BAB I
PENDAHULUAN

1.1 Latar Belakang Masalah

Perkembangan *malware* dari tahun ke tahun mengalami peningkatan yang sangat signifikan. Hal ini disebabkan karena semakin banyak dan semakin beragamnya perangkat yang terhubung ke jaringan internet. Selain menyerang sistem komputer, *malware* juga dikembangkan untuk menyerang *smartphone* dan perangkat *Internet of Things* (IoT). Dampak yang disebabkan oleh serangan *malware* juga sangat meresahkan, mulai dari terganggunya aktifitas penggunaan komputer hingga pencurian data dan perusakan sistem komputer.

Menurut survey yang dilakukan oleh Symantec pada tahun 2017 [7], terdapat 357 juta jenis *malware* baru pada tahun 2016. Jumlah itu meningkat 0,5 % dari tahun 2015 yaitu sebanyak 355 juta *malware*. Penyebaran *malware* tersebut dilakukan melalui *email spam* dan menurut survey tersebut, terdapat satu *malware* pada setiap 131 *email spam*. Pada perangkat *mobile* android ditemukan sebanyak 3600 jenis *malware* baru pada tahun 2016. Jumlah tersebut diprediksi akan meningkat setiap tahunnya dikarenakan semakin banyaknya perangkat yang terhubung ke internet.

Beberapa langkah yang sudah diterapkan untuk mempelajari serangan *malware* yang terjadi pada jaringan diantaranya dilakukan di *National Research Nuclear University* (Moscow, Russia) dengan cara menerapkan *honeypot* yang menjalankan layanan FTP. Layanan FTP rentan terhadap serangan *brute force* dan digunakan untuk mengirimkan *malware*. *Honeypot* digunakan karena perangkat keamanan moderen seperti *intrusion detection system* (IDS) bekerja

berdasarkan pola serangan yang sudah diketahui sebelumnya sehingga kemampuan untuk mengenali serangan baru sangatlah terbatas. Selain itu, penggunaan *honeypot* ditujukan untuk mendapatkan informasi yang lebih banyak tentang penyerang. Hasil akhir yang diperoleh adalah *honeypot* berhasil mencatat *username*, *password*, serta *file* yang dikirimkan oleh penyerang [2].

Penelitian yang dilakukan di *information security laboratory of* BITS india merancang sistem *honeypot* terdistribusi yang terdiri dari beberapa jenis *honeypot*. Dari percobaan yang dilakukan selama 21 hari, didapatkan hasil bahwa *port* yang menjadi perhatian penyerang adalah port RDP (3389), HTTPS (443), dan squid (3128) namun serangan *malware* yang terjadi sangat minim [3].

Penelitian yang dilakukan di *university of* ostrava (Republik Ceko) menjalankan *honeypot* untuk mempelajari berbagai serangan yang terjadi terhadap layanan yang dijalankan sistem operasi microsoft windows. Terdapat 6 (enam) *honeypot* yang dijalankan pada *virtual private server* (VPS) dan terhubung langsung dengan jaringan publik. Data yang dikumpulkan selama 1 tahun (februari 2015 – februari 2016) menunjukkan bahwa serangan paling banyak terjadi pada protokol *server message block* (SMB) yaitu sebanyak 3.395.877 koneksi dan terdapat 1.482.308 file yang dikirimkan penyerang. *File* yang berhasil dikumpulkan kemudian dianalisis menggunakan virustotal API dan ditemukan bahwa 88,4 % dari *file* tersebut adalah *malware* dengan jenis conficker [4].

Dari beberapa penelitian yang sudah dipaparkan sebelumnya, dapat disumpulkan bahwa *honeypot* digunakan untuk mendeteksi serangan dan *malware* pada jaringan. Hasilnya didapati bahwa *honeypot* dapat mengumpulkan

malware secara efektif melihat banyaknya *file* yang berhasil di*download* dan terbukti sebagai *malware* setelah dilakukan deteksi menggunakan virustotal API. Namun, permasalahan yang ditemukan adalah belum adanya sistem maupun aplikasi yang terintegrasi dengan *honeypot* yang dapat digunakan untuk melakukan deteksi dan analisis terhadap *malware* yang ditemukan. Analisis terhadap *malware* diperlukan untuk menentukan langkah yang akan diambil untuk mencegah serangan *malware* tersebut. Untuk mengatasi permasalahan tersebut, peneliti merancang aplikasi deteksi *malware* yang terintegrasi dengan *honeypot*. Aplikasi yang akan dibuat menggunakan *collective intelligence framework* (CIF) dalam melakukan proses deteksi dan analisis *malware*. *Collective intelligence framework* (CIF) memungkinkan *malware* dianalisis menggunakan *threat intelligence* dari berbagai sumber untuk memberikan informasi yang lengkap dan akurat.

1.2 Rumusan Masalah

Dari permasalahan yang sudah dikemukakan diatas, maka permasalahan yang dapat dirumuskan adalah bagaimana *collective intelligence framework* (CIF) dapat meningkatkan kemampuan deteksi *malware* pada *honeypot*.

