Jurnal Teknologi 9 (1) 2021 80-91

JURNAL TEKNOLOGI

http://jurnalftijayabaya.ac.id/index.php/JTek

DOI: https://doi.org/10.31479/jtek.v9i1.119

pISSN 1693-0266
eISSN 2654-8666

Implementasi *Honeypot* Menggunakan *Dionaea* dan *Kippo* sebagai Penunjang Keamanan Jaringan Komunikasi Komputer

Adi Widiatmoko Wastumirad[*] dan Moh. Irzam Darmawan

Sekolah Tinggi Meteorologi Klimatologi dan Geofisika, Tangerang Selatan 15221

*) *Corresponding author*: widiatmokoadi@gmail.com

(*Received*: *21 Oct 2021* • *Revised*: *17 Nov 2021* • *Accepted*: *27 Nov 2021*)

Abstract

Today, the internet has become the most used tool for delivering information. Through the internet, people can search for information by freely accessing a web page. This freedom of access often raises security issues in the website provider's internal network. These security issues can be in the form of misuse of information, threats, and other attacks on the provider's internal network. Based on these conditions, a technique is needed to protect critical data on the website owner's server from various attacks. In this research, a Honeypot security system has been implemented using Dionaea and Kippo in the Demilitarized Zone to increase the security of a network. The methodology of this research is Waterfall Model for software engineering. The system that has been built is can detect, take action, record attack logs and display them in the form of a website in real-time.

Abstrak

Di masa sekarang, internet sudah menjadi media penyampaian informasi yang paling utama. Melalui internet, masyarakat dapat mencari informasi dengan mengakses suatu halaman *website* dengan bebas. Kebebasan akses ini seringkali menimbulkan isu keamanan pada jaringan internal penyedia *website*. Isu keamanan tersebut dapat berupa penyalahgunaan informasi, ancaman, dan serangan lainnya pada jaringan internal penyedia. Berdasarkan kondisi tersebut, dibutuhkan sebuah cara untuk melindungi data-data penting pada server pemilik *website* dari berbagai serangan. Pada penelitian ini, telah diimplementasikan sebuah sistem keamanan *Honeypot* menggunakan *Dionaea* dan *Kippo* pada *Demilitarized Zone* untuk meningkatkan keamanan suatu jaringan. Metodologi penelitian ini menggunakan *Waterfall Model* untuk rekayasa perangkat lunak. Sistem yang telah dibangun ini mampu mendeteksi, melakukan tindakan, mencatat *log* serangan dan menampilkannya ke dalam bentuk *website* secara *real time*.

Keywords : *Network Security System, Kippo, Dionaea, Honeypot, Demilitarized Zone.*

PENDAHULUAN

Di masa sekarang, hampir setiap orang membutuhkan agar komputer yang digunakan terhubung dengan jaringan internet. Terhubungnya suatu komputer dengan suatu jaringan eksternal atau internet tentu memiliki potensi ancaman keamanan yang lebih besar dibandingkan dengan komputer yang tidak terhubung. Pada dasarnya, keamanan jaringan komputer memiliki sifat berbanding terbalik dengan akses jaringan dimana apabila akses jaringan semakin mudah maka keamanan jaringan komputer akan semakin rentan [1]. Untuk mengatasi permasalahan tersebut, penelitian ini mengangkat model sistem keamanan kombinasi dari *Instrusion Prevention System* (IPS) dan *Intrusion Detection System* (IDS). IPS merupakan aplikasi untuk memonitor lalu lintas jaringan, melakukan pencegahan dini terhadap intrusi atau kejadian yang dapat membuat jaringan menjadi berjalan tidak sebagaimana mestinya, dan mendeteksi aktivitas yang mencurigakan [2]. IDS adalah suatu sistem aplikasi yang dapat memonitor lalu lintas jaringan melalui aktivitas paket data yang mencurigakan atau melanggar aturan (*rules*) keamanan jaringan dan kemudian mampu mencatat dari aktivitas jaringan tersebut [3].

Pada penelitian sebelumnya, Cahyanto telah melakukan implementasi *Honeypot Dionaea* menggunakan mesin virtual dengan distro *Honey Drive* [4]. Kemudian, Agustino telah melakukan implementasi *Honeypot* yang dapat mengalihkan penyerang dengan menjadikan *Honeypot* seolah-olah merupakan server yang sesungguhnya sehingga menjadi tempat berinteraksi bayangan bagi penyerang yang ingin melakukan serangan ke dalam layanan *Cloud Computing* [5]. Selanjutnya, Asadullah telah mengimplementasikan *Honeypot* yang bertugas untuk menjebak penyerang ke dalam ruang palsu dan penggunaan IDS untuk melacak data penyerang yang masuk ke *Honeypot* [6]. Nugroho telah mengimplementasikan *Kippo Honeypot* sebagai sistem keamanan jaringan komunikasi komputer dengan perancangan *System Development Life Cycles* (SDCL) yang dibangun menggunakan metode *waterfall* [7]. Abdillah telah mengimplementasikan sebuah sistem keamanan menggunakan *Demilitarized Zone* (DMZ) dan *Hybrid Intrusion Detection and Prevention System* (IDPS) untuk keamanan LAN [8].

Dibandingkan dengan penelitian-penelitian sebelumnya, pada penelitian ini dilakukan implementasi *Honeypot* menggunakan *Dionaea* dan *Kippo* berbasis *Demilitarized Zone* (DMZ) ke dalam sistem atau server. Tujuannya adalah untuk membangun suatu sistem kemanan jaringan yang dapat sekaligus melakukan monitoring, pencegahan atau pengalihan, pendeteksian, dan pelaporan terhadap aktivitas jaringan. Sistem *Honeypot* yang dibangun dalam penelitian ini digunakan untuk mendeteksi serangan *brute force* menggunakan *Kippo* dan mendeteksi serangan *malware* menggunakan *Dionaea*, serta melakukan tindakan pencegahan atau pengalihan terhadap serangan menggunakan jaringan DMZ. Selain itu, pada jaringan ini juga dilakukan pemotongan jalur komunikasi ke jaringan internal sehingga virus, trojan, dan sejenisnya tidak dapat lagi memasuki jaringan [9].

