# IMPLEMENTASI MODERN HONEY NETWORK (MHN) UNTUK KEAMANAN JARINGAN (STUDI KASUS: DINAS PERPUSTAKAAN DAN KEARSIPAN PROVINSI RIAU)

**Nadilla Asyanin Sembiring Depari[1], Ery Safrianti[2], Linna Oktaviana Sari[3]**
Mahasiswa Program Studi Teknik Informatika[1], Dosen Teknik Elektro Program Studi Teknik Informatika[2], Dosen Teknik Elektro Program Studi Teknik Informatika [3]
Teknik Informatika, Jurusan Teknik Elektro, Fakultas Teknik Universitas Riau
Kampus Binawidya Km 12,5 Simpang Baru Panam, Pekanbaru, Riau
Jurusan Teknik Elektro Universitas Riau
E-Mail : nadilla.asyanin@student.unri.ac.id

***ABSTRACT***

*Network security is an important aspect that must be maintained. Attacks on network security systems often occur at the Riau Provincial Library and Archives Service. One of the technologies in improving network security is Honeypot. Honeypot is fake system designed to trick attackers. There are many kind of honeypot, one of them is Modern Honey Network (MHN). This research is useful for implementing the MHN on the existing network in the Library and Archives Service of Riau Province. Two Operation System were built, namely a Web Server and an MHN Server. The Web Server is built using Moodle to trap hackers to attack into the system. Honeypot sensors in MHN namely Cowrie, Glastopf and Dionaea was installed into the Web Server. Tests carried out by attacking port 22, portscanning and DDOS using LOIC Tool to find out whether the attack can be detected by the honeypot sensor. Test result show that the attack was successfully detected by MHN.*

**Keyword:** *Honeypot, Modern Honey Network (MHN), Cowrie, Dionaea, Glastopf*

## 1 PENDAHULUAN

Seiring dengan pesatnya perkembangan teknologi jaringan, semakin besar pula ancaman dan gangguan terhadap kinerja dalam teknologi tersebut. Salah satu teknik dalam mempertahankan keamanan jaringan adalah menggunakan *firewall*. *Firewall* adalah sistem atau perangkat yang dianggap aman untuk melaluinya dan mencegah lalu lintas jaringan yang tidak aman (Prasetyo, 2008). Jenis Keamanan ini banyak diimplementasikan di banyak instansi dan perusahaan. Salah satunya adalah Dinas Perpustakaan dan Kerasipan Provinsi Riau.

Dinas Perpustakaan dan Kearsipan Provinsi Riau adalah instansi yang bergerak dibidang perpustakaan, arsip, dan dokumentasi. Dinas Perpustakaan dan Kearsipan Provinsi Riau merupakan salah satu instansi yang menjadi target serangan. Terbukti dengan banyaknya serangan yang masuk. Serangan terjadi setiap hari meskipun bukan serangan yang berat dan fatal, namun tidak menjamin data tetap aman karna serangan dapat terus berkembang. *Honeypot* merupakan salah satu jenis keamanan jaringan yang mampu mengidentifikasi penyerangan sehingga cocok diusulkan untuk diimplementasikan pada Dinas Perpustakaan dan Kearsipan Provinsi Riau. *Honeypot* adalah sumber daya keamanan yang mempunyai nilai jika sistem disusupi atau diserang.

*Honeypot* merupakan sistem yang dirancang untuk diperiksa dan diserang

(Andros dan Lucas, 2014). *Honeypot* memiliki berbagai jenis contohnya MHN (*Modern Honey Network*). MHN adalah *software opensource* yang di buat bertujuan untuk mempermudah instalasi *honeypot*. Adapun kegunaan dari MHN ini adalah mengelola dan menganalisa data serangan dan mempermudah membangun *honeypot* baru. Ada beberapa *honeypot* yang sudah terintergrasi oleh MHN antara lain *hpfeed*, *nmomesyne*, *honeymap*, *MongoDB*, *dionaea*, *conpot*, *snort*, *kippo*, *glastopf*, *amun*, dan *wordpot* (Laksana dkk, 2017).

Berdasarkan hasil wawancara dan observasi ditempat penelitian, didapatkan masalah masalah yang sering terjadi adalah *Port Scanning*, DDOS, serangan terhadap SSH. Atas dasar pertimbangan hasil wawancara dan melihat serangan yang terjadi, maka penelitian tentang perancangan dan implementasi MHN dilakukan. Jenis *honeypot* yang diimplementasikan adalah *Coewie*, *Dionaea*, dan *Glastopf*. Penelitian ini bertujuan sebagai alat bantu keamanan jaringan guna mengurangi terjadinya serangan-serangan dan pencarian celah kelemahan lainnya di Dinas Perpustakaan dan Kearsipan Provinsi Riau. Untuk melakukan perancangan dan implementasi *honeypot* di Dinas Perpustakaan dan Kearsipan Provinsi Riau, maka dibuatlah skripsi ini dengan judul "*Implementasi Modern Honey Network untuk keamanan jaringan data di Dinas Perpustakaan dan Kearsipan Provinsi Riau*"

## 2 TINJAUAN PUSTAKA

Tinjauan Pustaka berisi teori-teori yang mendukung penelitian meliputi Honeypot, Modern Honey Network, Cowrie Honeypot, Glastopf Honeypot, Dionaea Honeypot, Moodle, Port Scanning, DDoS, Nmap dan LOIC.

