BAB I
PENDAHULUAN

1.1 Latar Belakang

Semua aplikasi modern baik berbasis *web*, *desktop* bahkan *mobile* menggunakan *database* untuk menyampaikan dan menyimpan informasi. *Database management system* (DBMS) yang umum digunakan di zaman modern ini adalah SQL (*structured query language*) dan NoSQL(*Not Only SQL*). Menurut data dari DB-Engines Ranking, Penggunaan DBMS NoSQL mengalami peningkatan. Banyak aplikasi modern yang menggunakan *database* NoSQL karena menawarkan performa dan keunggulan dalam *scaling* yang baik. NoSQL menyimpan data secara non-tabular dan data yang bersifat non-relasional serta memiliki sintak yang berbeda dengan SQL.

Perkembangan teknologi ini tidak diimbangi oleh keamanan itu sendiri. Kasus kriminal pada dunia maya sering terjadi pada setiap tahunnya dan masih terus berkembang dan semakin banyak jenisnya. Berdasarkan data OWASP Top 10, serangan terbanyak pada survei tahun 2017 yang telah dilakukan organisasi tersebut adalah serangan berbasis injeksi seperti NoSQL, SQL, OS dan LDAP. Pada tahun 2022, kasus kriminal dunia maya yang sering terjadi di Indonesia adalah insiden kebocoran data dan peretasan terhadap beberapa aset digital pemerintahan. Pada tahun 1998, *database* non-relasional ditemukan oleh Carlo Strozzi yang merupakan pengembangan dari *database* SQL. Saat ini, terdapat 3 faktor yang mempengaruhi pemilihan sistem basis data NoSQL yaitu data harus bisa bergerak secara *flexible*, harus mampu bergerak secara cepat dengan data dan

user yang besar dan yang terakhir peningkatan performa untuk dapat memuaskan *user* yang menginginkan pengolahan data yang cepat. Terlepas dari beberapa keunggulan yang NoSQL tawarkan, NoSQL juga memiliki beberapa kerentanan salah satunya serangan NoSQL *Injection* yang dapat dieksploitasi oleh penyerang sehingga dapat digunakan untuk mengekstrak data pada *database* secara ilegal. Terdapat beberapa jenis serangan NoSQL *Injection* yakni: PHP *Tautologies injection*, Union queries, Javascript *Injection*, *Piggybacked queries, Origin violation.* (Shachi et al., 2021).

Salah satu aplikasi modern yang menggunakan DBMS NoSQL adalah Rocket.Chat. Rocket.Chat merupakan aplikasi komunikasi tim *open source* dengan fitur dan tampilannya mirip dengan Slack. Pada aplikasi Rocket.Chat pernah ditemukan celah keamanan NoSQL Injection sehingga seorang penyerang dapat melakukan eksfiltrasi data yang ada pada *database* NoSQL yang digunakan.

Blind NoSQL Injection (*Blind NoSQLi*) merupakan salah satu jenis kerentanan dari jenis serangan injeksi yang metode serangannya cukup rumit sehingga *Penetration Tester (penetration tester)* membutuhkan waktu yang lama untuk dapat menembus *server database*. Nama *blind* mengacu pada kompleksitas proses yang terlibat dalam memasukkan sintaks satu per satu untuk mengambil informasi dari database. Dalam serangan *Blind* NoSQLi, serangan ini bergantung pada pengiriman *query* SQL ke *database* yang memaksa aplikasi memberikan nilai balikan yang berbeda tergantung pada kebenaran atau ketidakbenaran *query* tersebut *TRUE* atau *FALSE* (Yew Joe & Selvarajah, 2021).

Pada umumnya, serangan ini tidak menghasilkan respons langsung dari aplikasi, sehingga pengujian keamanan seringkali buta atau *blind*. Dalam hal ini, pengujian dilakukan tanpa pengetahuan secara pasti tentang query yang dikirimkan dan respons yang dihasilkan, melainkan hanya berdasarkan analisis respons yang diterima. Oleh karena itu, serangan ini dikenal sebagai Blind NoSQL Injection.

