Journal of Information Technology and Computer Science (INTECOMS)
Volume 8 Nomor 5, Tahun 2025
e-ISSN : 2614-1574
p-ISSN : 2621-3249

EVALUASI CELAH KEAMANAN DENGAN METODOLOGI *VULNERABILITY ASSESSMENT* SEBAGAI PENILAIAN TINGKAT KERENTANAN PADA DOMAIN UNUD.AC.ID

SECURITY GAP EVALUATION USING THE VULNERABILITY ASSESSMENT METHODOLOGY AS A MEANS OF ASSESSING THE LEVEL OF VULNERABILITY IN THE UNUD.AC.ID DOMAIN

DD Hassel Putra Q[1], Ilham Ammarul Aziz[2], Eginna Gresia Purba[3], Dewa Made Wiharta[4], I Gusti Ayu Garnita Darmaputri[5]
Program Studi Teknik Elektro, Fakultas Teknik, Universitas Udayana[1,2,3,4,5]
hasselputra@student.unud.ac.id[1], ammarulaziz@student.unud.ac.id[2],
eginna.gresia002@student.unud.ac.id[3], wiharta@unud.ac.id[4], garnitaputri@unud.ac.id[5]

ABSTRACT

Website security is crucial for educational institutions managing sensitive data. Universitas Udayana has over 500 subdomains, yet a comprehensive security evaluation has not been conducted. This study analyzes the security of the ee.unud.ac.id subdomain as a sample using the Vulnerability Assessment method with white box testing. Testing was performed using OWASP ZAP, Nessus, RapidScan, SQLMap and Snort IDS, based on OWASP Top 10 and the CIA Triad. The results identified 27 threats, including 3 High, 7 Medium, and 11 Low risks, along with 24 alerts detected by Snort IDS, indicating external threats. Recommended mitigations include improving security configurations, strengthening firewalls, and updating systems. The implementation of Snort proved effective in real-time threat detection, while OWASP Top 10 and CIA Triad-based analysis provided clear mitigation guidelines. This study emphasizes the importance of regular security testing to protect academic systems from cyber threats.

Keywords*: Website Security, Vulnerability Assessment, White Box Testing, OWASP ZAP, Nessus, Intrusion Detection System, Snort, CIA Triad, OWASP Top 10*.

ABSTRAK

Keamanan *website* sangat penting bagi institusi pendidikan yang mengelola data sensitif. Universitas Udayana memiliki lebih dari 500 *subdomain*, tetapi belum dilakukan evaluasi keamanan menyeluruh. Penelitian ini menganalisis keamanan *subdomain* ee.unud.ac.id sebagai sampel menggunakan metode *Vulnerability Assessment* dengan teknik *white box testing*. Pengujian dilakukan dengan OWASP ZAP, Nessus, RapidScan, SQLMap dan Snort IDS, berdasarkan OWASP Top 10 dan CIA Triad. Hasilnya, ditemukan 27 ancaman dengan 3 *High*, 7 *Medium*, dan 11 *Low*, serta 24 *alert* dari Snort IDS, termasuk ancaman eksternal. Mitigasi yang disarankan meliputi peningkatan konfigurasi keamanan, *firewall*, dan pembaruan sistem. Implementasi Snort terbukti efektif dalam deteksi ancaman *real-time*, sementara analisis berbasis OWASP Top 10 dan CIA Triad memberikan panduan mitigasi yang jelas. Studi ini menegaskan pentingnya pengujian keamanan berkala untuk melindungi sistem akademik dari serangan siber.

Kata Kunci: Keamanan Website, Vulnerability Assessment, White Box Testing, OWASP ZAP, Nessus, Intrusion Detection System, Snort, CIA Triad, OWASP Top 10.

PENDAHULUAN

Aspek keamanan *website* adalah hal yang penting dari sistem informasi. Namun, masalah keamanan ini sering kali tidak diperhatikan oleh pemilik atau *administrator* sistem informasi. Persyaratan keamanan sistem aplikasi untuk melindungi data meliputi aspek kerahasiaan (*confidentiality*), aspek integritas (*integrity*) dan aspek ketersediaan (*availability*) dari keamanan sistem aplikasi [1]. Universitas Udayana, sebagai institusi pendidikan tinggi, mengelola lebih dari 500 *subdomain* di bawah *domain unud.ac.id*. Sekitar 50 di antaranya adalah sistem informasi utama yang terintegrasi melalui IMISSU (*Single Sign-On*), sementara sisanya berupa *website* yang dikelola oleh program studi, fakultas, dan unit-unit internal lainnya. Sebagian besar website ini

menggunakan *template* yang seragam, dikembangkan oleh Unit Sumber Daya Informasi (USDI). Namun, belum pernah dilakukan evaluasi keamanan secara menyeluruh terhadap *template* tersebut, sehingga meningkatkan risiko kerentanan yang dapat dieksploitasi pihak tidak bertanggung jawab.

Berdasarkan kondisi tersebut, pengujian keamanan menjadi langkah krusial dalam mengidentifikasi celah keamanan, mengukur tingkat risiko, serta menyusun rekomendasi mitigasi terhadap potensi serangan. *Subdomain* ee.unud.ac.id dipilih sebagai sampel penelitian karena menggunakan *template* standar yang sama dengan ratusan *subdomain* lainnya, sehingga hasil pengujian dapat mewakili kondisi keamanan secara lebih luas.

Penelitian ini mengadopsi standar dari *Open Web Application Security Project* (OWASP), khususnya OWASP Top 10 tahun 2021, untuk mengevaluasi potensi kerentanan pada sistem informasi berbasis web. Pendekatan ini memungkinkan identifikasi terhadap berbagai kelemahan umum yang sering menjadi sasaran serangan siber. Selain analisis kerentanan, penelitian juga mencakup pengujian penetrasi dan audit keamanan untuk memastikan bahwa kebijakan serta prosedur telah sesuai dengan standar industri.

Penelitian ini juga mengimplementasikan *Intrusion Detection System* (IDS), dalam hal ini Snort, untuk mendeteksi aktivitas mencurigakan dan potensi serangan secara *real-time*. Hasil dari penelitian ini diharapkan dapat memberikan gambaran kondisi keamanan *website* di Universitas Udayana dan membantu USDI sebagai pengelola sistem informasi untuk meningkatkan ketahanan terhadap ancaman siber, meminimalkan risiko kebocoran data, serta menjaga kepercayaan civitas akademika terhadap infrastruktur digital Universitas Udayana.

