Jurnal Sains Komputer & Informatika (J-SAKTI)
Volume 7 Nomor 2, September 2023, pp. 765-777
ISSN: 2548-9771/EISSN: 2549-7200
https://tunasbangsa.ac.id/ejurnal/index.php/jsakti

Optimisasi Strategi Security Mitigation Dengan Vapt Pada Website Absensi Praktikan Dan Asisten Laboratorium Praktek

Aulia Basyirah[1], Umar Yunan Kurnia Septo Hediyanto[2], Muhammad Fathinuddin[3]
[1,2,3]Universitas Telkom, Indonesia
e-mail: auliab@student.telkomuniversity.ac.id[1], umaryunan@telkomuniversity.ac.id[2], muhammadfathinuddin@telkomuniversity.ac.id[3]

Abstract

Information technology is growing rapidly alongside its users. One of the uses of information technology is websites, which have been widely adopted by various parties, including XYZ University, utilizing them for academic and internal purposes. One such website at the university is used for attendance tracking during practical sessions in the Faculty of XYZ. However, technological advancements have also brought an increase in security attacks on websites by unauthorized entities. Therefore, a vulnerability assessment was conducted using the Vulnerability Assessment and Penetration Testing (VAPT) method, employing automated scanning tools such as Nessus, Burpsuite, and OWASP ZAP to identify vulnerabilities in the website. During the testing, 27 security vulnerabilities were found and consolidated into 9 issues for exploitation and mitigation. Eventually, 4 out of the 9 security vulnerabilities were successfully mitigated.

Keywords*: Vulnerability, VAPT, automated scanning, exploitation, mitigation.*

Abstrak

Teknologi informasi tumbuh pesat bersama penggunanya. Salah satu pemanfaatan teknologi informasi adalah website telah banyak digunakan oleh berbagai pihak, termasuk Universitas XYZ yang memanfaatkannya untuk keperluan akademik dan entitas internal. Salah satu website di universitas tersebut digunakan untuk absensi praktikum di Fakultas XYZ. Namun, perkembangan teknologi juga membawa peningkatan serangan keamanan terhadap website oleh pihak yang tidak bertanggung jawab. Oleh karena itu, dilakukan vulnerability assessment menggunakan metode VAPT menggunakan tools automated scanning, yaitu Nessus, Burpsuite, dan OWASP ZAP untuk menemukan kerentanan pada website. Dalam pengujian tersebut, ditemukan 27 celah keamanan yang dikonsolidasikan menjadi 9 celah untuk dilakukan eksploitasi dan mitigasi. Pada akhirnya, 4 dari 9 celah keamanan berhasil dimitigasi.

Kata kunci*: Kerentanan, VAPT, automated scanning, eksploitasi, mitigasi.*

1. PENDAHULUAN

Perkembangan teknologi informasi yang pesat seiring dengan pertumbuhan penggunanya telah memudahkan akses dan penyebaran informasi. Oleh karena itu, keamanan informasi menjadi penting karena informasi merupakan aset berharga bagi berbagai pihak. Dalam konteks ini, keamanan informasi melibatkan teknologi komputer dan jaringan karena banyak sumber informasi yang berasal dari internet [1]. *Website* merupakan salah satu media yang digunakan oleh banyak pihak untuk menyebarkan informasi. Selain tingginya manfaat yang dirasakan, tingkat risiko dan ancaman penyalaahgunaan pada teknologi informasi seperti *website* juga semakin tinggi dan kompleks sehingga lebih rentan terhadap ancaman atau serangan jaringan [2]. Tingkat kerentanan dan serangan pada setiap *website* tentu

Jurnal Sains Komputer & Informatika (J-SAKTI)
Volume 7 Nomor 2, September 2023, pp. 765-777
ISSN: 2548-9771/EISSN: 2549-7200
https://tunasbangsa.ac.id/ejurnal/index.php/jsakti

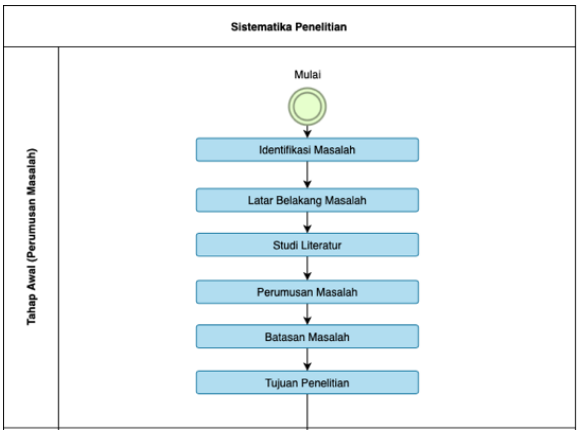
berbeda-beda, sehingga diperlukan pengujian celah keamanan dan penilaian risiko pada *website* terkait dengan mempertimbangkan faktor faktor yang ada [3]. Pedoman untuk penilai risiko pada suatu *website* yang dapat dikatakan aman dari serangan *cyber* mengacu pada tiga aspek, yaitu *confidentially*, *integrity*, dan *availability* atau CIA TRIAD [3].

