Jurnal Riset Sistem Informasi Dan Teknik Informatika (JURASIK)
Volume 8, Nomor 2, Agustus 2023, pp 627-639
Terakreditasi Nomor 204/E/KPT/2022 | ISSN: 2527-5771/EISSN: 2549-7839
https://tunasbangsa.ac.id/ejurnal/index.php/jurasik

Strategi Security Mitigation Dengan VAPT Pada Website Rekruitasi Asisten Praktikum

Yolanda Hafitzhah[1], Umar Yunan Kurnia Septo Hediyanto[2], Muhammad Fathinuddin[3]
[1,2,3]Universitas Telkom, Indonesia
Email: yolandahafitzhah@student.telkomuniversity.ac.id[1], umaryunan@telkomuniversity.ac.id[2], muhammadfathinuddin@telkomuniversity.ac.id[3]

Abstract

In information technology, the internet is one of the things that is very important and useful at this time, one of which is the existence of a website. Currently the website is used by various types of activities, one of which is the XYZ Faculty. The website is used to assist students in taking care of all administrative needs needed for the process of assistant recruitment activities. Therefore, a vulnerability assessment was carried out using the VAPT method using several tools, namely OWASP ZAP, Acunetix, and NetSparker to find vulnerabilities on the website. In this test, 17 vulnerability gaps were found which were combined into 9 gaps for exploitation and mitigation. And the final result, 5 out of 8 security holes were successfully mitigated.

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

Abstrak

Dalam teknologi informasi, internet merupakan salah satu hal yang sangat penting dan berguna pada saat ini salah satunya dengan adanya website. Saat ini website digunakan oleh berbagai jenis kegiatan salah satunya pada Fakutas XYZ. Website tersebut digunakan unntuk membantu mahasiswa dalam mengurus semua keperluan adminitrasi yang diperlukan untuk proses kegiatan rekruitaitasi asisten. Oleh karena itu, dilakukan vulnerability assessment menggunakan metode VAPT menggunakan bebarap tools, yaitu OWASP ZAP, Acunetix, dan NetSparker untuk menemukan kerentanan pada website. Dalam pengujian tersebut, ditemukan 17 celah kerentanan yang digabungkan menjadi 9 celah untuk dilakukan eksploitasi dan mitigasi. Dan hasil akhir, 5 dari 8 celah keamanan berhasil dimitigasi.

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

1. PENDAHULUAN

Dalam teknologi informasi, internet merupakan salah satu hal yang sangat penting dan berguna pada saat ini. Internet memberikan dampak yang sangat luar biasa bagi kehidupan sehari-hari seperti cara masyarakat saat ini berkomunikasi, bekerja, mendapatkan informasi dan bersosial media. Beberapa perubahan yang sangat signifikan dengan adanya internet sangat memberikan keuntungan yang luar biasa bagi kehidupan sehari-hari. Dalam hal mendapatkan informasi, dengan adanya internet masyarakat dapat mengakses informasi dengan mudah salah satunya menggunakan *website*. Saat ini sudah banyak informasi-informasi yang dapat diakese melalui *website* dan juga *website* dapat menyajikan beberapa data informasi pribadi pengguna yang ada di dalamnya. Karena informasi yang terdapat di dalam *website* adalah hal yang sangat penting, oleh karena itu hanya orang yang berhak saja yang dapat mengakses informasi tersebut. Membahas mengenai data-data penting yang terdapat di dalam sebuah *website* tidak terlepas dari celah kerentanaan yang ada di dalam *website* tersebut. Kerentanan merupakan kelemahan yang terdapat di dalam sebuah *website* dan dapat menyebabkan

adanya potensi resiko sistem sehingga pihak yang tidak bertanggung jawab atau biasa yang disebut dengan *attacker* masuk ke dalam sistem dan melakukan eksploitasi untuk mendapatkan informasi penting yang terdapat di dalam setiap website memiliki jenis kerentanan yang berbeda[1].

Website rekruitasi praktikum merupakan salah satu *website* administrasi rekruitasi praktikum yang terdapat pada universitas XYZ. *Website* ini berfungsi untuk membantu mahasiswa dalam mengurus semua keperluan adminitrasi yang diperlukan untuk proses kegiatan rekruitaitasi asisten seperti melakukan pengisian data pribadi mahasiswa, pemilihan laboratorium dan melampirkan berkas-berkas yang dibutuhkan selama kegiatan proses rekruitasi asisten. Banyaknya data penting mahasiswa yang terdapat dalam portal ini sehingga perlu dilakukan tindakan keamanan yang dapat mengamankan data data mahasiswa yang terdapat di dalam portal akademik. Tingkat kerentanan dan serangan pada setiap *website* tentu berbeda-beda, sehingga diperlukan pengujian celah keamanan dan penilaian risiko pada *website* terkait dengan mempertimbangkan faktor faktor yang ada[2].