### 2.1 Honeypot

*Honeypot* adalah sumber sistem informasi yang biasanya didesain bertujuan untuk mendeteksi, menjebak, dalam usaha percobaan penetrasi kedalam sistem. Suatu sistem yang terdiri dari beberapa *honeypot* disebut juga *honeynet*. Apabila *attacker* melakukan penyerangan kedalam sistem atau *server*, maka *honeypot* yang menyerupai *server* asli akan mengalami penyerangan terlebih dahulu, sedangkan sistem dan *server* asli tetap aman dibelakang *honeypot*. *Honeypot* dalam pengimplementasianya pada jaringan, dapat ditempatkan kedalam tiga kategori yakni:

1. Internal

   Penempatan *honeypot* didalam jaringan internal adalah cara yang baik untuk membuat system *early-werning* yang akan memberi informasi akan ancaman yang datang dari luar dan dalam jaringan local (Prasetyo, 2007).
2. Eksternal

   Lokasi ini merupakan penempatan yang paling tepat jika ingin mendeteksi penyerangan dari luar, karena *honeypot* secara langsung terkoneksi pada internet sehingga mudah ditemukan dan diserang.
3. DMZ (*Demilitarized Zone*)

   Penempatan *honeypot* dilokasi ini merupakan solusi terbaik untuk mengimplementasikan *honeypot*. Hal ini disebabkan karena DMZ terletak diantara jaringan internal dan jaringan eksternal. Pada *gateway* biasanya terdapat pengamanan sama seperti *firewall* sehingga trafik tidak sah yang menuju *honeypot* akan melewati *firewall* akan tercatat di *firewall log* dan menambah informasi yang terkumpul(Prasetyo, 2007).

### 2.2 Modern Honey Network

MHN adalah *software opensource* yang dibuat oleh perusahaan ThreatStream, yang bertujuan untuk mempermudah menginstalasi *honeypot*. Adapun kegunaan dari MHN ini adalah mengelola dan menganalisa data dari *honeypot* tersebut dan mempermudah membangun *honeypot* baru dan mengambil data. Ada beberapa *honeypot* yang sudah terintergrasi oleh Modern Honey Network (MHN) antara lain hpfeed, nmomesyne, honeymap, MongoDB, dionaea, conpot, snort, kippo,

glastopf, amun, dan wordpot (Laksana dkk, 2017).

### 2.3 Cowrie Honeypot

*Cowrie* merupakan salah satu *tools honeypot* SSH yang berfungsi sebagai *server* palsu yang termasuk dalam kategori *medium interaction* dimana layanan interaksi yang dapat berinteraksi dengan cara memberi tanggapan terhadap serangan yang dilakukan penyerang saat mencoba memberi *worm* ke dalam sistem komputer. *Cowrie* adalah *honeypot* SSH yang dapat digunakan untuk menganalisa metode pencegahan spesifik untuk lebih memahami motif penyerang dalam sistem (McCaughey, 2017).

### 2.4 Glastopf Honeypot

*Glastopf* adalah *low involvement* web *honeypot* yang mampu meniru ribuan *vulnerability web* untuk mengumpulkan data dari serangan yang menargetkan aplikasi *web*. Prinsip *glastopf* sangat sederhana yaitu membalas serangan itu dengan menggunakan respon penyerang mengharapkan usahanya untuk mengeksploitasi aplikasi *web*. Cara kerja *Glastopf* sama seperti cara kerja *web server*. Seseorang mengirim *request* ke *web server*, lalu *request* tersebut diproses, mungkin ada beberapa yang disimpan di database dan kemudian *server* membalas request tersebut (Wafi,2016)

### 2.5 Dionaea Honeypot

*Dionaea* adalah *honeypot* yang bersifat *low interaction honeypot* yang diciptakan sebagai pengganti Nepenthes (Cahyanto dkk 2016). *Dionaea* adalah perangkat lunak yang menawarkan layanan jaringan yang dapat dieksploitasi. Dalam tindakan yang dilakukannya adalah untuk menjebak atau mengeksploitasi *malware* yang menyerang jaringan, tujuan utamanya adalah mendapatkan salinan dari *malware* tersebut. (Arief, 2012). *Dionaea* adalah jenis *honeypot* yang dapat memberikan layanan jaringan yang nantinya dapat dieksploitasi. Tujuan dari dionaea adalah untuk mendapatkan salinan dari *malware* yang telah dikirim oleh penyerang, sehingga seorang administrator dapat memutuskan sesuatu untuk melindungi sistem induk, bahkan menciptakan antivirus baru.

### 2.6 Moodle

*Moodle* merupakan singkatan dari (*Modular Object-Oriented Dynamic Learning Environment*) adalah sebuah paket perangkat lunak yang diproduksi untuk berbasis internet dan *website* yang menggunakan prinsip *social constructionist pedagogy*(Wafi, 2016).

### 2.7 Port Scanning

Port Scanning adalah aktivitas yang digunakan untuk scan *port* TCP dan UDP dan melaporkan status mereka. *Port scanner* menggunakan beberapa *protocol* seperti TCP, UDP dan ICMP. *Port scanning* memungkinkan individu memeriksa dan menentukan layanan apa yang berjalan pada komputer target. *Port* digunakan oleh kedua *protocol* yaitu TCP dan UDP. (Wafi, 2016).

### 2.8 DDoS

*Distributed Denial of Service* (DDoS) merupakan salah satu jenis serangan *Denial of Service* yang menggunakan banyak *host* penyerang sekaligus untuk menyerang satu buah *host* target dalam sebuah jaringan. Hal ini akan membutuhkan waktu yang lama supaya bisa membanjiri *host* target. Banyaknya komputer yang menyerang sebuah sistem merupakan kelebihan yang menyebabkan betapa berbahayanya DDoS (Zam, 2011).