Untuk mencegah serangan *cyber*, perlu dilakukan *security testing* pada *website*. *Security testing* harus mengikuti standarisasi yang telah ditetapkan. Salah satu standarisasi yang sering digunakan, yaitu metode *Vulnerability Assessment and Penetration Testing* (VAPT). *Vulnerability assessment* adalah proses melakukan pemindaian sistem, perangkat lunak atau jaringan untuk mengetahui kelemahan dan celah yang ada di dalamnya. Sementara, *penetration testing* adalah proses yang mencoba untuk mengeksploitasi sistem untuk mengetahui kemungkinan eksploitasi dari celah keamanan yang ditemukan [4]. Dalam melakukan metode VAPT diperlukan penggunaan *software* keamanan dengan beragam pilihan *software* yang dapat digunakan sesuai dengan kebutuhan pengujiannya.

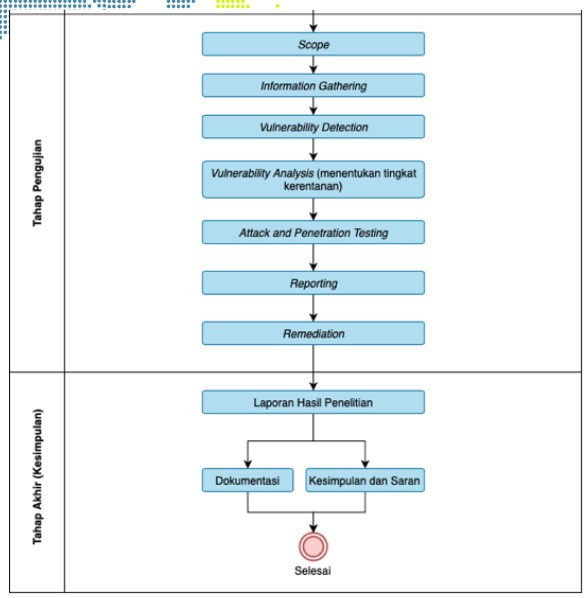
Berdasarkan informasi di atas, diperlukan solusi untuk mencegah serangan *cyber* pada *website* absensi praktikan dan asisten laboratorium praktek Universitas XYZ. *Website* ini berfungsi untuk memvalidasi data kehadiran mahasiswa pada kegiatan praktikum yang mempengaruhi penilaian akhir semester. Untuk melindungi validasi data, perlu dilakukan implementasi *vulnerability assessment* dengan menggunakan tiga *tools automated scan*, yaitu OWASP ZAP, Bupsuite, dan Nessus. Hasil dari *vulnerability assessment* dan *penetration testing* diharapkan dapat meminimalisir serangan *cyber* yang merugikan pengguna *website*.

2. METODOLOGI PENELITIAN

Sistematika penyelesaian masalah pada penelitian ini digambarkan dalam bentuk bagan yang berisi beberapa tahapan yang dilakukan dalam penelitian. Tahapan yang ada pada sistematika penyelesaian masalah dijabarkan secara sistematis dan terstuktur yang terbagi menjadi lima tahapan yaitu tahap identifikasi masalah, tahap perumusan masalah, tahap pengumpulan data, tahap analisis dan penelitian, dan tahap akhir seperti pada Gambar 1 berikut.



Jurnal Sains Komputer & Informatika (J-SAKTI)
Volume 7 Nomor 2, September 2023, pp. 765-777
ISSN: 2548-9771/EISSN: 2549-7200
https://tunasbangsa.ac.id/ejurnal/index.php/jsakti



Gambar 1. Sistematika Penelitian

1) Tahap Awal

Tahap Awal dalam penelitian ini yaitu melakukan identifikasi masalah yang berisikan kegiatan observasi dan pencarian fokus masalah yang akan diteliti terhadap objek penelitian. Setelah didapatkan permasalahan yang akan dijadikan sebagai topik penelitian, dilanjutkan dengan membuat latar belakang masalah sebagai. Kemudian terdapat tahap pemahaman terhadap studi literatur yang digunakan sebagai referensi teori untuk menguatkan pengetahuan dan ilmu-ilmu yang berkaitan dengan topik penelitian.

Pada tahap awal terdapat bagian penentuan perumusan masalah dan batasan masalah yang berfungsi sebagai koridor utama dalam pengerjaan pengerjaan penelitian. Setelah itu dilakukan penentuan tujuan penelitian yang bernilai positif terhadap objek penelitian berdasarkan perumusan masalah dan batasan masalah yang sudah ditetapkan

2) Tahap Pengujian

Tahap pengujian merupakan tahapan yang meliputi perencanaan, identifikasi data yang dibutuhkan, dan pengumpulan data terkait objek penelitian. Tahap *scope* perlu dilakukan untuk melakukan identifikasi awal mengenai objek penelitian dengan tujuan untuk memahami ruang lingkup penelitian dan menentukan metode serta *tool* yang tepat untuk digunakan selama pengujian.

Pengumpulan data dilakukan pada tahapan *information gathering* untuk mengetahui informasi umum mengenai objek penelitian sehingga dapat menggambarkan kemungkinan yang dapat terjadi pada objek penelitian. Pengujian dilanjutkan dengan dasar keilmuan berupa informasi umum dari objek penelitian untuk mendapatkan data yang lebih spesifik pada tahapan *vulnerability detection, vulnerability analysis* dan *attack and penetration testing*.

