Jurnal Amplifier November 2024 Vol 14 No 2
P-ISSN 2089-2020 dan E-ISSN 2622-2000
10.33369/jamplifier.v14i2.38240

Penerapan *Honeypot* Sebagai Sistem Keamanan Server Berbasis Linux

Feni Maharani [1*], Toibah Umi Kalsum[1], Hendri Alamsyah[1]

[1]*Program Studi Rekayasa Sistem Komputer Fakultas Ilmu Komputer Universitas Dehasen*,
*E-mail : fenimaharani62@gmail.com

ABSTRAK

Penerapan honeypot sebagai sistem keamanan *server* berbasis *linux* dapat membantu mengamankan *server*, dapat membantu mendeteksi serangan-serangan yang terjadi pada jaringan komputer, serta menampilkan hasil serangan yang terjadi di dalam *log* yang tersimpan pada *cowrie honeypot*. Dengan adanya honeypot dapat dijadikan pengalih perhatian dari penyerang dan mampu mengambil informasi tentang serangan yang terjadi serta informasi penyerang. *Honeypot* diimplementasikan menjadi sebuah sistem yang menjadi sistem tiruan dengan tujuan untuk menarik perhatian, mendeteksi, dan memeriksa serangan yang terjadi dan dilakukan oleh penyerang Berdasarkan pengujian yang telah dilakukan, maka dapat disimpulkan bahwa penerapan *honeypot* pada sistem keamanan server berbasis *linux* di SMK Negeri 3 Bengkulu mampu melakukan manipulasi port seperi *port* 22 dan 23,sehingga dapat menghindari dan mengamanan *port-port* dari serangan seperti *ping of death* yang di uji coba menggunakan aplikasi LOIC, SSH, Telnet, scanner port yang di uji coba menggunakan aplikasi *advanced port scanner*, yang tersimpan di dalam *log cowrie honeypot.*
***Kata kunci*: *Honeypot*, Sistem Keamanan, *Linux*.**

Abstract

The application of honeypot as a linux-based server security system can help secure *servers*, can help detect attacks that occur on computer networks, and display the results of attacks that occur in logs stored on *cowrie honeypots*. With the existence of a honeypot, it can be used as a distraction from the attacker and is able to take information about the attack that occurred and the information of the attacker. *Honeypot* is implemented into a system that becomes a dummy system with the aim of attracting attention, detecting, and examining attacks that occur and are carried out by attackers. Based on the tests that have been carried out, it can be concluded that the application of *honeypot* in the linux-based server security system at SMK Negeri 3 Bengkulu is able to manipulate ports such as *ports* 22 and 23, so that it can avoid and secure *ports* from attacks such as *ping of death* which was tested using the LOIC, SSH, Telnet applications, port scanners which were tested using *the advanced port scanner* application, which is stored in *the Cowrie Honeypot logs*.
***Kata kunci*: Honeypot, Security Sistem, Security System, Linux.**

1. PENDAHULUAN

Banyaknya kemudahan yang didapat oleh pengguna internet menyebabkan teknologi tersebut tumbuh dengan sangat cepat. Hampir semua aspek informasi dapat diperoleh melalui internet mulai dari pendidikan, hiburan, olahraga, pemerintahan, sekolah, dan lain-lain. Internet bisa diakses hampir semua kalangan baik anak-anak maupun dewasa untuk mencari informasi.

Kebutuhan akan jaringan komputer semakin bertambah penting, baik dalam pendidikan, pekerjaan maupun dalam sebuah permainan, salah satu hal penting dalam mengelola jaringan komputer yaitu keamanan dari jaringan itu sendiri, dengan banyaknya akses ke jaringan tersebut maka akan banyak pula peluang kejahatan yang terjadi di dalam jaringan tersebut, misalkan adanya pencurian data yang terjadi di jaringan tersebut ataupun adanya peretas yang mematikan sumber daya jaringan tersebut.

SMK Negeri 3 Kota Bengkulu merupakan salah satu Sekolah Menengah Kejuruan yang terdapat di Kota Bengkulu. Di sekolah tersebut sudah memiliki akses internet dengan menggunakan *provider* lintasarta dengan kecepatan bandwidth sebesar 120 Mbps dengan beberapa peralatan *mikrotik*, *switch*, LAN, *access point*, dan perangkat komputer lainnya. Berdasarkan hasil wawancara yang telah dilakukan sistem keamanan di sekolah saat ini masih menggunakan standar dari bawaan sistem operasi komputer dan belum adanya sistem keamanan yang mengamankan *server* di Sekolah. Oleh karena itu, dibutuhkan suatu sistem keamanan yang dapat membantu pihak sekolah dalam mengamankan *server* di sekolah dari penyerang sehingga dapat menghindari hal-hal yang tidak diinginkan.

Server SMK Negeri 3 Kota Bengkulu digunakan untuk Layanan website, Layanan e-raport, menyimpan data untuk situs internet, menyimpan dokumen maupun informasi mengenai SMK Negeri 3 Kota Bengkulu seperti menerima atau mengirim *email*, menampilkan *website* dan lainnya. Fungsi utama server adalah merespon setiap permintaan dari *clien* agar bisa diperoses baik permintaan data atau aplikasi untuk dijalankan oleh *clien.* Layanan yang terdapat di *server*.

Penelitian yang menerapkan jaringan poin to poin yang tidak pada jaringan asli untuk menguji sistem tersebut. Hasil dari penelitian adalah sistem *honeypot*

dapat merekam dan melabui penyerangan dengan *server* palsu dalam bentuk *file log*, dan pada sistem *Snort* dapat memberikan rekaman trafik yang jangal ke *server* dalam bentuk *file log* atau *elert* [1].

Ada juga penelitian yang menggunakan honeypot berbasis Raspberry pi dan ELK stack dalam memonitoring hasil yang didapatkan oleh honeypot. Penggunaannya tetap menggunakan *honeypot* yang mana pembuatan system mampu mendeteksi serangan pada jaringan. Sensor yang digunakan yaitu Raspberry pi berfungsi sebagai pemantauan ancaman keamanan dengan kelebihan secara ekonomis yaitu hemat biaya dan efektif menggantikan komputer desktop. Kelebihan yang lain yaitu membuat analisis log yang awalnya rumit untuk dianalisis menjadi lebih menarik selain itu ELK stack memudahkan pemusatan data dari berbagai sumber [2].

Dari kedua penelitian diatas yang membedakan dengan penelitian ini adalah bahwah penelitian ini menggunakan *software cowrie* yang berguna untuk mempermudah *installasi* pada *honeypot* dan melakukan penyamaran layanan pada *openssh* server. Konsep yang dipakai *cowrie* adalah pengalihan, yaitu setelah *openssh* berhasil diserang, *cowrie* akan menggarahkan *attacker* untuk masuk pada layanan palsu *honeypot*.