### 2.9 Nmap

*Nmap* (*Network Mapper*) adalah sebuah program *opensource* yang berguna untuk mengesksplorasi jaringan. Nmap didesain untuk dapat melakukan scan jaringan yang besar, juga dapat digunakan untuk melakukan *scan host* tunggal. Nmap menggunakan paket IP untuk menentukan *host-host* yang aktif dalam suatu jaringan

*port-port* yang terbuka, sistem operasi yang dipunyai, tipe *firewall* yang dipakai (Rosnelly dan Pulungan, 2011).

### 2.10 LOIC

LOIC adalah *software* DoS yang compatible di *windows*, LOIC dapat digunakan oleh satu komputer attack dan akan lebih baik digunakan dalam jumlah beberapa komputer, sehingga downtime akan lebih banyak (Ar, 2012).

## 3 METODE PENELITIAN

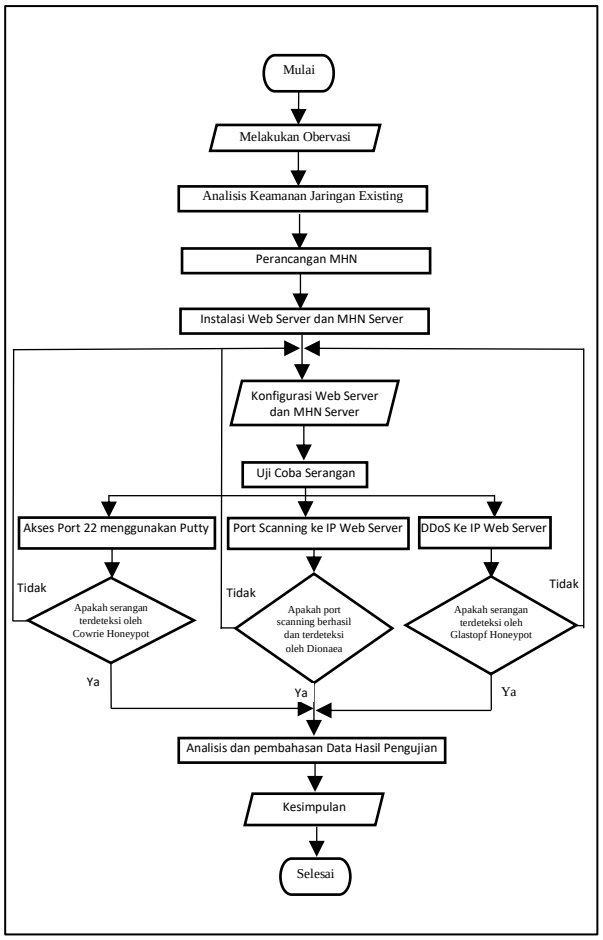
Metode Penelitian berisi tentang gambaran umum dari penelitian, alur penelitian, skema jaringan *existing*, instalasi dan konfigurasi Web Server, instalasi dan konfigurasi MHN Server, serta instalasi dan Konfigurasi Sensor Honeypot.

### 3.1 Gambaran Umum

Berdasarkan dengan penghantar pendahuluan, batasan penelitian, dan tinjauan pustaka yang dijelaskan pada bab-bab sebelumnya, penelitian yang akan dilakukan adalah merancang dan mengimplementasikan *Modern Honey Network*(*cowie, dioneae,* dan *glatsopf*) untuk alat bantu keamanan jaringan di Dinas Perpustakaan dan Kearsipan Provinsi Riau.

### 3.2 Alur Penelitian

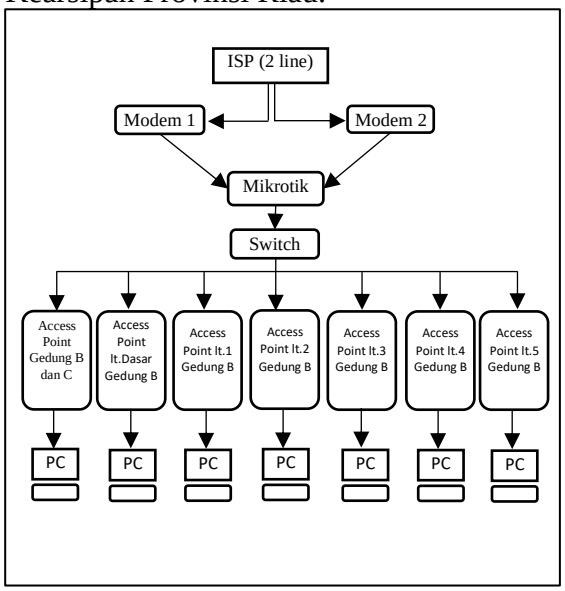
Diagram alir berguna untuk memudahkan untuk proses penelitian dari tahap awal hingga selesai dan mudah untuk menganalisa. Langkah kerja pada diagram alir dapat dilihat pada Gambar 1.




**Gambar1.** Diagram Alir Penelitian

### 3.3 Skema Jaringan Existing

Gambar 2 merupakan skema jaringan *existing* di Dinas Perpustakaan dan Kearsipan Provinsi Riau.




**Gambar2.** Skema Jaringan *Existing*

Tabel 1 menunjukan skema IP Address yang ada sesuai dengan jarigan *existing* pada Dinas Perpustakaan Dan Kearsipan Provinsi Riau.

