BAB II
TINJAUAN PUSTAKA

2.1 Tinjauan Penelitian Terdahulu

Ialah analisis terhadap beragam penelitian terdahulu berhubungan berkenaan masalah yang sedang ditelaah. Temuan dari penelitian sebelumnya dijadikan sebagai referensi untuk menetapkan topik, arah, dan tujuan penelitian. Kajian ini bertujuan untuk menyediakan bahan perbandingan serta acuan, sekaligus menghindari anggapan adanya kesamaan dengan penelitian yang dilakukan. Rincian perbadingan penelitian terdahulu yang dipakai menjadi rujukan oleh peneliti diantaranya :

Tabel 2. 1 Ringkasan Penelitian Terdahulu

No	Judul	Penulis	Metode	Hasil
1	Analisis Deteksi *Vulnerability* Pada Webserver Open Jurnal System Menggunakan OWASP Scanner	Yunanri. W, dkk	Metode *penetration testing* yang menggunakan OWASP	Terdapat hasil keretanan 40 level high, 1084 level medium, 356
2	Analisis Keamanan Sistem Informasi Akademik Menggunakan *Open Web Application Security Project Framework*	Muh. Amirul Mu'min, dkk	Metode *penetration testing* dan OWASP	12 kerentanan dengan 4 kerentanan pada level medium yakni *Absence of Anti-CSRF Tokens, Cross-Domain Misconfiguration, Missing Anti-clickjacking Header,* dan *Vulnerable JS Library,* 6 level low yakni *Cookie Without Secure Flag, Cookie without SameSite Attribute, Cross-Domain JavaScript Source File Inclusion, Server Leaks Information via "X-*

				Powered-By" HTTP Response Header Field(s), Timestamp Disclosure – Unix, dan *X-Content-Type-Options Header Missing,* dan 2 pada level informational yakni *Content-Type Header Missing dan Information Disclosure - Suspicious Comments*
3	ANALISIS KUALITAS KEAMANAN SISTEM INFORMASI E-OFFICE BERBASIS WEBSITE PADA STMIK ROSMA DENGAN MENGGUNAKAN OWASP TOP 10	Yudiana, dkk	Metode OWASP TOP 10	Terdapat 3 kerentanan dari Acunetix antara lain HTML form without CSRF protection level medium, Clickjacking: X-Frame-Options header missing level low, dan Password type input with auto-complete enabled level informatif. Sedangkan OWASP-ZAP terdapat 13 kerentanan.
4	*Penetration Testing* pada Website Universitas ARS Menggunakan *Open Web Application Security Project* (OWASP)	Syarif Hidayatulloh, dkk	Metode *Penetration Testing*	Hasil dipindai adalah 13 kerentanan. Dari 13 kerentanan tersebut ada 1 kerentanan yang berada pada tingkat ancaman yang sedang dan 12 berada pada tingkat ancaman yang rendah

Seperti tersedia dalam tabel 2.1, melalui penelitian yang telah dilaksanakan memakai teknik yang sama sebagai acuan atau rujukan terdapat beberapa perbedaan pada hasil yang diberikan. Walaupun ada beberapa perbedaan dalam beberapa proses, hasil yang didapat di akhir masih tetap sama yaitu apakah *website* yang diuji memiliki sebuah celah keamanan yang dapat disusupi oleh *hacker* dan

seberapa rentan celah keamanan itu. Oleh karena itu, penulis menggunakan metode yang sama dengan beberapa rujukan dari penelitian yang sudah dirangkum pada tabel di atas. Pemilihan metode vulnerability Asessment ini seperti beberapa rujukan pada jurnal pada Table 1 yaitu untuk melakukan serangkain proses untuk melakukan sebuah audit atau evaluasi kerentanan yang ada pada sebuah sistem dan memberikan sebuah informasi baru bagi pihak perusahaan bahwa terdapat celah keamanan bisa disalahgunakan beragam pihak yang tidak baik.