Berdasarkan uraian latar belakang untuk mencegah kejadian tersebut dibutuhkan sebuah sistem keamanan untuk menjaga *server* dari *attacker*. Salah satu cara mengatasinya dengan mengimplementasikan *honeypot* yang digunakan untuk menjadi pengalih perhatian dari penyerang dan mampu mengambil informasi tentang serangan yang terjadi serta informasi penyerang. *Honeypot* diimplementasikan menjadi sebuah sistem yang menjadi sistem tiruan dengan tujuan untuk menarik perhatian, mendeteksi, dan memeriksa serangan yang terjadi dan dilakukan oleh penyerang, maka penulis tertarik untuk mengangkat judul penelitian tentang Penerapan Honeypot Sebagai Sistem Keamanan Server Berbasis Linux.

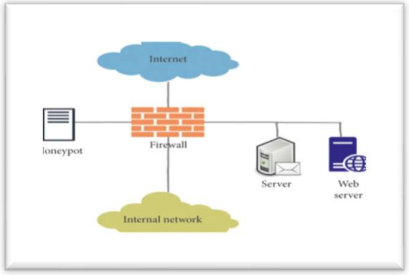
2. KERANGKA TEORITIS DAN PENGEMBANGAN HIPOTESA

A. Kerangka Teoritis

Honeypot merupakan teknologi keamanan yang bertujuan mengidentifikasi, mancari celah keamanan dan berkomprimasi aktif ketika terjadi aktifitas penyusupan keamanan teknologi informasi [3]. *Honeypot* merupakan salah satu upaya dalam membuat sistem palsu yang berguna untuk menjebak penyusup yang mempunyai tujuan buruk untuk mencegah usaha-usaha yang merugikan sistem [4].

Honeypot merupakan salah satu solusi yang dapat diberikan karena *honeypot* merupakan sebuah sistem umpan atau aplikasi simulasi yang mensimulasikan seluru jaringan untuk memikat penyerang dengan meyamarkan diri sebagai sistem yang rentan [5].

Berdasarkan uraian dan penjelasan para ahli diatas dapat diseimpulkan bahwa *honeypot* adalah sebuah sistem layanan palsu yang berfungsi untuk menjebak penyerang. Umumnya seorang penyerang di dunia jaringan mempunyai tujuan buruk yaitu melakukan pencurian atau perusakan data. *Honeypot* digunakan untuk menangkal usaha-usaha yang dapat merugikan sistem atau layanan.



Gambar 1. Honey Deployed Independently

Honeypot ini mengandung kerentanan sistem yang membuatnya menjadi target menarik bagi penyerang. *Honeypot* berguna untuk pengalihan agar penyerang masuk ke *server* palsu dan dapat melihat *log*/aktivitas penyerang terhadap *server* [6]. "Kegunaan *honeypot* lainnya adalah untuk mengetahui metodologi yang digunakan penyerang dalam menguasai sistem dan untuk mengumpulkan informasi sebagai bukti forensik".

Honeypot juga memiliki banyak manfaat, antara lain yaitu mitigasi risiko, berfungsi seperti IDS, untuk mencari tahu strategi serangan yang dilakukan oleh penyerang, mengidentifikasi pelaku penyerangan serta melakukan klasifikasi, sebagai bukti hukum untuk menuntut penyerang, dan bermanfaat untuk riset yang dapat mengetahui teknik dan eksploitasi terbaru [7].

Honeypot terbagi dua jenis berdasarkan kegunaannya yaitu, sebagai berikut *Honeypot produksi* berjenis *low interaction* biasanya digunakan pada *server* produksi oleh suatu organisasi atau perusahaan. Sedangkan *Honeypot* penelitian biasanya digunakan untuk penelitian serangan yang dilakukan komunitas *black hat* di berbagai jaringan. Honeypot penelitian lebih rumit untuk digunakan daripada honeypot produksi karena menangkap informasi lebih banyak dan biasanya digunakan oleh penelitian, militer, danorganisasi pemerintahan [8].

Honeypot memiliki klasifikasi berdasarkan tingkat interaksinya. Tingkat interaksi dapat didefinisikan sebagai tingkat aktivitas penyerang ke dalam sistem *honeypot*, semakin tinggi tingkat aktivitas penyerang maka semakin tinggi pula tingkat interaksi *honeypot* yang harus disiapkan. Berdasarkan interaksinya, terdapat dua jenis *honeypot* yaitu *Honeypot low interaction* menggunakan sistem operasi emulasi yang terpasang pada *honeypot* ketika berinteraksi dengan penyerang [9].

Honeypot low interaction memiliki interaksi yang terbatas kepada penyerang. Serangan yang dihadapi biasanya berupa port scanning dan juga *digital signature attack*.Interaksi pada *honeypot low interaction* dengan *host* lain terbatas sehingga kemampuan yang dimiliki terbatas dan penyerang dapat dengan mudah mengenalinya tetapi dibalik terbatasnya *honeypot low interaction* memiliki resiko yang rendah [10].

Honeypot low interaction juga didesain untuk mensimulasikan lay anan layaknya *server* asli dengan layanan tertentu seperti SSH. Layanan tersebut bukan

sistem operasi secara keseluruhan, layanan yang berjalan tidak bisa dieksploitasi untuk mendapatkan hak akses penuh terhadap honeypot. *Honeypot low interaction* lebih mudah diimplementasikan dan memiliki dampak resiko yang rendah pada jaringan maupun sistem. *Honeypotlow interaction* berfungsi seperti IDS pasif tanpa mengubah jalur lalu lintas jaringan yang ada.

Kelebihan *honeypot low interaction* diantaranya adalah memberikan pengalaman yang baik bagi yang belumberpengalaman dan masih dalam tahap pembelajaran membangun *honeypot*. Kelebihan lainnya adalah untuk *install, deploy*, dan maintenance sangat mudah. Begitu juga dengan analisa pada log yang dihasilkan juga lebih mudah. *Honeypot low interaction* memiliki beberapa kekurangan diantaranya adalah *log* yang dihasilkan sangat terbatas, kemampuan untuk menangkap serangan sudah diketahui sebelumnya, *honeypot low interaction* mudah terdeteksi oleh penyerang yang sudah profesional.

Tujuan dibuatnya *honeypot low interaction* diantaranya adalah untuk mengidentifikasi dan mendeteksi serangan yang dilakukan oleh *tools* otomatis, menipu penyerang yang masih *script kiddies*, mengalihkan serangan yang dilakukan oleh penyerang dari sistem asli, dan mendapatkan modus serangan yang dilakukan oleh penyerang.

