jurnalteknik.unisla.ac.id/index.php/informatika
JOUTICA Volume 8 No.1 Tahun 2023

ISSN: 2503-071X
E-ISSN: 2621-511X

ANALISIS KEAMANAN WEBSITE BPJS KESEHATAN MENGGUNAKAN METODE *VULNERABILITY ASESEMENT*

Yulia Taryana[1], Nono Heryana[2]
[1,2] *Informatika,Ilmu Komputer,Universitas Singaperbangsa Karawang*
Jl.H.S. Ronggowaluyo Teluk Jambe Timur – Karawang 41361
Telp. 0267641177
E-mail: [1]yuliataryana.yt@gmail.com, [2]nono@unsika.ac.id

ABSTRAK
Keamanan system saat ini sangat diperlukan karena sedang maraknya hacker yang berhasil membobol situs-situs yang memiliki jutaan data. Salah satu permasalahan yang sempat beredar dimedia sosial bahwa terjadi kebocoran data pada BPJS Kesehatan. BPJS Kesehatan adalah badan hukum yang dibentuk untuk menyelenggarakan program jaminan kesehatan. Tujuan penelitian ini adalah untuk menganalisa keamanan system website BPJS Kesehatan menggunakan metode Vulnerability Asesement dengan tool OWASP ZAP. OWASP ZAP (Zed Attack Proxy) merupakan sebuah aplikasi yang digunakan untuk pengujian penetrasi dalam menemukan celah keamanan pada suatu aplikasi website. Tahapan metode penelitian yang digunakan pada penelitian ini adalah mengidentifikasi masalah, melakukan studi literature, persiapan penetrasi, mengumpulkan informasi aplikasi, mengidentifikasi kerentanan aplikasi, dan mendapat hasil tes penetrasi. Hasil dari penelitian ini adalah Website BPJS Kesehatan tergolong aman untuk digunakan.

Kata Kunci: *BPJS Kesehatan, Keamanan Sistem, OWASP ZAP.*

ABSTRACT
The current security system is very much needed because there are many hackers who have succeeded in breaking into sites that have millions of data. One of the problems that had circulated on social media was that there was a data leak at BPJS Health. BPJS Health is a legal entity formed to administer the health insurance program. The purpose of this study was to analyze the BPJS Health website security system using the Vulnerability Assessment method with the OWASP ZAP tool. OWASP ZAP (Zed Attack Proxy) is an application that is used for penetration testing in finding security holes in a website application. The stages of the research method used in this study are identifying problems, conducting literature studies, preparing for penetration, collecting application information, finding application vulnerabilities, and getting penetration test results. The results of this study are the BPJS Health website is classified as safe to use.

Keywords: *BPJS Health, Security System, OWASP ZAP.*

1. PENDAHULUAN

Saat ini perkembangan teknologi mengalami perubahan yang sangat pesat. Hal ini terlihat dari semakin banyaknya pengguna website, baik untuk kepentingan institusi, pendidikan, organisasi, maupun pribadi. Keamanan adalah aspek penting dari segalanya.

Badan Penyelenggara Jaminan Sosial (BPJS) merupaka badan aturan publik yang dibentuk berdasarkan Undang-Undang Nomor 24 mengenai Penyelenggaraan Jaminan Sosial Tahun 2011 yang ditujukan pada setiap peserta atau anggota keluarga. BPJS Kesehatan merupakan badan pengawas yang menyelenggarakan program jaminan kesehatan. BPJS Kesehatan didirikan pada 1 Januari 2014. BPJS Kesehatan awalnya bernama PTAskes dan diubah menjadi BPJS Kesehatan.