Berbagai jenis data yang sudah didapatkan pada tahapan sebelumnya, dilakukan analisis lebih lanjut untuk mendapatkan hasil yang sesuai dengan tujuan

penelitian sebelumnya. *Vulnerability detection* dilakukan dengan menggunakan *tools* Nessus, Burp suite *professional*, dan OWASP ZAP untuk menemukan celah kerentanan pada *website* target yang selanjutnya dilakukan klasifikasi kerentanan berdasarkan *risk level*. Celah kerentanan yang ditemukan pada *website* absensi praktikan dan asisten laboratorium praktek akan di evaluasi menggunakan metode VAPT. Berdasarkan klasifikasi kerentanan yang sudah ditentukan, tahap selanjutnya menentukan kerentanan apa saja yang akan dilakukan *penetration testing* dan *remediation* sesuai dengan tingkat dan dampak risikonya. Setelah hasil analisis didapatkan, perlu dilakukan *reporting* terhadap pihak terkait dengan objek penelitian sebagai bukti bahwa pengujian telah berhasil dilakukan.

3) Tahap Akhir

Tahap akhir merupakan bagian akhir dalam pelaporan yang berisikan penarikan kesimpulan hasil penelitian yang akan menjawab permasalahan yang telah dirumuskan serta dengan batasan masalah yang sudah ditetapkan. Selain itu, pada tahap ini disertakan juga lampiran dokumentasi hasil penelitian yang meliputi hasil tahapan pengujian maupun tahapan analisis dan penelitian.

3. HASIL DAN PEMBAHASAN

Pada bagian ini diberikan hasil penelitian yang dilakukan sekaligus dibahas secara komprehensip. Hasil bisa berupa gambar, grafik, tabel dan lain-lain yang mempermudah pembaca paham dan diacu di naskah. Jika bahasan terlalu panjang dapat dibuat sub-sub judul, seperti contoh berikut.

3.1. *Scope*

Pada penelitian ini, analisis kerentanan dijalankan pada port 443 atau *protocol* HTTPS menggunakan *tools* Nessus, Burp Suite, dan OWASP ZAP.

3.2. Perancangan Sistem

Pada proses pengujian celah keamanan dan analisis sebuah *website* dibutuhkan *hardware* dan *software* yang mendukung untuk berjalannya proses tersebut. Oleh karena itu dilakukan perancangan *hardware* dan *software* yang akan digunakan dalam proses pengujian dan analisis. Spesifikasi *hardware* dan *software* yang akan digunakan pada pengujian ini dapat dilihat pada Tabel 1 dan Tabel 2.

Tabel 1. Spesifikasi Hardware

Perangkat	Informasi	
Komputer Host	*Name*	Macbook Air Retina 2019
	Processor	1,6 GHz Dual-Core Inter Core i5
	Memory	8GB
	Storage	128GB SSD
	System Type	64-bit *Operating System*
	Operating System	macOS Ventura 13.2.1 (22D68)
Komputer Virtual	*Processor*	1
	Memory	4000 MB
	Storage	60 GB
	System Type	64-bit
	Operating System	Kali Linux 2023.1 Kali-rolling

Jurnal Sains Komputer & Informatika (J-SAKTI)
Volume 7 Nomor 2, September 2023, pp. 765-777
ISSN: 2548-9771/EISSN: 2549-7200
https://tunasbangsa.ac.id/ejurnal/index.php/jsakti

Tabel 2. Spesifikasi Software

Perangkat	Informasi
Operating System	Kali Linux 2023.1 Kali-rolling
Virtual Machine	VirtualBox 7.0.8
Scanning Tools	NMAP
Vulnerability Scanning and Analysis Tools	Nessus
	Burp Suite
	OWASP ZAP

3.4. Information Gathering

Pada pengujian kali ini, *website* target merupakan website Absensi Asisten Praktikum dan Praktikan Fakultas Rekayasa Industri yang akan diuji menggunakan *tool* NMAP. Hasil dari *information gathering* dapat dilihat pada Tabel 3.

Tabel 3. Hasil Information Gathering

Spesifikasi	Keterangan
Nama Domain	Tap2go-dev.virtualfri.id
IP Address	103.41.206.192
Port	22, 80, 81, 82, 84, 88, 443, 8802, 8080, 8081, 8084

3.5. Vulnerability Detection and Analysis

Pada penelitian ini *website* yang diuji adalah *website* Absensi Asisten Praktikum dan Praktikan Fakultas Rekayasa Industri dengan *domain* tap2go-dev.virtualfri.id menggunakan *tools* OWASP ZAP, Burp Suite, dan Nessus dengan melakukan *automated scanning* pada *website* target.

a) *Vulnerability Scanning* dengan OWASP ZAP

OWASP ZAP melakukan *scanning* berdasarkan *framework* yang mereka miliki, yaitu 10 *security risk*. Selain itu OWASP ZAP bekerja dengan melakukan dengan memantau lalu lintas *website* untuk mengidentifikasi serangan yang sedang berlangsung.