2.2 Sistem Manajemen Tugas Akhir (Simanta)

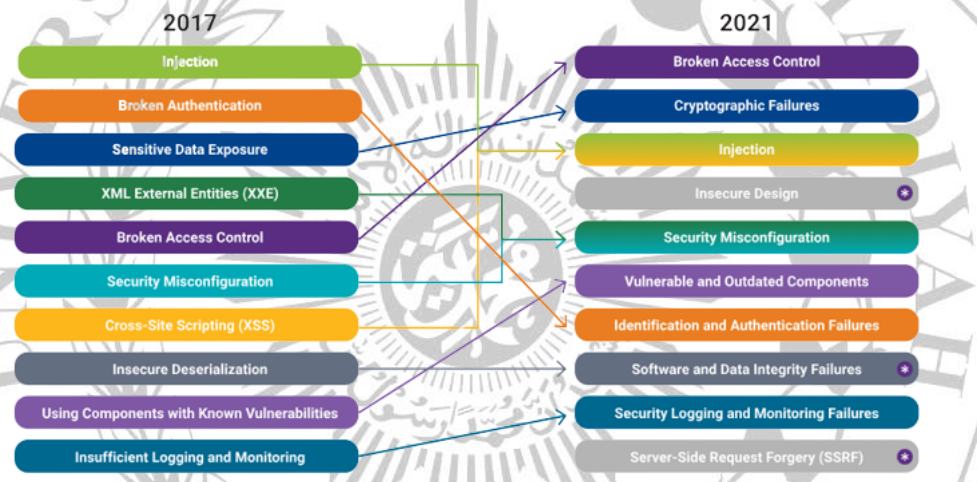
Sistem Manajemen Tugas Akhir merupakan sebuah sistem yang dapat membantu proses pelaksanaan kegiatan penyusunan Tugas Akhir (TA) oleh mahasiswa diawali prosedur mendaftar hingga proses kelulusan atau yudisium. Dalam *website* Sistem Manajemen Tugas Akhir terdapat fitur tersedia yang digunakan mahasiswa antara lain seperti fitur *menu dashboard* sebagai penampil ringkasan status tugas akhir, progress tugas akhir serta pengumuman, terdapat juga fitur ketersediaan dosen untuk mahasiswa mengetahui kapan dosen tersedia untuk bimbingan. Adapun *menu* lainnya seperti seminar proposal, SK tugas akhir, Ujian TA, dan yudisium.

2.3 Vulnerability

Vulnerability merupakan sesuatu yang mempengaruhi sebuah data yang mengancam aset komponen *confidentiality, integrity, availability* atau bisa disebut (CIA) [5]. Vulnerability bukan *software bugs* atau kekurangan kemanan jaringan saja. Tetapi, kekurangan karyawan tanpa mendapatkan pelatihan, kurangnya dokumentasi, atau prosedur belum diterapkan dengan benar dapat menjadi masalah. *Vulnerability* dapat dikelompokkan menjadi tiga kategori, yakni kekurangan dalam system tersebut, akses yang mengarah pada kerentanan sistem, juga keahlian *hacker* teruntuk melancarkan serangan. Kecacatan dalam sebuah sistem disebabkan oleh kelemahan dari *website* tersebut [14]. Beberapa *tools* yang digunakan antara lain *dirsearch, nmap, dig, whatweb.*

2.4 OWASP (Open Web Application Security Project) Top 10 2017

Open Web Application Security Project ialah daftar tersusun dari komunitas OWASP, memuat sepuluh celah keamanan utama yang bisa berisiko pada keamanan *website*. Yang kerap diperbarui serta disesuaikan oleh kemajuan teknologi. [1]. Berlandaskan *website* resmi OWASP, menjelaskan "komunitas terbuka yang diperuntukkan supaya memungkinkannya organisasi menignkatkan, beli, dan memelihara program yang bisa dipercayai" [11]. Pada tahun 2021 sendiri *framework* ini mengalami beberapa perubahan dari kerentanan yang ada pada 2017 dalam gambar sebagai berikut.



Gambar 2. 1 *Top 10 Vulnerability Web by OWASP*

Pada Gambar 2.1, terdapat sepuluh kerentanan yang ada dalam sebuah aplikasi beserta penjelasannya :

1. *A1-Broken Access Control*

   Kerentanan ini dapat mengakses fungsionalitas data yang seharusnya tidak dapat di akses, sehingga peretas bisa mengakses data dengan hak akses tinggi, semisal penambahan pengguna, serta melihat informasi sensitif yang seharusnya hanya bisa diakses oleh admin [8].

