VULNERABILITY ASSESMENT UNTUK MENCARI CELAH KEAMANAN WEB APLIKASI E-LEARNING PADA UNIVERSITAS XYZ

Muhammad Aziz
S1 Informatika, Universitas Teknokrat Indonesia
Jl. ZA Pagaralam, No 9-11, Labuhanratu, Bandarlampung
Email: azizshotokanryu212@gmail.com

Abstrak

Penelitian ini dilakukan atas dasar perkembangan teknologi informasi dan komunikasi pada saat ini membawa kemudahan bagi kehidupan manusia. Salah satu hal yang berkembang cukup pesat adalah aplikasi berbasis web. Menjamurnya aplikasi berbasis web menjadi tantangan sendiri bagi para pengembang aplikasi berbasis web dalam mengembangkan aspek keamanan. Vulnerability Assesment terhadap web aplikasi E-Learning bertujuan untuk mendeteksi kerentanan, mendeskripsikan kerentanan, menilai kerentanan berdasarkan CVSS (Common Vulnerability Scoring System), dan memberikan solusi. Tahapan penelitian yang digunakan adalah Vulnerability Assesment and Penetration Testing (VAPT) Life Cycle. Dalam mencari kerentanan pada penelitian ini menggunakan nessus vulnerability scanning versi home.

Dari hasil vulnerability scanning ditemukan kerentanan critical, kerentanan high, kerentanan medium, dan keretanan low. Pada masing-masing kerentanan tentunya memiliki dampak kerentanan yang berbeda, namun pada kerentanan critical yaitu Elasticsearch Transport Protocol Unspecified Remote Code Execution memiliki dampak yang paling serius dengan base score 9.8, sehingga overall risk level pada Web aplikasi E-Learning adalah High. Jadi dapat disimpulkan Web aplikasi E-Learning pada Universitas XYZ dikatakan rentan, karena memiliki dampak serius yang mempengaruhi Confidentiality, Integrity, dan Availability pada Web Aplikasi E-Learning melalui kerentanan yang dimilikinya. Maka pihak Universitas XYZ harus segera melakukan perbaikan dan evaluasi terhadap keamanan pada Web Aplikasi E-Learning agar resiko kerentanan pada Web Aplikasi E-Learning dapat dikurangi.

Kata Kunci: *Nessus*, *Overall Risk Level*, *VAPT Life Cycle*, *Web* Aplikasi *E-Learning*.

1. Pendahuluan

A. Latar Belakang

Perkembangan Teknologi Informasi dan Komunikasi pada saat ini membawa kemudahan bagi kehidupan manusia. Salah satu hal yang berkembang cukup pesat adalah aplikasi berbasis *web*. Aplikasi berbasis *web* dipilih karena aplikasi tersebut dapat berjalan di berbagai *platform* dan juga termasuk aplikasi yang ringan untuk digunakan. Menjamurnya aplikasi berbasis *web* menjadi tantangan sendiri bagi para pengembang aplikasi berbasis *web* dalam mengembangkan aspek keamanan pada aplikasi tersebut (Yudha, *et all.* 2018).

Berdasarkan data Id-SIRTII (*Indonesian Security Incident Response Team on internet Infrastructure*) merilis trafik serangan tahunan di seluruh Indonesia tahun 2018, tercatat ada 10 serangan di *internet* seperti aktivitas *trojan*, percobaan *user*, percobaan *dos*, percobaan pengintaian, percobaan pengintaian yang berhasil, upaya mendapatkan hak *administrator*, pelanggaran kebijakan, *denial of service*, dan serangan yang tidak diketahui (Id-SIRTII/CC 2018).

Kerentanan pada aplikasi berbasis *web* bisa beragam, tergantung dari *module, library, CMS*, dan *database* yang dipakai. Karena aplikasi berbasis *web* terdiri dari banyak komponen, sehingga aplikasi berbasis *web* mempunyai banyak sisi untuk diserang. Maka dari itu diperlukan kegiatan *vulnerability assesment* terhadap celah keamanan pada setiap aplikasi berbasis *web.* Pada peneilitian ini penulis akan melakukan *vulnerability assessment* terhadap *web* aplikasi *E-Learning* Universitas XYZ sebagai objek penelitian. Alasan penulis memilih *web* aplikasi *E-Learning* sebagai objek penelitian, karena *web* aplikasi *E-Learning* memiliki manfaat dalam proses belajar mengajar yang dilakukan secara *online*, seperti memperjelas informasi pada saat tatap muka, melengkapi dan memperkaya informasi dalam pembelajaran, meningkatkan efektifitas dan efesiensi dalam menyampaikan materi, dan mengatasi keterbatasan ruang dan waktu. Tetapi dikhawatirkan proses belajar mengajar yang dilakukan secara *online* dapat dipengaruhi oleh serangan yang dilakukan *attacker* dengan memanfaatkan kerentanan yang ditemukan olehnya. Agar dalam proses belajar mengajar dengan menggunakan *web* Aplikasi *E-Learning* tidak terganggu oleh pihak lain yang dapat menyebabkan proses belajar menjadi terhambat, seperti adanya aktifitas serangan yang menyebabkan sistem *web* aplikasi *E-Learning* menjadi rusak, atau bahkan adanya pencurian informasi data pengguna, maka harus dilakukan evaluasi terhadap celah keamanan *web* aplikasi *E-Learning* agar proses belajar menggunakan aplikasi *E-Learning* menjadi lebih efesien dan tidak terganggu oleh

aktifitas serangan yang dilakukan oleh pihak yang tidak bertanggung jawab.

