

# Bab I   Pendahuluan

## I.1   Latar Belakang

Saat ini banyak perusahaan menggunakan layanan-layanan bisnis yang mendukung proses bisnis pada perusahaannya. Banyak layanan yang dipakai seperti Odoo, SAP, layanan web seperti MySQL, dan Apache. Layanan tersebut sangat penting untuk keberlanjutan proses bisnis di perusahaan karena saat ini banyak perusahaan yang bergantung kepada layanan yang digunakannya.

Layanan-layanan tersebut dapat disimpan oleh perusahaan di server sendiri ataupun dengan menyewa cloud. Dengan adanya Docker yang menggunakan teknologi container yang bersifat *lightweight* atau ringan, implementasi container sangat membantu manajemen aplikasi sumber daya cloud (Pahl, Brogi, Soldani, & Jamshidi, 2017).

Banyak kasus yang terjadi saat ini berkaitan dengan matinya sebuah layanan. Hal tersebut dapat disebabkan karena berbagai hal, diantaranya adalah bencana alam, kegagalan sistem, ataupun *maintenance*. Matinya sebuah layanan dapat mengganggu proses bisnis sebuah perusahaan. Banyak kerugian yang ditimbulkan, dan kehilangan data salah satunya. Untuk menanggulunginya, perusahaan dapat melakukan perpindahan layanan dari satu *server* ke *server* lainnya agar ketika satu *server* mati, layanan masih dapat dijalankan di *server* lainnya.

Sebagai solusi dari permasalahan diatas, dapat dilakukan dengan live migration pada *Virtual Machine* (VM) dan Container. Teknologi *live migration* adalah proses untuk memindahkan VM yang sedang berjalan dari satu *host* ke *host* lainnya. (Clemente-Castelló, Fernandez, Mayo, & Quintana-Ortí, 2014).

Tetapi dengan VM, sering terjadi kegagalan saat proses migrasi karena sumber daya komputasi seperti CPU atau RAM tidak *compatible* dengan platform yang digunakan (Svärd, Walsh, Hudzia, Tordsson, & Elmroth, 2014). Penggunaan VM juga dapat menggunakan *system resources* dari komputer seperti CPU dengan sangat besar. Hal tersebut disebabkan oleh virtualisasi sistem operasi virtual. Itu berarti sama saja dengan menjalankan dua sistem operasi secara bersamaan.

Solusi lainnya dengan menggunakan docker. Docker adalah *platform* terbuka dan sebuah tools untuk virtualisasi yang dapat membuat *container* yang ringan dan portabel untuk berbagai aplikasi (Merkel, 2015). Dengan menggunakan

docker, masalah utama virtualisasi menggunakan VM yang menggunakan banyak *system resources* dapat teratasi. Hal ini disebabkan karena docker hanya melakukan virtualisasi di level sistem operasi. Dimana mengisolasi aplikasi dan segala kebutuhannya agar dapat berjalan di segala sistem operasi di sebuah wadah yang disebut *container*.

*Checkpoint/Restore in Userspace* (CRIU) adalah sebuah *project* untuk mengimplementasikan fungsi *checkpoint* dan *restore* pada linux. CRIU adalah sebuah *software* untuk sistem operasi Linux. Yang memiliki fungsi untuk membekukan aplikasi yang sedang berjalan dan menyimpannya di *disk*. Dan *restore* ke sistem lain sebagai backup agar layanan masih dapat berjalan.(CRIU, 2013a)

CRIU *Live migration* adalah proses memindahkan aplikasi atau *container* yang sedang berjalan kepada host lain tanpa interupsi pada aplikasi atau *container*. Dengan *live migration* layanan atau aplikasi masih berjalan dan dapat digunakan (CRIU, 2013b).

Berdasarkan hasil studi literatur di atas. Pada tugas akhir ini penulis akan melakukan analisa proses sistem ketika proses *live migration* sedang berjalan.

## **I.2 Perumusan Masalah**

Rumusan masalah yang akan dibahas pada penelitian ini adalah sebagai berikut:

1. Berdasarkan latar belakang mengenai terinterupsinya layanan yang sedang berjalan. Oleh karena itu diterapkan proses migrasi container layanan menggunakan CRIU.
2. Penelitian ini membandingkan file-file proses sistem yang mempengaruhi CPU Usage ketika sebelum dan saat proses migrasi container dilakukan.

## **I.3 Tujuan Penelitian**

Berdasarkan perumusan masalah, tujuan yang ingin dicapai adalah sebagai berikut:

1. Dapat mengimplementasikan migrasi container layanan menggunakan CRIU.

2. Mengidentifikasi file-file proses sistem yang mempengaruhi CPU Usage ketika migrasi container dilakukan.

#### **I.4 Manfaat Penelitian**

Manfaat yang didapatkan dari penelitian ini adalah:

1. Menjadikan penelitian ini sebagai panduan pengimplementasian proses *live migration* pada *docker* menggunakan CRIU.
2. Menjadikan penelitian ini sebagai rekomendasi untuk menentukan spesifikasi minimal *hardware* CPU yang dibutuhkan.

#### **I.5 Batasan Masalah**

Batasan masalah dari penelitian yang dilakukan adalah sebagai berikut:

1. Menggunakan Ubuntu 16.04 LTS sebagai sistem operasi
2. Menggunakan Docker sebagai container
3. Menggunakan Magento sebagai objek live migration
4. Menggunakan CRIU Live migration
5. Menganalisa proses sistem ketika live migration berjalan
6. Menganalisa file-file proses sistem bertipe FIFO, REG, a\_inode, dan sock.
7. Penelitian hanya melakukan pemindahan aplikasi layanan