PENGAMANAN DATA CLOUDFRI MENGGUNAKAN METODE *SECURITY HARDENING*

SECURING CLOUDFRI DATA USING SECURITY HARDENING METHOD

Yunan Sukho Aditya[1], Umar Yunan Kurnia Septo Herdianto[2], Muhammad Fathinuddin[3]

[1,2,3] Universitas Telkom, Bandung

[1]**yunansukho@student.telkomuniversity.ac.id**, [2]**umaryunan@telkomuniversity.ac.id**, [3]**muhammadfathinuddin@telkomuniversity.ac.id**

Abstrak

Security Hardening merupakan metode yang digunakan untuk melakukan *Hardening* atau penutupan celah agar suatu sistem terhindar dari serangan. Penelitian ini bertujuan untuk melakukan identifikasi *vulnerability* yang ada pada objek yaitu savsoftquiz.cloudfri.id untuk meminimalisir ancaman. Dengan digunakannya metode *Security Hardening*, dapat dilakukannya analisis dengan pengujian *scanning vulnerability* dan *exploit*.

Pengujian yang dilakukan untuk menemukan *vulnerability* yang terdapat pada savsoftquiz.cloudfri.id menggunakan *scanning tools* yang hasilnya akan digunakan untuk melakukan *exploit* pada objek. *Scanning tools* yang digunakan adalah HCL AppScan sedangkan untuk *exploit* menggunakan Metasploit, Burp Suite, dan Sqlmap. Hasil dari *scanning* mendapatkan beberapa *vulnerability* yang memiliki *threat level high*, sedangkan hasil dari melakukan *exploit* mendapatkan informasi yang menunujukan bahwa *email* dan *password* pada objek tidak terenkripsi. Hasil dari *exploit* juga menunjukan bahwa objek sudah terpasang WAF yang dapat menahan serangan *SQL Injection*.

Kata kunci: ***Security Hardening, Hardening, Vulnerability, Exploit***

Abstract

Security Hardening is a method used to perform Hardening or closing gaps so that a system is protected from attacks. This study aims to identify vulnerabilities that exist in the object, namely savsoftquiz.cloudfri.id to minimize threats. By using the Security Hardening method, analysis can be done by testing scanning vulnerabilities and exploits.

Tests carried out to find vulnerabilities contained in savsoftquiz.cloudfri.id using scanning tools whose results will be used to exploit objects. Scanning tools used are HCL AppScan while for exploits use Metasploit, Burp Suite, and Sqlmap. The results of the scanning get several vulnerabilities that have a high-level threat, while the results of the exploit get information that shows that the email and password on the object are not encrypted. The results of the exploit also show that the object has a WAF installed that can withstand SQL Injection attacks.

Keywords: ***Security Hardening, Hardening, Vulnerability, Exploit***

1. Pendahuluan

Perkembangan jumlah dan penggunaan *website* pada era teknologi informasi saat ini menunjukan peningkatan yang pesat. Banyak *website* digunakan untuk saling berbagi informasi antara satu sama lain. *Website* banyak digunakan karena *flexible* dan mudah pemakaiannya. Seseorang hanya membutuhkan waktu beberapa menit untuk membuat maupun membuka suatu *website*. Hal inilah yang menyebabkan perkembangan *website* meningkat.

Dengan mudahnya pembuatan dan penggunaan *website*, maka akan mudah juga suatu *website diretas (hack)* oleh *hacker*. Banyak ribuan *website* telah diretas oleh *hacker*. *Website* Pemerintahan dan Pendidikan menjadi target yang diminati oleh para *hacker* untuk dilakukan *deface*. Dengan banyaknya peretasan pada *website* maka dibutuhkan pengamanan data.

Pengamanan data ini perlu dilakukan untuk menghindari serangan dari *hacker*, *cracker*, dan sejenisnya yang bermaksud untuk mencuri atau merubah data yang ada pada *website*. Terdapat banyak serangan yang bisa dilakukan kepada *website*, contohnya seperti: *Phising, DoS, Bruteforce Attack, Defacing*, dan masih banyak lagi.

Serangan – serangan tersebut perlu dihindari untuk menghindari kerusakan ataupun kehilangan data pada *website*. Untuk menghindari suatu serangan, banyak cara yang dapat dilakukan, seperti: menggunakan *Firewall, Two Step Authentication, SSL*, atau melakukan *Hardening*.

Penggunaa *Hardening* dibutuhkan untuk pengamanan sistem. *Hardening* berguna untuk menutup celah – celah yang rentan di serang oleh para *hacker*. Penutupan celah – celah inilah yang membuat sistem jadi sulit untuk diserang. *Hardening* bisa digunakan pada semua sistem, termasuk sistem cloudfri Telkom University.

Hardening yang dilakukan pada cloudfri bisa menggunakan metode *security hardening*. Dimana *security hardening* ini memiliki empat tahapan, yaitu *access, analyze, remediate*, dan *manage*. Tahapan *access* berguna untuk mencari celah keamanan pada sistem, tahap *analyze* berguna untuk menganalisis tingkat keamanan dan dampak dari celah pada sistem, tahap *remediate* berguna untuk mencari cara untuk menutup celah yang ditemukan pada sistem, dan tahap *manage* yang berguna untuk menutup celah yang ada pada sistem. Cara ini digunakan untuk menghindari serangan terhadapa cloudfri yang dapat memberikan kerugian kepada Fakultas Rekayasa Industri.

Untuk menemukan celah – celah yang terdapat pada cloudfri bisa dilakukan dengan cara melakukan scanning vulnerability. Scanning ini bertujuan untuk memberikan informasi vulnerability apa saja yang terdapat pada cloudfri. Setelah vulnerability diketahui, maka selanjutnya dapat dilakukan analisis vulnerability berdasarkan threat level pada vulnerability tersebut. Hasil analisis yang didapat bisa dijadikan sebagai acuan dalam melakukan hardening pada cloudfri.

Setelah melakukan hardening pada sistem cloudfri dengan mengacu pada vulnerability yang didapat, maka sistem cloudfri akan sulit diserang karena celah – celah yang ada pada sistem sudah ditutup. Penutupan celah yang terdapat pada cloudfri akan menghasilkan keamanan lebih pada cloudfri. Hal ini dikarenakan cloudfri sudah menutup jalur yang kemungkinan akan dilakukan penyerangan oleh hacker. Hal ini juga membuat sistem cloudfri akan lebih stabil karena tidak ada gangguan terhadap sistem tersebut.

