

BAB I

PENDAHULUAN

1.1 Latar Belakang

Aplikasi berbasis *website* adalah aplikasi yang dikembangkan menggunakan *HTML*, *PHP*, *JS*, *CSS* dan memiliki *database* pada saat ini. Dari tahun ke tahun teknologi yang digunakan untuk membangun aplikasi berbasis *website* semakin berkembang. Hal ini di dasarkan pada kebutuhan dari banyak sektor, ini bertujuan untuk mempermudah pekerjaan manusia di berbagai sektor, tetapi dari kemudahan tersebut terdapat masalah keamanan yang selalu muncul dari tahun ke tahun [1].

Banyak sekali ancaman terhadap aplikasi berbasis *website* yang bisa muncul karena beberapa faktor seperti tidak menggunakan *update* terbaru, kelalaian *programmer*, kesalahan konfigurasi yang dilakukan oleh *admin server* dan *programmer*. Keamanan harus mempunyai unsur-unsur kerahasiaan, integritas dan ketersediaan. Keamanan *website* merupakan upaya untuk melindungi *website* dari serangan *hacker* yang dapat mengakibatkan kerusakan, pencurian dan penipuan. Serangan yang sering dilakukan oleh *hacker* adalah *SQL Injection*, *Credential Bruteforce*, *Cross-site Scripting* dan *Cross-site Requests Forgery* [2].

SQL injection merupakan metode serangan dengan menggunakan pernyataan *SQL* atau kueri *SQL* yang dimodifikasi untuk mengeksploitasi cara halaman *web* berkomunikasi dengan basis *data backend*. Serangan ini dapat bekerja pada halaman *web* yang rentan yang menggunakan basis *data backend* seperti *MySQL*, *Oracle* dan *MSSQL* [3].

Bruteforce merupakan upaya serangan untuk mendapatkan akses sebuah akun dengan menebak *username* dan *password* yang menggunakan *tool* atau tidak untuk membantu penyerang menebak kredensial dari akun. *Bruteforce* adalah teknik serangan lama yang sampai saat ini masih digunakan dan masih dianggap efektif [4].

Cross-Site Scripting (XSS) merupakan celah keamanan pada aplikasi *web* yang terjadi ketika *input* yang diterima dari pengguna tidak divalidasi atau disaring dengan baik. Hal ini memungkinkan penyerang untuk menyisipkan sintaks *HTML* atau *JavaScript* yang dapat digunakan untuk memanipulasi konten *website* [5].

Cross-Site Request Forgery (CSRF) merupakan celah keamanan yang memungkinkan penyerang melakukan permintaan tidak sah tanpa harus mengakses aplikasi secara langsung. Dengan demikian, penyerang dapat membuat permintaan untuk tujuan jahat [6].

Distributed Denial of Service (DDoS) adalah serangan jaringan terkoordinasi yang sangat merusak, terdistribusi, dan berskala besar. Penyerang menggunakan banyak mesin boneka di bawah kendalinya untuk meluncurkan serangan penolakan layanan (*DoS*) pada target secara bersamaan yang akhirnya menyebabkan sistem target kehabisan sumber daya atau bahkan *crash*, membuat sistem target tidak dapat menyediakan layanan yang diperlukan untuk pengguna normal. Sejak serangan *DDoS* pertama terjadi pada tahun 1999, serangan *DDoS* telah menjadi salah satu ancaman *cyber* yang paling luas dan mematikan [7].

Website Resmi Dinas Komunikasi dan Informatika Kabupaten XYZ merupakan sarana penyebaran informasi terkait gambaran potensi pemerintah Kabupaten XYZ, dengan adanya *website* ini aliran informasi antara pemerintah dan masyarakat dapat dilakukan dengan mudah. Manfaat *website* diantaranya: sebagai media penyampaian informasi secara resmi dari pemerintah Kabupaten XYZ kepada masyarakat, sebagai media interaksi dengan masyarakat secara digital, sebagai tempat masyarakat menyampaikan aspirasi, memudahkan masyarakat mengenal pemimpin nya, sebagai media promosi dan menjadi tolak ukur bagaimana aktif atau tidak nya kegiatan pemerintah.

Website harus memiliki tampilan yang bagus dan mudah digunakan oleh masyarakat, selain itu *website* harus memiliki keamanan dari serangan *hacker* karena *website* tersebut memiliki *database* untuk menyimpan *data-data* informasi resmi dari pemerintah Kabupaten XYZ. Sebagai contoh *data* berita dan *data* pengguna *website* tersebut.

Ada banyak serangan lain yang dapat dilakukan terhadap aplikasi berbasis *website*. Penting untuk mengadopsi praktik keamanan yang baik seperti melakukan *secure coding*, membuat fungsi untuk menangkal karakter khusus pada *SQL*, menggunakan karakter khusus pada *password*, melakukan *hashing* terhadap *password*, menggunakan fitur *captcha*, enkripsi *data* pada *database*, menggunakan fungsi *escaping*, menggunakan *token csrf* dan melakukan uji keamanan untuk mengetahui potensi kelemahan guna mencegah serangan *hacker*.

