KESATRIA: Jurnal Penerapan Sistem Informasi (Komputer & Manajemen)
Terakreditasi Nomor 204/E/KPT/2022 | Vol. 5, No. 4, Oktober (2024), pp. 1732-1740

Optimalisasi Analisis Keamanan Menggunakan Acunetix Vulnerability Pada Rekam Medis Elektronik

Zulfiqar Tamin[1], Yuhandri[2], Sumijan[3]*
[1,2,3]Fakultas Ilmu Komputer, Universitas Putra Indonesia "YPTK" Padang, Indonesia
E-mail: fiqartamin@gmail.com[1], yuhandriyunus@ gmail.com[2],soe@upiyptk.org[3]

Abstract

The use of the internet and web applications has significantly increased across various sectors, including education, healthcare, finance, and entertainment. However, web applications are highly vulnerable to various types of cyberattacks, such as SQL Injection, Cross-Site Scripting (XSS), and code injection, which can threaten the confidentiality, availability, and integrity of data. In line with technological advancements, the 2022 Ministry of Health regulation mandates that all healthcare facilities in Indonesia implement Electronic Medical Records (EMR). Universitas Andalas Hospital (RS UNAND) has adhered to this policy by developing a web-based EMR system. This study aims to evaluate and analyze the security of the EMR application used at RS UNAND. The Vulnerability Assessment process in this study was conducted using the Acunetix Web Vulnerability Scanner tool, which is designed to identify and assess vulnerabilities in web applications. The results of the first scan revealed that the RS UNAND EMR application had significant vulnerabilities, with a threat level of 3 (high). This scan identified 573 alerts, including 1 high-level, 253 medium-level, 2 low-level, and 317 informational alerts. These issues were followed by a thorough recap and further analysis to determine optimization steps. Several major vulnerabilities identified included HTML Form Without CSRF Protection, User Credentials Sent in Clear Text, Directory Listing, Source Code Disclosure, Git Repository Found, Multiple Vulnerabilities Fixed in PHP Versions, and Slow HTTP Denial of Service Attack. Optimization measures were then taken through a comprehensive review of the source code and enhancements to the security features of the EMR application. After the optimization, the second scan showed a significant reduction in the threat level, with the RS UNAND EMR application dropping to threat level 1 (low), with 12 alerts, consisting of 0 high and medium-level alerts, 9 low-level alerts, and 3 informational alerts. This study underscores the importance of regular security assessments and the optimization of security features to protect sensitive data in electronic medical record systems.

Keywords: *We would like to encourage you to list your keywords in this section.*

1. Pendahuluan

Penggunaan internet telah meningkat pesat secara global, dan penggunaan aplikasi *web* dalam banyak bidang kehidupan sehari-hari, seperti pendidikan, kesehatan, keuangan, dan hiburan, juga meningkat. Namun terdapat peningkatan jumlah masalah keamanan aplikasi *web* yang secara langsung mengancam kerahasiaan, ketersediaan, dan integritas data[1]. Aplikasi *web* rentan terhadap serangan keamanan, seperti *SQL injection*, yang dapat membahayakan data dan privasi pengguna[2]. Berbagai solusi telah diusulkan untuk mengurangi keparahan ancaman ini, seperti *firewall aplikasi web* (*WAF*). Serangan *SQL Injection Attack* (*SQLIA*) merupakan salah satu serangan yang paling parah yang dapat digunakan terhadap aplikasi *web* berbasis database[3]. Aplikasi *web* merupakan *platform* yang populer dalam menyampaikan informasi melalui internet, menyediakan berbagai

ISSN: 2720-992X

layanan online seperti situs jejaring sosial, email, perbankan internet, dan aplikasi *e-commerce* dengan memanfaatkan beragam teknologi dan komponen *web*. Meskipun demikian, aplikasi *web* rentan terhadap serangan keamanan, seperti *cross-site scripting* (*XSS*) dan injeksi kode, yang dapat mengancam keamanan data dan privasi pengguna[4]. Berbagai ancaman ini, penting untuk menerapkan solusi keamanan yang efektif guna melindungi aplikasi web dan data pengguna dari potensi serangan.