Untuk memulai proses *scan* pada *tool* OWASP ZAP diperlukan informasi mengenai URL dari website target. Pada pengujian ini penulis akan memilih tipe *scanning*, yaitu *automated scan* pada halaman utama OWASP ZAP. Setelah itu memasukan URL website target, yaitu https://tap2go-dev.virtualfri.id dan proses *scanning* sudah bisa dimulai. Hasil kerentanan yang terdeteksi oleh OWASP ZAP dapat dilihat pada Tabel 4.

Tabel 4. Kerentanan Yang Terdeteksi Oleh Owasp Zap

Jenis Kerentanan	Severity, Confidence
Content Security Policy (CSP) Header Not Set	*Medium, High*
Application Error Disclosure	*Medium, Medium*
Missing Anti-clickjacking Header	*Medium, Medium*
Vulnerable JS Library	*Medium, Medium*
Absence of Anti-CSRF Tokens	*Medium, Low*
Server Leaks Version Information via "Server" HTTP Response Header Field	*Low, High*

Jurnal Sains Komputer & Informatika (J-SAKTI)
Volume 7 Nomor 2, September 2023, pp. 765-777
ISSN: 2548-9771/EISSN: 2549-7200
https://tunasbangsa.ac.id/ejurnal/index.php/jsakti

Jenis Kerentanan	Severity, Confidence
Strict-Transport-Security Header Not Set	*Low, High*
Cookie without secure flag	*Low, Medium*
Cookie without SameSite Attribute	*Low, Medium*
Information Disclosure – Debug Error Messages	*Low, Medium*
Server Leaks Information via "X-Powered-By" HTTP Response Header Field(s)	*Low, Medium*
X-Content-Type-Options Header Missing	*Low, Medium*

b) *Vulnerability Scanning* dengan Burp Suite

Tool berikutnya yang digunakan pada pengujian ini adalah Burp Suite *professional*. Burp Suite melakukan *scan website* dengan menemukan celah keamanan dengan memantau dan mengintai lalu lintas web. Burp Suite digunakan untuk menemukan celah keamanan pada website target dengan URL https://tap2do-dev.virtualfri.id. Langkah pertama setelah menentukan *website* yang akan diuji, diperlukan untuk menentukan batasan konfigurasi pada jenis *scanning* yang akan dilakukan. Pada pengujian ini penulis menggunakan tipe *scan*, yaitu *crawl and audit* dan konfigurasi *scan* menggunakan tipe *deep*. Selain itu perlu ditentukan jenis *protocol settings* yang akan digunakan, yaitu *scan using my specified protocols*. Pemilihan tipe ini dikarenakan *website* yang diuji menggunakan *protocol* HTTPS dan pemilihan tipe *protocol settings* ini untuk mencegah Burp Suite melakukan *scanning* pada *protocol* HTTP *website* target. Hasil kerentanan yang terdeteksi oleh OWASP ZAP dapat dilihat pada Tabel 5.

Tabel 5. Kerentanan Yang Terdeteksi Oleh Burp Suite

Jenis Kerentanan	Severity, Confidence
Vulnerable version of the library 'moment.js' *found*	*High, Tentative*
TSL cookie without secure flag set	*Medium, Firm*
Vulnerable version of the library 'bootstrap' *found*	*Medium, Tentative*
Vulnerable version of the library 'jQuery' *found*	*Medium, Tentative*
Password field with autocomplete	*Low, Certain*
Strict transport security not enforced	*Low, Certain*

c) *Vulnerability Scanning* dengan Nessus

Nessus melakukan *scan website* dengan menemukan kerentanan pada infrastruktur IT, termasuk jaringan, *server*, perangkat jaringan, dan aplikasi. Untuk melakukan *scanning* pada Nessus perlu menetapkan *website* yang akan diuji, langkah selanjutnya yaitu memilih fitur '*New Scan*' pada Nessus untuk memulai proses *scan* yang baru dan memilih jenis "*Web Application Test*'. Pengisian target diisi dengan IP *host*, yaitu 103.41.206.192 dan nama *host*, yaitu tap2go-dev.virtualfri.id. Selanjutnya menentukan konfigurasi dari *scanning* yang akan dilakukan dan proses *scanning* akan dijalankan secara otomatis. Hasil kerentanan yang terdeteksi oleh OWASP ZAP dapat dilihat pada Tabel 6.

Tabel 6. Kerentanan Yang Terdeteksi Oleh Nessus

Jenis Kerentanan	Severity
PHP Unsupported Version Detection	Critical
HSTS Missing from HTTPS Serverz	Medium
Git Repositoy Served by Web Server	Medium

Jurnal Sains Komputer & Informatika (J-SAKTI)
Volume 7 Nomor 2, September 2023, pp. 765-777
ISSN: 2548-9771/EISSN: 2549-7200
https://tunasbangsa.ac.id/ejurnal/index.php/jsakti

Jenis Kerentanan	Severity
Web Application Potentially Vulnerable to Clickjacking	Medium
Web Server Allows Password Auto-Completion	Low

