PENGUJIAN SISTEM INFORMASI AKADEMIK UNIVERSITAS X MELALUI PENDEKATAN PENETRATION TESTING BERDASARKAN OWASP TOP 10

Fernanda Tinambunan, Achmad Junaidi, Agung Mustika Rizki
Informatika, UPN "Veteran" Jawa Timur
Jl. Rungkut Madya No.1, Gn. Anyar, Kec. Gn. Anyar, Surabaya, Jawa Timur
19081010179@student.upnjatim.ac.id

ABSTRAK

Universitas X memiliki beberapa situs web yang menjadi sarana penting untuk mengelola data krusial terkait proses perkuliahan. Namun seiring dengan kemajuan teknologi informasi yang semakin kompleks, sehingga ancaman dan tantangan keamanan siber juga meningkat. Oleh karena itu, diperlukan pengujian keamanan menggunakan pendekatan Penetration Testing karena teknik ini tidak hanya mengidentifikasi kerentanan permukaan, tetapi juga mencoba mengeksploitasi kerentanan tersebut secara mendalam, tahapannya pengumpulan data, analisa celah keamanan, pengujian, dan penulisan laporan, dengan fokus pada OWASP Top 10. Pengujian ini bertujuan mengidentifikasi dan mengevaluasi kerentanan keamanan pada Sistem Informasi Akademik Universitas X. Hasil pemindaian menunjukkan 23 celah keamanan, 20 di-antaranya sesuai dengan kategori yang diidentifikasi oleh OWASP TOP 10. Dari hasil ini, dapat disimpulkan bahwa penerapan OWASP TOP 10 sebagai acuan standar keamanan dalam melakukan uji penetrasi terbukti efektif dalam mengidentifikasi dan mengevaluasi celah keamanan yang signifikan pada sistem, serta memberikan arah yang tepat dalam meningkatkan tingkat keamanan pada website Sistem Informasi Akademik.

Kata kunci : *: Penetration Testing, OWASP Top 10, Sistem Informasi Akademik, Keamanan Sistem*

1. PENDAHULUAN

Sistem informasi telah menjadi krusial dalam operasi bisnis modern karena perannya dalam menyimpan, mengelola, dan mengakses data penting yang berpengaruh pada pengambilan keputusan, produktivitas, dan efisiensi perusahaan. Namun, meningkatnya kompleksitas teknologi informasi dan ancaman siber membuat keamanan sistem informasi menjadi aspek yang sangat penting. Pada Februari 2022, beberapa website milik Universitas X mengalami serangan peretasan yang menyebabkan deface atau perubahan tampilan visual situs web. Situs penting seperti Sistem Informasi Akademik, yang menyimpan informasi perkuliahan dan data akademik mahasiswa, memiliki celah keamanan yang berpotensi menyebabkan kerugian finansial, merusak reputasi, dan melanggar privasi data mahasiswa. Karena itu, penting dilakukan pengujian keamanan Sistem Informasi Akademik untuk mengevaluasi tingkat keamanannya.

Beberapa standar keamanan dapat digunakan sebagai dasar uji penetrasi, seperti ISO Standard, ISSAF, NIST CSF, dan OWASP. Dalam tesis Burkan & Tanase tentang Analisis dan Kerangka Keamanan Siber untuk Perusahaan Teknologi Informasi, terdapat analisis terhadap kerangka kerja OWASP, ISO 27000/27001, dan NIST. Dalam analisis tersebut, OWASP terlihat unggul karena bersifat open source dan dapat diakses tanpa biaya besar, menjadi pilihan terbaik terutama bagi perusahaan dengan keterbatasan ekonomi [1]. Keunggulan lainnya adalah daftar Top 10 OWASP yang selalu diperbarui secara rutin oleh pakar keamanan website global, mencakup tiga kategori baru, penamaan ulang empat kategori, dan konsolidasi baru pada tahun 2021 [2]. Dengan demikian, perusahaan yang mengadopsi OWASP dapat menghindari pedoman usang. OWASP Top 10 lebih fokus pada keamanan aplikasi web dan membantu perbaikan celah keamanan spesifik di aplikasi web, sementara NIST CSF memberikan panduan lebih luas untuk meningkatkan keamanan informasi secara menyeluruh dalam organisasi.

Studi kasus yang akan diteliti adalah website Sistem Informasi Akademik milik Universitas X. Sistem Informasi Akademik adalah sistem informasi web yang mengelola Kartu Rencana Studi (KRS), Kartu Hasil Studi (KHS), transkrip, dan data mata kuliah mahasiswa. Untuk menilai keamanan Sistem Informasi Akademik, dilakukan evaluasi dengan menerapkan teknik penetration testing sesuai dengan daftar OWASP TOP 10. Dari pengujian ini memperoleh pemahaman mengenai tingkat keamanan yang diterapkan pada situs web Sistem Informasi Akademik Universitas X serta melakukan analisis terhadap potensi celah keamanan yang mungkin ada dalam situs web tersebut.

