BAB 2
LANDASAN TEORI

2.1 Penelitian Terdahulu

Beberapa penelitian terdahulu yang relevan telah tersedia dan dapat dijadikan sebagai acuan untuk membandingkan hasil penelitian yang dilakukan oleh penulis. Kerangka kerja yang saya buat akan disandingkan dengan kerangka data observasi dari penelitian terdahulu untuk menemukan perbedaan atau persamaan yang penting.

Penelitian pertama dengan judul "Penerapan Metode *Vulnerability Assessment* untuk Identifikasi Keamanan Website berdasarkan OWASP ID Tahun 2021 pada Universitas papua (UNIPA)" permasalahan dalam penelitian ini mengacu pada adanya potensi serangan siber yang menjadi suatu permasalahan yang tidak bisa dihindari, kelemahan keamanan website, kelalaian pengguna. Penelitian menunjukkan bahwa solusi kerentanan yang menggunakan sistem khusus seperti anti-CSRF, CSP, CDN, Strict-Transport-Security Header, dan pengecekan timestamp untuk membuat website proporsional. Pengguna harus menggunakan versi terbaru dari browser (Darmawan dkk., 2024).

Penelitian kedua dengan judul "Vulnerability Assessment Untuk Meningkatkan Kualitas Keamanan *Web* pada aplikasi berbasis web" masalah dalam penelitian ini adalah aplikasi web yang dibuat tanpa mempertimbangkan masalah keamanan dan dirancang oleh orang yang tidak berpengalaman dalam keamanan web, sehingga sangat rentan terhadap serangan seperti sql injection, denial of service, dan berbagai jenis malware. Pada penelitian ini, sistem operasi kali Linux digunakan untuk menjalankan Nmap. Host target yang diuji mendukung metode POST, OPTIONS, GET, dan HEAD. Hasil pengujian VA dengan nmap juga menunjukkan bahwa

target tersebut diidentifikasi sebagai HTTP open proxy. Setelah itu, target yang diuji tidak diidentifikasi untuk cross site scripting. Selain itu, host target tidak mendeteksi sql injection (Mira Orisa dan Ardita, 2021).

Penelitian ketiga dengan judul "Analisis Metode Open Web Application Security Project (OWASP) Menggunakan Penetration Testing pada Keamanan Website Absensi pada aplikasi web absensi sub.domain.com", adapun temuan yang menjadi masalah dalam penelitian ini yaitu kerentanan atau gangguan keamanan sistem banyak terjadi di internet. Adanya serangan Malware, Eksploitasi dan Injeksi database. Pengamanan web dari gangguan atau serangan hacker perlu dilakukan untuk meminimalisir dengan cara *penetration testing* (Pentest) yaitu pengujian yang dilakukan terhadap web secara legal dengan aktivitas menyerupai *hacker.* Penelitian dilakukan dengan standarisasi keamanan Open Web Application Security Project (OWASP) dengan menggunakan tools security (Riandhanu, 2022).

Penelitian keempat dengan judul "Analisis *Security Assessment* Menggunakan Metode *Penetration Testing* Dalam Menjaga Kapabilitas Keamanan Teknologi Informasi Pertahanan Negara pada web server" permasalahan dari penulis yaitu kebutuhan keamanan yang efektif untuk mengatasi serangan-serangan siber, ancaman *cyber attack* pada pusat pertahanan siber kementerian pertahanan (Zen dkk., n.d.).

Penelitian kelima dengan judul "Analisis Security Mitigation dengan Metode Vulnerability Assessment and Penetration Testing (VAPT) (Kasus Website Kerja Praktek dan Pengabdian Masyarakat)" dalam permasalahan ini meliputi kerentanan yang dilakukan pada pengujian sebelumnya memerlukan pengujian keamanan kembali untuk menemukan kerentanan lebih lanjut dan saran untuk melindungi data pengguna dan mengurangi kemungkinan serangan yang mengancam sistem website (Fadillah dkk., 2023).

Penelitian keenam dengan judul "Vulnerability Assessment Dan Penetration Testing (Vapt) Menggunakan Metode *Zero Entry Hacking* (Zeh) Terhadap Website. Studi Kasus: Dinas Penanaman Modal dan PTSP Kota Tangerang Selatan" Adapun permasalahan yang diangkat oleh penulis yaitu Dinas Penanaman Modal dan PTSP Kota Tangerang Selatan yang menjadi salah satu instansi yang memaksimalkan penggunaan website yang berisikan informasi penting bagi publik memerlukan penjagaan keamanan sistemnya untuk menanggapi masalah serangan siber di era digital ini (Yaqi, 2023).

Berikut adalah hasil penelitian terdahulu yang berkaitan dengan pengujian yang dilakukan dalam penggunaan metode VAPT. Penjelasannya dapat ditemukan pada tabel dibawah ini.