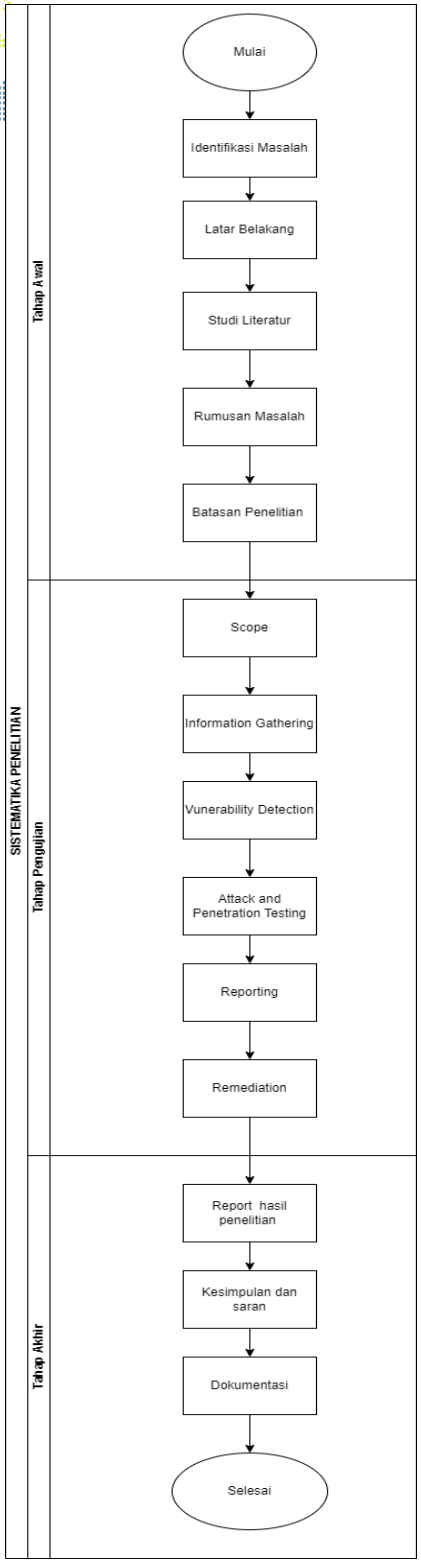
Dalam melakukan *security testing* menggunakan sebuah metode pengujian , dalam penelitian ini menggunakan metode *Vulnerability Asessment and Penetration Testing* (VAPT). *Vulnerability Asessment* merupakan kegiatan pengujian dengan menggunakan beberapa tools untuk menemukan celah keamanan yang terdapat pada *website* target sedangkan *penetration testing* adalah teknik keamanan yang digunakan oleh sebuah perusahaan atau organisasi untuk mengidentifikasi dan menguji kerentanan yang terdapat pada website. *Penetration testing* dilakukan dengan mensimulasikan penyarangan pada website target untuk membantu perusahaan dalam melakukan evaluasi dari celah keamanan yang terdapa pada *website* suatu perusahaan. Metode ini digunakan untuk membantu mengidentifikasi kerentanan keamanan yang terdapat pada sebuah *website* dan melakukan tindakan perbaikan untuk mengatasi kerentanan yang ditemukan supaya tingkat keamanan *website* lebih tinggi dan resiko serangan terhadap website menjadi berkurang[3].

Dalam melakukan pengujian kerentanan keamanan terhadap *website* rekruitasi asisten pada fakultas XYZ digunakan beberapa tools yaitu OWASP ZAP, Acunetix dan Netsparker. Ketiga tools ini digunakan untuk mengidentifikasi kerentanan yang terdapat *website* rekruitasi asisten. Kerentanan yang ditemukan akan dilakukan analisis dan mitigasi untuk mengurangi serangan yang dapat terjadi pada website[4].

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 Riset Sistem Informasi Dan Teknik Informatika (JURASIK)
Volume 8, Nomor 2, Agustus 2023, pp 627-639
Terakreditasi Nomor 204/E/KPT/2022 | ISSN: 2527-5771/EISSN: 2549-7839
https://tunasbangsa.ac.id/ejurnal/index.php/jurasik



Gambar 1. Sistematika Penelitian

1) Tahap Awal

Pada tahap awal penelitian ini adalah identifikasi permasalahan, setelah masalah sudah didapatkan makan tahapan selanjutnya adalah perumusan masalah untuk menentukan tujuan dari penelitian dengan menetapkan batasan masalah supaya penelitian ini tidak meluas. Dalam penelitian ini studi litaratur digunakan untuk mencari informasi terkait penelitian sebelumnya dengan permasalahan yang sesuai yang didapatkan dari sumber seperi jurnal dan buku.

