BAB I

PENDAHULUAN

Latar Belakang

Pada beberapa tahun ini teknologi kontainerisasi telah semakin popular. Kepopularitasan ini dikarenakan teknologi ini memerlukan sumber daya yang sedikit, mudah diakses, dan rendah akan *error* atau *lag* pada saat digunakan. Banyak perusahaan yang menggunakan teknologi *container* ini, khususnya pada proses pengujian (*testing*) dan proses pengawakutuan (*debugging*). Contohnya adalah docker yang digunakan untuk mencoba aplikasi yang telah dibuat apakah dapat dijalankan di berbagai *platform* atau tidak dan jika ditemukan kerusakan (*bug*) dalam suatu program docker dapat digunakan dalam proses *debugging*. Namun teknologi *container* ini masih belum banyak digunakan pada proses produksi. Karena masih ada celah dalam bidang keamanannya.

Docker menggunakan *client* dan *server*. *Docker client* mengirimkan *request* ke *docker daemon* untuk membangun, mendistribusikan, dan menjalankan *docker container*. Pada proses mengirimkan *request* ke *docker daemon*, docker rentan terkena serangan *docker daemon attack surface* yang memungkinkan terambilnya hak akses *root* pada saat proses pengiriman *request* dijalankan. *Surface* yang dimaksudkan adalah hak akses *root* untuk menggunakan docker. *User* yang memiliki akses ke *docker host* dan *docker daemon* secara otomatis mendapatkan kontrol penuh dari semua *container* dan *images* yang pada docker. *User* dengan hak akses root dapat membuat dan menghentikan *container*, menghapus *images*, memberikan perintah kepada *container* yang sedang berjalan, dan mengekspos informasi yang sensitif seperti *password* dan data-data lainnya. Untuk mencegah serangan itu terjadi, dilakukan pencegahan dengan menggunakan *rootless mode*. Dimana dalam mode tersebut tidak diperlukan akses *root* dalam penggunaan docker.

Rootless mode pada teknologi *container* berfungsi untuk menjalankan docker sebagai pengguna non-*root* pada *host* dan melindungi *host* dari potensi terjadinya serangan terhadap docker.

Topik dan Batasannya

Tugas akhir ini difokuskan pada proses pencegahan serangan *docker daemon attack surface* yang termasuk serangan dari *malicious insider* dan pencegahannya menggunakan *rootless mode* dan pengukuran CPU *usage* yang akan dibandingkan pada saat proses penggunaan docker berlangsung. Lalu berdasarkan data CPU *usage* yang telah diukur, akan didapatkan seberapa besar peningkatan CPU *usage* pada *rootless mode* dan apakah peningkatan tersebut sepadan dengan keuntungan yang didapatkan.

Untuk perumusan masalah yang akan dicangkup dalam penelitian ini adalah bagaimana mencegah serangan *docker daemon attack surface*, menunjukkan bahwa *rootless mode* pada

docker lebih aman daripada menggunakan docker dengan hak akses *root*, dan pengujian performansi CPU *usage* menggunakan perintah ‘docker stats’ pada saat proses penggunaan docker dengan hak akses *root* dan docker dengan *rootless mode*. Diharapkan *rootless mode* dapat mencegah serangan *docker daemon attack surface*, sehingga dapat menghasilkan docker yang lebih aman.

Terdapat beberapa batasan yang ada pada tugas akhir ini, yaitu sistem ini hanya dapat digunakan untuk docker. *Rootless mode* hanya digunakan pada saat proses pencegahan serangan dan serangan *docker daemon attack surface* hanya terjadi pada docker dan menyerang *docker daemon* untuk memperoleh hak akses *root*. Jika serangan berhasil dilakukan, maka dibuktikan dengan *user* dapat mengakses file */etc/shadow* yang seharusnya hanya dapat diakses oleh *user* yang mempunyai hak akses *root*.

Tujuan

Tujuan dari tugas akhir ini adalah untuk mencegah serangan *docker daemon attack surface* pada docker dengan menggunakan *rootless mode* dan mengukur penambahan beban CPU *usage* pada *rootless mode*. Data CPU *usage* akan diambil pada saat docker dijalankan dengan hak akses *root* dan *rootless mode* dalam jangka waktu sesingkat-singkatnya agar didapatkan hasil yang akurat lalu dibandingkan apakah pada *rootless mode* terjadi peningkatan beban CPU *usage* dan apakah peningkatan tersebut sepadan dengan keuntungan yang didapatkan. Pengujian serangan dilakukan terhadap dua *docker container* yang dijalankan secara bergantian, yaitu *docker container* yang dijalankan dengan hak akses *root* dan *docker container* yang dijalankan dengan *rootless mode*.

Organisasi Tulisan

Bagian-bagian selanjutnya pada tugas akhir ini akan memaparkan mengenai dasar teori terkait sistem pencegahan serangan yang akan dibangun pada bagian 2. Kemudian dilanjut dengan pembahasan mengenai perancangan dan pembangunan sistem pencegahan serangan pada bagian 3. Pada bagian 4 akan ditunjukkan hasil pengujian dan evaluasi sistem. Lalu pada bab 5 akan membahas kesimpulan dan saran dari penelitian tugas akhir ini.