3.6. Attack and Penetration Testing

Berdasarkan hasil scanning menggunakan tools yang sudah ditetapkan, yaitu OWASP ZAP, Burp Suite, dan Nessus ditemukan berbagai macam kerentanan. Jenis kerentanan yang ditemukan berbeda beda, namun ada juga kerentanan yang memiliki kesamaan dikarenakan tools yang digunakan dapat melakukan identifikasi kerentanan secara menyeluruh, yaitu dari segi aplikasi, network dan framework website. Kemudian, dilakukan penggabungan kerentanan sebagai batasan dalam melakukan eksploitasi dan mitigasi. Kerentanan yang digunakan dipilih berdasarkan nilai risiko dan dampak risiko yang dimiliki, selain itu dipilih juga kerentanan yang muncul pada hasil scanning di dua atau tiga tools yang berbeda, dan kerentanan yang jika dilakukan mitigasi dapat menghilangkan kerentanan lainnya. Hasil penggabungan kerentanan tersebut perlu dilakukan analisis penyebab dan juga eksploitasi untuk membuktikan apakah kerentanan yang sudah ditemukan memiliki dampak yang berbahaya terhadap website target. Berikut adalah hasil penggabungan kerentanan yang dapat dilihat pada Tabel 7.

Tabel 7. Penggabungan Kerentanan

Jenis Kerentanan	Severity
PHP Unsupported Version Detection	Critical
HSTS Missing from HTTPS Server	Medium
Vulnerable version of the library 'moment.js' found	High, Tentative
Vulnerable version of the library 'bootstrap' found	Medium, Tentative
Vulnerable version of the library 'jQuery' found	Medium, Tentative
Content Security Policy (CSP) Header Not Set	Medium, High
Absence of Anti-CSRF Tokens	Medium, Low
Password field with autocomplete	Low, Certain
X-Content-Type-Options Header Missing	Low, Medium

Berikut adalah penjelasan masing masing *vulnerability* yang akan dilakukan *penetration testing*:

a) PHP *Unsupported Version Detected*
   Kerentanan ini ditemukan pada *website* target dikarenakan penggunaan PHP pada *server* tidak menggunakan versi terbaru. Sehingga beberapa kode dan *framework* tidak mendukung pada *server* yang digunakan pada *website*.

b) HSTS *Missing from* HTTPS *Server*
   Pada *website* target *server* HTTPS jarak jauh tidak menerapkan HTTP Strict Transport Security (HSTS). HSTS berguna untuk menginstruksikan *browser* agar hanya bisa berkomunikasi melalui HTTPS. Kurangnya HSTS membuat *website* rentan terhadap *protocol downgrade attack, cookie hijacking, dan man-in the middle-man attack*.

c) *Vulnerable version of the library* 'moment.js' *found*
Versi *package* moment.js yang digunakan pada *website* target adalah versi 2.13.0. Versi ini memiliki kerentanan terhadap serangan *Reguler Expression Denial of Service* (ReDoS), yaitu serangan yang menyerang algoritma sehingga membuat sistem menjadi sangat lambat atau menyebabkan kegagalan pada sistem aplikasi.

d) *Vulnerable version of the library* 'bootstrap' *found*
Kerentanan ini ditemukan pada target dikarenakan penggunaan bootstrap yang telah lama tidak dilakukan update. Versi bootstrap yang digunakan yaitu 3.1.0 dimana pada versi bootstrap dibawah 3.4.1 dan 4.3.x sebelum 4.3.1 rentan terhadap jenis serangan *cross-site scripting* pada bagian atribut data-template, data-content, dan data-title.

e) *Vulnerable version of the library* 'jQuery' *found*
JQuery dalam *website* ini berfungsi sebagai *library* javascript yang dapat mempermudah *developer* dalam mengatur interaksi antara javascript dengan *website* yang berjalan di sisi *user* (*client-side scripting*). Temuan hasil pengujian yang telah dilakukan bahwa versi JQuery yang digunakan tidak menggunakan versi terbaru yaitu versi 3.2.1 dimana pada JQuery dibawah versi 3.4.0 terdapat kelemahan dari segi *code* sehingga rentan terhadap sanitasi *code*.

f) *Content Security Policy* (CSP) *Header Not Set*
Kerentanan ini ditemukan dikarenakan *website* target tidak mengkonfigurasi *Content Secuity* Policy. CPS berfungsi untuk mendeteksi dan memitigasi serangan tertentu, jika *website* tidak melakukan konfigurasi CSP dapat menyebabkan *website* rentan terhadap serangan XSS, *data injection,* dan *clickjacking*.

g) *Absence of Anti-CSRF Tokens*
Kerentanan ini menyebabkan pengguna luar dapat melakukan HTTP *request* terhadap *website* target tanpa sepengetahuan admin *website*. *Cross-Site Request Forgery* (CSRF) merupakan salah satu jenis serangan yang bertujuan untuk membuat korban melakukan transaksi tanpa disadari. Hal ini terjadi karena *website* utama tidak memiliki kode unik autentikasi untuk menyelesaikan transaksi [15]. Kerentanan ini menyebabkan *website* target rentan terhadap jenis serangan *cross-site scripting* atau XSS.