Table 1. Penelitian Terdahulu

No.	Peneliti	Judul	Metode	Hasil
1.	Candra Darmawan, Julius Panda Putra Naibaho, Alex De Kweldju, (2024)	Penerapan Metode Vulnerability Assessment untuk Identifikasi Keamanan Website berdasarkan OWASP ID Tahun 2021	Metode Vulnerability Assessment and Penetration Testing Life Cycle (VAPT). Metode VAPT pada penelitian ini melalui lima tahapan yaitu scope, information gathering, vulnerability assessment, risk assessment, dan reporting.	1. Diperoleh data alerts ID, alerts, risk, dan OWASP ID sebagai informasi kerentanan website UNIPA. 2. Metode vulnerability assessment menunjukkan, berdasarkan OWASP ID tahun 2021, mayoritas celah kerentanan adalah A05:2021 yaitu Security Misconfiguration serta paling sedikit A01:2021 yaitu Broken Access Control dan A08:2021 yaitu Software and Data Integrity Failures. Hasil membuktikan, konfigurasi sistem dan pengaturan keamanan website UNIPA masih belum baik.

Table 2. Lanjutan Penelitian Terdahulu

No.	Peneliti	Judul	Metode	Hasil
2.	Mira Orisa, Michael Ardita (2021)	VULNERABILITY ASSESSMENT UNTUK MENINGKATKAN KUALITAS KEAMANAN *WEB*	Metode vulnerability *assessment.*	1. Hasil pengujian VA dengan nmap menunjukkan bahwa target tersebut terdeteksi sebagai HTTP open proxy. 2. Hasil pengujian web server scanning termasuk mendeteksi metode pada protokol HTTP dengan perintah $nmap -p80,443 dan script http-methods pada host target yang mendukung metode POST, OPTIONS, GET, dan HEAD. 3. Hasil pengujian kategori script NSE nmap $nmap -p8080 menunjukkan bahwa target tersebut terdeteksi sebagai HTTP open proxy. 4. Hasil dari kategori script NSE nmap $ nmap mencakup script dns-brute.nse.nse di p 80 dan script http-unsafe-output-escaping di p 80 untuk menemukan dns yang valid. 5. Hasil dari kategori script NSE nmap $ nmap mencakup script dns-brute.nse.nse . Saat scanning dilakukan pada host target, tidak ditemukan cross site scripting.

Table 3. Lanjutan Penelitian Terdahulu

No.	Peneliti	Judul	Metode	Hasil
3.	Ichsan Octama Riandhanu (2022)	Analisis Metode Open Web Application Security Project (OWASP) Menggunakan Penetration Testing pada Keamanan Website Absensi	Penerapan metode OWASP sebagai standar pengujian aplikasi web.	Hasil dari penelitian ini dapat diambil kesimpulan bahwa analisis keamanan aplikasi berbasis web menggunakan metode OWASP telah terbukti mampu mengetahui kerentanan keamanan yang berada pada aplikasi web absensi subdomain.com. Berdasarkan hasil pengujian ditemukan kerentanan pada tingkat ancaman kritis (critical) sebanyak 1 temuan, tingkat tinggi (high) sebanyak 3 temuan, tingkat menengah (medium) sebanyak 4 temuan dan tingkat rendah (low) sebanyak 7 temuan. Kerentan yang ditemukan berdasarkan kategori OWASP top 10 adalah sensitive data exposure, security misconfiguration, dan using components with known vulnerabilities.

Table 4. Lanjutan Penelitian Terdahulu

No.	Peneliti	Judul	Metode	Hasil
4.	Bita Parga Zen, Rudy A.G Gultom, Agus H.S Reksoprodjo (2020)	ANALISIS SECURITY ASSESSMENT MENGGUNAKAN METODE PENETRATION TESTING DALAM MENJAGA KAPABILITAS KEAMANAN TEKNOLOGI INFORMASI PERTAHANAN NEGARA	Metode seperti Scanning Vulnerability standar Open Web Application Security Project, Common Vulnerability Scoring System yang digunakan untuk mengidentifikasi keamanan.	Dari hasil kerentanan menggunakan Common Vulnerability Scoring System (CVSS) terdapat 3 sistem yang terdeteksi bisa mengakibatkan data atau sistem diserang karena terdapat kelemahan pada server tersebut, dari 3 sistem tersebut diantaranya kerentanan diantaranya: Directory Listening artinya Folder yang terbuka, Insecure transition from http to https, HTML form without CSRF protection.

