BAB II
LANDASAN TEORI

2.1 Tinjauan Pustaka

Tinjauan pustaka dalam penelitian ini menganalisa dari beberapa tugas akhir maupun jurnal untuk menguji kerentanan keamanan suatu *website* di dalamnya, antara lain:

Dewanto (2018), untuk menguji kerentanan domain dan subdomain UII.AC.ID dari serangan pihak yang tidak bertanggung jawab maka diperlukan analisis terhadap system dan jaringan pada UII dari persepektif luar atau jaringan *public*. Melalui analisis ini, dapat diketahui letak kerentanan dari sistem yang ada. Salah satu metodenya adalah *penetration testing*, berdasarkan pada metode *Open Web Application Security Project* Top 10 (OWASP 10) tahun 2013 yang berfokus pada pengumpulan informasi dan pengujian system yang ada. Dalam penelitian ini, yang dilakukan oleh Adetya Putra Dewanto yang berjudul *Penetration Testing* Pada Domain Uii.ac.id menggunakan OWASP10 disebutkan bahwa OWASP bisa dijadikan sebagai dasar dalam pengujian keamanan terhadap *web application*. Dalam penelitian tersebut target yang diserang adalah *web* Uii.ac.id milik Universitas Islam Indonesia.

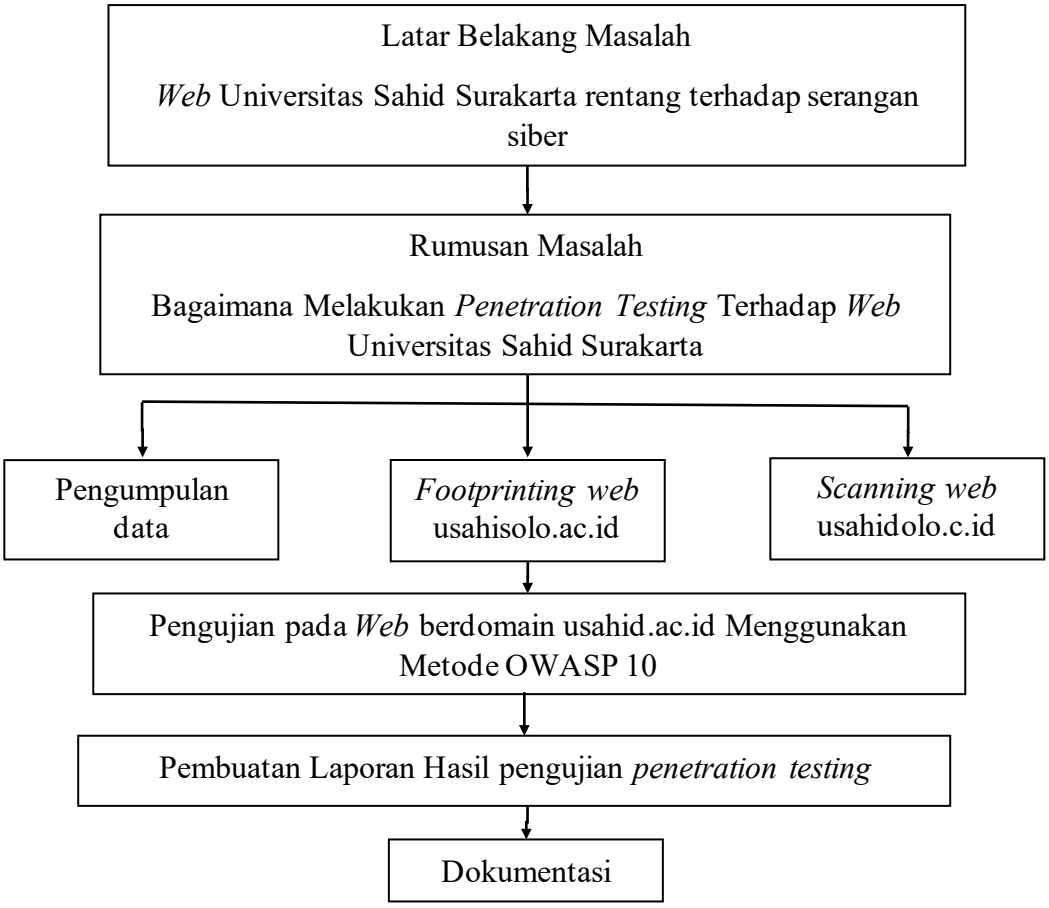
Iqbaludin (2018), pengujian Celah Keamanan Pada *Website Captive Portal* Dengan Menerapkan *Penetration Testing* (Studi Kasus: Teknik Informatika Universitas Pasundan). Demi mencegah rentannya keamanan *website captive portal* maka dilakukan upaya pengujian atau *penetration testing* dengan berdasar celah keamanan yang diperoleh dan berdasarkan OWASP Top 10 – 2017. Nantinya akan menghasilkan informasi berupa hasil pengujian pada *website captive portal* yang berguna sebagai bahan acuan pengembangan dalam melakukan keamanan.