2. TINJAUAN PUSTAKA

2.1. Penelitian Terdahulu

Tinjauan pustaka tujuan utamanya adalah mencari dan mengumpulkan data serta informasi terkait dengan penelitian, diantaranya mengenai penetration testing, vulnerability scanning, serta pengimplementasian mengenai standar keamanan OWASP TOP 10. Tinjauan pustaka ini merujuk kepada berbagai sumber seperti ebook, jurnal penelitian, website, buku, dan lain-lain. Semua sumber pustaka yang digunakan dalam penelitian ini akan dicantumkan secara terperinci dalam daftar pustaka.

Penelitian pertama yaitu penelitian dari jurnal yang dilakukan oleh Thurfah Afifa Rosaliah & Hananto yang menguji celah keamanan website menggunakan teknik penetration testing dan Metode OWASP TOP 10 dengan studi kasus Website SIM xxx. SIM merupakan sistem yang berguna untuk memonitor dan mengatur sistem lainnya, yang artinya sistem informasi ini akan banyak menyimpan data-data penting pengguna. Untuk itu informasi ini akan banyak menyimpan data-data penting pengguna. Untuk itu diperlukan implementasi keamanan yang kuat untuk mencegah pencurian data, penyalahgunaan data, atau pengambilalihan sistem oleh pihak yang tidak sah. Berdasarkan hasil analisis celah keamanan yang dilakukan dengan vulnarabilities scanning, ditemukan celah Broken Authentication, Sensitive Data Exposure, Security Misconfiguration, dan Clickjacking. Secara keseluruhan, dapat ditarik kesimpulan bahwa OWASP TOP 10 efektif sebagai pedoman standar keamanan untuk situs web, karena standard keamanan yang dimiliki OWASP lengkap dan detail dilihat dari konfigurasi halaman website maupun konfigurasi server. Banyak hasil penelitian ini yang merujuk pada daftar OWASP TOP 10. Maka dari itu metode OWASP TOP 10 menjadi metode pilihan penulis dalam melakukan uji penetrasi. [3]

Penelitian berikutnya yaitu penelitian dari jurnal yang dilakukan oleh Kusuma yang membahas Implementasi OWASP ZAP Untuk Pengujian Keamanan Sistem Informasi Akademik. Pengujian pada jurnal ini menguji Sistem Informasi Akademik Universitas Pancasila dengan metode penetration testing dan menggunakan tools OWASP ZAP versi 2.10.0. Zed Attack Proxy (ZAP) adalah aplikasi yang menyediakan scanner otomatis penetration testing untuk menemukan vulnerabilities dalam suatu web application. Berdasarkan hasil penetrasi OWASP ZAP website Sistem Informasi Akademik Universitas Pancasila memiliki 19 celah keamanan, maka kualitas website Sistem Informasi Akademik Universitas Pancasila berada ditingkat sedang. Diantaranya terdapat 4 kerentanan yang sesuai dengan standar OWASP TOP 10 2017, yaitu Broken Access Control, Security Misconfiguration, Vulnerable and Outdated Components, dan Software and Data Integrity Failures, sehingga penulis memilih tools OWASP ZAP untuk uji penetrasi karena dapat mendeteksi kerentanan yang ada pada standar OWASP TOP 10. [4]

Penelitian dari jurnal yang ditulis oleh Albahar dkk. yang membandingkan alat pengujian penetrasi untuk mendeteksi kerentanan web. Peneliti membandingkan 6 alat pengujian penetrasi yang paling banyak digunakan dalam penelitian atau jurnal akademik yaitu OWASP ZAP, Burp Suite Professional, Qualys WAS, Arachni, Wapiti3, Fortify WebInspect. Alat pengujian akan dibagi berdasarkan alat komersial (Qualys WAS, Fortify WebInspect, and Burp Suite Professional) dan non-komersial (OWASP ZAP, Arachni, and Wapiti3), lalu dibandingkan menggunakan standar dan metode sesuai referensi yang ada dan ditambah beberapa parameter yang mencakup: test coverage criteria, attack coverage criteria, vulnerability detection criteria, and efficiency criteria. Hasil yang didapat untuk commercial tools Burp Suite Professional adalah tools terbaik untuk mendeteksi OWASP Top 10 vulnerability, sedangkan untuk open-source tools yaitu OWASP ZAP. [5]

Penelitian selanjutnya yaitu penelitian dari jurnal yang dilakukan oleh Fachri dkk. yang Analisis Keamanan Webserver Menggunakan Penetration Test. Penelitian ini dilakukan pada web server dari Sistem Informasi Akademik pada perguruan tinggi. Metode yang digunakan dalam penelitian ini mencakup Information Gathering, Vulnerability Assessmant, Gaining Acces, Maintaining Acces, dan Clearing Track. Tujuan penelitian ini untuk mencari level kerentanan pada webserver dan memberi solusi untuk kerentanan tersebut. [6]

2.2. Penetration Testing

Penetration testing adalah proses mengeksploitasi sistem untuk menganalisis atau mengindentifikasi kelemahan pada konfigurasi sistem, kesalahan perangkat keras dan perangkat lunak. [7]