2) Tahap Pengujian

Tahap pengujian dilakukan untuk perancancangan implementasi strategi pengujian [5]. Langkah awal pada tahap ini dengan merencanakan *tools* yang akan digunakan pada saat melakukan *penetration testing* tahapan ini disebut dengan *pre-engagement interactions.* Setelah *tools* yang digunakan untuk melakukan pengujian sudah ditetapkan, langkah berikutnya mengumpulkan informasi tentang website target. Selanjutnya membuat permodelan ancaman yang dibutuhkan untuk melakukan *penetration testing*. Tahapan selanjutnya mencari kerentanan yang terdapat dalam *website* proses pencarian kerentanan dapat dilakukan dengan berbagai cara tergantung pada komponen yang akan diuji. Setelah mendapatkan celah keamanan akan dilakukan serangan pada website target dan selanjutnya akan dilakukan analisis secara manual untuk melihat tingkat kerentanan pada website. Hasil dari tingkat kerentaan yang didapatkan akan akan dilakukan *exploitation* dengan membuat akses ke website dengan melewati keamanan dari hasil analisis yang telah dilakukan sebelumnya. Dalam melakukan pengujian ini digunakan beberapa tools yaitu OWASP ZAP, Acunetix dan NetSparker.

3) Tahap Akhir

Pada tahap ini terdapat hasil dari *vulnerability analysis* dan *exploitation* yang sudah dilakukan pengujian Kedua hasil tersebut akan dibuatkan laporan yang berisi dokumentasi pengujian dan kesimpulan, selanjutnya peneliti akan memberikan rekomendasi yang ditujukan kepada institusi terkait.

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.

a) Scope

Scope merupakan tahapan awal yang digunakan sebelum melakukan *vulnerability assessment*[6]. Tahapan ini juga menentukan perancangan pengujian untuk mendapatkan infromasi dari website rekruitasi asisten praktikum yang akan dilakukan pengujian pada port 443 yaitu HTTPS dengan menggunakan tools OWASP ZAP, Acuneti dan NetSparker.

b) Perancangan Sistem

Pada proses pengujian celah keamanan dan analisis sebuah *website* dibutuhkan perancangan sistem seperti *hardware* dan *software* yang mendukung untuk proses penelitian [7]. 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

Komponen Perangkat Keras	**Informasi**	
Acer Nitro AN515-54 (*Main OS*)	Processor	Intel(R) Core(TM) i5-9300H CPU @ 2.40GHz (8 CPUs), ~2.4GHz
	Memory	16384MB RAM
	Hard Disk	469 GB
	System Type	64-bit operating system, x64-based processor
	Operating System	Windows 11 Pro 64-bit (10.0, Build 22621)

Tabel 2. Spesifikasi Software

Tipe	**Komponen Software**	**Versi**
Operating System	Windows	11
Vulnerability Scanning and Analysis Tools	OWASP ZAP Net Sparker Acunetix	Community edition Professional Edition Premium

c) Information Gathering

Information gathering merupakan tahap awal yang dilakukan untuk mencari informasi lebih banyak mengenai *website* target dengan menggunakan tools yang dapat membantu penelitian ini [8]. *Tools* yang digunakan untuk tahapan *information gathering* adalah Zenmap. Hasil dari *information gathering* dapat dilihat pada Tabel3.

Tabel 3. Hasil Information Gathering

No	**Spesifikasi**	**Keterangan**
1	Nama Domain	iris-dev.virtualfri.id
2	Alamat IP	103.41.206.192
3	Port Yang Digunakan	22, 80, 81 , 82 , 83 , 84, 85, 88, 89, 90, 443, 5432, 8001, 8002, 8008, 8080

d) Vulnerability Detection and Analysis

Pada penelitian dilakukan untuk melakukan pengujian celah keamanan dengan melakukan pemindai kerentaan terhadap *website* target [9]. Pada penelitian ini, vulnerability detection dan analysis menggunakan tools OWASPZAP, Acunetix dan NetSparker.

e) *Vulnerability Scanning* dengan OWASP ZAP

ZAP digunakan untuk mendeteksi kerentanan pada suatu website dengan melakukan *scanning website* target [9]. Langkah pertama untuk melakukan pengujian celah keamanan yaitu memilih *website* target yaitu *website* rekruitasi asisten praktikum. Langkah pertama untuk melakukan pengujian celah keamanan yaitu memilih *website* target yaitu *website* rekruitasi asisten praktikum, kemudian melakukan pengujian kerentanan menggunakan ZAP,

