

BAB 1. PENDAHULUAN

1.1. Latar Belakang

Interconnected Network (internet) merupakan media informasi terkini dan paling *up to date* berisikan data informasi yang dapat di akses secara global. Teknologi internet pada saat pandemi *Covid-19* ini sangat di butuhkan oleh masyarakat termasuk pelajar yang menempuh pendidikan selama belajar mandiri di rumah. Berdasarkan data dari *We Are Social* (Digital in 2016), pengguna internet aktif di seluruh dunia kini mencapai angka 3,41 miliar. Dari tahun ke tahun, jumlah pengguna internet bertumbuh hingga 10%. Aplikasi internet yang sering digunakan untuk mendapat informasi adalah aplikasi berbasis *web*. Perkembangan aplikasi berbasis *web* sangat pesat, seiring dengan perkembangan teknologi dan internet. Selain itu, aplikasi berbasis *web* banyak digemari bagi pengguna internet karena dapat diakses di berbagai platform komputer maupun di android hanya dengan menjalankan *web browser*. Sehingga, kemudahan proses *deployment* (penyebaran) aplikasi *web* beserta *software* pendukung seperti *web server*, *database server*, dependensi dan *environment* lain ke server sangat di butuhkan (Asmunin, 2017).

Secara umum ada dua metode *deployment* aplikasi web kedalam server. Pertama menginstall *web* aplikasi beserta *environment* yang dibutuhkan ke dalam *server* tunggal, kelebihanannya adalah *setup server* mudah, *simple* dan cepat dalam proses *deployment*. Tetapi metode tersebut memiliki kekurangan yaitu setiap aplikasi tidak tersiolasi, sehingga apabila *deploy* beberapa aplikasi yang masing-masing memiliki ketergantungan dengan paket versi tertentu dapat menimbulkan konflik dependensi (*dependecy hell*). Metode yang kedua yaitu dengan memanfaatkan teknologi virtualisasi berbasis *hypervisor*, jadi setiap aplikasi dan *dependency* yang di butuhkan *dideploy* kedalam *Virtual Machine* (VM) yang berbeda. Dengan metode ini dapat meningkatkan scalabilitas, karena setiap aplikasi berjalan pada *resource* (CPU, memory, I/O) yang berbeda sehingga dapat dengan mudah ditambahkan sesuai kebutuhan. Akan tetapi masalah klasik menjalankan *virtual machine* berbasis *hypervisor* adalah membutuhkan *resource*

yang besar. Karena setiap VM menjalankan *guest OS* beserta kernelnya sendiri terpisah dari host. Sehingga ketika menjalankan aplikasi yang mungkin besarnya hanya puluhan MB, VM juga harus menjalankan *guest OS* yang besarnya bisa mencapai 10GB. Maka dibutuhkan teknologi yang dapat menyediakan virtualisasi ringan (*Lightweight virtualization*) yang mengisolasi aplikasi beserta *environment* yang dibutuhkan dengan kebutuhan *resource* minimal yang dapat berjalan di berbagai infrastruktur untuk memudahkan proses *deployment* aplikasi (M.Fadlulloh Romadlon, 2017). Maka dari itu penulis melaksanakan pengembangan ini yaitu untuk memudahkan proses *deployment* (penyebaran) aplikasi *web* beserta *software* pendukung seperti *web server*, *database server*, dll ke server. Sehingga kita dapat mengambil kesimpulan permasalahan yang ada di antaranya adalah bagaimana membuat virtualisasi mesin menggunakan *Docker Container*.

1.2. Rumusan Masalah

Berdasarkan latar belakang yang telah diuraikan, maka dapat ditemukan rumusan masalah sebagai berikut:

1. Bagaimana rancangan sistem *hypervisor* terhadap *docker container* ?
2. Bagaimana pengujian sistem pada *docker container*?
3. Bagaimana *management docker container* ?
4. Bagaimana performa virtualisasi *docker container* ?

1.3. Batasan Masalah

Berdasarkan rumusan masalah terdapat batasan masalah yang dapat diambil:

1. Teknologi virtualisasi yang digunakan yaitu *Docker Container*.
2. Sistem operasi yang digunakan yaitu Linux Ubuntu 18.04 LTS.
3. *Mode* jaringan yang digunakan *mac vlan*.
4. *Service* yang dijalankan yaitu *portainer*, *nginx*, *web*, *phpmyadmin*.

1.4. Tujuan

Berdasarkan rumusan masalah terdapat tujuan sebagai berikut:

1. Mengetahui rancangan sistem *hypervisor* terhadap *docker container*.
2. Mengetahui pengujian sistem pada *docker container*.
3. Mengetahui *management docker container*.
4. Mengetahui performa virtualisasi *docker container*.

1.5. Manfaat

Berdasarkan tujuan tersebut terdapat manfaat yang dapat diambil, yaitu sebagai berikut:

1. Memberikan kenyamanan pada pengguna dalam minimnya ketersediaan *hardware* yang dimiliki.
2. Memudahkan virtualisasi dalam pengelolaan banyak aplikasi *web*.

Menambah wawasan bagi penulis dan pembaca mengenai penerapan teknologi virtualisasi berbasis *docker container*.