Table 5. Lanjutan Penelitian Terdahulu

No.	Peneliti	Judul	Metode	Hasil
5.	Muhammad Iqbal Fadillah, Umar Yunan Kurnia Septi Yanto, Muhammad Fathinuddin (2023)	Analisis Security Mitigation dengan Metode Vulnerability Assessment and Penetration Testing (VAPT) (Kasus Website Kerja Praktek dan Pengabdian Masyarakat)	Vulnerability Assessment and Penetration Testing (VAPT) dengan teknik gray box testing dan juga dengan tools Burp Suite, Acunetix dan Nessus.	1. Hasil mitigasi yang telah dilakukan menunjukan 4 dari 9 kerentanan berhasil di mitigasi dengan baik sehingga keamanan website menjadi lebih baik dari sebelumnya. Dengan menggunakan target website KPPM dan tools Burp Suite, Acunetix, dan Nessus, Detection of Vulnerability menghasilkan daftar kerentanan dengan hasil yang berbeda. Pengujian Burp suite menemukan sebelas kerentanan, terdiri dari satu kerentanan tingkat tinggi, empat kerentanan tingkat rendah, dan tiga kerentanan tingkat informasi. Pengujian nessus yang dilakukan pada port 443 Https menemukan tiga kerentanan tingkat medium, satu kerentanan tingkat rendah, dan dua kerentanan tingkat informational. Pengujian Acunetix menemukan dua kerentanan tingkat medium, tujuh kerentanan tingkat rendah, dan dua kerentanan tingkat informational.

Table 6. Lanjutan Penelitian Terdahulu

No.	Peneliti	Judul	Metode	Hasil
6.	Muhammad Yaqi (2023)	VULNERABILITY ASSESSMENT DAN PENETRATION TESTING (VAPT) MENGGUNAKAN METODE ZERO ENTRY HACKING (ZEH) TERHADAP WEBSITE. Studi Kasus: Dinas Penanaman Modal dan PTSP Kota Tangerang Selatan	Metode Zero Entry Hacking (Zeh)	1. Hasil pada penelitian ini berhasil ditemukan kerentanan dengan 3 tingkat medium dan 4 tingkat low. 2. Metode Zero Entry Hacking (ZEH) berhasil ditemukan celah kerentanan yang pada website Dinas Penanaman Modal dan PTSP Kota Tangerang Selatan antara lain, missing anti-clickjacking header, content security policy (csp) header not set, x-content-type- options header missing, cross domain javascript source file inclusion, cookie without secure flag, cookie without Httponly flag, server leaks information via 'x-powered-by' HTTP response header fields. 3. Hasil pengukuran kerentanan dengan CVSS yang menghasilkan tiga kerentanan dengan tingkat medium, dan empat kerentanan berada pada tingkat low, tanpa ditemukannya celah kerentanan dengan tingkatan high.

2.2 Kajian Pustaka

Bagian ini menjelaskan konsep dan prinsip dasar yang diperlukan untuk menyelesaikan masalah. Selain itu, literatur atau buku yang mendukung penelitian memberikan ringkasan teori. Tema penelitian berikut dapat disesuaikan dengan penulisan penelitian pustaka: teori dapat berupa uraian kualitatif atau alat yang relevan. Dalam kalimat atau pernyataan yang dirujuk, sumber teori yang digunakan harus disebutkan dan didaftarkan dalam Daftar Pustaka.

2.2.1 Kampus Unipdu Jombang

Universitas pesantren Tinggi Darul Ulum atau yang disingkat UNIPDU merupakan perguruan tinggi Islam swasta berbasis pesantren, dan menjadi universitas terkemuka di Indonesia. Unipdu Jombang didirikan oleh K.H. As'ad Umar dan berlokasi di tengah-tengah Pesantren Darul Ulum, Rejoso, Peterongan, Jombang. Saat ini, Unipdu Jombang telah terakreditasi oleh BAN-PT (Badan Akreditasi Nasional – Perguruan Tinggi) dengan peringkat B – Unggul. Pada tanggal 9 September 2001 kampus Unipdu ini diresmikan oleh Bapak Wakil Presiden RI Dr.HM Hamzah Haz dan disaksikan oleh para kyai sepuh pondok pesantren Darul 'Ulum dan pondok pesantren lain (unipdu.ac.id, 2024).

Gambar 1. Kampus Unipdu Jombang

2.2.2 Vulnerability

Dalam dunia keamanan sistem, "*vulnerability*" berarti adanya kelemahan atau cacat dalam sistem yang bisa menjadi jalan masuk serangan. Istilah ini mencakup kesalahan pada program, celah keamanan, atau pengaturan yang kurang baik, yang bisa dimanfaatkan oleh pihak yang tidak bertanggung jawab untuk merusak atau mencuri data. Sederhananya, vulnerability adalah titik lemah yang bisa membuat sistem menjadi lebih mudah disusupi atau diserang.

Penyebab adanya Vulnerability ini dikarenakan buatan manusia tidak ada yang benar- benar sempurna dalam bidang teknologi. Vulnerability sering kali muncul karena kesalahan yang dibuat pengembang saat melakukan coding atau menerapkan validasi yang kurang sempurna, sehingga meninggalkan celah pada infrastruktur sistem. Kondisi ini memungkinkan pihak luar atau asing yang sangat berisiko masuk ke dalam program, yang akhirnya dapat merusak privasi secara serius dan mengganggu keamanan data dalam sistem secara keseluruhan. Terdapat tiga jenis Vulnerability yang meliputi:

1) Vulnerability Perangkat Lunak

Kesalahan dalam pemrograman, seperti cross-site scripting (XSS) dan SQL injection, sering muncul selama proses pengembangan perangkat lunak. Kekeliruan bisa saja tidak disengaja, tetapi dapat dimanfaatkan oleh peretas. Di sisi lain, kelemahan pada desain terjadi ketika otentikasi permintaan pengguna tidak memadai, sehingga menciptakan celah yang bisa dieksploitasi oleh peretas.