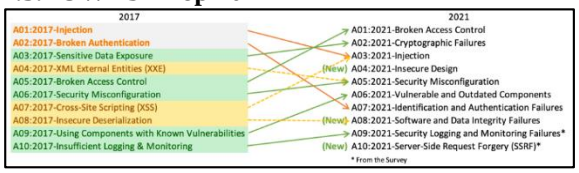
Tujuan utama dari pengujian penetrasi untuk menghindari insiden keamanan yang menyebabkan kerusakan dalam hal kerahasiaan, integritas, dan ketersediaan data. Pengujian penetrasi juga dapat membantu dalam pembentukan strategi keamanan informasi organisasi dengan mengidentifikasi kerentanan secara cepat dan akurat. [8]

Agar *penetration testing* dapat dilakukan dengan efektif dan hasilnya dapat didokumentasikan, diperlukan suatu pendekatan yang sistematis. Berikut tahapan berdasarkan Penetration Testing Execution Standard [9]:

1. Pre-engagement Interactions
   Tahap persiapan untuk pengujian, menentukan persetujuan dokumen dan alat yang diperlukan untuk pengujian
2. Intelligence gathering
   Tahap pengumpulan informasi tentang sistem target yang digunakan dalam pengujian.
3. Threat Modelling
   Pada tahap ini, akan diidentifikasi pendekatan pemodelan ancaman yang diperlukan untuk pengujian penetrasi.
4. Vulnerability Analysis
   Tahap ini mencari celah keamanan sistem yang dapat diserang oleh penyerang.
5. Exploitation
   Penguji mencoba menyerang keamanan sistem berdasarkan kerentanan yang telah diidentifikasi sebelumnya.
6. Post Exploitation
   Tahap ini menentukan tingkat keamanan sistem dan memberi rekomendasi perbaikan keamanan sistem.

7. Reporting
   Mendokumentasikan seluruh proses dalam bentuk yang mudah dimengerti oleh klien.

2.3. OWASP Top 10



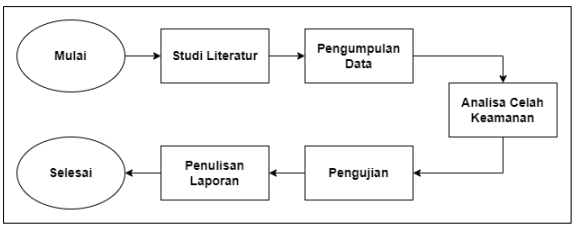
Gambar 1. Daftar OWASP Top 10:2021

OWASP Top 10 merupakan sebuah daftar kerentanan keamanan yang paling banyak mengancam keamanan suatu website, yang dirilis oleh komunitas OWASP. Daftar ini terus berkembang dan menyesuaikan dengan perkembangan teknologi website/aplikasi web. OWASP Top 10 ini diupdate secara rutin oleh tim yang terdiri dari para ahli keamanan situs web di berbagai belahan dunia. Terdapat tiga kategori baru, empat kategori yang mengalami perubahan nama, perubahan ruang lingkup, dan beberapa konsolidasi baru di Top 10 untuk 2021. [2]

1. A01:2021-Broken Access Control
2. A02:2021-Cryptographic Failures
3. A03:2021-Injection
4. A04:2021-Insecure Design
5. A05:2021-Security Misconfiguration
6. A06:2021-Vulnerable and Outdated Components
7. A07:2021-Identification and Authentication Failures
8. A08:2021-Software and Data Integrity Failures
9. A09:2021-Security Logging and Monitoring Failures
10. A10:2021-Server-Side Request Forgery

3. METODE PENELITIAN

Dalam menjelaskan masalah dalam penelitian perlu disusun suatu kerangka pemikiran atau alur penelitian untuk mempermudah pemahaman dalam penelitian tersebut. Berikut merupakan alur penilitian yang digunakan antara lain, mulai, studi pustaka, pengumpulan data, analisa celah keamanan, pengujian, penulisan laporan, dan selesai.



Gambar 2. Flowchart alur penelitian

3.1. Pengumpulan Data

Tujuan dari tahap ini untuk menemukan informasi sebanyak mungkin mengenai target baik itu individual ataupun organisasi. Tahap ini juga menentukan tujuan dan ruang lingkup pengujian, serta sistem yang akan ditangani dan metode pengujian yang digunakan [10]. Pengumpulan data dilakukan melalui dua tahap, yaitu information gathering dan footprinting Ini adalah langkah penting dalam banyak konteks, termasuk bisnis, keamanan cyber, penelitian, investigasi, dan banyak lagi. Informasi yang bisa dikumpulkan berupa IP Address, topology network, network resources, spesifikasi server dan informasi personal user seperti alamat, email, nomor telepon. Landasan untuk melakukan eksploitasi menggunakan berbagai tools sepert Netcraft, Whois, ICANN, dan Zenmap.