1.3 Batasan Masalah

Mengingat pembahasan mengenai *malware* dan *honeypot* sangatlah luas, maka pada penelitian ini peneliti menentukan batasan-batasan masalah sebagai berikut :

1. *Malware* diperoleh dengan cara menghubungkan *honeypot* kedalam jaringan publik telkom Indonesia menggunakan *ip address* publik dinamis.
2. Analisis *malware* yang dilakukan adalah analisis permukaan terhadap *malware* yang berhasil diunduh.
3. *Honeypot* yang digunakan adalah dionaea yang digunakan untuk meng-*capture malware*.
4. Bahasa pemrograman yang digunakan untuk mengembangkan aplikasi adalah php, html, dan bash.
5. Aplikasi yang akan dibuat adalah aplikasi berbasis *web*.
6. *Server* yang digunakan untuk menjalankan *CIF server* adalah *virtual machine (VM)* dengan sistem operasi linux ubuntu server 16.04.3 yang dijalankan pada Vmware Workstation 14.
7. *Server* yang digunakan untuk menjalankan menjalankan *honeypot* dan aplikasi adalah *single board computer (SBC)* Orange Pi Zero dengan sistem operasi linux ubuntu server 14.04.5.
8. *Threat intelligence* yang diaktifkan pada *server CIF* adalah *threat intelligence* yang menyediakan data *malware* yaitu emergingthreats.net, malc0de.com, vxvault.net, dan alienvault.com.
9. Layanan yang digunakan sebagai pembanding kemampuan deteksi *malware* adalah virustotal API.

1.4 Maksud dan Tujuan Penelitian

Penelitian ini dimaksudkan untuk meningkatkan kemampuan *honeypot* dalam mendeteksi dan menganalisis *malware*, dengan tujuan :

1. Membuat aplikasi yang mengintegrasikan *collective intelligence framework (CIF)* dengan *honeypot.*
2. Mengetahui kemampuan *collective intelligence framework (CIF)* dalam mendeteksi dan menganalisis *malware.*
3. Turut serta dalam pengembangan ilmu pengetahuan khususnya di bidang teknologi informasi (TI).

1.5 Metode Penelitian

Peneliti menjabarkan cara-cara memperoleh data-data yang digunakan untuk kebutuhan penelitian.

1.5.1 Pengumpulan Data

1.5.1.1 Observasi

Observasi dilakukan dengan cara memasang *honeypot* pada jaringan publik/internet agar dapat diserang oleh penyerang dari berbagai negara. Tujuan dari observasi ini adalah untuk mendapatkan data *ip address* penyerang serta mendapatkan salinan *malware* yang dikirimkan oleh penyerang. Data yang didapatkan digunakan untuk keperluan analisis.

1.5.1.2 Deskriptif

Metode deskriptif digunakan untuk memberikan gambaran yang jelas mengenai data yang diperoleh dari proses observasi. Data yang akan disajikan adalah data statistik serangan dan data perolehan *malware* dalam kurun waktu observasi. Data tersebut digunakan untuk menentukan desain serta fitur dari aplikasi yang akan dibuat dalam penelitian ini.

1.5.1.3 Studi Pustaka

Metode studi pustaka dilakukan untuk mengetahui fakta yang diperoleh dari penelitian-penelitian yang sudah dilakukan sebelumnya terkait dengan *honeypot* dan *malware*. Studi pustaka dilakukan dengan membaca penelitian maupun publikasi terkait dengan *honeypot* dan *malware* yang dipublikasikan secara *online* di situs ieeexplore.ieee.org serta situs yang menyediakan dokumentasi mengenai *software* yang digunakan dalam penelitian ini.

1.5.2 Metode Pengembangan Aplikasi

Metode yang digunakan dalam pengembangan aplikasi mengacu pada metode *System Development Life Cycle* (SDLC) yang terdiri dari beberapa proses diantaranya :

1. *Planning*

   Pada tahap ini peneliti melakukan persiapan dan perencanaan awal pengembangan sistem seperti analisis masalah dan penjadwalan.

2. *Analysis*

   Menganalisis kebutuhan *hardware, software, network*, dan kebutuhan fungsional.

3. *Design*

   Melakukan desain aplikasi,*user interface,* dan desain jaringan.

4. *Implementation*

   Mengimplementasikan desain aplikasi kedalam kode program, menjalankan aplikasi yang sudah dibuat kedalam *server*, dan melakukan pengujian.

5. *Maintenance*

   Melakukan pengawasan dan perbaikan jika ditemukan permasalahan pada aplikasi maupun *server*.

1.6 Sistematika Penulisan

Penyusunan sistematika penulisan bertujuan untuk mempermudah pembaca dalam mengetahui garis besar hal yang dipaparkan dalam laporan tugas akhir skripsi ini. Sistematika penulisan yang digunakan dalam penulisan laporan penelitian ini adalah sebagai berikut :

BAB I PENDAHULUAN

Bab ini menguraikan latar belakang masalah, rumusan masalah, batasan masalah, maksud dan tujuan penelitian, metode penelitian serta sistematika penulisan.

BAB II LANDASAN TEORI

Bab ini menguraikan tinjauan pustaka dan dasar teori yang akan digunakan dalam analisis, perancangan, dan pembuatan aplikasi.

BAB III ANALISIS DAN PERANCANGAN

Bab ini menguraikan tentang analisis masalah, analisis kebutuhan sistem, dan perancangan *server* serta aplikasi.

BAB IV IMPLEMENTASI DAN PEMBAHASAN

Bab ini akan membahas implementasi, hasil, serta analisis dari aplikasi yang sudah dibuat dan diimplementasikan.

BAB V PENUTUP

Bab ini berisi kesimpulan dan saran yang didapat dari pembuatan aplikasi ini yang dapat menjadi masukan bagi penelitian selanjutnya.

DAFTAR PUSTAKA