Subagja (2019), *penetration Testing* Terhadap *Website* Asosiasi Pekerja Professional Informasi Sekolah Indonesia. Untuk mengetahui kondisi dan pengukuran tingkat kerentananan system informasi pada *website* APISI, maka dilakukan pengujian kerentanan berdasar metodologi *Zero Entry Hacking* (ZEH)

tanpa social *enginering* lalu kerentanan tersebut dianalisis tingkat kerentananya dengan kalkulator berbasis *web Common Vulnerability Scoring System (CVSS)* v3.1. Untuk tingkat kerentanan yang terjadi merupakan akibat dari kegagalan dalam mempertahankan keamanan system informasi pada *website* APISI dan perlu segera diperbaiki. Penerapan dari rekomendasi akan diserahkan penuh pada kewenangan instansi terkait yaitu APISI.

2.2 Kerangka Pemikiran

Berikut ini adalah tahapan kerangka pemikiran yang akan dijalankan oleh penulis dalam proses pengujian *web* Universitas Sahid Surakarta seperti pada Gambar 2.1.



Gambar 2.1 Kerangka Pemikiran

Uraian dari kerangka berfikir sebagai berikut :

1. Latar Belakang Masalah

Latar belakang masalah pada tugas akhir ini adalah Universitas Sahid Surakarta belum pernah melakukan *self test* atau pengujian mandiri terhadap *website* mereka sendiri jadi dapat menyebabkan kemungkinan kerentanan keamanan *website* yang ada.

2. Rumusan Masalah

Rumusan masalah pada tugas akhir ini adalah bagaimana caranya melakukan *penetration testing* terhadap *website* Universitas Sahid Surakarta agar dapat terhindar dari serangan *hacker*, salah satu metode *self test* ini adalah *penetration testing* yang mempunyai kesamaan aktivitas dengan *hacking* namun dilakukan dengan aman dan legal.

3. Pengumpulan Data

Pengumpulan data dalam penelitian ini adalah melakukan langkah-langkah untuk pengujian penetration testing pada *website* Universitas Sahid Surakarta mulai dari pengumpulan data secara *offline* atau secara langsung pada pengelola *web* tersebut agar diberikan data atau informasi yang lebih lengkap, mencari segala informasi yang berkaitan dengan sistem tersebut untuk dilakukan analisis dan menentukan metode guna melakukan penyerangan, *scanning website* adalah dimana penyerang mengumpulkan celah keamanan lain yang berhubungan dengan jaringan korban secara lebih spesifik.

4. Analisis pengujian Sistem

Analisis pengujian sistem dalam penelitian ini adalah melakukan pengujian *penetration testing* menggunakan metode OWASP 10 sebagai daftar pedoman celah keamanan yang sering ditemukan dan dapat mengancam keamanan suatu *website*.

5. Laporan Hasil Pengujian

Proses penjabaran dan penjelasan hasil dari pengujian yang telah dilakukan menggunakan metode OWASP 10 disertakan solusi menurut metode pengujian.