Honeypot adalah suatu sistem yang dibuat mirip dengan server yang sesungguhnya sehingga dapat berpura-pura menjadi sasaran nyata serangan. *Honeypot* ini bertujuan untuk menjadi pengalih perhatian dari penyerang dan mampu mengambil informasi tentang serangan yang terjadi serta informasi penyerang [10]. *Honeypot* diimplementasikan menjadi sebuah sistem yang menjadi sistem tiruan dengan tujuan untuk menarik perhatian, mendeteksi, dan memeriksa serangan yang terjadi dan dilakukan oleh penyerang [11]. *Honeypot* tidak bekerja untuk mencatat traffic yang legal. Yang berinteraksi dengan *Honeypot* hanyalah user yang menggunakan sumber daya sistem secara ilegal. *Honeypot* akan mengambil alih sebagai sistem yang berhasil disusupi oleh penyerang dan bertindak seolah-olah merupakan server yang sesungguhnya. Pada implementasi *Honeypot*, penyerang tidak akan berhasil masuk ke sistem yang sesungguhnya, tetapi masuk ke sistem *Honeypot* yang bertindak sebagai server palsu [12]. Hal ini karena *Honeypot* dirancang untuk dapat menyerupai jaringan infrastruktur pada server yang sesungguhnya.

Dionaea merupakan jenis *Honeypot* yang dapat membuat beberapa tiruan *service* seperti HTTP, SMB, FTP, TFTP, SIP, dan MySQL serta secara otomatis juga dapat mendeteksi *malware*.

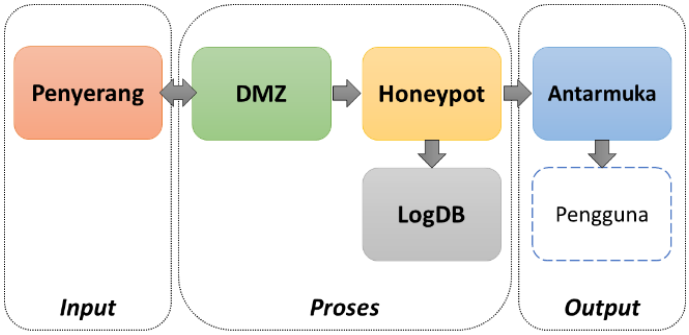
Dionaea dapat mengambil *malware* tersebut sebagai bahan analisis dan memiliki sebuah *database* untuk menyimpan aktivitas pada jaringan. *Kippo* adalah jenis *Honeypot* yang dapat membuat service SSH tiruan. Pengguna yang masuk pada SSH akan mendapatkan hak penuh terhadap sistem tiruan. *Kippo Honeypot* juga dapat berinteraksi secara pasif dengan penyerang. *Kippo* dapat mencatat *command shell* yang di lakukan oleh penyerang. *Honeypot* jenis ini juga mampu mencatat serangan yang bertujuan untuk mendapatkan username dan password yang digunakan penyerang. Dengan dikombinasikannya *Dionaea* dan *Kippo* pada sistem *Honeypot*, sistem pengaman pada layanan yang berhubungan dengan jaringan internet akan mengalami peningkatan.

METODE PENELITIAN

Penelitian ini dibangun menggunakan *Waterfall Model* untuk rekayasa perangkat lunak. Berdasarkan model tersebut, dilakukan pendekatan sistematis dan linier yang dimulai dari kajian kebutuhan pengguna dan kemudian diproses melalui perencanaan, pemodelan, pembangunan, dan pemasangan perangkat lunak [13]. Pada tahap kajian kebutuhan dilakukan analisis kebutuhan sistem dan pengumpulan data pendukung sistem. Hasil analisis kebutuhan sistem ini dapat dirinci menjadi tiga bagian, yaitu data masukan berupa data serangan yang diperoleh dari simulasi, data keluaran berupa informasi hasil pendeteksian serangan, dan kebutuhan antarmuka pada sistem untuk memberikan kemudahan dan kenyamanan bagi admin pada saat mengakses sistem. Pada tahap perencanaan dan pemodelan dilakukan perancangan perangkat lunak untuk antarmuka sistem menggunakan sistem operasi *Ubuntu Server* dan menggunakan *Kippo-Graph* dan *DionaeaFR*. Pada tahap pembangunan dan pemasangan dilakukan implementasi perangkat lunak pada perangkat serta dilakukan pengujian kinerja sistem.

Blok Diagram Sistem

Sistem secara keseluruhan terdiri atas tiga elemen utama yaitu: input, proses, dan *Output*. Penjelasan blok diagram sistem pada Gambar 1 dapat dijabarkan sebagai berikut :



Gambar 1. Blok diagram sistem

a. Input

Pada bagian input dilakukan simulasi penyerangan terhadap server dengan menggunakan teknik *port scanning*, *brute force*, dan *DoS attack*. *Port scanning* adalah teknik untuk mendeteksi port yang terbuka pada suatu jaringan. *Brute force* adalah teknik untuk mendapatkan akses ke suatu jaringan dengan cara menguji daftar *user* dan *password* pada pintu masuk jaringan tersebut. *DoS attack* adalah teknik memberikan gangguan pada suatu jaringan sehingga sumber daya dan layanan pada jaringan tersebut tidak dapat digunakan.

b. Proses

Bagian proses berisikan tindakan-tindakan yang bekerja ketika terjadi penyerangan terhadap server. Ketika terjadi serangan terhadap server dilakukan penyaringan paket data