1. Netcraft
   Halaman site report url target akan diisi pada kolom yang tersedia. Hasil pemindaian akan menunjukkan informasi jaringan seperti IPv4, IPv6, pendaftaran domain, server nama, kontak admin DNS, perusahaan hosting, dan juga catatan riwayat hosting
2. ICANN
   IP address yang telah ditemukan akan diisi pada kolom data lookup. Hasil pemindaian akan menunjukkan informasi jaringan IP (rentang ip address, versi ip address, dan kode negara) dan informasi pengelola web seperti nama, email, nomor telepon, dan alamat lokasi pengelola.
3. Zenmap – Nmap
   '`nmap -T4 -F [ip_address]`' pada Nmap digunakan untuk melakukan quick scan pada suatu alamat IP dengan menggunakan tingkat kecepatan pemindaian yang agresif dan membatasi pemindaian hanya pada sejumlah port yang umum digunakan.
   '`nmap -T4 -A -v [ip_address]`' menjalankan sebuah intense scan pada suatu alamat IP dengan menggunakan Nmap. Perintah ini secara keseluruhan akan melakukan intense scan pada alamat IP yang ditentukan dengan tingkat kecepatan pemindaian yang agresif, mengaktifkan deteksi otomatis untuk mendapatkan informasi tentang sistem operasi, versi perangkat lunak, dan skrip keamanan, serta menampilkan output dengan detail tambahan untuk analisis lebih lanjut. [11]

3.2. Memindai dan Analisa Celah Keamanan

Analisis kerentanan akan diterapkan menggunakan teknik vulnerability scanning. Vulnerability scanning adalah tindakan untuk mendapatkan informasi mengenai kerentanan dalam jaringan dengan memanfaatkan berbagai alat pemindaian jaringan dan pemindai kerentanan. Hal ini mencakup identifikasi aspek-aspek seperti port yang terbuka, bug aplikasi, dan pemahaman terhadap potensi serangan yang dapat menimbulkan dampak yang signifikan jika terjadi pada kerentanan situs web yang ada [12]. Vulnerability scanning bertujuan untuk menemukan tingkat kerentanan dan keamanan dalam Sistem Informasi Akademik dengan menggunakan alat khusus yang dirancang untuk mendeteksi kelemahan

keamanan. Tools yang digunakan pada penelitian ini yaitu OWASP ZAP, dengan fitur automated scan akan memindai target IP address 103.xxx.xxx.xxx, 144.xxx.xxx.xxx, dan domain x.xx.ac.id. Hasil pemindaian celah keamanan berupa laporan jenis celah keamanan, level risiko (high, medium, dan low), level keamanan (high, medium, low, dan informative), dan Analisa OWASP Top 10.

3.3. Pengujian

Eksploitasi adalah proses untuk mendapatkan akses dengan memanfaatkan kerentanan yang diterima sebelumnya melalui tahap analisis. Pengujian (penetration testing) berdasarkan celah keamanan pada website atau eksploitasi menggunakan tools yang bertujuan untuk mengetahui apakah website Sistem Informasi Manajemen (SIM) rentan terhadap serangan yang akan diujikan tersebut [8]. Pengujian akan menyerang jenis celah dengan level risiko high dan medium yang telah ditemukan pada tahap sebelumnya menggunakan alat Burp Suite. Dengan fitur repeater untuk memodifikasi permintaan dan memeriksa respon, fitur intruder untuk menguji berbagai wordlist dan payloads, variasi parameter, dan pengaturan serangan. Jika pengujian yang dilakukan berhasil menembus celah keamanan, maka keamanan yang telah diterapkan membutuhkan rekomendasi perbaikan.

3.4. Penulisan Laporan

Membuat laporan yang berisi penjelasan hasil dari tahap pengumpulan data, vulnerability scanning, dan report uji penetrasi yang sudah dilakukan disertai dengan rekomendasi perbaikan.

4. HASIL DAN PEMBAHASAN

Pengujian Keamanan pada Sistem Informasi Akademik Universitas X bertujuan untuk mencegah peretasan yang dilakukan oleh orang yang tidak bertanggung jawab. Hasil dari pengujian akan diberikan kepada pengembang sistem sebagai panduan untuk melakukan perbaikan yang diperlukan. Pengujian ini dilaksanakan dengan menggunakan teknik penetration testing berdasarkan standar keamanan OWASP TOP 10. Pada bagian ini akan dipaparkan hasil pengumpulan data, vulnerability scanning, dan pengujian.

4.1. Pengumpulan Data

Dari proses Information Gathering yang telah dilakukan pada Sistem Informasi Akademik, hasilnya terungkap dalam bentuk informasi yang terdokumentasi pada tabel 1.