6. Dokumentasi

Dokumentasi dalam penelitian ini adalah proses pengambilan dokumentasi setelah pengujian *penetration testing* sebagai bukti-bukti kerentanan sistem yang di ujikan.

2.3 Teori Pendukung

2.3.1 Keamanan Informasi

Keamanan informasi merupakan sebuah praktik untuk melindungi informasi dari akses yang tidak valid dalam penggunaan, pengungkapan, modifikasi, inspeksi, dan menghapus data (Fauzan & Rijayanti, 2018). Keamanan informasi memiliki tiga prinsip dasar yaitu: *Confidentiality*, *Integrity* dan *Availability*.

a. *Confidentiality* merupakan sebuah property, di mana informasi yang tidak diungkapkan kepada user yang tidak sah (Fauzan & Rijayanti, 2018).

b. *Integrity* adalah untuk menjaga, menjaminkan kelengkapan data tersebut tidak dapat dimodifikasikan secara tidak sah (Fauzan & Rijayanti, 2018).

c. *Availability* merupakan ketersediaan informasi yang dapat melayani tujuan dan tersedia jika dibutuhkan. Di mana sistem komputasi yang digunakan untuk menyimpan data dan memproses informasi (Fauzan & Rijayanti, 2018).

2.3.2 Penetration Testing

Merupakan teknik untuk menemukan kerentanan atau celah keamanan yang ada di halaman *website* untuk dapat membantu mengesampingkan akses ilegal ke dalam halaman *website* dan database. Dalam pengujian *penetration testing* pada halaman *web* terus dapat menjadi masalah signifikan, karena semakin banyak fitur pada aplikasi web maka semakin lama dalam pengujian tersebut (Bin Ibrahim & Kant, 2018).

2.3.3 *Website*

Website merupakan sebuah kumpulan halaman yang diawali dengan halaman muka yang berisikan informasi, iklan, serta program aplikasi (Septiyanto, 2017).

2.3.4 *Open Web Application Security Project* (OWASP)

OWASP (*Open Web Application Security Project*) adalah komunitas terbuka yang mendedikasikan untuk membuat sebuah organisasi yang bertujuan untuk mengembangkan, membeli, dan memelihara aplikasi yang terpercaya. Di OWASP pengunjung akan menemukan semua gratis dan terbuka. Seluruh *tools*, dokumen, forum, dan cabang OWASP bebas dan terbuka bagi semua orang yang tertarik memperbaiki aplikasi keamanan. OWASP mendukung pendekatan keamanan aplikasi sebagai masalah perseorangan, proses, dan masalah teknologi karena pendekatan paling efektif terhadap keamanan aplikasi membutuhkan perbaikan diseluruh area. OWASP adalah jenis organisasi baru yang bebas dari tekanan komersial sehingga memungkinkan untuk memberikan informasi terkait keamanan aplikasi yang tidak bias, praktis, dan efektif biaya. OWASP tidak terafiliasi dengan perusahaan teknologi manapun. Meskipun OWASP mendukung penggunaan teknologi keamanan komersial. Serupa dengan banyak proyek *software open-source*, OWASP menghasilkan beragam jenis materi dengan cara kolaborasi dan terbuka. Yayasan OWASP merupakan lembaga non-profit yang memastikan kesuksesan jangka panjang proyek. Hampir semua yang terasosiasi dengan OWASP adalah sukarelawan (OWASP, 2017).

2.3.5 OWASP TOP 10

OWASP Top 10 atau yang biasa disebut OWASP10 adalah sebuah daftar yang dirilis oleh komunitas OWASP yang berisikan 10 daftar teratas celah keamanan yang dapat mengancam keamanan suatu website daftar ini terus berkembang dan berubah-ubah mengikuti perkembangan teknologi website yang terus berkembang. OWASP Top 10 pertama kali dirilis tahun 2003 lalu update minor pada tahun 2004, 2007, 2010, 2013, dan yang terakhir 2017 (OWASP, 2017). OWASP Top 10 sendiri dibuat dengan tujuan untuk meningkatkan kesadaran tentang keamanan aplikasi dengan mengidentifikasi lebih dini beberapa

risiko celah keamanan yang sering muncul atau ditemui dalam banyak kasus. Berikut 10 daftar resiko paling berbahaya oleh OWASP tahun 2017:

1) SQL *Injection* adalah serangan di mana mengandung perintah SQL ke dalam kode SQL yang memungkinkan merusak *database*, identitas palsu, dan menjadi *administrator* dari database tersebut (Sagar, 2018).

