# Validasi keamanan infrastruktur web: analisis jaringan dan kekuatan enkripsi SSL/TLS instansi A

**Moch Wahyu Sampurno Utomo, Henni Endah Wahanani*, Achmad Junaidi**

Program Studi Informatika, Fakultas Ilmu Komputer, UPN "Veteran" Jawa Timur
*Email: 22081010046@student.upnjatim.ac.id; henniendah.if@upnjatim.ac.id*;
achmadjunaidi.if@upnjatim.ac.id

## Abstrak
Keamanan data organisasi sangat bergantung pada perlindungan aplikasi web terhadap berbagai ancaman siber. Penelitian ini melaksanakan *Vulnerability Assessment* (VA) pada platform web milik Instansi A melalui pendekatan non-destruktif. Metodologi yang digunakan mengombinasikan pemindaian otomatis menggunakan OWASP ZAP dan Nmap dengan analisis manual mencakup konfigurasi SSL/TLS, *header* HTTP, serta informasi domain via perangkat *curl*, *openssl*, dan *whois*. Temuan penelitian mengidentifikasi sejumlah celah, di antaranya absennya *security header*, atribut *cookie* yang kurang protektif, serta penggunaan konfigurasi server versi lama. Meski enkripsi SSL telah memenuhi standar modern, optimalisasi pada sisi keamanan HTTP masih sangat diperlukan. Studi ini menyimpulkan bahwa integrasi instrumen otomatis dan evaluasi manual efektif dalam menghasilkan penilaian keamanan yang komprehensif sesuai standar OWASP.

**Kata Kunci:** OWASP ZAP; HTTP Header; Nmap; SSL/TLS; vulnerability assessment; web security

# *Web infrastructure security validation: network analysis and SSL/TLS encryption strength of agency a*

## *Abstract*
*Organizational data security relies heavily on protecting web applications against various cyber threats. This study conducted a Vulnerability Assessment (VA) on Agency A's web platform using a non-destructive approach. The methodology used combined automated scanning using OWASP ZAP and Nmap with manual analysis covering SSL/TLS configuration, HTTP headers, and domain information via curl, openssl, and whois. The study's findings identified several gaps, including the absence of security headers, less protective cookie attributes, and the use of outdated server configurations. Although SSL encryption meets modern standards, optimization of HTTP security is still essential. The study concluded that the integration of automated tools and manual evaluations was effective in producing a comprehensive security assessment that meets OWASP standards.*

***Keywords:*** *Nmap; HTTP Header; OWASP ZAP; SSL/TLS; vulnerability assessment; web security*

## 1. Pendahuluan

Pesatnya transformasi digital telah memposisikan aplikasi web sebagai infrastruktur krusial dalam penyediaan layanan publik dan operasional bisnis, baik di sektor pemerintahan maupun swasta. Kendati demikian, ketergantungan yang tinggi terhadap teknologi web berbanding lurus dengan meningkatnya risiko keamanan informasi. Ancaman siber yang kian kompleks seperti eksploitasi *injection*, *cross-site scripting* (XSS), pembajakan sesi, hingga kesalahan konfigurasi keamanan menjadi tantangan nyata yang dapat memicu kebocoran data sensitif serta degradasi reputasi organisasi (World Economic Forum, 2024).

Sejumlah studi literatur mengonfirmasi bahwa kerentanan pada aplikasi web merupakan faktor utama dalam berbagai insiden keamanan. Menurut Oppenheim (2020), penerapan *Vulnerability Assessment* (VA) secara periodik sangat esensial sebagai langkah preventif untuk memitigasi risiko sebelum serangan terjadi. Penggunaan instrumen seperti OWASP ZAP telah terbukti efektif dalam mendeteksi kelemahan pada lapisan aplikasi, khususnya terkait validasi input dan manajemen sesi (OWASP Foundation, 2021). Di sisi lain, penggunaan Nmap memberikan kontribusi penting dalam pemetaan aset jaringan dan identifikasi layanan terbuka yang berpotensi menjadi pintu masuk serangan (Lyon, 2021).

Namun, terdapat celah dalam penelitian-penelitian terdahulu yang cenderung hanya mengandalkan pemindaian otomatis tanpa melakukan audit mendalam pada aspek enkripsi *transport-layer*. Padahal, protokol SSL/TLS yang kokoh adalah fondasi utama dalam menjamin keamanan data saat proses transmisi (Rescorla, 2022). Selain itu, ketergantungan penuh pada automasi sering kali menyisakan masalah *false positive* yang tidak terverifikasi. Kesenjangan inilah yang melatarbelakangi perlunya sebuah pendekatan evaluasi yang lebih integratif, menggabungkan alat otomatis dengan validasi teknis manual.

Berangkat dari urgensi tersebut, penelitian ini bertujuan untuk mengaudit postur keamanan web pada Instansi A dengan mengkombinasikan berbagai perangkat teknis. OWASP ZAP dan Nmap dioptimalkan untuk pemindaian sistemik, sementara analisis mendalam terhadap sertifikat digital, kekuatan *cipher*, dan *security headers* dilakukan secara manual menggunakan OpenSSL serta *curl*. Melalui metodologi yang bersifat non-destruktif, studi ini diharapkan tidak hanya mampu mengidentifikasi kerentanan secara akurat, tetapi juga memberikan model asesmen keamanan yang komprehensif dan dapat diadaptasi oleh instansi lain guna memenuhi standar keamanan global berbasis kerangka kerja OWASP.

