serangan *Slow Headers*.

1. Analisis Pembahasan Sistem *Web Server*

Pada penelitian ini, sistem *Web Server* yang digunakan sebagai objek pengujian, menggunakan tiga sistem *Web Server*. Walaupun dapat menampilkan halaman situs yang sama, akan tetapi metode pengamanan yang digunakan berbeda. Sistem akan dijalankan satu per satu, dan tidak secara bersamaan, tujuannya adalah agar perangkat yang digunakan untuk penelitian, tidak terlalu terbebani dengan banyaknya sistem yang berjalan sekaligus.

2. Analisis Hasil Pengujian Sistem *Web Server*

a. Web Server 1

Dampak yang terjadi ketika *Web Server* diakses melalui *browser virtual Client*, *Web Server* tidak dapat menampilkan halaman beranda dari *website* universitas xyz, hal tersebut dikarenakan belum adanya pengamanan yang diterapkan pada *Web Server* 1, serangan tersebut dapat dikatakan sukses, karena jumlah *Connected* (jumlah serangan yang berhasil dikirimkan) pada *switchblade* lebih besar dari jumlah error/dis*connected* (jumlah serangan yang gagal) koneksi yang dikirimkan, seperti pada gambar 5. Contoh dampak serangan pada saat mengakses *Web Server* melalui *browser* dapat dilihat pada gambar 7.

b. Web Server 2

Pengamanan ini berhasil menghalangi *attacker*, hal tersebut dapat dilihat dari hasil yang tampil pada *browser attacker* yang mendapatkan pesan *HTTP ERROR/INTERNAL SERVER ERROR (*walaupun *Web Server* mengalami paket *load* yang cukup tinggi diawal*)*, sedangkan ketika *Client* mengakses *Web Server* dari *browser*, *Client* masih dapat mengaksesnya dengan baik, walaupun *Web Server* sedang dalam kondisi diserang. Pada pengujian ini *Switchblade* tidak dapat mengirimkan jumlah serangan dengan baik, karena jumlah *Connected* (jumlah serangan yang berhasil dikirimkan) cukup kecil hal itu dapat dilihat pada gambar 8. Contoh dampak serangan pada saat mengakses *Web Server* menggunakan *browser* dapat dilihat pada gambar 10.

c. Web Server 3

Dampak yang terjadi ketika *Web Server* diakses melalui *browser virtual Client*, *Web Server* tidak dapat menampilkan halaman beranda dari *website* universitas xyz. Dari pengujian yang telah dilakukan ternyata Mod Evasive tidak dapat menangkal serangan Slow Headers. serangan tersebut dapat dikatakan sukses, karena jumlah *Connected* (jumlah serangan yang berhasil dikirimkan) pada *switchblade* lebih besar dari jumlah error/dis*Connected* (jumlah serangan yang gagal) koneksi yang dikirimkan, seperti pada gambar 11. Contoh dampak serangan dapat dilihat pada gambar 13.

Pada pengujian yang telah dilakukan, dapat diketahui bahwa parameter yang sangat berpengaruh pada pengujian yang telah dilakukan adalah parameter *timeout*, karena semakin kecil parameter yang digunakan semakin tinggi beban yang diterima oleh sistem *Web Server*. Parameter ini berfungsi sebagai batas pengiriman koneksi, jadi seberapa besar pun jumlah koneksi akan dikirimkan secepat mungkin sesuai parameter *timeout* (*timeout* disini dapat diartikan sebagai tekanan untuk mempercepat serangan, terhadap jumlah koneksi yang dikirmkan).

Dampak yang terjadi, mulai dari peningkatan pada setiap rekaman grafik masing-masing pengujian terhadap setiap *Web Server*, hingga dampak pada saat *Client* mengakses *Web Server* melalui *browser*, yang terjadi pada *Web Server* 3, dengan *Web Server* 1 yang tidak menerapkan metode pengamanan, hampir sama persis. Walaupun paket *load* pada *Web Server* 3 tidak sebanyak *Web Server* 1, akan tetapi *Client Web Server* 3, tidak dapat melakukan akses pada *website Web Server 3 tersebut* melalui *browser*, hal ini dikarenakan pengamanan *Mod Evasive* tidak dapat menangani serangan *slow headers*, karena *Mod Evasive* hanya dapat melakukan pemblokiran atau membatasi permintaan yang utuh saja (*Mod Evasive* tidak memiliki kemampuan untuk membatasi koneksi permintaan yang tidak utuh). Berbeda dengan *Mod* Security yang dapat melakukan pemblokiran terhadap serangan slow headers, walaupun paket permintaan yang dikirimkan tidak utuh. Hal ini dapat dilakukan *Mod Security*, karena pada *Mod Security* dapat ditambahkan sebuah *rule* tambahan seperti, menambahkan sebuah *plugin* (yang dapat dikhususkan untuk melindungi *Web Server*, dari suatu jenis serangan tertentu, termasuk serangan *Slow Headers*).

5. Kesimpulan dan Saran

A. Kesimpulan

Berdasarkan dari semua tahapan dan pengujian pada penelitian, maka dapat diambil kesimpulan sebagai berikut:

1. Jumlah atau tingginya paket yang masuk pada saat awal serangan yang dapat dilihat pada pengujian, tidak selalu dapat memberikan gambaran langsung terhadap dampak akses *website*, pada suatu *Web Server* yang sedang dilakukan pengujian dengan serangan *Slow Headers*.

