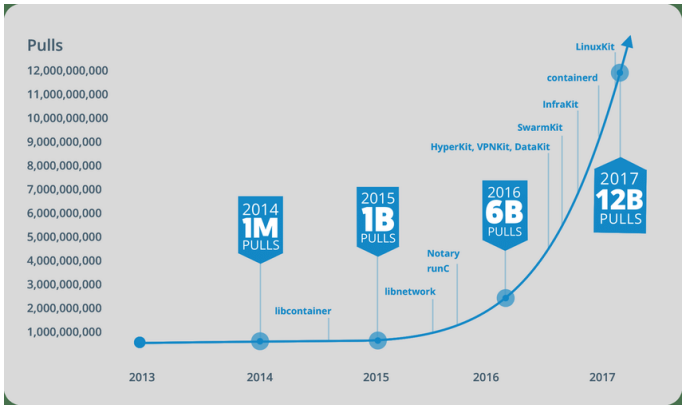
BAB I PENDAHULUAN

I.1 Latar Belakang

Salah satu teknologi populer yang digunakan untuk menyediakan layanan IT adalah *container*. *Container* adalah sistem yang dapat mengelola aplikasi dan juga menjalankan sistem operasi *kernel* tanpa memerlukan *virtual machine* sehingga setiap *container* memiliki proses yang terisolasi sendiri-sendiri dan tidak mengganggu *Host* OS atau *container* lain.

Teknologi terbaru dari *container* yang dikembangkan oleh Solomon Hykes pada tahun 2009 adalah Docker. Docker adalah sebuah platform virtualisasi *container* yang bisa digunakan untuk membangun, menjalankan, dan menyatukan berbagai *software* atau aplikasi lain yang dibutuhkan agar menjadi sebuah wadah (*container)*. Adanya *container* pada docker memberikan banyak kemudahan, akan tetapi perlu diperhatikan mengenai keamanan, *vulnerability* dan risiko dari penggunaannya.



Gambar I. 1 Tingkat Pertumbuhan Docker (Docker,2018)

Efe et al (2020) mengungkapkan seiring berjalannya waktu, laju pertumbuhan pada teknologi Docker meningkat sangat pesat. Peningkatan pesat ini terjadi antara tahun 2015 sampai 2017. Peningkatan tersebut membuat semakin banyak orang yang menggunakan docker. Hal tersebut membuat risiko dari penggunaan Docker juga semakin meningkat dan membuatnya menjadi lebih rentan. *Vulnerability* itu bisa menjadi celah dan membuatnya menjadi mudah terkena

serangan siber seperti *malware*, *ransomware*, *hacking*, dan lainnya oleh pihak yang tidak bertanggung jawab demi kepentingan pribadi atau kelompok.

Untuk mendeteksi adanya celah atau *vulnerability* pada suatu aplikasi dapat melakukan *scanning* terhadap aplikasi tersebut. *Scanning* yang digunakan itu disebut *vulnerability Scanner*. *Vulnerability Scanner* yang digunakan pada penelitian ini adalah suatu *open source vulnerability Scanner*. *Scanning* ini dilakukan untuk memperoleh informasi dari *vulnerability* tersebut yang kemudian akan dilakukan proses *Vulnerability Management.*

Dalam pelaksanaanya *scanning* ini akan dilakukan berdasarkan suatu *CyberSecurity framework* agar pengujian dari penelitian ini memiliki tahapan yang pasti dan jelas. *CyberSecurity framework* yang akan digunakan dalam penelitian ini merupakan standar yang dikeluarkan oleh *General Services Administration* (GSA) *Chief Information Security Officer* (CISO) dengan kode dokumen CIO-IT *Security*-17-80. Standar ini dipilih karena mempunyai tahapan yang cukup lengkap dalam melakukan pengujian *vulnerability Management* pada penelitian ini. Tahapan tersebut adalah *Scanning Capabilities*, *Vulnerability Scanning Process*, *Vulnerability Scan Reports*, *Remediation Verification, Re-Classification of Known Vulnerabilities*.

Selain memiliki fokus penelitian pada kerangka kerja *Vulnerability Management*, penelitian ini juga menganalisis penilaian proses mitigasi dari aset IT yang memiliki perbedaan versi. Dengan memiliki data kenaikan versi suatu *Software Container*, maka proses mitigasi *vulnerability* dapat dianalisis

Oleh karena itu, berdasarkan permasalahan yang sudah disebutkan sebelumnya, penelitian ini akan memberikan sebuah hasil analisis *vulnerability Management* pada Docker *Images* dan Aplikasi Docker *Images* Joomla menggunakan *vulnerability Scanner*. *Vulnerability Scanner* yang akan digunakan pada penelitian ini yaitu Clair *Scanner* dan JoomScan dengan acuan standar GSA CIO-IT *Security*-17-80.