## 2. Metode

Penelitian ini menggunakan pendekatan Vulnerability Assessment (VA) dengan metode non-destruktif, yaitu pengujian keamanan sistem tanpa melakukan eksploitasi aktif terhadap target. Pendekatan ini dipilih untuk memastikan proses analisis tidak menyebabkan gangguan layanan, kehilangan data, atau pelanggaran kebijakan keamanan dari pihak pemilik sistem.

### 2.1. Perencanaan dan Penentuan Ruang Lingkup

Penelitian diawali dengan tahap perencanaan yang berfokus pada penetapan batasan operasional, jadwal pelaksanaan, serta perolehan izin etis dari otoritas terkait. Objek yang menjadi fokus evaluasi adalah situs web publik milik Instansi A. Pemilihan situs ini didasarkan pada karakteristik sistemnya yang bersifat dinamis dan telah mengimplementasikan protokol HTTPS sebagai standar komunikasi data.

Penentuan ruang lingkup (*scoping*) dilakukan secara ketat guna memastikan bahwa pengujian hanya terbatas pada komponen publik, yang mencakup domain utama, direktori halaman web, serta *endpoint* yang tersedia bagi pengguna umum. Seluruh rangkaian prosedur pengujian dilaksanakan secara terkontrol dengan menjunjung tinggi prinsip *responsible disclosure*. Hal ini menjamin bahwa setiap aktivitas pemindaian dan analisis tidak mengganggu ketersediaan layanan maupun integritas data pada operasional sistem yang sedang berjalan.

### 2.2. Pengumpulan Informasi *(Reconnaissance)*

Tahap kedua dalam penelitian ini adalah pengumpulan informasi atau *reconnaissance*, sebuah proses krusial untuk memetakan karakteristik sistem target secara mendalam. Tahapan ini bertujuan untuk memproyeksikan gambaran umum mengenai arsitektur jaringan, identifikasi layanan aktif, serta spesifikasi teknologi yang diimplementasikan pada server web. Dalam pelaksanaannya, proses ini mengandalkan dua instrumen utama:

1. **Nmap (*Network Mapper*):** Alat ini digunakan untuk melakukan pemindaian terhadap alamat IP domain target. Tujuannya adalah untuk menginventarisasi *port* yang terbuka, mendeteksi layanan yang sedang berjalan (seperti protokol HTTP, HTTPS, atau SSH), serta melakukan *fingerprinting* terhadap versi perangkat lunak yang digunakan. Data yang diperoleh menjadi parameter penting dalam memetakan potensi kerentanan pada lapisan jaringan.
2. **WHOIS:** Prosedur ini dilakukan untuk mengekstraksi data registrasi domain, yang mencakup informasi *registrar*, entitas pemilik, masa aktif domain, hingga rincian konfigurasi DNS. Informasi tersebut berfungsi sebagai instrumen verifikasi legalitas domain serta membantu pemetaan keterhubungan infrastruktur jaringan secara administratif.

Seluruh data yang terhimpun pada fase *reconnaissance* ini kemudian disusun menjadi sebuah ringkasan struktur teknis yang komprehensif. Dokumen teknis tersebut menjadi fondasi utama dan panduan strategis dalam pelaksanaan tahap pemindaian kerentanan pada fase berikutnya

### 2.3. Pemindaian Otomatis *(Automated Scanning)*

Tahap ketiga merupakan fase inti dalam *Vulnerability Assessment*, yakni pelaksanaan pemindaian otomatis terhadap komponen aplikasi web. Instrumen utama yang digunakan dalam fase ini adalah OWASP ZAP (*Zed Attack Proxy*), yang dipilih karena kapabilitasnya sebagai perangkat *open-source* yang komprehensif serta dukungannya terhadap metode analisis non-destruktif.

Dalam penelitian ini, OWASP ZAP dioperasikan menggunakan mode Passive Scanning. Melalui mekanisme ini, perangkat akan memantau dan menganalisis lalu lintas data HTTP/HTTPS antara sisi klien dan server secara *real-time* tanpa melakukan modifikasi data ataupun pengiriman *payload* yang bersifat eksploitatif. Fitur-fitur utama yang dioptimalkan meliputi:

1. **Spidering:** Berfungsi untuk melakukan perayapan (*crawling*) sistematis guna memetakan seluruh struktur halaman serta *endpoint* yang tersedia pada situs target.
2. **Passive Scan Rules:** Digunakan untuk mengidentifikasi kesalahan konfigurasi keamanan yang bersifat umum, seperti ketiadaan *security headers*, pengelolaan *cookie* yang tidak aman (absennya atribut *Secure* atau *HttpOnly*), hingga deteksi konten campuran (*mixed content*).