2) Vulnerability Jaringan

Kelemahan pada jaringan mencakup semua komponen dari perangkat keras fisik hingga seluruh teknologi yang berjalan pada lapisan aplikasi dalam model OSI. Jaringan memiliki cakupan kerentanan yang luas karena melibatkan berbagai elemen, seperti perangkat keras dan perangkat lunak

di setiap tingkat jaringan. Setiap perangkat lunak dan perangkat keras berasal dari vendor yang berbeda, dan masing-masing memiliki risiko keamanan yang unik. Hal ini membuat pelacakan dan pengelolaan kerentanan jaringan menjadi rumit dan cukup menantang.

3) Vulnerability Konfigurasi dan Proses

Kesalahan dalam mengatur konfigurasi dapat membuat sistem lebih rentan terhadap masalah keamanan. Misalnya, risiko ini bisa muncul dalam cara pengaturan aliran data, penggunaan protokol, atau pengaturan autentikasi untuk memastikan jaringan berjalan sesuai yang diharapkan. Jika konfigurasi tidak tepat, lalu lintas data di jaringan bisa menjadi masalah dan berpotensi melanggar aturan yang telah ditetapkan (codingstudio, 2024).

Kerentanan atau vulnerability diklasifikasikan dalam tiga tingkat penilaian sebagai berikut:

1. **Tingkat Tinggi (High)**: Pada tingkat ini, kelemahan memiliki potensi besar menjadi ancaman serius.
2. **Tingkat Sedang (Medium)**: Pada tingkat sedang, kelemahan biasanya bersifat lokal, begitu pula dengan upaya pencegahan dan penanganannya yang ditujukan secara terbatas.
3. **Tingkat Rendah (Low)**: Pada tingkat rendah, kelemahan tidak signifikan, dan langkah pencegahan maupun penanganannya sudah sangat mencukupi.

2.2.3 Vulnerability Assessment (VA)

Vulnerability Assessment (VA) adalah proses yang bertujuan untuk mengidentifikasi, menilai, dan mengelompokkan tingkat risiko terkait kerentanan keamanan yang ada pada jaringan komputer, sistem, aplikasi, atau komponen lain dalam ekosistem IT. Vulnerability Assessment dapat dilakukan secara otomatis menggunakan platform atau tools khusus, sehingga proses pemantauan dapat berlangsung

terus-menerus untuk mendeteksi kerentanan baru yang muncul. Jika ditemukan kerentanan yang valid, langkah perbaikan dapat segera dilakukan. Penilaian kerentanan ini umumnya diterapkan pada aplikasi web, API, serta infrastruktur IT (Cyber Academy Indonesia, 2022).

2.2.4 Website

Website adalah kumpulan halaman yang berisi informasi tertentu yang dapat dengan mudah diakses oleh siapa saja yang terhubung ke internet, kapan saja dan di mana saja. Pemrograman web adalah cara atau proses untuk memberikan instruksi atau perintah kepada komputer yang terhubung ke internet untuk melakukan tugas atau fungsi tertentu. Dengan kata lain, pemrograman web adalah cara atau proses untuk memberikan instruksi atau perintah kepada komputer yang terhubung ke internet untuk melakukan tugas dan fungsi tertentu. Kemudian saat menjalankan aplikasi di web, Kita dapat menggunakan web browser seperti Opera, Firefox, Chrome, dll. Pada awalnya, *website* hanya digunakan oleh orang-orang pribadi, tetapi sekarang hampir semua perusahaan, instansi, bahkan bisnis kuliner memilikinya sebagai sumber informasi(Ummah, 2019).

1. Jenis – jenis *Website*

a) Website Statis

Website statis adalah *website* yang kontennya tidak pernah berubah atau di update dalam jangka waktu tertentu; contohnya adalah website profil perusahaan atau instansi.

b) Website Dinamis

Situs web yang dinamis adalah situs web yang konten atau komponen lainnya sering berubah secara teratur karena pengguna atau pengunjung lain dapat secara diam-diam mengubah informasi.

c) Website Interaktif

Website interaktif adalah jenis website yang memiliki kemiripan dengan website statis, namun menawarkan tingkat interaksi yang lebih tinggi. Situs ini dirancang agar pengguna dapat secara aktif berkontribusi, termasuk mengedit atau mengelola konten yang ada di dalamnya.