KAJIAN PUSTAKA

Keamanan Informasi

Gambar 1. Komponen Keamanan Informasi[2]

Keamanan Informasi (Gambar 1.) menurut *Committee of National Security Systems* (CNSS) adalah keamanan informasi sebagai proteksi atau perlindungan terhadap informasi atau elemen-elemen yang bersifat kritikal termasuk didalamnya sistem dan perangkat keras yang digunakan untuk menyimpan dan mentransmisikan informasi. Keamanan informasi mencakup beberapa komponen-komponen utama seperti manajemen keamanan informasi, keamanan data dan keamanan jaringan [2].

OWASP TOP 10

OWASP Top 10 adalah daftar yang diterbitkan oleh komunitas OWASP yang memberikan daftar 10 kerentanan teratas yang dapat mengancam keamanan situs web, beserta risiko, dampak dan penyalahgunaannya. Daftar ini terus bertambah dan berubah seiring dengan perkembangan teknologi situs web yang terus berkembang [3].

CIA TRIAD

CIA triad merupakan 3 aspek penting dari data yang terdiri oleh c*onfidentiality*, *integrity* dan *availability*. CIA triad dapat berdampak besar dalam sebuah bisnis komputerisasi karena data dapat diartikan sebagai komponen inti untuk berbagai macam bisnis. Data harus dijamin keintegritasannya, dimana pada informasi digital tidak memiliki kerusakan dan hanya dapat diakses oleh yang berwenang atas

informasi data tersebut. Jadi Integritas dapat diartikan sebagai keharusan menjaga keakuratan, konsistensi, dan kepercayaan data suatu sistem informasi [4].

Penetration Testing

Penetration testing merupakan metode dan prosedur yang dilakukan dengan tujuan untuk menguji dan melindungi keamanan suatu organisasi dengan cara menemukan kerentanan (*vulnerability*) yang ada didalam suatu organisasi dan memeriksa apakah penyerang (*attacker*) dapat mengeksploitasi hingga mendapatkan akses yang tidak sah. *Penetration testing* dibagi menjadi 3 jenis, yaitu *Black Box Testing* (tanpa informasi), *White Box Testing* (informasi penuh), dan *Gray Box Testing* (sedikit informasi) [5].

Vulnerability

Vulnerability atau celah keamanan merupakan kesalahan, kelemahan, kerentanan ataupun kecacatan dari sebuah desain, implementasi, operasi maupun manajemen pada sebuah sistem yang disebabkan oleh penyerangan *(attacking)* dari pihak luar terhadap sistem itu sendiri [6].

Vulnerability assessment

Vulnerability assessment adalah adalah proses utama perbaikan menuju kepada kematangan dan intergrasi sistem keamanan informasi. Sedangkan Pengujian Penetrasi (*penetration testing*) hanya potret efektifitas sistem keamanan informasi yang sudah diterapkan saat potret dilakukan. Karena dasar inilah, tanpa bermaksud mengesampingkan pengujian penetrasi, Penilaian Kerentanan dapat memberikan nilai tambah lebih kepada organisasi dibandingkan dengan Pengujian Penetrasi atas pentingnya Sistem Manajemen yang efisien [7].

OWASP ZAP

OWASP *Zed Attack Proxy* (ZAP) adalah alat pemindai kerentanan yang paling sering digunakan dalam pengujian dan dikembangkan secara open source oleh organisasi OWASP karena kemudahan instalasi dan penggunaanya, ZAP dapat digunakan baik oleh pemula serta penguji tingkat lanjut [8].

Nessus

Nessus *Scanner* merupakan aplikasi yang dikembangkan oleh Tenable Security untuk mendeteksi celah keamanan (*vulnerability*) dalam suatu jaringan. Nessus dikenal sebagai salah satu alat *vulnerability assessment* yang paling populer dan banyak digunakan oleh profesional keamanan siber. Aplikasi ini memiliki beberapa fungsi utama, antara lain *vulnerability scanning, configuration auditing, compliance ch*ecks, dan malware detection [9].

Nikto

Nikto *Scanner* adalah salah satu alat pemindai kerentanan *(vulnerability scanner*) berbasis *open-source* yang digunakan untuk mengidentifikasi kelemahan keamanan pada *server* web. Nikto dirancang untuk melakukan pemindaian *komprehensif* terhadap berbagai konfigurasi *server,* plugin, serta mendeteksi *file* atau direktori yang rentan terhadap eksploitasi. Alat ini mampu mendeteksi lebih dari 6700 *file* atau aplikasi web yang berbahaya, 1250 *server* dengan konfigurasi yang salah, serta 270 versi *server* yang sudah usang atau memiliki celah keamanan *(Sullo,* 2023) [10].

Rapidscan

Rapidscan adalah alat *multiscanner otomatis* yang digunakan untuk mendeteksi berbagai kerentanan pada aplikasi web. Alat ini menggabungkan hasil dari beberapa pemindai keamanan untuk mengidentifikasi celah umum seperti *SQL injection, XSS, LFI,* dan lainnya [11].

SQLMap

SQLMap adalah tools *open-source* yang terdapat dalam sistem operasi kali

linux. Aplikasi ini digunakan untuk mendeteksi dan mengeksploitasi kerentanan aplikasi web. *Tools* ini mampu mengambil alih *database server*. Dengan menggunakan SQLMap, penyerang atau *tester* dapat melakukan penyerangan pada *database* SQL, menjalankan perintah pada sistem operasi, mengambil detail struktur *database*, melihat dan menghapus data yang terdapat dalam *database* dan mengakses sistem *file* dari server [12].

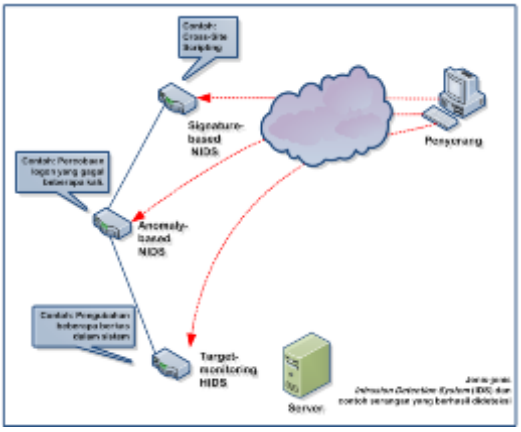
Nmap

Nmap *(Network Mapper)* adalah sebuah perangkat lunak sumber terbuka *(open source*) yang dirancang untuk mengeksplorasi dan mengaudit keamanan jaringan. Dikembangkan oleh Gordon Lyon (*Fyodor*). Nmap pertama kali dirilis pada tahun 1997 dan sejak itu menjadi alat standar dalam dunia keamanan siber. Nmap digunakan untuk memetakan jaringan, mengidentifikasi *host* yang aktif, serta mengeksplorasi *port* dan layanan yang berjalan pada suatu sistem [13].

