ANTIVIRUS: Jurnal Ilmiah Teknik Informatika (p – ISSN: 1978 – 5232; e – ISSN: 2527 – 337X)
Vol. 18 No. 1 Mei 2024, pp. 27 – 36

SISTEM KEAMANAN WEBSITE DARI SERANGAN DENIAL OF SERVICE, SQL INJECTION, CROSS SITE SCRIPTING MENGGUNAKAN WEB APPLICATION FIREWALL

Diterima Redaksi: 19 November 2023; Revisi Akhir:27 November 2023; Diterbitkan Online: 10 Mei 2024

Stefanus Eko Prasetyo[1)], Haeruddin[2)], Kelvin Ariesryo[3)]
[1, 2, 3)] Fakultas Teknologi Informasi dan Universitas Internasional Batam
[1, 2, 3)] Jalan Gajah Mada, Kec. Sekupang, Kota Batam, kepulauan Riau, Indonesia, kode pos: 29426
e-mail: stefanus.eko@uib.edu[1)], haeruddin@uib.edu[2)], 2032002.kelvin@uib.edu[3)]

Abstrak **:** *Internet dan aplikasi web sangat berperan penting dalam kehidupan modern saat ini. Beberapa aktivitas sehari hari*
seperti browsing, memesan tiket, membayar tagihan menjadi semakin mudah dengan menggunakan aplikasi web. Saat ini banyak orang yang menggunakan aplikasi web untuk produk atau jasa yang diinginkan. Pengguna yang memberikan nama, data-data pribadi, data pembayaran, bisa menjadi sumber penghasilan bagi para hacker yang menarget informasi rahasia pengguna. Hacker tidak hanya bisa mencuri data-data rahasia pengguna, tetapi juga bisa memasukkan malware ke dalam website yang diserang. Pada sebuah Penelitian lainnya, disebutkan bahwa sebuah server sangat rentan jika tidak memiliki firewall atau keamanan yang baik. Solusi yang diajukan oleh penulis tersebut adalah menambahkan sebuah layanan di antara user dan server sebagai perantara sehingga hacker tidak bisa langsung masuk ke dalam server sebuah aplikasi web. Dalam penelitian ini, peneliti akan menggunakan website yang dilindungi oleh cloudflare sebagai target untuk dilakukan penetration testing dengan menggunakan kali linux. Penelitian ini dilakukan untuk menguji apakah cloudflare dapat mencegah serangan-serangan yang akan dilancarkan oleh para hacker. Dengan menggunakan cloudflare, penulis dapat mengatur rule-rule serta tingkat keamanan dari website maka dapat dengan mudah mencegah serangan serangan yang dilakukan oleh para hacker. Sebelum menggunakan Cloudflare, potensi website untuk terserang sangat tinggi, semua uji serang terhadap website berhasil tembus ke dalam website dan tidak ada serangan yang diblok. Tetapi setelah dilakukan pemasangan Cloudflare, semua serangan dapat terblok. Dengan begitu dapat disimpulkan bahwa dari hasil pengujian sebelum dan sesudah menggunakan cloudflare, keseluruhan serangan dapat 100% terblok oleh rule yang sudah dibuat.

Kata Kunci— Cloudflare, Hacker, Internet, Website

Abstract**:** *The Internet and web applications play an important role in modern life today. Some of activities like browsing, booking tickets, paying bills are becoming easier using a website. Nowadays a lot of people are using web applications for the desired things. Users who provide personal or payment data, can be sources of income for hackers targeting user sensitive information. Hackers can not only steal confidential user data, but also insert malware into the website. In another study, it was mentioned that a server is very vulnerable if it doesn't have a good security. The solution proposed by the author is to add a service between the user and the server so that the hacker cannot directly enter the website server. In this study, researchers will use Cloudflare-protected websites as targets for penetration testing using Kali Linux. This research was conducted to test whether Cloudflare can prevent attacks that would be launched by hackers. By using Cloudflare, writer can set the rules and security level of the website so can easily prevent the attacks by hackers. Before using Cloudflare, the potential for a website to be attacked was very high, all attacks against the website were successful penetrating the website and no attacks were blocked. But after installing Cloudflare and created the security rule, all the penetration testing attack were blocked. It can be concluded that from the test results before and after using Cloudflare, all attacks can be 100% blocked by the rules that have been created.*

Keywords— Cloudflare, Hacker, Internet, Website

I. PENDAHULUAN

INTERNET dan aplikasi web sangat berperan penting dalam kehidupan modern saat ini. Beberapa aktivitas sehari hari seperti *browsing*, memesan tiket pesawat maupun kapal, membayar tagihan menjadi semakin mudah dengan menggunakan aplikasi web. Kebanyakan aplikasi web

DOI: https://doi.org/10.35457/antivirus.v18i1.3339

menggunakan *database* sebagai *back end* untuk menyimpan data-data penting seperti informasi pribadi pengguna, riwayat pembayaran pengguna, data Perusahaan, dan lain sebagainya [1]. Dengan adanya kemudahan pada aplikasi web saat ini, juga membuatnya mudah terserang oleh para *hacker*.

Dari sekian banyak kategori serangan yang ada, ada beberapa serangan yang sangat terkenal, yaitu: *XSS, DDoS,* dan juga *SQL Injection*. Serangan *XSS* (*Cross-Site Scripting*) sangat mudah di eksploitasi karena sangat banyak tools gratis yang mudah digunakan untuk menyerang aplikasi web bahkan bagi orang yang minim pengetahuan tentang *hacking*. *XSS* merupakan tipe serangan injeksi di mana penyerang memasukkan skrip berbahaya ke dalam aplikasi web yang mudah diserang. Akibat dari serangan *XSS* yang sukses adalah *session hijacking*, data sensitive yang tersebar, *csrf*, dan bahkan peniruan identitas korban [2].