Pendekatan yang umum digunakan untuk mendeteksi celah keamanan pada *website* yang memiliki *database* adalah dengan melakukan *Penetration Testing*. *Penetration Testing* adalah serangkaian kegiatan yang dilakukan untuk mengidentifikasi, mengeksploitasi kerentanan dan membantu mengonfirmasi efektivitas atau ketidak efektifan langkah-langkah keamanan yang telah dilaksanakan, sehingga sangat membantu *developer* agar tidak menulis sintak yang dapat membahayakan *website* atau berpotensi di eksploitasi oleh orang yang tidak bertanggung jawab [2].

Penelitian pertama membahas tentang teknik uji penetrasi *web server* menggunakan *SQL Injection* dengan *SQLMAP* di *Kali Linux*. Penelitian ini menghasilkan skenario simulasi penyerangan dengan menggunakan dua *virtual machine*. Komputer pertama di skenariokan menjadi komputer penyerang, selanjutnya komputer kedua difungsikan sebagai *web server* yang akan menjadi target. Setelah itu hasil dari serangan akan dianalisis dampaknya [8].

Penelitian selanjutnya membahas tentang implementasi *Hydra*, *FFUF* dan *WFUZZ* dalam serangan *bruteforce* pada *DVWA*. Penelitian ini menghasilkan serangan yang memeriksa semua kemungkinan kata sandi atau frasa sandi yang dikirimkan ke target dengan harapan dapat menebak dengan benar kredensial yang ditargetkan. Dengan membandingkan ketiga alat tersebut, kita dapat menganalisis perbedaan kecepatan dan akurasi serangan [4].

Penelitian selanjutnya membahas tentang ancaman terhadap keamanan informasi oleh serangan *cross-site scripting* (XSS) dan metode pencegahannya. Penelitian ini menghasilkan analisis terhadap jenis serangan XSS dan cara pencegahan terhadap serangan ini [5].

Penelitian selanjutnya membahas tentang *Detection of SQL Injection vulnerability in Codeigniter framework using static analysis*. Penelitian ini menghasilkan cara mendeteksi kerentanan *SQL Injection* di kode sumber pada *website* yang menggunakan *framework Codeigniter*, setelah itu memberikan sugesti ke *programmer* sehingga mereka dapat memperbaiki kode yang di tulis [9].

Penelitian selanjutnya membahas tentang analisis metode *security PTES (Penetration Testing Execution and Standart)* pada aplikasi *E-Leaning* Universitas Negeri Padang. Penelitian ini menghasilkan deteksi kerentanan teratas pada *website E-Learning* yaitu *Cross-Site Forgery, Development Configuration File, slow HTTP Denial of Service Attack, Weak Password, TLS 1.0 Enable* dan *exploitation* menggunakan teknik *SQL Injection* [10].

Penelitian selanjutnya membahas tentang *SDN-Defend: A Lightweight Online Attack Detection and Mitigation System for DDoS Attacks in SDN*. Penelitian ini menghasilkan sistem yang mendeteksi anomali lalu lintas jaringan dengan metode *deep-learning* dan mitigasi yaitu menggunakan model mitigasi *IP traceback* untuk menemukan penyerang dan secara efektif menyaring lalu lintas abnormal dengan mengirimkan perintah aturan aliran dari pengendali [7].

Penelitian selanjutnya membahas tentang *Snort Versus Suricata In Intrusion Detection*. Penelitian ini menghasilkan analisis komparatif yang komprehensif dari dua *NIDS* terkemuka, *Snort* dan *Suricata* dengan fokus pada arsitektur, kemampuan deteksi, dan metrik kinerjanya [11].

Berdasarkan penelitian sebelumnya, dalam penelitian ini penulis berfokus pada uji penetrasi keamanan dengan melakukan serangan *SQL Injection, Bruteforce* menggunakan pendekatan manual dan automasi, *Cross-Site Forgery* dan *Cross-Site Scripting* dengan pendekatan manual dan serangan *DDoS*. Sehingga dengan adanya penelitian ini bertujuan untuk mengetahui celah keamanan, eksploitasi celah dan merekomendasikan cara menangani celah keamanan serta deteksi serangan siber pada *website* Dinas Komunikasi dan Informatika Kabupaten XYZ agar terciptanya kerahasiaan, integritas dan ketersediaan *data* yang baik.