2. Dasar Teori

2.1 *Hosting*

Hosting adalah sebuah layanan berupa *server*. *Server* dapat disebut sebagai media atau tempat yang akan ditampilkan pada *website*, seperti file dan database yang digunakan[1]. *Hosting* harus bekerja secara penuh dan terus – menerus agar dapat diakses menggunakan internet.

Hosting dibagi menjadi beberapa jenis, *yaitu shared hosting, dedicated hosting, VPS (Virtual Private Server),* dan *colocation server*. Tujuan dari penggunaan *hosting* adalah, agar *website* yang telah dibuat dapat terhubung kedalam jaringan internet, sehingga *website* dapat dilihat oleh orang lain. Pada prinsipnya, jika *hosting* tidak aktif, maka *website* yang sudah ada tidak dapat diakses.

2.2 *Cloud Computing*

Cloud computing adalah suatu model komputasi yang memberikan kemudahan, kenyamanan, dan sesuai dengan permintaan (*on – demand access*) untuk mengakses dan mengonfigurasikan sumber daya komputasi (*network, server, storage, application, and service*) yang bisa dengan cepat dirilis tanpa adanya banyak interaksi dengan penyedia layanan[2]. *Cloud computing* berfokus pada otentikasi data, integritas data, *querying*, dan *outsourcing* data terenkripsi. Penelitian mereka mengatakan bahwa, resiko dapat timbul pada *operational trust modes, resource sharing, new attcak strategies*. Dalam *operational trust modes*, saluran komunikasi terenkripsi digunakan untuk penyimpanan *cloud* dan melakukan komputasi pada data terenkripsi yang disebut sebagai *homomorphic encryption*[3].

Cloud computing adalah model, bukan sebuah teknologi spesifik, menggambarkan operasional dan ekonomi model untuk penyediaan dan penggunaan infrastruktur IT dan layanan terkait. *Cloud computing* adalah arsitektur terdistribusi yang memusat sumber daya *server* pada *platform* yang dapat diskalakan sehingga dapat menyediakan sumber daya dan layanan komputasi sesuai permintaan. *Cloud computing* telah menjadi *platform variabel* bagi perusahaan untuk membangun infrastruktur mereka[3].

Cloud computing memiliki beberapa macam tipe yang biasa digunakan. Berikut merupakan macam – macam tipe cloud computing:

1. *Private Cloud*

Private cloud adalah penggunaan dari teknologi *cloud* yang digunakan oleh satu organisasi atau perusahaan secara *private*. Penggunaan *private cloud* banyak digunakan untuk interaksi bisnis yang sumber dayanya bisa diatur dan dioperasikan oleh organisasi atau perusahaan yang sama.

2. *Community Cloud*
   Community cloud biasa digunakan oleh komunitas, institusi atau organisasi. Penggunaan *community cloud* dikelola oleh internal maupun pihak ketiga suatu perusahaan, sehingga dapat meminimalisir biaya yang digunakan perusahaan.
3. *Public Cloud*
   Public cloud merupakan layanan yang menggunakan model publik, sehingga siapa saja dapat mengakses layanan ini. Teknologi ini tidak memerlukan biaya dan gratis, namun jika berbayar dengan harga tertentu akan mendapatkan manfaat lebih dari layanan tersebut.
4. *Hybrid Cloud*
   Hybrid cloud merupakan gabungan dari layanan *private cloud* dan *public cloud*. Layanan pada jenis ini memiliki interaksi B2B (*Business to Business*) atau B2C (*Business to Consumer*).

2.3 Keamanan Data

Keamanan data telah menjadi bagian dari pengembangan teknologi informasi mengingat bahwa berjuta – juta bit informasi telah dipertukarkan dalam jaringan komputer terutama di internet[4]. Masalah keamanan data dapat diklasifikasikan ke dalam beberapa dimensi. Sedangkan menurut[5], keamanan data adalah dengan pencegahan diri dan deteksi terhadap tindakan pengganggu yang tidak dikenali dalam sistem komputer.

Keamanan data adalah perlindungan data di dalam suatu sisten melawan terhadap otorisasi tidak sah, modifikasi, atau perusakan dan perlindungan sistem terhadap penggunaan tidak sah atau modifikasi. Ada empat aspek utama dalam keamanan data yaitu:

1. *Privacy/Confidentiality* yaitu usaha menjaga data informasi yang bersifat pribadi dari orang yang tidak berhak mengakses.
2. *Integrity* yaitu usaha untuk menjaga data atau informasi tidak diubah oleh yang tidak berhak.
3. *Authentication* yaitu usaha atau metode untuk mengetahui keaslian dari informasi misalnya, apakah informasi yang dikirm, dibuka oleh orang yang benar atau layanan dari *server* yang diberikan benar berasal dari *server* yang dimaksud.
4. *Availability* berhubung dengan ketersidiaan sistem dan data (informasi) ketika dibutuhkan.

2.4 *Security Hardening*

Security hardening dapat didefinisikan sebagai proses, metodologi, produk, atau kombinasi apapun yang digunakan untuk menambahkan fungsionalitas keamanan, menghilangkan kerentanan, dan mencegah eksploitasi dalam perangkat lunak[6]. *Security hardening* merupakan metode yang digunakan untuk mengamankan suatu sistem dengan cara melakukan pengerasan pada celah sistem yang rentan terhadap serangan. Proses *hardening* menuntut pendekatan metodis untuk mengaudit, mengidentifikasi, menutup, dan mengontrol potensi kerentanan keamanan di seluruh sistem yang ada. Tujuan *hardening* adalah untuk menghilangkan risiko sistem keamanan komputer.