DDoS (Distributed Denial of Service) merupakan sebuah serangan yang paling banyak digunakan oleh para *hacker*. Cara kerja serangan ini adalah penyerang mengirim banyak *request* berupa *zombie* secara terus menerus sehingga *traffic* dari server yang diserang akan penuh. Akibat dari serangan ini adalah server menjadi *down* dan *user* tidak dapat mengakses server tersebut [3].

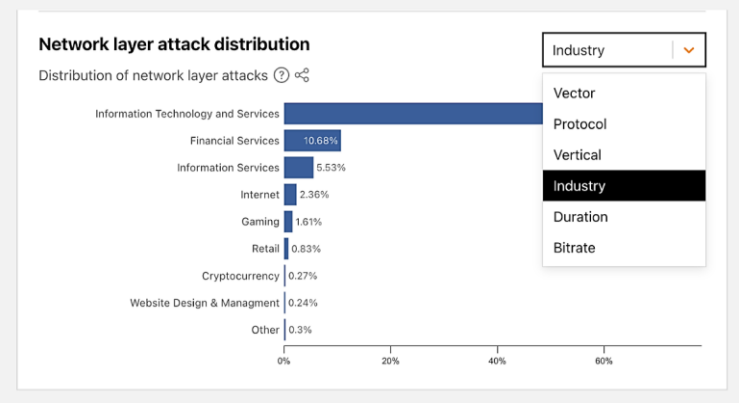
SQL *Injection* adalah sebuah teknik mengeksploitasi kerentanan korban dengan mengirim SQL *statement* berbahaya ke dalam *database*. Jika serangan berhasil, maka penyerang dapat memodifikasi, menghapus, maupun mengambil data tanpa perlu hak akses istimewa. *SQL query* merupakan senjata mendasar yang digunakan dalam melancarkan serangan ini [4]. *SQL Injection* adalah jenis serangan yang memanfaatkan *coding* dari aplikasi web yang tidak benar yang memungkinkan penyerang *menginject* SQL *command* ke dalam *login form* dan memungkin kan penyerang memperoleh akses ke dalam data yang di simpan dalam *database* [5].

Saat ini banyak orang yang menggunakan aplikasi web untuk produk atau jasa yang diinginkan. Pengguna yang memberikan nama, data-data pribadi, data pembayaran, bisa menjadi sumber penghasilan bagi para *hacker* yang menarget informasi rahasia pengguna. *Hacker* tidak hanya bisa mencuri data-data rahasia pengguna, tetapi juga bisa memasukkan *malware* ke dalam *website* yang diserang. Keamanan aplikasi *website* sangat berdampak pada keberlangsungan sebuah bisnis yang dapat mempengaruhi reputasi dan juga kinerja [6]. Dalam sebuah Penelitian, ada penulis yang menyatakan pendekatan *secure coding* dapat dilakukan oleh developer untuk mencegah serangan SQL *Injection*. Penulis tersebut memfokuskan Penelitian *SQL Injection* pada: *input* validasi URL, sanitasi data, PDO untuk eksekusi *query*, *query*, dan *session tokenization*. Penelitian yang dilakukan terbukti sangat efektif dan efisien dalam pencegahan *SQL Injection* [7].

Pada sebuah Penelitian lainnya, disebutkan bahwa sebuah server sangat rentan jika tidak memiliki *firewall* atau keamanan yang baik. Solusi yang diajukan oleh penulis tersebut adalah menambahkan sebuah layanan di antara *user* dan *server* sebagai perantara sehingga *hacker* tidak bisa langsung masuk ke dalam server sebuah aplikasi *web* [8].

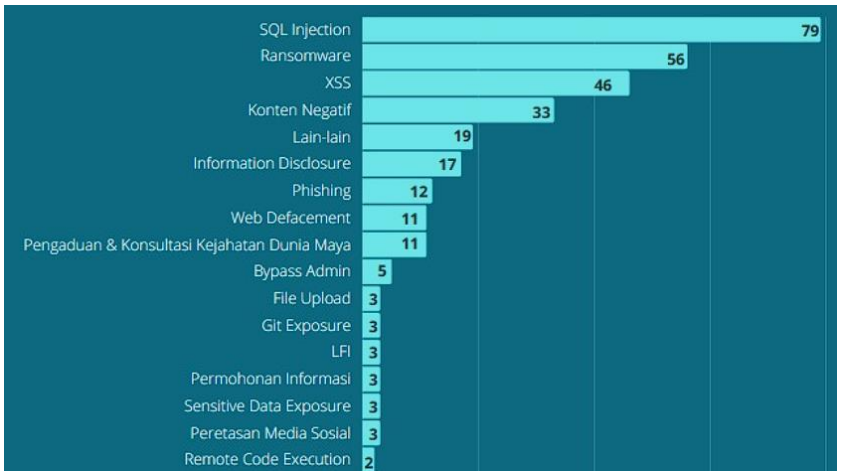
Berikut ini merupakan persentase data dari serangan *DDoS* yang terjadi di dunia pada tahun 2023. Bisa dilihat bahwa serangan pada sektor industri teknologi informasi menjadi yang tertinggi. Berbagai serangan itu dilakukan oleh sekelompok *hacktivist* pro-Rusia *Revil, Killnet*, dan *anonymous* Sudan terhadap *website* barat. Serangan terbesar pada tahun 2023 ini adalah *ACK flood DDoS* yang berasal dari *Mirai-variant botnet* yang terdiri dari sekitar 11.000 alamat IP dan menargetkan ISP dari Amerika dengan kecepatan mencapai 1.4Tbps.

