BAB I PENDAHULUAN

I.1 Latar Belakang

Teknologi informasi (TI) dalam bentuk digitalnya memiliki berbagai macam jenis dan menjadi semakin populer di kalangan masyarakat dunia, salah satunya internet (Danuri & Suharnawi, 2017). Berdasarkan survei tahun 2016 yang dikeluarkan oleh Asosiasi Penyelenggara Jaringan Internet Indonesia, sebanyak 132,7 juta masyarakat Indonesia sudah terhubung dengan layanan internet. Alasan mengapa semakin banyaknya masyarakat yang terhubung dengan internet, karena dapat memberikan berbagai kemudahan, seperti mengakses atau mendapatkan informasi yang dibutuhkan semakin cepat. Semakin pesatnya perkembangan teknologi jaringan internet, terciptalah istilah dunia baru yang dinamakan *cyberspace.*

Cyberscpace adalah media elektronik dalam jaringan komputer yang banyak dipakai untuk keperluan komunikasi satu arah maupun timbal-balik secara online (terhubung langsung). Selain itu cyberspace dapat diartikan juga sebagai suatu Imaginary Location (tempat aktivitas elektronik dilakukan) dan sebuah massy virtual yang terbentuk melalui komunikasi yang terjalin dalam sebuah jaringan komputer.

Munculnya *cyberspace* menghasilkan bentuk lingkungan yang berbeda-beda dan memiliki istilah yaitu *cybercrime*, Internet Fraud, dan lain-lain (Albidin, 2015). Perkembangan teknologi yang semakin berkembang selain membawa dampak positif juga diikuti dengan dampak negatif yang ditemukan pada celah keamanan sistem. Salah satu kejahatan internet adalah *cybercrime. Cybercrime* sendiri merupakan aktivitas kejahatan dengan menggunakan teknologi komputer atau jaringan komputer sebagai alat, sasaran maupun tempat terjadinya kejahatan (Arifin, Z. n.d., 2017).

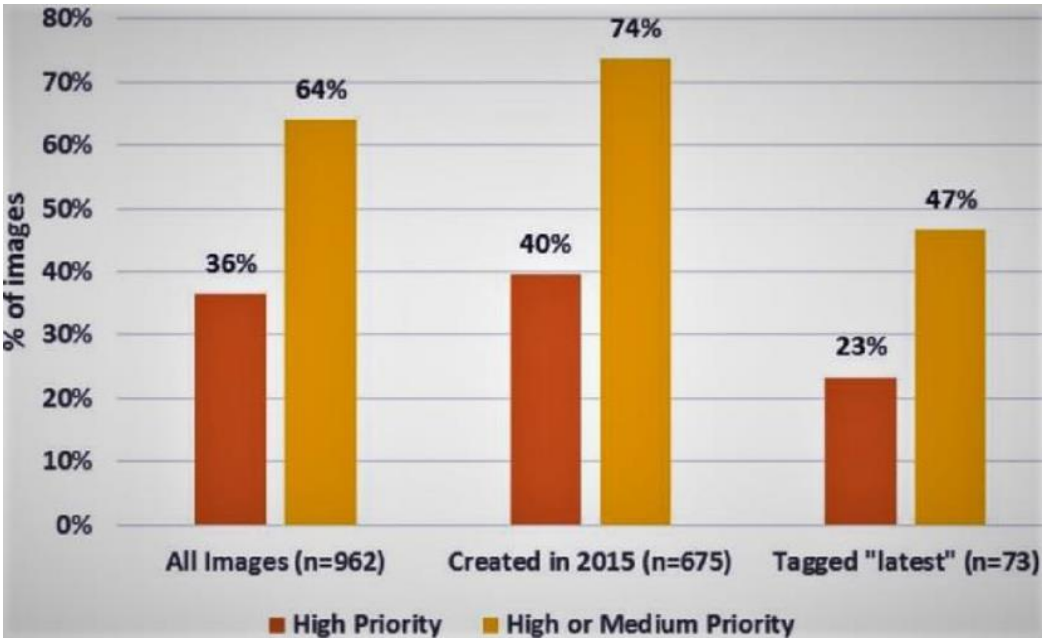
Selain itu cybercrime juga terbagi menjadi dua jenis, yaitu fasilitas kejahatannya melalui teknologi informasi dan kejahatan yang memiliki sasaran sistem dan fasilitas teknologi informasi (Sari, 2018). Dalam hal itu, internet menjadi media

teknologi informasi yang mudah untuk disalahgunakan sebagai alat kriminalitas dan dapat membahayakan masyarakat.