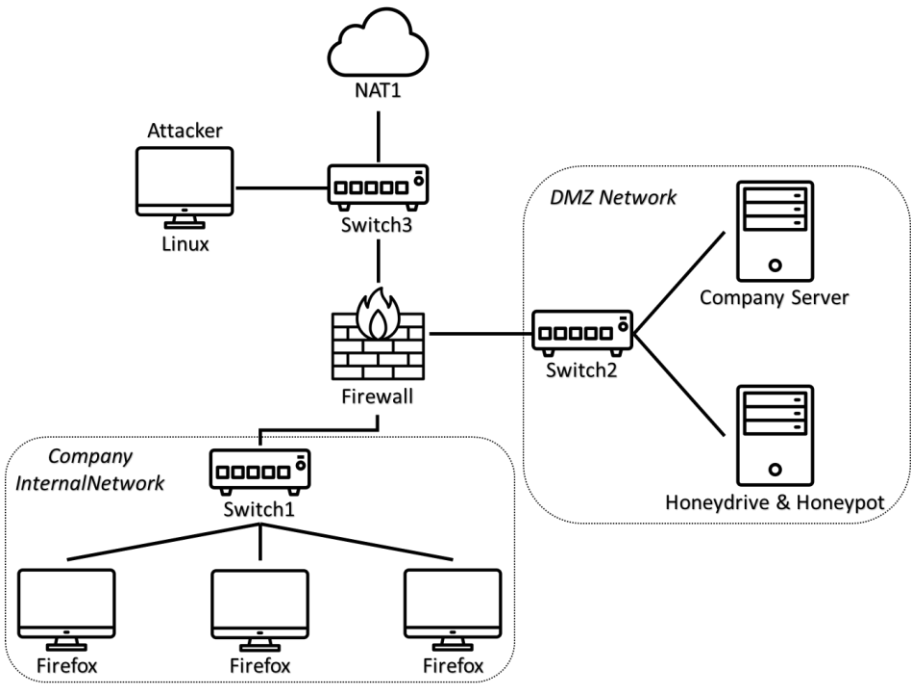
berdasarkan *RulesDB* untuk mengecek lalu lintas data pada lingkup jaringan. *RulesDB* adalah aturan-aturan yang tersimpan pada *firewall*, yang memberikan izin pada arus lalu lintas data dan memberikan izin akses kepada server. Serangan pada server diarahkan menuju jaringan DMZ yang terdapat *Honeypot* sebagai server bayangan. *Honeypot* kemudian akan menggali informasi data dan tujuan dari serangan tersebut serta menahannya di dalam sistem *Honeypot* sehingga jaringan internal dapat terlindungi. Semua aktivitas serangan tersimpan di dalam *LogDB Honeypot* untuk ditampilkan pada *website*.

c. *Output*

Bagian *output* berupa tampilan antarmuka dari hasil penyerangan terhadap server yang bersumber dari *LogDB* pada *Dionaea* dan *Kippo*. Bentuk tampilannya berupa *command line* dan informasi pada *website*.

Skema Topologi

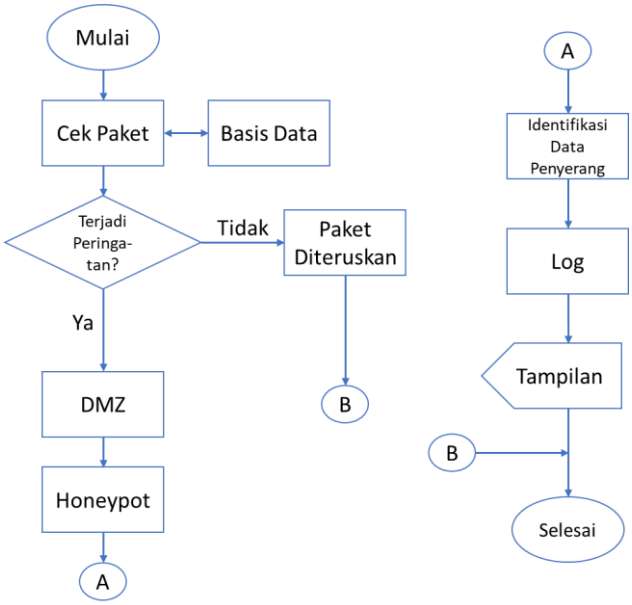
Topologi jaringan yang diangkat dalam penelitian ini menggunakan *firewall* sebagai sistem keamanan pertama seperti ditunjukkan pada Gambar 2. *Firewall* adalah sebuah sistem yang terdapat didalam sistem komputer yang dapat melindungi komputer yang terkoneksi ke jaringan komputer dari berbagai macam gangguan dan ancaman dari pengguna yang tidak bertanggung jawab. *Firewall* dapat digunakan untuk melindungi data dan informasi yang bersifat pribadi yang terhubung dengan internet agar tidak dapat diakses oleh pihak yang tidak berhak atau tidak bersangkutan. *Firewall* dapat melakukan pemblokiran apabila terdapat percobaan akses oleh pihak yang tidak bersangkutan [14]. Pada jaringan DMZ terdapat *company server* sebagai server yang sesungguhnya dan *Honeypot* sebagai server bayangan. Berdasarkan aturan pada *firewall*, serangan dari PC penyerang akan diarahkan menuju jaringan DMZ, kemudian serangan ke *company server* akan diarahkan menuju *Honeypot* sehingga data pada server yang sesungguhnya dapat terlindungi.



Gambar 2. Skema Topologi Jaringan

Diagram Alir Sistem

Diagram alir pada Gambar 3 merupakan algoritma sistem yang digunakan dalam penelitian ini. Proses sistem dimulai dengan pengecekan paket data sesuai database dan membandingkan dengan aturan yang sudah tersimpan dalam database aturan data untuk melakukan penyaringan lalu lintas data. Aturan lalu lintas data disesuaikan dengan tiga jenis serangan yang sudah dijabarkan sebelumnya. Jika paket data terdeteksi memiliki kecocokan dengan tiga jenis serangan tersebut, maka sistem akan memberikan peringatan adanya serangan terhadap server dan serangan tersebut oleh *firewall* akan diarahkan ke jaringan DMZ yang di dalamnya terdapat *Honeypot*. Setelah itu, *Honeypot* akan membuat penyerang terjebak di dalam server bayangan.