Berdasarkan uraian yang telah dijelaskan, maka dari itu penulis mengangkat judul "Vulnerability Assesment Untuk Mencari Celah Keamanan Web Aplikasi E-Learning Pada Universitas XYZ", dari hasil *vulnerability scanning* akan menghasilkan laporan yang berisi tentang penemuan jumlah kerentanan, detail tentang setiap temuan, tingkat keparahan resiko, beserta dampak, dan rekomendasi untuk memperbaiki kerentanan tersebut, dan persentase kerentanan yang ditemukan untuk melihat *overall risk level* pada *Web* Aplikasi *E-Learning*. Sehingga penelitian ini dapat digunakan untuk memberikan evaluasi keamanan terhadap *Web* Aplikasi *E-Learning* Universitas XYZ.

B. Landasan Teori

1. *Vulnerability Assesment*

Vulnerability Assesment merupakan fase pendekatan untuk mengidentifikasi kerentanan yang ada dalam infrastruktur. Kerentanan dalam hal *IT System* dapat di defenisikan sebagai kelemahan potensial dalam sistem, atau jika dieksploitasi dapat mengakibatkan realisasi serangan terhadap system (Kumar, 2014).

2. Kategori Vulnerability Assesment

Menurut GOV-CSIRT (*Government Computer Security Incident Response Team*) pada tahun 2021,. *Vulnerability* atau kerentanan dibagi menjadi tiga penilaian yaitu:

1. *Level High* (Tinggi)
   Pada *level* ini terdapat kelemahan yang berpotensi tinggi menjadi ancaman, sedangkan fitur atau langkah untuk tingkat pencegahan maupun penanganannya tidak memadai.
2. *Level Medium* (Sedang)
   Pada *level* ini tingkatan kelemahan bersifat lokal dan upaya penanganan dan pencegahan bersifat lokal juga.
3. *Level Rendah* (Low)
   Pada *level* ini kelemahan rendah, upaya pencegahan dan penanganan yang diharapkan sangat memadai.

3. Jenis – Jenis *Vulnerability Assesment*

Menurut ISACA (2012), Terdapat empat jenis *vulnerability assessment* yaitu:

1) *Network Based Scans*
   Network Based Scans digunakan untuk mengidentifikasi kemungkinan serangan keamanan jaringan. Dikarenakan pada pemindaian ini dapat menyebutkan layanan yang berjalan, memindai berbagai *port TCP* yang mendengarkan, memeriksa *banner* sistem atau menggunakan sejumlah teknik lain untuk menentukan jenis dan versi *host* atau perangkat.
2) *Host Based Scans*
   Host Based Scans digunakan untuk menemukan dan mengidentifikasi kerentanan pada *server*. Pemindaian ini dapat memberikan visibilitas yang lebih besar kepengaturan konfigurasi sistem dan *detail* tambahan, sambil mencakup *port* dan layanan juga terlihat oleh *network based scans*, tetapi menawarkan visibilitas yang lebih besar ke pengaturan konfigurasi dan *patch history* dan sistem *scan*.
3) *Wireless Network Scans*
   Wireless Network Scans dari jaringan *WI-FI* organisasi biasanya berfokus pada titik-titik serangan dalam infrastruktur jaringan nirkabel. Selain pemindaian jaringan nirkabel juga dapat memvalidasi bahwa jaringan perusahaan atau organisasi dikonfigurasi dengan aman.
4) *Application Scans*
   Application Scans dapat digunakan untuk menguji situs *web* dalam mendeteksi kerentanan perangkat lunak yang diketahui dan konfigurasi yang salah dalam aplikasi jaringan atau *web*.

4. Manfaat *Vulnerability Assesment*

Menurut ISACA (2012), Manfaat keamanan pada *vulnerability assessment* adalah:

1) Mengurangi sebagian besar resiko keamanan suatu perusahaan.
2) Mendapatkan wawasan berharga tentang lingkungan perusahaan atau organisasi.
3) Hasil *vulnerability assessment* bisa digunakan tidak hanya untuk menargetkan rencana remediasi tetapi juga untuk menunjukan masalah sistematik seperti kesenjangan dalam manajemen *patch* atau manajemen siklus hidup *asset*.
4) Mendeteksi kesalahan konfigurasi jaringan, dan menemukan layanan yang tidak sah yang berjalan pada sistem internal.

5. *CVSS* (*Common Vulnerability Scoring System*)

Menurut Kumar (2014), *CVSS* (*Common Vulnerability Scoring System*) merupakan sistem penilaian yang digunakan untuk menilai suatu kerentanan terhadap sistem. Pada Tabel 1 mencantumkan skor *CVSS* berdasarkan peringkat kerentanan.