**Tabel1.** Skema IP *Address*

| **Perangkat** | ***Interface*** | ***IP Address*** | ***Gateway*** |
|---|---|---|---|
| *Mikrotik* | ISP 1 | 192.168.100.2/24 | 192.168.100.1/24 |
| | ISP 2 | 192.168.200.2/24 | 192.168.200.1/24 |
| | LAN 3 | 192.168.206.1/24 | 192.168.206.1/24 |
| | LAN 4 | 192.168.207.1/24 | 192.168.207.1/24 |

### 3.4 Instalasi dan Konfigurasi Web Server

Pada penelitian kali ini, *web server* yang akan dibuat adalah *web server* yang menggunakan CMS (Content Management System) yakni moodle 3.6. Untuk melakukan instalasi CMS moodle 3.6, terlebih dahulu harus memiliki Apache, MyQSL/MariaDB, dan PHP. Setelah semua komponen terinstall, maka selanjutnya adalah membuat *database* Moodle. Langkah instalasi dan konfigurasi Web Server sebagai berikut:

1. Lakukan installasi ubuntu 18.04 LTS. Setelah itu akses ubuntu yang telah diinstal sebelumnya seperti Gambar 3.

**Gambar 3.** Ubuntu *Web Server*

2. Install java.

```
# sudo su
# sudo apt-get install
openjdk-11-jdk-headless
# sudo apt-get update
```

3. Instalasi Apache dan PHP.

```
# sudo apt-get install
software-properties-common
# sudo add-apt-repository
ppa:ondrej/php
# sudo apt-get update
# sudo apt upgrade -y
# sudo apt install apache2 -
y
# sudo systemctl status
apache2
```

4. instalasi php dan install modul-modul yang dibutuhkan moodle.

```
# sudo apt install php php-
common php-mbstring php-
xmlrpc php-soap php-gd php-
xml php-intl php-mysql php-
libapache2-mod-php-y php-
idap php-zip php-curl
```

5. Edit konfigurasi php.ini agar kinerjanya lebih powerfull.

```
# sudo
/etc/php/apache2/php.ini
# file_uploads = on
# allow_url_fopen = on
# memory_limit = 512M
# upload_max_file_size = 64M
# max_execution_time = 360
# cgif.fix_pathinfo = 0
```

6. Instalasi Database MariaDB. MariaDB merupakan basisdata yang diimplementasikan sebagai basisdata moodle.

```
# sudo apt-get install
mariadb-server -y
```

7. Konfigurasi keamanan pada MariaDB.

```
# sudo
mysql_secure_installation

# enter current password for
root (enter for none):
# set root pasword? (y/n): Y
# new password:
# reenter new pasword:
# remove anonymus user? Y
# disallow root login
remotely (Y/N): Y
# remote test database and
access to it? (y/n): Y
```

```
# reload previlage tables
now?(y/n):y
```

8. konfigurasi MariaDB.

```
#sudo nano
/etc/mysql/mariadb.conf.d/50
-server.cnf
# default_storage_engine =
innodb
# innodb_file_per_table = 1
# innodb_file_format =
barracuda
# innodb_large_prefix = 1
```

9. Lakukan restart untuk menyimpan konfigurasi dengan perintah:

```
# sudo systemct1 restart
mariadb
```

10. Masuk kedalam aplikasi mysql.

```
# sudo mysql -u root -p
# CREATE DATABASE moodle
DEFAULT CHARACTER SET
utf8mb4;
# COLLATE
utf8mb4_unicode_ci;
# CREATE USER
'nadillaasyani'@'localhost'
IDENTIFIED BY
'pangeran';
# GRANT ALL PRIVILEGES ON
moodle.*TO
'usrmoodle'@'localhost';
# FLUSH PRIVILEGES;
# EXIT
```

11. Download moodle dengan perintah:

```
# sudo wget
https://download.moodle.org/
download.php/direct/stable38
/moodle-latest-38.tgz
# tar xzvf moodle-latest-
38.tgz
```

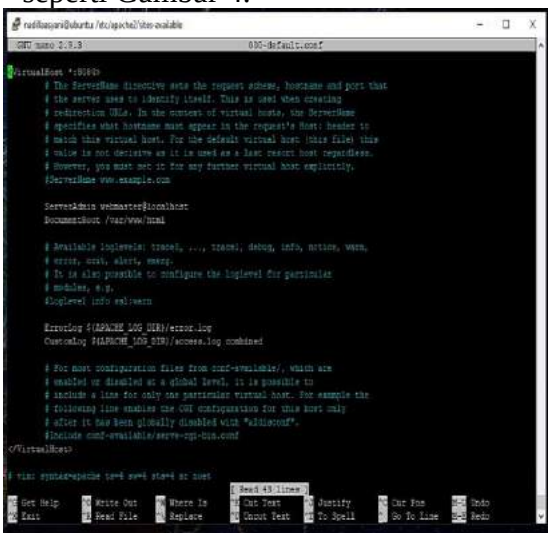
12. Membuat folder moodle. Folder utama moodle harus pindah ke direktori */var/www/*. Lalu mengubah hak akses folder utama moodle tersebut, agar moodle tersebut dapat dikonfigurasi dengan perintah:

```
# sudo mkdir -p
/var/www/moodle/data
# sudo mv moodle
/var/www/moodle/web
# sudo chown -R www-
data:www-data
/var/www/moodle
# sudo chmod -R 775
/var/www/moodle
```