Jurnal Riset Sistem Informasi Dan Teknik Informatika (JURASIK)
Volume 8, Nomor 2, Agustus 2023, pp 627-639
Terakreditasi Nomor 204/E/KPT/2022 | ISSN: 2527-5771/EISSN: 2549-7839
https://tunasbangsa.ac.id/ejurnal/index.php/jurasik

setelah dilakukan pengujian maka hasil scan akan didapatkan dan tahap selanjutnya adalah melakukan analisis terhadap hasil pengujian kerentanan yang didapatkan menggunakan *tool* ZAP. Hasil kerentanan yang terdeteksi oleh OWASP ZAP dapat dilihat pada Tabel 4.

Tabel 4. Kerentanan Yang Terdeteksi Oleh Owasp Zap

Jenis Kerentanan	Risk Level
Secure Pages Include Mixed Content (Including Scripts)	*Medium*
Absence of Anti-CSRF Tokens	*Low*
Cookie Without Secure Flag	*Low*
Cookie without Sam eSite Attribute	*Low*
Timestamp Disclosure - Unix	*Low*
Information Disclosure - Suspicious Comments	*Informational*

f) *Vulnerability Scanning* dengan NetSparker

Netsparker merupakan salah satu tool yang digunakan untuk melakukan pengujian keamanan dengan melakukan scanning pada website untuk menemukan kerentanan yang ada pada website target yaitu website rekruitasi asisten praktikum. Langkah pertama yaitu penentuan web target[10]. Kemudian melakukan pengujian celah keamanan dengan *scanning website* menggunakan NetSparker. Setelah mendapatkan hasil scan, *alert* yang terdeteksi akan ditampilkan dan selanjutnya akan dilakukan analisis terhadap hasil pengujian celah keamanan menggunakan tool NetSparker. Hasil kerentanan yang terdeteksi oleh NetSparker dapat dilihat pada Tabel 5.

Tabel 5. Kerentanan Yang Terdeteksi NetSparker

Jenis Kerentanan	Risk Level
Autocomplete is Enabled	*Low*
Passive Mixed Content ove	*Low*
Autocomplete Enabled (Password Field)	*Informational*
Forbidden Resource	*Informational*
Session Cookie Not Marked as Secure	*High*
Active Mixed Content over HTTPS	*Medium*
Critical Form Send to HTTP	*Medium*

g) *Vulnerability Scanning* dengan Acunetix

Acunetix merupakan salah satu tool yang digunakan dalam penelitian untuk melakukan pengujian keamanan pada website target [5]. Langkah pertama yaitu menentukan website target dengan memasukkan URL target selanjutnya akan dilakukan proses scanning pada website target. Setelah pemindaian selesai, Acunetix akan menghasilkan laporan tentang temuan dan kerentanan keamanan yang ditemukan. Hasil kerentanan yang terdeteksi oleh ACUNETIX dapat dilihat pada Tabel 6.

Tabel 6. Kerentanan Yang Terdeteksi Oleh Acunetix

Jenis Kerentanan	Risk Level
Git repository found	*High*
Development configuration files	*Medium*
User credentials are sent in clear text	*Medium*

Jenis Kerentanan	Risk Level
Vulnerable JavaScript libraries	*Medium*
Clickjacking: X-Frame-Options header	*Low*
Cookies with missing, inconsistent or contradictory properties	*Low*
Cookies without Secure flag set	*Low*
Documentation files	*Low*
HTTP Strict Transport Security (HSTS) not implemented	*Low*
Outdated JavaScript libraries	*Informational*

h) Attack and Penetration Testing

Berdasarkan pengujian yang telah dilakukan pada tahapan vulnerable detection menggunakan tiga tools yaitu OWASP ZAP, Acunetix dan Netsparker ditemukan berbagai jenis kerentanan dengan *risk level* yang berbeda-beda pada. Hal ini disebabkan oleh teknologi dari *tools* yang digunakan pada saat melakukan pengujian, namun terdapat beberapa jenis kerentanan yang memiliki kesamaan walau menggunakan tools yang berbeda. Oleh karena itu pada tahap ini dilakukan penggabungan jenis kerentanan yang akan dilakukan eksploitasi lebih lanjut kemudiaan dilakukan analisis kerentanan lebih dalam agar dapat dilakukan tahapan selanjutnya yaitu mitigasi [11]. Tahapan ini dilakukan untuk mengetahui kerentanan yang sudah ditemukan tidak memiliki potensi ancaman keaman pada *website* rekruitasi asisten praktikum.[12] Berikut adalah hasil penggabungan kerentanan yang dapat dilihat pada Tabel 7.