Intrusion Detection System

Intrusion Detection System (IDS) adalah sistem pencegahan dengan menggunakan software atau hardware yang bekerja secara otomatis untuk memonitor keadaan pada jaringan komputer dan dapat menganalisis masalah keamanan jaringan. IDS adalah tools, metode, dan sumber daya yang memberikan bantuan untuk melakukan identifikasi, memberikan laporan terhadap aktivitas jaringan komputer [14].

Intrusion Detection System berfungsi melakukan pengamatan terhadap kegiatan-kegiatan yang tidak lazim pada jaringan sehingga awal dari langkah para penyerang bisa diketahui. Dengan demikian *Administrator* bisa melakukan tindakan pencegahan dan bersiap atas kemungkinan yang akan terjadi. Dalam mengenali pola serangan, ada beberapa metode bagaimana IDS bekerja yaitu: *Signature Based IDS* dan *Anomaly Based IDS* [15].



Gambar 2. Cara Kerja IDS[15]

Snort

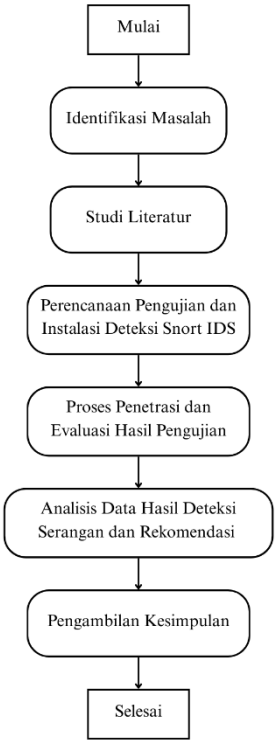
Snort merupakan sistem deteksi dan pencegahan intrusi jaringan (*Network Intrusion Detection and Prevention System*/NIDPS) berbasis sumber terbuka. Snort mampu menganalisis lalu lintas jaringan dan aliran data secara *real time*. Sistem ini dapat melakukan analisis protokol serta mendeteksi berbagai jenis serangan. Pada implementasinya sebagai NIDS, Snort memeriksa paket-paket data berdasarkan aturan (*rules*) yang ditulis oleh pengguna. Aturan Snort dapat ditulis dalam berbagai bahasa, memiliki struktur yang baik, mudah dibaca, dan juga dapat dimodifikasi dengan mudah.

Setiap kali terdapat paket data yang masuk ke dalam jaringan, Snort akan memeriksa perilaku jaringan. Jika terdeteksi penurunan performa jaringan, Snort akan menghentikan pemrosesan paket tersebut, membuang paket, dan menyimpan detailnya ke dalam basis data signature [16].

METODE PENELITIAN

Tahapan Penelitian

Penelitian ini dilakukan di Unit Sumber Daya Informasi (USDI) Universitas Udayana. Pengujian dilakukan secara *real-time* pada web Teknik Elektro Universitas Udayana. Waktu pengerjaan penelitian ini dimulai dari bulan Maret 2024 sampai dengan bulan Februari 2025. Alur tahapan penelitian dapat dilihat pada Gambar 3.



Gambar 3. *Flowchart* Tahapan Penelitian

Berikut penjelasan dari *flowchart* tahapan penelitian dari Gambar 3:

Langkah 1. Identifikasi Masalah

Tahapan pertama dilakukan dengan mengidentifikasi isu-isu keamanan pada *domain* unud.ac.id, khususnya pada *subdomain* ee.unud.ac.id, yang digunakan sebagai sampel representatif. Permasalahan yang diangkat mencakup proses instalasi dan konfigurasi Snort sebagai *Intrusion Detection System* (IDS), pelaksanaan *penetration testing* berbasis *framework* OWASP Top 10 2021 dan prinsip CIA Triad, serta evaluasi terhadap tingkat keamanan sistem berdasarkan hasil pengujian yang dilakukan. Identifikasi permasalahan ini menjadi dasar dalam perumusan arah penelitian serta penyusunan tahapan metode yang relevan.

Langkah 2. Studi Literatur

Tahapan kedua yaitu studi literatur dilakukan untuk memperkuat landasan teori dan kerangka konseptual penelitian, dengan menelaah referensi dari buku, jurnal ilmiah, serta sumber daring terpercaya. Studi ini juga mencakup pembatasan ruang lingkup penelitian dan perencanaan pembagian tugas dalam pelaksanaannya.

Langkah 3. Perencanaan Pengujian dan Instalasi Deteksi Snort IDS

Tahapan ketiga yaitu melibatkan perancangan alur pengujian, penyusunan dokumen perizinan untuk pelaksanaan evaluasi, serta identifikasi perangkat lunak yang akan digunakan. Instalasi dan konfigurasi alat pemindai keamanan serta penyusunan *rules* pada Snort IDS menjadi bagian integral dari tahapan ini.

Langkah 4. Proses Penetrasi dan Evaluasi Hasil Pengujian

Tahapan keempat yaitu Pengujian keamanan dilaksanakan melalui serangkaian tahapan: *information gathering*, *scanning*, *exploitation*, dan *reporting*. Proses ini memanfaatkan berbagai *tools penetration testing* untuk mensimulasikan serangan, sekaligus mengevaluasi kapabilitas IDS dalam mendeteksi dan merespons ancaman yang terjadi selama simulasi.