DOI: https://doi.org/10.35457/antivirus.v18i1.3339



Gambar 1 Data Infografis Serangan DDoS

Di bawah ini adalah data statistik dari laporan yang diajukan setiap daerah di Indonesia pada tahun 2022. Dapat dilihat bahwa serangan yang banyak diajukan adalah serangan *SQL Injection* dan diikuti oleh *Ramsomware* dan XSS.



Gambar 2 Data Infografis Laporan Serangan di Indonesia

Pada tahun 2023, di Indonesia terdapat 347.17 juta serangan digital sejak Januari hingga Juni 2023. Angka serangan tersebut sempat turun pada bulan April, tetapi melonjak Kembali pada bulan berikutnya. Menurut jenisnya, serangan yang paling banyak ditemui diindonesia dikenal dengan istilah *misc activity* dan ditemukan sebanyak 119.94 juta serangan sejak Januari hingga Juni 2023.

Gambar 3 Data Infografis Jumlah serangan di Indonesia 2023

Dalam penelitian ini, peneliti akan menggunakan *website* yang dilindungi oleh *cloudflare* sebagai target untuk dilakukan pengujian *penetration testing* dengan menggunakan kali *linux* untuk menghasilkan website yang aman [9]. *Penetration Testing* adalah sebuah metode pengujian kerantanan

DOI: https://doi.org/10.35457/antivirus.v18i1.3339

dan keamanan dari sebuah sistem, jaringan komputer, maupun kelemahan aplikasi *web* [10]. Penelitian ini dilakukan untuk menguji apakah *cloudflare* dapat mencegah serangan-serangan yang akan dilancarkan oleh para *hacker*. Serangan yang akan digunakan adalah *DDoS, SQL Injecion*, dan *XSS*.

a. Tujuan dan Manfaat Penelitian

1. Tujuan Penelitian

Sesuai dengan rumusan masalah yang telah ter paparkan di atas, tujuan dari penelitian ini adalah untuk membuat sebuah sistem keamanan aplikasi berbasis web untuk mencegah serangan dari *DDoS*, *SQL Injection*, dan *XSS* dengan menggunakan *cloudflare*.

2. Manfaat Penelitian

Setelah melakukan penelitian yang didukung dengan data-data yang akurat sehingga kebenarannya dapat diterima, maka harapan penulis hasil dari penelitian ini dapat dijadikan pengembangan teoritis bagi peneliti berikutnya dan berguna bagi masyarakat. Penelitian ini juga diharapkan dapat meningkatkan sistem keamanan aplikasi berbasis web oleh setiap orang yang menggunakannya.

II. TINJAUAN PUSTAKA

A. Serangan XSS (Cross-Site Scripting)

Serangan *XSS* (*Cross-Site Scripting*) sangat mudah di eksploitasi karena sangat banyak tools gratis yang mudah digunakan untuk menyerang aplikasi web bahkan bagi orang yang minim pengetahuan tentang *hacking*. *XSS* merupakan tipe serangan injeksi di mana penyerang memasukkan skrip berbahaya ke dalam aplikasi web yang mudah diserang. Akibat dari serangan *XSS* yang sukses adalah *session hijacking*, data sensitive yang tersebar, *csrf*, dan bahkan peniruan identitas korban [2].

B. DDoS (Distributed Denial of Service)

DDoS (Distributed Denial of Service) merupakan sebuah serangan yang paling banyak digunakan oleh para *hacker*. Cara kerja serangan ini adalah penyerang mengirim banyak *request* berupa *zombie* secara terus menerus sehingga *traffic* dari server yang diserang akan penuh. Akibat dari serangan ini adalah server menjadi *down* dan *user* tidak dapat mengakses server tersebut [3].

C. SQL Injection

SQL *Injection* adalah sebuah teknik mengeksploitasi kerentanan korban dengan mengirim SQL *statement* berbahaya ke dalam *database*. Jika serangan berhasil, maka penyerang dapat memodifikasi, menghapus, maupun mengambil data tanpa perlu hak akses istimewa. *SQL query* merupakan senjata mendasar yang digunakan dalam melancarkan serangan ini [4]. *SQL Injection* adalah jenis serangan yang memanfaatkan *coding* dari aplikasi web yang tidak benar yang memungkinkan penyerang *menginject* SQL *command* ke dalam *login form* dan memungkin kan penyerang memperoleh akses ke dalam data yang di simpan dalam *database* [5].

D. Firewall

Pada sebuah Penelitian lainnya, disebutkan bahwa sebuah server sangat rentan jika tidak memiliki *firewall* atau keamanan yang baik. Solusi yang diajukan oleh penulis tersebut adalah menambahkan sebuah layanan di antara *user* dan *server* sebagai perantara sehingga *hacker* tidak bisa langsung masuk ke dalam server sebuah aplikasi *web* [8].

E. Load Balancer

Load balancer berfungsi untuk mengelompokkan koneksi *traffic* yang melewati *cloudflare* menjadi beberapa bagian dan memisahkan bebas koneksi *traffic* sehingga tidak terdapat beban yang berlebih pada *server* [11].

