BAB II
LANDASAN TEORI

2.1 Keamanan Informasi

Keamanan informasi merupakan upaya untuk melindungi asset dari bahaya yang dapat menggaggu ketersediaan (*availability*), kerahasiaan (*confidentiality)*, dan integritas (*integrity*) ketiga elemen ini dikenal sebagai prinsip dasar keamanan informasi atau CIA Triad. Dalam konteks aplikasi web, keamanan informasi menjadi sangat penting karena banyaknya transaksi dan penyimpanan data sensitif yang dilakukan melalui jaringan internet (Putra et al., 2020). Penggunaan teknologi informasi, keamanan dan kewaspadaan terhadap resiko bocornya informasi menjadi prioritas utama. Keamanan informasi tidak hanya menjaga data informasi tetap rahasia, tetapi juga memastikan bahwa informasi yang ada tersedia dan dibutuhkan (Mantra et al., 2020).

Beberapa bahaya dapat muncul dalam hal keamanan informasi. Secara umum, ancaman yang dapat terjadi pada sebuah situs web termasuk salah satunya *SQL injection*. Penilaian resiko terhadap keamanan informasi yang dapat menghambat dan merugikan suatu organisasi atau lembaga diperlukan untuk mengantisipasi ancaman dan serangan siber. Penilaian resiko dapat digunakan sebagai referensi untuk memperbaiki atau meminimalkan masalah serta untuk menghindari hal-hal yang tidak diinginkan (Jonny et al., 2021).

Keamanan informasi merupakan komponen penting dalam menjaga asset informasi suatu organisasi. Menurut Whitman & Mattord, 2013 Dalam jurnal (Firzah A Basyarahil, Hanim Maria Astuti, 2017) berikut merupakan beberapa jenis keamanan data:

a. *Operational security* yaitu keamanan yang berfokus pada strategi untuk melindungi karyawan, asset fisik, dan lokasi kerja dari beberapa bahaya, seperti kebakaran, akses tanpa izin, dan bencana alam.
b. *Personal security* merupakan keamanan yang menggabungkan keamanan fisik untuk melindungi individu di perusahaan dalam organisasi.
c. *Physical security* adalah keamanan yang bertujuan untuk memastikan kekuatan perusahaan agar tidak terganggu.
d. *Communications security* yaitu keamanan dengan memastikan bahwa media komunikasi, teknologi komunikasi, dan konten, tetap aman juga kemampuan untuk menggunakan alat tersebut untuk mencapai tujuan perusahaan.
e. *Network security* merupakan keamanan yang menekankan pada perlindungan alat dan organisasi jaringan, jaringan dan konten, serta kemampuan utnuk menggunakan jaringan untuk memenuhi kebutuhan komunikasi data organisasi.

2.2 Keamanan Web

Keamanan suatu web merujuk pada praktik, kebijakan dan teknologi yang digunakan untuk melindungi aplikasi web dari ancaman siber. Banyak pengembang masih berfokus pada estetika dan fungsionalitas tanpa mempertimbangkan aspek keamanan. Padahal, celah keamanan seperti *SQL Injection* dapat dimanfaatkan penyerang untuk mengakses atau memodifiaksi data penting. Pentingnya keamanan web sebagai lapisan perlindungan terhadap berbagai bentuk serangan termasuk *SQL Inejction*. Salah satu praktik terbaik adalah penerapan validasi input, penggunaan header keamanan, dan penerapan kebijakan CSP (Content Security Policy). (Nurelasari & Gumilang Al Farabi, 2024)

Tujuan dari proses keamanan web adalah untuk mencegah berbagai ancaman keamanan seperti serangan *SQL Injection*. Kerentanan terhadap serangan *SQL Injection* merupakan salah satu elemen keamanan situs web yang paling berbahaya. Serangan ini dapat memungkinkan penyerang mengakses atau mengubah data dalam database yang terhubung ke situs web (Y. Natanael et al., 2024).