Hardening memiliki lima macam jenis yang biasa digunakan oleh organisasi maupun individu, yaitu:

1. *Network Hardening*
   Network hardening merupakan penguatan yang dilakukan untuk jaringan. Untuk menerapkannya, pastikan firewall dikonfigurasi dengan benar dan semua rule (aturan) diaudit secara teratur.
2. *Server Hardening*
   Server hardening merupakan penguatan yang dilakukan pada *server*. Biasanya dilakukan dengan menempatkan semua *server* di pusat data yang aman.
3. *Application Hardenig*
   Application hardening merupakan penguatan pada aplikasi. Penguatan ini dilakukan dengan menghapus semua komponen atau fungsi yang tidak dibutuhkan.
4. *Database Hardening*
   Database hardening merupakan penguatan pada basis data atau *database*. *Hardening* ini diterapkan pada *database* dengan membuat batasan admin, misalnya dengan mengontrol akses istimewa yang dapat dilakukan pengguna dalam *database*.
5. *Operating System Hardening*
   Operating system hardening dilakukan dengan menerapkan *update* (pembaruan) OS, paket layanan, dan *patch* secara otomatis.

2.5 OWASP

Open Web Application Security Project (OWASP) adalah sebuah organisasi internasional yang bersifat non-profit, didirikan oleh OWASP foundation pada 21 April 2004 di Ameriaka Serikat. Keanggotaan OWASP berasal dari para ilmuwan, peneliti, dan sektor swasta yang menerbitkan laporan artikel, Penerapan *Framework* OWASP dan Network Forensics untuk Analisis, Deteksi, dan Pencegahan Serangan Injeksi di Sisi Host-Based 10 alat/peralatan, dan dokumen yang bersifat open source. OWASP fokus pada peningkatan keamanan perangkat lunak dan didedikasikan untuk memungkinkan organisasi dalam mengembangkan, memperoleh, mengoperasikan, dan memelihara aplikasi terpercaya untuk menjamin keamanan yang dibuat atau dikembangkan. OWASP memiliki misi untuk mengamankan software, sehingga orang-orang dan organisasi dapat membuat keputusan terhadap resiko keamanan yang benar. OWASP merupakan vendor netral yang tidak berafiliasi dengan perusahaan teknologi manapun, tidak mendukung atau merekomendasikan produk atau layanan komersial. Proyek yang sudah dibuat dan dipublikasikan ada 363 proyek dan semua berkaitan dengan keamaan aplikasi, diantara proyek tersebut yaitu OWASP Top Ten Project, OWASP ASVS Assessment tool, OWASP Zed Attack Proxy Project, OWASP Testing Guide [14].

2.6 *Vulnerability*

Vulnerability adalah kelemahan dalam aplikasi yang dapat berupa *bug* implementasi atau cacat desain yang memungkinkan *attacker* membahayakan pengguna aplikasi dan mendapatkan hak istimewa tambahan[7]. *Vulnerability* adalah potensi yang beresiko bagi sistem. *Attacker* dapat menggunakan *vulnerability* ini untuk *mengexploit* sistem dan mendapatkan akses untuk informasi yang tidak sah.

Vulnerability yang sering dilakukan *exploit* biasanya berada pada *level* tingkat *software*, karena dari *exploit* bisa dilakukan dengan sebuah *remote access*. Berikut merupakan *vulnerability* yang menjadi *target* oleh *hackker*:

1. *Firmware*
   Firmware adalah sebuah *software* atau disebut *mini operating system* yang telah tertanam langsung pada *hardcoded* kedalam sebuah *chip* pada suatu perangkat tertentu.
2. *Operating System*
   Operating system dengan keamanan apapun seperti *Linux/Mac* tetap memiliki celah keamanan. Untuk meminilamlisir celah pada *operating system* diupayakan untuk mengaktifkan fitur *automatic update* agar *operating system* selalu melakukan *update* ketika telah tersedia.
3. *Software*
   Software yang telah di instal pada komputer dapat menjadi jalan masuk *hacker* terutama pada *software* yang telah terhubung langsung ke internet.
4. *Brainware*
   Brainware merupakan seseorang yang mengoperasikan komputer, apabila pengguna tersebut memiliki pemahaman yang kurang, maka akan mempermudah seseorang untuk meretas dan mengambil informasi yang dimiliki.

2.7 *Exploit*

Exploit merupakan salah satu teknik yang sering digunakan para *hacker* untuk menyerang keamanan sebuah sistem[8]. Sedangkan menurut [11] *exploit* adalah sebuah perangkat lunak yang berfungsi untuk menyerang jaringan komputer, namun tidak selalu bertujuan untuk melancarkan aksi yang tidak diinginkan.

Ada beberapa metode untuk mengklasifikasikan *exploit*. Cara untuk mengklasifikasikan paling umum adalah dengan melihat cara *exploit* membuat kontak dengan *software* yang rentan. *Remote exploit* bekerja melalui jaringan dan *mengexploit* celah keamanan tanpa adanya akses terlebih dahulu ke sistem korban. *Local exploit* mengharuskan adanya akses terlebih dahulu ke sistem yang rentan dan biasanya meningkatkan keleluasaan orang yang melakukan *exploit*. Exploit pada umumnya dikategorikan dan dinamai berdasarkan kriteria berikut:

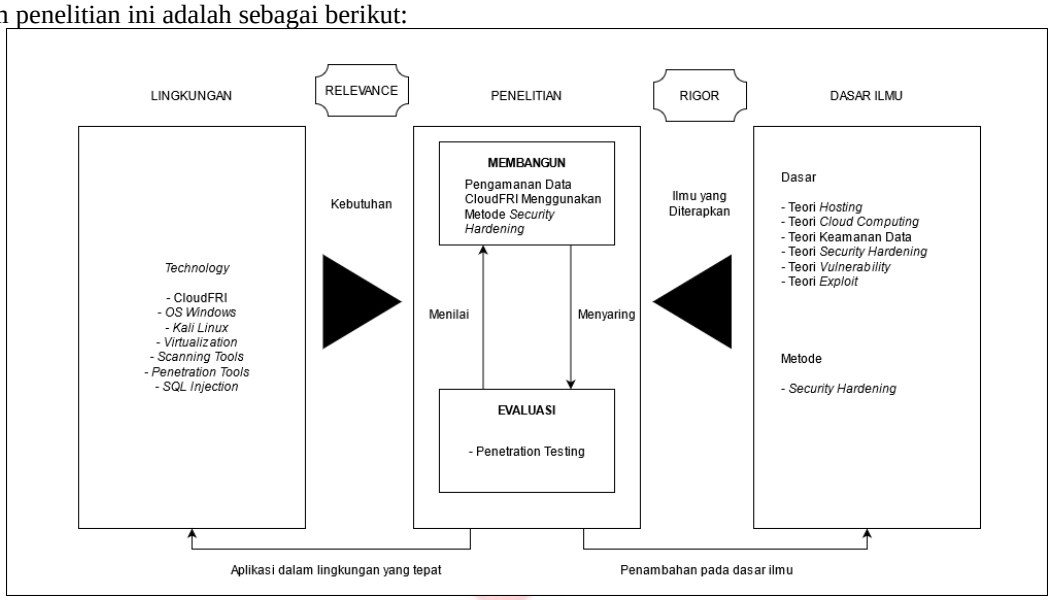
1. Jenis celah keamanan yang mereka *exploit*
2. Apakah *exploit* perlu dijalankan pada mesin yang sama dengan program yang memiliki celah atau dapat dijalankan pada satu mesin berbeda untuk menyerang program yang berjalan pada komputer lain
3. Hasil dari menjalankan *exploit* (*EoP*, *DoS*, *Spoofing*, dll)