Tabel 1. Hasil information gathering

No.	Proses	Hasil Akhir
1.	Netcraft	IP Address **103.xxx.xxx.xxx** dan **114.xxx.xxx.xxx**
2.	Whois	**103.xxx.xxx.xxx** Data tidak ditemukan
		114. xxx.xxx.xxx Nama pemilik domain : PT. xxx Alamat pemilik domain : Jl. xxx Nomor Telepon pemilik domain : +1.62xxxxxxxxxx Email pemililk domain : xxx@xxx.com
3.	ICANN	**103.xxx.xxx.xxx** Name teknisi: Axxxx Email teknisi: xxx@yahoo.com No Telepon: +62-xx-xxxxxxx Kind: individual Alamat domain: Jl. xxx, Jawa Timur
		114.xxx.xxx.xxx Name: xxx Hostmaster Email: xxx@xxx.co.id Phone: +62-xx-xxxxxxxx 62-xxxxxxxxxx Fax: +62-xx-xxxxxxxx Kind: individual Mailing Address: PT xxx, Jl. xxx, Jakarta Pusat
4.	Zenmap-NMap	**103.xxx.xxx.xxx open ports :** Port 80 (HTTP) Port 443 (Remote Desktop Protocol) Port 4444 (Transfer Control Protocol) Port 8443 (HTTPS Client Authentication connection protocol) **103.xxx.xxx.xxx Filtered Ports :** 996 Ports **103.xxx.xxx.xxx OS** : Tomato 1.27 – 1.28 (Linux 2.4.20)
		114.xxx.xxx.xxx open ports : Port 80 (HTTP) Port 443 (Remote Desktop Protocol)

No.	Proses	Hasil Akhir
		Port 4444 (Transfer Control Protocol) Port 3389 Port 8443 (HTTPS Client Authentication connection protocol) **114.xxx.xxx.xxx Filtered Ports :** 997 Ports **114.xxx.xxx.xxx OS** : Tomato 1.27 – 1.28 (Linux 2.4.20) **114.xxx.xxx.xxx Hostname :** 114-xxx-xxx-xxx.resources.xxx.com - PTR

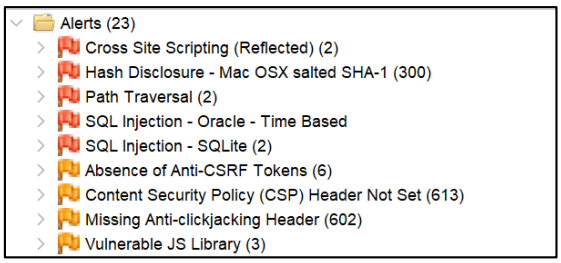
4.2. Vulnerability Scanning

Scanning vulnerabillity dilakukan untuk mengidentifikasi potensi kerentanan atau kelemahan dalam situs web dengan menggunakan alat OWASP ZAP yang secara spesifik dibuat untuk menemukan celah keamanan berdasarkan tingkat risiko dari setiap temuan kerentanan. Untuk menentukan tingkat risiko berdasarkan OWASP Risk Rating Methodology dengan model risiko standar. [13]

$$Resiko = Kemungkinan\ x\ Dampak \quad (1)$$

Faktor-faktor yang membentuk "kemungkinan" dan "dampak":

1. Mengidentifikasi Risiko
2. Faktor untuk Memperkirakan Kemungkinan
3. Faktor-Faktor untuk Memperkirakan Dampak
4. Menentukan Tingkat Keparahan Risiko
5. Memutuskan Apa yang Harus Diperbaiki
6. Menyesuaikan Model Penilaian Risiko Anda



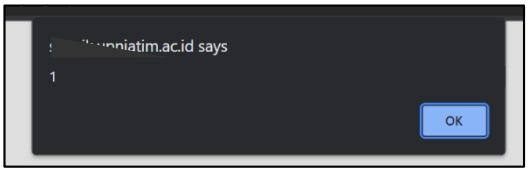
Gambar 3. Celah yang ditemukan pada Sistem Informasi Akademik

Berdasarkan gambar 3. hasil scanning vulnerability terdapat 23 celah keamanan yang ditemukan pada OWASP ZAP, 5 celah berisiko tinggi, 4 celah berisiko sedang, 9 celah berisiko rendah, dan 5 celah information. Dari 23 celah keamanan ditemukan 20 celah yang masuk kedalam kategori OWASP TOP 10 yaitu Broken Access Control, Injection, Insecure Design, Security Misconfiguration, Vulnerable and Outdated Components, dan Software and Data Integrity Failures.

4.3. Pengujian

Setelah mendapat hasil dari vulnerability testing yaitu beberapa vulnerability, akan dilanjutkan dengan tahap exploitation. Tahap ini akan menampilkan beberapa simulasi exploitation.

1. Cross Site Scripting (Reflected)
   Pada hasil scanning menggunakan OWASP ZAP, website Sistem Informasi Akademik terdeteksi terdapat vulnerability pada serangan cross site scripting (reflected). Penyerangan ini akan dilakukan menggunakan aplikasi Burpsuite. hasil yang didapat dari penyerangan ini terdapat respon code 200 yaitu penyerangan berhasil.



Gambar 4. Respon cross site scripting (reflected)

2. Hash Disclosure - Mac OSX salted SHA-1
   Pengujian dilakukan menggunakan website https://md5decrypt.net/en/Sha1/, untuk mengubah data atau informasi dari teks yang tidak dapat dibaca menjadi teks yang dapat dibaca oleh manusia atau disebut dekripsi. Hasil dekripsi yaitu hash yang dimasukkan tidak valid, dapat dikatakan celah hash disclosure pada website Sistem Informasi Akademik merupakan celah berstatus false positives.