2.3 Vulnerability Assessment

Kerentanan yaitu suatu kelemahan yang dapat digunakan untuk mengganggu sistem dengan data atau infromasi di dalamnya (Subkhi Mahmasani, 2020). *Vulnerability assessment* merupakan proses sistematis untuk mengidentifikasi, mengevaluasi, dan mengklasifikasikan kerentanan dalam sistem informasi. Proses ini menjadi langkah awal yang penting dalam pengujian keamanan karena memungkinkan organisasi memahami titik- titik lemah sebelum dieksploitasi oleh pihak tidak bertanggung jawab. Dilakukan untuk menentukan celah atau area yang dapat dimasuki oleh penyerang. penyerang kemudian dapat menggunakan kerentanan yang ada melalui celah ini (Yaqi, 2023).

Vulnerability assessment melibatkan teknik seperti pemindaian otomatis menggunaan *tools* (Nikto, OWASP ZAP), serta analisis manual. Output dari proses ini adalah daftar potensi kerentanan beserta tingkat resikonya. (Goel & Mehtre, 2015).

Dilakukan *scan* jaringan untuk mengidentifikasi kelemahan keamanan dan mengenali, mengukur, dan mengklasifikasikan kelemahan keamanan dalam sistem komputer jaringan, dan saluran komunikasi. Membantu mempercepat proses penemuan kelemahan karena kelemahan tersebut diketahui publik dan telah diuji oleh programmer (Fahrina, 2023).

Perangkat lunak *Vulnerability scanning* merupakan *tools* penting bagi penguji penetrasi dan merupakan langkah pertama dalam pengujian. Perangkat lunak ini mencari jaringan untuk perangkat yang mendukung *Internet Protocol* (IP) (Fahrina, 2023).

2.4 Penetration Testing

Penetration testing adalah prosedur yang mensimulasikan serangan nyata dengan tujuan menemukan dan mengidentifikasi berbagai serangan yang mungkin terjadi karena kelemahan sistem (Suprianto, 2022). *Penetration testing* menilai keamanan infrastruktur organisasi dengan meniru serangan nyata. Selama *penetration testing*, langkah-langkah keamanan secara aktif dievaluasi untuk kelemahan desain, kelemahan teknis, dan potensi ancaman (Fahrina, 2023).

Hasil pemeriksaan tersebut akan menghasilkan sejumlah kerentanan yang dapat digunakan oleh penguji penetrasi. Salah satu cara penerapan uji penetrasi yang dapat digunakan yaitu melakukan serangan injeksi, terutama injeksi pada database atau server, yang dapat terjadi dalam kasus *SQL Injection* (Christina Sari et al., 2024).

2.5 SQL Injection

Metode penyerangan *SQL Injection* adalah teknik serangan yang memungkinkan penyerang memasukkan perintah SQL ke dalam input aplikasi sehingga dapat mengakses atau mengubah database secara tidak sah. *SQL Injection* menjadi salah satu serangan tertua namun masih efektif, karena banyak sistem yang belum menerapkan validasi input dengan benar. (Tanang Anugrah et al., 2022). Serangan ini memungkinkan penyerang untuk mencuri atau mengubah data di web server, yang dapat merugikan banyak pihak (Madani, 2024). Penyerang dapat memasukkan perintah SQL berbahaya dan mengubah logika perintah SQL untuk mendapatkan akses ke database dan informasi penting lainnya (Riyanti et al., 2024).

Mengirimkan data yang tidak dapat diverifikasi ke interpreter dalam bentuk perintah atau *query*, penyerang dapat menggunakan data ini untuk menipu interpreter dengan memberi perintah yang tidak diinginkan atau menggunakan data yang sudah ada tanpa menggunakan otentikasi yang tepat (Dharmawan et al., 2022).

SQL Injection bekerja dengan memasukkan perintah *query* SQL sebagai input, yang dapat dilakukan melalui halaman web atau *prompt* perintah. Halaman web mengumpulkan data dari user dan kemudian membuat *query SQL* untuk masuk ke dalam database. *SQL Injection* juga dapat disebut sebagai tindakan yang menipu *query* database sehingga seseorang yang tidak terlatih dapat mengetahui dan mendapatkan data di database sistem (Nursapdahi et al., 2022).