2) *Broken Authentication* adalah mengirim informasi sesi dan data manajemen akun (pembuatan aku, ubah kata sandi, pulihkan kata sandi) yang melalui *website* yang terkena kode sesi dan kredensial yang tidak dilindungi (Dehalwar, Kalam, Kolhe, & Zayegh, 2018).

3) *Sensitive Data Exposure* adalah jenis kerentanan keamanan di mana *aplikasi web* gagal dalam melindungi data rahasia dari suatu organisasi. Data sensitif termasuk dalam informasi pribadi, informasi kesehatan, informasi keuangan yang dapat digunakan dengan baik dalam serangan *phishing*, penipuan kartu, dan penipuan *email* (Sagar, 2018).

4) *XML External Entities* (XEE) yang bersifat kerentanan esoterik dibandingkan dengan serangan aplikasi *web* lainnya. Penyerangan ini mengirim input data XML. Parser XML yang dikonfigurasi dengan buruk yang berisi referensi ke entitas eksternal. Dalam beberapa kasus, penyerang menjalankan eksekusi kode jarak jauh untuk mengekstrak informasi rahasia, detail pemindaian *port* (Dehalwar et al., 2018).

5) *Broken Access Control* yaitu serangan yang terjadi ketika pembatasan pada hak akses tidak di lindungi dengan benar yang memberikan penyerangan mendapatkan kesempatan untuk mengeksploitasi kelemahan ini dan karenanya dapat mencapai akses ke fungsi dari akun orang lain atau informasi pribadi dari suatu organisasi (Sagar, 2018).

6) *Security Misconfiguration* yaitu serangan yang bersifat yang terjadi di karena kan kesalahan dalam konfigurasi keamanan aplikasi *website* atau *server*. Kesalahan konfigurasi kecil dapat membuat data tersebut dipertaruhkan. Contohnya perangkat lunak yang ketinggalan versi dan

mengaktifkan atau menonaktifkan fitur yang tidak diinginkan tanpa mengetahui fungsinya (Sagar, 2018).

7) *Cross-Site Scripting* (XSS) adalah serangan injeksi kode di sisi klien di mana penyerang mengeksekusi skrip / sebuah struktur kode khusus yang dikirimkan ke situs *web* yang sah jika sisi *server* memiliki validasi tersebut maka struktur kode tersebut tidak dijalankan (Sagar, 2018).

8) *Insecure Deserialzation* adalah proses pengambilan data mentah dari file atau soket jaringan untuk merekonstruksi model objek. Dari data yang tidak dipercayai tidak dapat deserialisasi tanpa cukup memverifikasi bahwa data tersebut dihasilkan valid / asli. Data yang di serial dapat digunakan dengan mudah, tetapi data yang tidak disterilisasi dapat mudah dimodifikasi oleh penyerang jika tidak dilindungi oleh fungsi enkripsi (Dehalwar et al., 2018).

9) *Using Components With Know Vulnerabilities* yaitu serangan yang menggunakan sumber terbuka atau *open-source libraries* yang memasang skrip kode serupa dipasang pada server dapat membuat masalah. Dengan menetapkan sebuah kebijakan keamanan yang mengambil atau mengakses semua informasi kesalahan dari *website* dapat meminimalkan risiko menggunakan kerentanan yang diketahui dari serangan di alternatif *website* (Dehalwar et al., 2018).