Gambar 3. Diagram alir sistem

Selanjutnya, dilakukan uji coba serangan ke server dengan menggunakan *port scanning*, *brute force*, dan *DoS attack*. Pada bagian ini, *Honeypot* akan berinteraksi dengan penyerang untuk mendapatkan informasi yang dicari penyerang dan mengidentifikasi data penyerang, aktivitas terhadap *Honeypot* akan terekam pada log database *Honeypot*. Informasi dan data dari log *Honeypot* ini kemudian diolah menjadi tampilan antarmuka yang ditampilkan pada *website* yang dibangun menggunakan aplikasi *Kippo-Graph*, dan *DionaeaFR*. Jika paket data yang lewat tidak sesuai aturan basis data, maka lalu lintas data akan diteruskan ke jaringan dan sistem selesai.

Implementasi Sistem

Sistem operasi *Ubuntu Server* digunakan sebagai server yang sesungguhnya pada jaringan DMZ. Pada *Ubuntu Server* dilakukan konfigurasi agar beberapa *port* pada sistem operasi *Ubuntu Server* tidak dapat dideteksi oleh penyerang. Sistem operasi *HoneyDrive* digunakan sebagai server *Honeypot* untuk menjebak penyerang yang akan melakukan serangan atau *penetration testing* ke company server. *Penetration Testing* definisi dalam modul *Certified Ethical Hacker* (CEH) adalah metode evaluasi keamanan sistem jaringan atau komputer dengan mensimulasikan serangan dari sumber yang berbahaya dan merupakan bagian dari *security audit*. Simulasi serangan yang dilakukan dirancang seperti kasus yang bisa dibuat oleh *black hat cracker*, *hacker*, dan sebagainya. Tujuannya adalah untuk mengetahui dan menentukan berbagai macam serangan yang mungkin dapat dilakukan pada sistem beserta berbagai akibat yang dapat terjadi karena kelemahan sistem [15].

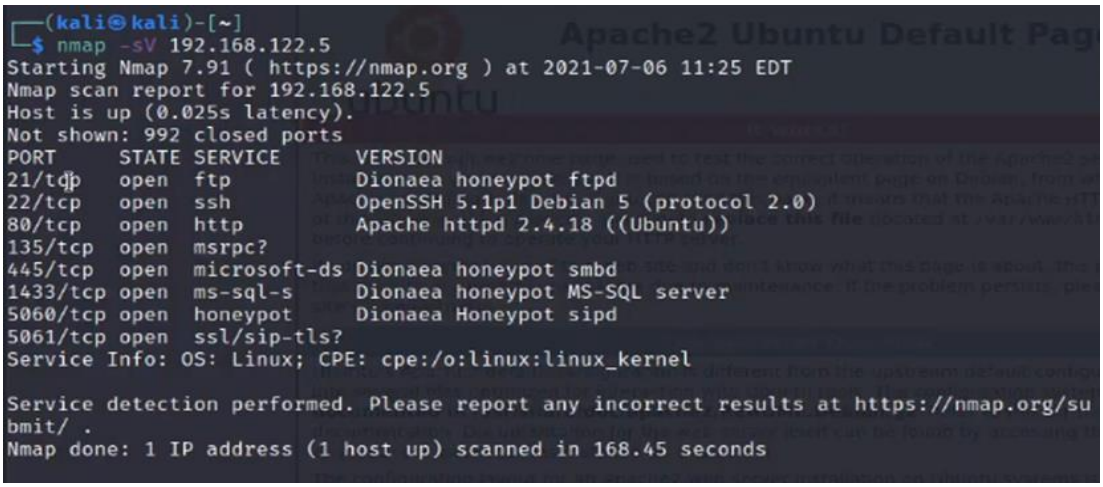
Pada *HoneyDrive* dilakukan konfigurasi *Kippo* dan *Dionaea* sehingga *log* dari serangan dapat tercatat. *Kali Linux* merupakan sistem operasi yang digunakan untuk melakukan serangan dengan *penetration testing* terhadap server. Serangan dilakukan dengan menggunakan *tools nmap*, *bruteforce*, dan *DoS attack*. Perancangan antarmuka dibangun menggunakan aplikasi web *Kippo-Graph* dan *DionaeaFR*. *Kippo-Graph* merupakan tampilan *website* dari *Kippo Honeypot* yang berguna untuk memantau dan mengetahui hasil serangan yang terjadi. *Kippo-Graph* menampilkan hasil *log shell* serangan yang dilakukan oleh penyerang untuk mencari celah agar dapat masuk ke dalam sistem komputer atau server target.

Implementasi sistem dirancang berdasarkan skema topologi jaringan yang telah dibuat. Proses pembuatan sistem *Honeypot* dimulai dari instalasi *Ubuntu Server*, kemudian dilakukan instalasi *HoneyDrive* sebagai sistem operasi *Honeypot* dan dilakukan instalasi *Kali Linux* sebagai perangkat untuk melakukan simulasi serangan ke server. Pada HoneyDrive dilakukan instalasi dan konfigurasi *Kippo Honeypot* dan *Dionaea* agar jika terdapat serangan, *company server* dapat terlindungi dan serangan akan diarahkan menuju *Honeypot*. Terakhir, pada perangkat *Honeypot* dilakukan instalasi *Kippo-Graph* dan *DionaeaFR* sebagai tampilan *website Honeypot*.