Rekam Medis adalah dokumen yang memuat data identitas pasien, pemeriksaan, pengobatan, tindakan, dan pelayanan lain yang diberikan kepada pasien. Rekam Medis Elektronik adalah bentuk rekam medis yang disusun menggunakan sistem elektronik untuk pengelolaan rekam medis[5].

Penilaian kerentanan aplikasi *web* melibatkan deteksi, analisis, dan peringatan terhadap komponen rentan dalam aplikasi *web*. Pengujian penetrasi dilakukan dengan tujuan mengeksploitasi kerentanan untuk menilai kelayakan ancaman dunia maya yang berpotensi memengaruhi sebuah organisasi[6].

Vulnerability Assessment (*VA*) merupakan bagian dari *risk assessment* yang terdiri dari *risk analysis*, *policy development*, *training and implementation*, dan *vulnerability assessment and Penetration Testing*. *Vulnerability Assessment* (*VA*) adalah proses pemindaian sistem atau *software* dan jaringan untuk mengetahui kelemahan dan celah yang ada, celah ini memberikan *backdoor* ke penyerang untuk menyerang korban. Sebuah sistem sebaiknya memiliki akses kontrol terhadap *vulnerability*, *boundary condition vulnerability*, *input validation vulnerability*, *authentication vulnerabilities*, *configuration weakness vulnerabilities*, dan *exception handling vulnerabilities*[7].

Vulnerability Assessment (*VA*) ini juga pernah diimplementasikan pada aplikasi *Open Journal System* (*OJS*) versi 2.4.7 menggunakan *tool OWASP* sebagaimana penelitian yang dilakukan oleh Guntoro, hasil dari penelitian tersebut menemukan bahwa total celah atau vulnerability yang ditemukan berjumlah 6049 dimana dapat dikatakan bahwa *OJS* versi 2.4.7 memiliki banyak celah atau kerentanan yang tidak direkomendasi untuk di gunakan, sehingga peneliti menyarankan agar user menggunakan versi terbaru yang dikeluarkan oleh pihak *OJS Public knowledge project* (*PKP*)[8].

Penelitian mengenai *Vulnerability Assessment* (*VA*) ini juga pernah dilakukan oleh Listartha, peneliti mencari kerentanan dengan teknik yang terotomatis dan manual dengan mencari kerentanan yang diketahui berdasarkan *OWASP 2017* dengan menggunakan aplikasi *Burp Suite*. Hasil deteksi kerentanan kemudian dipetakan dalam tiga tingkat bahayanya dengan melihat risiko eksploitasinya. Hasil penelitian ini mampu mengeksploitasi celah-celah keamanan yang ada pada *website* target meskipun sistem tersebut dibangun menggunakan *framework* yang telah teruji sekalipun[9].

Al Fajar juga melakukan penelitian tentang *Vulnerability Assessment* (*VA*) mengguankan *tool Acunetix* untuk melakukan *scanning* kerentanan pada *Website* studi kasus dan ditemukan kerentanan yang bersifat kredensial seperti serangan *SQL Injection*, direktori berupa *listing*, *HTML* tanpa perlindungan *Cross Site Request Forgery* (*CSFR*), dan serangan *Cross Site Scripting* (*XSS*) yang mencapai *level High Critical*, sehingga keamanan pada aplikasi *Website* tersebut belum dikatakan aman dengan ditemukannya *web alert* yang berbahaya. Penulis pada penelitian ini memberikan rekomendasi terkait penelitan lanjutan agar *vulnerability assessment* ini dilanjutkan pada penelitian yang lebih mendalam mengenai kerentanan aplikasi dari segi temporal dan environmental agar penilaian dari kerentanan yang dialami memiliki penilaian yang lebih akurat[10].