1.2 Permasalahan

Website Dinas Komunikasi dan Informatika Kabupaten XYZ berfungsi sebagai sarana publikasi dan informasi, memberikan gambaran tentang potensi pemerintah XYZ serta meningkatkan pelayanan kepada masyarakat dalam mengakses informasi. Melalui *website* ini, masyarakat dapat mengetahui kebijakan Pemerintah Kabupaten XYZ. Mengingat peran pentingnya, *website* ini harus memastikan keamanan *data* dan integritas informasi yang disajikan. Namun, terdapat beberapa masalah keamanan. Masalah pertama adalah kerentanan terhadap serangan *SQL Injection*, serangan ini merupakan ancaman serius terhadap *website*. Dalam serangan ini penyerang dapat menyisipkan perintah *SQL* melalui *input* yang disediakan untuk pengguna yang dapat menyebabkan *query* di sisi *server* dapat di manipulasi, kemudian masalah kedua adalah kerentanan terhadap serangan *Bruteforce*, *website* dengan mekanisme autentikasi yang lemah seperti tanpa adanya batasan percobaan *login* dan tanpa penggunaan fitur keamanan tambahan seperti *captcha*, sangat rentan terhadap jenis serangan ini, kemudian masalah ketiga adalah kerentanan terhadap serangan *Cross-Site Scripting website* yang tidak memvalidasi *syntax html, javascript* terhadap *input* yang diterima sehingga konten *website* dapat dimanipulasi tanpa mengubah kode sumber, kemudian masalah ke-empat adalah kerentanan terhadap serangan *Cross-Site Forgery*, *website* tidak menggunakan *token csrf* pada halaman yang memproses *data* yang di *input* dapat menyebabkan permintaan yang tidak sah tanpa harus mengakses aplikasi secara langsung, kemudian masalah ke-lima adalah potensi kinerja layanan menurun akibat serangan *DDoS* yang menyebabkan layanan terganggu atau layanan mati total.

1.3 Batasan Masalah

Berikut ini adalah batasan masalah dari penelitian ini.

1. Penelitian ini berfokus pada uji penetrasi keamanan di salah satu *subdomain website XYZ.go.id* yaitu *subdomain web.XYZ.go.id*.
2. Penelitian ini berfokus pada serangan *bruteforce, SQL Injection, Cross-Site Scripting, Cross-Site Requests Forgery* dan *DDoS*.

3. Uji penetrasi keamanan menggunakan pendekatan manual dan automasi pada serangan *Bruteforce* dan *SQL Injection* dan untuk serangan *Cross-Site Scripting* dan *Cross-Site Requests Forgery* menggunakan pendekatan manual dan serangan *DDoS*.
4. Mengimplementasikan *Intrusion Detection System*.

Oleh karena itu, sangat penting untuk melakukan uji penetrasi keamanan guna mengidentifikasi celah keamanan yang ada, melaporkan temuan tersebut dan memberikan rekomendasi cara menangani celah tersebut serta solusi mendeteksi serangan siber. Langkah-langkah ini diperlukan untuk memastikan bahwa *website* Dinas Komunikasi dan Informatika Kabupaten XYZ aman dari serangan siber dan *data* sensitif yang disimpan di dalamnya tidak dimodifikasi oleh pihak yang tidak berwenang. Dengan demikian, dapat dijaga integritas informasi yang disajikan kepada masyarakat dan dipertahankan kepercayaan publik terhadap layanan online yang disediakan oleh pemerintah.

1.4 Tujuan

Tujuan dari Penelitian ini adalah:

1. Melakukan serangan *SQL injection* dan *Bruteforce* dengan pendekatan automasi dan manual.
2. Melakukan serangan *Cross-Site Scripting* dan *Cross-Site Requests Forgery* dengan pendekatan manual.
3. Melakukan serangan *Distributed Denial of Service*.
4. Menganalisis kerentanan dan dampaknya.
5. Memberikan rekomendasi cara menangani celah keamanan yang ditemukan dan solusi mendeteksi serangan siber.

1.5 Manfaat

Manfaat dari penelitian ini adalah:

1. Membantu *developer* dan *administrator server* untuk menangani serangan *SQL Injection*, *Bruteforce*, *Cross-Site Scripting* dan *Cross-Site Requests Forgery* dan serangan *Distributed Denial of Service*.

2. Dapat meningkatkan keamanan *website* dan memberikan peringatan serangan siber secara *real-time*.
3. Membantu menjaga kepercayaan masyarakat terhadap layanan *online* yang disediakan.

1.6 Sistematika Penulisan

Berikut ini adalah sistematika penulisan dari penelitian ini.

1. Bab 1 Pendahuluan

Jelaskan tentang apa saja yang dibahas pada Bab 1. Penjelasan memuat bagian-bagian penting pada Pendahuluan.

2. Bab 2 Kajian Pustaka

Jelaskan tentang apa saja yang dibahas pada Bab 2. Penjelasan memuat bagian-bagian penting pada Kajian Pustaka.

3. Bab 3 Desain Sistem

Jelaskan tentang apa saja yang dibahas pada Bab 3. Penjelasan memuat bagian-bagian penting pada Desain Sistem

4. Bab 4 Eksperimen dan Analisis

Jelaskan tentang apa saja yang dibahas pada Bab 4. Penjelasan memuat bagian-bagian penting pada Eksperimen dan Analisis.

5. Bab 5 Penutup

Jelaskan tentang apa saja yang dibahas pada Bab 5. Penjelasan memuat bagian-bagian penting pada Penutup.