2.2.5 Web Security

Serangkaian tindakan dan strategi yang dikenal sebagai web security digunakan untuk melindungi situs web dari berbagai ancaman dan serangan siber yang dapat merusak dan membahayakan data dan pengguna yang mengaksesnya. Web security Melindungi situs web dari berbagai serangan yang dapat dilakukan oleh peretas dan ancaman dunia maya lainnya sangat penting. Di tengah pesatnya perkembangan ancaman siber, kehadiran sistem Web security yang kuat menjadi sangat diperlukan untuk menjaga kerahasiaan informasi dan menekan kerentanan terhadap risiko siber.

Peretasan adalah salah satu ancaman utama yang sering dihadapi situs web. Peretas bisa mencoba masuk tanpa izin, mencuri data penting, atau bahkan mengubah tampilan situs. Selain peretasan, ancaman tambahan seperti cross-site scripting (XSS) dan cross-site request forgery (CSRF), yang memungkinkan pencurian atau manipulasi data pengguna. Untuk menangkal berbagai ancaman tersebut, keamanan web menggunakan beberapa cara seperti kebijakan keamanan yang ketat. Contohnya, Content Security Policy (CSP) adalah langkah untuk mencegah serangan XSS dengan membatasi sumber eksternal yang diakses oleh situs. Selain itu, enkripsi data, autentikasi dua faktor, dan firewall juga merupakan perlindungan yang efektif.

Selain melindungi dari serangan, keamanan web juga menekan risiko peretasan dan pembobolan data. Dengan rajin

memperbarui sistem, menjaga integritas data, serta mengamankan server dan database, sebuah situs bisa menjadi lebih aman dan terhindar dari ancaman yang merugikan (Ardita, 2024).

2.2.6 Serangan DoS (*Denial of Service*) dan DDoS (*Distributed Denial of Service*)

Serangan siber yang dikenal sebagai "Denial of Service", atau juga dikenal sebagai "DoS", adalah serangan siber yang menargetkan sistem atau sumber layanan cloud dengan banyak permintaan yang berulang melalui satu komputer melalui jaringan internet. Jenis serangan ini juga dikenal sebagai "Distributed Denial of Service", yang merupakan versi lebih lanjut dari DoS yang menggunakan beberapa perangkat komputer untuk menyerang target sistem. Kedua jenis serangan ini menghentikan pengguna untuk menggunakan sistem atau layanan untuk sementara atau bahkan tanpa batas. (Wicaksono dan Suartana, 2023).

2.2.7 Malware

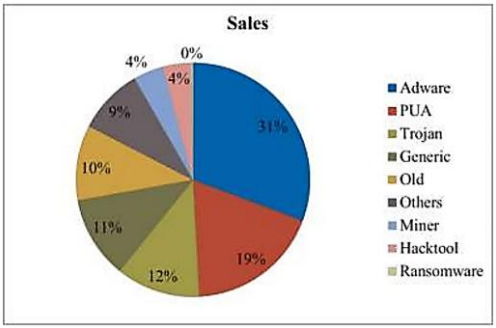
Malicious software, juga dikenal sebagai malware, adalah perangkat lunak berbahaya yang memiliki kemampuan untuk mengganggu atau bahkan merusak komputer, data, dan jaringan kita. Malware terbagi menjadi dua kategori: yang independen dan yang membutuhkan program host (program induk). Sebuah survey yang dilakukan oleh Stack Overflow pada tahun 2018 menunjukkan bahwa serangan malware pada Windows mencapai 49,4 persen, diikuti oleh MacOs 27,4 persen, Linux 23,5 persen, dan BSD/Linux 0,2 persen. Sistem operasi 32-bit adalah sistem operasi yang paling sering ditargetkan oleh pencipta malware.

a) Macam - Macam Malware

1. Virus adalah program replikasi diri yang dipasang pada perangkat lunak yang sah dan membutuhkan interaksi pengguna untuk menginfeksi sistem dengan berhasil. Virus memiliki kemampuan untuk menyebar dan berkembang di dalam sistem komputer, sehingga mereka dapat memperluas, mengurangi kapasitas memori dan hard disk. Salah satu cara untuk mencegah virus adalah dengan menghindari lampiran e-mail yang berasal dari sumber yang tidak diketahui.
2. Trojan. jenis malware yang menyerupai kuda Trojan, kuda yang terkenal dalam mitologi Yunani. Bisa berupa program apa pun yang terlihat seperti program yang sah, tetapi memiliki beberapa kode yang berpotensi berbahaya. Membutuhkan upaya tambahan untuk menyembunyikan diri karena sifatnya parasit dan tidak replikasi.
3. Spyware. jenis kode berbahaya ini digunakan untuk memata-matai aktivitas korban dan mencuri informasi sensitif. Pencurian identitas adalah modus operandi yang paling umum dan merupakan ancaman besar bagi pengguna sistem publik online yang tidak memiliki perlindungan keamanan.
4. Worm adalah program replikasi diri yang menyebar melalui kerentanan jaringan. Berbeda dengan virus, worm tidak perlu terikat pada program lain dan tidak membutuhkan interaksi pengguna untuk berjalan.
5. Trapdoor adalah pintu masuk alternatif ke sistem. Jenis malware ini digunakan untuk merusak mekanisme keamanan yang telah ditanamkan ke dalam sistem. Programer biasanya membuatnya untuk menguji fungsi kode tertentu dalam waktu yang sangat singkat, sehingga seringkali tidak sengaja tertinggal.
6. Logic Bomb adalah jenis malware yang dapat menyerang sistem informasi dengan menggunakan logika penciptanya. biasanya terdiri dari program yang mempromosikan waktu dan peristiwa yang baik.