13. Konfigurasi virtual host.

```
# cd /etc/apache/sites-
available/
# sudo nano 000-default.conf
```

14. Masukan konfigurasi virtual host seperti Gambar 4.

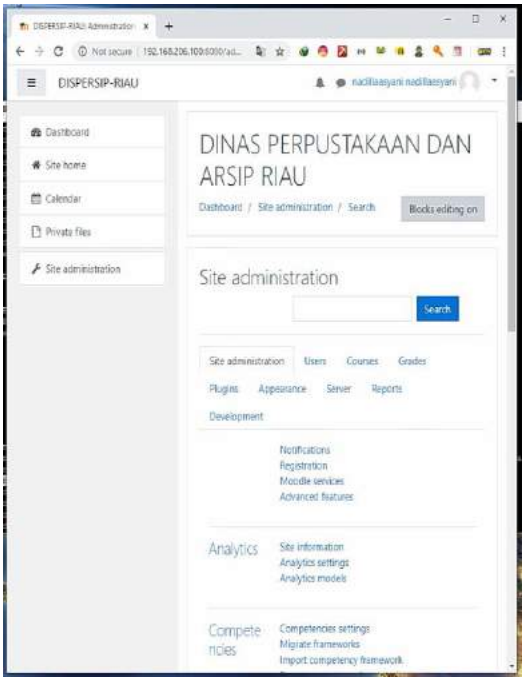


**Gambar 4.** Konfigurasi *Virtual Host*

15. Aktifkan virtual host dan *restart* Apache

```
#sudo systemctl restart
apache2
```

16. Akses web server melalui web browser dengan alamat IP http://192.168.206.100:8080.
Tampilan awal *web server* seperti Gambar 5.




**Gambar 5.** Tampilan Awal Moodle

### 3.5 Instalasi dan Konfigurasi MHN Server

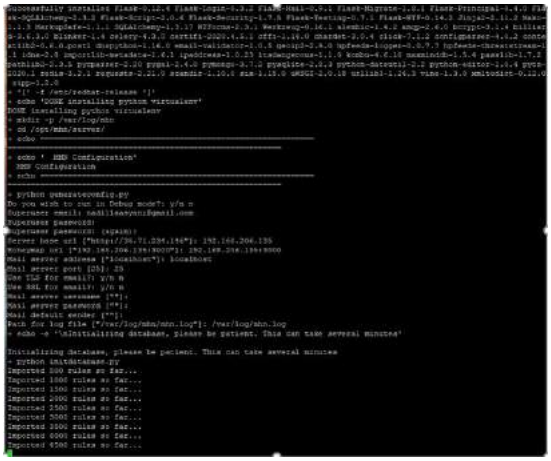
Pada Ubuntu 18.04 yang lain, *login* dengan *username* dan *password* yang telah dibuat sebelumnya. Masuk ke *root* untuk install git yang berguna dalam *clone repository* MHN dengan melakukan perintah:

```
# sudo apt-get update && sudo apt-get upgrade
# sudo install git -y
# cd /opt
# sudo git clone https://github.com/hansahdarmawan/mhn.git
```

1. Perintah installasi MHN dengan perintah:

```
# cd /mhn
# sudo ./install.sh
```

2. konfigurasi MHN Server seperti Gambar 6.



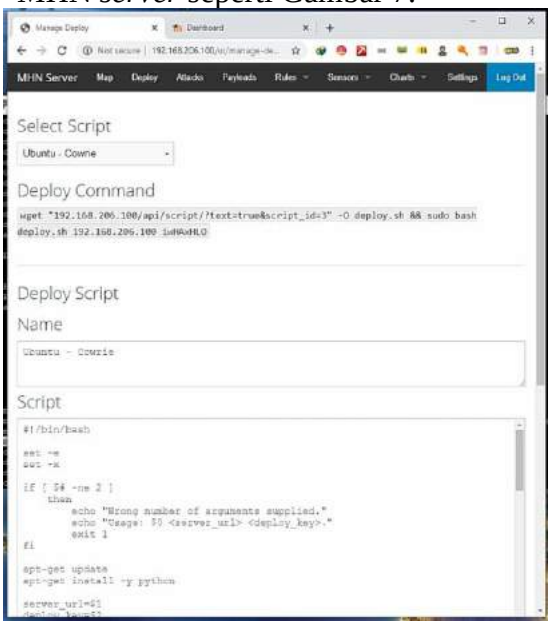
**Gambar 6.** Konfigurasi MHN *Server*

### 3.6 Instalasi dan Konfigurasi Sensor Honeypot

Implementasi sensor *honeypot* langsung dilakukan pada *web server* dengan mengakses dan merujuk pada MHN *server web* melalui *browser*. Dalam hal ini *honeypot* yang akan diimplementasi adalah cowrie, glastopf, dan dionaea.

1. *Cowrie Honeypot*

   Untuk instalasi *Cowrie Honeypot*, buka *script Cowrie* pada menu *deploy* MHN *server* seperti Gambar 7.



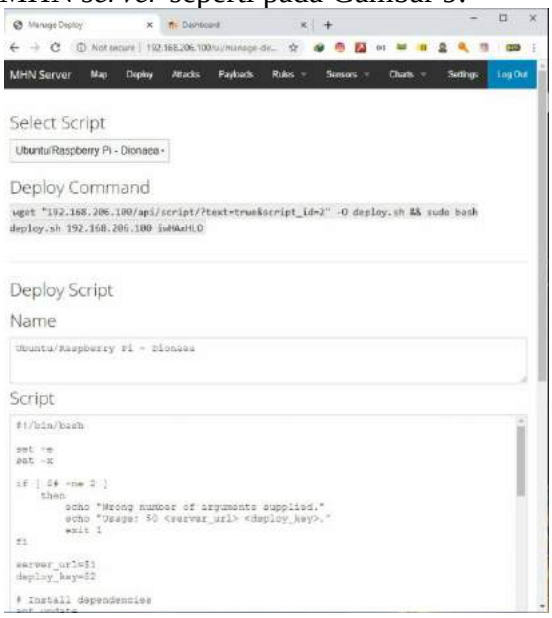

**Gambar 7.** *Script Cowrie* pada MHN *Server*

Salin script diatas dan tempel pada *root web server* untuk instalasi *Cowrie Honeypot* seperti Gambar 8.