Meski jumlah pengguna BPJS Kesehatan semakin hari semakin meningkat, namun potensi serangan peretasan tetap ada. Ini menghancurkan dan bahkan dapat dibajak oleh peretas berdasarkan buruknya kinerja kesalahan akses BPJS Kesehatan. Keamanan adalah solusi untuk sistem kesehatan BPJS. Masalah keamanan berkaitan dengan kerahasiaan, integritas, dan ketersediaan layanan data pada sistem yang diimplementasikan. Pihak BPJS Kesehatan juga mengakui adanya potensi penyusupan sehingga data dari 279 juta penduduk Indonesia bisa saja bocor dan dijual di dunia maya. Kesehatan, presiden BPJS, mengatakan sistem keamanan yang digunakan adalah standar ISO 27001 dan dikatakan berlapis-lapis, tetapi pelanggaran data masih dapat dikompromikan.

Keamanan informasi dalam suatu web menjadi amat berarti. Keamanan data suatu web ialah salah satu prioritas yang amat penting untuk seorang website development. Bila seorang melalaikan keamanan itu hingga seseorang hacker bisa mengutip informasi berarti serta apalagi

ISSN: 2503-071X
E-ISSN: 2621-511X

mangacak- acak bentuk website tersebut (Suparyanto dan Rosad (2015, 2020).

Masalah keamanan memerlukan penerapan metode yang bisa mengklaim keamanan data, transaksi, dan komunikasi. Kurangnya keamanan sistem akan berdampak buruk. Peretas dapat dengan sangat mudah mengambil alih sistem yang dibangun. Hal ini dapat menyebabkan masalah pada data pribadi dan data yang sangat penting bagi perusahaan atau organisasi, data tersebut tidak boleh diketahui orang lain, tetapi dapat diakses oleh hacker. Hacker adalah orang yang memiliki kemampuan tinggi di bidang teknologi informasi. Perlu tindakan cepat untuk melindungi BPJS Kesehatan.

Oleh karena itu, dalam menanggapi permasalahan tersebut, penulis memberikan solusi yaitu menganalisis keamanan website dengan menggunakan tool Open Web Application Security Project (OWASP). Analisis ini mengidentifikasi berbagai kerentanan yang memungkinkan terdapat penyerangan pada website BPJS Kesehatan.

1.1. BPJS Kesehatan

BPJS Kesehatan merupakan badan pengawas yang menyelenggarakan program jaminan kesehatan. BPJS Kesehatan didirikan pada 1 Januari 2014. BPJS Kesehatan awalnya bernama PTAskes dan diubah menjadi BPJS Kesehatan. Badan Penyelenggara Jaminan Sosial atau BPJS merupakan lembaga yang dibentuk untuk menyelenggarakan program jaminan sosial di Indonesia. BPJS Kesehatan merupakan jaminan kesehatan nasional yang memberikan manfaat kesehatan untuk seluruh masyarakat dengan premi terjangkau (Wulansari et al., 2015).

1.2. Website Atau Situs

Website adalah kumpulan halaman – halaman yang saling berhubungan baik statis maupun dinamis yang digunakan untuk menampilkan informasi tekstual, gambar, gambar diam atau bergerak, animasi, suara, atau kombinasinya yang membentuk rangkaian bangunan, masing-masing dengan jaringan web (Mulyanto & Haryanti, 2021).

Situs web adalah cara untuk menampilkan diri Anda di internet. Situs web adalah tempat di Internet yang dapat diakses oleh semua orang di dunia (Dahlan et al., 2014). Fitur paling dasar dari sebuah situs web adalah memiliki informasi / konten statis, yaitu hampir tidak berubah (Yum Thurfah Afifa Rosaliah, 2021).

1.3. *Vulnerability Asesement*

Vulnerability Asesement merupakan kerangka kerja konseptual komprehensif yang mencakup definisi kerentanan untuk mengukur risiko. Bergantung pada tujuan penggunaan hasil penilaian, ini dapat berkisar dari niat untuk menginformasikan politik internasional hingga tindakan di tingkat masyarakat. *Vulnerability Assessment* digunakan untuk melakukan pengujian pada point-point yang berpotensial masuknya serangan. Selain itu juga mengidentifikasi masa berlakunya versi sebuah software, mengidentifikasi port-port yang terbuka, dan dapat juga mengidentifikasi aplikasi apa saja yang sedang berjalan. *Vulnerability Assessment* digunakan untuk mendeteksi kelemahan dalam jaringan (Mira Orisa & Ardita, 2021).