Menurut OWASP ZAP (2023) kerentanan injeksi terus menjadi yang paling umum di aplikasi web (Mutedi & Tjahjono, 2022) *SQL Injection* memiliki beberapa

tipe, diantaranya *error-based, union-based*, dan *inferensial (blind). Inferential SQL Injection* sendiri terbagi dua:

1. *Boolean-based blind* digunakan penyerang mengevaluasi respons aplikasi berdasarkan hasil logika benar/salah dari *query* yang disisipkan. Contoh payload:

   ```
   a. Payload: id=1′   AND 1=1 --
   b. Payload: id=1′   AND 1=2 --
   ```

2. *Time-based blind* digunakan penyerang menentukan kebenaran kondisi dengan mengamati waktu respons yang tertunda (*delay*). Contoh payload:

   ```
   c. Payload: id=1′     AND IF (1=1, SLEEP(5),  0) --
   ```

Kedua metode ini dianggap lebih tersembunyi namun tetap berbahaya, karena tidak memberikan keluaran langsung, tetapi masih dapat digunakan untuk mencuri informasi.

2.6 *Tools* Eksploitasi Dan Pemindaian

Eksploitasi merupakan fase yang menentukan apakah sebuah sistem dapat diserang, biasanya dengan serangan yang mengganggu sistem. Selama proses eksploitasi, seseorang harus memahami sistem mana yang harus diserang dan mengetahui kerentanan sistem (Alanda et al., 2021). Eksploitasi sering digunakan secara legal atau illegal untuk penguji penetrasi mencari celah kelemahan sistem yang sebagai target. Bisa juga disebut sebagai perangkat lunak yang menyerang bug keamanan tertentu, tetapi tidak selalu bertujuan untuk melakukan Tindakan yang tidak diinginkan.(Bruno, 2019)

1. SQLMap merupakan alat yang secara otomatis menemukan dan mengeksploitasi kerentanan *SQL Injection*. Dapat digunakan untuk menguji aplikasi web untuk mengetahui kerentanan *SQL Injection* dan mendapatkan akses ke database yang rentan. (Prasetya, I. A., & Safriadi, 2015) Berbagai teknik *SQL Injection* yang didukung oleh SQLMap termasuk *Boolean blind, time blind, error blind, union based, inteferential, dan out-of-band* (Lika et al., 2018)
2. OWASP ZAP merupakan aplikasi *open source* yang memungkinkan untuk melakukan *scanning* kerentanan aplikasi web. Memiliki kemampuan *spider, active scan*, dan analisis *header* keamanan (Christina Sari et al., 2024)
3. Nikto merupakan pemindaian sever web *open source* (GPL) yang menguji web server yang dapat mendeteksi file sensitif, konfigurasi yang buruk, dan informasi server yang dapat digunakan dalam serangan. (Pormes et al., 2024)
4. Nmap digunakan untuk port *scanning* dan informasi jaringan, membantu dalam tahap *reconnaissance* dari penetration testing. (Arafat, 2020)

2.11 Tabel Matriks Penelitian

Tabel matriks penelitian terdapat pada tabel 2.1.

Tabel 2. 1 Matriks penelitian Terkait

No	Penulis	Judul	Metode Penelitian	Operating Sistem, Tools	Kelebihan	Kekurangan	Gap
1	Rangga wahyu setiawan, wahid miftahul ashari	Infiltrasi Union: *SQL injection* untuk ekstraksi kredensial admin	Eksperimen manual teknik	Windows, Manual SQL Injection	Menunjukkan bahwa adanya kerentanan terhadap eksploitasi data	Terbatas menggunakan teknik *union-based*	Tidak menguji teknik *SQL Injection* yang lain seperti *inferenstial blind*
2	Yehezkiel Natanael, Rangga Felicia, Essy Malays Sari Sakti	Analisis keamanan informasi Bagi Pengguna Website Menggunakan Kali linux melalui Teknik *SQL Injection*	Penetration (eksperimen)	Kali linux, SQLMap	Menggunakan SQLMap dan sistem nyata	Tidak menjelaskan teknik *SQL Injection* yang digunakan	Belum fokus pada teknik yang digunakan secara eksplisit