F. Kali Linux

Kali linux adalah sebuah platform *penetration testing* dan audit keamanan dengan alat canggih untuk mengidentifikasi, mendeteksi, dan mengeksploitasi kerentanan yang ditemukan pada target [12].

DOI: https://doi.org/10.35457/antivirus.v18i1.3339

Kali linux ini adalah pembaharuan dari *back track linux* yang stabil dan sempurna dan merupakan turunan dari *Debian* [13].

G. *Cloudflare*

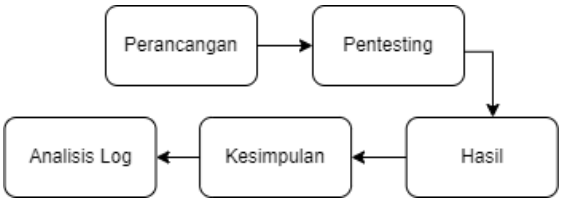
Cloudflare adalah sebuah layanan keamanan *website* ataupun aplikasi berbasis *web* yang disediakan oleh Perusahaan Amerika Serikat yang menyediakan jasa *Conten Delivery Network* (CDN), keamanan internet, pencegahan *DDoS*, pemantauan *traffic* jaringan, dan banyak layanan lainnya [14]. Terdapat beberapa *webserver Cloudflare* di seluruh dunia. *webserver* ini mendistribusikan data kepada *client* untuk mengurangi jumlah latensi dan meningkatkan kecepatan. Penggunaan *Cloudflare* dapat membuat web menjadi lebih cepat saat diakses oleh pengguna, dan dapat terlindungi dari *hacker* [15].

H. *Penetration Testing*

Penetration Testing adalah sebuah metode pengujian kerentanan dan keamanan dari sebuah sistem, jaringan komputer, maupun kelemahan aplikasi web [10]. Tujuan dari metode ini adalah untuk memastikan apakah ad suatu celah dalam sebuah jaringan. Hal ini bertujuan supaya dapat melakukan pencegahan secara dini sebelum *website* diserang oleh *hacker*.[16]

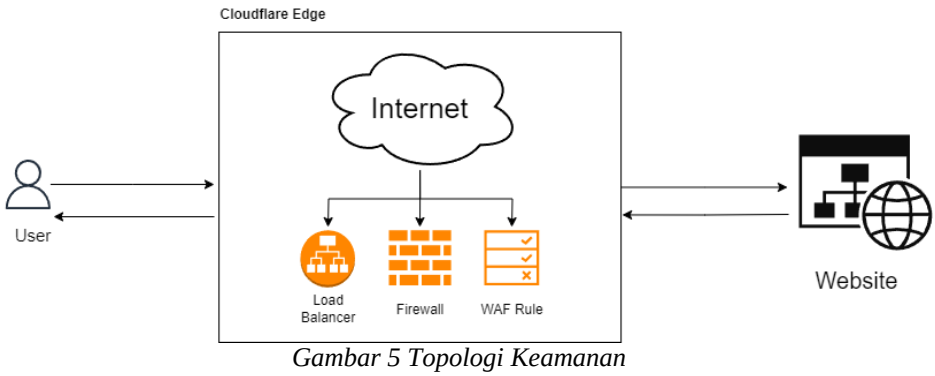
III. METODE PENELITIAN

Pada penelitian ini, yang akan dilakukan adalah melakukan uji serangan pada *website* yang telah dilindungi oleh *cloudflare*. Berikut ini adalah gambar dari proses rancangan penelitian yang akan dilakukan.



Gambar 4 Alur Penelitian

Dalam Penelitian ini, akan diimplementasikan sebuah sistem keamanan aplikasi berbasis web dari serangan *DOS*, *SQL Injection*, *Cross-site scripting* (*XSS*) menggunakan *WAF*. Fungsi dari *WAF* ini adalah melindungi web dari *hacker*, memblok pesan atau notifikasi yang tidak diinginkan, dan juga memonitor *bandwith* [17]. *WAF* yang akan digunakan adalah *Cloudflare*. *Cloudflare* adalah sebuah layanan keamanan *website* ataupun aplikasi berbasis *web* yang disediakan oleh Perusahaan Amerika Serikat yang menyediakan jasa *Conten Delivery Network* (CDN), keamanan internet, pencegahan *DDoS*, pemantauan *traffic* jaringan, dan banyak layanan lainnya [14]. Implementasi ini akan dilakukan pada sebuah aplikasi berbasis *web* dengan menggunakan topologi keamanan seperti gambar di bawah ini.



Gambar 5 Topologi Keamanan

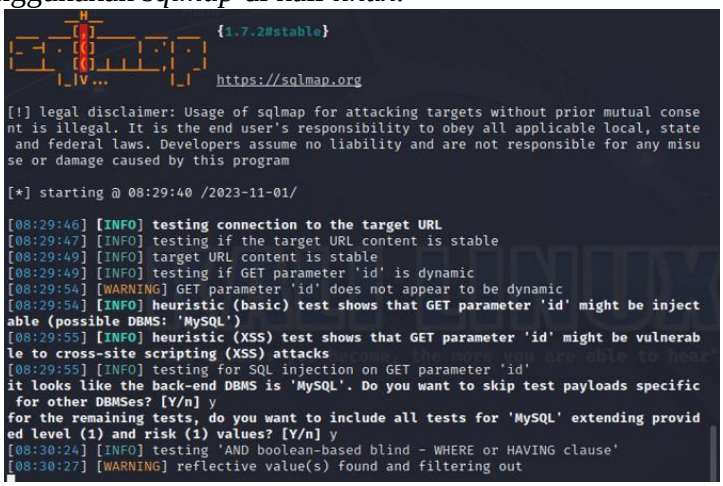
Saat *user* ingin mengakses sebuah aplikasi *web* melalui internet, maka akan disaring oleh *cloudflare* terlebih dahulu dengan menggunakan *firewall*, *load balancer*, dan juga *rule* pada *cloudflare*. *Load balancer* berfungsi untuk mengelompokkan koneksi *traffic* yang melewati *cloudflare* menjadi beberapa bagian dan memisahkan bebas koneksi *traffic* sehingga tidak terdapat beban yang berlebih