h) *Password field with autocomplete*
Website target memungkinkan kata sandi pengguna dapat tersimpan pada perangkat masing-masing melalui *cache browser* yang digunakan. Hal ini memungkinkan kata sandi dapat terlihat ketika pengguna mengakses kembali *website* target. Menyimpan kata sandi di *cache browser* secara signifikan dapat meningkatkan risiko keamanan, karena pada komputer publik pengguna lainnya dapat melihat kata sandi pengguna sebelumnya jika mengakses *website* yang sama. [16]

i) X-Content-Type-Options *Header Missing*
X-Content-Type-Option adalah konfigurasi yang digunakan untuk mencegah MIME (Multi-purpose Internet Mail Extensios) *type sniffing* pada aplikasi berbasis *website* melalui *browser*. MIME *type* berfungsi untuk menentukan

jenis *file* yang dikirimkan kepada *browser* saat melakukan *request* terhadap server *browser* yang dituju [15]. *Sniffing* MIME menyebabkan *website* menjadi rentan terhadap serangan *cross-site scripting* (XSS) dapat menyisipkan *script payload* XSS terhadap *website* target dikarenakan MIME *type* dapat memberitahukan jenis *file* yang dibutuhkan, sehingga penyerang.

Sebelum melakukan mitigasi terhadap kerentanan yang dipilih, akan dilakukan penetration testing terlebih dahulu untuk memberikan bukti pendukung terhadap kerentanan yang ada. Hasil dari penetration testing dapat dilihat pada Tabel 8.

Tabel 8. Hasil Penetration Testing

Jenis Kerentanan	Hasil Penetration Testing
HSTS Missing from HTTPS Server	Dilakukan pengujian menggunakan pihak ketiga, yaitu *hstspreload* dan menunjukan hasil bahwa *website* target belum terkonfigurasi HSTS pada *header*
Content security policy (CSP) header not set	Kerentanan ini rentan terhadap serangan *clickjacking* dan dilakukan eksplotasi menggunakan kode <iframe> pada tombol *sign in* yang menyebabkan pengguna website akan diarahkan ke website yang tidak seharusnya..
Absence of Anti-CSRF Tokens	Kerentanan ini rentan terhadap serangan XSS dan dilakukan eksploitasi dengan metode XSS dengan jenis *reflected*. Dilakukan penyisipan kode pada *form input* dan berhasil menampilkan *pop up* berisikan informasi *path file* pada sistem.

3.7. Remediation

Pada tahapan *remediation* terdapat proses perbaikan atau mitigasi yang dilakukan pada *website* absensi praktikan dan asisten praktikum berdasarkan celah kerentanan yang sudah ditentukan. Kemudian terdapat proses verifikasi dengan melakukan pengujian ulang pasca mitigasi terhadap *website* absensi praktikan dan asisten praktikum untuk memverifikasi hasil mitigasi dan melakukan analisis alasan terhadap kerentanan yang belum termitigasi secara maksimal

a) Perancangan Mitigasi

Berikut adalah tahapan-tahapan mitigasi yang dilakukan pada masing masing kerentanan. Mitigasi dilakukan berdasarkan hasil analisis tingkat kerentanan dan hasil proses *attack and penetration testing*. Tahapan yang dilakukan dapat dilihat pada Tabel 9.

Tabel 9. Perancangan Mitigasi

Jenis Kerentanan	**Tahapan Perbaikan**
PHP Unsupported Version	Lakukan *upgrade* versi PHP menjadi yang terbaru pada sisi *server website* target.
Library Version Detection	Lakukan *upgrade library* yang terpasang di *web server*, berdasarkan hasil pengujian sebelumnya adalah dengan melakukan *upgrade* pada beberapa *library* berikut: - Moment.js - JQuery Bootstrap
HSTS Missing from HTTPS Server	Menambahkan *package mode_header* pada *web server*, kemudian melakukan konfigurasi kode pada file.htaccess untuk menjalankan

Jenis Kerentanan	Tahapan Perbaikan
	package yang sudah ditambahkan. Isi dari kode yang dikonfigurasikan di *file* .htaccess sebagai berikut: <IfModule mod_headers.c> Header always set Strict- Transport- Security "max-age=31536000; includeSubDomains; preload" </IfModule>
Content Security Policy (CSP) Header Not Set	Melakukan konfigurasi pada *file* config.php dengan menambahkan kode sebagai berikut: $config['headers'] = array('Content-Security-Policy: default-src \'self\'; script-src \'self\' \'unsafe-inline\' \'unsafe-eval\'; style-src \'self\' \'unsafe-inline\'; img-src \'self\' data:; font-src \'self\';'); dan $config['x_frame_options'] = 'SAMEORIGIN';
Absence of Anti-CSRF Tokens	Melakukan konfigurasi pada *file* config,php dengan menambahkan kode sebagai berikut: $config[‘csrf_protection’]=TRUE;
Password field with autocomplete	Mengkonfigurasi attribute ‘autocomplete=”off”’ di dalam tag *form password* pada *website* target. Kemudian melakukan konfigurasi pada *file* form_helper.php dengan kode sebagai berikut: $data[‘autocomplete’]=‘off’
X-Content-Type-Options Header Missing	Melakukan konfigurasi *header* ‘Content-Type’ dan *header* ‘X-Content-Type-Options’ menjadi *nonsniff* pada *file* config.php dengan kode sebagai berikut: $config['headers'] = array(‘X-Content-Type-Options nosniff’,); Serta melakukan konfigurasi pada *file* .htaccess dengan kode sebagai berikut: <IfModule mod_headers.c> Header set X-Content-Type- Options "nosniff’; </IfModule>

b) Verifikasi Pasca Mitigasi

Setelah dilakukan mitigasi, perlu dilakukan pengujian kembali terhadap *website* target dengan tujuan memverifikasi tahapan mitigasi yang sudah dilakukan. Hasil dari pengujian pasca mitigasi dapat dilihat pada Tabel 10.