Honeypot high interaction menggunakan sistem operasi asli untuk lebih memotivasi penyerang dalam menyerang sistem sehingga strategi maupun modus serangan dapat dicatat dan dianalisis lebih detail. *Honeypot high interaction* mampu memproses dan membedakan antara paket yang bersih dengan paket yang dikirim oleh penyerang sehingga paket tersebut tidak dapat merusak *server* asli.

Kelebihan *honeypot high interaction* diantaranya adalah serangan yang diterima *honeypot high interaction* bisa jadi serangan yang asli dan belum dikenal, ini membuat serangan tersebut bermanfaat untuk dipelajari. Serangan yang diterima mempermudah pengguna *honeypot* untuk mempelajari metode yang digunakan penyerang, dan mencegah serangan pada masa mendatang dan mendapatkan pengetahuan tentang ancaman tersebut.

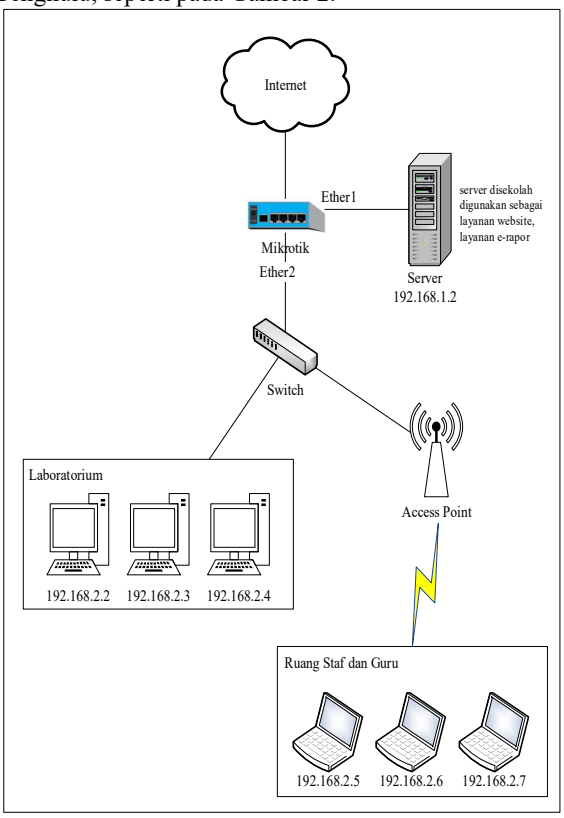
Honeypot high interaction memiliki kekurangan diantaranya untuk membuat, konfigurasi, *deploy, dan maintenance* memakan waktu yang lama karena harus menyesuaikan teknologi yang digunakan seperti *IDS*, *firewall*, dan lain sebagainya, Analisa serangan memakan waktu yang lama, dan Resiko yang dihasilkan oleh *honeypot high interaction* sangat tinggi, jika tidak ada tindakan pencegahan maupun perlindungan tambahan maka dapat merugikan organisasi dari serangan yang dilakukan oleh penyerang.

3. METODE RISET

A. Diagram Global Sistem Lama

Pada SMK Negeri 3 Kota Bengkulu sudah memiliki akses internet dengan menggunakan *provider* lintasarta dengan kecepatan bandwidth sebesar 120 Mbps dengan beberapa peralatan *mikrotik*, *switch*, LAN, *access point*, dan perangkat komputer lainnya. Keamanan pada jaringan komputer sebagai bagian dari sebuah sistem yang sangat penting untuk menjaga validitas dan integritas data serta menjamin ketersediaan layanan bagi penggunanya. Keamanan pada jaringan komputer harus dilindungi dari segala macam serangan dan usaha-usaha penyusupan atau pemindaian oleh pihak yang tidak berhak. Berdasarkan hasil wawancara yang telah dilakukan sistem keamanan di sekolah saat ini masih menggunakan standar dari bawaan sistem operasi komputer dan belum adanya sistem keamanan yang mengamankan *server* di Sekolah. Oleh karena itu, dibutuhkan suatu sistem keamanan yang dapat membantu pihak sekolah dalam mengamankan *server* di sekolah dari penyerang sehingga dapat menghindari hal-hal yang tidak diinginkan.

Jaringan komputer yang terdapat di SMK Negeri 3 Kota Bengkulu menggunakan topologi *star*. Adapun skema jaringan komputer di SMK Negeri 3 Kota Bengkulu, seperti pada Gambar 2.

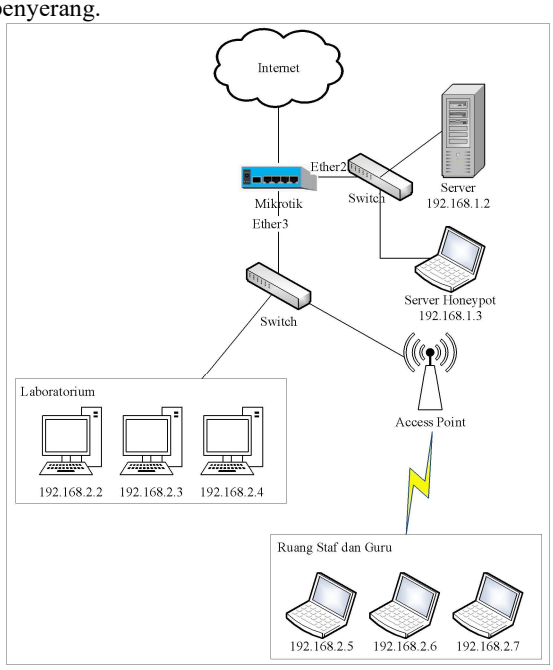


Gambar 2. Skema Jaringan di SMK Negeri 3 Bengkulu

B. Diagram Global Sistem Baru

Untuk mencegah segala macam serangan dan usaha-usaha penyusupan atau pemindaian oleh pihak yang tidak berhak dibutuhkan suatu sistem keamanan untuk menjaga server dari attacker. Salah satu cara mengatasinya dengan mengimplementasikan honepot yang digunakan untuk menjadi pengalih perhatian dari penyerang dan mampu mengambil informasi tentang serangan yang terjadi serta informasi penyerang. Honeypot diimplementasikan menjadi sebuah sistem yang menjadi sistem tiruan dengan

tujuan untuk menarik perhatian, mendeteksi, dan memeriksa serangan yang terjadi dan dilakukan oleh penyerang.