Seluruh temuan dari hasil pemindaian otomatis ini didokumentasikan ke dalam *scan report* terperinci. Laporan tersebut selanjutnya menjadi bahan mentah yang wajib divalidasi melalui proses verifikasi manual guna menjamin akurasi data serta mengeliminasi potensi kesalahan identifikasi (*false positive*).

### 2.4. Pemeriksaan Manual *(Manual Verification)*

Langkah berikutnya adalah proses pemeriksaan manual yang bertujuan untuk memvalidasi dan memastikan relevansi hasil pemindaian otomatis terhadap kondisi aktual sistem. Tahapan ini sangat krusial sebagai prosedur mitigasi terhadap *false positive*—yakni anomali yang terdeteksi oleh mesin pemindai namun tidak merepresentasikan risiko keamanan yang nyata. Untuk mencapai akurasi data yang tinggi, penelitian ini memanfaatkan beberapa instrumen teknis:

1. **curl:** Perangkat ini digunakan untuk melakukan permintaan HTTP secara langsung guna mengevaluasi respons dari server secara mendalam. Melalui metode ini, peneliti dapat melakukan inspeksi pada *response header* untuk memverifikasi implementasi elemen keamanan seperti *Content-Security-Policy* (CSP), atribut pada *Set-Cookie*, serta informasi pada *header Server*. Selain itu, *curl* digunakan untuk mengonfirmasi konsistensi pengalihan protokol dari HTTP ke HTTPS.
2. **openssl:** Instrumen ini difokuskan untuk mengaudit konfigurasi SSL/TLS secara komprehensif. Dengan *openssl*, peneliti dapat mengidentifikasi versi protokol TLS yang aktif, menganalisis kekuatan *cipher suite* yang digunakan, serta memeriksa validitas dan masa kadaluarsa sertifikat digital. Evaluasi ini menjadi parameter utama dalam mengukur ketangguhan enkripsi dan kepatuhan sistem terhadap standar keamanan komunikasi data modern.

Sinergi antara hasil pemindaian otomatis dan analisis manual ini menghasilkan potret keamanan yang lebih holistik, yang mencakup evaluasi mendalam baik pada lapisan aplikasi maupun lapisan *transport* (jaringan).

### 2.5. Analisis Hasil dan Rekomendasi

Fase final dari penelitian ini adalah analisis komprehensif serta interpretasi terhadap seluruh data pengujian. Semua temuan yang diperoleh dari pemindaian otomatis dan validasi manual dikompilasi ke dalam sebuah matriks hasil pengujian yang mencakup parameter-parameter berikut:

1. Instrumen pengujian yang digunakan,
2. Klasifikasi dan deskripsi temuan keamanan,
3. Analisis dampak potensial terhadap integritas sistem, serta
4. Rekomendasi perbaikan teknis.

Proses analisis dilakukan dengan melakukan komparasi antara temuan lapangan dengan kerangka kerja OWASP Top 10 untuk menentukan taksonomi kerentanan serta tingkatan risiko yang dihasilkan. Dalam merumuskan saran perbaikan, penelitian ini menerapkan prinsip prioritas mitigasi, di mana penanganan difokuskan pada celah keamanan dengan tingkat risiko tertinggi yang memiliki potensi dampak paling kritis.

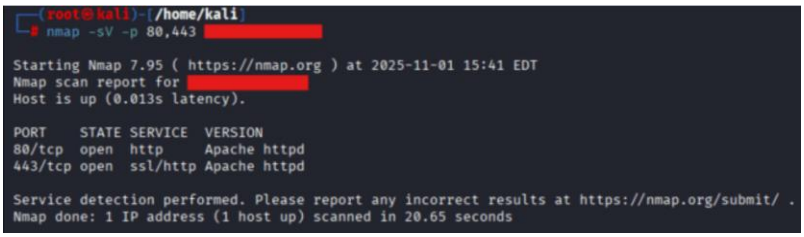
Output dari tahapan ini tidak sekadar berupa inventarisasi celah keamanan, melainkan sebuah cetak biru strategi peningkatan keamanan berkelanjutan (*continuous security improvement*). Hasil ini diharapkan dapat menjadi panduan strategis bagi Instansi A dalam memperkuat postur keamanan aplikasi web mereka terhadap ancaman siber di masa depan.