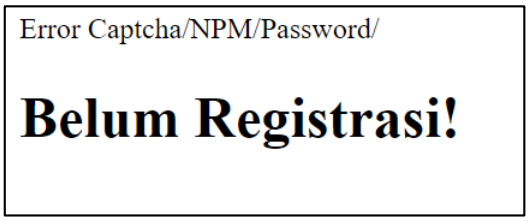
Gambar 5. Hasil dekripsi hash

3. SQL Injection - Oracle - Time Based
   Dicoba penyerangan dengan fitur intruder milik burpsuite menggunakan payloads yang telah diatur untuk penyerangan SQL Injection – Time Based. Hasil yang didapat dapat dilihat pada gambar 6. terdapat respon code 200, 302, dan 500. Waktu yang dibutuhkan untuk merespon request tidak sesuai dengan input payload, berdasarkan respon tersebut sehingga dapat dikatakan bahwa SQL Injection – Oracle – Time Based tidak dapat dilakukan pada website Sistem Informasi Akademik.

Request	Payload	Status code	Response received
0		200	39
12	')) or sleep(5)='	200	39
13	;waitfor delay '0:0:5'--	200	40
20	'));waitfor delay '0:0:5'--	200	40
31	')) or benchmark(10000000,...	200	40
33	1 or pg_sleep(5)--	302	40
19	));waitfor delay '0:0:5'--	200	41
23	1 or benchmark(10000000,...	500	41
24	" or benchmark(10000000,...	500	42
28	') or benchmark(10000000,...	200	42
11	")) or sleep(5)="	500	43
16	"waitfor delay '0:0:5'--	200	43

Gambar 6. Hasil payloads SQL injection – time based

4. SQL Injection – SQLite
   Dicoba penyerangan dengan fitur intruder milik burpsuite menggunakan payloads yang telah diatur untuk penyerangan SQL Injection – SQLite. Berdasarkan dari hasil penyerangan payload yang diinputkan, terdapat respon code 200, namun waktu respon dari website Sistem Informasi Akademik tidak sesuai dan respon code 302, dapat dikatakan bahwa SQL Injection - SQLite tidak dapat dilakukan pada website Sistem Informasi Akademik.

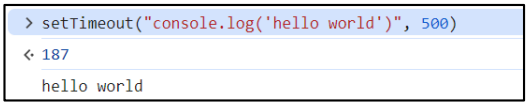


Gambar 7. Respon Payloads SQL Injection – SQLite

5. Path Traversal
   Disimulasikan penyerangan menggunakan Teknik Brute Force pada intruder burpsuite dengan payloads yang telah diatur untuk penyerangan Path Traversal. Hasil penyerangan terdapat respon code 200 namun tidak mendapatkan respon apapun dari website Sistem Informasi Akademik dan respon code 302, dapat dikatakan penyerangan Path Traversal tidak dapat dilakukan pada website Sistem Informasi Akademik.

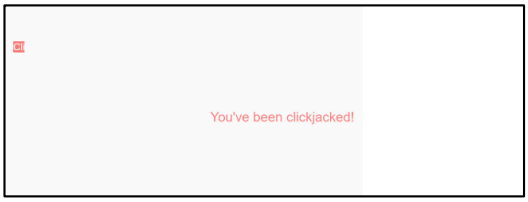
Gambar 8. Hasil Payloads Path Traversal

6. Content Security Policy (CSP) Header Not Set
   Pengujian celah keamanan ini dilakukan secara manual menggunakan inspect network dan console pada browser. Terlihat response headers pada 'https://x.xx.ac.id' tidak terdapat informasi Content-Security-Policy. Selanjutnya, pada console mencoba menginjeksi script 'setTimeout("console.log('hello world')", 500)', dan hasilnya terlihat pada gambar 9. script berhasil dijalankan. Ini artinya Sistem Informasi Akademik terbukti belum mengatur Content-Security-Policy header.



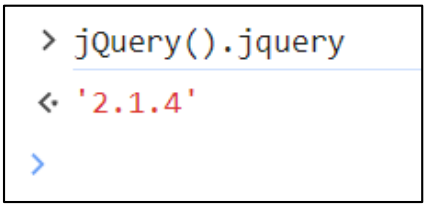
Gambar 9. Injeksi script pada console Sistem Informasi Akademik

7. Missing Anti-clickjacking Header
   Pengujian akan dilakukan menggunakan tools Burp Clickbandit pada BurpSuite. Clickbandit memungkinkan untuk membuat serangan clickjacking. Hasil pengujian yaitu terdapat tanda dimana pengguna menekan button dan akan tampil seperti gambar 10 'You've been clickjacked' Sistem Informasi Akademik berhasil diserang.



Gambar 10. Berhasil Menyerang Dengan Clickjacking

8. Vulnerable JS Library
   Pengujian celah keamanan ini dilakukan secara manual menggunakan console pada browser dengan menulis perintah 'jQuery().jquery' pada console. Sistem Informasi Akademik menggunakan jQuery versi 2.1.4. Ini artinya terbukti belum memperbarui komponen-komponen JavaScript (JS) library yang pada Sistem Informasi Akademik.