I.2 Perumusan Masalah

Berdasarkan latar belakang diatas, maka rumusan permasalahan untuk penelitian ini adalah sebagai berikut:

1. Bagaimana implementasi dan analisis *Vulnerability Management* pada Docker *Images* dan Aplikasi Docker *Images* berdasarkan suatu standar ?
2. Bagaimana menelusuri *vulnerability* pada Docker *Images* dan Aplikasi Docker *Images* ?
3. Bagaimana proses mitigasi hasil v*ulnerability* dengan membuat sistem baru yang serupa dengan sistem lama tetapi memiliki versi lebih tinggi ?

I.3 Tujuan Penelitian

Berdasarkan perumusan masalah, maka tujuan dari penelitian ini adalah sebagai berikut:

1. Menerapkan standar GSA CIO-IT *Security*-17-80 untuk melakukan *Vulnerability Management* pada Docker *Images* dan Aplikasi Docker *Images*.
2. Mengimplementasikan *open source vulnerability scanner* untuk mendapatkan *vulnerability* pada Docker *Images* dan Aplikasi Docker *Images*.
3. Mengelola *vulnerability* Docker *Images* dan Aplikasi Docker *Images* dengan membuat sistem versi – 1 dan versi – 2 dengan versi *software* lebih tinggi pada versi – 2 dari versi - 1.

I.4 Manfaat Penelitian

Adapun manfaat penelitian ini secara adalah sebagai berikut :

1. Teoritis
    - Memberikan sumbangan pengetahuan terkait pengelolaan *vulnerability* pada *vulnerable* docker berdasarkan standar GSA CIO-IT *Security*-17-80.

- Memberikan gambaran penggunaan standar GSA CIO-IT *Security*-17-80 sebagai panduan pengelolaan keamanan suatu aset IT berdasarkan tahapan yang rinci.

2. Praktis
   - Memberikan gambaran fungsi tools *vulnerability scanning* pada Docker.
   - Memberikan gambaran cara penggunaan Clair Scanner dan JoomScan pada Docker *Images* dan Aplikasi Docker *Images*.

I.5 Batasan Penelitian

Adapun batasan masalah pada penelitian ini sebagai berikut:

1. Penggunaan standar GSA CIO-IT *Security*-17-80 dibatasi pada modul *Vulnerability Management* dan tidak pada modul yang lain.
2. Tahap eksperimen dari penelitian ini menggunakan simulasi pada skala laboratorium.

I.6 Sistematika Penelitian

Sistematika penulisan pada penelitian ini terdiri dari tujuh bab, yang tersusun sebagai berikut:

BAB I **PENDAHULUAN**

Bab ini berisi penjelasan latar belakang, rumusan masalah, tujuan penelitian, manfaat penelitian, batasan masalah dan sistematika penelitian.

BAB II **TINJAUAN PUSTAKA**

Bab ini berisi uraian teori–teori seperti vulnerability, docker, dan yang lainnya sesuai dengan permasalahan yang dihadapi, judul penelitian, dan penelitian terdahulu yang memiliki keterkaitan dengan penelitian yang sedang dilakukan.

BAB III **METODOLOGI PENELITIAN**

Bab ini berisi penjelasan mengenai konseptual model untuk merumuskan solusi dari permasalahan yang ada. Selain itu, ada juga sistematika penelitian yang digunakan untuk menjelaskan langkah-langkah penyelesaian masalah.

BAB IV **RANCANGAN PENGUJIAN**

Bab ini berisi penjelasan rancangan sistem dan penggunaan *tools open source* terhadap skenario yang dilakukan pada *Vulnerability* docker *images* dan aplikasi docker *Images* dalam bentuk simulasi menggunakan metode GSA CIO-IT *Security*-17-80.

BAB V **IMPLEMENTASI DAN HASIL PENGUJIAN**

Bab ini berisi pengujian yang dilakukan berdasarkan rancangan pengujian dengan menghasilkan data *Vulnerability container* docker dan joomla pada versi - 1 dan versi - 2. Pengujian yang dilakukan menggunakan *tools open source* khusus docker *images Scanner* yaitu Clair *Scanner*, sedangkan untuk aplikasi Joomla menggunakan JoomScan.

BAB VI **ANALISIS**

Bab ini berisi analisis dari data hasil pengujian yang telah dilakukan pada bab sebelumnya mengenai *Vulnerability* yang ditemukan pada docker *container* dan joomla. Analisis ini dilakukan menggunakan ukuran pengelolaan *Vulnerability Management* berdasarkan standar GSA CIO-IT *Security*-17-80.

BAB VII **KESIMPULAN DAN SARAN**

Bab ini berisi penjelasan kesimpulan dari penelitian yang telah dilakukan, rancangan sistem dan skenario pengujian, perancangan dan analisis usulan, serta memberikan saran untuk penelitian selanjutnya.