Tabel 7. Penggabungan Kerentanan

Jenis Kerentanan	Risk Level
Vulnerable JavaScript libraries	*Medium*
Clickjacking: X-Frame-Options header	*Low*
HTTP Strict Transport Security (HSTS) not implemented	*Low*
Cookie Without Secure Flag	*Low*
Cookie without SameSite Attribute	*Low*
Absence of Anti-CSRF Tokens	*Low*
Secure Pages Include Mixed Content (Including Scripts)	*Medium*
Autocomplete is Enabled	*Low*

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

1) *Secure Pages Include Mixed Content (Including Scripts)*
   Kerentanana ini terjadi karena halaman website dapat diakses melalui HTTPS *Hypertext Transfer Protocol Secure (*HTTPS) yang memiliki konten campuran sehingga Sebagian konten yang dikirimkan melalui *Hypertext Transfer Protocol* (HTTP) bukan HTTPS akibatnya halaman web mengandung komponen yang tidak aman. Kerentanan ini dapat mengakibatkan data yang tidak dienskripsi dapat disadap atau diganti.
2) *Vulnerable JavaScript libraries*
   Kerentanan ini ditemukan karena *website* target menggunakan satu atau lebih javascript [13].

Jurnal Riset Sistem Informasi Dan Teknik Informatika (JURASIK)
Volume 8, Nomor 2, Agustus 2023, pp 627-639
Terakreditasi Nomor 204/E/KPT/2022 | ISSN: 2527-5771/EISSN: 2549-7839
https://tunasbangsa.ac.id/ejurnal/index.php/jurasik

3) *Clickjacking: X-Frame-Options header*
   Kerentanan *Clickjacking: X-Frame-Options header* adalah kerentanan yang disebabkan karena server tidak mengirimkan *X-Frame-Options header* sehingga beresiko terkena serangan c*lickjacking. X-Frame-Options header* dapat digunakan untuk menunjukkan sebuah browser diizinkan untuk mengubah halaman di dalam sebuah frame atau iframe yang terdapat di halaman website [14].
4) HTTP Strict Transport Security (HSTS) not implemented
   Kerentanan ini muncul karena belum mengimplementasikan *HTTP Strict Transport Security (HSTS)* pada website target. HTTPS perlu digunakan agar dapat menunjukkan browser bahwa situs web hanya dapat diakses menggunakan HTTPS [15].
5) *Cookie Without Secure Flag*
   Kerentanan *Cookie Without Secure Flag* ialah kerentanan yang terjadi karena cookie diatur tanpa *secure flag* sehingga cookie dapat diakses melalui koneksi yang tidak terenskripsi *Cookie without SameSite Attribute.*
6) *Absence of Anti - Cross Site Request Forgery (CSRF) Tokens*
   Cross Site Request Forgery adalah serangan yang memaksa pengguna yang diautentikasi untuk mengirimkan permintaan ke website yang sedang diautentikasi. Contoh dari serangan ini adalah penyerang dapat mengubah alamat email pengguna atau bisa melakukukan tranksaksi tanpa sepengatahuan pengguna.
7) *Autocomplete is Enabled*
   Netsparker mendeteksi kerentanan yaitu fitur *Autocomplete di* pada satu atau beberapa kolom formulir yang mengandung data sensitif pengguna dan disimpan oleh browser yang mengakses website sehingga dapat menyebabkan penyerang menggunakan Kembali website yang digunakan dengan data yang sama tanpa sepengetahuan pengguna.

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
Clickjacking: X-Frame-Options header	Pengujian *Clickjacking* menggunakan salah satu fitur yang ada pada tool burp suite yaitu fitur burp clickbandit. Fitur ini digunakan untuk menguji kerentanan *clickjacking* dengan membuat serangan yang memiliki inframe pada website target yang asli
HTTP Strict Transport Security (HSTS) not implemented	Pengujian dengan melakukan inspect pada *network* pada *browser* yang dikirimkan melalui HTTP. Pengujian ini dilakukan untuk mengetahui respon dari request yang dikirimkan melalui *request* url.

Jenis Kerentanan	Hasil Penetration Testing
Cookie without SameSite Attribute	Dilakukan pengecekan pada website untuk dapat mengetahui apakah website memilki cookie dengan attribute SameSite. Proses pengecekan dilakukan dengan melakukan inspect pada website target dan dapat dilihat pada bagian *application website* rekruitasi tidak memiliki cookie SameSite attribute.
	dan berhasil menampilkan *pop up* berisikan informasi *path file* pada sistem.

i) Remediation

Pada tahapan ini dilakukan perancangan mitigasi dari kerentanan yang sudah ditentukan sebelumnya [6]. Selanjutnya akan dilakukan implementasi mitigasi terhadap sistem dan melakukan pengujian ulang untuk mengetahui kerentanan yang telah berhasil dihilangkan. Kerentanan yang tidak berhasil dihilangkan setelah dilakukan tahapan mitigasi akan dilakukan analisis terhapat kerentaan yang tidak dapat diatasi tersebut.