Proses terbentuknya celah keamanan dapat terjadi karena adanya pendistribusian informasi antar aplikasi yang berbeda. Informasi dapat didistribusikan melalui berbagai cara, salah satunya menggunakan kontainer Docker. Kontainer Docker menjadi wadah yang populer untuk menjalankan beberapa *service* layanan, hanya pada satu *host*. Hampir sama dengan sistem virtualisasi, kontainer menyediakan *environment* dan standar yang mudah untuk mengemas serta menyebarkan banyak aplikasi (Shu, et all., 2017).

Docker adalah platform yang berisi kontainer yang dapat mewadahi aplikasi dalam sebuah *image*, dan menjadikannya lebih ringan untuk diakses atau saat pengirimannya. Saat ingin membuat *image*, tidak harus membuatnya dari awal, namun dapat membuat hubungan dengan *Images parent*, sehingga *Images parent* dan *Images child* memiliki hubungan dan saling berkaitan (Shu, et all., 2017). Docker sendiri memiliki dua jenis *Images repository*, yaitu resmi dan komunitas. Dengan *Images* Docker yang memiliki banyak campur tangan organisasi dan berbagai pihak ketiga, mengakibatkan tingkat keamanan dan kerentanan ataupun serangan *malware* sangat mungkin terjadi (Shu, et all., 2017).

Setelah dilakukannya pemeriksaan menunjukkan lebih dari 30% *Images repository* yang berisi *Images* tidak aman saat dilakukan beberapa macam pengujian serangan keamanan (Efe, et all., 2020).



Gambar I. 1 Kerentanan *Images* Resmi
(EFE, A., ASLAN, U., & KARA, A. M., 2020).

Berdasarkan gambar I.1, disajikan data yang menunjukkan hasil yang menjabarkan setiap images resmi dari Docker Hub. Dari semua Images ditemukan lebih dari 33% memiliki kerentanan tingkat tinggi dan hampir 66% memiliki kerentanan tingkat tinggi sampai sedang.

Kerentanan menjadi kelemahan terbesar keamanan sistem dan jaminan suatu informasi (Goel & Mehtre, 2015). Sistem yang tidak memiliki kerentanan dapat memberikan lebih banyak jaminan informasi dan keamanan sistem. Walaupun hampir tidak mungkin untuk memiliki sistem yang bebas 100% dari kerentanan, namun dengan menghapus kerentanan sebanyak mungkin akan meningkatkan keamanan sistem. Oleh karena itu penting nya dilakukan Vulnerability Management untuk mengelola kerentanan dan bagaimana cara mengatasi kerentanan yang telah ditemukan. Pada penelitian ini akan membahas mengenai Vulnerability Management pada Docker dan Docker Images dengan standar yang digunakan yaitu NIST CSF. Pada standar akan menggunakan empat inti kerangka yaitu Identify, Protect, Detect, Respond , dan Recovery. Selain itu untuk mendapatkan data eksperimen akan menggunakan alat *scanning* yang digunakan adalah Docker scan dan OpenSCAP yang merupakan open source Vulnerability scanner.

Berdasarkan uraian permasalahan di atas, pada penelitian akan dijelaskan bagaimana hasil analisis kerentanan Docker dan Docker Images menggunakan Vulnerability *scanner* yaitu Docker scan dan OpenSCAP dengan mengacu pada standar NIST CSF. Dari hasil analisis nantinya akan memberikan rekomendasi kerentanan dan perbandingan kinerja antara open source Vulnerability scanner.