Gambar 11. JQuery yang digunakan Sistem Informasi Akademik

9. Absence of Anti-CSRF Tokens
   Dicari kata kunci 'csrf' pada respon pada halaman tersebut dan hasilnya tidak menemukan token Anti-CSRF yang umum digunakan seperti: `anticsrf`, `CSRFToken`, `__RequestVerificationToken`, `csrfmiddlewaretoken`, `authentic_token`, `OWASP_CSRFTOKEN`, `anoncsrf`, `csrf_token`, `_csrf`,

`_csrfSecret`, `__csrf_magic`, `CSRF`, `_token`, `_csrf_token didalam HTML`, ini artinya Sistem Informasi Akademik terbukti belum mengatur Anti-CSRF.

Gambar 12. Mencari Token CSRF

4.4. Rekomendasi Perbaikan

Perbaikan keamanan sistem melibatkan serangkaian langkah-langkah yang dirancang untuk meningkatkan tingkat keamanan suatu sistem. Jika pengujian yang dilakukan berhasil menembus celah keamanan, maka keamanan yang telah diterapkan membutuhkan rekomendasi perbaikan.

1. Pemantauan Komponen
   Dilakukan audit reguler terhadap komponen-komponen yang digunakan dalam sistem. Buat inventarisasi yang jelas tentang semua framework, library, modul, dan dependensi yang ada.
2. Kebijakan dan Pemantauan Pembaruan
   Menetapkan kebijakan yang jelas untuk memperbarui komponen secara teratur. Tentukan frekuensi pembaruan dan prioritas berdasarkan kerentanan yang ditemukan. Pastikan tim pengembangan secara konsisten memperbarui komponen ke versi terbaru yang aman.
3. Implementasi whitelisting untuk input
   Algoritma whitelisting adalah proses memverifikasi dan mengizinkan hanya input yang telah ditentukan sebelumnya, sementara memblokir atau menolak input yang tidak sesuai dengan daftar yang telah ditetapkan.
4. Implementasi X-Frame-Options
   X-Frame Options adalah sebuah header dari HTTP yang disebut juga sebagai header keamanan HTTP. Header ini akan memberi perintah kepada web browser ketika menangani konten di dalamnya. Alasan utama dari langkah awal ini adalah untuk melindungi dari serangan clickjacking, dengan mencegah rendering bingkai pada halaman, termasuk merender pada <frame>, <iframe>, atau <object>. Iframe digunakan untuk menyematkan dan mengisolasi konten pihak ketiga ke dalam website. Contoh penggunaan iframe termasuk tombol berbagi ke sosial media, Google Map, pemutar audio, pemutar video, dan juga iklan pihak ketiga.
   Header X-Frame-Options memiliki tiga perintah yang bisa dipilih yaitu DENY yang melarang browser untuk memuat situs dalam frame apa pun, SAMEORIGIN artinya browser hanya akan memuat situs dalam frame jika asal domainnya sama dengan situs itu sendiri, dan ALLOW-FROM uri yang hanya memuat situs dalam frame jika asal domain dari situs yang memuat iframe cocok dengan URI yang diberikan.
5. JavaScript: Automatically Including CSRF Tokens as an AJAX Request Header
   JavaScript dapat digunakan untuk secara otomatis menyertakan token CSRF (Cross-Site Request Forgery) sebagai header dalam permintaan AJAX. CSRF token biasanya digunakan untuk mengamankan aplikasi web dengan memastikan bahwa setiap permintaan yang dikirim berasal dari pengguna yang sah. Library JQuery memungkinkan untuk mengganti pengaturan default agar header ditambahkan secara otomatis ke semua permintaan AJAX. JQuery mengekspos API yang disebut $.ajaxSetup() yang dapat digunakan untuk menambahkan anti-csrf-tokenheader ke permintaan AJAX. Dokumentasi API untuk $.ajaxSetup() dapat ditemukan di sini. Fungsi csrfSafeMethod() yang ditentukan di bawah ini akan memfilter metode HTTP yang aman dan hanya menambahkan header ke metode HTTP yang tidak aman.
6. Mengimplementasikan Content Security Policy
   Mengimplementasikan Content Security Policy untuk mencegah serangan XSS dan SQL Injection. Dapat dilakukan dengan mengonfigurasi CSP dengan <meta> tag HTML atau PHP