Analisis kerentanan adalah identifikasi kelemahan pada aplikasi, sistem operasi, dan infrastruktur jaringan. Analisis kelemahan tidak mengidentifikasi celah atau kelemahan dalam sistem. Pada saat yang sama, kerentanan adalah kelemahan dalam desain sistem, implementasi sistem, atau kontrol operasional yang dapat dimanfaatkan untuk melanggar kebijakan keamanan sistem. Penilaian kerentanan berfokus pada menemukan berbagai kerentanan publik pada semua sistem computer di jaringan target (Mulyanto & Haryanti, 2021).

Menurut (Digdo, 2017)sebagian orang selalu mengaitkan antara metode Penetration Testing dengan metode Vulnerability Asesment. Adapun perbedaan antara Metode Penetration Testing dengan Metode Vulnerability Asesment, ditunjukkan pada tabel berikut (Mulyanto & Haryanti, 2021).

Tabel 1. Perbedaan Metode Penetration Testing dan Vunerability Asessment

No	*Penetration Testing*	*Vulnerability Asesment*
1	Identifikasi Beberapa Celah Keamanan	Identifikasi semua Celah Keamanan
2	Pendekatan risiko IT	Pendekatan Bisnis dan Resiko IT
3	Pembuktian Secara Teknis	Pendekatan secara Teori

1.4. Keamanan Website

Keamanan Website adalah suatu cabang teknologi yang dikenal dengan nama keamanan informasi yang diterapkan pada website. Sasaran keamanan website antara lain adalah sebagai perlindungan terhadap informasi/data (Elu, 2017).

Keamanan situs web adalah salah satu tugas pertama administrator situs web atau pengguna. Sebagian besar pengguna hanya peduli dengan desain tampilan dan konten yang menarik pengunjung sebanyak mungkin. Jika pemroses atau pengguna mengabaikan keamanan situs web, pengguna akan dirugikan karena seseorang dapat mengambil data penting di situs web, atau bahkan merusak tampilan situs web. Hal terpenting dari keamanan situs adalah untuk melindungi komputer, aplikasi, dan jaringan, tujuannya adalah

ISSN: 2503-071X
E-ISSN: 2621-511X

untuk melindungi informasi yang terkandung di dalamnya (Mulyanto & Haryanti, 2021).

1.5. OWASP

Open Web Application Security Project (OWASP) merupakan kerangka kerja sumber terbuka yang berfokus pada peningkatan keamanan perangkat lunak aplikasi. OWASP adalah organisasi yang dirancang untuk menemukan kerentanan dalam aplikasi web. OWASP tidak terafiliasi dengan perusahaan teknologi manapun, namun tetap mendukung penggunaan teknologi keamanan komersial. OWASP menghasilkan beragam jenis proyek dengan cara kolaborasi yang terbuka, di antaranya Web Security Testing Guide (WSTG), OWASP Top Ten, WSTG Checklist v4.2, dan lain sebagainya (Judul et al., 2022).

Terdapat 11 yang dapat dilakukan untuk mengevaluasi dan menguji keamanan situs web berupa pengumpulan informasi, manajemen konfigurasi, transmisi aman, otentikasi, manajemen sesi, otorisasi, enkripsi, dan validasi data berdasarkan standar yang diterbitkan oleh OWASP. Ada langkah-langkahnya. , Penolakan layanan, penanganan kesalahan. Gambar 1 menunjukkan perbandingan OWASP 2013 versi 10 dan 2017 OWASP 10 (Guntoro et al., 2020).