Penelitian yang dilakukan oleh Kristianto, Rahman and Bahri pada *Website* servio diperoleh kerentanan–kerentanan seperti *HTML* tanpa perlindungan *CSRF*, *clickjacking*, dan beberapa *web alert* informational. Hasil yang ditemukan *Acunetix* berada pada level medium, yang berarti kerentanan terjadi karena kesalahan konfigurasi dan *site coding* yang lemah[11]. Sandy dan Solihin menyebutkan bahwa untuk mengurangi kerentanan

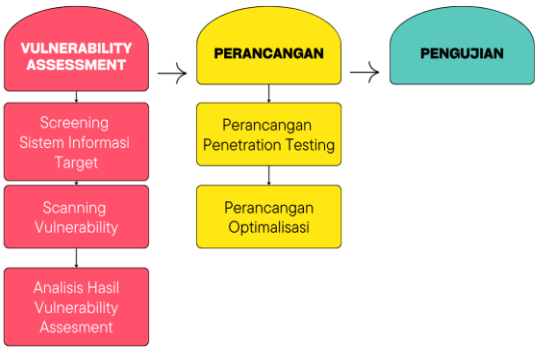
pada keamanan sistem serta mengurangi risiko kemungkinan kehilangan data, maka perlu dilakukannya audit pada sistem *e-Learning* yang diangun menggunakan sistem NIST[12].

Berdasarkan dari penelitian sebelumnya yang sudah dilakukan, peneliti melakukan analisa dan pengujian menggunakan *tool Acunetix Web Vulnerability Scanner* dan kerentanan ini dikelompokan kedalam *OWASP Top 10 2021*. Hasil pengujian selanjutnya dilakukan optimalisasi agar celah kerentanan yang berkategori *high* dan *medium* mejadi hilang, karena kategori celah kategori ini merupakan pintu masuk oleh *hacker* untuk masuk lebih jauh kedalam aplikasi yang ditelit.

2. Metodologi Penelitian

All Metode penelitian ini dilakukan secara sistematis agar hasil yang diperoleh dapat dijadikan pedoman dalam melakukan penelitian, agar hasilnya sesuai dengan tujuan yang telah ditetapkan dan dapat dilaksanakan dengan baik. Kerangka kerja penelitian ini menjelaskan masalah dan menganalisisnya sampai pada kesimpulan dan rekomendasi.

Penelitian ini bertujuan untuk melakukan uji dan analisis *vulnerability* Aplikasi Rekam Medis Elektronik, uji dan analisis *vulnerability* dilakukan menggunakan aplikasi *Acunetix Web Vulnerability Scanner*. Mempermudah proses penilaian kerentanan, dibuat kerangka penelitian seperti Gambar 1.



Gambar 1. Kerangka Penelitian

2.1. *Vulnerability Assessment*

Proses penilaian kerentanan dilakukan dengan menggunakan alat *Acunetix Vulnerability Web Scanner*.

2.1.1. *Screening* Sistem Informasi Target

Proses *vulnerability assessment* pada sistem informasi target ini akan dilakukan sebanyak 2 kali iterasi. *Scanning* iterasi pertama akan dilakukan sebelum optimalisasi untuk mengetahui *alert* dan kerentanan yang terdeteksi oleh *Acunetix*, sedangkan *scanning* iterasi kedua akan dilakukan setelah proses optimalisasi untuk mengetahui perubahan yang terjadi pada aplikasi.

2.1.2. *Scanning vulnerability*

Scanning vulnerability merupakan proses identifikasi dan penilaian kerentanan keamanan dalam sistem informasi, perangkat lunak, atau jaringan. Proses ini dilakukan untuk menemukan potensi celah keamanan yang dapat dimanfaatkan oleh penyerang.

2.2. Perancangan

Perancangan proses penetration testing dan implementasi optimalisasi *web alert* pada penelitian ini didasarkan pada analisa yang telah dilakukan dari hasil *vulnerability assessment* menggunakan *acunetix vulnerability web scanner* pada aplikasi berbasis *web* Rekam Medis Elektronik RS Unand.

2.2.1. Perancangan *Penetration Testing*

Penetration testing adalah proses evaluasi keamanan sistem komputer atau jaringan dengan cara aktif mengeksploitasi kelemahan yang ada.

2.2.2. Perancangan Optimalisasi

Perancangan optimalisasi *vulnerability* akan dilakukan berdasarkan hasil analisis *vulnerability assessment* menggunakan *Acunetix Vulnerability Web Scanner*, dengan mempertimbangkan hasil *penetration testing*.