Gambar 3. Skema Jaringan Komputer yang Barui

C. Prinsip Kerja

Penerapan Honeypot yang dibangun ini hanya mengamankan server sekolah, dimana server sekolah digunakan sebagai layanan website dan e-raport. Honeypot bertugas untuk membuat server bayangan yang dapat membantu memanipulasi serangan yang masuk seolah-olah telah menyerang server.

Penerapan honeypot ini tidak mengubah keseluruhan sistem jaringan yang saat ini sedang berjalan, karena yang diamankan adalah server sekolah, sehingga dilakukan penambahan perangkat switch dan laptop yang dijadikan sebagai server bayangan honeypot.

Pada *server* akan di *install* sistem operasi *linux Ubuntu server* versi 22.04. Kemudian *honeypot* di *install* pada sistem operasi *linux Ubuntu server* tersebut. *Honeypot* yang digunkan adalah *crowie low interaction* yang akan membantu mendeteksi serta mengatasi penyerangan yang terjadi kepada *server* asli di sekolah. *Honeypot* hanya menggamankan *server* asli sekolah, dengan cara membuat penyerangan seoplah-olah menyerang *server* asli sekolah, namun pada kenyataannya yang diserang merupakan *server* bayangan dari *honeypot.*

4. HASIL DAN PEMBAHASAN

A. Hasil

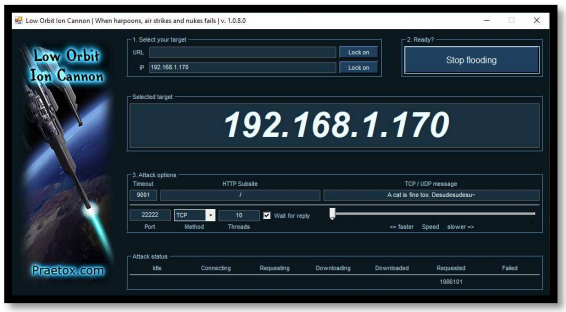
Dari serangkaian pengujian implementasi honeypot sebagai sistem keamanan server berbasis linux Pada SMK N 3 Kota Bengkulu berjalan dengan baik sesuai dengan rancangan dan kegunaan pada SMK N 3 Kota Bengkulu.

B. Serangan Ping Of Death Dalam Jaringan

Untuk melakukan serangan Ping of Death, disini menggunakan aplikasi LOIC, adapun hasil serangan ping of death ke server honeypot tidak dapat dilakukan, seperti dapat dilihat pada tampilan gambar dibawah ini:

Gambar 4. Tampilan Aplikasi LOIC Gagal Melakukan Ping of Death

Dari tampilan diatas dapat dilihat kegiatan Ping of Death menggunakan aplikasi LOIC tidak dapat berjalan ditandai dengan request = 0. Akan tetapi jika sudah di ketahui port yang digunakan maka akan bisa melakukan Ping of Death, seperti gambar dibawah ini:

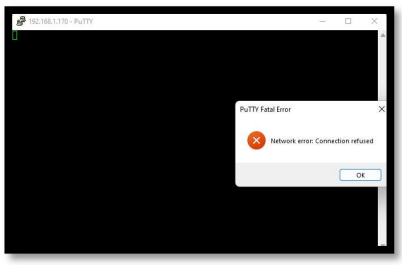


Gambar 5. Tampilan Aplikasi LOIC Berhasil Melakukan *Ping of Death*

Setelah di ketahui port yang digunakan maka ping of death dapat dilakukan, yang ditandai dengan request berjalan sebanyak 1988101, yang arti dapat melakukan pengiriman paket sebesar 1988101.

a. Mengakses Port 22 SSH Ke Server Pada Jaringan

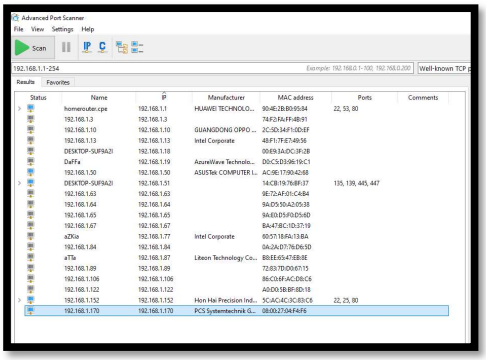
Server yang sudah di install dan konfigurasi *cowrie honeypot,* berhasil melakukan manipulasi port SSH. Sehingga port 22 tidak lagi dapat diakses untuk masuk ke sistem utama server, seperti dapat dilihat pada tampilan gambar dibawah ini:



Gambar 6. Tampilan Gagal Akses Port 22

b. Port Scaning Terhadap Data Server

Server yang sudah di install dan konfigurasi cowrie honeypot, berhasil menghindari atau memperlama proses port scanner, port scanner dilakukan dengan menggunakan aplikasi advanced port scanner, seperti dapat dilihat pada tampilan gambar dibawah ini:

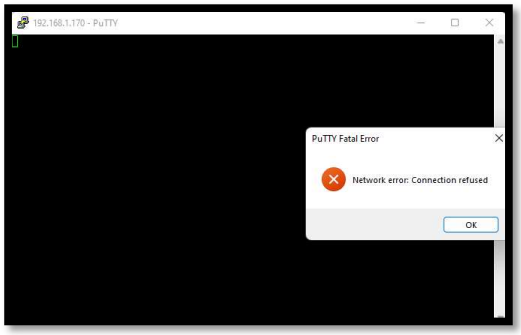


Gambar 7. Tampilan Gagal Melakukan Port Scanner

Dari hasil port scanner diatas server yang sudah di install cowrie honeypot tidak menampilkan port yang terbuka. Sehingga akan mempersulit untuk melakukan hack terhadap server.

c. Mengakses Port 23 Telnet Ke Server Pada Jaringan

Server yang sudah di install dan konfigurasi cowrie honeypot, berhasil melakukan manipulasi port Telnet. Sehingga port 23 tidak lagi dapat diakses untuk masuk ke sistem utama server, seperti dapat dilihat pada tampilan gambar dibawah ini:



Gambar 8. Tampilan Gagal Akses Port 23

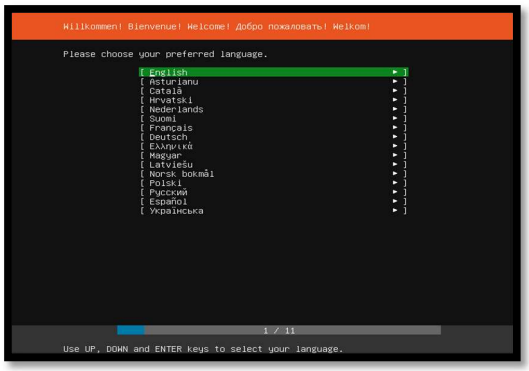
.