Tabel 2. Perbandingan OWASP

OWASP TOP 10 - 2013	OWASP TOP 10 – 2017
A1 – Injeksi	A1 – Injeksi
A2 – Otentikasi dan manajemen sesi yang buruk	A2 – Otentikasi yang buruk
A3 – Cross-Site Scripting (XSS)	A3 – Data sensitif yang terekspos
A4 – Referensi obyek langsung tidak aman	A4 – XML External Entities (XXE)
A5 – Kesalahan konfigurasi keamanan	A5 – Akses kontrol yang buruk
A6 – Data sensitif yang terekspos	A6 – Kesalahan konfigurasi keamanan
A7 – Kehilangan fungsi kontrol tingkatan akses	A7 – Cross-Site Scripting (XSS)
A8 – Cross-Site Request Forgery (CSRF)	A8 – Deserialisasi yang tifak aman
A9 – Menggunakan komponen rentan yang diketahui	A9 – Menggunakan komponen rentan yang diketahui
A10 – Redireksi dan Forward yang tidak tervalidasi	A10 – Pencatatan dan pemantauan yang tidak cukup

1.6. OWASP-ZAP

OWASP ZAP (Zed Attack Proxy) adalah aplikasi yang digunakan untuk pengujian penetrasi untuk menemukan celah keamanan pada aplikasi website. ZAP secara otomatis menyediakan pemindai (Guntoro et al., 2020).

ZAP juga dikenal sebagai proyek unggulan OWASP. ZAP juga menyediakan seperangkat program yang memungkinkan Anda untuk mendeteksi Celah Keamanan secara manual selain pendeteksi otomatis. ZAP juga dapat menghasilkan laporan dalam format HTML dan XML (Ula, 2019).

1.7. Metode

Metode pada penelitian ini terdapat beberapa tahap, yaitu:

(a) Mengidentifikasi Masalah

Pada tahap pertama ini adalah mengidentifikasi masalah, dan masalah yang didapat ialah telah terjadinya kebocoran data BPJS Kesehatan. Berdasarkan masalah yang didapat tersebut menandakan bahwa website BPJS Kesehatan terdapat kerentanan. Masalah tersebut terlansir pada KOMPAS.com dan sempat viral dimedia sosial twitter.

Gambar 1. Kebocoran data BPJS - KOMPAS.com

Gambar 2. Kebocoran data BPJS – Twitter

ISSN: 2503-071X
E-ISSN: 2621-511X

(b) Melakukan Studi Literatur

Tahap kedua ini yaitu studi literature dimana dilakukan pengumpulan informasi dari permasalahan yang ada. Informasi didapatkan dengan teknik pengumpulan data dari beberapa sumber seperti jurnal ilmiah,buku,artikel, catatan literatur dan penelitian-penelitian yang sudah ada sebelumya.

(c) Persiapan Penetrasi

Sebelum dilakukan penetrasi alangkah baiknya melakukan persiapan terlebih dahulu dengan menyiapkan perangkat lunak serta aplikasi yang dibutuhkan. Dan memasang OWASP ZAP versi 2.11.0.

(d) Mengumpulkan Informasi Aplikasi

Dilakukannya pengumpulan informasi website BPJS Kesehatan menggunakan ZAP. Dan informasi yang didapat yaitu berupa URL, http, https, Get dan lain sebagainya. Kemudian hasilnya tersimpan pada session OWASP ZAP.

(e) Mengidentifikasi Kerentanan Aplikasi

Aplikasi yang digunakan yaitu OWASP ZAP versi 2.11.0, yang dimana akan dilakukan pemindaian website BPJS Kesehatan menggunakan OWASP ZAP. Kemudian semua informasi yang tersimpan akan dipindai dan hasilnya akan tersimpan juga pada session OWASP ZAP.

(f) Hasil Tes Penetrasi

Pada tahapan terakhir yaitu menganalisis kerentanan yang didapat pada website BPJS Kesehatan.