DOI: https://doi.org/10.35457/antivirus.v18i1.3339

pada *server* [11]. Untuk mengetes apakah aplikasi berbasis web yang terlindungi oleh cloudflare dapat mencegah serangan yang masuk, maka akan dilakukan sebuah percobaan berupa *penetration testing* pada website dengan menggunakan *kali linux*. *Kali linux* adalah sebuah platform *penetration testing* dan audit keamanan dengan alat canggih untuk mengidentifikasi, mendeteksi, dan mengeksploitasi kerentanan yang ditemukan pada target [12].

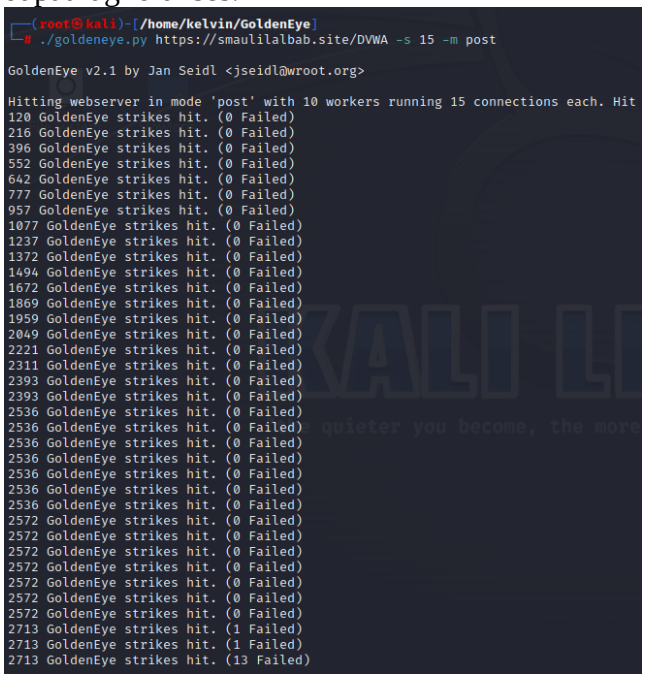
IV. HASIL DAN PEMBAHASAN

Pada bagian ini akan membahas bagaimana cara menggunakan *cloudflare* untuk mencegah serangan dari para *hacker*. Untuk serangan pertama yang akan dilakukan adalah *SQL Injection*. Pada serangan ini, penulis akan menggunakan *tools* di *kali linux* yang bernama *sqlmap*. Tujuan dari *sqlmap* adalah untuk mencuri data data dari penting *website* yang diserang Gambar di bawah ini merupakan proses penyerangan menggunakan *sqlmap* di kali *linux*.



Gambar 6 SQL Map

Untuk serangan kedua, akan dilakukan serangan *DDoS*. *Tools* yang akan digunakan pada kali *linux* adalah *GoldenEye*. Cara kerja *tools* ini adalah dengan mengirim banyak *request* pada target supaya target mulai melambat dan lama kelamaan servernya menjadi *down*. Dengan *server* yang *down*, maka sebuah *website* tidak dapat lagi diakses.

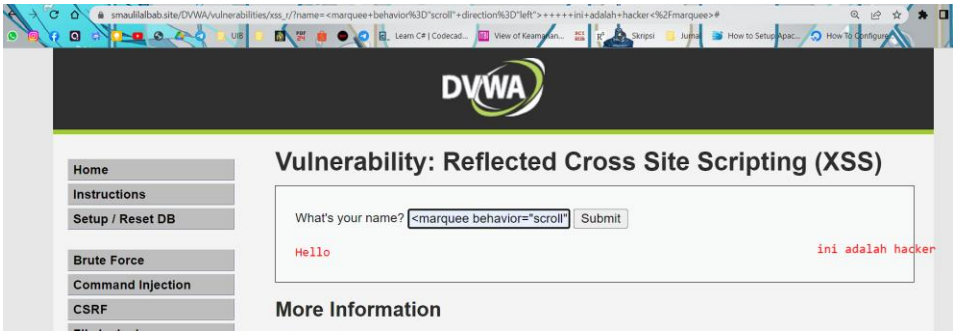


Gambar 7 GoldenEye

Untuk serangan yang terakhir adalah XSS. Gambar di bawah ini merupakan sebuah serangan XSS dimana penyerang memasukkan skrip ke dalam kolom *text* dan akan muncul hasil sesuai yang

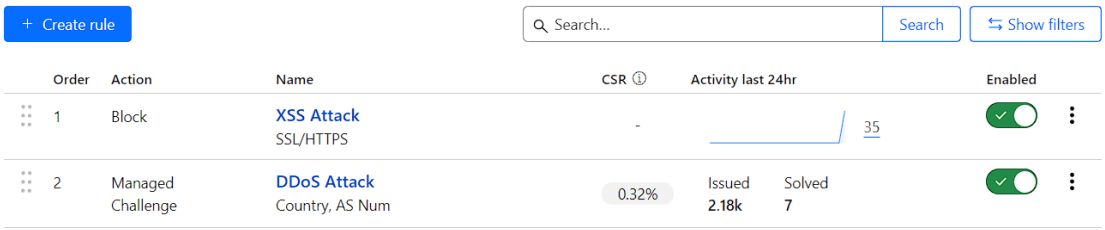
DOI: https://doi.org/10.35457/antivirus.v18i1.3339

diinginkan oleh *hacker*. Jenis serangan ini dapat beragam, dapat berupa memasukkan *link* berbahaya, *malware*, dan banyak lagi.