Tabel 10. Hasil Scanning Pasca Mitigasi

Vulnerability Scanning Pra Mitigasi	Vulnerability Scanning Pasca Mitigasi	Keterangan
PHP Unsupported Version Detection (Critical)	PHP Unsupported Version Detection (Critical)	Perlu dilakukan pengecekan lebih lanjut terhadap keseluruhan konfigurasi dari sisi server oleh developer
HSTS Missing from HTTPS Server (Medium)	HSTS Missing from HTTPS Server (Information)	Implementasi mitigasi berhasil dilakukan, nilai risiko atau *severity* dari kerentanan tersebut sudah menurun.
Vulnerable version of the library ‘moment.js’ found	-	Kerentanan berhasil ditutup

Vulnerability Scanning Pra Mitigasi	Vulnerability Scanning Pasca Mitigasi	Keterangan
(High, Tentative)		
Vulnerable version of the library 'bootstrap' found (Medium, Tentative)	Vulnerable version of the library 'bootstrap' found (Medium, Tentative)	Implementasi sudah dilakukan untuk mengurangi kemungkinan terjadi serangan *cross-site scripting*. Perlu dilakukan pengecekan lebih lanjut terhadap keseluruhan *source code* oleh developer
Vulnerable version of the library 'jQuery' found (Medium, Tentative)	-	Kerentanan berhasil ditutup
Content Security Policy (CSP) Header Not Set (Medium, High)	Content Security Policy (CSP) Header Not Set (Medium, High)	Implentasi mitigasi sudah dilakukan namun perlu dilakukan pengecekan lebih lanjut dari segi kode nya
Absence of Anti-CSRF Tokens (Medium, Low)	Absence of Anti-CSRF Tokens (Medium, Low)	Ketika dilakukan mitigasi menyebabkan *website* tidak bisa diakses dan *error* ketika *login*. Maka dari itu untuk mitigasi ini tidak bisa dilakukan, perlu dilakukan analisis secara mendalam oleh *developer* untuk melakukan mitigasi CSRF
Password field with autocomplete (Low, Certain)	-	Kerentanan berhasil ditutup
X-Content-Type-Options Header Missing (Low, Medium)	X-Content-Type-Options Header Missing (Low, Medium)	Implentasi mitigasi sudah dilakukan namun perlu dilakukan pengecekan lebih lanjut dari segi kode nya

Berdasarkan hasil pengujian ulang celah keamanan pasca mitigasi terhadap *website* Absensi praktikan dan asisten laboratorium praktek, ditemukan perbedaan hasil dari pengujian celah keamanan sebelum dilakukan mitigasi. Terdapat beberapa celah keamanan yang sudah teratasi dengan baik dan tidak terdeteksi kembali saat dilakukan pengujian ulang setelah dilakukan mitigasi, dengan begitu mitigasi yang sudah dilakukan telah berhasil menutup beberapa celah kerentanan pada *website*. Namun ditemukan juga beberapa keretanan yang sudah dilakukan mitigasi namun kerentanan tersebut belum teratasi dengan baik. Sehingga diperlukan analisis lebih lanjut terkait kerentanan tersebut untuk dilakukan pengecekan secara menyeluruh mengenai *source code website* secara keseluruhan dan pada *package* bootstrap serta pengecekan konfigurasi pada *web server*. Pada tahapan mitigasi celah keamanan pada *website, developer* memiliki peranan penting untuk melakukan konfigurasi lebih dalam terkait dengan keamanan *website* melalui *soure code website*.

4. SIMPULAN

Berdasarkan pengujian menggunakan standar *Vulnerability Assessment and Penetration Testing* (VAPT) yang meliputi pengumpulan informasi, analisis keretanan, eksploitasi, dan mitigasi terhadap *website* Absensi praktikan dan asisten laboratorium praktek dapat disimpulkan bahwa dari hasil *vulnerability detection* menggunakan *tools* OWASP ZAP, Burpsuite dan Nessus dengan fokus hasil *scanning* terhadap *port* 443 ditemukan total sebanyak 27 kerentanan. Kerentanan yang ditemukan dilakukan analisis untuk menentukan tingkat risiko dan dampak risiko terhadap *website*. Hal ini berguna untuk tahapan selanjutnya, yaitu pembatasan jumlah kerentanan yang akan dilakukan eksploitasi dan mitigasi berdasarkan tingkat risiko kerentanan dan tahapan mitigasi dari masing masing kerentanan yang ditemukan. Dilakukan pemilihan sebanyak 9 kerentanan dari total 27 kerentanan yang ada, 9 kerentanan tersebut dilakukan *penetration testing* untuk membuktikan bahwa *website* rentan terhadap beberapa jenis serangan *cyber* dan dilakukan mitigasi untuk setiap kerentanan. Dari 9 kerentanan yang dilakukan mitigasi, terdapat 3 kerentanan yang berhasil dilakukan mitigasi dan tidak terdeteksi kembali saat proses *scanning* pasca mitigasi. Bagi kerentanan yang belum berhasil dilakukan mitigasi, perlu dilakukan analisis lebih lanjut baik dari segi *source code* secara keseluruhan maupun dari sisi konfigurasi *server*.