Tabel 1. *List Common Vulnerability Scoring System*

CVSS Score	Criticality
0,0	Info
0,1 – 3,9	Low
4,0 – 6,9	Medium
7,0 – 8,9	High
9,0 – 10, 0	Critical

Pada Tabel 1 merupakan tabel yang menjelaskan *CVSS Score* menjadi rujukan dari kerentanan yang ditemukan, masing-masing *score* terdapat tingkat penilaian untuk *Info*, *Low*, *Medium*, *High*, dan *Critical*. *CVSS* menilai beberapa aspek dalam menilai kerentanan suatu sistem yang dibagi menjadi 8 bagian besar yaitu *Attack Vector*, *Attack Complexity*, *Privilege Required*,

User Interaction, *Scope*, *Confidentiality*, *Integrity*, dan *Availability*. *Confidentiality*, *Integrity*, *Availability* menjadi ujung tombak penilaian dari kerentanan suatu sistem, jika suatu kerentanan tidak mengancam ketiga aspek tersebut maka skor kerentannya akan tetap 0 karena tidak mengancam aspek apapun (Yohan, 2018).

2. Penelitian Terkait

A. Tinjauan Pustaka

Penelitian ini dikembangkan dari beberapa refrensi yang telah di dapat dan berhubungan dengan objek permasalah yaitu:

Penelitian yang dilakukan oleh Rizki Nurdin (2017), yang membahas "Analisa Keamanan Internet Menggunakan Nessus dan Ethereal Universitas Putra Indonesia YPTK Padang". Pada penelitian tersebut bertujuan untuk menganalisa keamanan jaringan dengan menggunakan *tool nessus*, dalam melakukan *vulnerability scanning* ditemukan empat *vulnerability* beserta penjelasan dari empat *vulnerability* tersebut. sehingga pada saat dianalisa menggunakan *ethereal* untuk meng*capture* paket yang berjalan pada jaringan ditemukan satu *host* yang sedang melakukan *login* ke *FTP server*, sehingga *user* dan *password* dapat diketahui oleh orang yang tidak bertanggung jawab.

Adapun penelitian yang dilakukan oleh Mia Zattu Maharani, *et al*. (2017), membahas "Analisis Keamanan website menggunakan metode Scanning dan Perhitungan Security Metriks". Dimana pada penelitian tersebut melakukan *Vulnerabilty assesment* dengan menggunakan *accunetix* dan *web* yang diuji adalah *igracias* telkom dan *ppdu* telkom pada kampus telkom *university*. Pada penelitian ini menghasilkan hasil *Vulnerability Scanning* berupa perhitungan *security metrics* yang menampilkan *score* akhir dari hasil *Vulnerability Assesment*.

Sedangkan penelitian yang dilakukan oleh Ari Marta Tania, *et al*. (2018), membahas "Keamanan Website Menggunakan Vulnerability Assesment". Dimana pada penelitian tersebut melakukan evaluasi keamanan untuk menemukan celah-celah kerentanan pada *website* dan memberikan solusi terhadap kerentanan yang ditemukan. Evaluasi keamanan ini mencegah adanya resiko kehilangan data penting serta pengeluaran anggaran tambahan jika *website* terjadi *malfunction* ataupun *crash*. Hasil dari penelitian ini adalah memberikan solusi perbaikan terhadap kerentanan yang ditemukan untuk mengurangi resiko kerentanan.

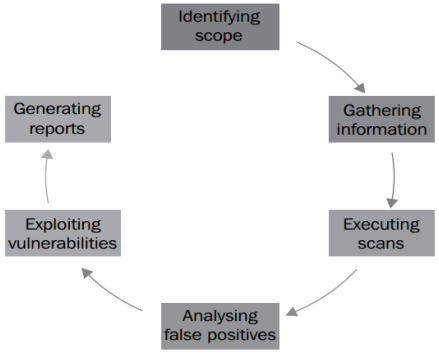
Berikutnya penelitian yang dilakukan oleh Kauka Noor Fathur Rizko (2018), membahas "Security Assesment Menggunakan Tools Nessus Untuk Mencari Celah Keamanan Web Aplikasi Repositori Di Institusi Pendidikan XYZ". Dimana pada penelitian tersebut mengangkat masalah mengenai belum adanya dokumentasi tentang celah keamanan aset, menimbulkan kekhawatiran bagi pihak institusi tersebut, karena celah kerentanan dapat dimanfaatkan oleh penyerang untuk mengambil informasi dari sebuah aset. Salah satu aset penting milik institusi pendidikan XYZ yang belum memiliki dokumentasi tentang celah keamanan adalah *web* aplikasi repositori. Hasil dari penelitian ini adalah menghasilkan laporan yang berisi rekomendasi perbaikan dari kerentanan yang ditemukan untuk keperluan dokumentasi dan evaluasi untuk meningkatkan sistem keamanan pada *web* aplikasi tersebut.

Penelitian lain yang dilakukan oleh Yunanri.W, *et al* (2018), membahas "Analisis Deteksi Vulnerability Pada Webserver Open Jurnal System Menggunakan OWASP Scanner". Dimana pada penelitian tersebut mengangkat masalah mengenai aplikasi *webserver* sering sekali mendapatkan serangan dari berbagai pihak yang tidak bertanggung jawab yang sering disebut *hacker* atau peretas. Berbagai macam alasan *hacker* mencari celah pada *webserver* bertujuan untuk mendapatkan informasi pada sebuah organisasi dan perusahaan untuk kepentingan-kepentingan yang membuat kerugian pada pihak lain. Sehingga hasil dari penelitian ini ditemukan beberapa kerentanan dalam *open* jurnal *system* yang dapat memanipulasi *file* lokal, mengunggah file dengan melakukan serangan *Cross-Site Scripting* (*XSS*).