Berikut adalah beberapa rekomendasi dan tahapan yang dapat diberikan pada terhadap website asisten rekruitasi praktikum berdasarkan dengan jenis kerentanan yang sudah ditemukan dan diuji,. Tahapan dan rekomendasi perbaikan yang dilakukan dapat dilihat pada Tabel 9.

Tabel 9. Perancangan Mitigasi

Jenis Kerentanan	Tahapan Perbaikan
Vulnerable JavaScript libraries	Melakukan upgrade ke versi terbaru *java script libraries* dan update beberapa skrip di file php.
Clickjacking: X-Frame-Options header	Melakukan konfigurasi pada website dengan menambahkan konfigurasi *header X-Frame-Options dan header* CSP.
HTTP Strict Transport Security (HSTS) not implemented	Mengimplementasikan HTTP Strict Transport Security pada website target.
Secure Pages Include Mixed Content (Including Scripts)	Memastikan semua konten yang ada pada halaman website termasuk library yang ada pada website menggunakan SSL/TLS yang aman dengan cara halaman pada website yang memiliki SSL/TLS hanya boleh mengirimkan konten melalui SSL/TLS dan tidak boleh ada konten apapu yang dikirimkan melalui HTTP yang tidak dienskripsi.
Cookie Without Secure Flag	Mengaktifkan *cookie security* pada website supaya setiap cookie yang berisi informasi sensitif atau token yang berbahaya, penting bahwa cookie tersebut dikirimkan melalui saluran yang terenskripsi dengan memastikan *secure flag* diaktifkan.
Cookie without SameSite Attribute	Pastikan semua attribute SameSite dikonfigurasikan menjadi " lax" dan " strict" untuk semua cookie yang terdapat pada website.
Absence of Anti-CSRF Tokens	Mengaktifkan fitur CSRF *protection* pada website dengan menggunakan fungsi csrf_token() yang sudah ada pada framework CodeIgneter untuk membuat dan menambahkan token csrf.
Autocomplete is Enabled	Menambakan atribut autocomplete="off" ke semua tag form atau kolom "input" yang terdapat website. Berikut merupakan

Jenis Kerentanan	Tahapan Perbaikan
	konfigurasi ke attributes yang akan dinokatifkan: <input type="text" name="username" autocomplete="off"> Konfigurasi ini dapat ditambahkan pada beberapa bidang ang sensitif pada website seperti kata sandi atau bidang lainnya yang tidak boleh dilengkapi secara otomatis oleh browser.

a) Verifikasi Pasca Mitigasi

Berdasarkan tahapan mitigasi yang telah dilakukan sebelumnya, dilakukan kembali pengujian ulang dengan menggunakan beberapa tools yaitu OWASP ZAP, Acunetix dan Netsparker. Pengujian ulang setelah mitigasi ini dilakukan untuk mengetahui hasil dari implementasi mitigasi yang sudah dilakukan pada tahapan sebelumnya berhasil atau tidak mengatasi kerentanan yang terdapat pada *website* asisten rekruitasi praktikum. 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
Vulnerable JavaScript libraries	*Vulnerable JavaScript libraries*	Sudah dilakukan *upgrade library* namun perlu dilakukan pengecekan secera menyeluruh dari segi code.
Clickjacking: X-Frame-Options header	*Clickjacking: X-Frame-Options header*	Implementasi mitigasi sudah dilakukan akan tetapi kerentanan ini masih perlu pengcekan secara berkala agar dapat meningkatkan keaman pada *website* target.
HTTP Strict Transport Security (HSTS) not implemented	*HTTP Strict Transport Security (HSTS) not implemented*	Sudah dilakukan implementasi HSTS pada website akan tetapi perlu dilakukan pengecekan lebih lanjut oleh web developer dari segi code dan konfigurasi yang terdapat pada *website* target.
Secure Pages Include Mixed Content (Including Scripts)	-	Kerentanan berhasil ditutup
Cookie Without Secure Flag	-	Kerentanan berhasil ditutup
Cookie without SameSite Attribute	-	Kerentanan berhasil ditutup
Absence of Anti-CSRF Tokens	-	Kerentanan berhasil ditutup
Autocomplete is Enabled	-	Kerentanan berhasil ditutup