HASIL DAN PEMBAHASAN

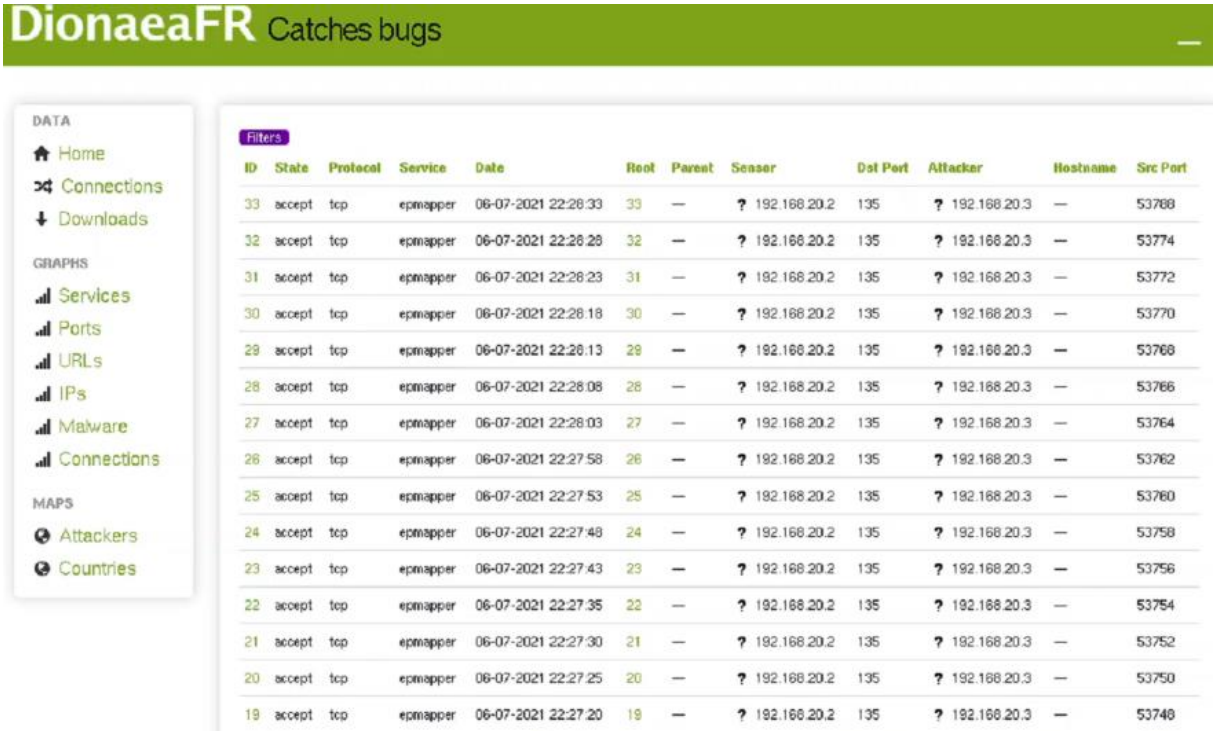
Port Scanning

Serangan *port scanning* merupakan salah satu *tools* serangan dari *penetration testing*. Serangan *port scanning* dilakukan dengan menggunakan *tools nmap* pada Kali Linux. Gambaran umum metodenya adalah melakukan *port scanning* terhadap IP *company server* dengan perintah *nmap*. Tampilan pengujian *nmap* dapat dilihat pada Gambar 4.



Gambar 4. Pengujian *nmap*

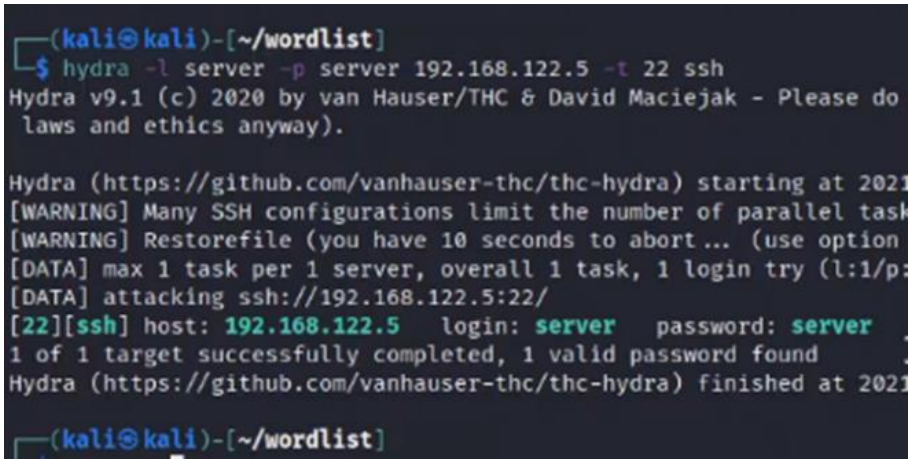
Pada pengujian ini diperoleh hasil bahwa *tools nmap* mampu melakukan *scan* pada alamat IP alias *company server* (192.168.122.5) dan memberikan informasi *port* yang terbuka beserta jenis layanan dari delapan *port*, yaitu *port* 21 *service ftp*, *port* 22 *service ssh*, *port* 80 *service http*, *port* 135 dan *port* 445 *service smb*, serta *port* 1433, 5060, dan 5061. Server *Honeypot* berhasil menipu penyerang dengan memberikan informasi *open port* tersebut kecuali *port* 80 *service http*. Konfigurasi *custom port service ftp* dan *ssh* pada *Ubuntu Server* juga telah berhasil dilakukan sehingga mampu menyembunyikan *service port* tersebut dari serangan dan menggantinya dengan *service port Honeypot*. Aktivitas *port scanning* oleh penyerang dapat dideteksi dan direkam oleh *DionaeaFR*. Tampilan *log DionaeaFR* dapat dilihat pada Gambar 5.