## 3. Hasil dan Pembahasan

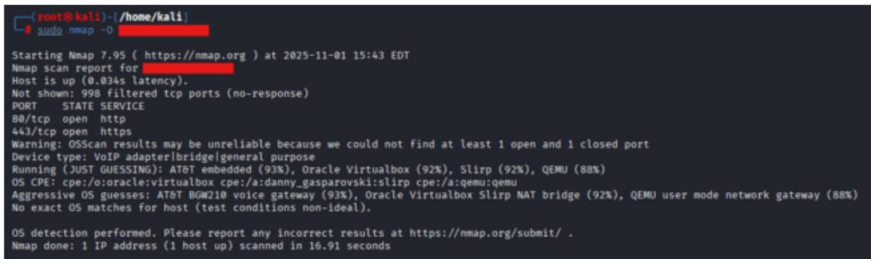
Bagian ini memaparkan seluruh temuan teknis yang diperoleh dari proses *Vulnerability Assessment* (VA) non-destruktif terhadap platform web Instansi A. Fokus utama dari pemaparan ini adalah mengevaluasi efektivitas konfigurasi keamanan pada lapisan jaringan, arsitektur aplikasi web, serta integritas sertifikat SSL/TLS dan validitas informasi domain. Hal ini bertujuan untuk memproyeksikan gambaran holistik mengenai postur keamanan sistem informasi yang menjadi objek penelitian. Evaluasi dilakukan melalui sinergi instrumen otomatis, yakni OWASP ZAP dan Nmap, yang didukung oleh perangkat verifikasi manual seperti curl, OpenSSL, dan Whois. Metodologi hibrida ini memungkinkan identifikasi kelemahan secara presisi di berbagai lapisan mulai dari infrastruktur jaringan hingga manajemen identitas digital dengan tetap menjamin ketersediaan layanan tanpa menimbulkan gangguan operasional pada sistem yang berjalan.

### 3.1. Identifikasi Port dan Layanan Aktif (Nmap)

Tahap awal evaluasi dilakukan menggunakan Nmap (*Network Mapper*), sebuah instrumen pemindaian jaringan yang dioptimalkan untuk mengaudit aksesibilitas infrastruktur target. Fokus dari pengujian ini adalah untuk menginventarisasi seluruh port yang terbuka, mengidentifikasi jenis layanan (*services*) yang sedang berjalan, serta melakukan estimasi terhadap sistem operasi yang diimplementasikan pada server web Instansi A. Data yang dihimpun melalui proses ini krusial untuk memetakan *attack surface* (permukaan serangan) dan mendeteksi potensi kerentanan pada level infrastruktur sebelum beralih ke lapisan aplikasi.




**Gambar 1.** Nmap




**Gambar 2.** Hasil nmap

Berdasarkan visualisasi hasil pengujian pada Gambar 1 dan 2, identifikasi terhadap infrastruktur jaringan Instansi A menghasilkan beberapa poin temuan utama:

1. **Aksesibilitas Port:** Hanya *port* 80 (HTTP) dan 443 (HTTPS) yang teridentifikasi dalam status terbuka (*open*). Hal ini mengonfirmasi bahwa server didedikasikan sepenuhnya untuk layanan web berbasis protokol standar.
2. **Identifikasi Layanan dan Lingkungan:** Target menggunakan Apache HTTP Server versi 2.4.x yang dioperasikan di dalam lingkungan virtualisasi. Informasi ini memberikan indikasi mengenai arsitektur server yang digunakan untuk mendukung skalabilitas layanan.

3. **Evaluasi Keamanan Jaringan:** Tidak ditemukannya *port* tambahan yang terbuka menunjukkan bahwa implementasi kebijakan *firewall* telah dikonfigurasi dengan prinsip *least privilege*, sehingga meminimalkan *entry point* bagi ancaman luar.

Secara keseluruhan, meskipun konfigurasi *firewall* pada lapisan jaringan sudah cukup solid, penggunaan Apache versi 2.4.x menjadi catatan penting. Eksposur versi server yang tidak diperbarui ke rilis terbaru dapat meningkatkan risiko terhadap **unpatched vulnerabilities** atau kerentanan yang telah diketahui publik (*known exploits*). Oleh karena itu, diperlukan langkah pembaruan (*patching*) secara berkala untuk memitigasi potensi eksploitasi pada kerentanan lama.

### 3.2. Evaluasi Keamanan Aplikasi Web (OWASP ZAP)

Tahapan evaluasi dilanjutkan dengan mengimplementasikan OWASP ZAP (*Zed Attack Proxy*) untuk membedah aspek keamanan pada lapisan aplikasi web. Pengujian ini dioperasikan melalui metode *passive scanning*, sebuah pendekatan audit yang berfokus pada analisis lalu lintas data tanpa mengintervensi integritas sistem. Dengan mekanisme ini, identifikasi kelemahan konfigurasi dilakukan murni melalui observasi terhadap respons server, sehingga tidak memerlukan pengiriman *payload* eksploitasi yang berisiko mengganggu stabilitas operasional layanan.


**Gambar 3.** Hasil OWASP ZAP

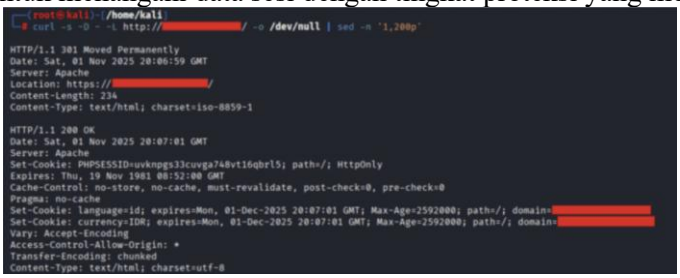
Berdasarkan data yang dihimpun pada Gambar 3, instrumen OWASP ZAP berhasil mengidentifikasi sejumlah anomali keamanan pada lapisan aplikasi. Temuan tersebut secara garis besar diklasifikasikan sebagai berikut:

1. **Absensi *Security Headers*:** Komponen krusial seperti *Content-Security-Policy* (CSP), *X-Frame-Options*, dan *Strict-Transport-Security* (HSTS) belum diimplementasikan pada response server. Ketiadaan mekanisme pertahanan ini secara signifikan meningkatkan kerentanan sistem terhadap serangan *Clickjacking* dan *Cross-Site Scripting* (XSS), terutama dalam skenario pemuatan konten dari sumber eksternal yang tidak terpercaya.
2. **Kelemahan Manajemen Sesi:** Ditemukan sejumlah *cookie* yang tidak dilengkapi dengan atribut keamanan esensial, yakni *Secure* dan *HttpOnly*. Kondisi ini memperbesar risiko intersepsi data sesi melalui koneksi yang tidak terenkripsi maupun akses ilegal via skrip di sisi klien (*client-side scripts*).
3. **Eksposur Informasi Teknis:** Terdapat komentar internal di dalam kode sumber HTML yang masih terekspos. Hal ini berpotensi membocorkan rincian teknis sensitif yang dapat dimanfaatkan oleh penyerang untuk memahami struktur internal aplikasi.
4. **Kerentanan Kontrol Transaksi:** Formulir input pada aplikasi belum mengintegrasikan token anti-CSRF (*Cross-Site Request Forgery*), yang membuka celah bagi pihak ketiga untuk memalsukan permintaan atas nama pengguna yang sah.

Selain itu, pemindaian ini berhasil memetakan penggunaan pustaka pihak ketiga seperti jQuery, Bootstrap, dan Font Awesome. Secara kolektif, hasil temuan menunjukkan bahwa sebagian besar risiko keamanan pada platform Instansi A bersumber dari konfigurasi yang belum optimal pada *web server* dan lapisan logika aplikasi, bukan pada kerusakan struktural kode.

### 3.3. Pemeriksaan Header dan Cookie (CURL)

Tahapan berikutnya melibatkan penggunaan curl sebagai instrumen verifikasi manual untuk menginspeksi HTTP response header secara *real-time*. Prosedur ini bertujuan untuk melakukan audit langsung terhadap parameter keamanan yang dikirimkan oleh server pada setiap permintaan. Fokus utama dari pengujian ini adalah untuk memvalidasi keberadaan *security headers* serta memeriksa secara mendalam atribut keamanan pada set-cookie, guna memastikan apakah server telah menginstruksikan peramban (*browser*) untuk menangani data sesi dengan tingkat proteksi yang memadai.




**Gambar 4.** Hasil CURL

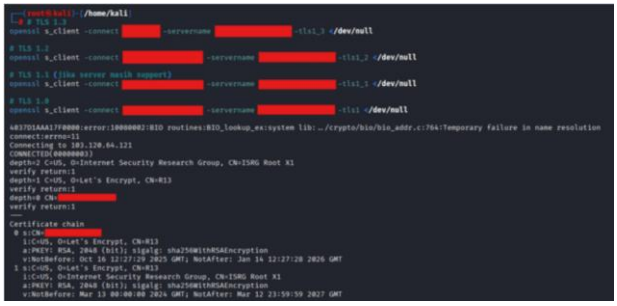
Berdasarkan data yang disajikan pada Gambar 4, hasil eksekusi perintah curl -s -D - -L terhadap domain target memberikan rincian teknis mengenai perilaku server dan manajemen sesi sebagai berikut:

1. **Implementasi Enkripsi *Default*:** Server secara otomatis melakukan pengalihan (*redirect*) dari protokol HTTP ke HTTPS. Hal ini menunjukkan bahwa mekanisme enkripsi pada jalur komunikasi telah diaktifkan secara menyeluruh untuk menjamin privasi transmisi data bagi setiap pengguna.
2. **Anomali Atribut Cookie Sesi:** Identitas sesi yang direpresentasikan melalui *cookie* PHPSESSID terdeteksi telah menggunakan atribut HttpOnly, yang efektif mencegah akses skrip sisi klien. Namun, absennya atribut Secure menjadi celah kritis, karena server tidak membatasi pengiriman *cookie* tersebut hanya pada jalur terenkripsi.
3. **Kurangnya Proteksi pada Cookie Tambahan:** Beberapa *cookie* lain yang berkaitan dengan preferensi pengguna ditemukan tidak memiliki *flag* keamanan esensial seperti **Secure** maupun **SameSite**. Ketiadaan atribut *SameSite* khususnya, dapat meningkatkan kerentanan terhadap serangan *Cross-Site Request Forgery* (CSRF).