3. Metode Penelitian

A. Metode Penelitian

Metode penelitian atau tahapan-tahapan yang digunakan adalah *Vulnerability Assesment* dan *Penetration Testing Life Cycle*, yang merupakan bagian yang menjelaskan fase-fase utama dalam *Vulnerability Assesment* dan *Penetration Testing* (Kumar, 2014). Pada *VAPT Life Cycle* terdapat enam tahapan yang dapat dilihat pada Gambar 1.



Gambar 1. Tahapan Penlitian

Pada Gambar 1 menjelaskan tahapan-tahapan yang akan dilakukan dalam penelitian, penjelasan dari tahapan-tahapan penelitian tersebut adalah:

1) *Identyfing Scope*
   Tahap pertama penulis menentukan lingkup penelitian yang akan diteliti, pada penelitian ini penulis menggunakan *Web* aplikasi *E-learning* pada Universitas XYZ sebagai objek dari penelitian yang akan dilakukan.
2) *Information Gathering*
   Tahap kedua merupakan tahapan yang penulis gunakan untuk mengumpulkan informasi tentang sistem target dengan *tools whois*, *dig*, *nslookup*, *NMAP*.

3) *Vulnerability Scanning*
Tahap ketiga merupakan tahapan yang penulis gunakan untuk mencari kerentanan pada *Web* aplikasi *E-learning* dengan menggunakan *tool Nessus*.
4) *False Positive Analysis*
Tahap keempat, dimana hasil pemindaian, penulis akan mendapatkan daftar kerentanan dari *Web* aplikasi *E-learning*. Salah satu kegiatan utama yang harus dilakukan dengan *output* yang menjadi *false positive analysis* yaitu menghilangkan atau memastikan bahwasanya kerentanan yang ditemukan bukan kerentanan yang salah.
5) *Vulnerability Exploitation*
Tahap kelima, merupakan tahapan yang bertujuan untuk menembus sistem target berdasarkan eksploitasi yang tersedia untuk kerentanan yang di identifikasi atau eksploitasi kerentanan yang terkenal yang tersedia secara publik yang dapat dimanfaatkan.
6) *Generating Report*
Tahap keenam merupakan tahapan pembuatan laporan yang berisi tentang kerentanan pada *Web* aplikasi *E-learning*, berserta dampaknya, dan memberikan rekomendasi untuk memperbaiki kerentanan pada *Web* aplikasi *E-learning*.

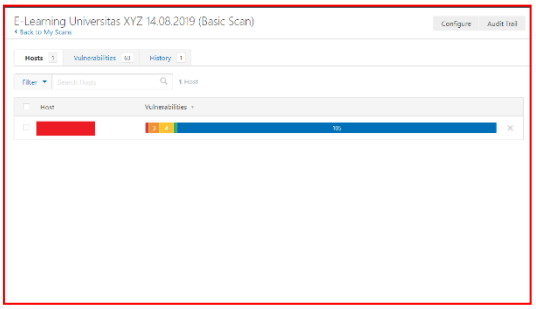
B. Pengujian

Pada penelitian ini dalam melakukan *vulnerability assessment* terhadap *Web* aplikasi *E-Learning* pada Universitas XYZ, penulis menggunakan bantuan aplikasi *Nessus* untuk melakukan pengujian *vulnerability scanning* yang akan menghasilkan daftar kerentanan beserta penjelasan terhadap kerentanan, dampak dari kerentanan, persentase kerentanan yang ditemukan, dan solusi untuk mengatasi kerentanan sehingga dapat digunakan untuk mengevaluasi serta meningkatkan keamanan terhadap *Web* aplikasi *E-Learning* Universitas XYZ.

4. Hasil dan Pembahasan

A. *Vulnerability Scanning* (*Executing Scan*)

Tahapan ini dilakukan pemindaian kerentanan pada *Web* Aplikasi *E-Learning* Universitas XYZ menggunakan *tool Nessus Home*. Hasil pemindaian kerentanan pada *Web* Aplikasi *E-Learning* Universitas XYZ dapat dilihat pada Gambar 2.



. **Gambar 2**. Hasil *Vulnerability Scanning*

Pada Gambar 2 merupakan hasil *vulnerability scanning* yang penulis lakukan terhadap *Web* Aplikasi *E-Learning* Universitas XYZ. *Scanning* ini dilakukan pada Tanggal 14 Agustus 2019, Adapun daftar kerentanan dapat dilihat pada Tabel 2.

Tabel 2. Daftar Kerentanan

No	Nama Kerentanan	Base Score	Tingkat Kerentanan
1.	*Elasticsearch Transport Protocol Unspecified Remote Code Execution*.	9.8	*Critical*
2.	*Elasticsearch ESA-2015-06*.	9.8	*High*
3.	*CGI Generic SQL Injection (blind)*.	7.5	*High*
4.	*Elasticsearch Unrestricted Access Information Disclosure*	5.3	*Medium*
5.	*SSL/TLS Protocol Initialization Vector Implementation Information Disclosure Vulnerability*.	5.3	*Medium*
6.	*Web Application Potentially Vulnerable to Clickjacking*.	4.3	*Medium*
7.	*Web Server Transmits Cleartext Credentials*.	2.6	*Low*

Pada daftar kerentanan yang telah ditemukan, tentunya memiliki dampak kerentanan yang berbeda di setiap kerentanan. Adapun penjelasan dari kerentanan tersebut adalah sebagai berikut:

1) ***Elastisearch Transport Protocol Unspecified Remote Code Execution.*** Kerentanan ini menjelaskan bahwa versi *Elasticsearch* yang digunakan berisi kecacatan yang memungkinkan seorang penyerang dapat mengeksekusi *code* arbitrer melalui serangan *remote*, kerentanan tersebut terjadi pada *port* 9200/*tcp/elasticsearch* yang merupakan *port Elasticsearch* dengan *base score CVSS* 9.8 dan tingkat kerentanannya adalah *critical*.
2) ***Elasticsearch ESA*-2015-06**. Kerentanan ini menjelaskan kerentanan yang sama dengan kerentanan sebelumnya, karena pada kerentanan ini seorang penyerang dapat mengeksekusi *code* arbitrer melalui serangan *remote* dengan memanfaatkan *port* 9200/*tcp/elasticsearch* yang merupakan *port Elasticsearch*.
3) ***CGI Generic SQL Injection (Blind).*** Kerentanan ini menjelaskan bahwa penyerang mungkin dapat melakukan serangan *SQL*

Injection yang dapat memodifikasi *response* dari *web* aplikasi dengan menggunakan penambahan karakter khusus pada *link web* aplikasi, sehingga penyerang dapat mengeksploitasi kerentanan ini untuk membaca data dari basis data, dan bahkan merubah atau memodifikasi basis data dari jarak jauh. Kerentanan ini terjadi pada *port* 80/*tcp*/*www* dan port 8080/*tcp*/*www* dengan *base score* CVSS 7.5 dan tingkat kerentannya adalah *High*.

4) ***Elasticsearch Unrestricted Access Information Disclosure*.** Kerentanan ini menjelaskan bahwa versi *Elasticsearch* yang digunakan dipengaruhi oleh kerentanan pengungkapan informasi dari basis data, dikarenakan pada *Elasticsearch* tidak membatasi sumber daya melalui autentikasi, jadi seorang penyerang dapat mengeksploitasi kerentanan ini sehingga penyerang dapat menghasilkan beberapa informasi dari *database* terutama "*password*". Kerentanan ini terjadi pada *port 9200/tcp/elasticsearch* dengan *base score CVSS* 5.3 dan tingkat kerentanannya adalah *medium*.
5) ***SSL/TLS Protocol Initialization Vector Implementation Information Disclosure Vulnerability*.** Kerentanan ini menjelaskan bahwa dalam protokol *SSLv3/TLSv1* yang diterapkan pada *server* dapat dipengaruhi oleh serangan *BEAST*. Sehingga *web* yang menggunakan layanan terenkripsi (*HTTPS*) dapat memungkinkan serangan *man-in-the-middle* untuk mendapatkan *header HTTP* melalui *Blockwise Chosen Boundary Attack* (*BCBA*) pada sesi *HTTPS*. Kerentanan ini terjadi pada *port 443/tcp/www* dengan *base score* 5.3 dan tingkat kerentanannya adalah *medium*.
6) ***Web Application Pottentially Vulnerable to Clickjacking.*** Kerentanan ini menjelaskan bahwa aplikasi *web* memiliki kerentanan terhadap *Clickjacking*, dimana seorang penyerang dapat menipu pengguna untuk mengklik area halaman rentan yang berbeda dari apa yang dirasakan pengguna sebagai halaman tersebut. Kerentanan ini terjadi pada *port 80/tcp/www* dengan *base score* 4.3 dan tingkat kerentanannya adalah *medium*.
7) ***Web Server Transmits Cleartext Credentials.*** Kerentanan ini menjelaskan bahwa *web server* berisi bidang formulir yang berisi input tipe "Kata Sandi" yang mengirimkan informasi ke *server web* dalam teks jelas sehingga seorang penyerang dapat menyadap dan memperoleh *username* dan *password* yang *valid*.

B. *Vulnerability Port Service*

Berikut ini merupakan *port service* yang memiliki *vulnerability* pada *Web* Aplikasi *E-Learning* Universitas XYZ dapat dilihat pada Tabel 3.

Tabel 3. *Vulnerability Port Service*

Service Port	Threat Level
9200/*tcp/elasticsearch*	*Critical*
80/*tcp/www*	*High*
8080/*tcp/www*	*High*
9200/*tcp/elasticsearch*	*High*
9200/*tcp/elasticsearch*	*Medium*
443/*tcp/www*	*Medium*
80/*tcp/www*	*Medium*
8080/*tcp/www*	*Low*

Pada Tabel 3 menjelaskan bahwa kerentanan *Web* Aplikasi *E-Learning* pada *port* 9200 memiliki tiga kerentanan, dimana pada masing-masing kerentanan memiliki *base score* 9.8, dan 5.3 dengan tingkat kerentanan *critical*, *high*, dan *medium*. Kemudian pada *port* 80 memiliki dua kerentanan, dimana pada masing-masing kerentanan memiliki *base score* 7.5 dan 4.3 dengan tingkat kerentanan *high* dan *medium*, dan pada *port* 8080 memiliki dua kerentanan, dimana pada masing-masing kerentanan memiliki *base score* 7.5 dan 2.6 dengan tingkat kerentanan *high* dan *low*. Sehingga dapat diketahui bahwa pada *Web* aplikasi *E-Learning* memiliki satu kerentanan *critical* (kritis) yaitu pada *Elasticsearch* di *port* 9200, *Elasticsearch* merupakan suatu *database* berbasis *NoSQL* yang berfokus pada *search engine*. Kerentanan pada *Elasticsearch* ini memiliki dampak yang dapat menyebabkan seorang penyerang dapat menginjeksi *code* arbitrer secara *remote* dan bahkan penyerang dapat melakukan pengungkapan informasi dari basis data, dikarenakan pada *Elasticsearch* tidak memiliki autentikasi dalam penggunaan sumber daya sehingga seorang penyerang dapat memanfaatkan kerentanan ini untuk melakukan eksploitasi.