7. Rootkit adalah kumpulan program yang digunakan untuk mengubah operasi normal sistem operasi. bermaksud untuk menyembunyikan tindakan berbahaya yang dilakukannya. Mengganti utilitas biasa seperti kernel, netstat, ls, dan ps dengan kumpulan program mereka sendiri biasanya memungkinkan penyaringan aktivitas yang berbahaya sebelum tampilan hasilnya.
8. Bot dan Botnet. Bot adalah program komputer yang melakukan tindakan sesuai dengan perintah pembuatnya. Botnet adalah nama jaringan yang digunakan. Sering digunakan dalam lingkungan tertutup untuk menyelesaikan banyak tugas berbahaya melalui teknik remote controller.
9. Adware adalah program yang mengiklankan produk atau iklan, seringkali dengan iklan yang tidak layak dan sangat mengganggu.

b) Gambar berikut menunjukkan kategori malware yang paling banyak terdeteksi di Indonesia (Q1) tahun 2018. (Hayaty, 2020):



Gambar 2. Malware yang paling banyak terdeteksi

2.2.8 Serangan *Defacing*

Deface website merupakan tindakan yang dilakukan oleh seorang hacker atau peretas dengan tujuan untuk merusak atau mengubah tampilan homepage sebuah website. Tindakan ini sering dilakukan untuk menodai reputasi

organisasi atau individu, mengeksploitasi kerentanan dalam sistem keamanan, atau menyampaikan pesan politik atau ideologis.

Pelaku tindakan deface web seringkali mengincar website yang memiliki celah dan kelemahan dalam keamanan nya, peretas akan menggunakan berbagai teknik penyerangan untuk melakukan eksploitasi kelemahan ataupun celah keamanan website target sehingga peretas dapat memiliki akses ke dalam server target untuk melakukan berbagai macam tindakan kejahatan seperti deface website untuk berbagai macam hal seperti menunjukkan kelemahan keamanan, melakukan propaganda politik dan agama, menjual produk, untuk kesenangan pribadi (Aji, 2023).

2.2.9 Reverse Proxy

Reverse proxy adalah server yang berfungsi sebagai perantara antara pengguna dan server tujuan. Ketika pengguna mengirimkan permintaan, server ini akan menerima permintaan tersebut dan meneruskannya ke server tujuan. Setelah server tujuan memberikan respons, reverse proxy akan menyampaikan respons tersebut kembali kepada pengguna(PT Biznet Gio Nusantara, 2023).

2.2.10 Web Application Firewall (WAF)

Web Application Firewall (WAF) adalah jenis firewall yang melacak, menyaring, dan memblokir informasi yang dikirim oleh klien ke situs web atau aplikasi web. WAF menganalisis data yang masuk untuk memahami logika aplikasi web dan mendeteksi lalu lintas berbahaya yang berpotensi merusak situs web. Teknologi ini sering digunakan oleh perusahaan sebagai langkah perlindungan untuk mencegah eksploitasi, serangan malware, dan berbagai ancaman lainnya terhadap keamanan website. Manfaat menggunakan Web

Application Firewall (WAF) mencakup pencegahan berbagai jenis serangan, seperti(Indonesian Cloud, 2022):

1. **Cross-site scripting (XSS):** Serangan ini memungkinkan penyerang menyisipkan dan menjalankan skrip berbahaya di browser pengguna lain.
2. **Structured Query Language (SQL) Injection:** Serangan ini dapat memengaruhi aplikasi yang menggunakan database SQL, memungkinkan penyerang mengakses atau bahkan memodifikasi data sensitif.
3. **Web session hacking:** Penyerang dapat menyusup dengan mencuri ID sesi dan berpura-pura menjadi pengguna yang sah. Dalam kebanyakan kasus, ID sesi disimpan dalam cookie atau URL.
4. **Serangan Distributed Denial-of-Service (DDoS):** Serangan ini membanjiri jaringan dengan lalu lintas berlebih sehingga layanan menjadi tidak tersedia bagi pengguna. Baik firewall jaringan maupun WAF dapat membantu mengendalikan serangan ini, meskipun metode yang digunakan berbeda.

2.2.11 Acunetix Web Vulnerability

Acunetix Vulnerability Scanner merupakan salah satu program yang paling populer di pasar dalam mendeteksi celah keamanan SQL injection dan XSS. Ini adalah salah satu cara untuk melakukan evaluasi keamanan website menggunakan perangkat lunak dengan optimal dan dengan khusus dirancang untuk mengidentifikasi kerentanan yang ada pada suatu sistem (Zirwan, 2022).