Langkah 5. Analisis Data Hasil Deteksi Serangan dan rekomendasi

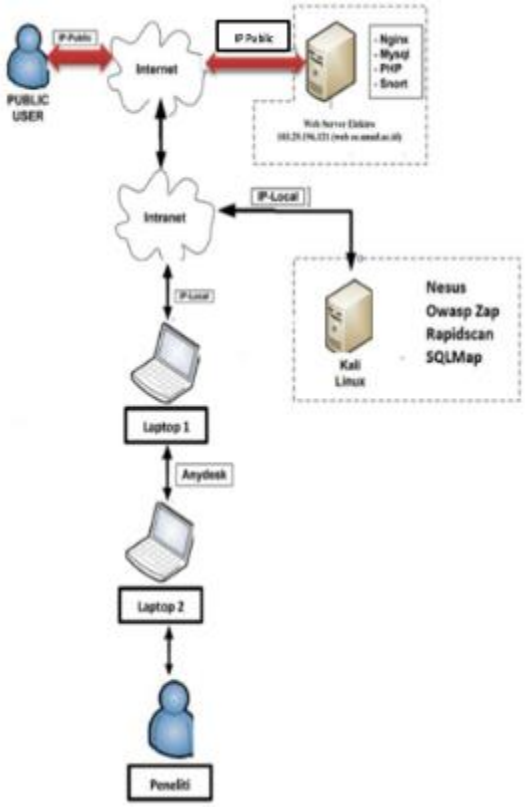
Tahapan kelima yaitu analisis difokuskan pada data yang dihasilkan dari sistem IDS dan *tools* pengujian penetrasi. Efektivitas Snort dalam mengidentifikasi serta mengklasifikasikan jenis serangan dianalisis secara kuantitatif dan kualitatif. Selain itu, setiap kerentanan yang ditemukan dikaji dan dibandingkan untuk menghasilkan rekomendasi mitigasi berdasarkan OWASP Top 10 2021 dan prinsip CIA Triad.

Langkah 6. Pengambilan Kesimpulan

Tahapan terakhir yaitu Kesimpulan disusun berdasarkan hasil analisis keseluruhan, mencakup efektivitas deteksi serangan oleh IDS, evaluasi tingkat kerentanan sistem, serta relevansi dan implikasi hasil terhadap peningkatan keamanan situs unud.ac.id.

Topologi Penelitian

Topologi penelitian menjelaskan struktur dan hubungan antar komponen yang digunakan dalam penelitian.



Gambar 4. Topologi Jaringan

Topologi jaringan pada lingkungan pengujian dirancang untuk memisahkan akses antara jaringan lokal dan jaringan publik. Web server yang menjadi objek pengujian berada pada jaringan lokal dengan IP 172.16.xxx.xxx dan dikelola secara internal oleh USDI. Akses dari luar dilakukan melalui IP publik 103.29.196.121. Server ini menjalankan sejumlah layanan utama seperti Nginx, MySQL, dan PHP, serta telah ditambahkan Snort sebagai Host-based Intrusion Detection System (HIDS) guna memantau lalu lintas jaringan dan mendeteksi potensi serangan. Snort dikonfigurasi untuk mengenali berbagai pola serangan, termasuk *scanning*, *DoS*, dan eksploitasi berbasis protokol. Pengujian dilakukan menggunakan Kali Linux yang dijalankan dalam lingkungan virtual lokal, dengan akses bertahap melalui SSH dan remote desktop dari perangkat yang berada di luar jaringan kampus.

HASIL DAN PEMBAHASAN

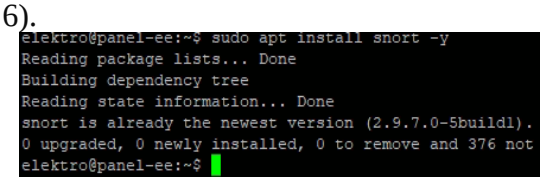
Instalasi dan Konfigurasi Snort sebagai *Intrusion Detection System* (IDS)

Penelitian ini mengimplementasikan Snort sebagai *Host-based Intrusion Detection System* (HIDS) pada *web server* ee.unud.ac.id untuk mendeteksi ancaman yang masuk melalui jaringan eksternal. Metode yang digunakan adalah teknik deteksi *signature-based* karena kemampuannya mengenali pola serangan yang telah terdokumentasi.

Proses instalasi dimulai dengan pembaruan sistem Ubuntu dan instalasi dependensi yang diperlukan. Setelah itu, dilakukan pengecekan antarmuka jaringan menggunakan perintah ip a, di mana ens160 berfungsi sebagai IP lokal dan ens192 sebagai IP publik (103.29.196.121) yang menjadi fokus pemantauan Snort.

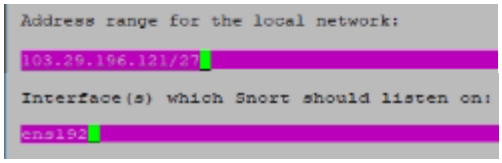
Gambar 5. *Network Interface* dari *web server*

Setelah sistem diperbarui dan pemeriksaan konfigurasi IP *address*, proses instalasi Snort dapat dilakukan dengan perintah sudo apt install snort -y (Gambar 6).



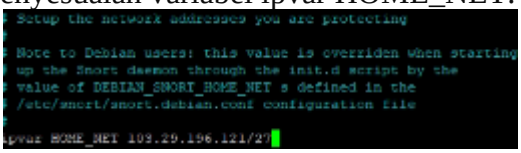
Gambar 6. Instalasi *Software* IDS Snort

Konfigurasi Snort terfokus pada *interface* ens192 karena lalu lintas dari pengguna eksternal masuk melalui antarmuka ini.



Gambar 7. Konfigurasi Penentu *Address Range Network* dan *Interface* Jaringan

Snort dikonfigurasi untuk memantau jaringan dengan mendefinisikan HOME_NET sebagai 103.29.196.121/27. Pengaturan dilakukan melalui file konfigurasi /etc/snort/snort.conf, dengan penyesuaian variabel ipvar HOME_NET.



Gambar 8. Mengubah variabel IP pada snort.conf

Validasi konfigurasi dilakukan menggunakan sudo snort -T -c /etc/snort/snort.conf untuk memastikan konfigurasi bebas dari kesalahan.

Untuk pemantauan *real-time*, digunakan perintah snort -A console -q -c /etc/snort/snort.conf -i ens192 -K ascii, di mana *output* ditampilkan langsung di konsol dalam format teks ASCII. Snort dalam mode ini mampu mendeteksi dan mencatat aktivitas mencurigakan yang melewati antarmuka publik server.