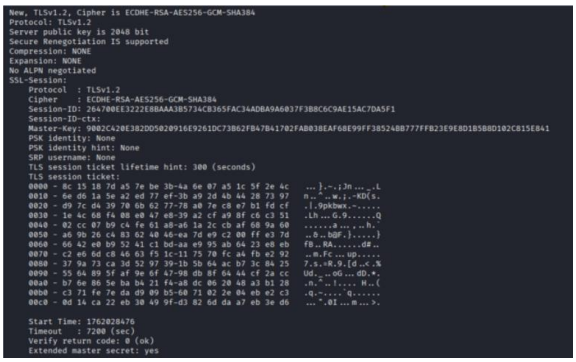
Analisis teknis ini mengindikasikan bahwa meskipun jalur komunikasi telah terenkripsi, manajemen *cookie* masih memiliki kelemahan yang signifikan. Ketiadaan atribut *Secure* membuka peluang terjadinya Session Hijacking, di mana data sesi dapat terekspos jika terdapat gangguan atau degradasi pada koneksi aman, terutama saat pengguna mengakses aplikasi melalui infrastruktur jaringan publik yang tidak terpercaya.

### 3.4. Pemeriksaan Sertifikat dan Konfigurasi SSL/TLS (OpenSSL)

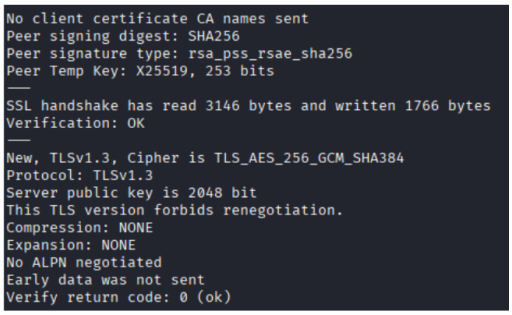
Langkah selanjutnya melibatkan penggunaan instrumen OpenSSL untuk melakukan audit mendalam terhadap konfigurasi protokol SSL/TLS serta integritas sertifikat digital pada platform Instansi A. Prosedur ini bertujuan untuk memverifikasi kekuatan algoritma enkripsi yang diimplementasikan, memastikan penggunaan versi protokol yang aman, serta memvalidasi rantai sertifikasi (*certificate chain*) guna menjamin bahwa identitas digital server telah terotentikasi oleh otoritas sertifikat (*Certificate Authority*) yang sah. Hasil dari evaluasi ini sangat krusial untuk menentukan tingkat ketangguhan saluran komunikasi dalam melindungi data dari ancaman *man-in-the-middle* (MITM).



**Gambar 5.** Hasil OpenSSL



**Gambar 6.** Server Mendukung TLS




**Gambar 7.** SHA 256

Berdasarkan data teknis yang tersaji pada Gambar 5, 6, dan 7, hasil eksekusi perintah openssl s_client terhadap domain Instansi A memberikan gambaran komprehensif mengenai postur keamanan pada lapisan *transport* sebagai berikut:

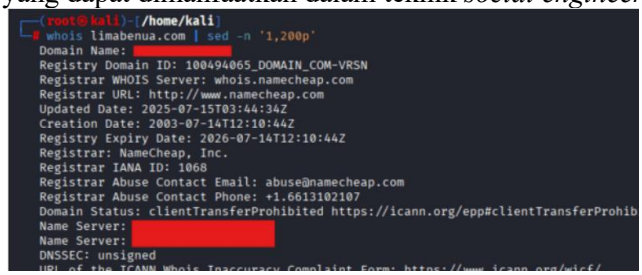
1. **Validitas dan Otentikasi Sertifikat:** Sertifikat digital terkonfirmasi telah diterbitkan oleh Otoritas Sertifikat (*Certificate Authority*) yang bereputasi dan masih dalam masa berlaku aktif. Hal ini menjamin bahwa identitas digital server dapat dipercaya oleh peramban pengguna.
2. **Evaluasi Protokol TLS:** Infrastruktur server telah mengimplementasikan protokol **TLS 1.2** dan **TLS 1.3**. Penonaktifan protokol usang seperti TLS 1.0 dan 1.1 menunjukkan kepatuhan terhadap standar keamanan modern guna memitigasi risiko serangan degradasi protokol (*downgrade attacks*).
3. **Kekuatan *Cipher Suite* dan Enkripsi:** Penggunaan *cipher suite* **TLS_AES_256_GCM_SHA384** mengindikasikan penerapan algoritma enkripsi yang sangat kuat dan efisien. Didukung dengan panjang kunci **RSA 2048-bit** serta algoritma tanda tangan **SHA256**, sistem ini telah memenuhi kriteria proteksi data yang tangguh terhadap upaya *brute-force* konvensional.

Secara keseluruhan, konfigurasi lapisan *transport* telah mengikuti praktik terbaik (*best practices*) keamanan terkini. Namun, terdapat catatan terkait manajemen siklus hidup sertifikat, di mana belum terdeteksi adanya mekanisme pembaruan otomatis (*automated renewal*). Absennya fitur ini

menimbulkan risiko operasional berupa potensi kegagalan akses layanan (karena sertifikat kadaluarsa) apabila proses pengawasan manual tidak dilakukan secara disiplin.

### 3.5. Pemeriksaan Informasi Domain (WHOIS)

Tahap final dari rangkaian pengujian ini melibatkan penggunaan protokol WHOIS untuk melakukan audit administratif terhadap aset domain Instansi A. Prosedur ini bertujuan untuk mengekstraksi data fundamental terkait status registrasi, memverifikasi masa aktif domain guna mencegah risiko kehilangan kepemilikan, serta mengevaluasi efektivitas perlindungan privasi (*privacy protection*) pada data pemilik domain. Analisis ini penting untuk memetakan keterhubungan infrastruktur secara legal dan memastikan bahwa informasi sensitif mengenai administrator sistem tidak terekspos secara publik yang dapat dimanfaatkan dalam teknik *social engineering*.