NoSQLMap merupakan *tool* otomasi yang cukup membantu *Penetration Tester* dalam melakukan pengujian sistem *database* berbasis NoSQL. NoSQLMap memiliki kekurangan pada jenis serangan *Blind* NoSQLi karena eksfiltrasi data dilakukan secara linear, sehingga membutuhkan waktu yang cukup lama. Dengan penelitian yang dilakukan, proses injeksi dan sintaksis yang dihasilkan di-*generate* secara otomatis. Untuk itu, penelitian ini akan menerapkan dua algoritma, yaitu Algoritma Linear Search dan Binary Search, yang akan dibandingkan. Perbandingan ini didasarkan pada fakta bahwa Algoritma Binary Search memiliki kecepatan yang lebih tinggi dibandingkan Algoritma Linear Search (Nugroho & Mandala, 2020). Tujuan dari penelitian ini adalah untuk merancang dan mengimplementasikan otomatisasi *Blind NoSQL Injection* menggunakan *Binary Search* serta menganalisis performa waktu untuk setiap algoritma yang diimplementasikan pada DBMS NoSQL.

1.2 Rumusan Masalah

Berdasarkan pada latar belakang yang dijelaskan sebelumnya maka rumusan masalah dalam penelitian ini adalah sebagai berikut :

1. Bagaimana membangun *tool* untuk mengotomasi serangan *Blind NoSQL Injection.*
2. Bagaimana mengoptimasi serangan *Blind NoSQL Injection* dengan Algoritma *Binary Search.*
3. Bagaimana pencegahan terhadap serangan NoSQL *Injection*.

1.3 Tujuan

Adapun tujuan yang ingin dicapai dalam penelitian ini adalah:

1. Membangun *tool* untuk mengotomasi serangan *Blind NoSQL Injection.*
2. Melakukan optimasi serangan *Blind NoSQL Injection* menggunakan Algoritma *Binary Search*.
3. Mengidentifikasi pencegahan terhadap serangan NoSQL *Injection* pada aplikasi web.

1.4 Manfaat

Penelitian ini diharapkan dapat memberikan manfaat sebagai berikut :

1. Manfaat Bagi Peneliti

Manfaat penelitian ini bagi pengembang yaitu mengetahui mekanisme serangan NoSQLi pada DBMS NoSQL dan implementasi Algoritma *Binary Search* pada serangan *Blind NoSQLi* khususnya pada MongoDB.

2. Manfaat Bagi Pengguna

Setelah penelitian ini dilakukan, diharapkan dapat memberikan suatu manfaat bagi penguji keamanan (*Penetration Tester*) untuk dapat mengoptimasi serangan *Blind NoSQLi* serta dapat melakukan pengamanan dan pencegahan serangan NoSQL *Injection* pada *Website*.

1.5 Batasan Masalah

Berdasarkan rumusan masalah dan tujuan penelitian, maka untuk mewujudkan penelitian yang sesuai dengan masalah yang ada diperoleh batasan-batasan masalah penelitian sebagai berikut :

1. Pengembangan *tool* ini hanya ditujukan untuk serangan *NoSQL Injection* khususnya pada MongoDB saja.
2. Algoritma yang dibandingkan *Binary Search* dan *Linear Search*
3. Algoritma yang digunakan untuk optimasi adalah *Binary Search*
4. *Tool* dibangun dengan menggunakan bahasa pemrograman python
5. *Report* yang dihasilkan berekstensi csv
6. Website sebagai ujicoba dibangun dengan bahasa pemrograman NodeJS.
7. Jenis serangan yang tersedia berjenis *DB Attacks Boolean-based Blind Injection*.

1.6 Metodologi Penelitian

1.6.1. Tempat dan Waktu Penelitian

Penelitian ini dilaksanakan dalam waktu sekitar 6 (enam) bulan sejak diberikan izin penelitian. Dalam periode tersebut, waktu yang digunakan mencakup 2 bulan untuk pengumpulan data, 2 bulan untuk pengolahan data, dan 2 bulan untuk implementasi yang meliputi penyajian dalam bentuk skripsi dan proses bimbingan. Berikut adalah rangkaian tahapan kegiatan penelitian yang dilakukan.