3. Metodelogi Penelitian

3.1 Model Konseptual

Model konseptual adalah suatu diagram dari satu set hubungan antara faktor – faktor tertentu yang memberi dampak terhadap suatu kondisi target. Tujuan dari model ini adalah untuk mewujudkan sebuah kerangka

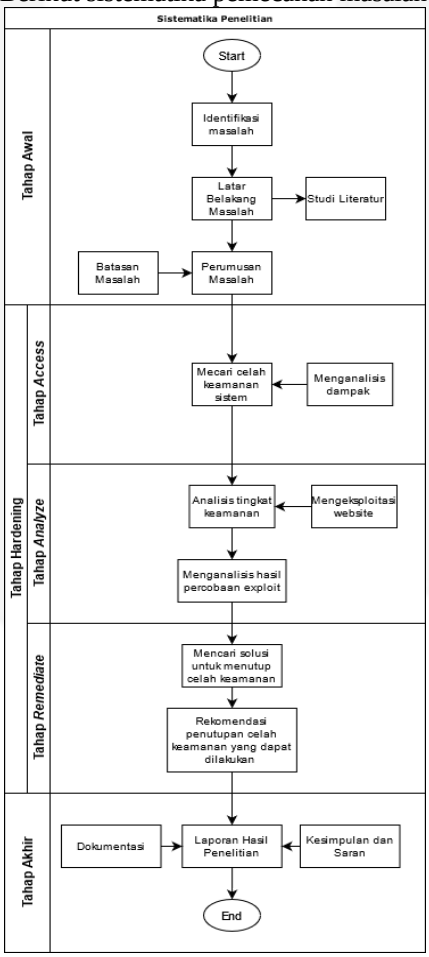
terstruktur yang digunakan untuk memahami tujuan dari sebuah penelitian. Model konseptual yang digunakan dalam penelitian ini adalah sebagai berikut:



Gambar 1 Model Konseptual

3.2 Sistematika Penelitian

Sistematika penyelesaian masalah merupakan pemecahan masalah, sebuah proses terencana yang dilakukan untuk mencapai tujuan penelitian. Berikut sistematika pemecahan masalah yang dilakukan:

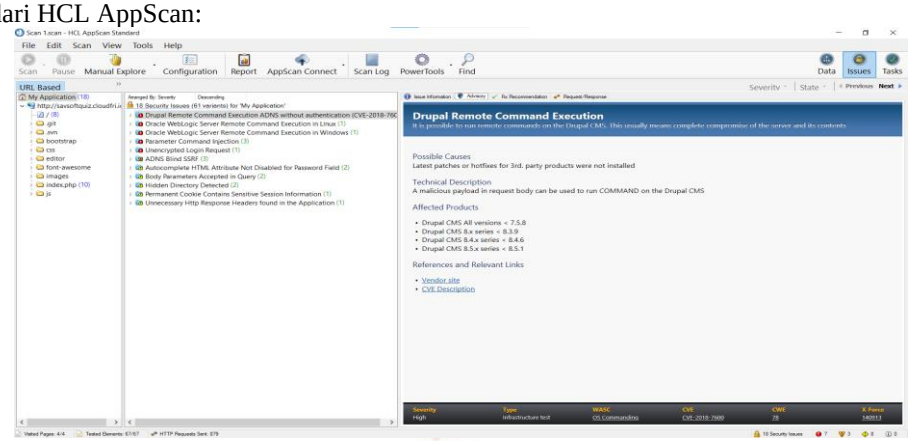


Gambar 2 Sistematika Penelitian

4. Hasil dan Analisis

4.1 Hasil dan Analisis *Vulnerability* Berdasarkan HCL AppScan

Scanning yang dilakukan menggunakan HCL AppScan pada objek savsoftquiz.cloudfri.id menunjukan bahwa savsoftquiz.cloudfri.id memiliki beberapa *vulnerability*. Data yang ditampilkan oleh HCL AppScan berupa *vulnerability*, *threat level*, dan CWE (*Common Weakness Enumeration*). Beberapa *vulnerability* memiliki *threat level high*, hal ini menunjukan bahwa objek savsoftquiz.cloudfri.id rentan terhadap serangan. Berikut hasil scan dari HCL AppScan:



Gambar 3 Hasil *Scanning* dari HCL AppScan

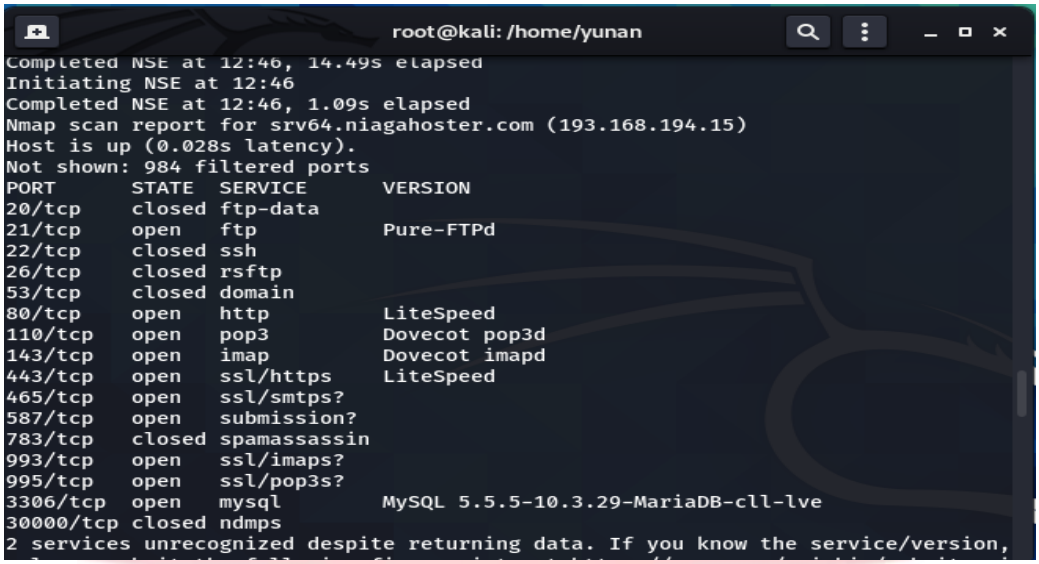
Dari hasil *scan* pada objek yang dilakukan HCL AppScan didapatkan *vulnerability*, CWE, dan *threat level*. Pada *vulnerability* terdapat deskripsi tentang jenis *vulnerability* dan cara penanganan *vulnerability*. Terdapat juga icon berwarna merah yang menandakan bahwa *vulnerability* tersebut memiliki *threat level high*. Selain itu terdapat juga CWE ID, ID tersebut memberikan *indikasi bug*, *flaws*, dan *fault* yang terdapat pada objek. Berikut merupakan data vulnerability dari hasil scan yang dilakukan oleh HCL AppScan:

Tabel 1 Data *Vulnerability* dari Hasil *Scan* Menggunakan HCL AppScan

No	*Vulnerability*	*Common Weakness Enumeration*	*Threat Level*
1.	*Drupal Remote Command Execution ADNS without authentication (CVE-2018-7600)*	78	*High*
2.	*Oracle WebLogic Server Remote Command Execution in Linux*	78	*High*
3.	*Oracle WebLogic Server Remote Command Execution in Windows*	78	*High*
4.	*Parameter Command Injection*	78	*High*

4.2 Hasil dan Analisis *Vulnerability* Berdasarkan Nmap

Scanning yang dilakukan menggunakan Nmap pada objek savsoftquiz.cloudfri.id menunjukan bahwa savsoftquiz.cloudfri.id memiliki beberapa *port* yang terbuka. Nmap menggunakan DNS dari objek untuk mendapatkan *port* apa saja yang terbuka. DNS dari savsoftquiz.cloudfri.id adalah 193.168.194.15. Dengan diketahuinya DNS objek, maka Nmap dapat melakukan scanning untuk mencari port apa saja yang terbuka pada objek. Berikut merupakan port yang terbuka pada savsoftquiz.cloudfri.id:



Gambar 4 *Port* Terbuka savsoftquiz.cloudfri.id

Hasil *scan* yang dilakukan oleh Nmap melalui DNS 193.168.194.15 menunjukan *port*, *state*, *service*, dan *version* dari DNS yang telah di *scan*. Terdapat beberapa *port* yang terbuka pada savsoftquiz.cloudfri.id diantaranya *port* 21, *port* 80, *port* 110, *port* 143, *port* 443, *port* 465, *port* 587, *port* 993, *port* 995, dan *port* 3306. Fungsi dari setiap *port* adalah sebagai berikut[9]:

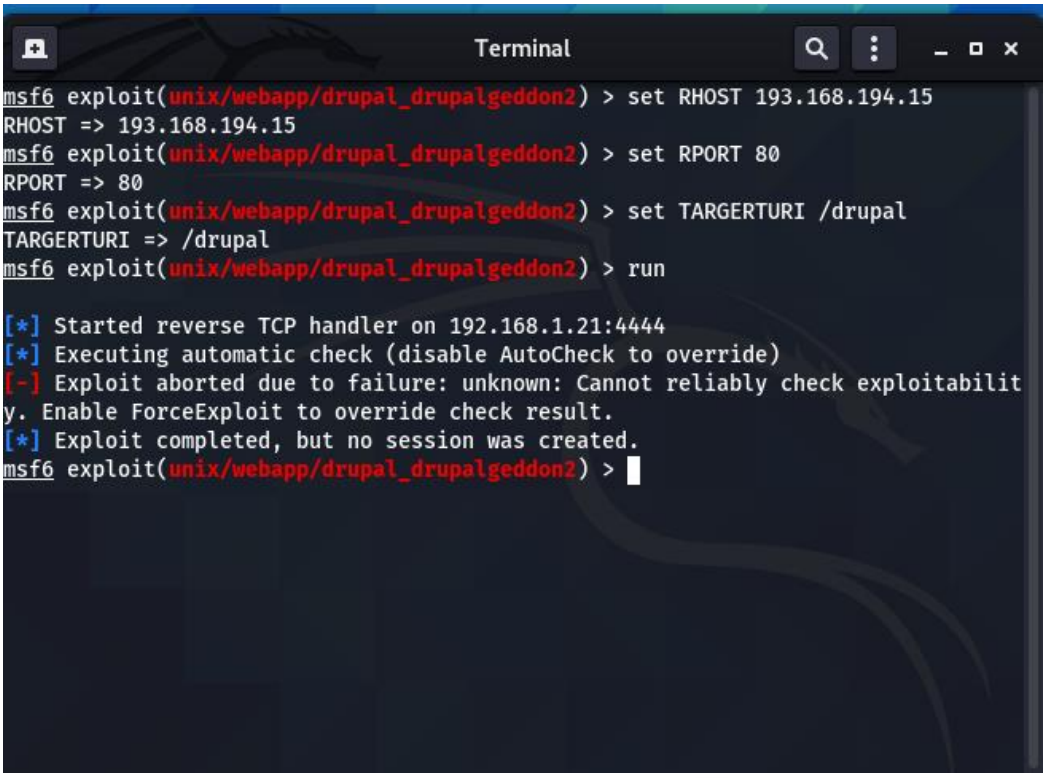
Tabel 2 Daftar *Port* dan Fungsinya

Port	**Fungsi**
21	*Port* ini digunakan untuk koneksi FTP (*File Transfer Protocol*), *port* ini digunakan oleh FTP klien untuk melakukan koneksi ke FTP *server* ketika *user* akan mengakses FTP server.
80	*Port* ini digunakan untuk mengkoneksikan *web server*, paling umum digunakan untuk mengakses internet atau biasa disebut HTTP *port server*.
110	*Port* ini digunakan untuk kepentingan surat menyurat secara elektronik atau email, *port* ini melayani konektivitas jaringan yang menggunakan protokol POP3 atau IMAP4.
143	*Port* ini melayani IMAP4 *server*. Berbeda dengan *port* 110, *port* ini hanya melayani pengambilan informasi atau email menggunakan *server* IMAP4 saja dan tidak menggunakan POP3.
443	*Port* ini bertugas sebagai pintu komunikasi data ke *server* yang menggunakan protokol HTTPS
465	*Port* ini digunakan untuk SSL (*Secure Socket Layer*)
587	*Port* ini digunakan untuk MSA (*Mail Submission Agent*)
993	*Port* ini merupakan *port default* IMAP dan terenkripsi oleh SSL/TLS
995	*Port* ini merupakan *port default* POP3 dan terenkripsi oleh SSL/TLS
3306	*Port* ini digunakan untuk melakukan proses manajemen *database*

4.3 Hasil dan Analisis *Exploit* Berdasarkan Metasploit

Pada tahap ini membahas tentang pengujian yang dirancang untuk melakukan *exploit* terhadap *vulnerability Drupal Remote Command Execution ADNS without authentication* (*CVE-2018-7600*) dengan menggunakan beberapa *module* yang ada pada Metasploit.

Exploit yang dilakukan pada objek menggunakan beberapa module yang terdapat pada Metasploit. Salah satu module yang terdapat pada Metasploit adalah module drupalgeddon2. Module ini digunakan untuk melakukan exploit pada objek untuk mengetahui apakah objek dapat diakses dari jarak jauh oleh penyerang, module ini juga dapat menunjukan username serta password yang terdapat pada objek.



Gambar 5 Hasil *Exploit* Pada savsoftquiz.cloudfri.id Menggunakan Metasploit

Target dari *exploit* serta hasil dari *exploit* yang dilakukan pada savsoftquiz.cloudfri.id. *Target* yang dituju adalah 193.168.194.15 dan menggunakan *port* 80 sebagai jalur untuk melakukan *exploit*. Hasil dari *exploit* menunjukan bahwa Metasploit tidak mendapatkan *session* ketika melakukan *exploit*. Berdasarkan hasil tersebut bisa dikatakan bahwa *exploit* yang dilakukan terhadap savsoftquiz.cloudfri.id gagal karena tidak menampilkan informasi apapun dari objek. Hal ini bukan tanpa sebab, kegagalan *exploit* ini disebabkan savsoftquiz.cloudfri.id yang sudah terinstal *firewall* (Imunify360). *Firewall* inilah yang menghentikan *exploit* karena *firewall* menganggap bahwa *exploit* yang dilakukan merupakan ancaman untuk savsoftquiz.cloudfri.id.

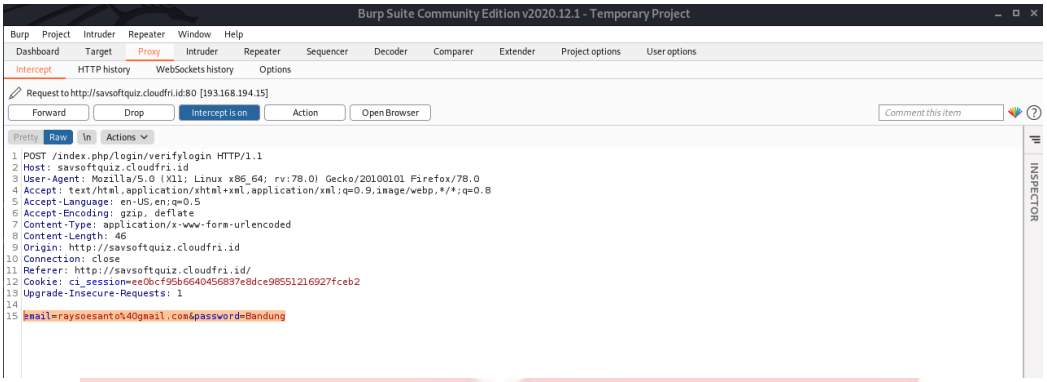
Dengan *firewall* yang sudah terinstal, savsoftquiz.cloudfri.id dapat terhindar dari beberapa serangan. Walaupun *exploit* yang dilakukan menggunakan Metasploit mendapatkan kegagalan karena terdapat *firewall* yang terpasang pada savsoftquiz.cloudfri.id. savsoftquiz.cloudfri.id tidak bisa hanya mengandalkan *firewall* saja. Hal ini dikarenakan masih banyak jenis serangan yang bisa menembus keamanan dari *firewall*. Dengan ini maka savsoftquiz.cloudfri.id perlu melakukan penambahan pengamanan. Dengan mengacu pada salah satu jurnal yang berjudul "SECURITY AUDITING PADA VULNERABILITY MACHINE MENGGUNAKAN OPEN SOURCE IDS DAN VULNERABILITY SCANNER BERDASARKAN NIST CYBERSECURITY FRAMEWORK"[10], serta solusi yang diberikan oleh HCL AppScan, maka rekomendasi penambahan pengamanan untuk savsoftquiz.cloudfri.id adalah sebagai berikut:

1. Melakukan peningkatan Drupal ke versi yang lebih tinggi
2. Terapkan Prinsip Privilege Terkecil ke semua sistem dan layanan.