2.3. Pengujian

Skenario pengujian yang dirancang didasarkan pada tahapan perancangan *penetration testing* dan optimalisasi. Tahapan ini meliputi identifikasi kerentanan, eksploitasi kerentanan, dan verifikasi hasil eksploitasi.

3. Hasil dan Pembahasan

Identifikasi kebutuhan untuk perbaikan sistem yang telah dilakukan berdasarkan hasil analisis *vulnerability assessment* menggunakan *acunetix web vulnerability scanner*.

3.1. *Vulnerability Assesment*

3.1.1. *Screening* Sistem Informasi Target

Aplikasi *web* yang menjadi target dalam penelitian ini adalah aplikasi Rekam Medis Elektronik Rumah Sakit Universitas Andalas (RME RS UNAND), aplikasi ini menunjang kegiatan perumahsakitan dalam pencatatan riwayat kunjungan pasien dari pasien mulai masuk kerumah sakit, sampai keluar dari rumah sakit Universitas Andalas.

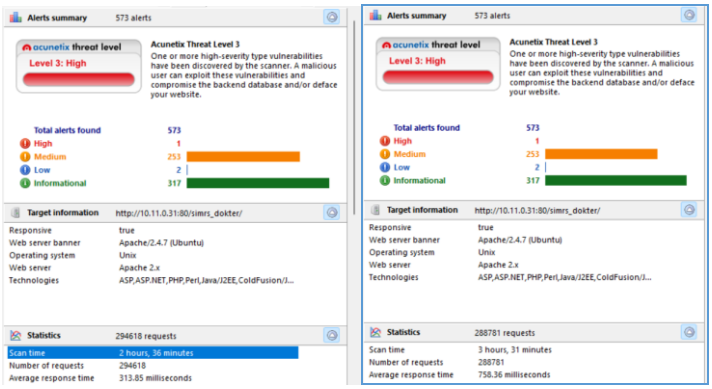
3.1.2. *Scanning Vulnerability*

Aplikasi rekam medis elektronik proses *scanning vulnerability* telah dilakukan sebanyak dua kali dan memakan waktu lebih kurang 3 jam terlihat pada Tabel 1, berikut rincian lama waktu yang dibutuhkan serta jumlah alert yang dihasilkan pada proses *scanning*.

Tabel 1. RME RS UNAND Scanning Iterasi 1

No	Tanggal Pengujian	Durasi Pengujian	Total *Web Alert*
1	01 Juni 2024	3 Jam 31 Menit	573
2	01 Juni 2024	2 Jam 36 Menit	573

Hasil *scanning* pada rekam medis elektronik iterasi 1 *scanning* 1, target yang sudah dilakukan dan memakan waktu 3 jam 31 menit yang tersaji pada Gambar 4.3. Hasil *scanning* mendapatkan informasi bahwa RME RS UNAND memiliki kerentanan *threat level* 3 (*High*) dengan total *alert* yang ditemukan sebanyak 573.



Gambar 2. *Scanning* Iterasi 1 Menggunakan *Acunetix*

3.1.3. Analis Hasil *Vulnerability Assessment*

Informasi pada Tabel 2 memberikan informasi lebih detail jumlah *web alert*, jenis kerentanan yang ditemukan, dan tingkat keparahan setiap potensi serangan. Informasi ini tidak hanya memberikan gambaran yang komprehensif tentang kerentanan yang ada tetapi juga membantu dalam menyusun strategi optimalisasi keamanan yang lebih efektif dan mengetahui detail spesifik tentang kerentanan yang ada.