Proses dan Hasil Pengujian Keamanan (*Penetration Testing*)

Pengujian keamanan *website* ee.unud.ac.id menggunakan metode *white box testing*, dengan akses penuh melalui SSH ke *web server* dan sebagai admin situs. Tahapan pengujian mengikuti skema *Vulnerability Assessment* dan *Penetration Testing*, dimulai dengan *information gathering*, *network mapping*, dan dilanjutkan dengan eksploitasi terkontrol terhadap kerentanan yang ditemukan. Selanjutnya, Nmap digunakan untuk memetakan jaringan dan mengidentifikasi layanan terbuka. Ditemukan *port* HTTP (80), HTTPS (443), SMTPS (465), dan SNMP (161) dalam kondisi terbuka, dengan 97 *port* lainnya berada dalam status *filtered*.

Tabel 1 Daftar Kerentanan Menggunakan *Tools* Nikto, OWASP ZAP, dan Nessus

Kategori	Detail
IPv4 address & IP Location	103.29.196.121 (ee.unud.ac.id) (Indonesia - Bali - Badung - Universitas Udayana)
Sistem Operasi	Linux Ubuntu 18.04.4 LTS (Bionic)
Server	Nginx 1.14.0
Protocol Version	TLSv1.2
OpenSSL	1.1.1 (Dibangun pada 11 Sep 2018)
Versi PHP	7.2.24
Database	MySQL 5.7.29
Framework/Library	Laravel (backend), Handlebars.js
JavaScript *Libraries*	jQuery 3.3.1, jQuery Migrate, jQuery UI 1.11.4, Moment.js 2.20.1, OWL Carousel, Lightbox, Modernizr 2.6.2, DataTables 1.10.7
Cookie	XSRF-TOKEN, laravel_session

Tahapan awal dilakukan melalui proses *information gathering* dan *network mapping* untuk mengidentifikasi profil target secara menyeluruh (Tabel 1.). Peneliti mencoba mengumpulkan informasi teknis dan mengidentifikasi target yang akan diuji dengan memanfaatkan *tools* Whatweb, Whois, dan Wappalyzer di mana penggunaan *tools* ini mendeteksi *plugin*, nomor versi, identifikasi *Content Management System* (CMS), alamat email, dan modul dari *framework* pada *website* target dalam penelitian. Tabel 4.3 menunjukkan informasi hasil *Footprinting* dan *Reconnaissance* yang berguna dalam konteks tahap evaluasi celah keamanan.

Tahapan untuk mengumpulkan informasi lebih mendalam terkait pemetaan jaringan dan mengidentifikasi layanan yang aktif, *port* yang terbuka, konfigurasi sistem, serta sumber daya *(resources)* yang ada dalam ruang lingkup target.

Tabel 2. Rangkuman Hasil Tahap *Network Mapping*

Proses	*Tool*	Status	Hasil
Find Open Ports (*port* lain hanya terbuka untuk *IP internal*)	Nmap	Sukses	IP Address: 103.29.196.121 *Port State Service:* 80/tcp *Open* http, 443/tcp *Open* https, 465/tcp *Open* smtps

Check for FTP Service	Nmap	Gagal	Tidak terdeteksi *port* 21/tcp terbuka dari luar jaringan
Check for SNMP Service	Nmap	Sukses	*Port* 161/udp tidak terdeteksi terbuka dari luar jaringan.
Full UDP Port Scan	Nmap	Gagal	Sebagian besar (65532) *port* UDP berada dalam status *open|filtered.*

Berdasarkan hasil (Tabel 2.) pemindaian menggunakan Nmap dan verifikasi melalui konfigurasi UFW pada server target, dapat disimpulkan bahwa *port* yang benar-benar terbuka dan dapat diakses dari luar (eksternal) adalah *port* 80/tcp (HTTP), 443/tcp (HTTPS), dan 465/tcp (SMTPS). Sementara itu, *port* lain seperti 22/tcp (SSH), 161/udp (SNMP), serta *port* FTP (20 dan 21) memang diatur dalam status *ALLOW*, namun hanya dapat diakses oleh alamat IP internal tertentu (misalnya jaringan kampus dengan *subnet* 172.16.x.x atau 10.10.x.x).

Tahapan selanjutnya adalah *scanning* dan eksploitasi kerentanan, dimana tujuan lanjutan dari proses ini untuk membuktikan celah yang teridentifikasi, mendapatkan bukti dan dokumentasi hasil. Dapat juga memahami sejauh mana keamanan terhadap serangan yang dapat di eksploitasi oleh penyerang. *Tools* yang digunakan diantaranya Nikto, OWASP ZAP, Nessus, Rapidscan dan SQLMap.

1) Nikto

Nikto mendeteksi potensi celah keamanan miskonfigurasi atau *vulnerability scanning* ini dilakukan pada *domain* yaitu ee.unud.ac.id dengan IP 103.29.196.121 menggunakan Nikto v2.5.0. Hasil analisis menampilkan informasi terkait sertifikat SSL, konfigurasi *header* keamanan yang tidak diatur, serta potensi risiko keamanan lainnya.

Hasil pemindaian menggunakan Nikto mengungkapkan kelemahan konfigurasi pada server web, seperti tidak adanya header keamanan *X-Frame-Options*, *Strict-Transport-Security*, dan *X-Content-Type-Options* yang membuka risiko serangan *clickjacking*, *downgrade* HTTPS, dan salah interpretasi MIME. Cookie seperti XSRF-TOKEN, PHPSESSID, dan laravel_session juga ditemukan tanpa *flag Secure* dan HttpOnly, meningkatkan potensi serangan XSS dan pencurian sesi. Selain itu, server masih menggunakan nginx versi lama (1.14.0), dan beberapa direktori serta file sensitif (/downloads/, /staff/, web.config) terdeteksi dapat diakses publik, menunjukkan lemahnya kontrol akses direktori. Temuan ini menekankan perlunya penguatan konfigurasi keamanan pada tingkat aplikasi dan server.

2) OWASP ZAP

Proses dalam melakukan eksplorasi manual dengan menavigasikan situs web secara langsung menggunakan *browser* yang dikonfigurasi sebagai *proxy*, sehingga setiap *request* dan *response* HTTP dapat diamati untuk memahami struktur situs serta titik serangan potensial. Selanjutnya, digunakan *AJAX Spidering* untuk menjelajahi situs yang dinamis dan berbasis JavaScript, yang memungkinkan pendeteksian halaman atau *endpoint* tersembunyi yang tidak terjangkau oleh metode *crawling* tradisional.

Setelah eksplorasi selesai, dilakukan *Active Scan*, yaitu pengujian otomatis yang mengirimkan *payload* uji keamanan ke parameter yang ditemukan. Selain itu, *Attack Mode* diaktifkan untuk menguji setiap *request* yang melewati OWASP ZAP secara *real-time*, sehingga setiap halaman yang dikunjungi selama eksplorasi manual akan langsung dianalisis terhadap kemungkinan serangan.