**Gambar 8.** Instalasi *Cowrie* pada *Web Server*

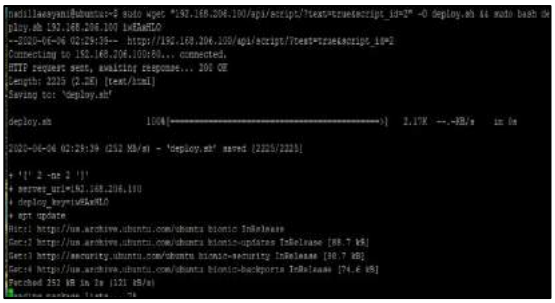
2. *Dionaea Honeypot*

Untuk instalasi *Dionaea Honeypot*, buka *script Dionaea* pada menu *deploy* MHN *server* seperti pada Gambar 9.




**Gambar 9.** *Script Dionaea* pada MHN *Server*

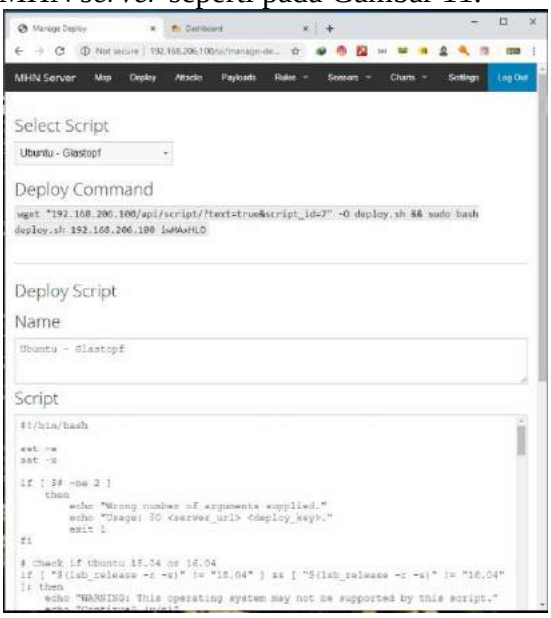
Salin *script* diatas dan tempel pada *root web server* seperti pada Gambar 10.



**Gambar 10.** Instalasi *Dionaea* pada *Web Server*

3. *Glastopf Honeypot*

Untuk instalasi Glastopf *honeypot*, buka *script Glastopf* pada menu *deploy* MHN *server* seperti pada Gambar 11.




**Gambar 11.** *Script Glastopf* pada MHN *Server*

Salin *script* diatas dan tempel pada *root web server* seperti Gambar 12.

**Gambar 12.** Instalasi *Glastopf* pada Web Server

## 4 HASIL DAN PEMBAHASAN

Pada hasil dan pembahasan berisi implementasi penelitian yakni implementasi MHN dan uji coba serangan terhadap hasil implementasi MHN.

### 4.1 Implementasi MHN

Implementasi MHN berisi implementasi Web Server, MHN Server, Cowrie Honeypot, Dionaea Honeypot, dan Glastopf Honeypot.

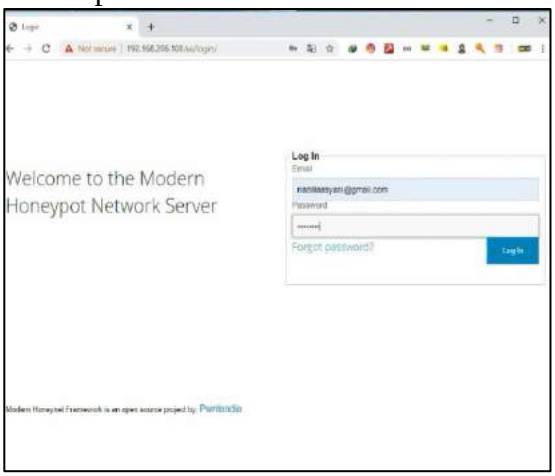
1. Implementasi Web Server

Penggantian tema moodle dengan memilih turn editing on pada kolom administration. Peneliti mengklik edit settings, appearance,theme lalu memilih theme selector dan memilih tema yang ada, yang mana sebelumnya sudah di-*upload* ke server tersebut. Sehingga web interface moodle menjadi seperti Gambar 13.


**Gambar 13.** Desain *Web Interface* Moodle

2. Implementasi MHN Server

Untuk mengakses MHN server pada *web browser* dilakukan dengan mengetik http://192.168.206.100/ maka tampilannya akan seperti Gambar 14.




**Gambar 14.** Tampilan MHN *Server* di *Web Browser*

3. Implementasi Cowrie Honeypot

Tampilan status cowrie honeypot akan seperti Gambar 15.


**Gambar 15.** Status *Cowrie* Telah *Running*

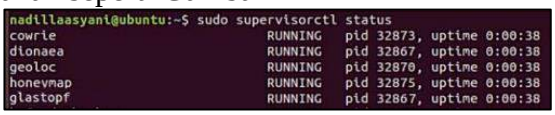
4. Implementasi Dionaea Honeypot

Tampilannya status *Dionaea Honeypot* akan seperti Gambar 16 ini:


**Gambar 16.** Status *Glastoft* Telah *Running*

5. Implementasi Glastopf Honeypot

Tampilan status *Glastopf Honeypot* akan seperti Gambar 17



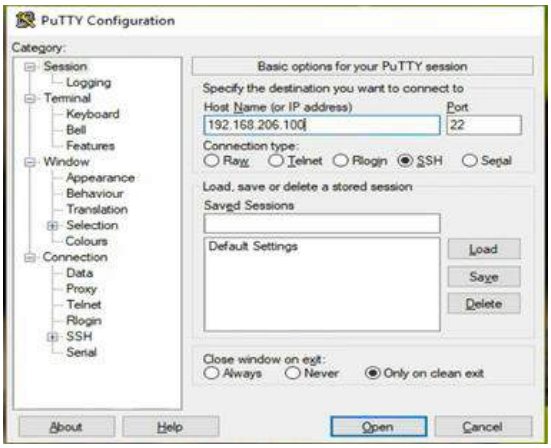

**Gambar 17.** Status *Glastoft* Telah *Running*

## 4.2 Uji Coba Serangan

Uji coba serangan dilakukan terhadap Cowrie Honeypot, Dionaea Honeypot, Glastopf Honeypot.