2. Kembali stabilnya suatu lalu lintas paket yang masuk tidak selalu menjadi indikator suatu *Web Server* telah bebas dari pengaruh serangan *Slow Headers*, hal itu dapat dilihat pada pengujian *Web Server* 1 dan *Web Server* 3 yang telah dibahas pada bab 4, walaupun lalu lintas paket yang masuk sudah kembali stabil, akan tetapi tetap saja dampak dari serangan tersebut masih

berpengaruh, salah satu penyebabnya adalah masih tingginya kenaikan *memory* yang digunakan pada *server* saat pengujian serangan berlangsung.

3. Metode pengamanan terbaik pada penelitian ini, yang digunakan untuk menangkal atau meminimalisir serangan *Slow Headers* adalah *Mod Security*, hal tersebut dapat dilihat pada saat pengujian berlangsung, hanya metode pengamanan *Mod Security* yang dapat mempertahankan *Web Server*, agar situs tetap dapat diakses. Walaupun demikian, metode pengamanan ini memiliki paket *load* yang paling tinggi diawal akan tetapi hanya terjadi dalam waktu yang singkat. Selain itu saat pengujian berlangsung, hanya *Web Server* yang menerapkan metode pengamanan *Mod Security* saja, yang mengalami kenaikan penggunaan *memory* yang paling sedikit diantara *Web Server* yang lain. Sedangkan pengamanan *Mod Evasive* tidak dapat melakukan perlindungan dengan baik dikarenakan, pengamanan ini hanya dapat melakukan pembatasan terhadap paket permintaan yang utuh saja, sedangkan serangan *Slow Headers* melakukan permintaan yang tidak utuh.

B. Saran

Berdasarkan kesimpulan yang telah dijelaskan sebelumnya mengenai hasil dari pengujian pengkondisian dengan menggunakan serangan *Slow Headers,* maka beberapa hal yang dapat penulis sarankan adalah sebagai berikut:

1. Untuk melakukan pengamanan menggunakan *Mod Security* diharuskan juga menggunakan *CRS (Core Rules Set),* karena *Mod Security* tidak dapat melakukan pengamanan terhadap serangan tersebut tanpa tambahan *module* dan *rules*.
2. Untuk penerapan pengamanan *Mod Security* pada *Web Server* sebenarnya, disarankan untuk menambahkan perlindungan lainnya seperti, *Cloudflare*, *Sucuri* atau yang lainnya, karena sebenarnya sistem *rules* dari *Mod Security* yang diperuntukkan untuk serangan *Slow Headers* masih dalam tahapan eksperimental yang mana pengamanan ini belum stabil.

Daftar Pustaka

Adriant, M. F. and Mardianto, I. (2015) 'IMPLEMENTASI WIRESHARK UNTUK PENYADAPAN (SNIFFING) PAKET DATA JARINGAN', *Fakultas Teknologi Industri, Universitas Trisakti*.

Ambarita, A. (2016) 'Implementasi Sistem E-Learning Menggunakan Software Moodle Pada Politeknik Sains Dan Teknologi Wiratama Maluku Utara', 1(September).

Gediminas (2019) *What is Apache? An In-Depth Overview of Apache Web Server*, *Hostinger.com*.

Impe, K. Van (2015) *Defending Against Apache Web Server DDoS Attacks*, *securityintelligence.com*.

Mishra, D. P. and Sourav, K. (2012) 'DDoS Detection and Defense: Client Termination Approach', pp. 2–6. doi: 10.1145/2381716.2381859.

Nulismuh (2016) *Sistem Pembelajaran Kuliah Daring DIKTI*, *Universitas Jember*.

OWASP (2018) *OWASP Switchblade 4.0*, *OWASP*.

Park, J. *et al.* (2014) 'Analysis of Slow Read DoS Attack and Countermeasures', *Proceedings of the International Conference on Cyber-Crime Investigation and Cyber Security*.

Rifqi, A. (2012) 'PERANCANGAN WEB SERVER MENGGUNAKAN BAHASA PEMROGRAMAN PYTHON 2.3', *Universitas Diponegoro, Semarang.*, pp. 1–7. doi: 10.1109/TPAMI.2007.40.

Sakasuti (2019) *Menentukan Arah Pendidikan Daring di Indonesia*, *RISTEKDIKTI*.

Suroto, S. (2018) 'A Review of Defense Against Slow HTTP Attack', *JOIV : International Journal on Informatics Visualization*, 1(4), p. 127. doi: 10.30630/joiv.1.4.51.

Ubuntu-ID (2014) *Linux untuk umat manusia*, *Ubuntu id*.

Budiman, A., Sucipto, A. and Dian, A.R. (2021). Analisis Quality of Service Routing MPLS OSPF Terhadap Gangguan Link Failure. Techno.com, 20(1), pp.28–37.

Ahdan, Syaiful. (2015). STRESS TESTING TO THE NETWORK TOPOLOGY USING NS2 MODELING AND SIMULATION OF NETWORK. 10.13140/RG.2.2.16100.58249.