2. PEMBAHASAN

2.1 Planning

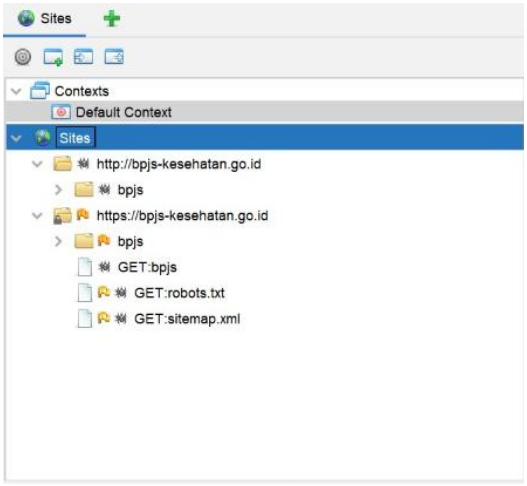
Pada tahap ini dipersiapkan kebutuhan yang dibutuhkan untuk melakukan pengujian celah keamanan pada website BPJS Kesehatan. Adapun yang dibutuhkan sebagai berikut.

Tabel 3. Peralatan yang dibutuhkan

Peralatan	*Spesifikasi*
Laptop	OS : Windows 10 Home 64-bit Processor : Intel(R) Core(TM) i7-8565U Memory : 8192MB RAM
OWASP ZAP	OWASP ZAP versi 2.11.0
Koneksi Internet	Up to 20 Mbps
Web Browser	Chrome

2.2 Information Gathering

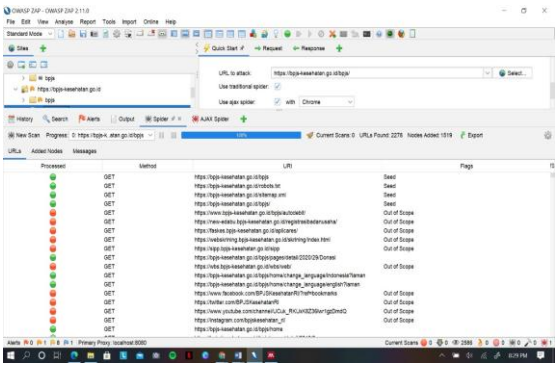
Di tahap ini peneliti mengumpulkan informasi yang terdapat pada website BPJS Kesehatan. Situs yang dipindai akan tersimpan langsung pada session OWASP ZAP. Pemindaian pada OWASP ZAP tidak semuanya kelemahan, tapi juga dapat mendeteksi ada atau tidaknya file tersebut.



Gambar 3. Sites BPJS Kesehatan

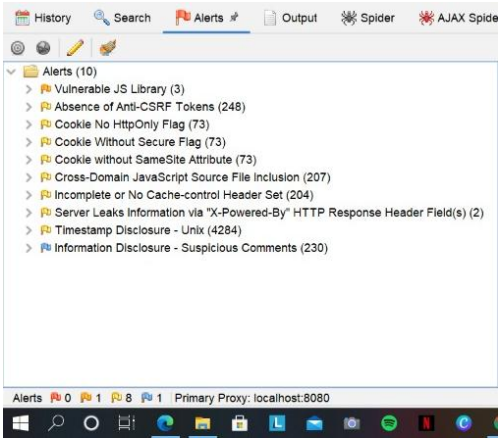
2.3 Operasi Pengujian Kerentanan

Proses pengujian atau pendeteksian kerentanan pada website BPJS Kesehatan dilakukan seperti yang telah dijelaskan sebelumnya yaitu menggunakan tool OWASP ZAP.



Gambar 4. Proses Pemindaian Website BPJS Kesehatan

ISSN: 2503-071X
E-ISSN: 2621-511X



Gambar 5. Kerentanan pada Web BPJS Kesehatan

Proses awal pengujian dilakukan pemindaian pada web BPJS Kesehatan, proses pemindaian ini berlangsung untuk mendapatkan kerentanan pada web BPJS Kesehatan seperti pada Gambar 4. Pada Gambar 5 didapatkan hail kerentanan pada web BPJS Kesehatan dengan risk level yang berbeda-beda.