Tabel 2. Sebaran Web Alert RME RS UNAN Iterasi 1

No	Jenis *Web Alert*	*Level Severity*	Jumlah *Alert*
1	*Git repository found*	*High*	1
2	*Directory listing*	*Medium*	246
3	*HTML form without CSRF protection*	*Medium*	2
4	*Multiple vulnerabilities fixed in PHP versions*	*Medium*	1
5	*Slow HTTP Denial of Service Attack*	*Medium*	1
6	*Source code disclosure*	*Medium*	2
7	*User credentials are sent in clear text*	*Medium*	1
8	*Clickjacking: X-Frame-Options header missing*	*Low*	1
9	*Login page password-guessing attack*	*Low*	1
10	*Broken links*	*Informational*	20
11	*Email address found*	*Informational*	42
12	*Error page web server version disclosure*	*Informational*	1
13	*Password type input with auto-complete enabled*	*Informational*	1
14	*Possible internal IP address disclosure*	*Informational*	247
15	*Possible username or password disclosure*	*Informational*	6
		Total *Web Alert*	573
		Thread Level	3 (*High*)

Hasil dari *vulnerability assessment* selanjutnya akan diproses pada tahap pengujian untuk dilakukan identifikasi lebih lanjut sehubungan dengan detail serangan, jenis *web alert*, serta teknik optimalisasi yang akan dilakukan. Setiap *alert* yang terdeteksi akan dianalisis secara mendalam untuk memahami akar penyebab kerentanan tersebut dan bagaimana penyerang dapat memanfaatkannya. Informasi data yang terdapat pada Tabel 2 selanjutnya dilakukan klasifikasi *web alert* ke dalam kategori *OWASP Top 10 2021*, sehingga *klasifikasi web alert* tersebut tercantum pada Tabel 3.

Tabel 3. Rekapitulasi *Web Alert* Berdasarkan *OWASP Top 10-2021*

No	OWASP *Top 10 Category*	Ada	*Web Alert*	*Level Severity*
1	A01:2021 *Broken Access Control.*	☐	*HTML form without CSRF protection.*	*Medium*
			Possible username or password disclosure.	*Informational*
2	A02:2021 *Cryptographic Failures.*	☐	*User credentials are sent in clear text.*	*Medium*
3	A03:2021 *Injection.*	☐	*Email address found.*	*Informational*
4	A04:2021 *Insecure Design.*	-	Tidak ada.	-
5	A05:2021 *Security Misconfiguration.*	☐	*Directory listing.*	*Medium*
			Source code disclosure.	*Medium*
			Possible internal IP address disclosure.	*Informational*

No	OWASP *Top* 10 *Category*	Ada	*Web Alert*	*Level Severity*
6	A06:2021 *Vulnerable and Outdated Components.*		*Git repository found.*	*High*
			Multiple vulnerabilities fixed in PHP versions.	*Medium*
			Clickjacking: X-Frame-Options header missing.	*Low*
			Error page web server version disclosure.	*Informational*
7	A07:2021 *Identification and Authentication Failures.*		*Slow HTTP Denial of Service Attack.*	*Medium*
			Login page password-guessing attack.	*Low*
8	A08:2021 *Software and Data Integrity Failures.*	-	Tidak ada.	-
9	A09:2021 *Security Logging and Monitoring Failures.*	-	Tidak ada.	-
10	A10:2021 *Server-Side Request Forgery (SSRF).*	-	Tidak ada.	-

3.2. Perancangan

3.2.1. Perancangan Penetration Testing

Proses penetration testing dilakukan terhadap setiap jenis *web alert* yang teridentifikasi pada iterasi pertama dari *vulnerability assessment*. Pendekatan ini membantu dalam mengevaluasi dan memastikan bahwa sistem tidak rentan terhadap serangan seperti *Injection* (*SQL Injection*, *Command Injection*), *Broken Authentication*, *Sensitive Data Exposure*, dan kerentanan lain yang sering dimanfaatkan oleh penyerang. Hasil dari *penetration testing* ini digunakan untuk merumuskan langkah optimalisasi dalam memperbaiki atau menghilangkan potensi celah keamanan yang ditemukan, sehingga meningkatkan tingkat keamanan sistem secara keseluruhan. Tabel 4 mejelaskan proses *penetration testing* yang dilakukan dalam upaya pencegahan serangan berdasarkan kerentanan yang diinformasikan *tools Acunetix*.

