BAB I

PENDAHULUAN

1.1 Latar Belakang

Server web menjadi bagian penting dari infrastruktur internet saat ini. Sebagian besar dari arsitektur server web yang digunakan saat ini, bertujuan untuk meningkatkan kinerja server web hanya dengan menggunakan *single backend* server web. Permasalahan yang muncul adalah bagaimana single backend server web mampu menangani lonjakan permintaan data yang sangat banyak. Itulah sebabnya, membangun infrastruktur server web yang andal dan sangat tersedia sangat penting. Server web tunggal tidak cukup untuk mendukung aplikasi web lalu lintas tinggi sehingga harus mempertimbangkan menggunakan clustering web server untuk meningkatkan keandalan dan ketersediaan server web (Moniruzzaman, 2014).

Denial of Service (DoS) merupakan salah satu serangan cyber teratas yang tumbuh secara exponsional yang menyerang setiap perusahaan, Instansi, dan Organisasi dengan presentase 43 %. Jenis serangan ini paling banyak merusak operasi bisnis dan mempengaruhi secara global. Dengan cara membanjiri *bandwidth* atau *resources* dari mesin atau sistem, penyerang melakukan *denies* sehingga pengguna yang sah tidak bisa mengakses layanan (Sanders, 2019).

Pencegahan yang dapat dilakukan untuk mencegah serangan Denial of Service (DoS) adalah dengan membangun arsitektur jaringan dan server yang menerapkan metode penyebaran aliran data dan sumber daya pada *node* agar tidak terjadi *down* pada server yang hal tersebut dapat dilakukan dengan menerapkan

sistem penyebaran data pada *node* (Kaur & Sharma, 2018).

Sistem penyebaran data tersebut adalah Load balancing, Load Balancing berfungsi untuk melakukan *reassigning* seluruh beban ke node yang berbeda dari sistem kolaboratif untuk membuat sumber daya pemanfaatan lebih efektif dan untuk meningkatkan waktu *job respons* yang lebih efisien, sekaligus menghilangkan suatu kondisi di mana beberapa node mengalami *over loaded* sementara yang lain dalam keadaan *under loaded*. Dengan menerapkan teknik ini seluruh beban akses dapat diterapkan kepada semua node secara merata (Anselmi, 2020).

Terdapat masalah yang lain, yaitu saat melakukan implementasi biasanya terdapat kendala dimana sistem yang dijalankan dan di testing di server-server fisik yang terlalu banyak kemudian jika terjadi kerusakan akan menimbulkan kerugian biaya, *troubleshooting* yang terlalu rumit, penggunaan daya yang terlalu besar, serta kebutuhan perangkat fisik yang lebih banyak (Wang, Zhu, & Cheng, 2019).

Didasarkan pada efisiesi tersebut, maka diadopsilah arsitektur *containerization* atau kontainer untuk mengimplementasikan pembangunan arsitektur jaringan dan sistem. Teknologi ini bukan termasuk teknologi baru, karena telah banyak perusahaan dan organisasi IT yang telah mengadopsinya. (Evans, 2018),

Containerization merupakan teknologi virtualisasi yang memiliki keunggulan dibanding virtual machine yaitu boot lebih cepat, memiliki overhead kinerja lebih sedikit, dan menggunakan lebih sedikit sumber daya. Dengan memanfaatkan teknologi ini untuk deployment web application dan web server secara kolaboratif, ini memungkinan menjalankan sistem tanpa harus

mengeluarkan banyak biaya dalam pengadaan komponen fisik dan dapat memfasilitasi proses pembangunan, testing, pemeliharaan sistem tingkat rendah, serta konsolidasi server dengan lebih baik (Wang, Zhu, & Cheng, 2019).

Berdasarkan penjelasan di atas maka penulis mengambil judul skripsi " Sistem Keamanan Web Server Dengan Arsitektur Containerization Terhadap Serangan Denial Of Service" dengan maksud membangun sistem keamanan pada web server terhadap serangan Denial of Service (DoS), dengan metode pengamanan yang digunakan yaitu Load Balancing dengan diterapkan menggunakan arsitektur *containerization* atau kontainer.

1.2 Perumusan Masalah

Berdasarkan latar belakang masalah yang telah dijelaskan sebelumnya maka dapat rumusan masalah sebagai berikut :

1 Bagaimana keamanan web server terhadap serangan Denial of Service (DoS)?

2 Bagaimana implementasi load balancing pada pembangunan sistem keamanan web server ?

3 Bagaimana proses konfigurasi dan efisiensi sistem keamanan web server yang diterapkan pada arsitektur containerization ?

4 Bagaimana cara melakukan monitoring web server berbasis arsitektur containerization?

1.3 Batasan Masalah

Adapun yang menjadi batasan dalam penelitian ini adalah sebagai berikut

1 Serangan yang dicegah adalah Denial of Service (DoS)

2 Sistem keamanan dibangun diatas Docker dan menggunakan citra Ubuntu 18.04.

3 Metode pengamanan yang diterapkan load balancing.

4 Penerapan Load balancing menggunakan algoritma Round Robin

5 Beberapa baris konfigurasi mungkin bersifat pribadi dan tidak akan dipublikasikan dilaporan ini, seperti nama pengguna, kata sandi, IP publik dan lain-lain

6 Sistem operasi untuk penerapan dan pengujian menggunakan Linux Ubuntu

7 Skema pengujian menyerang jaringan web server dan arsitektur container

1.4 Tujuan Penelitian

Adapun Tujuan dari penelitian yang dilakukan oleh penulis dengan judul " Sistem Keamanan Web Server Dengan Arsitektur Containerization Terhadap Serangan Denial Of Service" :

1 Membuat sistem keamanan pada web server terhadap serangan Denial of Service (DoS) yang diterapkan pada arsitektur containerization.

2 Implementasi metode load balancing untuk pengamanan web server dari serangan Denial of Service (DoS).

3 Implementasi proses deployment dan monitoring web server pada arsitektur containerization

1.5 Manfaat Penelitian

Ada pun manfaat dari pembangunan Sistem Keamanan Web Server Dengan Arsitektur Containerization Terhadap Serangan Denial Of Service sebagai berikut:

1. Bagi penulis adalah sebagai sarana untuk mengimplementasikan pengetahuan yang telah didapatkan dalam mata kuliah "Jaringan Komputer"

“Keamanan Jaringan” serta “Desain dan Manajemen Jaringan” selama perkuliahan berlangsung, dan dapat membangun arsitektur jaringan web server berbasis containerization yang aman dari serangan Denial of Service.

2. Bagi mahasiswa maupun pembaca sebagai sarana untuk penelitian lebih lanjut tentang keamanan jaringan dalam lingkup serangan pada web server dan media virtualisasi container serta implemetasi metode load balancing..