Gambar 5. Respon *Honeypot Dionaea*

Brute Force

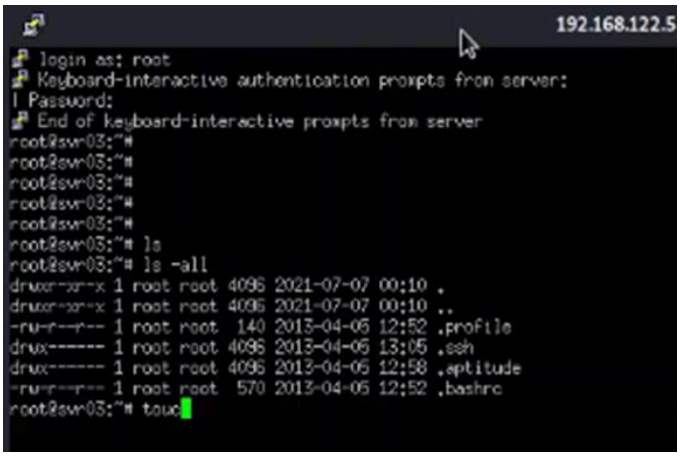
Brute Force dilakukan dengan menggunakan perintah *hydra Kali Linux*. Serangan tersebut bertujuan untuk mendapatkan *login* dan *password* dari IP target. Pada pengujian dilakukan serangan *brute force* terhadap *service port* 22 *ssh*. Pengujian dilakukan dengan mengunduh file *wordlist* pada *Kali Linux* untuk melakukan serangan *brute force*. Kemudian, serangan *brute force* dilakukan dengan memberikan perintah *hydra* menggunakan *wordlist username.txt* dan *wordlist password.txt*. Selanjutnya dilakukan serangan *brute force* dengan perintah: hydra -l server -p server 192.168.122.5 -t 22 ssh.



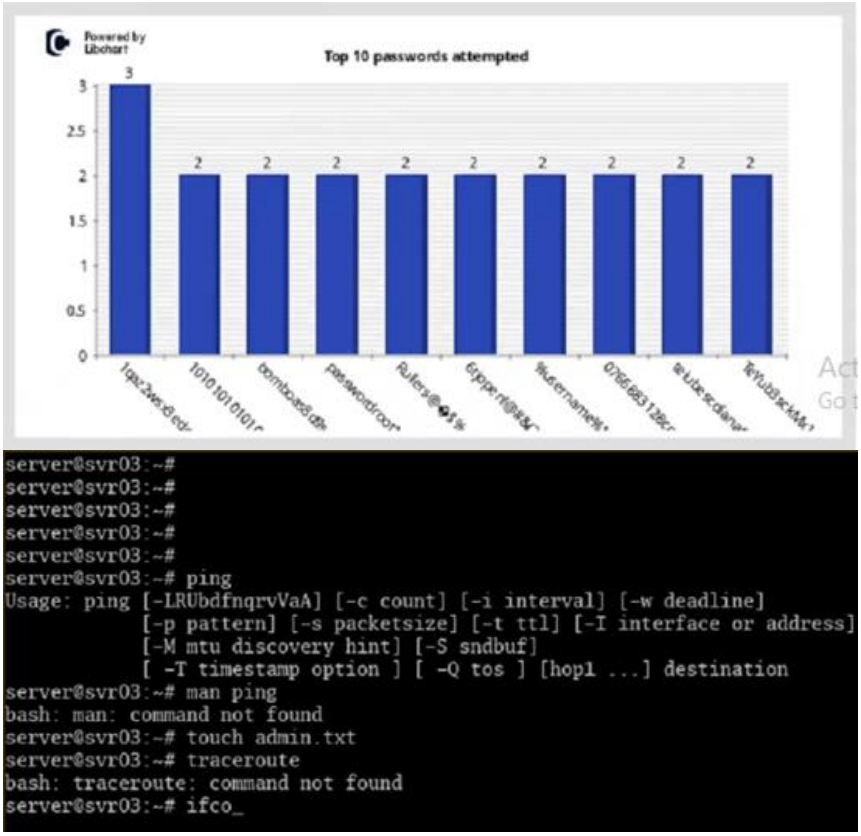
Gambar 6. *Brute force* Hydra

Selanjutnya dilakukan pengujian serangan *brute force* menggunakan *Putty* ke *service port 22*. Serangan IP target 192.168.122.5 pada *port 22* layanan *ssh* menggunakan *Putty* dengan *id user login* adalah *root* dan *password* adalah 123456. Pada aktivitas ini, server *Honeypot* berhasil

merespon dengan baik yaitu penyerang berhasil dialihkan masuk ke server *Honeypot* dan menjebak penyerang dengan cara seolah-olah *Honeypot* menjadi server yang sesungguhnya. Hasil pengujian menggunakan *brute force* terhadap *port 22 service ssh* dapat disimpulkan bahwa sistem keamanan jaringan menggunakan *Honeypot* berjalan sukses karena telah berhasil memberikan *login* dan *password* pengguna untuk masuk ke server *Honeypot* secara acak. Gambar 6 dan Gambar 7 menunjukkan bahwa ketika ada pengguna baru yang mencoba mencari informasi *login* dan *password* pengguna, *Honeypot* meresponnya dengan memberikan informasi *login* dan *password* berupa server/server dan akses *root*/123456, sehingga seolah-olah *port* 22 adalah server yang sesungguhnya padahal akses tersebut adalah *service port* 22 dari *Honeypot*. Pada halaman *Kippo Graph* dan *Kippo-Playlog* juga dapat dimonitor aktifitas serangan yang dilakukan oleh penyerang terhadap server *Honeypot* seperti ditunjukkan pada Gambar 8.