No	Penulis	Judul	Metode Penelitian	Operating Sistem, Tools	Kelebihan	Kekurangan	Gap
3	Nursaspdhani, Arif Senja Firani, Mochamad Alfan Rosid, Sukma Aji	Studi Analisa serangan *SQL Injection*	Simulasi serangan dan deteksi	Kali linux, DVWA,snort	Fokus pada deteksi intrusi	Tidak mengeksploitasi *SQL Injection* secara penuh	Belum melakukan eksplorasi eksploitasi *SQL Injection* secara nyata
4	Gaguk Suprianto	Penetration Testing Pada Sistem Informasi Jabatan Universitas Hayam Wuruk Perbanas	*Penetration testing* (manual testing)	Windows acunetix, browser	Identifikasi berbagai celah keamanan	Tidak fokus pada *SQL Injection*	Tidak memberikan penjelasan eksploitasi *SQL Injection*
5	Rifqi Azis, Setiadi Yazid	Pengujian Kerentanan Website WordPress dengan menggunakan *Penetration Testing*	Penetration testing	Kali linux, WPScan, Burp Suite	Fokus CMS popular (wordpress)	Tidak membahas *SQL Injection* secara spesifik	Belum melakukan teknik *SQL Injection* lanjutan

		untuk Mengahasilkan Website yang Aman					
6	Dayan Sigasatia, M. Hafid Totohendarto, Joko Saputro	Penetration Testing untuk Menguji Kerentanan pada Sistem Informasi Akademik di Sekoalh Tinggi Teknologi XYZ	*Penetration testing (vulnerability scanning)*	windows *Penetration testing (vulnerability scanning),* acunetix	Menguji port dan celah kerentanan	Tidak melakukan pengujian *SQL Injection* lanjutan	Belum melakukan eksplorasi *SQL Injection*
7	Marcell Dwi Purnomo, Ahmad Chuhsyairi	Pengujian keamanan sistem menggunakan metode Penetration Testing di Website Diskominfotandi kota Bekasi	*Penetration testing*	Kali linux Netcraft, dirb, web browser, SQLMap	Menemukan kerentanan *SQL Injection* pada website instansi	Kurang menjelaskan lebih detail pada teknik yang digunakan	Belum melakukan pengujian menggunakan *blind SQL Injection*

No	Penulis	Judul	Metode Penelitian	Operating Sistem, Tools	Kelebihan	Kekurangan	Gap
8	Sudiharyanto lika, roy dwi putra halim, ihsan verdian	Analisa serangan *SQL injection* menggunakan SQLMap	Eksperimen SQLMap otomatis	Kali linux SQLMap	Menunjukkan eksploitasi pada database	Tidak menyertakan validasi keberhasilan hasil eksploitasi	Belum sampai melakukan validasi keberhasilan eksploitasi
9	Badaruddin bin halib, edy Budiman, hario jati setyadi	Strategi hacking web server dengan SQLMap di kali linux	*Penetration testing* eksperimen	Kali linux SQLMap	Menjelaskan strategi injeksi yang dilakukan	Tidak menjelaskan lebih mengenai payload yang digunakan	Tidak menjelaskan teknik eksploitasi yang digunakan
10	Rudi hermawan	Teknik uji penetrasi web server menggunakan *SQL Injection* dengan SQLMap di kali linux	Simulasi eksploitasi	Kali linux SQLMap	Eksploitasi berhasil dilakukan	Tidak membahas mitigasi untuk memperbaiki sistem	Belum menyertakan rekomendasi perbaikan