Berdasarkan hasil pengujian ulang setelah tahapan mitigasi terdapat beberapa kerentanan yang sudah hilang dan juga kerentanan yang yang tidak berhasil dimitigasi. Kerentanan yang tidak berhasil dimitigasi perlu dilakukan perbaikan dan pengecekan lebih lanjut dari segi code dan konfigurasi yang terdpat pada *website* rekruitasi asisten praktikum. Namun terdapat beberapa kerentanan yang berhasil dimitigasi dengan baik sehingga tahapan mitigasi ini dapat

meningkatkan keamanan yang terdapat pada *website* target.

b) Verifikasi Pasca Mitigasi

Berdasarkan tahapan mitigasi yang telah dilakukan sebelumnya, dilakukan kembali pengujian ulang dengan menggunakan beberapa tools yaitu OWASP ZAP, Acunetix dan Netsparker. Pengujian ulang setelah mitigasi ini dilakukan untuk mengetahui hasil dari implementasi mitigasi yang sudah dilakukan pada tahapan sebelumnya berhasil atau tidak mengatasi kerentanan yang terdapat pada *website* asisten rekruitasi praktikum. Hasil dari pengujian pasca mitigasi dapat dilihat pada Tabel 11.

Tabel 11. Hasil Scanning Pasca Mitigasi

Vulnerability Scanning Pra Mitigasi	**Vulnerability Scanning Pasca Mitigasi**	**Keterangan**
Vulnerable JavaScript libraries	*Vulnerable JavaScript libraries*	Sudah dilakukan *upgrade library* namun perlu dilakukan pengecekan secera menyeluruh dari segi code.
Clickjacking: X-Frame-Options header	*Clickjacking: X-Frame-Options header*	Implementasi mitigasi sudah dilakukan akan tetapi kerentanan ini masih perlu pengcekan secara berkala agar dapat meningkatkan keaman pada *website* target.
HTTP Strict Transport Security (HSTS) not implemented	*HTTP Strict Transport Security (HSTS) not implemented*	Sudah dilakukan implementasi HSTS pada website akan tetapi perlu dilakukan pengecekan lebih lanjut oleh web developer dari segi code dan konfigurasi yang terdapat pada *website* target.
Secure Pages Include Mixed Content (Including Scripts)	-	Kerentanan berhasil ditutup
Cookie Without Secure Flag	-	Kerentanan berhasil ditutup
Cookie without SameSite Attribute	-	Kerentanan berhasil ditutup
Absence of Anti-CSRF Tokens	-	Kerentanan berhasil ditutup
Autocomplete is Enabled	-	Kerentanan berhasil ditutup

Berdasarkan hasil pengujian ulang setelah tahapan mitigasi terdapat beberapa kerentanan yang sudah hilang dan juga kerentanan yang yang tidak berhasil dimitigasi . Kerentanan yang tidak berhasil dimitigasi perlu dilakukan perbaikan dan pengecekan lebih lanjut dari segi code dan konfigurasi yang terdpat pada *website* rekruitasi asisten praktikum. Namun terdapat beberapa kerentanan yang berhasil dimitigasi dengan baik sehingga tahapan mitigasi ini dapat meningkatkan keamanan yang terdapat pada *website* target.

4. SIMPULAN

Berdasarkan pengujian yang telah dilakukan menggunakan *Vulnerability Assessment and Penetration Testing* (VAPT) yang meliputi pengumpulan informasi,

analisis kerentanan, eksploitasi, dan mitigasi terhadap *website* rekruitasi asisten praktikum dapat disimpulkan bahwa dari hasil *vulnerability detection* menggunakan *tools* OWASP ZAP, Acunetix dan NetSparker dengan hasil *scanning* ditemukan kerentanan dengan jumlah 23 kerentanan. Kerentanan yang telah ditemukan dilakukan analisis untuk menentukan tingkat risiko dan dampak risiko terhadap *website* target. Hal ini bertujuan untuk langkah berikutnya, yakni mengurangi jumlah kerentanan yang akan dieksploitasi dan mitigasu untuk mengurangi risiko kerentanan yang terdapat terhadap website dengan melakukan tindakan pencegahan sesuai dengan tingkat risiko dan langkah-langkah mitigasi yang sesuai untuk setiap kerentanan yang ditemukan. Dilakukan pemilihan jenis kerentanan sebanyak 8 kerentanan dari jumlah total 23 kerentanan yang ada, 8 kerentanan tersebut dilakukan *penetration testing* untuk membuktikan bahwa *website* rentan terhadap beberapa jenis serangan *attacker* dan dilakukan mitigasi untuk setiap kerentanan. Dari 8 kerentanan yang dilakukan mitigasi, terdapat 5 kerentanan yang berhasil dilakukan mitigasi dan tidak terdeteksi kembali saat proses *scanning* pasca mitigasi. Kerentanan yang belum berhasil dilakukan mitigasi, perlu dilakukan analisis lebih lanjut baik dari segi *source code* secara keseluruhan maupun dari sisi konfigurasi *server* oleh *web developer*.