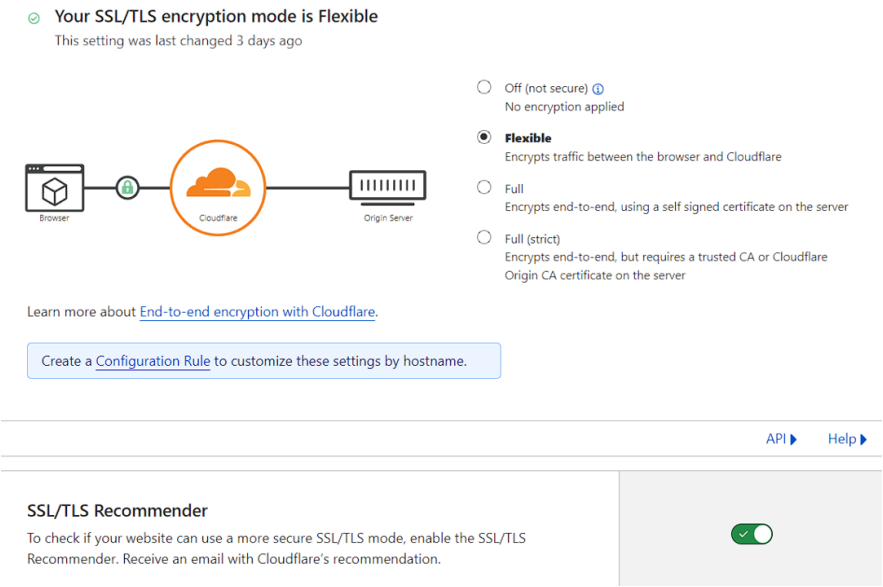


Gambar 8 XSS Attack

Untuk pencegahan dari serangan-serangan yang telah dilakukan, maka penulis akan membuat *rules* pada *firewall* untuk mencegah serangan *XSS* dan *DDoS*, membuat SSL/TLS *encryption* menjadi *flexible*, serta mengaktifkan mode "*I'm under attack*." Pada rules pertama jika ada orang yang ingin melakukan serangan *XSS* dengan memasukkan skrip berbahaya, maka akan langsung diblok. Pada *rules* kedua jika seseorang yang ingin mengakses *website* tersebut, maka akan diberi akses, sedangkan jika bot dari serangan *DDoS* yang ingin mengakses *website* tersebut dengan jumlah besar, akan langsung diblok. Dan dengan mengaktifkan mode *"I'm under attack*," maka sudah dapat mencegah *sqlmap* yang menyerang.

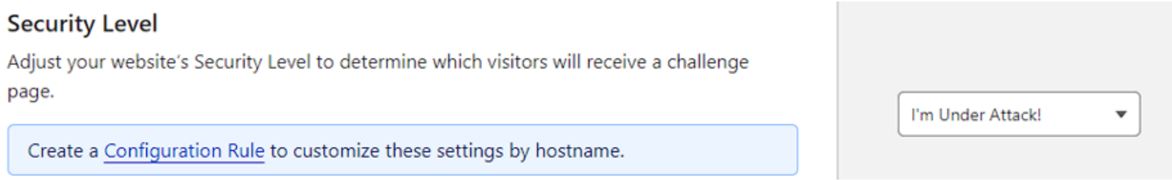


Gambar 9 Rule Pada Cloudflare



Gambar 10 Setting SSL/TLS

DOI: https://doi.org/10.35457/antivirus.v18i1.3339



Gambar 11 Setting Security Level Pada Cloudflare

Setelah mengaktifkan *rules* pada *firewall*, makan semua serangan dari para *hacker* bisa ter *block* dan tidak membahayakan *website* yang diserang. Berikut gambar di bawah ini menunjukkan kalau semua serangan yang dilakukan harus menerima tantangan dan dapat membahayakan *website*, maka serangan tersebut akan di blok.

Activity log

Edit columns

Date	Action taken	Country	IP address	Service
Nov 9, 2023 11:36:09 PM	Managed Challenge	Singapore	157.245.61.246	Security level
Nov 9, 2023 11:36:09 PM	Managed Challenge	Singapore	157.245.61.246	Security level
Nov 9, 2023 11:36:09 PM	Managed Challenge	Singapore	157.245.61.246	Security level
Nov 9, 2023 11:36:09 PM	Managed Challenge	Singapore	157.245.61.246	Security level
Nov 9, 2023 11:36:09 PM	Managed Challenge	Singapore	157.245.61.246	Security level
Nov 9, 2023 11:36:09 PM	Managed Challenge	Singapore	157.245.61.246	Security level
Nov 9, 2023 11:36:09 PM	Managed Challenge	Singapore	157.245.61.246	Security level
Nov 9, 2023 11:36:09 PM	Managed Challenge	Singapore	157.245.61.246	Security level
Nov 9, 2023 11:36:09 PM	Managed Challenge	Singapore	157.245.61.246	Security level