Tabel 4. Perancangan Penetration Testing

No	*Web Alert* Terdeteksi	Skenario Pengujian
1	*HTML form without CSRF protection.*	Pengujian mengacu pada hasil eksploitasi *Acunetix*. Formulir yang tidak memiliki CSRF dilakukan eksekusi diluar alamat aplikasi *web*.
2	*User credentials are sent in clear text.*	Pengujian dilakukan dengan cara melihat lalu lintas pengiriman data *post* ataupun get.
3	*Directory listing.*	Akses ecara langsung *web listing* yang diperoleh dari ekploitasi Acunetix.
4	*Source code disclosure.*	Melakukan pengujian dan akses langsung terhadap *source code* yang terekspose secara langsung.
5	*Git repository found.*	Eksploitasi hasil *scanning .git*.
6	*Multiple vulnerabilities fixed in PHP versions.*	Lakukan pemeriksaan terhadap versi PHP yang digunakan.
7	*Slow HTTP Denial of Service Attack.*	Lakukan simulasi serangan *HTTP DoS* pada *web server*.

3.2.2. Perancangan Optimalisasi

Optimalisasi ini dilakukan untuk memperbaiki kelemahan keamanan yang teridentifikasi pada sistem informasi. Perancangan optimalisasi *vulnerability* akan dilakukan berdasarkan hasil analisis *vulnerability assessment* dan *penetration testing*. Proses optimalisasi secara umum diterapkan dengan melakukan perbaikan konfigurasi pada sistem informasi, teknik ini dilakukan dengan mengubah pengaturan sistem informasi untuk memperkuat keamanan. Tabel 5 mendeskripsikan proses yang dilakukan dalam tahapan optimalisasi.

Tabel 5. Perancangan Optimalisasi

No	*Web Alert* Terdeteksi	Skenario Pengujian
1	*HTML form without CSRF protection.*	Konfigurasi *config.php*, menambahkan baris kode "*$config['csrf_protection']=TRUE*" serta penerapan "*helper form_open() dan form_close()"* yang disediakan oleh framework yang *Codeigniter 3*
2	*User credentials are sent in clear text.*	Implementasi protokol *HTTPS/SSL*, proses ini dilakukan dengan menggunakan *Certbot*.
3	*Directory listing.*	Konfigurasi "*directive Options -Indexes*" kedalam file *.htaccess* pada direktori */assets*, */sources*, dan */.git.*
4	*Source code disclosure.*	*Setting permission* kepemilikan diubah menjadi permission 600 dengan perintah "chmod 600" pada direktori yang terdampak.
5	*Git repository found.*	Konfigurasi file *.htaccess* kedalam *directory /.git*, dan selanjutnya menambahkan kode "*RedirectMatch 404 /\.git*" dan juga *"<DirectoryMatch "^/.*/\.git/"></DirectoryMatch>*".
6	*Multiple vulnerabilities fixed in PHP versions.*	Update PHP versi pada server.
7	*Slow HTTP Denial of Service Attack.*	Optimalisasi dengan menginstal *software Fail2ban* dimana software ini melakukan *banned* secara otomatis terhadap *request* yang tidak normal.

3.3. Pengujian

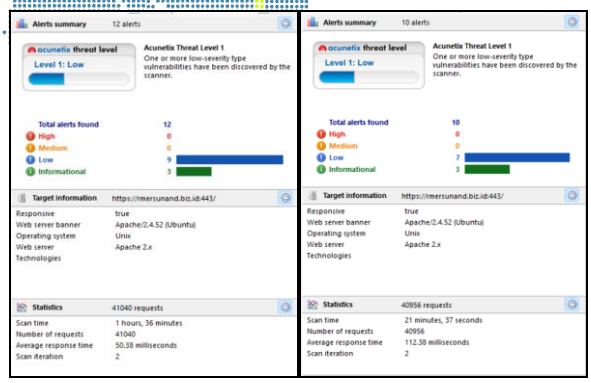
Setelah melakukan perbaikan berdasarkan hasil iterasi pertama, peneliti menjalankan pemindaian ulang untuk mengevaluasi efektivitas tindakan optimalisasi yang telah diambil. Aplikasi RME RS UNAND proses scanning iterasi kedua dilakukan sebanyak 2 kali dengan rincian sebagai tercantum pada Tabel 6.