No	Penulis	Judul	Metode Penelitian	Operating Sistem, Tools	Kelebihan	Kekurangan	Gap
11	Naomi Augusta, Asep Id Hadiana, Fajri Rakhmat Umbara	Sistem keamanan website dengan multi metode untuk mencegah *SQL Injection*	Eksperimen preventif SQLi	windows SQL Query, Stress, MS Client Stats	Fokus pada pencegahan *SQL Injection*	Tidak melakukan eksploitasi	Hanya fokus pada pencegahan tanpa melakukan pengujian langsung pada sistem nyata
12	Dwiky AL ASyam, endnag wahyu pamungkas	Analisis keamanan database aplikasi web dengan *SQL Injection* menggunakan *penetration tools*	*Penetration testing*	windows Web browser	Menguji parameter pada url	*Tools* terbatas, kurang eksplorasi secara teknis	Tidak menjelaskan lebih lanjut mengenai teknik *SQL Injection* yang digunakan
13	Nico Natanael	Web *penetration testing* dalam mencari kerentanan *SQL Injection*	Simulasi eskploitasi	Kali linux OWASP Juice Shop	Menunjukkan celah login dasar	Hanya menggunakan simulasi	Tidak melakukan eksploitasi pada website nyata

No	Penulis	Judul	Metode Penelitian	Operating Sistem, Tools	Kelebihan	Kekurangan	Gap
14	Nafa Nisaul Ummah	*Penetration Testing* Pada Aplikasi Berbasis Web Terhadap Serangan *Sql Injection* Menggunakan Metode *Boolean Based Blind* Dan *Time Based Blind*	*Penetration testing,* eksperimen	Kali linux SQLMap	Fokus eksploitasi teknik *inferensial*, uji nyata ke aplikasi publik	Tidak membandingkan teknik lain	Menggunakan teknik *inferensial blind* pada sistem nyata dan rekomendasi mitigasi

2.12 Tabel *Literatur Review*

Tabel *literatur review* tertera pada tabel 2.2.

Tabel 2. 2 Tabel *Literatur Review*

No	Penulis	Judul Penelitian	Hasil
1	Rangga Wahyu Setiawan, Wahid Miftahul ashari	*Infiltrasi Union: SQL injection* untuk ekstraksi kredensial admin	*SQL Injection* yang menggunakan metode union berhasil menghilangkan kredensial administrator dari tabel tersembunyi dan mengakses akun sistem. Meskipun eksploitasi kerentanan *SQL Injection* dilakukan secara manual, penelitian berhasil membuktikan efek nayata dari kerentanan yang ada.
2	Yehezkiel Natanael, Rangga Felicia, Essy Malays Sari Sakti	Analisis keamanan informasi Bagi Pengguna Website Menggunakan Kalilinux melalui Teknik *SQL Injection*	Sqlmap terbukti berhasil menemukan kerentanan dan menggunakan database sensitif. Menekankan efektivitas tools otomatisasi pada penelitian ini, tetapi belum membahas secara menyeluruh jenis teknik *SQL Injection* yang digunakan.
3	Nursapdhani, Arif Senja Firani, Mochamad Alfan Rosid, Sukma Aji	Studi Analisa serangan *SQL Injection*	Intrusi sistem dan DVWA *application* menguji *SQL Injection* terhadap server snort IDS, fokus pada pengawasan sistem deteksi intrusi (snort) yang digunakan untuk mendeteksi serangan.