4.2 Pembahasan

Pada bab ini membahas tentang implementasi sistem berdasarkan pada analisa dan perancangan yang telah dibuat pada bab sebelumnya, antara lain :

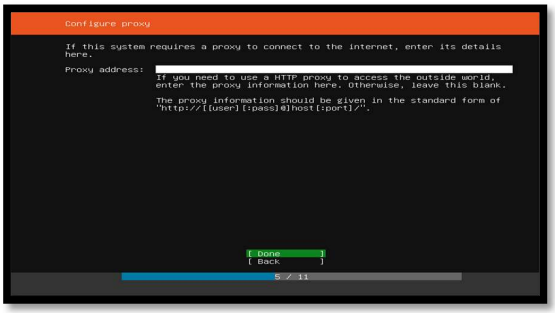
4.2.1 Installasi Linux

Tahap awal dalam Optimalisasi Keamanan Jaringan Menggunakan Firewall Berbasis Linux Pada SMK N 1 Seluma dengan Server linux ubuntu server 20.04 dengan menerapkan Iptable pada squid yang terintegrasi pada webmin dengan menggunakan linux, yaitu melakukan installasi linux, tahap awal installasi linux setelah dilakukan *booting* menggunakan disk installer adalah pilihan Bahasa, seperti gambar dibawah ini:



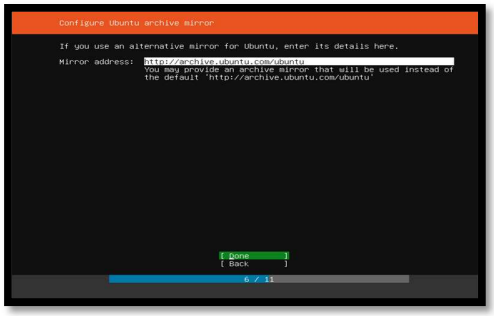
Gambar 9. Tampilan Piihan Bahasa Install Linux

Pada penelitian ini penulis menggunakan Bahasa inggris, setelah dilakukan pilihan Bahasa maka dilanjutkan ke dialog selanjutnya yaitu konfigurasi proxy, seperti gambar dibawah ini:



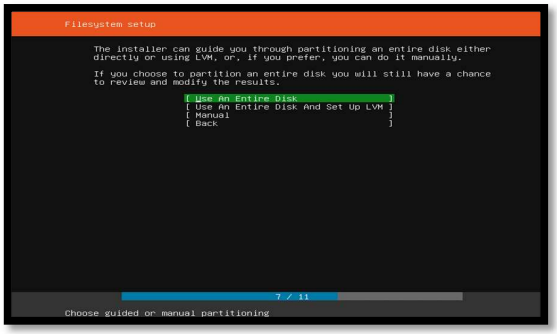
Gambar 10. Tampilan Input Proxy Linux

Pada penelitian ini tidak menggunakan proxy karena jaringan voip digunakan khusus untuk internal SMK N 1 Seluma, selanjutnya pilih done maka dilanjutkan ke dialog selanjutnya yaitu konfigurasi ubuntu aktif (respotary linux), seperti gambar dibawah ini:



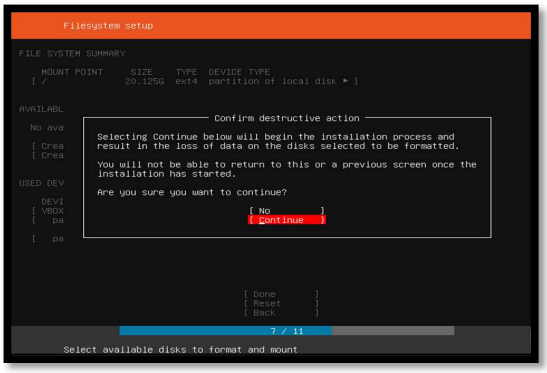
Gambar 11. Tampilan Pilihan Respotary Linux

Respotary linux berfungsi untuk server tujuan Ketika melakukan *update* dan *upgrade linux* secara otomatis, selanjutnya masuk ke dialog selanjutnya yaitu penggunaan harddisk, seperti gambar dibawah ini:



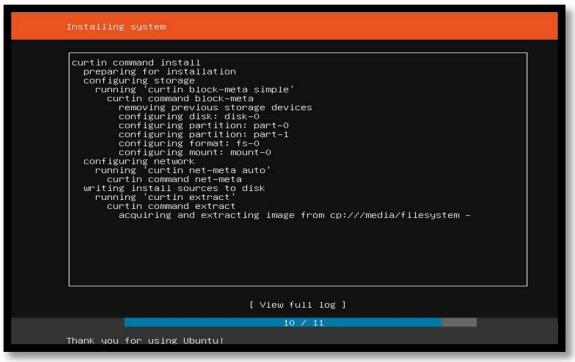
Gambar 12. Tampilan Pilihan HDD Sistem Linux

Disini digunakan seluruh kapasitas hdd, selanjutnya masuk ke dialog selanjutnya yaitu konfirmasi penggunaan hdd, seperti gambar dibawah ini:



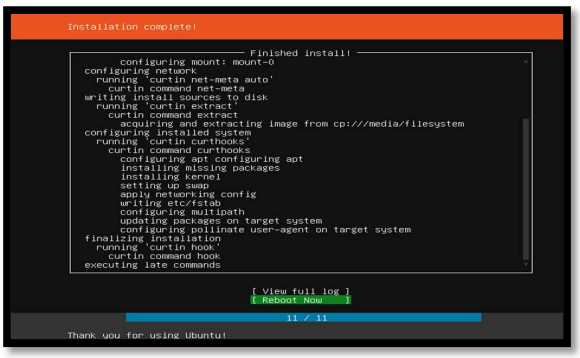
Gambar 13. Tampilan Komfirmasi Pilihan HDD Sistem Linux

Menu diatas merupakan menu konfirmasi penggunaan hdd, selanjutnya masuk ke proses installasi harddisk, seperti gambar dibawah ini:



Gambar 14. Tampilan Proses Install Linux

Setelah proses installasi selesai maka masuk ke dialog senjutnya, seperti gambar dibawah ini:



Gambar 15. Tampilan Install Linux Selesai

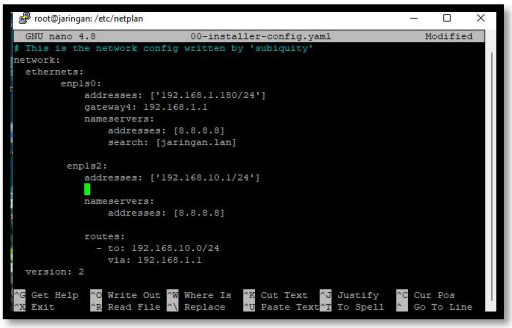
Dialog diatas merupakan pemberitahuan installasi linux selesai dilakukan dan system meminta untuk dilakukan reboot (restart).