C. *Generating Report*

Generating report merupakan tahapan terakhir yang penulis gunakan untuk membuatkan laporan dari kegiatan *vulnerability assesment*, Tahapan pelaporan merupakan tahap yang paling penting, karena fase ini adalah memberikan rekomendasi tentang temuan hasil identifikasi kerentanan. Fase *reporting* merupakan kegiatan untuk mendokumentasikan hasil identifikasi, sehingga kerentanan yang ditemukan dapat di dokumentasikan dengan baik serta dapat dilakukan tindakan pengurangan resiko dan memberikan rekomendasi kepada pihak Universitas XYZ tentang kerentanan pada *Web* Aplikasi *E-Learning* yang dimilikinya.

Setelah proses identifikasi, ditemukan beberapa tingkat kerentanan pada *Web* Aplikasi *E-Learning* Universitas XYZ yaitu *critical*, *high*, *medium*, dan *low*, tentunya dari masing-masing kerentanan tersebut memiliki dampak kerentanan yang berbeda. sehingga pada tahap pembuatan laporan ini akan mendokumentasikan setiap kerentanan yang ditemukan dan memberikan rekomendasi terhadap kerentanan dalam bentuk laporan seperti yang ditunjukan oleh Tabel 4.

Tabel 4. Laporan *Vulnerability Assesment*

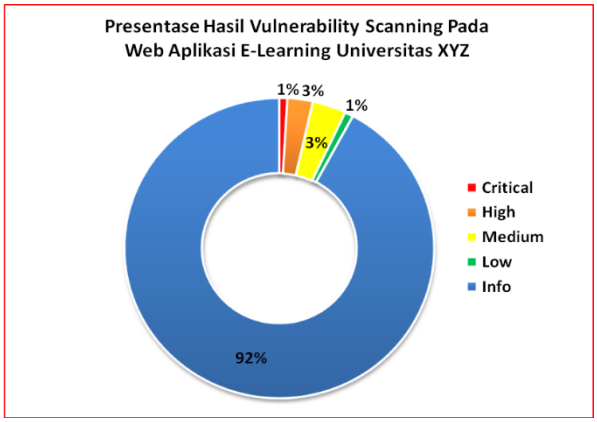
Nama Kerentanan	Dampak Kerentanan	Solusi
Elasticsearch Transport Protocol Unspecified Remote Code Execution	Penyerang dapat melakukan serangan injeksi *code* arbitrer melalui jarak jauh dikarenakan versi elasticsearch yang digunakan terlalu lawas.	Lakukan *Upgrade Elasticsearch* ke versi 1.7.0 atau ke versi terbaru.
Elasticsearch ESA-2015-06.	Kerentanan ini menjelaskan bahwa versi *Elasticsearch* sebelum 1.6.1 rentan terhadap serangan *remote* tanpa autentikasi dengan memanfaatkan *port* 9200.	Lakukan *Upgrade Elasticsearch* ke versi 1.7.0 atau versi terbaru.
CGI Generic SQL Injection (blind).	Penyerang dapat membaca data sensitive dari basis data melalui serangan *SQL Injection*	Lakukan perubahan pada *script CGI* yang terpengaruh sehingga *script* tersebut keluar dari argumen dengan benar
Elasticsearch Unrestricted Access Information Disclosure.	Sumber daya pada basis data tidak dibatasi menggunakan autentikasi, sehingga dapat di exploitasi penyerang untuk mengungkapkan informasi.	Aktifkan autentikasi pengguna pada elasticsearch.yml , dan pastikan *port elasticsearch* tidak public
SSL/TLS Protocol Initialization Vector Implementation Information Disclosure Vulnerability.	*SSLv3/TLSv1* dapat dipengaruhi oleh serangan *man-in-the-middle* untuk mendapatkan *header http.*	Konfigurasikan TLS 1.1 pada *web browser.*

Tabel 4. Laporan *Vulnerability Assesment* (Lanjutan)

Nama Kerentanan	Dampak Kerentanan	Solusi
Web Application Potentially Vulnerable to Clickjacking.	Penyerang dapat menipu pengguna untuk mengklik area halaman rentan, kerentanan ini terjadi pada halaman *web server* pada port 80.	Konfigurasikan *X-Frame-Options* atau *Content Security Policy* dengan *frame ancestors.*
Web Server Transmits Cleartext Credentials.	Seorang penyerang dapat menghasilkan *username* dan *password valid* dengan melakukan serangan *sniffing*. Kerentanan ini terjadi pada *port* 8080 halaman *web server.*	Gunakan *HTTPS* agar autentikasi *login* dapat terenkripsi.

D. Persentase *Vulnerability Scanning*

Persentase ini di dapat dari jumlah kerentanan yang ditemukan pada proses *vulnerability scanning*, presentase ini dibuat agar pihak Universitas XYZ dapat mengetahui tingkat kerentanan yang dimiliki *Web* Aplikasi *E-Learning* dengan mudah, sehingga dari persentase *vulnerability scanning* ini dapat dijadikan pihak Universitas XYZ untuk melakukan evaluasi terhadap keamanan *Web* Aplikasi *E-Learning*. Persentase kerentanan dapat dilihat pada Gambar 3.