Name	Risk Level	Number of Instances
SQL Injection	High	1
SQL Injection - SQLite	High	1
Vulnerable JS Library	High	5
Buffer Overflow	Medium	40
Content Security Policy (CSP) Header Not Set	Medium	541
Missing Anti-clickjacking Header	Medium	427
Secure Pages Include Mixed Content (Including Scripts)	Medium	366
Vulnerable JS Library	Medium	7

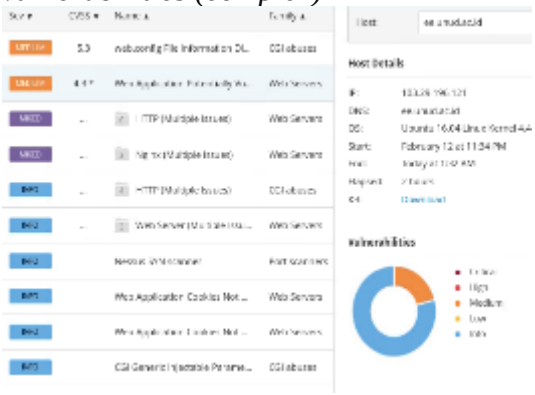
Gambar 9. Hasil Identifikasi Celah OWASP ZAP

Daftar celah hasil *scanning* dilakukan dengan metode *AJAX spider*, manual *explore*, dan *active scan*. Setiap celah yang teridentifikasi divalidasi dengan pengujian lanjutan untuk memastikan potensi eksploitasi aktual.

Ada beberapa tingkatan keparahan (*severity*) risiko yang dihasilkan, yaitu 3 level *high* dengan kerentanan yang jika tidak ditangani sangat berdampak merusak sistem, 5 level *medium* adalah kerentanan yang risiko kerusakan medium dan 11 level *low* adalah kerentanan yang resiko berdampak pada sistem rendah dan untuk level *informational* tidak berdampak.

3) Nessus

Tahap selanjutnya adalah melakukan identifikasi celah keamanan (*vulnerability)* dengan Nessus. Identifikasi celah ini dilakukan pada *domain* ee.unud.ac.id dengan tipe pemindaian "*Scan for all web vulnerabilities (complex)*".



Gambar 10. Hasil Identifikasi Celah Nessus

Hasil *scanning* pada domain ee.unud.ac.id menunjukkan adanya beberapa kerentanan dengan tingkat keparahan *Medium* (tiga kerentanan dengan skor CVSS 5.3 dan 4.3), serta sejumlah kerentanan informasi lainnya. Salah satu hasil pemindaian menggunakan *tool* Nessus mengungkapkan tiga kerentanan utama, yakni potensi pengungkapan informasi sensitif melalui file konfigurasi, kerentanan terhadap serangan *clickjacking*, dan tidak diaktifkannya mekanisme HSTS yang berpotensi meningkatkan risiko *downgrade attack* dan manipulasi sesi. Temuan ini selanjutnya diuji melalui proses *penetration testing* untuk memverifikasi keberadaan celah, memperoleh bukti eksploitasi, serta mendokumentasikan hasil sebagai bagian dari validasi keamanan sistem.

4) Rapidscan

RapidScan digunakan sebagai pemindai multi-modul untuk mengidentifikasi celah umum pada infrastruktur web target. Alat ini mengintegrasikan berbagai modul seperti pencarian DNS, *banner grabbing*, pemindaian *port*, dan serangan simulasi seperti *Slowloris DoS*.

Gambar 11. Hasil Identifikasi Celah Rapidscan

Hasil pemindaian menunjukkan keberadaan layanan MySQL dengan tingkat risiko rendah serta rekomendasi perbaikan terkait konfigurasi. Beberapa modul pemindaian tidak dapat dijalankan karena ketergantungan eksternal yang tidak aktif, namun sebagian besar fungsi inti RapidScan tetap berjalan. *Tool* ini memberikan gambaran awal tentang kondisi keamanan server, meskipun diperlukan pengujian tambahan yang lebih spesifik untuk validasi temuan dengan risiko menengah hingga tinggi.

5) SQLMap

Pengujian injeksi otomatis menggunakan SQLMap dilakukan pada berbagai *endpoint* situs web https://ee.unud.ac.id dengan parameter intensitas tinggi (--level=5, --risk=3) dan akses admin untuk menjangkau berbagai fitur menu dan parameter tersembunyi. SQLMap menguji sejumlah parameter *input* POST berbasis *multipart/form-data* pada *endpoint* seperti */post*, */page*, dan

/staffs, menggunakan teknik *time-based blind* dan *stacked queries.*

Gambar 12. Hasil Identifikasi Celah SQLMap

Meskipun terdapat beberapa indikator awal yang mengarah pada kemungkinan *SQL Injection*, seluruh hasil pengujian berakhir sebagai *false positive* atau tidak dapat dieksploitasi lebih lanjut. Hal ini mengindikasikan bahwa *framework* yang digunakan Laravel telah menyediakan proteksi efektif, seperti validasi CSRF token otomatis dan sanitasi *input* yang memadai. Pengujian menunjukkan bahwa tidak ada parameter yang berhasil diinjeksi SQL secara aktual, menandakan sistem relatif aman dari jenis serangan ini.

Hasil Deteksi Snort dan Hasil Pengujian

Snort beroperasi sebagai HIDS yang berfungsi untuk mendeteksi serangan yang menargetkan sistem secara *real-time*. Sistem ini memantau lalu lintas jaringan yang masuk dan keluar dari perangkat tempat Snort dipasang menggunakan *library* libpcap. *Library* ini menangkap paket dalam berbagai format protokol, seperti TCP, UDP, ICMP, dan lainnya. Snort kemudian menganalisis paket yang melewati antarmuka jaringan perangkat tersebut dengan membandingkan pola lalu lintas terhadap aturan (*rules*) yang telah ditentukan. Proses ini memungkinkan identifikasi berbagai ancaman, seperti serangan berbasis *host*, akses ilegal, eksploitasi layanan, serta aktivitas mencurigakan lainnya.