Gambar 12 Serangan DDoS yang di Challenge

Activity log

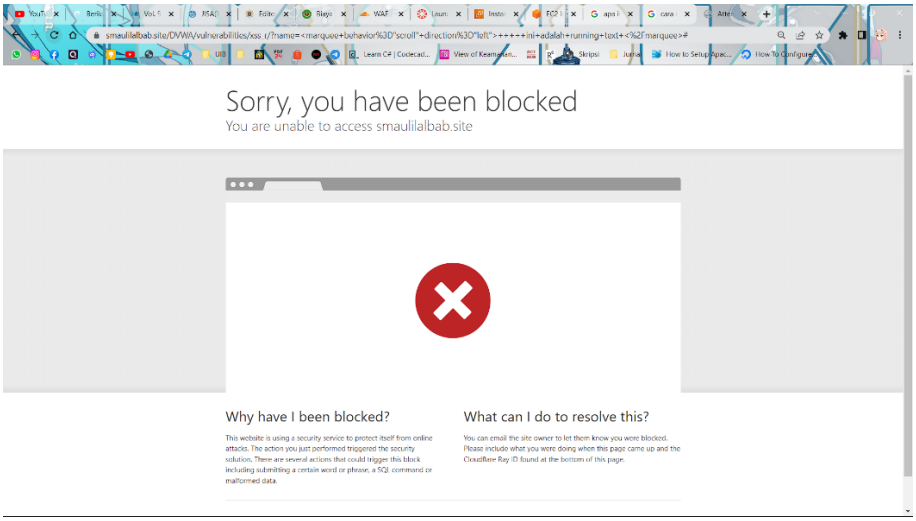
Edit columns

Date	Action taken	Country	IP address	Service
Nov 8, 2023 9:33:21 PM	Block	Indonesia	113.212.165.194	Custom rules
Nov 8, 2023 9:33:17 PM	Block	Indonesia	113.212.165.194	Custom rules
Nov 8, 2023 9:32:51 PM	Block	Indonesia	113.212.165.194	Custom rules
Nov 8, 2023 9:32:43 PM	Block	Indonesia	113.212.165.194	Custom rules
Nov 8, 2023 9:32:40 PM	Block	Indonesia	113.212.165.194	Custom rules
Nov 8, 2023 9:32:31 PM	Block	Indonesia	113.212.165.194	Custom rules
Nov 8, 2023 9:32:30 PM	Block	Indonesia	113.212.165.194	Custom rules
Nov 8, 2023 9:32:28 PM	Block	Indonesia	113.212.165.194	Custom rules
Nov 8, 2023 9:32:28 PM	Block	Indonesia	113.212.165.194	Custom rules
Nov 8, 2023 9:32:25 PM	Block	Indonesia	113.212.165.194	Custom rules

Gambar 13 Serangan XSS Yang Diblok

Setelah menggunakan *cloudflare*, jika ada *hacker* yang ingin menyerang *website* yang dituju, maka akan langsung di blok oleh *cloudflare* dan tidak dapat diakses lagi. Gambar di bawah ini merupakan tampilan *website* yang diblok karena *hacker* berusaha menyerang *website* dengan menggunakan *XSS Attack*.

DOI: https://doi.org/10.35457/antivirus.v18i1.3339



Gambar 14 Tampilan Website Hacker Yang Diblok

V. KESIMPULAN DAN SARAN

Di jaman sekarang, *website* yang tidak dilindungi oleh *firewall*, dapat menjadi sasaran empuk bagi para *hacker*. *Hacker* yang menyerang sebuah *website*, dapat membuat, mengambil, mengubah, dan juga menghapus data data penting yang berada di dalam *database website* tersebut. Banyak jenis serangan yang dapat dilakukan oleh *hacker*, beberapa diantaranya yaitu: *DDoS*, *XSS*, dan juga *SQL Injection*. Tiga serangan ini merupakan beberapa serangan yang sangat terkenal di dunia, dan banyak digunakan oleh para *hacker*.

Untuk mencegah supaya *hacker* tidak dapat menyerang sebuah *website*, maka diperlukan *firewall* dalam sebuah *website*. *Cloudflare* adalah salah satu *firewall* yang sangan banyak digunakan. Beberapa jenis serangan akan dilakukan dengan menggunakan *kali linux* .Dengan menggunakan *cloudflare*, penulis dapat mengatur *rule-rule* serta tingkat keamanan dari *website* maka dapat dengan mudah mencegah serangan serangan yang dilakukan oleh para *hacker*. Sebelum menggunakan *Cloudflare*, potensi *website* untuk terserang sangat tinggi, semua uji serang terhadap *website* berhasil tembus ke dalam *website* dan tidak ada serangan yang diblok. Namun setelah dilakukan pemasangan *Cloudflare*, semua uji serangan dapat terblok. Dengan begitu dapat disimpulkan bahwa dari hasil pengujian sebelum dan sesudah menggunakan *cloudflare*, keseluruhan serangan dapat 100% terblok oleh *rule-rule* yang sudah dibuat.

Kelemahan dari layanan *firewall cloudflare* ini adalah jika server *cloudflare down*, maka akan berpengaruh juga pada *website* walaupun penyedia *hosting* berjalan lancar. Tetapi server dari *cloudflare* sangat jarang *down*, jadi tidak perlu khawatir jika *website* terserang oleh para *hacker*.