Gambar 7. *Brute force Putty*



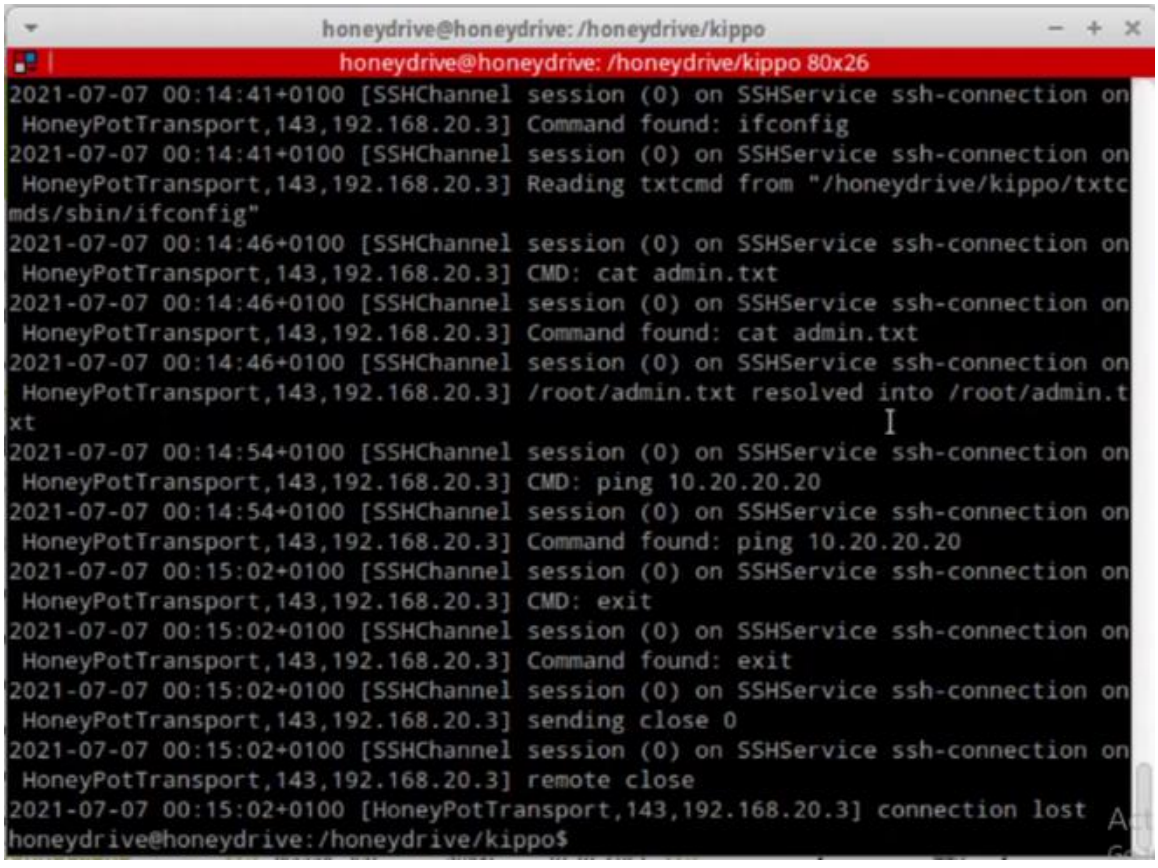
Gambar 8. Respon *Honeypot Kippo*

DoS Attack

DoS attack dilakukan dengan menggunakan *tools hping* pada *Kali Linux* seperti pada Gambar 9.

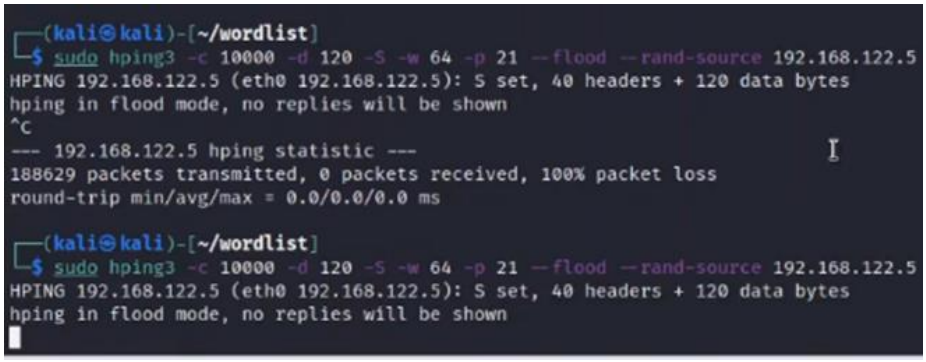
Gambar 9. Perintah *DoS attack*

Dalam pengujian penyerangan *DoS attack* telah berhasil terjadi interaksi dengan *Kippo Honeypot*. Penyerang mencoba melakukan *ping* kemudian *Kippo Honeypot* mencatat *ping* yang terjadi sehingga *Kippo Honeypot* merespon serangan seperti ditunjukkan pada Gambar 10.



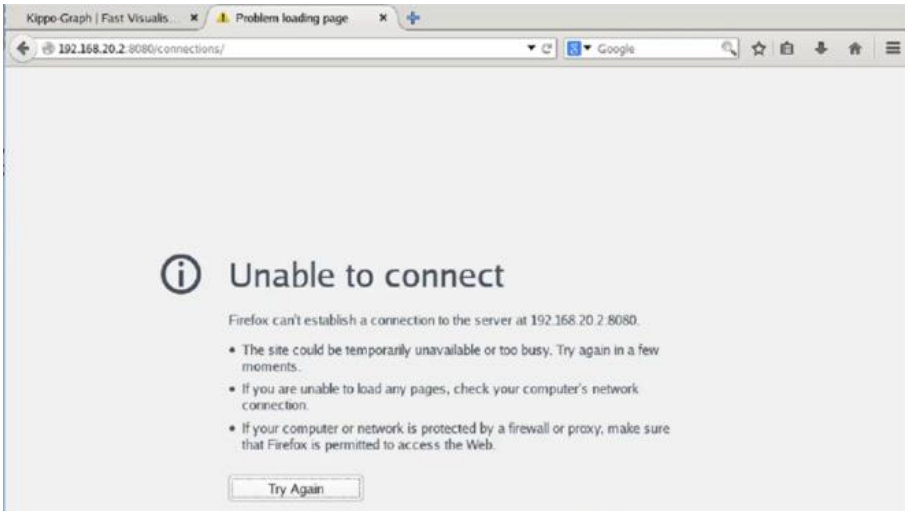
Gambar 10. Respon *Honeypot* pada *DoS attack*

Kemudian dilakukan serangan secara lebih spesifik terhadap paket serangan *DoS* yang akan dilakukan seperti pada Gambar 11.