DAFTAR PUSTAKA

[1] Nurul, S., Anggrainy, S., & Aprelyani, S. (2022). *Faktor-Faktor Yang Mempengaruhi Keamanan Sistem Informasi: Keamanan Informasi, Teknologi Informasi Dan Network (Literature Review SIM)*. *3*(5). https://doi.org/10.31933/jemsi.v3i5.

[2] Bustami, A., & Bahri, S. (2020). Ancaman, Serangan dan Tindakan Perlindungan pada Keamanan Jaringan atau Sistem Informasi: Systematic Review. In *Jurnal Pendidikan dan Aplikasi Industri (UNISTEK)* (Vol. 7, Issue 2).

[3] gtslearning. (2014). *CompTIA Security+ SY0-401 Official Study Guide*. www.gtslearning.com.

[4] Goel, J. N., & Mehtre, B. M. (2015). Vulnerability Assessment & Penetration Testing as a Cyber Defence Technology. *Procedia Computer Science*, *57*, 710–715. https://doi.org/10.1016/j.procs.2015.07.458.

[5] Widjajarto, A., Lubis, M., & Ayuningtyas, V. (2021). Vulnerability and risk assessment for operating system (OS) with framework STRIDE: Comparison between VulnOS and Vulnix. In *Indonesian Journal of Electrical Engineering and Computer Science* (Vol. 23, Issue 3, pp. 1643–1653). Institute of Advanced Engineering and Science. https://doi.org/10.11591/ijeecs.v23.i3.pp1643-1653.

[6] Mu'min, Muh. A., Fadlil, A., & Riadi, I. (2022). Analisis Keamanan Sistem Informasi Akademik Menggunakan Open Web Application Security Project Framework. *Jurnal Media Informatika Budidarma*, *6*(3), 1468. https://doi.org/10.30865/mib.v6i3.4099.

[7] Aboelfotoh, S. F., & Hikal, N. A. (n.d.). A Review of Cyber-security Measuring and Assessment Methods for Modern Enterprises.

[8] Pangalila, R., Noertjahyana, A., & Andjarwirawan, J. (2015). *Penetration Testing Server Sistem Informasi Manajemen dan Website Universitas Kristen Petra*.

[9] Arvin Cadiente, K. R., Castro, R. A., van Gica, E. A., Marie Mora, K. C., & Ternio, J. v. (n.d.). Applying Vulnerability Assessment And Penetration Testing (Vapt) And

Network Enhancement On The Network Infrastructure Of Journey Tech Inc. In *Innovatus* (Vol. 3).

[10] Almaarif, A., & Lubis, M. (2020). Vulnerability Assessment and Penetration Testing (VAPT) Framework: Case Study of Government's Website. *International Journal on Advanced Science, Engineering and Information Technology*, *10*(5), 1874–1880. https://doi.org/10.18517/ijaseit.10.5.8862.

[11] Kuncoro, A. W., Informatika, J., Rahma, F., & Jurusan Informatika, M. E. (2022). *Analisis Metode Open Web Application Security Project (OWASP) pada Pengujian Keamanan Website: Literature Review*. https://www.sciencedirect.com.

[12] Sunardi, Riadi, I., & Ananda Raharja, P. (2019). Vulnerability Analysis of E-voting Application using Open Web Application Security Project (OWASP) Framework. In *IJACSA) International Journal of Advanced Computer Science and Applications* (Vol. 10, Issue 11). www.ijacsa.thesai.org.

[13] Indera, R., Budiono, A., & Yunan Kurnia Septo Hediyanto, U. (2023). Vulnerability Assessment Pada Situs Web KPPM FRI Dengan Burp Suite dan Intruder.

[14] Bayu Rendro, D., & Nugroho Aji, W. (2020). *Analisis Monitoring Sistem Keamanan Jaringan Komputer Menggunakan Software Nmap (Studi Kasus Di SMK Negeri 1 Kota Serang)*. *7*(2).

[15] Wardana, W., Almaarif, A., & Widjajarto, A. (2022). *Vulnerability Assessment and Penetration Testing On The Xyz Website Using Nist 800-115 Standard*. *7*(1).

[16] Bhargav-Spantzel, Abhilasha., ACM Digital Library., & Association for Computing Machinery. Special Interest Group on Security, A. (2011). *Proceedings of the 7th ACM workshop on Digital identity management.* ACM.

[17] Mlyatu, M. M., & Sanga, C. (2023). Secure Web Application Technologies Implementation through Hardening Security Headers Using Automated Threat Modelling Techniques. *Journal of Information Security*, *14*(01), 1–15. https://doi.org/10.4236/jis.2023.141001