10) *Insufficient Logging and Monitoring* yang tidak digunakan jika *software* tidak dikonfigurasi dengan benar pada jaringan, maka aktivitas dari jaringan tidak dapat di monitor untuk mengekstrak data tersebut. Selain itu, memerlukan waktu lama untuk diperlukan, terkadang dibutuhkan 200 hari untuk mendeteksi serangan. Dan melaporkan masalah pada waktu yang tepat maka dapat menyediakan solusi sehingga dapat mencegah yang akan datang serangan (Dehalwar et al., 2018).

2.3.6 Alerts

Alerts adalah lembar jendela setelah proses *scanning* selesai semuanya, yang berisi hasil proses scanning dan terkumpul secara urut pada bagian *Alerts* di dalam *tools* OWASPZap (OWASP10 2017).

2.3.7 Scanning Tools

Scanning adalah tahapan di mana penyerang mengumpulkan segala informasi yang berhubungan dengan jaringan korban secara lebih spesifik. *Scanning* juga dapat diartikan sebagai bentuk pendeteksian sistem yang masih hidup dan dapat diakses melalui internet dan apa saja *service* yang ditawarkan. Tahap ini merupakan resiko tinggi, jika penyerang dapat menemukan kelemahan dari sebuah sistem, maka penyerang dapat mengeksploitasi jaringan tersebut. Terdapat banyak cara dan *tools* dalam melakukan proses *scanning* dapat dilakukan secara manual atau otomatis menggunakan *tools* yang banyak bersebaran yang dapat memudahkan penyerang dalam melakukan *scanning*. Beberapa contoh *tools* yang dapat digunakan antara lain The Harvester, Nmap, dan Masscan.

2.3.8 Web Analisys Scanning

Web analisys scanning adalah tahap dimana seorang penyerang melakukan analisis mendalam terhadap *web* target yang akan diserang (*attack*). Terdapat beberapa cara dalam melakukan *web analisys scanning* antara lain dengan cara manual menggunakan *browser* atau dengan menggunakan *tools vulnerability scanner* yang banyak bersebaran. Beberapa contoh *tools vulnerability scanner* antara lain WPScan dan OWASPZap.

1) WPScan

WPScan merupakan salah satu *tools vulnerability scanner* yang digunakan untuk melihat dan mendeteksi kelamahan pada *web* yang bertipe *WordPress* (Wpscan, 2015). Celah keamanan yang biyasanya terdapat dalam web yang bertipe *WordPress* biasanya celah keamanan ditemukan dalam plugin atau *theme* yang digunakan oleh *user* pada *web WordPress* mereka. Fungsi WPScan Antara lain:

a. *List Plugin.*

b. *List Theme.*

c. *Brute Force Weak Password* dan *Username.*

d. *Listing Direktori.*

e. Melihat kemungkinan *vulnerabilities*.

2) OWASPZap

OWASPZap adalah sebuah *tools vulnerabilities scanner* yang dibuat oleh organisasi OWASP *tools* inI adalah suatu proyek dari OWASP yang paling aktif karena terus dikembangkan, *tools* ini bersifat *opensource* sehingga siapa saja juga bisa mengembangkan *tools* ini. Fitur yang ada dalam OWASPZap antara lain *Intercepting Proxy,Active and Passive Scanners, spider scan, report Generation, Brute Force(using OWASP dirbuster code), Fuzzing(using fuzzdb & OWASP JBrosfuzz), Extensibility,Auto tagging,Port scanner, Parameter analysis, Smart card support, Session comparison, invoke external apps, Api+headless mode, Dynamis SSL Certificates, Anti CSRF token handling* (Owaspzap, 2016). Dengan banyaknya fitur yang terdapat dalam OWASPZap sehingga memudahkan dalam melakukan scanner terhadap suatu *web,* selain itu OWASPZap sangat mudah digunakan sehingga memudahkan pemula dalam melakukan *scanning* terhadap *web*.

2.3.9 WHOIS Domain

WHOIS berisikan informasi domain. Dengan cek domain WHOIS, Anda bisa mengetahui ketersediaan domain, tanggal kedaluwarsa domain, serta identitas pemilk domain seperti nama, informasi kontak, alamat, nomor telepon, hingga alamat email (hostinger.co.id, 2021).