2. *A2-Cryptographic Failures*

Kerentanan ini bisa terjadi karena kekeliruan saat mengimplementasikan metode kriptografi, semisal lemahnya pemakaian algoritma atau mengatur kata sandi yang tidak baik, dapat menjadikan data sensitive lebih rentan dieksploitasi serta dicuri peretas..

3. *A3-Injection*

   Kerentanan ini bisa terjadi ketika penyerang memasukkan perintah atau memasukan kode berbahaya berupa *SQL injecton* atau *XSS* ke aplikasi web melalui fitur web seperti kotak pencarian, kolom komentar, atau bahkan URL parameter [10].

4. *A4-Insecure Design*

   Kerentanan ini bisa terjadi ketika terdapat kelemahan keamanan dalam desain atau arsitektur aplikasi, penyerang dapat memanfaatkan celah ini untuk menyerang sistem, mencuri data sensitif, kurangnya pertimbangan keamanan selama proses pengembangan, atau merusak sistem.

5. *A5-Security Misconfiguration*

   Kerentanan ini bisa terjadi ketika konfigurasi keamanan tidak dikelola dengan benar pada sistem aplikasi. Penyerang dapat memanfaatkan konfigurasi yang salah untuk mangakses sumber daya yang tidak diizinkan atau melakukan tindakan yang tidak diinginkan.

6. *A6-Vulnerable and Outdated Component*

   Kerentanan ini bisa terjadi akibat pemakaian perangkat lunak yang telah lama atau diketahui rentan diserang. Peretas selalu mencari peluang salah satunya muncul ketika terdapat kelemahan keamanan dalam desain atau arsitektur aplikasi yang memungkinkan penyerang untuk menyerang sistem dengan memanfaatkan celah tersebut.

7. *A7-Identification and Authentication Failures*

   Kerentanan ini bisa terjadi ketika kegagalan dalam proses identifikasi dan otentikasi pengguna seperti tidak menerapkan otentikasi dua faktor, lemahnya pemakaian kata sandi, atau kesalahan Ketika prosedur login pengguna.

8. *A8-Software and Data Integrity Failures*

Kerentanan ini bisa dialami ketika memungkinkan peretas teruntuk merombak atau menghancurkan data atau perangkat lunak yang dipakai oleh aplikasi web. Penggunaan input dan output data yang dihasilkan sering dimanfaatkan oleh penyerang untuk membuat pengguna percaya bahwa hasil unduhan yang mereka terima berasal dari situs web yang sah.

9. *A9-Security Logging and Monitoring Failures*

   Kerentanan ini dialami saat gagal mengawasi serta pencatatan kegaitan keamanan aplikasi web dapat mengakibatkan penundaan dalam menemukan serangan atau sukar teruntuk melacak insiden yang dialami.

10. *A10-Server-Side Request Forgery*

    Kerentanan ini terjadi ketika serangan dari penyerang memanipulasi *server* teruntuk mengirimkan permintaan ke *database* yang dilarang atau tidak aman.

2.5 Penetration Testing

Penetration Testing merupakan metode untuk menilai keamanan perangkat atau sistem komputer dengan mensimulasikan serangan siber yang menyerupai kondisi nyata. Metode ini dilakukan oleh seorang profesional yang dikenal sebagai pentester. Pentester bertugas mengidentifikasi celah atau kerentanan dalam sistem keamanan. Setelah celah ditemukan, mereka akan mencoba mengeksploitasinya untuk mendapatkan akses ke dalam sistem. [9]. Sesudahnya, dilaksanakan uji coba berperan menjadi *hacker* yang ingin mencuri atau mengotrol data pribadi.

Terdapat tiga jenis metode *Penetration Testing* yang biasa digunakan, yakni *white-box testing* merupakan metode yang dilaksanakan dengan memberikan informasi lengkap tentang *server* yang ingin diuji coba [1]. *Black-box testing* sebaliknya dari *white-box testing* informasi yang diberikan sangat sedikit mengenai target. *Gray-box testing* ialah penggabungan keduanya dilakukan melalui memberikan informasi terbatas mengenai sistem yang ingin diuji, metode ini lebih berfokus terhadap kerentanan individual yang terdapat disekeliling sistem [1].