**Gambar 8.** Hasil WHOIS

Berdasarkan data yang dipaparkan pada Gambar 8, hasil penelusuran melalui protokol WHOIS terhadap domain Instansi A menunjukkan kondisi administratif sebagai berikut:

1. **Status Registrasi dan Tata Kelola:** Domain terkonfirmasi berada dalam status aktif dan dikelola oleh *registrar* resmi. Hal ini menjamin legalitas serta keberlangsungan hak akses publik terhadap layanan web tersebut.
2. **Privasi Data Administratif:** Implementasi fitur **WHOIS Privacy Protection** telah diaktifkan dengan baik. Pengaktifan fitur ini sangat krusial karena berhasil menyembunyikan identitas personil serta rincian kontak pemilik domain dari akses publik, sehingga memitigasi risiko serangan berbasis *social engineering* dan pengumpulan informasi oleh pihak yang tidak berwenang.
3. **Proteksi Keamanan Domain:** Status administrasi menunjukkan parameter **"clientTransferProhibited"**, yang mengindikasikan adanya protokol pengamanan tambahan untuk mencegah pemindahan kepemilikan domain secara ilegal atau tanpa otoritas resmi. Selain itu, masa aktif domain yang masih panjang menunjukkan manajemen aset digital yang terencana dengan baik.

Secara keseluruhan, tata kelola administratif domain Instansi A telah memenuhi standar keamanan yang optimal. Perlindungan terhadap data registrasi yang solid memastikan bahwa tidak ada informasi administratif yang dapat disalahgunakan untuk melancarkan serangan siber pada tingkat lanjut.

### 3.6. Analisis Hasil dan Rekomendasi

| No | Lapisan Keamanan | Tools Utama | Hasil Utama | Tingkat Risiko | Rekomendasi Utama |
|---|---|---|---|---|---|
| 1 | Jaringan | Nmap | Port 80 (HTTP) dan 443 (HTTPS) terbuka; firewall efektif, tidak ada port tambahan | Rendah | Perbarui versi Apache dan tutup port tidak digunakan |
| 2 | Aplikasi Web | OWASP ZAP, curl | Tidak ada header keamanan (CSP, HSTS, X-Frame-Options); cookie tanpa Secure & SameSite | Menengah | Tambahkan security header dan atur flag cookie dengan aman |
| 3 | Transport / Enkripsi | OpenSSL | TLS 1.3 aktif, cipher kuat, sertifikat valid | Rendah | Aktifkan auto-renew sertifikat dan nonaktifkan cipher lama |
| 4 | Identitas Domain | Whois | Domain aktif, perlindungan privasi WHOIS aktif, registrar resmi | Rendah | Pemantauan rutin masa berlaku domain |

**Gambar 9.** Hasil Analisis

Berdasarkan rangkaian pengujian yang telah dipaparkan, dapat disimpulkan bahwa arsitektur keamanan web Instansi A memiliki postur yang solid pada lapisan jaringan dan *transport*, namun masih menyisakan celah kerentanan pada lapisan aplikasi yang memerlukan atensi segera. Ringkasan tingkat risiko pada setiap lapisan dijabarkan sebagai berikut:

1. **Lapisan Jaringan (Nmap):** Implementasi kebijakan *firewall* terbukti sangat efektif dengan hanya mengizinkan lalu lintas pada *port* esensial (HTTP dan HTTPS). Hal ini meminimalkan permukaan serangan pada level infrastruktur, sehingga diklasifikasikan dalam kategori **Risiko Rendah**.
2. **Lapisan Aplikasi (OWASP ZAP & *curl*):** Absennya *HTTP Security Headers* serta pengelolaan *cookie* yang belum menyertakan atribut *Secure* dan *SameSite* menjadi titik lemah utama. Kondisi ini menciptakan celah terhadap ancaman *Cross-Site Scripting* (XSS), *Clickjacking*, dan *Session Hijacking*, sehingga ditempatkan pada kategori **Risiko Menengah**.
3. **Lapisan Transport (OpenSSL):** Adopsi protokol TLS 1.3 dan penggunaan *cipher suite* yang tangguh menunjukkan komitmen terhadap standar enkripsi modern. Keamanan jalur komunikasi data berada pada kategori **Risiko Rendah**.
4. **Lapisan Identitas (WHOIS):** Manajemen aset digital telah dikelola dengan baik melalui perlindungan privasi aktif dan status administratif yang terkunci, menempatkannya pada kategori **Risiko Rendah**.

Secara akumulatif, hasil evaluasi menetapkan tingkat risiko sistem pada kategori Menengah (*Medium Risk*). Strategi mitigasi di masa mendatang harus diprioritaskan pada penguatan konfigurasi keamanan aplikasi, khususnya melalui standardisasi *HTTP Security Headers* dan pengetatan manajemen sesi pengguna. Penelitian ini sekaligus membuktikan bahwa sinergi antara automasi pemindaian dan validasi manual mampu menghasilkan analisis keamanan yang holistik, presisi, serta tetap menjaga integritas operasional sistem target.