5. KESIMPULAN DAN SARAN

Pengujian celah keamanan website Sistem Informasi Akademik Universitas X menggunakan teknik Penetration Testing berdasarkan OWASP TOP 10, merujuk pada proses uji penetrasi yang dilakukan terhadap Sistem Informasi Akademik dengan tujuan untuk mengidentifikasi dan mengevaluasi celah keamanan yang mungkin ada. Hasil pemindaian terhadap website Sistem Informasi Akademik Universitas X mengungkapkan adanya 23 celah keamanan yang teridentifikasi, dimana sebanyak 20 celah tersebut masuk ke dalam kategori yang telah diidentifikasi oleh OWASP TOP 10 yaitu Broken Access Control, Injection, Insecure Design, Security Misconfiguration, Vulnerable and Outdated Components, dan Software and Data Integrity Failures. Teridentifikasi 5 celah dengan risiko tinggi, 4 celah dengan risiko sedang, 9 celah dengan risiko rendah, dan 5 celah tergolong dalam kategori informasi. Pengujian dilakukan secara spesifik terhadap celah keamanan dengan tingkat risiko tinggi yaitu Cross Site Scripting Reflected, Hash Disclosure, SQL Injection - Oracle - Time Based, SQL Injection – SQLite, dan Path Traversal. Pengujian juga dilakukan terhadap celah dengan tingkat risiko sedang, seperti Content Security Policy Header Not Set, Missing Anti-clickjacking Header, Vulnerable JS Library, dan Absence of Anti-CSRF Tokens. Temuan ini memberikan gambaran yang signifikan bahwa sebagian besar celah keamanan yang terdeteksi sejalan dengan kategori kerentanan utama yang telah didefinisikan oleh standar OWASP TOP 10. Dari hasil

ini, dapat disimpulkan bahwa penerapan OWASP TOP 10 sebagai acuan standar keamanan dalam melakukan uji penetrasi terbukti efektif dalam mengidentifikasi dan mengevaluasi celah keamanan yang signifikan pada sistem. Dalam konteks pengembangan lebih lanjut, disarankan agar penelitian selanjutnya fokus pada pengujian celah keamanan yang lebih mendalam terhadap website Sistem Informasi Akademik. Hal ini penting untuk mengidentifikasi dan mengeksplorasi kemungkinan kelemahan yang mungkin tidak terdeteksi. Adapun saran yang sangat diinginkan untuk penelitian berikutnya adalah menggunakan OWASP Testing Framework, penggunaan kerangka kerja ini diharapkan dapat memberikan struktur yang lebih terorganisir dan komprehensif dalam menemukan berbagai celah keamanan yang lebih rinci dan terperinci pada sistem yang diuji

DAFTAR PUSTAKA

[1] E. Burkan and A. Tanase, "The Perceived Value of Cybersecurity Analyses and Frameworks for an IT Company," University of Agder, 2021.

[2] OWASP, "OWASP Top 10:2021," owasp. Accessed: Nov. 28, 2023. [Online]. Available: https://owasp.org/Top10/id/

[3] Y. Thurfah Afifa Rosaliah and B. Hananto, *Pengujian Celah Keamanan Website Menggunakan Teknik Penetration Testing dan Metode OWASP TOP 10 pada Website SIM xxx*. 2021.

[4] G. H. A. K. Kusuma, "IMPLEMENTASI OWASP ZAP UNTUK PENGUJIAN KEAMANAN SISTEM INFORMASI AKADEMIK," *Jurnal Teknologi Informasi*, vol. 16, no. 2, pp. 178–186, Aug. 2022, doi: https://doi.org/10.47111/JTI.

[5] M. Albahar, D. Alansari, and A. Jurcut, "An Empirical Comparison of Pen-Testing Tools for Detecting Web App Vulnerabilities," *Electronics (Switzerland)*, vol. 11, no. 19, Oct. 2022, doi: 10.3390/electronics11192991.

[6] F. Fachri, A. Fadlil, I. Riadi, A. Dahlan, Y. Jln Soepomo, and I. Artikel, "Analisis Keamanan Webserver Menggunakan Penetration Test," *JURNAL INFORMATIKA*, vol. 8, no. 2, 2021, [Online]. Available: http://ejournal.bsi.ac.id/ejurnal/index.php/ji

[7] J. N. Goel and B. M. Mehtre, "Vulnerability Assessment & Penetration Testing as a Cyber Defence Technology," in *Procedia Computer Science*, Elsevier, 2015, pp. 710–715. doi: 10.1016/j.procs.2015.07.458.

[8] H. M. Z. Al Shebli and B. D. Beheshti, "A Study on Penetration Testing Process and Tools," 2018.

[9] Geeksforgeeks, "Penetration Testing Execution Standard (PTES)." Accessed: Nov. 28, 2023. [Online]. Available: https://www.geeksforgeeks.org/penetration-testing-execution-standard-ptes/

[10] kamarkamsib, "Penetration Testing," Github. Accessed: Nov. 28, 2023. [Online]. Available: https://github.com/kamarkamsib/penetration-testing

[11] A. R. Patel, "Cyber Security Nmap Cheat Sheet," Medium. Accessed: Nov. 28, 2023. [Online]. Available: https://akashranjanpatel.medium.com/cyber-security-nmap-cheat-sheet-b8b557794668

[12] E. Irawadi Alwi and F. Umar, "Analisis Keamanan Website Menggunakan Teknik Footprinting dan Vulnerability Scanning," 2020.

[13] OWASP Team, "OWASP Risk Rating Methodology", Accessed: Jan. 19, 2024. [Online]. Available: https://owasp.org/www-community/OWASP_Risk_Rating_Methodology