			Meskipun menekankan pentingnya deteksi dini, penelitian ini tidak membahas eksploitasi database atau dampak langsungnya.
4	Gaguk Suprianto	*Penetration Testing* Pada Sistem Informasi Jabatan Universitas Hayam Wuruk Perbanas	Menggunakan *penetration testing* untuk menemukan masalah keamanan dalam sistem informasi akademik Menunjukkan XSS, akses shell, dan kelemahan otentikasi. Tidak berfokus pada *SQL Injection*, tetapi menunjukkan efek celah input yang tidak baik.
5	Rifqi Azis, Setiadi Yazid	Pengujian Kerentanan Website WordPress dengan menggunakan *Penetration Testing* untuk Mengahsailkan Website yang Aman	Penelitian ini menunjukkan bahwa pengujian rutin sangat penting untuk situs yang menangani data sensitif, tetapi hal itu masih umum dan tidak spesifik untuk metode *SQL Injection*.
6	Dayan Sigasatia, M. Hafid Totohendarto, Joko Saputro	*Penetration Testing* untuk Menguji Kerentanan pada Sistem Informasi Akademik di Sekolah Tinggi Teknologi XYZ	Peneliti menemukan 5 ancaman pada sistem informasi akademik sekolah tinggi XYZ. Dengan berbagai tingkat dan celah, mulai dari informasional hingga medium, beberapa port yang terbuka, dan tidak menemukan celah *SQL injection*.
7	Marcell Dwi Pornomo, Ahmad Chuhsyairi	Pengujian keamanan sistem menggunakan metode *Penetration Testing* di Website Diskominfotandi kota Bekasi	Hasil penelitian terdapat dua masalah yang ditemukan pada website DPPPA, Metode *Penetration testing* menunjukkan bahwa *SQL Injection* memiliki masalah penting dan sensitive

			dalam penyampaian data, hal itu menunjukkan SQLmap mampu mengekstrak informasi sensitif.
8	Sudiharyanto Lika, Roy Dwi Putra Halim, Ihsan Verdian	Analisa serangan *SQL injection* menggunakan SQLMap	Berhasil menggunakan situs web menggunakan SQLMap untuk mengidentifikasi database, tabel dan lainnya, serta mengekstrak data pribadi. Penelitian yang dilakukan menggambarkan proses secara teknis dengan jelas.
9	Badaruddin Bin Halib, Edy Budiman, Hario Jati Setyadi	Strategi hacking web sserver dengan SQLMap di kali linux	Serangan *SQL Injection* berhasil mengambil data penting melalui form input. Meskipun penelitian dapat dengan jelas menunjukkan teknik serangan *SQL Injection*, peneliti tidak menampilkan payload yang digunakan, alat dan metode nya. Penelitian ini kurang menganalisis dari sisi teknis nya.
10	Rudi Hermawan	Teknik uji penetrasi web server mengguankan *SQL Injection* dengan SQLMap di kali linux	Dengan menggunakan SQLMap, simulasi penetrasi berhasil mengeksploitasi situs target yang menampilkan database, tabel, serta mengakses server target melalui SSH. Namun penelitian ini tidak mencakup matrik perlindungan pada keamanan informasi.
11	Naomi Augusta, Asep Id Hadiana, Fajri Rakhmat Umbara	Sistem keamanan website dengan multi metode untuk mencegah *SQL Injection*	Serangan *SQL injection*, dapat dicegah dengan menggunakan prosedur penyimpanan aman. Fokus pada efisiensi dan keamanan pertanyaan. Pengujian aplikasi lima halaman menunjukkan peningkatan signifikaan dalam efisiensi prosedur penyimpanan.

No	Penulis	Judul Penelitian	Hasil
12	Dwiky AL ASyam, endnag wahyu pamungkas	Analisis keamanan database aplikasi web dengan *SQL Inejction* menggunakan penetration tools	*SQL injection* dapat digunakan untuk memanfaatkan kelemahan dalam validasi input dan keamanan database, karena beberapa parameter dalam url memungkinkan serangan *SQL injection* yang memungkinkan paenyerang mengakses database dan mengambil data sensitif.
13	Nico Natanael	Web *penetration testing* dalam mencari kerentanan SQL Injection	Hasil penelitian menemukan celah keamanan pada form login, kemudian melakukan eksploitasi dengan payload sederhana, dan berhasil mengakses website tersebut. Penelitian membuktikan ke efektif an kerentanan dasar *SQL Injection*. Namun, pengujian ini dilakukan di lingkungan simulasi.