Tabel 6. RME RS UNAND Scanning Iterasi 2

No	Tanggal Pengujian	Durasi Pengujian	Total *Web Alert*
1	04 Agustus 2024	1 Jam 36 Menit	12
2	04 Agustus 2024	0 Jam 21 Menit	10

Berdasarkan *scanning* iterasi kedua diperoleh hasil bahwa baik pada *scanning* pertama maupun pada *scanning* yang kedua, total *web alert* pada aplikasi RME RS UNAND mengalami penurunan jika dibandingkan dengan total *web alert* pada *scanning* iterasi pertama. Iterasi pertama total *web alert* sejumlah 573, setelah dilakukan optimalisasi scanning pada iterasi kedua total alert menjadi diturun yaitu sebesar 12 dan 15 *alert*.

Scanning iterasi kedua yang telah dilakukan, *threat level* aplikasi RME RS UNAND mengalami perubahan jika dibandingkan dengan hasil *scanning* iterasi pertama. *Scanning* iterasi pertama *threat level* yang diperoleh berada pada *level high*, sedangkan hasil *scanning* setalah implementasi optimalisasi pada aplikasi RME RS UNAND, *threat level* sudah berada pada *level low*, seperti yang tercantum pada Gambar 3.



Gambar 3. *Scanning* Iterasi 2 Menggunakan *Acunetix*

Berdasarkan hasil *scanning* iterasi kedua tersebut, didapatkan informasi bahwa hasil scanning iterasi kedua menampilkan penurunan jumlah *web alert* dengan detail informasi seperti pada Tabel 7

Tabel 7. Sebaran *Web Alert* RME RS UNAND Iterasi 2

No	Jenis *Web Alert*	*Level Severity*	Jumlah *Alert*
1.	*Clickjacking: X-Frame-Options header missing*	*Low*	1
2.	*Cookie without HttpOnly flag set*	*Low*	1
3.	*Cookie without Secure flag set*	*Low*	2
4.	*Login page password-guessing attack*	*Low*	1
5.	*Possible sensitive files*	*Low*	2
6.	*Slow response time*	*Low*	2
7.	*Password type input with auto-complete enabled*	*Informational*	1
8.	*Possible username or password disclosure*	*Informational*	2
		Total *Web Alert*	12
		Thread Level	1 (*Low*)

Setelah melakukan serangkaian optimalisasi kerentanan pada aplikasi RME RS UNAND, pada Tabel 8 berikut ditampilkan perbandingan data *web alert* berkategori *high* dan *medium* sebelum dan setelah diterapkan optimalisasi.

Tabel 8. Perbandingan *Web Alert* Sebelum dan Sesudah Optimalisasi

No.	*Web Alert*	Sebelum Optimalisasi	Setelah Optimalisasi
1.	*Git repository found*	1	0
2.	*Directory listing*	246	0
3.	*HTML form without CSRF protection*	2	0
4.	*Multiple vulnerabilities fixed in PHP versions*	1	0
5.	*Slow HTTP Denial of Service Attack*	1	0
6.	*Source code disclosure*	2	0
7.	*User credentials are sent in clear text*	1	0
	Jumlah Kasus	254	0

Mengacu kepada hasil rekapan optimalisasi yang telah dilakukan sebagaimana yang tersaji pada Tabel 7, maka dapat dikatakan bahwa proses optimalisasi yang dilakukan pada *web alert level high* hingga *medium* yang berhasil diekspose oleh *Acunetix* di aplikasi RME RS UNAND telah dapat dioptimalisasi yang dibuktikan dengan menurunnya *threat level* yang awal nya ber *Level* 3 (*High*) setelah dilakukan optimalisasi dan dilakukan *scanning* iterasi kedua maka hasil yang didapatkan berubah menjadi *Level* 1 (*Low*).