Banyak pelanggan di bidang pemerintahan, militer, pendidikan, perbankan, keuangan, dan e-commerce, serta perusahaan besar dari berbagai negara, telah memilih Acunetix Web Vulnerability sebagai tools pilihan mereka. Daftar semua kerentanan yang ditemukan dalam target pemindaian selama pemindaian merupakan bagian penting dari hasil scan. Ini

dapat berupa peringatan web atau peringatan jaringan, tergantung pada jenis scan, dan tanda diklasifikasikan menurut empat tingkat keparahan (Al Fajar, 2020):

1. **High Risk Alert Level 3** : Kerentanan dianggap sebagai yang paling berbahaya, karena targetnya paling rentan terhadap hacking dan pencurian data.
2. **Medium Risk Alert Level 2** : Kerentanan disebabkan oleh misconfigurasi server dan sitecoding yang buruk memungkinkan intrusi dan gangguan server.
3. **Low Risk Alert Level** 1 : Kerentanan berasal dari kurangnya enkripsi lalu lintas data atau jalur direktori pengungkapan.
4. **Information Alert** : Ini adalah item yang menarik yang ditemukan selama pemeriksaan dan dianggap menarik. Misalnya, ini dapat menunjukkan alamat IP atau alamat email internal, pencocokan string pencarian yang ditemukan di database Google Hacking, atau informasi tentang layanan yang ditemukan selama pemeriksaan.

Standar report yang digunakan menggunakan OWASP Top 10 2021 dengan 10 kategori yang meliputi: (A01) Broken Access Control, (A02) Cryptographic Failures, (A03) Injection, (A04) Insecure Design, (A05) Security Misconfiguration, (A06) Vulnerable and Outdated Components, (A07) Identification and Authentication Failures, (A08) Software and Data Integrity Failures, (A09) Security Logging and Monitoring Failures, (A10) Server-Side Request Forgery.

2.2.12 OWASP ZAP

OWASP ZAP merupakan sebuah aplikasi yang digunakan dalam menemukan kerentanan pada suatu aplikasi web. OWASP ZAP menyediakan scanner secara otomatis. Dalam penggunaan OWASP ZAP, scanner dapat digunakan untuk menguji server, jaringan, perangkat dan endpoints. Saat

melakukan scanning, tahap-tahap yang dilakukan adalah Explore, Attack dan Report (Gibran dkk., 2023).

Zed Attack Proxy (ZAP) oleh Checkmarx adalah tools pengujian penetrasi sumber terbuka yang gratis. ZAP dirancang khusus untuk menguji aplikasi web dan bersifat fleksibel serta dapat diperluas. ZAP dikenal sebagai "proxy manipulator-in-the-middle". Ia berada di antara browser penguji dan aplikasi web sehingga ia dapat mencegat dan memeriksa pesan yang dikirim antara browser dan aplikasi web, mengubah konten jika diperlukan, lalu meneruskan paket tersebut ke tujuan. Ia dapat digunakan sebagai aplikasi mandiri, dan sebagai proses daemon (Zaproxy, 2024).

Gambar 3. ZAP berada di antara browser penguji dan aplikasi web

Gambar 4. ZAP dikonfigurasi untuk terhubung ke proksi

Berikut ini adalah 10 ancaman keamanan website berdasarkan OWASP Top 10 2021 (owasp.org, 2021):

1. A01:2021-Broken Access Control

Akses kontrol menetapkan aturan yang memastikan pengguna hanya dapat melakukan tindakan sesuai dengan izin yang diberikan. Kegagalan tersebut dapat mengakibatkan informasi yang diberikan tidak sah, perubahan atau penghapusan data, atau pelaksanaan fungsi bisnis di luar batas yang diperbolehkan untuk pengguna tersebut.

2. A02:2021-Cryptographic Failures

Kegagalan dalam penerapan kriptografi dapat menyebabkan data sensitif tidak terlindungi dengan baik, hal ini sering terjadi ketika algoritma enkripsi yang lemah digunakan atau jika proses implementasinya tidak sesuai standar keamanan. Akibatnya, informasi penting seperti kredensial login, data pribadi, atau rahasia bisnis dapat diakses oleh pihak tidak berwenang. Selain itu, kelemahan ini juga dapat dimanfaatkan oleh peretas untuk menyerang sistem, mencuri data, atau bahkan mengendalikan perangkat yang terhubung.

3. A03:2021-Injection

Serangan injeksi terjadi ketika aplikasi gagal memvalidasi, memfilter, atau membersihkan data yang dimasukkan oleh pengguna. Akibatnya, data yang berisi perintah berbahaya dapat disisipkan ke dalam query dinamis, perintah, atau prosedur tersimpan. SQL, NoSQL, perintah OS, peta hubungan objek (ORM), LDAP, dan injeksi pada bahasa ekspresi (EL) atau pustaka navigasi grafik objek adalah beberapa jenis injeksi yang paling umum. Meninjau kode sumber aplikasi adalah cara terbaik untuk mengidentifikasi risiko ini. Sangat disarankan untuk melakukan pengujian otomatis pada semua parameter, header, URL, cookie, dan data input seperti JSON, SOAP, dan XML.