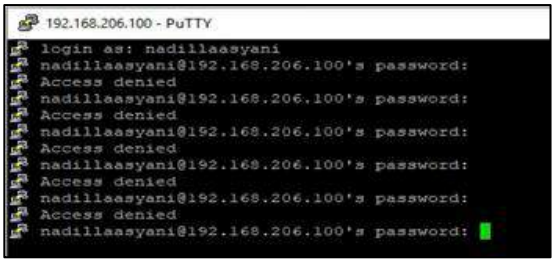
1. Uji Coba Cowrie Honeypot

Pengujian untuk sensor ini adalah dengan mencoba mengakses *web server* menggunakan putty dan masuk melalui *port* 22. Uji coba dapat terlihat seperti Gambar 18.




**Gambar 18.** Akses *Web Server* Menggunakan Putty

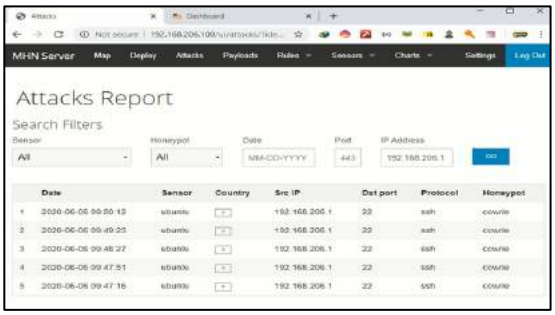
Tampilan pada putty akan seperti Gambar 19.




**Gambar 19.** Akses *Web Server* Menggunakan Putty Tidak Berhasil

Gambar 19 menunjukan bahwa uji coba akses *web server* melalui *port* 22 dilakukan dengan memasukan *password* yang tidak benar sebagai contoh tindakan akses tidak sah. Uji coba akses menggunakan *password* yang tidak benar dilakukan sebanyak 5 kali. Maka akses ke *port* 22 tidak bisa dilakukan.

Hasil Pengujian serangan yang dilakukan muncul di MHN *server*. Data serangan yang masuk berasal dari 5 kali percobaan akses *port* 22 dengan memasukan *password* yang tidak benar. Data serangan yang masuk dan dideteksi oleh *Cowrie Honeypot* adalah seperti Gambar 20.



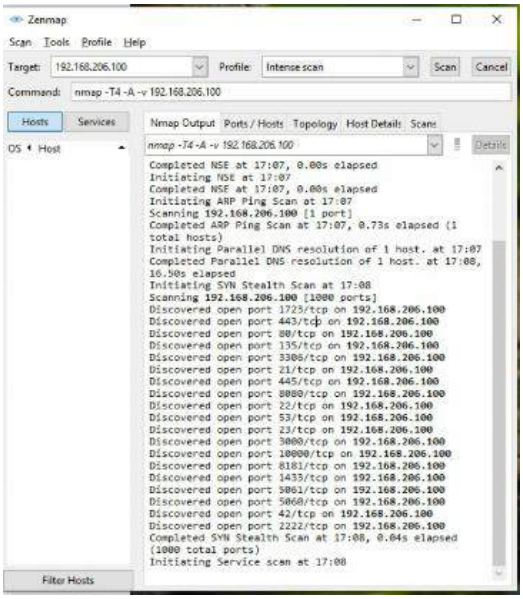

**Gambar 20.** Data Serangan pada *Cowrie Honeypot*

2. Uji Coba Dionaea Honeypot

Pengujian ini menggunakan skema pengujian *port scanning* menuju IP *web server*. Perintah yang dijalankan saat pengujian ini berlangsung adalah:

```
Nmap -T4 -A -v 192.168.206.100
```

Perintah ini menjelaskan bahwa *port scanning* yang dilakukan ke alamat IP 192.168.4.168 secara *intense scan* dan ditunjukan oleh Gambar 21.

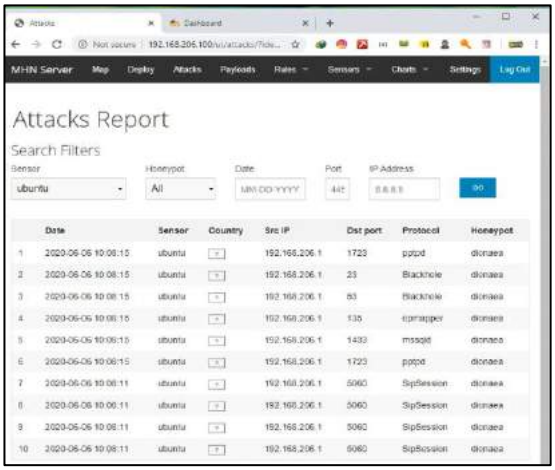



**Gambar 21.** *Port Scanning* IP *Web Server*

Hasil Pengujian serangan yang dilakukan muncul di MHN server. Data serangan yang masuk berasal dari 1 kali percobaan *port scanner* yang ditujukan ke IP *web server*.

