BAB 1 PENDAHULUAN

1.1. Latar Belakang

Internet merupakan teknologi yang perkembangannya paling cepat dalam era iot saat ini, dengan seiring berkembangnya internet semakin berkembang juga pola pola serangan baru yang dibuat oleh seorang hacker demi mendapatkan keuntungan bagi dirinya sendiri. Salah satu yang menyebabkan internet berkembang dengan cepat antara lain faktor kebutuhan setiap orang maupun perusahaan semakin meningkat dan salah satu cara untuk mempercepat komunikasi ataupun pelayanan yaitu dengan menggunakan layanan internet karena dengan internet kita dapat berkomunikasi dengan jarak sejauh apapun.

Dengan menggunakan internet tentu kita memerlukan suatu interface untuk memudahkan kita dalam menjalankan suatu komunikasi. Salah satu user interface yang dapat kita gunakan yaitu sebuah aplikasi berbasis *website*. Aplikasi *website* sendiri sudah mulai berkembang dari tahun 1993 dimana kala itu *website* diumumkan dapat digunakan secara gratis. Tentu dengan seiring berkembangnya sebuah teknologi *website* yang dapat digunakan untuk saling berkomunikasi, tukar menukar serta memperbaharui informasi secara real time, sebuah *website* akan membutuhkan sebuah environment untuk menampung data ataupun informasi yang kian membanyak demi memudahkan pengguna maupun pengelola dalam melakukan access. Salah satu environment yang dapat digunakan yaitu menggunakan sistem *database*, akan tetapi dengan menggunakan sistem environment *database* tersebut apabila tidak dikelola dengan baik data ataupun informasi akan semakin rentan untuk dicuri oleh para hacker demi keuntungan diri sendiri. Sebuah sistem keamanan akan sangat diperlukan apabila *website* tersebut sudah memiliki sistem *database*.

Adanya sebuah sistem *database* dapat menguntungkan seorang pengelola *website*, namun dapat pula merugikan, mengapa merugikan ? karena dengan menggunakan sistem *database* tentu akan menciptakan sebuah celah bagi para hacker untuk dapat masuk ataupun menerobos sistem aplikasi *website* dengan

menggunakan *payload* url atau merubah suatu url dengan menggunakan kode atau *syntax database*, metode tersebut dapat disebut juga metode SQL *Injection* dimana maksud dari metode tersebut, menambahkan suatu url *website* dengan menggunakan *syntax database*, jika sistem keamanan yang kita buat pada aplikasi *website* lemah maka akan dengan mudah bagi para hacker menerobos masuk kedalam sistem *database* kita dan hasil dari serangan tersebut dapat mengancam keamanan data dari *website* yang diserang karena apabila serangan ini berhasil menembus sebuah *database* maka konsekuensi yang dapat terjadi yaitu kebocoran data, dengan serangan SQL *Injection* ini bila berhasil dilakukan akan bisa melakukan *dump* data, dimana ini sangat berbahaya karena kita ketahui bahwa *dump* sendiri berfungsi untuk melihat isi dari sebuah tabel yang ada di *database*.

Berdasarkan dari laporan yang didapatkan dari *website* OWASP ditahun 2020 serangan SQL *Injection* masih menempati urutan pertama dari 10 tipe serangan *website* lainnya (OWASP 2020, hlm. 1). Ini dikarenakan SQL *Injection* merupakan serangan yang paling beresiko apabila dapat menembus sebuah *website*. Dampak dari serangan tersebut bahkan bisa saja mengambil alih *website* yang terkena serangan SQL *Injection*.

Sebagai contoh kasus SQL *Injection* yang terjadi pada tahun 2020 Menurut (Tom 2020, hlm 1) terdapat suatu celah yang ada pada aplikasi “Where Is My Train” aplikasi yang diakuisisi oleh google ini diyakini oleh Anil Tom memiliki kerentanan pada URL dimana dia melihat parameter yang dapat dimanfaatkan. Hal ini merupakan salah satu hal yang krusial, karena dengan parameter yang terlihat seorang penyerang dapat menyisipkan *payload* berupa query sql, atau dari URL yang terlihat parameternya tersebut dapat salin dan melakukan serangan SQL *Injection* melalui tools pembantu seperti contohnya dengan menggunakan tools SQLMAP.