4. A04:2021-Insecure Design

Kategori baru yang dihadirkan pada tahun 2021 ini berfokus pada risiko yang timbul akibat kelemahan dalam desain dan arsitektur sistem. Hal ini mencakup penggunaan metode pemodelan ancaman untuk mengidentifikasi risiko sejak awal pengembangan. Selain itu, penerapan pola desain yang lebih aman dan penggunaan referensi arsitektur yang tepat menjadi langkah penting untuk mengurangi kerentanan sistem. Dengan fokus pada elemen-elemen ini, kategori ini bertujuan untuk membantu pengembang menciptakan sistem yang lebih kuat dan tahan terhadap berbagai ancaman.

5. A05:2021-Security Misconfiguration

Beberapa kesalahan konfigurasi dapat terjadi akibat perubahan pada perangkat lunak dengan pengaturan yang kompleks, kurangnya pertahanan yang memadai atau langkah-langkah *security hardening* di seluruh lapisan aplikasi, atau kesalahan dalam mengatur izin pada layanan cloud. Tanpa adanya proses konfigurasi keamanan aplikasi yang konsisten dan terintegrasi, sistem akan berada dalam risiko yang tinggi.

6. A06:2021-Vulnerable and Outdated Components

Komponen yang rentan merupakan tantangan umum yang sulit untuk diuji dan dinilai risikonya. Ketika perangkat lunak memiliki kerentanan, tidak lagi didukung, atau ketinggalan zaman dapat meningkatkan risiko. Situasi ini sering terjadi di lingkungan di mana proses *patching* atau proses memperbarui perangkat lunak dilakukan secara bulanan atau triwulanan dan berada di bawah kendali perubahan. Akibatnya, organisasi dapat tetap terekspos terhadap kerentanan yang seharusnya sudah diperbaiki, selama berhari-hari bahkan berbulan-bulan, tanpa perlindungan yang memadai.

7. A07:2021-Identification and Authentication Failures

Kegagalan dalam identifikasi merupakan bagian penting dalam daftar Top 10, namun peningkatan aksesibilitas framework yang distandarisasi dapat memberikan bantuan. Verifikasi identitas pengguna, otentikasi, dan pengelolaan sesi sangat krusial untuk melindungi dari serangan yang berkaitan dengan otentikasi.

8. A08:2021-Software and Data Integrity Failures

Kategori baru untuk tahun 2021 ini berfokus pada pembuatan asumsi mengenai pembaruan perangkat lunak, data penting, dan pipeline CI/CD tanpa memverifikasi integritasnya. Ketika infrastruktur dan kode tidak mencegah pelanggaran integritas, integritas data dan perangkat lunak menjadi lemah. Sebagai contoh, deserialisasi yang tidak aman dapat terjadi pada data atau objek yang telah dicoding atau di serialisasi dalam struktur yang dapat dilihat dan diubah oleh penyerang.

9. A09:2021-Security Logging and Monitoring Failures

Kategori ini diperluas untuk mencakup berbagai jenis kegagalan yang sulit diuji dan kurang diwakili dalam data CVE/CVSS. Kegagalan tersebut dapat mempengaruhi visibilitas sistem, pemberian peringatan insiden, serta investigasi forensik. Tujuan dari kategori ini adalah untuk mendukung proses deteksi, peningkatan pengawasan, dan respons terhadap serangan yang sedang berlangsung. Tanpa log dan pemantauan yang cukup, aktivitas penjebolan tidak dapat diidentifikasi. Kegagalan dalam pencatatan, pemantauan, deteksi, dan respons aktif sering terjadi.

10. A10:2021-Server-Side Request Forgery

Kategori ini menggambarkan insiden dengan tingkat kejadian yang relatif rendah tetapi memiliki cakupan pengujian yang baik serta nilai dampak dan potensi eksploitasi yang tinggi. SSRF (Server-Side Request Forgery) terjadi ketika aplikasi web mengakses sumber daya jarak jauh tanpa memvalidasi URL yang diberikan pengguna. Hal ini memungkinkan penyerang memanipulasi aplikasi untuk mengirimkan permintaan yang disesuaikan ke lokasi yang tidak diinginkan, bahkan jika lokasi tersebut dilindungi oleh firewall, VPN, atau mekanisme kontrol akses lainnya seperti ACL (Access Control List). Karena aplikasi web modern sering menawarkan fitur yang memudahkan pengguna, proses pengambilan URL menjadi umum, sehingga insiden SSRF semakin sering terjadi. Selain itu, keparahan SSRF meningkat karena layanan berbasis cloud dan kompleksitas arsitektur cloud yang terus berkembang.