Tabel 4. Kerentanan pada website BPJS Kesehatan

Type	*URL*	*Risk*	*Confidence*
Vulnerable JS Library	https://bpjs-kesehatan.go.id/bpjs/themes/admin/js/jquery-1.7.2.js	Medium	Medium
Absence of Anti-CSRF Tokens	https://bpjs-kesehatan.go.id/bpjs/	Low	Medium
Cookie No HttpOnly Flag	https://bpjs-kesehatan.go.id/bpjs/	Low	Medium
Cookie Without Secure Flag	https://bpjs-kesehatan.go.id/bpjs/	Low	Medium
Cookie without SameSite Attribute	https://bpjs-kesehatan.go.id/bpjs/	Low	Medium
Cross-Domain JavaScript Source File Inclusion	https://bpjs-kesehatan.go.id/bpjs/	Low	Medium
Incomplete or No Cache-control Header Set	https://bpjs-kesehatan.go.id/bpjs/	Low	Medium
Secure Pages Include Mixed Content	https://bpjs-kesehatan.go.id/bpjs/multimedia/index/114	Low	Medium
Server Leaks Information via "X-Powered-By" HTTP Response Header Field(s)	https://bpjs-kesehatan.go.id/robots.txt	Low	Medium
Timestamp Disclosure - Unix	https://bpjs-kesehatan.go.id/bpjs/	Low	Low
Information Disclosure - Suspicious Comments	https://bpjs-kesehatan.go.id/bpjs/	Informational	Low

Pada penelitian ini ditemukan sebelas kerentanan pada web BPJS Kesehatan, informasinya sebagai berikut. Terdapat 9 kasus yang level risk rendah (low), 1 kasus dengan level risk menegah (medium), dan 1 kasus lainnya hanya bersifat informational seperti pada Tabel 3.

3.4 Deskripsi Kerentanan dan Solusi Penanganan Kerentanan

Berdasarkan tool OWASP ZAP untuk menangani celah kerentanan yang ditemukan web BPJS Kesehatan yang sudah dipaparkan sebelumnya, maka rekomendasi solusi penanganan kerentanan pada web BPJS Kesehatan sebagai berikut.

1) Jquery perpustakaan yang diidentifikasi, versi 1.7.2 rentan. Solusinya harap tingkatkan ke versi jquery terbaru.
2) Tidak ada token Anti CSRF yang ditemukan dalam formulir pengiriman HTML. Pemalsuan permintaan lintas situs adalah serangan yang memaksa korban untuk mengirim permintaan HTTP ke target tanpa korban mengetahui atau berniat menjadi korban. Penyebab utamanya adalah fungsionalitas atau kemampuan aplikasi yang menggunakan URL/tindakan formulir yang dapat diprediksi dengan cara yang berulang. Sifat serangannya adalah CSRF mengeksploitasi kepercayaan yang diberikan situs web kepada penggunanya. Sebaliknya, skrip lintas situs (XSS) mengeksploitasi kepercayaan yang dimiliki pengguna untuk situs web.

   Serangan CSRF efektif dalam beberapa situasi, termasuk:

- Korban memiliki sesi aktif di situs target.
- Korban diautentikasi melalui otentikasi HTTP di situs target.
- Korban berada di jaringan lokal yang sama dengan situs target.

  CSRF terutama digunakan untuk melakukan tindakan terhadap situs target menggunakan hak korban, tetapi teknik terbaru telah ditemukan untuk mengungkapkan informasi dengan mendapatkan akses ke respons. Risiko pengungkapan informasi meningkat secara dramatis ketika situs target rentan terhadap XSS, karena XSS dapat digunakan sebagai platform untuk CSRF, memungkinkan serangan untuk beroperasi dalam batas-batas kebijakan asal yang sama.

ISSN: 2503-071X
E-ISSN: 2621-511X

Solusinya adalah Fase: Arsitektur dan Desain Gunakan perpustakaan atau kerangka kerja yang diperiksa yang tidak memungkinkan kelemahan ini terjadi atau menyediakan konstruksi yang membuat kelemahan ini lebih mudah untuk dihindari.