Gambar 3. Persentase *Vulnerability Scanning*

Pada Gambar 4.9 menjelaskan bahwa persentase *vulnerability scanning* di dapatkan dari jumlah kerentanan yang sudah ditemukan terdapat kerentanan dengan tingkat kerentanan yang berbeda-beda. Namun hanya terdapat satu kerentanan yang bersifat *critical* (kritis) yaitu pada ***Elasticsearch***. Hal ini dibuktikan dengan dampak yang ditunjukan oleh *Elasticsearch* sangat tinggi berdasarkan *Confidentiality*, *Integrity*, *Availability*, berikut ini adalah perhitungannya.

1) Kerentanan pada *Elasticsearch Transport Protocol Unspecified Remote Code Execution* dinilai dari beberapa aspek yaitu *Attack Vector* (*AV*), *Attack Complexity* (*AC*), *Privilege Required* (*PR*), *User Interacation* (*UI*), *Confidentiality* (*C*), *Integrity* (*I*), dan *Availabiltiy* (*A*). Penilaian delapan aspek ini merujuk pada Bab 2 halaman 41 Pada Tabel 2.15.
2) Nilai dari delapan aspek pada kerentanan tersebut adalah (AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:H). Berikut ini adalah tabel aspek penilaian yang ditunjukan oleh Tabel 5.

Tabel 5. Aspek Penilaian

Metric	Metric Value	Numerical Value
Attack Vector	*Network*	0,85
Attack Complexity	*Low*	0,77
Privilege Required	*None*	0,85
User Interaction	*Noe*	0,85
Scope	*Unchanged*	6,42
Confidentiality	*High*	0,56
Integerity	*High*	0,56
Availability	*High*	0,56

3) Menghitung *Exploitability Subscore*
   Dengan rumus:
   $8{,}22 \times \text{AV} \times \text{AC} \times \text{PR} \times \text{UI}$

   Maka Hasilnya:
   $8{,}22 \times 0{,}85 \times 0{,}77 \times 0{,}85 \times 0{,}85$
   $= 3{,}887042775$

4) Menghitung *Impact Subscore* (*ISC*)
   Dengan rumus:
   $1\text{-}((1\text{-}impact_{conf}) \times (1\text{-}impact_{int}) \times (1\text{-}impact_{avail}))$

   Maka Hasilnya:
   $1 - (1 - 0{,}56) \times (1 - 0{,}56) \times (1 - 0{,}56)$
   $= 0{,}914816$

5) Menghitung *Impact Scope*:
   Dengan rumus:
   $6{,}42 \times ISC_{base}$

   Maka Hasilnya:
   $6{,}42 \times 0{,}914816 = 5{,}87311872$

6) Menghitung *Base Score*:
   Dengan rumus:
   $(Minimum[impact + Exploitability])$

   Maka Hasilnya:
   $Minimum(5{,}87311872 + 3{,}887042775)$
   $= 9{,}760161495$ (9.8)

Berdasarkan analisa dari perhitungan yang penulis lakukan bahwa ***overall risk level*** pada *Web* Aplikasi *E-Learning* adalah **"*High*"**, dikarenakan pada *threat level score* yang paling tinggi ditunjukan oleh kerentanan *critical* yaitu 9.8. Sehingga nilai tersebut yang menjadi alasan penulis untuk dijadikan sebagai ***overall risk.*** Jadi dapat disimpulkan *Web* Aplikasi *E-Learning* pada Universitas XYZ memiliki **tingkat kerentanan yang tinggi,** karena memiliki dampak serius yang mempengaruhi *Confidentiality*, *Integrity*, dan *Availability* pada *Web* Aplikasi *E-Learning* melalui kerentanan yang dimilikinya. maka pihak Universitas XYZ harus segera melakukan perbaikan dan evaluasi terhadap keamanan pada *Web* Aplikasi *E-Learning* agar resiko kerentanan pada *Web* Aplikasi *E-Learning* dapat dikurangi.

5. Kesimpulan dan Saran

A. Kesimpulan

Berdasarkan hasil penelitian mengenai "Vulnerability Assesment Untuk Mencari Celah Keamanan Web Aplikasi E-Learning Pada Universitas XYZ", maka dapat disimpulkan:

1) Dalam melakukan evaluasi keamanan terkait kerentanan *Web* Aplikasi *E-Learning* Pada Universitas XYZ dapat menggunakan *tool nessus*, hal ini dibuktikan dengan hasil *vulnerability scanning* menggunakan *tool nessus* yang dapat memberikan daftar kerentanan, penjelasan di setiap kerentanan, dampak dari kerentanan, serta rekomendasi untuk mengatasi kerentanan yang telah ditemukan.
2) Daftar kerentanan yang telah ditemukan oleh *tool nessus* pada penelitian ini dikategorikan berdasarkan tingkat kerentanannya yaitu *critical*, *high*, *medium*, *low*, dan *info*, serta disajikan dalam bentuk *Diagram pie* agar memudahkan pihak Universitas XYZ untuk melihat persentase kerentanan *Web* Aplikasi *E-Learning* yang dimilikinya.
3) Penggunaan *Vulnerability Assesment* dapat digunakan untuk mengetahui apa yang perlu diperbaiki dari sistem *Web* Aplikasi *E-Learning* Universitas XYZ agar sistemnya cukup tangguh dari potensi ancaman yang memanfaatkan celah kerentanan. Sehingga penggunaan *vulnerability assessment* dengan menggunakan *tool nessus* dapat digunakan untuk mengevaluasi kerentanan *Web* Aplikasi *E-Learning* Pada Universitas XYZ.
4) *Overall Risk Level Web* Aplikasi *E-Learning* Universitas XYZ berada pada *level High*,

sehingga diharuskan untuk melakukan perbaikan dan evaluasi terhadap keamanan *Web* Aplikasi *E-Learning*.

