

BAB I

PENDAHULUAN

1.1 Latar Belakang

Docker dan Kubernetes telah muncul sebagai solusi untuk menangani skalabilitas infrastruktur. Docker memungkinkan pengemasan aplikasi ke dalam *Container*. Docker muncul sebagai respons terhadap masalah yang timbul dalam proses pengembangan dan penyebaran perangkat lunak. Sementara Docker mengatasi masalah pengemasan dan menjalankan aplikasi, Kubernetes lahir untuk mengatasi tantangan dalam manajemen *Container* dalam skala besar. Dengan menggunakan *Container*, muncul kebutuhan untuk mengelola 2 *Container* yang berjalan secara bersamaan dengan efisien. Kubernetes memberikan solusi untuk masalah ini dengan menyediakan *Platform open-source* untuk skalabilitas aplikasi *Container*. Dengan Kubernetes, pengguna dapat mengelola aplikasi mereka dengan cara yang aman, dan skalabilitas. Kubernetes menyediakan *Platform* yang kuat untuk menskala *Container*.

Salah satu teknologi virtualisasi berbasis *Container* yang paling populer saat ini adalah Docker. Docker berfungsi sebagai alat yang efektif untuk menjalankan *Container*. Namun, mengelola *Container* untuk menyediakan berbagai layanan dan melayani sejumlah besar pengguna merupakan tantangan besar bagi Docker. Tantangan utama dalam hal ini adalah proses penskalaan. Ketika sumber daya yang tersedia tidak cukup untuk menghadapi perubahan intensitas beban kerja dari waktu ke waktu, aplikasi dapat mengalami penurunan kinerja atau menghadapi lonjakan permintaan pengguna yang tinggi. Dalam kondisi seperti itu, pemanfaatan sumber daya yang dialokasikan untuk *Container* bisa menjadi tidak efisien. Oleh karena itu, penskalaan diperlukan seiring berjalannya waktu.

Untuk mewujudkan hal tersebut, diperlukan *Container*, yaitu sebuah sistem *open source* (dapat diakses oleh semua orang) yang berperan sebagai

Controller untuk mengelola beberapa *Container*, sehingga manajemen *Container* menjadi lebih mudah. Salah satu fungsinya adalah penskalaan yang secara khusus didasarkan pada metrik infrastruktur, seperti pemanfaatan CPU dan Memory.

Oleh karena itu, penelitian ini dilakukan untuk membandingkan *Container* mana yang lebih unggul dalam penskalaan menggunakan metode *Horizontal Scaler*. *Container* yang akan dibandingkan adalah Docker dan Kubernetes, karena keduanya sangat populer dan sering digunakan oleh perusahaan besar seperti Google dan Ruang Guru. Oleh karena itu penulis tertarik untuk menjadikan topik di atas sebagai bahan Riset Karya Akhir dengan judul "IMPLEMENTASI *SCALING* PADA DOCKER DAN KUBERNETES MENGGUNAKAN METODE *HORIZONTAL SCALER* UNTUK LAYANAN *WEB HOSTING WORDPRESS* (STUDI KASUS DI DSTI UNIVERSITAS BINA DARMA)".

1.2 Rumusan Masalah

Berdasarkan latar yang dibahas dalam penelitian yang akan dilakukan maka peneliti merumuskan masalah yaitu terjadinya *overload* pada *Web hosting Wordpress*, masalah ini terjadi ketika *Web hosting Wordpress* kewalahan karna banyak pengguna atau permintaan layanan tiba-tiba naik, sehingga membuat kinerja *Web hosting Wordpress* menjadi menurun.

1.3 Batasan Masalah

Maka dari itu agar permasalahan tidak *overload* dan lebih optimal, maka dalam penelitian ini penulis membatasi permasalahan yaitu:

- a. Menggunakan Docker dan Kubernetes sebagai alat mengatur sumber daya dengan memastikan penggunaan CPU dan Memori yang lebih optimal supaya tidak terjadi *overload*.
- b. Membandingkan Docker dan Kubernetes menggunakan metode *Horizontal Scaler* yang terbaik untuk digunakan.

1.4 Tujuan Penelitian

Penelitian ini bertujuan untuk membandingkan performa proses penskalaan antara Docker dan Kubernetes. Kubernetes menggunakan metode *Horizontal Scaler*, sementara Docker juga menggunakan metode *Horizontal Scaler* yang serupa. Parameter yang digunakan untuk perbandingan antara Docker dan Kubernetes meliputi uji beban (*Load Testing*) untuk mengukur skalabilitas, serta waktu penskalaan naik (*Scaling up*) dan penskalaan turun (*Scaling down*) untuk mengevaluasi performa.

1.5 Manfaat Penelitian

Manfaat yang diharapkan dari adanya penelitian ini adalah sebagai berikut :

1. Menambah pengalaman dan keterampilan sehingga meningkatkan minat dan kreatifitas.
2. Menambah ragam penelitian tentang pengembangan konsep Docker dan Kubernetes serta *Scaling* untuk optimalisasi layanan *Web hosting Wordpress*.
3. Menambah ragam penelitian tentang *Container* dan pengelolaan *Web hosting Wordpress* yang baik dan benar.