3) Cookie disetel tanpa tanda HttpOnly. yang berarti bahwa cookie dapat diekspos ke JavaScript. Jika halaman ini dapat menjalankan skrip berbahaya, maka halaman ini dapat mengakses cookie dan mengirimkannya ke situs web lain. Jika ini cookie sesi, pembajakan sesi dimungkinkan dapat terjadi.
4) Cookie telah disetel tanpa tanda aman, yang berarti bahwa cookie dapat diakses melalui koneksi yang tidak terenkripsi. Solusinya adalah Kapan pun cookie berisi informasi sensitif atau merupakan token sesi, maka cookie harus selalu diteruskan menggunakan saluran terenkripsi. Pastikan bahwa tanda aman diatur untuk cookie yang berisi informasi sensitif tersebut.
5) Cookie telah disetel tanpa atribut SameSite, yang berarti cookie dapat dikirim sebagai hasil dari permintaan 'lintas situs'. Atribut SameSite adalah tindakan balasan yang efektif untuk pemalsuan permintaan lintas situs, penyertaan skrip lintas situs, dan serangan waktu. Solusinya adalah Pastikan atribut SameSite diatur ke 'lax' atau idealnya 'strict' untuk semua cookie.
6) Halaman berisi satu atau beberapa file skrip dari domain pihak ketiga. Solusinya adalah Pastikan file sumber JavaScript dimuat hanya dari sumber tepercaya, dan sumber tidak dapat dikontrol oleh pengguna akhir aplikasi.
7) Header kontrol cache belum disetel dengan benar atau tidak ada, memungkinkan browser dan proxy untuk menyimpan konten dalam cache. Solusinya adalah ika memungkinkan, pastikan header HTTP kontrol-cache disetel dengan no-cache, no-store, must-revalidate.
8) Halaman tersebut mencakup konten campuran, yaitu konten yang diakses melalui HTTP, bukan HTTPS. Solusinya adalah Halaman yang tersedia melalui SSL/TLS harus sepenuhnya terdiri dari konten yang dikirimkan melalui SSL/TLS. Halaman tidak boleh berisi konten apa pun yang dikirimkan melalui HTTP yang tidak terenkripsi. Ini termasuk konten dari situs pihak ketiga.
9) Server web/aplikasi membocorkan informasi melalui satu atau lebih header respons HTTP "X-Powered-By". Akses ke informasi tersebut dapat memfasilitasi penyerang untuk mengidentifikasi kerangka kerja/komponen lain yang diandalkan oleh aplikasi web Anda dan kerentanan yang mungkin dialami oleh komponen tersebut. Solusinya adalah Pastikan server web Anda, server aplikasi, penyeimbang beban, dll. dikonfigurasi untuk menekan header "X-Powered-By".
10) Stempel waktu diungkapkan oleh aplikasi/server web – Unix. Solusinya adalah Verifikasi secara manual bahwa data stempel waktu tidak sensitif dan tidak dapat digabungkan untuk mengungkapkan pola yang dapat dieksploitasi.
11) Tanggapan tersebut tampaknya berisi komentar mencurigakan yang dapat membantu penyerang. Catatan: Kecocokan yang dibuat dalam blok skrip atau file bertentangan dengan seluruh konten, bukan hanya komentar. Solusinya adalah Hapus semua komentar yang mengembalikan informasi yang dapat membantu penyerang dan perbaiki masalah mendasar yang mereka rujuk.

3. KESIMPULAN

Hasil yang diperoleh dari pengujian ini, ditemukan sebelas celah kerentanan pada website BPJS Kesehatan yang didapatkan melalui pemindaian tool OWASP ZAP versi 2.11.0. Dan juga diperoleh solusi-solusi untuk menangani kerentanan pada website BPJS Kesehatan tersebut.

Dengan demikian, penelitian dilakukan uji penetrasi menggunakan OWASP ZAP versi 2.11.0 yang bertujuan untuk menguji keamanan website BPJS Kesehatan. Dari penelitian yang sudah dilakukan, dapat disimpulkan bahwa Website BPJS Kesehatan tergolong aman untuk digunakan dan OWASP ZAP sangat efektif bila digunakan untuk mendeteksi kerentanan.