4. Kesimpulan

Upaya optimalisasi keamanan pada aplikasi Rekam Medis Elektronik (RME) RS UNAND terbukti efektif, seperti ditunjukkan oleh penurunan threat level dari Level 3 (high) pada scanning pertama menjadi Level 1 (Low) pada scanning kedua menggunakan Acunetix Vulnerability Web Scanner. Proses ini mencakup evaluasi web alert, review source code, pengaturan konfigurasi, dan implementasi solusi yang berhasil mengatasi berbagai kerentanan, meningkatkan keamanan aplikasi secara signifikan. Penelitian ini menegaskan pentingnya vulnerability assessment dan optimalisasi berkelanjutan untuk melindungi data sensitif dalam sistem RME.

Daftar Pustaka

[1] N. Albalawi, N. Alamrani, R. Aloufi, M. Albalawi, A. Aljaedi, and A. R. Alharbi, "The Reality of Internet Infrastructure and Services Defacement: A Second Look at Characterizing Web-Based Vulnerabilities," *Electron.*, vol. 12, no. 12, 2023, doi: 10.3390/electronics12122664.

[2] F. M. Alotaibi and V. G. Vassilakis, "Toward an SDN-Based Web Application Firewall: Defending against SQL Injection Attacks," *Futur. Internet*, vol. 15, no. 5, pp. 1–15, 2023, doi: 10.3390/fi15050170.

[3] M. S. Aliero, I. Ghani, K. N. Qureshi, and M. F. Rohani, "An algorithm for detecting SQL injection vulnerability using black-box testing," *J. Ambient Intell. Humaniz. Comput.*, vol. 11, no. 1, pp. 249–266, 2020, doi: 10.1007/s12652-019-01235-z.

[4] M. Indushree, M. Kaur, M. Raj, R. Shashidhara, and H. N. Lee, "Cross Channel Scripting and Code Injection Attacks on Web and Cloud-Based Applications: A Comprehensive Review," *Sensors*, vol. 22, no. 5, pp. 1–20, 2022, doi: 10.3390/s22051959.

[5] Menteri Kesehatan RI, *Peraturan Menteri Kesehatan Republik Indonesia No 24 Tahun 2022 Tentang Rekam Medis*, No 24 2023. Menteri Kesehatan RI, 2022. [Online]. Available: https://yankes.kemkes.go.id/unduhan/fileunduhan_1662611251_882318.pdf

[6] Jarupunphol, P. Seatun, S. Buathong, and Wipawan, "Measuring Vulnerability Assessment Tools' Performance on the University Web Application," *Pertanika J. Sci. Technol.*, vol. 31, no. 6, pp. 2973–2993, 2023, doi: 10.47836/pjst.31.6.19.

[7] 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.

[8] Guntoro, C. Loneli, and M. Musfawati, "Analisis Keamanan Web Server Open Journal System (Ojs) Menggunakan Metode Issaf Dan Owasp (Studi Kasus Ojs Universitas Lancang Kuning)," *JIPI (Jurnal Ilm. Penelit. dan Pembelajaran Inform.*, vol. 5, no. 1, p. 45, 2020, doi: 10.29100/jipi.v5i1.1565.

[9] E. Listartha, G. Arna, J. Saskara, D. Gede, and S. Santyadiputra, "Vulnerability Testing and Security Penetration on Prodi XYZ Thesis Management Web Applications," *Sci. Comput. Sci. Informatics J.*, vol. 4, no. 2, pp. 1–14, 2021.

[10] F. Al Fajar, "Analisis Keamanan Aplikasi Web Prodi Teknik Informatika Uika Menggunakan Acunetix Web Vulnerability," *Inova-Tif*, vol. 3, no. 2, p. 110, 2020, doi: 10.32832/inova-tif.v3i2.4127.

[11] F. Kristianto, S. Rahman, and S. Bahri, "Analisis Kerentanan Pada Website Servio Menggunakan Acunetix Web Vulnerability," *Jtriste*, vol. 9, no. 1, pp. 46–55, 2022, doi: 10.55645/jtriste.v9i1.363.

[12] S. Sandy and H. H. Solihin, "Audit Keamanan dan Manajemen Risiko pada e-Learning Universitas Sangga Buana," *J. Manaj. Inform.*, vol. 11, no. 1, pp. 1–14, 2021, doi: 10.34010/jamika.v11i1.3641.