4.4 Hasil dan Analisis *Exploit* Berdasarkan Burp Suite

Pada tahap ini membahas tentang pengujian yang dirancang untuk melakukan *exploit* terhadap *vulnerability Parameter Command Injection* dengan menggunakan metode *intercept* pada Burp Suite. Serangan *intercept* digunakan untuk mengumpulkan informasi dari komunikasi antara *website* dengan *server*. Informasi yang didapatkan dengan melakukan *intercept* bisa berupa *username*, *password*, dan *session coockies*. Dengan informasi yang didapatkan tersebut, attacker dapat melakukan serangan secara remote (jarak jauh).

Pada kasus ini, *exploit* yang dilakukan menggunakan Burp Suite adalah melakukan *intercept* pada savsoftquiz.cloudfri.id disaat berjalannya *request* untuk login. *Exploit* dilakukan untuk mengetahui *email* serta *password* yang digunakan pada saat melakukan *login* kedalam savsoftquiz.cloudfri.id.



Gambar 6 *Intercept* Burp Suite

Hasil *intercept* menunjukan bahwa *email* serta *password* tidak terenkripsi pada saat melakukan *request login* ke savsoftquiz.cloudfri.id. Dengan *email* dan *password* yang tidak terenkripsi, savsoftquiz.cloudfri.id akan mudah terkena serangan. Hal ini dikarenakan attacker mendapatkan informasi yang bersifat *critical* dari savsoftquiz.cloudfri.id.

Email serta *password* yang tidak terenkripsi bisa terjadi karena komunikasi dari savsoftquiz.cloudfri.id tidak terenkripsi oleh *SSL/TLS*. Komunikasi yang tidak terenkripsi oleh *SSL/TLS* sangat berbahaya dikarenakan dapat menunjukan informasi yang sangat jelas dari savsoftquiz.cloudfri.id.

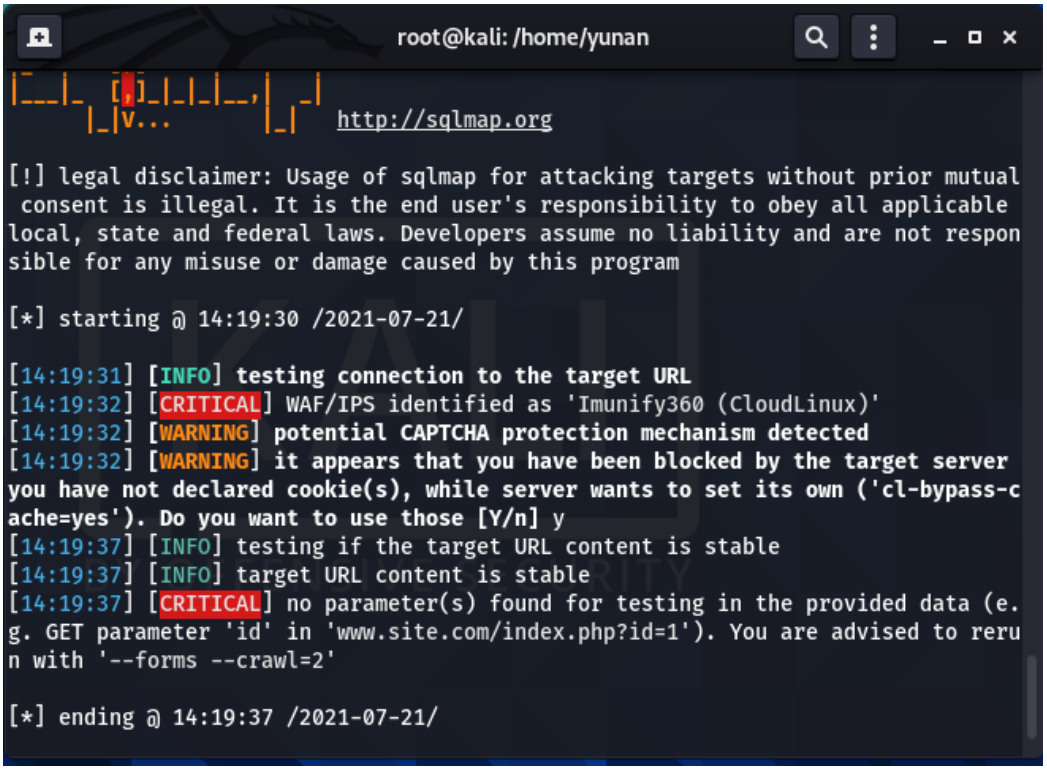
Penanganan dari tidak terenkripsinya *email* serta *password* pada savsoftquiz.cloudfri.id dapat dilakukan dengan beberapa cara. Dengan mengacu pada salah satu jurnal dengan judul "Analisa Keamanan Web Server dari Serangan Remote Os Command Injection pada Instansi Pemerintahan Kota Banda Aceh"[12], serta solusi yang diberikan oleh HCL AppScan, maka rekomendasi untuk penanganan dari *vulnerability* yang ada pada savsoftquiz.cloudfri.id adalah sebagai berikut:

1. Mewaspadai aktifitas yang berusaha masuk ke sistem jaringan melalui notifikasi *alert* IDS Snort.
2. Melakukan validasi terhadap Url dan *memfillter* bentuk *request* yang mengarah terhadap tindakan *command injection*.
3. Melakukan *Input Validation*, dengan mengasumsikan bahwa semua *input* adalah berbahaya.
4. Melakukan *Environment Hardening*, dengan menjalankan kode menggunakan *lowest privileges* yang diperlukan untuk menyelesaikan tugas yang diperlukan.
5. Melakukan enkripsi terhadap semua komunikasi yang dijalankan.

4.5 Hasil dan Analisis *Exploit* Berdasarkan Sqlmap

Pada subbab ini membahas tentang pengujian yang dirancang untuk melakukan *exploit* terhadap objek savsoftquiz.cloudfri.id dengan menggunakan metode *SQL Injection* pada Sqlmap. Serangan *SQL Injection* adalah serangan yang merubah *query* normal pada aplikasi menjadi *query* berbahaya yang memungkinkan pengaksesan dan pemrosesan *database* secara tidak normal (Nugraha, Djanali, & Pratomo, 2013).