Hasil analisis lalu lintas disimpan dalam *file log*, yang mencakup informasi detail mengenai paket yang terdeteksi, termasuk alamat IP sumber dan tujuan, protokol yang digunakan, serta klasifikasi ancaman berdasarkan aturan deteksi. Format penyimpanan *log* ini biasanya berupa *plaintext*, JSON, atau format lain yang dapat dianalisis lebih lanjut. Dalam penelitian ini, pengambilan data dilakukan secara manual dengan membaca dan menyeleksi isi *file log* berdasarkan bagian yang dianggap penting. Langkah ini bertujuan untuk menghindari informasi redundan dan memastikan bahwa data yang dikumpulkan relevan untuk analisis keamanan pada perangkat yang dipantau.

Tabel 3. Hasil deteksi pada snort

Alert Type	*Classification*	*Priority*	*Source IP & Port*	*Destination IP & Port*
ICMP PING	*Misc activity*	3	16.50.246.37	103.29.196.121
EXPLOIT ntpdx overflow attempt	*Attempted Administrator Privilege Gain*	1	159.223.81.166:35880	103.29.196.121:123
DDOS Trin00 Master to Daemon default password	*Attempted Denial of Service*	2	103.29.196.113:61517	103.29.196.121:27444
DDOS mstream client to handler	*Attempted Denial of Service*	2	103.29.196.113:34844	103.29.196.121:15104

SCAN UPnP service discover attempt	*Detection of a Network Scan*	3	128.232.21.75:50249	103.29.196.121:1900
EXPLOIT ntpdx overflow attempt	*Attempted Administrator Privilege Gain*	1	198.98.58.11:51165	103.29.196.121:123

Hasil tabel 3. menunjukkan pengujian terdapat sejumlah aktivitas lalu lintas jaringan yang diklasifikasikan berdasarkan tingkat prioritas ancaman. Prioritas 1 mencerminkan ancaman tingkat tinggi yang berpotensi menyebabkan eskalasi hak akses atau kerusakan sistem melalui serangan *buffer overflow* atau injeksi kode berbahaya. Prioritas 2 menunjukkan ancaman tingkat menengah seperti *port scanning, vulnerability scanning,* atau upaya eksploitasi yang gagal. Sementara itu, Prioritas 3 lebih bersifat informatif, berkaitan dengan pola lalu lintas jaringan yang tidak biasa namun berpotensi berkembang menjadi ancaman yang lebih serius.

Snort berhasil mendeteksi berbagai jenis trafik yang berhubungan dengan aktivitas pengujian menggunakan RapidScan dan Nessus. Aktivitas ini meliputi *ICMP PING, traceroute, UPnP service discover attempt, MS-SQL ping attempt,* serta upaya eksploitasi seperti *ntpdx overflow attempt, DDOS Trin00 Master,* dan *mstream client to handler.* Hasil deteksi menunjukkan adanya upaya eksplorasi layanan jaringan serta pemindaian *port* yang bertujuan untuk mengidentifikasi potensi kerentanan pada server target.

Selain itu, terdapat sejumlah trafik dari sumber eksternal yang bukan bagian dari pengujian, seperti aktivitas *ICMP traceroute* dan *ntpdx overflow attempt* yang dideteksi berasal dari IP luar. Aktivitas ini mengindikasikan adanya upaya eksplorasi jaringan oleh pihak eksternal yang dapat mengancam keamanan server target.

Tabel 4 Daftar Kerentanan Menggunakan *Tools* Nikto, OWASP ZAP, dan Nessus

Vulnerability	*Level*	*Tool*
Vulnerable JS Library	*High*	OWASP ZAP
Buffer Overflow	*Medium*	OWASP ZAP
Missing Anti-clickjacking Header	*Medium*	Nikto, Nessus, OWASP ZAP
Content Security Policy (CSP) Header Not Set	*Medium*	Nessus, OWASP ZAP
Secure Pages Include Mixed Content	*Medium*	OWASP ZAP
Vulnerable JS Library	*Medium*	OWASP ZAP
Application Error Disclosure	*Medium*	OWASP ZAP
Big Redirect Detected	*Low*	OWASP ZAP

Hasil tabel 4. menunjukkan kerentanan yang terdeteksi pada *subdomain* ee.unud.ac.id melalui pengujian menggunakan Nikto, OWASP ZAP, dan Nessus. Rekomendasi perbaikan disusun berdasarkan OWASP Top 10 2021 dan prinsip-prinsip CIA Triad guna meningkatkan keamanan aplikasi web.

Dalam kategori A06:2021 – *Vulnerable and Outdated Components*, penggunaan *library JavaScript* (JS) yang rentan, seperti *jQuery Validation* dan *jQuery-migrate*, menjadi titik fokus utama. Komponen tersebut masih menggunakan versi lama yang berpotensi dieksploitasi oleh penyerang untuk menyisipkan skrip berbahaya, mengganggu integritas data, atau menjalankan kode berbahaya. Rekomendasi perbaikan adalah memperbarui *library* tersebut ke versi terbaru agar risiko tersebut dapat diminimalkan.

Buffer overflow diklasifikasikan dalam A09:2021 – *Security Logging and Monitoring Failures*. Kerentanan ini

memungkinkan *input* berlebih untuk dieksekusi oleh aplikasi, memicu eksekusi kode berbahaya atau eskalasi hak akses. Solusi yang direkomendasikan adalah menerapkan *return length checking* untuk memastikan *input* data tidak melebihi kapasitas *buffer*. Implementasi ini akan menjaga *Integrity* aplikasi dan mencegah kerusakan sistem yang dapat mengganggu *Availability*.

Pada aspek A05:2021 – *Security Misconfiguration*, absennya *header anti-clickjacking* memungkinkan penyerang menyisipkan halaman eksternal berbahaya ke dalam *frame* aplikasi. Implementasi *header* "*X-Frame-Options"* dan *"Content-Security Policy* (CSP)" menjadi solusi mitigasi utama. Selain itu, penggunaan *frame busting* JavaScript dapat membantu mencegah pemuatan halaman eksternal dalam *frame* aplikasi, menjaga *Confidentiality* dan *Integrity* aplikasi.

Masih dalam kategori A05:2021, tidak adanya header *Content-Security-Policy* (CSP) membuka peluang injeksi skrip berbahaya dari sumber eksternal.