DAFTAR PUSTAKA

[1] K. D. Sharma and R. Jhunjhunwala, "Web Application Security Using VAPT JatinKushwah , Kushagra Dutt Sharma , Raj Jhunjhunwala , Tanisha," vol. 2, no. 9, pp. 389–394, 2020, doi: 10.35629/5252-0209389394.

[2] A. Kakareka, "What Is Vulnerability Assessment?," *Comput. Inf. Secur. Handb.*, pp. 483–494, 2017, doi: 10.1016/B978-0-12-803843-7.00031-4.

[3] K. A. R. Cadiente, R. A. Castro, E. A. Van Gica, K. C. Marie Mora, and J. V Ternio, "Applying Vulnerability Assessment and Penetration Testing (Vapt) and Network Enhancement on the Network Infrastructure of Journey Tech Inc," *Innovatus*, vol. 3, no. 1, pp. 2651–6993, 2020.

[4] Fabiana Meijon Fadul, "Analisis Keamanan Website Prodi Sistem Informasi Uinsu Menggunakan Metode ABC" vol. 4, no. 4, pp. 325–329, 2019.

[5] A. Zirwan, "Pengujian dan Analisis Kemanan Website Menggunakan Acunetix Vulnerability Scanner," *J. Inf. dan Teknol.*, vol. 4, no. 1, pp. 70–75, 2022, doi: 10.37034/jidt.v4i1.190.

[6] Y. Khera, D. Kumar, S. Sujay, and N. Garg, "Analysis and Impact of Vulnerability Assessment and Penetration Testing," *Proc. Int. Conf. Mach. Learn. Big Data, Cloud Parallel Comput. Trends, Prespectives Prospect. Com. 2019*, no. May, pp. 525–530, 2019, doi: 10.1109/COMITCon.2019.8862224.

[7] F. Yudha and A. M. Panji, "Perancangan Aplikasi Pengujian Celah Keamanan Pada Aplikasi Berbasis Web," *Cyber Secur. dan Forensik Digit.*, vol. 1, no. 1, pp. 1–6, 2018, doi: 10.14421/csecurity.2018.1.1.1216.

[8] OWASP, "OWASP 2021," *OWASP Top 10- 2021*, 2021. https://owasp.org/Top10/#welcome-to-the-owasp-top-10-2021

[9] Dona Rose Mathew and Jetty Benjamin, "Penetration Testing and Vulnerability Scanning of Web Application Using Burp Suite," *Natl. Conf. Emerg. Comput. Appl.*, vol. 3, no. 1, pp. 271–277, 2021, doi: 10.5281/zenodo.5094090.

[10] C. Joshi and K. Singh, "Performance Evaluation of Web Application Security Scanners

for More Effective Defense," *Int. J. Sci. Res. Publ.*, vol. 6, no. 6, p. 660, 2016, [Online]. Available: www.ijsrp.org

[11] A. W. Kuncoro, J. Informatika, F. Rahma, and M. E. Jurusan Informatika, "Analisis Metode Open Web Application Security Project (OWASP) pada Pengujian Keamanan Website: Literature Review," *Automata*, vol. 3, no. 1, pp. 1–5, 2021, [Online]. Available: https://www.sciencedirect.com

[12] I. G. A. S. Sanjaya, G. M. A. Sasmita, and D. M. Sri Arsa, "Information technology risk management using ISO 31000 based on issaf framework penetration testing (Case study: Election commission of x city)," *Int. J. Comput. Netw. Inf. Secur.*, vol. 12, no. 4, pp. 30–40, 2020, doi: 10.5815/ijcnis.2020.04.03.

[13] A. Zerouali, V. Cosentino, T. Mens, G. Robles, and J. M. Gonzalez-Barahona, "On the Impact of Outdated and Vulnerable Javascript Packages in Docker Images," *SANER 2019 - Proc. 2019 IEEE 26th Int. Conf. Softw. Anal. Evol. Reengineering*, no. March, pp. 619–623, 2019, doi: 10.1109/SANER.2019.8667984.

[14] Patel and R. Goyena, "HTTP Header Field X-Frame-Options," *J. Chem. Inf. Model.*, vol. 15, no. 2, pp. 9–25, 2019.

[15] C. Jackson and A. Barth, "Hsts," pp. 1–46, 2012.