Dari uji coba serangan *port scanner* yang dilakukan, diketahui bahwa pada *Web Server* terdapat *port-port* yang terbuka. Port yang terbuka adalah port 1723, 443, 88, 135, 3305, 21, 445, 8080, 22, 53, 23, 300, 10000, 5181, 1433, 5061, 5060, 42, 2222. Semua port ini merupakan port yang tersedia atau terbuka dan bisa menjadi jalan masuk serangan.

Data serangan port scanning pada Gambar 21 masuk dan dideteksi oleh *Dionaea Honeypot* adalah seperti Gambar 22.




**Gambar 22.** Data Serangan Pada *Dionaea Honeypot*

Hasil yang ditunjukan sama seperti *honeypot* sebelumnya. Dari Gambar 22 dapat dilihat serangan yang ditujukan ke *port* yang ditangani oleh *dionaea* akan masuk dan dideteksi oleh *Dionaea Honeypot*. Contoh serangan yang ditangai oleh *Dionaea* pada Gambar 22 adalah serangan yang mengarah ke *port* 1723, 23, 53, 135, 1433, 1723, 5060. Dengan demikian, dapat disimpulkan bahwa *Dionaea honeypot* dapat berjalan dengan baik dalam mendeteksi serangan yang masuk ke sistem.

3. Uji Coba Glastopf Honeypot

Pengujian serangan terhadap glastopf honeypot menggunakan skema serangan Distributed Denial of Service (DDOS). *Tool* yang digunakan adalah LOIC. *Tool* ini bersifat *open source* dan dapat digunakan pada sistem operasi *windows*. Pengujian ini ditunjukan oleh Gambar 23.


**Gambar 23.** Data Serangan Pada Dionaea *Honeypot*

Berdasarkan Gambar 23, dilakukan percobaan serangan terhadap web server melalui *port* 80. Pengujian ini menggunakan HTTP sebab port tersebut terbuka.

Hasil Pengujian serangan *port* 80 menggunakan LOIC masuk dan dideteksi oleh *Glastopf Honeypot* adalah seperti Gambar 24.


**Gambar 24.** Data Serangan Pada *Glastopf Honeypot*

*Web Interface* dari *glastopf honeypot* sama dengan hasil yang ditunjukan oleh *cowrie honeypot*, akan tetapi *glastopf* hanya membuka *port* HTTP sehingga penyerang yang berusahan melakukan aktivitas ke *port* HTTP pada *server* akan ditampilkan pada MHN *server*. Dengan demikian, dapat disimpulkan bahwa *Glastopf honeypot* dapat berjalan dengan baik dalam mendeteksi serangan yang masuk ke sistem.

## 5 KESIMPULAN

Berdasarkan pembahasan dan hasil yang didapatkan dari bab sebelumnya, didapatkan kesimpulan sebagai berikut:

1. Modern Honey Network dapat di implementasikan pada jaringan Existing di Dinas Perpustakaan dan Kearsipan Provinsi Riau berdasarkan hasil ujicoba sensor pada *MHN Server* yang berhasil mendeteksi serangan dan aktivitas pada *Web Server*.
2. Sensor Honeypot yakni Cowrie berhasil mendeteksi serangan port 22, Dionaea berhasil mendeteksi *port scanning* yang ditujukan ke webserver dan Glastopf berhasil mendeteksi serangan yang masuk ke port 80 dalam jaringan.

## DAFTAR PUSTAKA

Andros R, Lukas. 2014. *Implementasi Honeypot Dengan Raspberry Pi Sebagai Alat Bantu Pendeteksi Keamanan Jaringan Dan Penangkap Malware*. Jakarta: Jurnal Teknik dan

Ilmu Komputer Vol. 04 No. 13, Jan – Mar 2015. Hal. 14

Ar, A. A. (2012). *Evaluasi Penerapan Autentikasi Pengguna Wireless Lan Berbasis Radius Server Universitas Bina Darma*. Palembang: Universitas Bina Darma. Hal.6

Arief, M. 2012. *Implementasi Honeypot Dengan Menggunkan Dionaea Dijaringan Hotspot FIZZ*. Bandung: Politeknik Telkom. Hal.2

Cahyanto, T. A. dkk. 2016. *Analisis dan Implementasi Honeypot Menggunakan Dionaea Sebagai Penunjang Keamanan Jaringan*. Jember: JUSTINDO, Jurnal Sistem & Teknologi Informasi Indonesia, Vol. 1, No. 2, Agustus 2016. Hal. 87

Laksana, D.D. dkk. 2017. *Implementasi Honeypot Dengan Modern Honey Network*. Bandung: e-Proceeding of Applied Science: Vol.3, No.3 Desember 2017 | Page 1815. ISSN : 2442-5826

McCaughey, R. J. 2017. *Deception Using An SSH Honeypot*. California: Naval Postgraduate School. Standard Form 298 (Rev. 2–89) Prescribed by ANSI Std. 239–1. Hal. 23

Prasetyo, H. 2008. *Perancangan dan Implementasi Sistem Honeypot Sebagai Alat Bantu Keamanan Jaringan Komputer pada PT. IP Teknologi Komunikasi*. Jakarta: Universitas Islam Negeri Syarif Hidayatullah. Hal. 23

Posnelly, R., R. Pulungan. 2011. *Membandingkan Analisis Trafik Data Pada Jaringan Komputer antara Wireshark dan NMAP*. Yogyakarta: Konferensi Nasional Sistem Informasi 2011. Hal. 942

Wafi, H. 2016. *Implementasi Sistem Keamanan Honeypot dengan Modern Honey Network pada Jaringan Wireless*. Jakarta: Universitas Islam Negeri Syarif Hidayatullah. Hal.25-27

Zam, E. (2011). *Buku Sakti Hacker*. Jakarta: MediaKita.