I.2 Rumusan Masalah

Berdasarkan uraian latar belakang yang telah disebutkan, perumusan masalah pada penelitian ini adalah sebagai berikut:

1. Bagaimana hasil perbandingan kinerja pengumpulan data pada Vulnerability *scanner* antara OpenSCAP dan Docker scan ?
2. Bagaimana hasil perbandingan lama waktu *scanning* pada Vulnerability *scanner* antara OpenSCAP dan Docker scan ?
3. Bagaimana proses mitigasi hasil Vulnerability antara Docker dan Docker Images versi lama dan versi baru ?
4. Bagaimana implementasi dan analisis Vulnerability Management berdasarkan standar NIST CSF ?
5. Bagaimana kategori data setelah dilakukan analisis Vulnerability antara Docker dan Docker Images ?

I.3 Tujuan Penelitian

Tujuan penelitian tugas akhir ini sesuai dengan rumusan masalah yang ada. Hal ini berkaitan dengan Docker untuk melakukan implementasi dan analisis kerentanan. Adapun tujuan penelitian pada tugas akhir ini adalah:

1. Mengetahui perbandingan kinerja pengumpulan data pada Vulnerability *scanner* antara OpenSCAP dan Docker scan.
2. Mengetahui perbandingan lama waktu *scanning* pada Vulnerability *scanner* antara OpenSCAP dan Docker scan.
3. Mengidentifikasi *vulnerability* antara Docker dan Docker *Images* versi lama dan versi baru
4. Mengetahui implementasi dan analisis Vulnerability Management berdasarkan standar NIST CSF

5. Melakukan analisis kategori data Vulnerability Docker dan Docker Images.

I.4 Manfaat Penelitian

Adapun manfaat pada penelitian ini sebagai berikut:

1. Teoritis
   - Memberikan kontribusi keilmuan terkait pengelolaan kerentanan pada Docker dan Docker *Images*.
   - Membantu proses *vulnerability Management* menggunakan standar NIST CSF.
2. Praktis

   Memberikan rekomendasi berupa kajian dan pembanding *tools open source container scanner* yang bermanfaat untuk menganalisa *vulnerability* pada Docker dan Docker *Images*.

I.5 Batasan Penelitian

Berikut merupakan batasan pada penelitian :

a. Penelitian hanya memberikan analisis *vulnerability* pada *vulnerable* Docker dan *vulnerable* Docker *Images* berdasarkan standar NIST CSF.

b. Dari 23 Kategori yang terdapat pada inti kerangka NIST CSF, penelitian ini hanya membahas 7 Kategori.

c. Penelitian ini terbatas pada *vulnerability Management*.

I.6 Sistematika Penulisan

Penelitian ini memiliki sistematika pelaporan sebagai berikut :

BAB I Pendahuluan

BAB ini berisi penjelasan latar belakang untuk melakukan penelitian, rumusan masalah, tujuan penelitian, manfaat penelitian, serta batasan penelitian yang menjadi dasar penulis untuk melakukan penelitian.

BAB II Tinjauan Pustaka

BAB ini memuat teori yang relevan dengan judul penelitian serta uraian penelitian sebelumnya.

BAB III Metodologi Penelitian

BAB ini mendefinisikan langkah-langkah penelitian, diantaranya pembangunan model konseptual serta merancang sistematika pemecahan masalah.

BAB IV Tahap Desain Platform Pengujian

BAB ini membahas mengenai perancangan sistem, perancangan topologi serta skenario pengujian yang dilakukan pada penelitian ini.

BAB V Tahap Simulasi dan Pengujian

BAB ini menjabarkan hasil dari skenario pengujian berupa data eksperimen dan data yang terkumpul berdasarkan kategori yang ada pada standar NIST CSF. Selain itu juga di kelompokkan sesuai dengan versi – 1 dan versi – 2.

BAB VI Tahap Analisis

BAB ini menjelaskan mengenai hasil analisis yang didapat dari menganalisis data eksperimen yang telah terkumpul sebelumnya.

BAB VII Kesimpulan dan Saran

BAB ini menjelaskan kesimpulan dari hasil penelitian serta saran yang diperlukan agar penelitian lebih baik.