4.2.2 Konfigurasi Linux Ubuntu Server 20.04

Konfigurasi linux ubuntu server dilakukan untuk pengaturan IP Address yang digunakan pada masing-masing interface (Ethernet Card) yang digunakan pada linux ubuntu server 20.04, Adapun konfigurasi ini dilakukan melalui terminal linux dengan menggunakan putty, dengan mengetik perintah berikut:

nano 00-installer-config.yaml

perintah diatas merupakan perintah untuk membuka file konfigurasi dengan menggunakan editor linux, sehingga akan tampil file konfigurasi, selanjutnya di isi konfigurasi-konfigurasi yang diperlukan, seperti gambar dibawah ini:

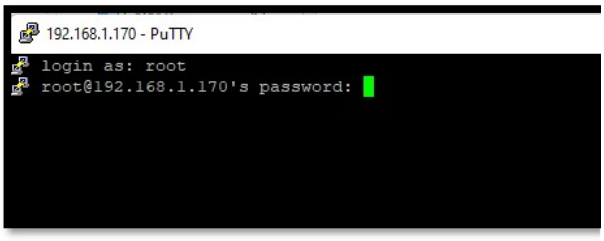


Gambar 15. Tampilan Konfigurasi IP Address

Selanjutnya masukan username dan password login, setelah berhasil melakukan login, maka akan masuk ke prompt ubuntu server 20.04, seperti dapat dilihat pada tampilan gambar dibawah ini:

4.2.3 Install dan Konfigurasi Cowrie Honeypot

Login ke server menggunakan putty, seperti dapat dilihat pada tampilan gambar dibawah ini:



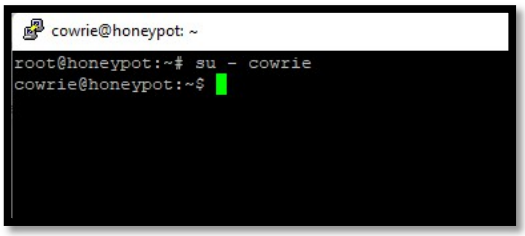
Gambar 16. Tampilan *Login Root*

Selanjutnya buat pengguna khusus untuk menjalankan honeypot Cowrie. Pengguna ini akan digunakan untuk mengisolasi honeypot dari seluruh sistem, dengan menggunakan perintah berikut:

sudo adduser --disabled-password cowrie

selanjutnya login ke cowrie dengan menggunakan perintah berikut ini:

su – cowrie



Gambar 17. Tampilan Login User Cowrie

Selanjutnya lakukan kloning repositori Cowrie dari GitHub dan instal menggunakan lingkungan virtual Python, dengan menggunakan perintah berikut ini:.

git clone http://github.com/cowrie/cowrie

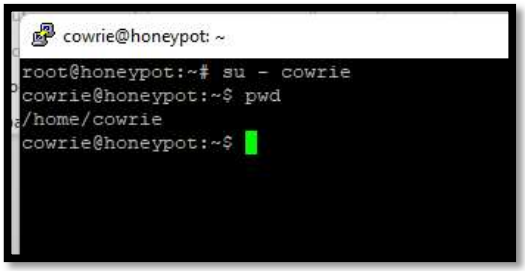
selanjutnya masuk ke folder cowrie, dengan mengetik perintah berikut pada terminal

cd cowrie

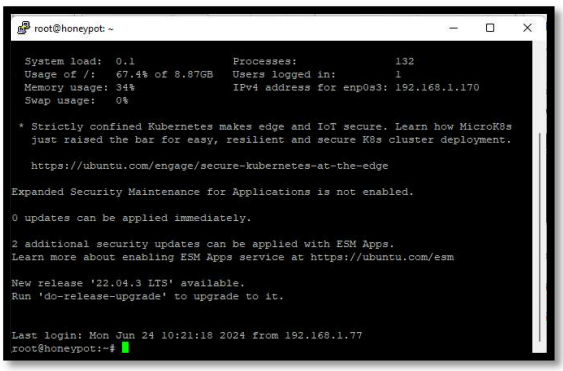
virtualenv --python=python3 cowrie-env

setelah itu perikasa pwd nya, dengan cara mengetik perintah berikut pada terminal:

pwd

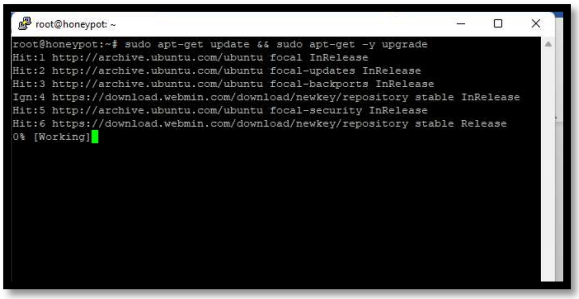


*Gambar*19. Tampilan pwd *Cowrie*



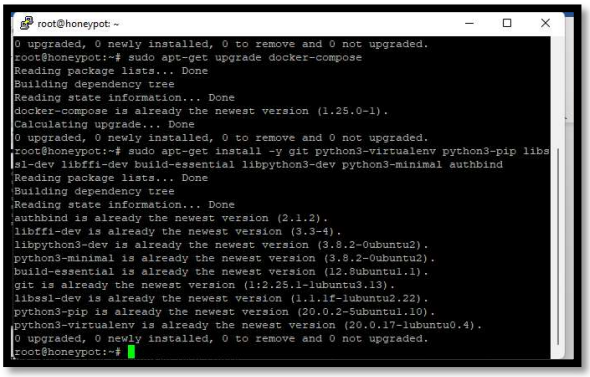
Gambar 20. Tampilan Berhasil Login

Setelah berhasil login selanjutnya lakukan update dan upgrade sistem.untuk melakukan update dan upgrade sistem dapat menggunakan perintah berikut pada terminal linux: sudo apt-get update && sudo apt-get -y upgrad update dan upgrade sistem dapat dilihat pada tampilan gambar dibawah ini:



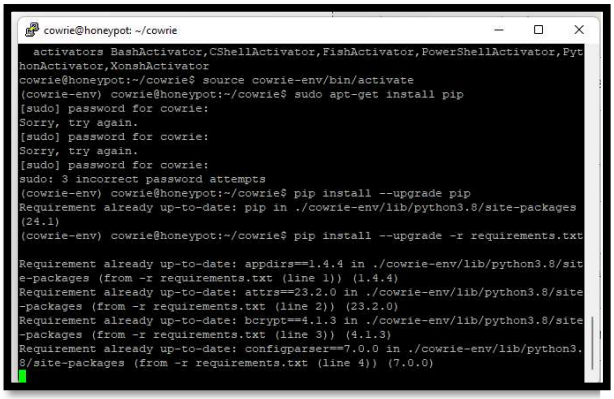
Gambar 21. Tampilan *Update* dan *Upgrade*

Selanjutnya *install* paket-paket yang dibutuhkan oleh cowrie honeypot, Adapun install paket-paket yang dibutuhkan dapat dilakukan dengan mengetik perintah berikut pada terminal: sudo apt-get install -y git python3-virtualenv python3-pip libssl-dev libffi-dev build-essential libpython3-dev python3-minimal authbind



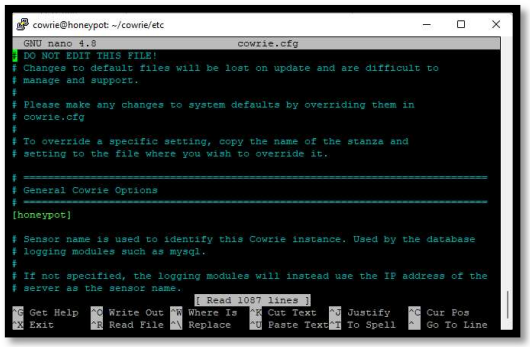
Gambar 22. Tampilan *Install* Paket Pendukung *Honeypot*

Selanjutnya aktifkan folder cowrie, yaitu dengan cara mengetik perintah berikut pada terminal:
source cowrie-env/bin/activate
python -m pip install --upgrade pip
python -m pip install --upgrade -r requirements.txt



Gambar 23. Tampilan *Install Requiretmen Honeypot*

Selanjutnya edit file cowrie.cfg, dengan mengetik perintah berikut pada terminal
cd /etc
cp cowrie.cfg.dist cowrie.cfg
nano cowrie.cfg



Gambar 24. Tampilan *File cowrie.cfg*

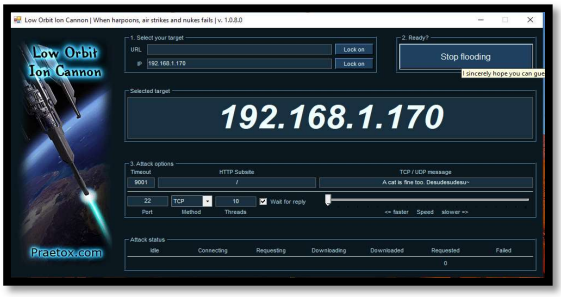
Pada *file cowrie.cfg* yang perlu di edit adalah pada bagian enable telnet, edit dari false menjadi true. Selanjutnya konfigurasi chown sistem, dengan cara mengetik perintah berikut pada terminal:
sudo touch /etc/authbind/byport/22
sudo chown cowrie /etc/authbind/byport/22
sudo chmod 770 /etc/authbind/byport/22
selanjutnya jalankan cowrie honeypot, dengan mengetik perintah berukut pada terminal:
bin/cowrie start
atau
authbind --deep /home/cowrie/cowrie/bin/cowrie start

4.3 Hasil

Dari serangkaian pengujian dimulai dari installasi sampai dengan tahap implementasi honeypot sebagai sistem keamanan server berbasis linux Pada SMK N 3 Kota Bengkulu berjalan dengan baik sesuai dengan rancangan dan kegunaan pada SMK N 3 Kota Bengkulu.

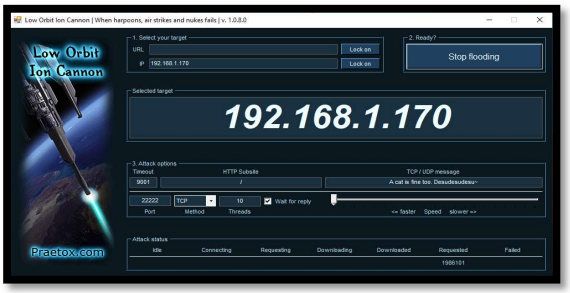
a. Serangan Ping Of Death Dalam Jaringan

Untuk melakukan serangan Ping of Death, disini menggunakan aplikasi LOIC, adapun hasil serangan ping of death ke server honeypot tidak dapat dilakukan, seperti dapat dilihat pada tampilan gambar dibawah ini:



Gambar 25. Tampilan Aplikasi LOIC Gagal Melakukan *Ping of Death*

Dari tampilan diatas dapat dilihat kegiatan Ping of Death menggunakan aplikasi LOIC tidak dapat berjalan ditandai dengan request = 0. Akan tetapi jika sudah di ketahui port yang digunakan maka akan bisa melakukan Ping of Death, seperti gambar dibawah ini:

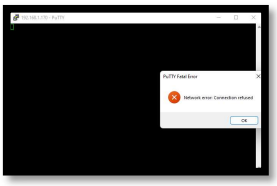


Gambar 26. Tampilan Aplikasi LOIC Berhasil Melakukan Ping of Death

Setelah di ketahui port yang digunakan maka ping of death dapat dilakukan, yang ditandai dengan request berjalan sebanyak 1988101, yang arti dapat melakukan pengiriman paket sebesar 1988101.

b. Mengakses Port 22 SSH Ke Server Pada Jaringan

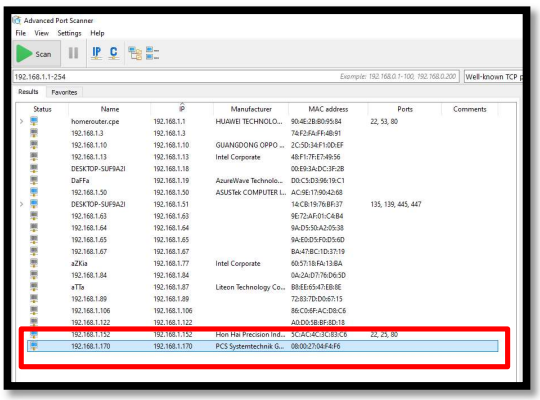
Server yang sudah di install dan konfigurasi cowrie honeypot, berhasil melakukan manipulasi port SSH. Sehingga port 22 tidak lagi dapat diakses untuk masuk ke sistem utama server, seperti dapat dilihat pada tampilan gambar dibawah ini:



Gambar 27. Tampilan Gagal Akses SSH (Port 22)

c. Port Scaning Terhadap Data Server

Server yang sudah di install dan konfigurasi cowrie honeypot, berhasil menghindari atau memperlama proses port scanner, port scanner dilakukan dengan menggunakan aplikasi advanced port scanner, seperti dapat dilihat pada tampilan gambar dibawah ini:

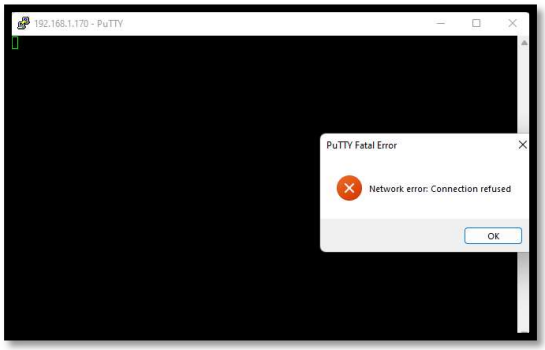


Gambar 28. Tampilan Gagal *Port Scanner*

Dari hasil port scanner diatas server yang sudah di *install cowrie honeypot* tidak menampilkan port yang terbuka. Sehingga akan mempersulit untuk melakukan hack terhadap server.

d. Mengakses Port 23 Telnet Ke Server Pada Jaringan

Server yang sudah di install dan konfigurasi cowrie honeypot, berhasil melakukan manipulasi port Telnet. Sehingga port 23 tidak lagi dapat diakses untuk masuk ke sistem utama server, seperti dapat dilihat pada tampilan gambar dibawah ini:



Gambar 29. Tampilan Gagal Akses Telnet (Port 23)

Dari serangkaian pengujian yang dilakukan di dapat hasil seperti pada table dibawah ini:

TABEL 1. HASIL PENGUJIAN

No	Pengujian	Hasil Pengujin	Keterangan
1.	Serangan *ping of death* dalam jaringan.	Dengan menggunakan cowrie honeypot, ping of death tidak dapat berjalan pada port yang normal digunakan untuk melakukan ping of death yaitu port 22 dan 23 dengan hasil request = 0	Baik
2.	Mengakses *port* 22 SSH ke *server* pada jaringan.	Port 22 (SSH) tidak dapat di akses setelah menerapkan cowrie honeypot. Untuk melakukan akses ke SSH masih dapat dilakukan dengan port unik yang telah di tentukan yaitu 22222	Baik
3.	Serangan *port scaning* terhadap data *server.*	Server honeypot dapat memperlambat atau bahkan menutup proses scanner port terhadap server, sehingga akan memperlama proses scanner port atau bahkan dapat menutup akses port scanner	Baik
4.	Mengakses Telnet ke server pada jaringan.	Port 23 (Telnet) tidak dapat di akses setelah menerapkan cowrie honeypot. Untuk melakukan akses ke Telnet masih dapat dilakukan dengan port unik yang telah di tentukan yaitu 22222	Baik

5. KESIMPULAN

Kesimpulan yang dapat diambil setelah implementasi honeypot sebagai sistem keamanan server berbasis linux Pada SMK N 3 Kota Bengkulu adalah *cowrie honeypot* dapat memanipulasi port-port utama untuk dapat mengakses sistem utama server yaitu port 22 dan 23. Dimana port tersebut dapat dimanipulasi dengan angka port yang unik. SSH dan Telnet tidak dapat diakses dengan menggunakan port yang umum untuk SSH dan Telnet yaitu port 22 dan 23, akan tetapi untuk masuk ke sistem utama server masih dapat dilakukan melalui port unik yang telah ditentukan yaitu 22222. Port scanner tidak dapat menampilkan port server yang terbuka.

Saran dari penulis apabila ada pembaca yang mau mengembangkan penelitian ini adalah Untuk penelitian selajutnya dapat dilakukan pengembangan sistem keamanan menggunakan cowrie honeypot sebaiknya dipadukan dengan sistem firewall untuk lebih meningkatkan keamanan. Sistem keamanan Jaringan komputer dapat dikembangkan dengan penerapan report secara real time

4. REFERENSI

[1] E. Mustofa, Masrusi, Muh ; Aribowo, "Penerapan Sistem Keamanan Honeypot Dan Ids Pada Jaringan Nirkabel (Hotspot)," *Sist. Keamanan Honeypot dan IDS*, vol. 1, no. 1, pp. 111–118, 2013.

[2] I. A. Romadhan, S. Syaifudin, and D. R. Akbi, "Implementasi Multiple Honeypot pada Raspberry Pi dan Visualisasi Log Honeypot Menggunakan ELK Stack," *J. Repos.*, vol. 2, no. 4, pp. 475–484, 2020, doi: 10.22219/repositor.v2i4.114.

[3] 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*, 2017, pp. 196–201, [Online]. Available: https://knsi.stikom-bali.ac.id/index.php/eproceedings/article/view/37.

[4] T. A. Cahyanto, H. Oktavianto, and A. W. Royan, "Analisis Dan Implementasi Honeypot Menggunakan Donaea Sebagai Penunjang Keamanan Jaringan," *J. Sist. Teknol. Inf. Indones.*, vol. 1, no. 2, pp. 86–92, 2016.

[5] N. Arkaan and D. V. S. Y. Sakti, "Implementasi Low Interaction Honeypot Untuk Analisa Serangan Pada Protokol SSH," *J. Nas. Teknol. dan Sist. Inf.*, vol. 5, no. 2, pp. 112–120, 2019, doi: 10.25077/teknosi.v5i2.2019.112-120.

[6] R. Dermawati and M. H. Siregar, "Implementasi Honeypot Pada Jaringan Internet Labor," *J. Ilm. Edutic*, vol. 7, no. 1, pp. 20–30, 2020.

[7] S. H. Wibowo *et al.*, *Cyber Crime di Era Digital*. 2023.

[8] A. F. Nurrahman, "Low-Interaction Honeypot Dengan Dionaea Untuk Mendukung Keamanan Jaringan," *J. Informatics Technol.*, vol. 2, no. 4, pp. 28–37, 2019.

[9] R. Purwoko, D. P. Febriyan, G. P. Adhe, W. S. Y. Laksito, S. Siswanti, and M. Hasbi, "JEPIN (Jurnal Edukasi dan Penelitian Informatika) Honeypot-as-a-Service dengan Kubernetes Cluster," *Jepin*, vol. 9, no. 2, pp. 204–2017, 2023.

[10] Arkaan, N. & Sakti, D. V. S. Y., "Implementasi Low Interaction Honeypot Untuk Peningkatan Keamanan Server dan Analisa Serangan Pada Protokol SSH," *J. Nas. Teknol. dan Sist. Inf.*, vol. 5, no. 2, 2019.