Gambar 11. Perintah *DoS attack* secara spesifik

Selanjutnya dapat dilihat pada web server *Honeypot* bahwa penyerang telah berhasil melakukan serangan namun tertuju pada *Honeypot* sebagai server bayangan. Layanan yang tampak mengalami ganguan adalah web server *Honeypot*. Respon *Honeypot* disajikan pada Gambar 12.



Gambar 12. Hasil *DoS attack* pada *Honeypot*

Kemudian dapat dilihat bahwa *Ubuntu Server* yang bertindak sebagai server yang sesungguhnya tetap aman dan terhindar dari serangan seperti pada gambar 13.

Gambar 13. Kondisi web *Ubuntu Server*

KESIMPULAN

Kontribusi dalam penelitian ini adalah pengimplementasian *Honeypot* menggunakan *Dionaea* dan *Kippo* pada sistem keamanan jaringan. Sistem *Honeypot* yang dibangun telah berhasil mendeteksi lalu lintas serangan *port scanning*, *brute force*, dan *DoS attack*. *Honeypot* juga telah berhasil merekam aktivitas serangan sehingga dapat dilakukan tindakan pencegahan serangan agar informasi yang terdapat pada *company server* dapat terlindungi. Semua aktivitas penyerangan terekam dalam log *Honeypot* dan dapat ditampilkan ke dalam bentuk *website*.

DAFTAR PUSTAKA

[1] Sukirmanto, "Rancang Bangun dan Implementasi Keamanan Jaringan Komputer Menggunakan Metode Intrusion Detection System pada SMP Islam Terpadu PAPB", Skripsi, Universitas Semarang, Semarang, 2013.

[2] D. Stiawan, A. H. Abdullah, and M. Y. Idris, "Characterizing Network Intrusion Prevention System", *International Journal of Computer Applications*, Vol.14 No.1 pp. 11–18, 2011.

[3] R. Alder *et al*, *Snort 2.1 Intrusion Detection*, Second Edition, Rockland, Syngress Publishing, 2004.

[4] T. A. Cahyanto, H. Oktavianto, and A.W. Royan, "Analisis Dan Implementasi *Honeypot* Menggunakan Donaea Sebagai Penunjang Keamanan Jaringan", *Jurnal Sistem dan Teknologi Informasi Indonesia*, Vol.1 No.2 pp. 86–92, 2016.

[5] D. P. Agustino, Y. Priyoatmojo, and N. W. W. Safitri, "Implementasi *Honeypot* Sebagai Pendeteksi Serangan dan Melindungi Layanan Cloud Computing", in *E-Proceedings KNS&I STIKOM Bali*, pp. 196–201, 2017.

[6] M. H. Asadullah, "Sistem Keamanan Server Dengan *Honeypot* Dan Instrusion Detection System (IDS) (Studi Kasus Perusahaan Printing SOMATEX)", Skripsi, Universitas Sebelas Maret, Surakarta, 2019.

[7] A. A. Nugroho, "Analisa dan Implementasi *Honeypot* pada Sistem Keamanan Jaringan Komunikasi Komputer", Skripsi, Sekolah Tinggi Meteorologi Klimatologi dan Geofisika, Tangerang Selatan, 2020.

[8] M. A. Abdillah, "Implementasi Dan Analisa Jaringan Perimeter Menggunakan Demilitarized Zone (DMZ) Dan Hybrid Intrusion Detection And Prevention System (IDPS) Untuk Keamanan LAN", Skripsi, Sekolah Tinggi Meteorologi Klimatologi dan Geofisika, Tangerang Selatan, 2020.

[9] I. Anugrah and R. H. Rahmanto, "Sistem Keamanan Jaringan Local Area Network Menggunakan Teknik Demilitarized Zone", *Jurnal Penelitian Ilmu Komputer, Sistem Embedded & Logic*, Vol.5, pp. 91–106, 2017.

[10] A. F. Nurrahman, "Implementasi Virtual Low-Interaction *Honeypot* Dengan *Dionaea* Untuk Mendukung Keamanan Jaringan", *Diponegoro Journal of Informatics and Technology*, Vol.2 No.4 pp. 28–37, 2013.

[11] A. Amirullah, "Rancang Bangun Sistem Pengidentifikasi Serangan Pada Jaringan Komputer Universitas Hasanuddin", Skripsi, Universitas Hasanuddin, Makassar, 2010.

[12] M. M. Mustofa and E. Aribowo, "Penerapan Sistem Keamanan *Honeypot* dan IDS Pada Jaringan Nirkabel (Hotspot)", *Jurnal Sarjana Teknik Informatika*, Vol.1 No.1 pp. 111–118, 2013.

[13] R. S. Pressman and B. R. Maxim, *Software Engineering: A Practitioner's Approach*, New York, McGraw-Hill Education, 2020.

[14] S. Khadafi, S. Nurmuslimah, and F. K. Anggakusuma, "Implementasi Firewall Dan Port Knocking Sebagai Keamanan Data Transfer Pada Ftp Server Berbasiskan Linux *Ubuntu Server*", *Network Engineering Research Operation*, Vol.4 No.3, pp. 181–188, 2019.

[15] R. Pangalila, N. Agustinus, and A. Justinus, "Penetration Testing Server Sistem Informasi Manajemen dan *Website* Universitas Kristen Petra", *Jurnal Infra*, Vol.3 No.2, 2015.