Tabel 1.1 Waktu Pelaksanaan Penelitian

Tahun	2022				2023				
Bulan	Sep	Okt	Nov	Des	Jan	Feb	Mar	Apr	Mei
Penulisan Proposal									
Perancangan Sistem									
Pembuatan Bab I – V									
Pembuatan Sistem/Program									
Pengujian Sistem									
Penulisan Laporan Akhir									

1.6.2. Bahan dan Alat Penelitian

Dalam penelitian ini, diperlukan perangkat keras dan perangkat lunak sebagai sarana pendukungi untuk implementasi. Perangkat keras dan perangkat lunak tersebut menjadi alat dan bahan penelitian yang penting.

1. Hardware

A. Sistem Operasi: Ubuntu 20.04 WSL & Windows 10

B. Processor: Corei5

C. RAM: 4GB

2. Software

A. VSCode

B. Chrome dan Firefox *browser*

C. Python3 / 2

D. pip

E. Firefox *addons hackbar/ Burp suite*

F. NodeJS

G. MongoDB *Server* dan *Client*

1.6.3. Pengumpulan Data dan Informasi

Data dan informasi yang mendukung penulisan ini diperoleh melalui kegiatan penelusuran literatur, pencarian data melalui internet, dan konsultasi dengan sumber-sumber yang relevan. Sumber data dan informasi yang digunakan termasuk data dari skripsi, laporan praktikum, media elektronik, serta beberapa pustaka yang relevan. Beberapa teknik pengumpulan data yang diterapkan meliputi:

1. Studi kepustakaan yang dilakukan sebelum dilaksanakannya analisis data sebagai bahan pertimbangan dan wawasan penelitian tentang *Blind NoSQL Injection*, DBMS NoSQL, MongoDB dan konsep-konsep yang tercakup dalam penulisan.
2. Observasi dari laboratorium simulasi yang dibangun dengan NodeJS dan MongoDB yang memiliki celah keamanan terhadap serangan *Blind.*

3. Oberservasi dari *tool exploit* yang dibangun dengan menerapkan Algoritma *Linear Search* dan *Binary Search* dengan memberikan output *timestamp* pada tiap mendapat masing-masing huruf.

1.6.4. Analisis Data

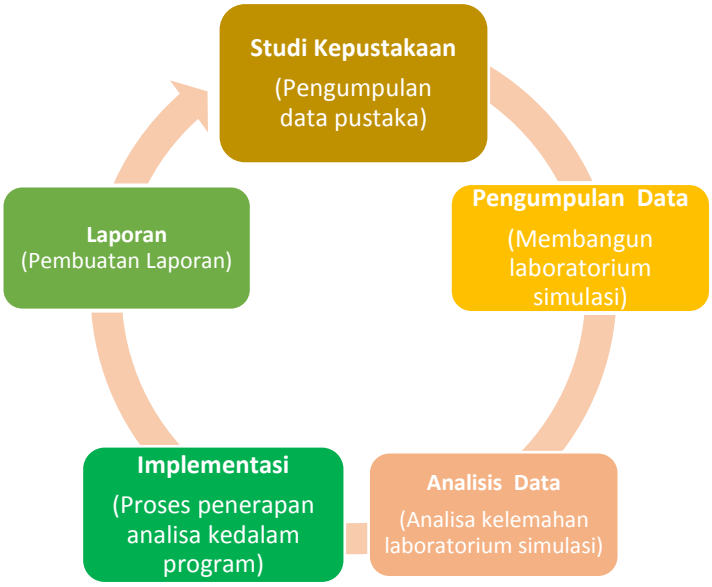
Penelitian ini menggunakan pendekatan kuantitatif untuk membandingkan dua algoritma, Algoritma *Linear Search* dengan *Binary Search*. Data yang digunakan adalah waktu yang dibutuhkan untuk mendapatkan setiap *sequence* huruf *password* dalam jumlah tertentu saat menjalankan algoritma. Semua waktu dijumlahkan dan kemudian dirata-rata. Hasil rata-rata terkecil yang diperoleh dari penelitian ini nantinya akan menunjukkan mana algoritma yang lebih optimal.