## 4. Kesimpulan

Penelitian ini telah berhasil melaksanakan evaluasi komprehensif terhadap postur keamanan aplikasi web pada Instansi A melalui metodologi *Vulnerability Assessment* yang mengintegrasikan penggunaan OWASP ZAP, Nmap, serta analisis mendalam pada protokol SSL/TLS. Temuan teknis menunjukkan bahwa infrastruktur pada lapisan jaringan dan *transport* telah dikonfigurasi dengan standar yang tinggi. Hal ini dibuktikan dengan pembatasan akses hanya pada *port* esensial (HTTP dan HTTPS) serta pengadopsian standar enkripsi modern TLS 1.3 dengan *cipher* yang tangguh, mencerminkan tata kelola keamanan jaringan dan sertifikat digital yang efektif.

Meskipun demikian, celah keamanan yang signifikan masih teridentifikasi pada lapisan aplikasi. Kelemahan utama terletak pada absennya implementasi *security headers* krusial—seperti *Content-Security-Policy* (CSP) dan *X-Frame-Options*—serta manajemen *cookie* yang belum mencapai titik optimal karena belum konsisten dalam menerapkan atribut *Secure, HttpOnly,* dan *SameSite*. Eksistensi celah ini secara langsung meningkatkan eksposur sistem terhadap risiko serangan *Cross-Site Scripting* (XSS), *Clickjacking*, dan *Session Hijacking*, yang berpotensi mengkompromikan kerahasiaan serta integritas data pengguna.

Secara keseluruhan, tingkat risiko keamanan pada web Instansi A diklasifikasikan dalam kategori menengah (*medium risk*). Sebagai langkah strategis, prioritas perbaikan harus diarahkan pada penguatan konfigurasi lapisan aplikasi, standarisasi *HTTP Security Header*, serta optimalisasi manajemen sesi secara berkala. Model asesmen ini dapat direplikasi oleh instansi lain sebagai kerangka kerja keamanan preventif yang sistematis, terukur, dan selaras dengan standar global OWASP Top 10 serta praktik terbaik dalam industri keamanan informasi.

## 5. Ucapan terimakasih

Penulis menyampaikan terima kasih yang sebesar-besarnya kepada Tim Seminar UNISAYOGYA yang telah meluangkan waktu dan memberikan kontribusi dalam penyusunan serta penyediaan template jurnal ini. Dukungan tersebut sangat membantu dalam proses penulisan, penyusunan format, serta penyelarasan gaya penulisan agar sesuai dengan standar publikasi ilmiah. Penulis juga berterima kasih kepada semua pihak yang telah memberikan dukungan moral dan teknis selama proses penelitian dan

penyusunan karya ilmiah ini. Semoga hasil penelitian ini dapat memberikan manfaat bagi peningkatan kesadaran dan penerapan keamanan aplikasi web di berbagai instansi.

**Daftar Pustaka**

Alenezi, M., & Al-Haidari, F. (2023). Comprehensive Security Assessment of Web Applications: Challenges and Best Practices. *Journal of Cyber Security and Information Systems*, 12(2), 45–62.

Bernhard, J. (2022). *Automated Vulnerability Scanners: A Comparative Study on Detection Accuracy*. Berlin: Springer Nature. 1-210.

Cwekan, S., & Shumba, R. (2021). Mitigating Cross-Site Scripting (XSS) and SQL Injection using OWASP Best Practices. *International Journal of Computer Science & Information Security*, 19(5), 112–125.

Kals, S., Kirda, E., Kruegel, C., & Vigna, G. (2020). SecuBat: A Generic Web Vulnerability Scanner. In *Proceedings of the 15th International Conference on World Wide Web*. Edinburgh, Scotland: ACM Press. p. 247–256.

Lazuardi, M. R. (2021). *Evaluasi Keamanan Sistem Informasi Menggunakan Metode Vulnerability Assessment pada Instansi Publik*. Tesis. Universitas Gadjah Mada; 2021.

Lyon, G. F. (2021). *Nmap Network Scanning: The Official Nmap Project Guide to Network Discovery and Security Scanning*. Third Edition. California: Insecure.Com LLC. 1-468.

Oppenheim, J. (2020). *Preventive Cybersecurity: Implementing Regular Vulnerability Assessments*. Academic Press. 50-125.

OWASP Foundation. (2021). *OWASP Top 10:2021 - The Ten Most Critical Web Application Security Risks*. [cited 2024 May 20]. Available from: https://owasp.org/www-project-top-ten/

Stallings, W. (2021). *Cryptography and Network Security: Principles and Practice*. Eighth Edition. Harlow: Pearson Education. 1-845.

Suleman, M., & Zulfiqar, A. (2019). Comparative Analysis of Automated Web Vulnerability Scanners. In *Proceedings of the 2nd International Conference on Computing and Information Sciences*. Karachi; 2019. p. 45–52.