DAFTAR PUSTAKA

[1] P. Sharma, R. Johari, and S.S Sharma, *Combined Approach to prevent XSS Attacks and SQL injection*. CPS IEEE, 2012.

[2] B. Gogoi, T. Ahmed, and H. K. Saikia, "Detection of XSS Attacks in Web Applications: A Machine Learning Approach," *International Journal of Innovative Research in Computer Science & Technology*, vol. 9, no. 1, pp. 1–10, Jan. 2021, doi: 10.21276/ijircst.2021.9.1.1.

[3] A. Aljuhani, T. Alharbi, and B. Taylor, "Mitigation of Application Layer DDoS Flood Attack Against Web Servers," *Journal of Information Security and Cybercrimes Research*, vol. 2, no. 1, 2019, doi: 10.26735/16587790.2019.002.

[4] S. Rai and B. Nagpal, "Detection & Prevention of SQL Injection Attacks: Developments of the Decade," 2019. [Online]. Available: https://www.researchgate.net/publication/332409784

[5] A. A. Onyekachi, A. O. Agbakwuru, and D. O. Njoku, "SQL Injection Attack on Web Base Application: Vulnerability Assessments and Detection Technique An Enhanced Query Process Algorithm for Distributed Database system View project Review of Prospect and Challenges of IOT in Nigeria Business View project SQL Injection Attack on Web Base Application:

DOI: https://doi.org/10.35457/antivirus.v18i1.3339

Vulnerability Assessments and Detection Technique," *International Research Journal of Engineering and Technology*, 2021, [Online]. Available: https://www.researchgate.net/publication/353257660

[6] Lakhno V *et al.*, "EXPERIMENTAL STUDIES OF THE FEATURES OF USING WAF TO PROTECT INTERNAL SERVICES IN THE ZERO TRUST STRUCTURE," *J Theor Appl Inf Technol*, vol. 15, no. 3, 2022, [Online]. Available: www.jatit.org

[7] J. Harefa, G. Prajena, A. Alexander, A. Muhamad, E. V. S. Dewa, and S. Yuliandry, "SEA WAF: The Prevention of SQL Injection Attacks on Web Applications," *Advances in Science, Technology and Engineering Systems Journal*, vol. 6, no. 2, pp. 405–411, Mar. 2021, doi: 10.25046/aj060247.

[8] G. H. A. Kusuma, "Sistem Firewall untukPencegahan DDOS ATTACK di Masa Pandemi Covid-19," *Journal of Informatics and Advanced Computing (JIAC)*, vol. 3, no. 1, 2022.

[9] S. E. Prasetyo, N. Hasanah, and G. Wijaya, "Pengujian Keamanan Learning Management System TutorLMS Terhadap Kerentanan Insecure Design dan Broken Access Control," *Telcomatics*, vol. 7, no. 2, Dec. 2022, doi: 10.37253/telcomatics.v7i2.7357.

[10] S. E. Prasetyo and N. Hassanah, "Analisis Keamanan Website Universitas Internasional Batam Menggunakan Metode ISSAF."

[11] A. Salim, G. Surono, E. B. Pabelan, and A. Raizaldi, "JBPI-Jurnal Bidang Penelitian Informatika Ciptaan disebarluaskan di bawah Lisensi Creative Commons Atribusi 4.0 Internasional Penerapan Load Balancing Metode Per Connection Classifier Berbasis Router Mikrotik di PT.Asuransi Jiwa Nasional," 2023. [Online]. Available: https://ejournal.kreatifcemerlang.id/index.php/jbpi

[12] T. S. Gunawan, M. K. Lim, M. Kartiwi, N. A. Malik, and N. Ismail, "Penetration testing using Kali linux: SQL injection, XSS, wordpres, and WPA2 attacks," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 12, no. 2, pp. 729–737, Nov. 2018, doi: 10.11591/ijeecs.v12.i2.pp729-737.

[13] T. Yusnanto *et al.*, "Analisa Infrastruktur Jaringan Wireless dan Local Area Network (WLAN) Meggunakan Wireshark Serta Metode Penetration Testing Kali Linux," *Journal on Education*, vol. 04, no. 04, pp. 1470–1476, 2022.

[14] G. H. A. Kusuma, "Sistem Firewall untuk Pencegahan DDOS ATTACK di Masa Pandemi Covid-19," *Journal of Informatics and Advanced Computing (JIAC)*, vol. 3, no. 1, 2022.

[15] Fandy, Rosmasari, and G. M. Putra, "Pengujian Kinerja Web Server Atas Penyedia Layanan Elastic Cloud Compute (EC2) Pada Amazon Web Services (AWS)," *Adopsi Teknologi dan Sistem Informasi (ATASI)*, vol. 1, no. 1, pp. 21–35, Jun. 2022, doi: 10.30872/atasi.v1i1.45.

[16] H. Haeruddin, "Analisa dan Implementasi Sistem Keamanan Router Mikrotik dari Serangan Winbox Exploitation, Brute-Force, DoS," *JURNAL MEDIA INFORMATIKA BUDIDARMA*, vol. 5, no. 3, p. 848, Jul. 2021, doi: 10.30865/mib.v5i3.2979.

[17] Z. Salim Alwan, M. F. Younis, and Z. S. Alwan, "Detection and Prevention of SQL Injection Attack: A Survey International Journal of Computer Science and Mobile Computing Detection and Prevention of SQL Injection Attack: A Survey," 2017. [Online]. Available: https://www.researchgate.net/publication/320108029

DOI: https://doi.org/10.35457/antivirus.v18i1.3339