1.6.5. Aspek Etika dalam Penelitian

Penelitian ini memperhatikan aspek etika terkait dengan pengumpulan data dan informasi. Laboratorium simulasi telah disiapkan secara khusus untuk observasi menggunakan data dummy sehingga prinsip-prinsip perlindungan privasi dan integritas tetap dijunjung tinggi.

Hal ini mencakup pemastian bahwa data yang digunakan telah dihasilkan dengan cara yang sah dan etis, serta penghindaran penggunaan teknologi atau metode yang dapat membahayakan privasi atau merugikan individu atau kelompok tertentu.

1.6.6. Prosedur Penelitian



Gambar 1.1 Diagram Alur Penelitian

Tahapan penelitian melibatkan beberapa proses dalam membangun sistem, di antaranya:

1. Pada tahap Studi Kepustakaan, penelusuran literatur yang meliputi konsep dan teori terkait NoSQL injection, MongoDB, dan algoritma *Binary search* dilakukan. Tujuan dari tahapan ini adalah untuk memperoleh pemahaman yang mendalam serta landasan pemikiran yang dapat mendukung penelitian ini. Untuk mencapai hal tersebut, pencarian referensi dilakukan melalui buku, artikel ilmiah, jurnal, dan sumber-sumber lainnya yang relevan dengan penelitian ini.
2. Dalam tahap ini, dilakukan pembuatan laboratorium simulasi guna memperoleh data serangan. Simulasi dilakukan melalui konfigurasi *server* Linux dengan pemasangan *web server* dan *database* MongoDB.

Selanjutnya, pada *web server* terdapat aplikasi *web* yang dibangun menggunakan NodeJS dengan rentan terhadap serangan NoSQL *Injection*.

3. Analisis sistem, pada tahap ini, dilakukan analisis sistem dan *penetration testing* terhadap laboratorium yang berjalan di server. Metode pengujian yang digunakan adalah *greybox testing*. Pada tahap ini, peneliti akan melakukan pengujian keamanan aplikasi *web*, dengan eksploitasi pada kerentanan NoSQL *injection*.
4. Implementasi, pada tahap ini dilakukan implementasi dan penerapan algoritma *Linear Search* dan *Binary Search* ke dalam sistem yang dibuat. Metode yang diterapkan dalam pengembangan sistem adalah metode SDLC (*System Development Life Cycle*)
5. Laporan adalah bentuk dokumentasi secara *hardcopy* dari sebuah aplikasi yang telah dibangun.

1.7 Sistematika Penulisan

Adapun sistematika penulisan yang diajukan dalam skripsi ini adalah sebagai berikut :

BAB I : PENDAHULUAN

Bab ini menerangkan tentang latar belakang, ruang lingkup permasalahan, tujuan dan manfaat, metode penelitian dan sistematika penulisan.

BAB II : TINJAUAN PUSTAKA

Bab ini memuat landasan teori yang digunakan dalam

penelitian, yang diperoleh dari berbagai sumber yang relevan. Teori-teori ini disusun berdasarkan jurnal, artikel, dan penelitian terdahulu yang memiliki kaitan dengan topik penelitian.

BAB III : ANALISIS DAN PERANCANGAN

Bab ini memuat identifikasi masalah, pemecahan masalah, perancangan sistem dan perancangan pengujian. Bab ini juga memaparkan flowchart dan UML terkait sistem yang akan dibangun.

BAB IV : PEMBAHASAN

Bab ini menjelaskan hasil dari analisis dan diskusi mengenai program yang telah dirancang, serta mencakup evaluasi terhadap kelebihan dan kekurangan sistem yang telah dibangun.

BAB V : PENUTUP

Bab ini memuat rangkuman dan rekomendasi dari penulis sebagai langkah perbaikan di masa depan untuk sistem yang dibahas.