Serangan *SQL Injection* yang dilakukan terhadap savsoftquiz.cloudfri.id menggunakan *command* "sqlmap -u http://savsoftquiz.cloudfri.id --dbs" adalah sebagai berikut:



Gambar 7 *SQL Injection* Pada savsoftquiz.cloudfri.id

Hasil *exploit* menunjukan bahwa *SQL Injection* yang dilakukan Sqlmap pada savsoftquiz.cloudfri.id mengalami pemblokiran. Pada Gambar 7 terdapat pernyataan "[CRITICAL] WAF/IPS identified as Imunify360 (CloudLinux)". Hal ini menandakan bahwa savsoftquiz.cloudfri.id sudah terpasang sebuah WAF (*Web Application Firewall*) dengan nama Imunify360 sehingga serangan *SQL Injection* yang dilakukan terhadap savsoftquiz.cloudfri.id terblokir. Pemblokiran terjadi karena *firewall* menganggap bahwa *command SQL Injection* yang dimasukkan berbahaya untuk savsoftquiz.cloudfri.id.

Dengan hasil tersebut dapat disimpulkan bahwa untuk saat ini savsoftquiz.cloudfri.id tidak perlu menambahkan pengamanan untuk pencegahan terhadap serangan *SQL Injection* dikarenakan sudah terpasangnya WAF. Hasil pengujian dari jurnal berjudul "Implementasi Web Application Firewall Dalam Mencegah Serangan Injection Pada Website"[13], juga mengalami kegagalan dalam melakukan serangan *SQL Injection* karena WAF yang sudah terpasang pada website. Hal ini dapat diartikan bahwa keamanan savsoftquiz.cloudfri.id sudah cukup untuk menahan *SQL Injection*.

5. Kesimpulan

Setalah dilakukannya pengujian menggunakan scanning tools serta melakukan exploit pada savsoftquiz.cloudfri.id, didapatkan hasil analisis Pengamanan Data Cloudfri menggunakan Security Hardening dan kesimpulan yang didapat adalah sebagai berikut:

1. Analisis dari hasil *scanning* menggunakan HCL AppScan mendapatkan beberapa *vulnerability* pada savsoftquiz.cloudfri.id yaitu *Drupal Remote Command Execution ADNS without authentication (CVE-2018-7600), Oracle WebLogic Server Remote Command Execution in Linux, Oracle WebLogic Server Remote Command Execution in Windows,* dan *Parameter Command Injection* yang masing – masing *vulnerability* memiliki *threat level high* serta CWE id dengan nomer 78 pada setiap *vulnerability*.
2. Hasil dari *exploit* menggunakan metode *Drupal* melalui *port* 80 sebagai jalur *exploit* pada savsoftquiz.cloudfri.id untuk membuktikan *vulnerability Drupal Remote Command Execution* mengalami kegagalan, sehingga data yang diinginkan dari pengujian *exploit* tidak mendapatkan informasi apapun.
3. Hasil dari *exploit* menggunakan Sqlmap dengan menggunakan metode *SQL Injection* mengalami pemblokiran yang disebabkan sudah terpasangnya WAF (*Web Application Firewall*) pada *website* savsoftquiz.cloudfri.id.

Referensi:

[1] Muharam Masyhar. 2020. *Implementasi Pengguna Website Sebagai Media Informasi dan Media Pemasaran Hasil pertanian dan Peternakan Desa Sumberejo*. Yogyakarta. Universitas Islam Indonesia.

[2] I Gusti Ngurah Wikranta Arsa. 2019. *Analisis Sistem Cloud Computing IAAS Penyedia Server Cloud dengan Standar NIST Special Publication 800 – 145*. Bali. STMIK STIKOM Bali.

[3] Shakeeba S. Khan. 2015. *Security in Cloud Computing using Cryptographic Algorithms*. International Journal of Innovative Research in Computer and Communication Engineering.

[4] Amir Mahmud Hasibuan. 2017. *Rancang Bangun Aplikasi Keamanan Data Menggunakan Metode AES Pada Smartphone*. Medan. STMIK Budi Darma.

[5] Amat Suroso. 2016. *Implementasi Keamanan Data Dengan Algoritma Kunci Simetris Rijndael Menggunakan VB.NET 2008*. STMIK Bani Saleh.

[6] Azzam Mourad, Marc-Andre Laverdiere, dan Mourad Debbabi. 2006. *Security Hardening of Open Source Software*. Canada. Concordia University.

[7] Jai Narayan Goel dan BM Mehtre. 2015. *Vulnerability Assessment & Penetration Testing as a Cyber Defence Technology*. India. University of Hyderabad.

[8] Muhammad Alkhayudi. 2019. *Analisis dan Perancangan Keamanan Akses Router Mikrotik dari Serangan Exploit*. UIB.

[9] Dicky Septian Firdaus, Ritzkal, dan Ade Hendri Hendrawan. 2019. *Analisis Keamanan Vulnerability pada Server Cloud Open Media Vault di Fakultas Teknik Universitas Ibn Khaldun Bogor*. Bogor. Universitas Ibn Khaldun.

[10] Heri Sultan Fransiscus Sitinjak, Umar Yunan Kurnia Septo Hediyanto, dan Aditya Widjajarto. 2020. *Security Auditing in Vulnerability Machine Using Open Source IDS and Vulnerbaility Scanner Based on NIST Cyber Security Framework*. Bandung. Universitas Telkom.

[11] Prama Wira Ginta, Galih Putra Kusuma, dan Edi Kusuma Negara. 2013. *Implementasi Tools Network Mapper Pada Lokal Area Network (LAN)*. Bengkulu. Universitas Dehasen.

[12] Lisa Handasari Yanti, Iqbal, dan Banta Cut. 2019. *Analisa Keamanan Web Server dari Serangan Remote Os Command Injection pada Instansi Pemerintahan Kota Banda Aceh*. Aceh Besar. Universitas Abulyatma.

[13] Bangkit Wiguna, Wahyu Adi Prabowo, dan Ridho Ananda. 2020. *Implementasi Web Application Firewall Dalam Mencegah Serangan SQL Injection Pada Website*. Purwokerto. Institut Teknologi Telkom.

[14] Ade Kurniawan. 2019. *Penerapan Framewaork OWASP dan Network Forensics untuk Analisis, Deteksi, dan Pencegahan Serangan Injeksi di Sisi Host-Based*. Batam. Universitas Universal