Salah satu cara pencegahan yang dapat dilakukan untuk menangkal serangan berjudul SQL *Injection* kita dapat melakukan enkripsi URL terhadap *website*

yang telah kita buat sehingga dengan begitu URL yang biasanya dijadikan sebagai kelemahan untuk menaruh *payload* berupa command untuk melakukan query tidak dapat terbaca dan juga tidak dapat disisipi *payload syntax* SQL. selain itu penerapan enkripsi pada URL dikarenakan banyak dari serangan SQL *Injection* dilakukan melalui URL khususnya parameter yang biasanya disisipkan oleh *payload* yang berisi query SQL untuk mengakses *database*.

Berdasarkan dari latar belakang yang telah penulis jabarkan, maka penulis akan melakukan penelitian yang terkait dengan serangan SQL *Injection* dan mencoba untuk menerapkan enkripisi URL untuk mencegah terjadinya serangan SQL *Injection* terhadap keamanan *website*. Oleh karena itu, maka penulis memberi judul penelitian ini "PENCEGAHAN SERANGAN SQL *INJECTION* DENGAN MELAKUKAN ENKRIPSI PARAMTER *UNIFORM RESOURCES LOCATOR* (URL) MENGGUNAKAN ALGORITMA KRIPTOGRAFI HASH DAN *FILTER* ESCAPE STRING".

1.2. Rumusan Masalah

Berdasarkan dari pemaparan latar belakang diatas, maka penulis menyimpulkan bahwa terdapat rumusan masalah sebagai berikut:

1. Apakah mengenkripsi parameter URL dengan menggunakan Algoritma HASH dapat mengamankan sebuah *website* dari serangan SQL *Injection* ?
2. Apakah mengenkripsi parameter URL dengan menggunakan Algoritma HASH memberikan hasil yang optimal dalam mencegah SQL *Injection* ?

1.3. Tujuan Penelitian

Berdasarkan dari pemaparan latar belakang dan rumusan masalah diatas, maka penulis menyimpulkan bahwa tujuan dari dilakukannya penelitian ini yaitu untuk mengamankan *website* dari serangan SQL *Injection* yang dilakukan pada parameter URL.

1.4. Manfaat Penelitian

Berdasarkan latar belakang, rumusan masalah, dan tujuan penelitian yang telah penulis paparkan, dapat disimpulkan bahwa penelitian ini memiliki manfaat sebagai berikut:

1. Dapat membuktikan dengan menggunakan enkripsi parameter URL pada *website* dapat mencegah serangan SQL *Injection*.
2. Dapat dijadikan refrensi untuk penelitian yang terkait dengan pembahasan serangan SQL *Injection*.

1.5. Batasan Masalah

Batasan masalah dari penelitian ini adalah:

1. Penanganan SQL *Injection* ini bekerja pada kasus yang berbasis *website*.
2. Penelitian membahas cara melindungi *website* dari jenis serangan SQL *Injection* dengan melakukan Enkripsi pada bagian parameter URL.
3. Penanganan serangan SQL *Injection* ini khusus pada *website* yang menggunakan *database My*SQL, dan bahasa pemrograman PHP Native.
4. Pengujian dilakukan pada *website localhost*.

1.6. Luaran Yang Diharapkan

Luaran yang diharapkan dari penelitian ini merupakan sebuah solusi yang dapat dimanfaatkan oleh para pembuat *website* sebagai sebuah cara untuk mengamankan *website* dari serangan berjenis SQL *Injection*.

1.7. Sistematika Penulisan

Pada pembuatan laporan penelitian ini penulis menggunakan sistematika penulisan agar mudah untuk dipahami. Berikut ini adalah bentuk sistematika penulisan.

BAB 1 PENDAHULUAN

Pada bab ini penulis membahas mengenai latar belakang, rumusan masalah, tujuan penelitian, manfaat penelitian, Batasan masalah, luaran yang diharapkan dari penelitian ini, dan juga sistematika penulisan penelitian ini.

BAB 2 LANDASAN TEORI

Pada bab ini penulis membahas tentang definisi teori – teori yang penulis gunakan untuk memperkuat serta menjadi referensi untuk mendasari penelitian yang penulis jalani.

BAB 3 METODOLOGI PENELITIAN

Pada bab ini penulis membahas tentang metode, kerangka berfikir, serta jadwal kegiatan yang dilakukan dalam penelitian yang dijalani.

BAB 4 PEMBAHSAN

Pada bab ini penulis membahas tentang penerapan serta pengujian dari topik penelitian yang diambil.

BAB 5 PENUTUP

Pada bab ini penulis membahas tentang kesimpulan dan saran dari semua penelitian yang telah dilakukan.