PUSTAKA

Dahlan, M., Latubessy, A., Nurkamid, M., & Anggraini, L. (2014). Pengujian Dan Analisa Keamanan Website Terhadap Serangan Sql Injection (Studi Kasus : Website Umk). *Jurnal Sains Dan Teknologi*, *7*(1), 13–19.

Elu, A. M. (2017). Rancang Bangun Aplikasi Pendeteksian Vulnerability Structured Query Language (Sql) Injection Untuk Keamanan Website. *Respati*, *8*(22), 111–124. https://doi.org/10.35842/jtir.v8i22.53

Guntoro, G., Costaner, L., & Musfawati, M. (2020). Analisis Keamanan Web Server Open Journal System (Ojs) Menggunakan Metode Issaf Dan Owasp (Studi Kasus Ojs Universitas Lancang Kuning). *JIPI (Jurnal Ilmiah Penelitian Dan Pembelajaran Informatika)*, *5*(1), 45. https://doi.org/10.29100/jipi.v5i1.1565

Judul, H., Industri, F. T., & Indonesia, U. I. (2022). *PENGUJIAN KEAMANAN SISTEM*

ISSN: 2503-071X
E-ISSN: 2621-511X

INFORMASI BERBASIS WEB BERDASARKAN FRAMEWORK OWASP WSTG v4 . 2 (STUDI KASUS : SISTEM SEKAWAN v1 UNIVERSITAS ISLAM INDONESIA) PENGUJIAN KEAMANAN SISTEM INFORMASI BERBASIS WEB BERDASARKAN FRAMEWORK OWASP WSTG v4 . 2 (STUDI KASUS : SISTEM SEKAWAN v1 UNIVERSITAS ISLAM INDONESIA). *2*.

Mira Orisa, & Ardita, M. (2021). Vulnerability Assesment Untuk Meningkatkan Kualitas Kemanan Web. *Jurnal Mnemonic*, *4*(1), 16–19. https://doi.org/10.36040/mnemonic.v4i1.3213

Mulyanto, Y., & Haryanti, E. (2021). Sumbawa Menggunakan Metode Vulnerability Asesement. *Jinteks*, *3*(3), 394–400. https://smanika-sumbawabesar.sch.id.

Pirsa, N., & Sumijan. (2020). Meningkatkan Keamanan Sistem Informasi Puskesmas Terpadu dengan Metode Grey-Box Penetration Test Menggunakan Computer Assisted Audit Techniques. *Jurnal Informasi Dan Teknologi*, *2*, 4–9. https://doi.org/10.37034/jidt.v2i4.79

Riadi, I., Yudhana, A., & W, Y. (2020). Analisis Keamanan Website Open Journal System Menggunakan Metode Vulnerability Assessment. *Jurnal Teknologi Informasi Dan Ilmu Komputer*, *7*(4), 853. https://doi.org/10.25126/jtiik.2020701928

Sama'i, S., Yuswadi, H., Toha, A., & Sutomo, S. (2018). Pelayanan Kesehatan Peserta Bpjs Kesehatan. *Politico*, *18*(1), 1–23. https://doi.org/10.32528/politico.v18i1.1650

Suparyanto dan Rosad (2015. (2020). 済無No Title No Title No Title. *Suparyanto Dan Rosad (2015*, *5*(3), 248–253.

Ula, M. (2019). Evaluasi Kinerja Software Web Penetration Testing. *TECHSI - Jurnal Teknik Informatika*, *11*(3), 336. https://doi.org/10.29103/techsi.v11i3.1996

Wulansari, I., Suprayogi, A., & Nugraha, A. (2015). Pembuatan Aplikasi Sebaran Lokasi Fasilitas Kesehatan Penerima Bpjs Kesehatan Di Kota Semarang Berbasis Android. *Jurnal Geodesi Undip*, *4*(4), 240–247.

Yum Thurfah Afifa Rosaliah, J. J. B. H. (2021). Pengujian Celah Keamanan Website Menggunakan Teknik Penetration Testing dan Metode OWASP TOP 10 pada Website SIM. *Senamika*, *2*(September), 752–761.