B. Saran

Berdasarkan dari kesimpulan hasil pembahasan yang telah diuraikan, maka saran yang dapat diberikan untuk pengembangan lebih lanjut dari penelitian *Vulnerability Assesment* Untuk Mencari Celah Keamanan *Web* Aplikasi *E-Learning* Pada Universitas XYZ adalah sebagai berikut:

1) Diharapkan dalam pengembangan penelitian selanjutnya dapat melakukan *Vulnerability assessment* dengan menerapkan *ISO* 27001.
2) Perlu dilakukan teknik *penetration testing* yang dapat mengeksploitasi celah kerentanan secara mendalam, sehingga akan memberikan rekomendasi yang lebih baik untuk melakukan perbaikan atau meningkatkan keamanan terhadap *web* yang ditunjuk sebagai target *vulnerability assessment*.
3) Dapat menggunakan *tool Nessus versi Profesional* untuk kebutuhan *Scanning* lebih dari 16 *Host* atau dapat menggunakan *tools vulnerability scanning* lainnya yang *free*.

Daftar Pustaka

CVSS. (2018). *Common Vulnerability Scoring System*. [online]. Available at: https://www.first.org/cvss/v3.0/specification-document (Diakses: 22 Mei 2019)

CWE. (2018). *Common Weakness Enumeration*. [online]. Available at: https://cwe.mitre.org/index.html (Diakses: 22 Mei 2019)

CVE. (2019). *Common Vulnerability and Exposure*. [online]. Available at: https://cve.mitre.org/ (Diakses: 22 Mei 2019)

Elastic Stack Security Disclosures. (2018). Report Issues | Elastic. [online]. Available at: https://www.elastic.co/community/security (Diakses: 29 Mei 2019)

GovCSIRT. (2012). *Methodology Vulnerability Assesment*. [online]. Available at: https://govcsirt.kominfo.go.id/254/ (Diakses: 20 Mei 2019)

Graves, K. (2010). *Certified Ethical Hacker*. Canada:Wiley.

ISACA. (2017). *Vulnerability Assesment*. ISACA:USA.

Id-SIRTII/CC. (2018). *PEMANTAUAN TRAFIC INTERNET*. [online]. Available at: https://www.idsirtii.or.id/trafik/bulanan/2018.html (Diakses: 1 April 2019).

Kumar, H. (2014), *Learning Nessus for Penetration Testing*. Birmingham:Packt Publishing Ltd.

Kizza, J. M. (2015). *Guide To Computer Network Security*. London:Springer.

Kumar, B. L. V. V., Kumar, D. K. R., & Santhi, V. (2016). *Penetration Testing using Linux Tools: Attacks and Defense Strategies*. International Journal of Engineering Research and Technology.

Maharani, M. Z., Andrian, H. R., & Juli, S. I. I. (2017). Analisis Keamanan Website Menggunakan Metode Scanning Dan Perhitungan Security Metriks. *E-Proceeding of Applied Science*.

Muliono Yohan., (2018). *Mengenal Istilah Common Vulnerability Scoring System.*[online]. Available at: http://socs.binus.ac.id/2018/12/13/mengenal-istilah-common-vulnerability-scoring-system/ (Diakses: 8 Juli 2019)

National Vulnerability Database. (2017). Available at: https://nvd.nist.gov/ (Diakses: 11 Mei 2019)

Nurdin, R. (2017). *Analisa Keamanan Internet Menggunakan Nessus dan Ethereal Universitas Putra Indonesia YPTK Padang*. Jurnal Teknologi Informasi dan Pendidikan.

Perez, A. (2014). *Network Security*. New Jersey:Wiley.

Rizko, K. N. F. (2018). *Security Assesment Menggunakan Tools Nessus Untuk Mencari Celah Keamanan Web Aplikasi Repositori Di Institusi Pendidikan XYZ*. Teknik Informatika, Fakultas Teknik, Universitas Muhammadiyah Yogyakarta.

Singh, A. (2012). Metasploit Penetration Testing Cookbook. Birmingham:Packt Publishing Ltd

Tania, A. M., Setiyadi, D., & Khasanah, F. N. (2018). *Keamanan website menggunakan vulnerability assessment*. Informatics For Educator And Professionals.

Yunanri, W., Riadi, I., & Yudhana, A. (2018). *Analasis Deteksi Vulnerability Pada Web Server Open Jurnal System Menggunakan OWASP Scanner*. Jurnal Teknologi Informasi.

Yudha, F. and T, A. M. P. M. (2018) Perancangan Aplikasi Pengujian Celah Keamanan Pada Aplikasi Berbasis Web, *CyberSecurity dan Forensik Digital*