SIMPULAN

Penelitian ini telah melakukan *penetration testing* dan deteksi serangan terhadap *subdomain* ee.unud.ac.id sebagai representasi dari situs web dengan struktur dan *template* serupa dalam domain *unud.ac.id*. Berdasarkan hasil pengujian dan analisis, diperoleh beberapa kesimpulan sebagai berikut:

1. Instalasi dan konfigurasi Snort sebagai *Intrusion Detection System* (IDS) dimulai dengan pembaruan sistem, pemasangan Snort, serta pengaturan alamat IP. Snort dikonfigurasikan untuk memantau lalu lintas jaringan pada antarmuka *ens192* yang menggunakan IP publik 103.29.196.121. *File* konfigurasi utama *snort.conf* disesuaikan untuk mendeteksi lalu lintas pada *HOME_NET*. Snort diatur agar mampu mendeteksi ancaman seperti DDoS, *SQL injection*, dan eksploitasi kerentanan, dan dijalankan secara *real-time* dengan hasil deteksi ditampilkan langsung pada konsol.
2. Pengujian penetrasi dilakukan berdasarkan *framework* OWASP Top 10 2021 dan prinsip CIA Triad, mencakup tahapan *information gathering*, *scanning*, *exploitation*, dan *reporting*. *Tools* yang digunakan antara lain WhatWeb, Wappalyzer, Nikto, Nessus, OWASP ZAP, dan SQLMap. Hasil pengujian mengungkapkan berbagai tingkat kerentanan yang diklasifikasikan sebagai *High, Medium, Low,* dan Informational.
3. Pemindaian menggunakan OWASP ZAP dalam *Attack Mode* dan *Manual Explore* mengidentifikasi 25 potensi ancaman keamanan, yang terdiri atas 3 kategori *High*, 5 *Medium*, 11 *Low*, dan 6 *Informational*. Sementara itu, hasil pengujian dari Nessus menunjukkan adanya 3 kerentanan dengan tingkat keparahan *Medium*, masing-masing dengan skor CVSS 5.3 dan 4.3. Hasil ini menunjukkan bahwa sistem belum dapat dikategorikan aman sepenuhnya dan masih memiliki celah yang perlu ditindaklanjuti.
4. Implementasi Snort pada *server hosting* situs mendeteksi 24 jenis *alert type* dan 5 klasifikasi serangan. Sebanyak 15 berasal dari pengujian internal menggunakan Kali Linux, sedangkan 9 lainnya berasal dari alamat IP eksternal yang tidak berhubungan dengan proses pengujian, yang menunjukkan adanya ancaman nyata dari pihak luar terhadap sistem.
5. Rekomendasi perbaikan yang disusun merujuk pada OWASP Top 10 2021 dan prinsip CIA Triad. Tindakan yang disarankan mencakup pembaruan rutin terhadap komponen perangkat lunak, konfigurasi server untuk menghindari pengungkapan informasi teknis, penerapan *security headers* seperti *X-Frame-Options*, CSP, dan HSTS, penguatan sistem *logging* dan *monitoring*, serta pengaturan *cookie* dengan atribut *Secure* dan *HttpOnly*.

Selain itu, audit sistem secara berkala dan pelarangan penggunaan skrip dari sumber yang tidak terpercaya merupakan langkah penting untuk menjaga integritas dan keamanan sistem.

DAFTAR PUSTAKA

[1] F. A. Fajar, "Analisis keamanan aplikasi web prodi teknik informatika uika menggunakan acunetix web. 2," vol. 3, no. 2, pp. 110–120, 2020.

[2] M. E Whitman and H. J. Mattord, "Principles of Information Security," 2012.

[3] OWASP (Open Web Application Security Project), "OWASP Top 10 - 2021," Accessed: Aug. 10, 2024. [Online]. Available: https://owasp.org/Top10/id/A00_2021_Introduction/

[4] P. R. Kumar, P. H. Raj, and P. Jelciana, (2018). "Exploring Data Security Issues and Solutions in Cloud Computing. Procedia Computer Science, vol. 125, no. 1, p.p. 691–697. Jan. 2018, doi: 10.1016/j.procs.2017.12.089

[5] G. Jayasuryapal, P. M. Pranay, H. Kaur and Swati, "A Survey on Network Penetration Testing," *2021 2nd International Conference on Intelligent Engineering and Management (ICIEM)*, London, United Kingdom, 2021, pp. 373-378, doi:10.1109/ICIEM51511.2021.944532.

[6] OWASP, "*Vulnerabilities,"* Accessed: Aug. 10, 2024. [Online]. Available: https://owasp.org/www-community/vulnerabilities/

[7] Itg, "Pemahaman Metodologi Vulnerability Assessment dan Pengujian Penetrasi," Accessed: Aug. 10, 2024. [Online]. Available: https://itgid.org/pemahaman-metodologi-vulnerability-assessment-dan-pengujian-penetrasi/

[8] F. P. E. Putra, U. Ubaidi, A. Hamzah, W. A. Pramadi, and A. Nuraini, "Systematic Literature Review: Security Gap Detection On Websites Using Owasp Zap," Brilliance: Research of Artificial Intelligence, vol. 4, no. 1, pp. 348–355, Jul. 2024, doi: 10.47709/brilliance.v4i1.4227..

[9] Tenable, "*Tenable Nessus*," 2023.

[10] C. Sullo, "*Nikto Web Scanner Documentation*," 2023.

[11] M. Zaidan, "Eksplorasi Keamanan Website Dinas XY di Jawa Timur dalam Pendekatan Pengujian Brute Force untuk mendeteksi kerentanan," 2024.

[12] S. Lika, R. D. P. Halim, and I. Verdian, "Analisa Serangan SQL Injeksi Menggunakan SQLMap," Jurnal Sistem dan Teknologi Informasi, vol. 4, no. 2, pp. 88-94, Nov. 2019.

[13] G. F. Lyon. "Nmap Network Scanning: The Official Nmap Project Guide to Network Discovery and Security Scanning," Insecure, Sunnyvale, CA, USA, 2009.

[14] D. Ariyus, "Intrusion Detection System: Sistem Deteksi Penyusupan Pada Jaringan Komputer," 2007.

[15] M. H. Dar and S. Z. Harahap, "Implementasi Snort Intrusion Detection System (IDS) Pada Sistem Jaringan Komputer," : Fakultas Sains dan Teknologi Universitas Labuhanbatu, vol. 6, no. 3, pp. 14-23, Sep. 2018.

[16] Kumar, Vinod, and O. P. Sangwan. "Signature based intrusion detection system using SNORT." *International Journal of Computer Applications & Information Technology*, vol. 1, no. 3, pp. 35-41, 2012.