

**IMPLEMENTASI HONEYPOT SEBAGAI PENDETEKSI MALWARE
PADA LAYANAN CLOUD COMPUTING**

Skripsi

Diajukan Sebagai Salah Satu Syarat
Untuk Memperoleh Gelar Sarjana Teknik Informatika



Disusun oleh :

Arya Ervan Leoresta

10650062

**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI SUNAN KALIJAGA
YOGYAKARTA**

2014

**IMPLEMENTASI HONEYPOT SEBAGAI PENDETEKSI MALWARE
PADA LAYANAN CLOUD COMPUTING**

Skripsi

Diajukan Sebagai Salah Satu Syarat
Untuk Memperoleh Gelar Sarjana Teknik Informatika



Disusun oleh :

Arya Ervan Leoresta

10650062

**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI SUNAN KALIJAGA
YOGYAKARTA**

2014



PENGESAHAN SKRIPSI/TUGAS AKHIR

Nomor : UIN.02/D.ST/PP.01.1/2874/2014

Skripsi/Tugas Akhir dengan judul : Implementasi Honeypot Sebagai Pendeteksi Malware Pada Layanan Cloud Computing

Yang dipersiapkan dan disusun oleh :
Nama : Arya Ervan Leoresta
NIM : 10650062
Telah dimunaqasyahkan pada : Jum'at, 29 Agustus 2014
Nilai Munaqasyah : B
Dan dinyatakan telah diterima oleh Fakultas Sains dan Teknologi UIN Sunan Kalijaga

TIM MUNAQASYAH :

Ketua Sidang

Dr. Imam Riadi, M.Kom
NIY . 60020397

Penguji I

Sumarsono, M.Kom
NIP.19710209 200501 1 003

Penguji II

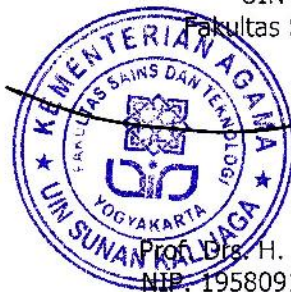
Arief Ikhwani W, M.Cs
NIP.

Yogyakarta, 26 September 2014

UIN Sunan Kalijaga

Fakultas Sains dan Teknologi

Dekan



Prof. Drs. H. Akh. Minhaji, M.A, Ph.D
NIP. 19580919 198603 1 002



SURAT PERSETUJUAN SKRIPSI/TUGAS AKHIR

Hal : Permohonan

Lamp : -

Kepada

Yth. Dekan Fakultas Sains dan Teknologi

UIN Sunan Kalijaga Yogyakarta

di Yogyakarta

Assalamu'alaikum wr. wb.

Setelah membaca, meneliti, memberikan petunjuk dan mengoreksi serta mengadakan perbaikan seperlunya, maka kami selaku pembimbing berpendapat bahwa skripsi Saudara:

Nama : Arya Ervan Leoresta

NIM : 10650062

Judul Skripsi : Implementasi Honeypot Sebagai Pendeteksi Malware Pada Layanan Cloud Computing

sudah dapat diajukan kembali kepada Program Studi Teknik Informatika Fakultas Sains dan Teknologi UIN Sunan Kalijaga Yogyakarta sebagai salah satu syarat untuk memperoleh gelar Sarjana Strata Satu dalam Teknik Informatika

Dengan ini kami mengharap agar skripsi/tugas akhir Saudara tersebut di atas dapat segera dimunaqsyahkan. Atas perhatiannya kami ucapkan terima kasih.

Wassalamu'alaikum wr. wb.

Yogyakarta, 21 Agustus 2014

Pembimbing

Dr. Imam Riadi, M.Kom
NTY: 60020397

PERNYATAAN KEASLIAN SKRIPSI

Yang bertanda tangan di bawah ini :

Nama : Arya Ervan Leoresta

Nim : 10650062

Program Studi : Teknik Informatika

Fakultas : Sains dan Teknologi

Menyatakan bahwa skripsi dengan judul **Implementasi Honeypot Sebagai Pendeteksi Malware Pada Layanan Cloud Computing** tidak terdapat pada karya yang pernah diajukan untuk memperoleh gelar sarjana di suatu Perguruan Tinggi, dan sepengetahuan saya tidak terdapat karya atau pendapat yang pernah ditulis oleh orang lain, kecuali yang secara tertulis diacu dalam naskah ini dan disebutkan dalam daftar pustaka.

Yogyakarta, 21 Agustus 2014

Yang Menyatakan,



Arya Ervan Leoresta
NIM : 10650062

MOTTO

Man Jadda Wa Jadda

“Siapa yang bersungguh-sungguh, Pasti akan berhasil”

Man Shobaro Zafiro

“Siapa yang sabar akan Beruntung”

Man Saaro ‘Alaa Darbi Washola

“Siapa yang berjalan di jalur-Nya akan sukses”

Impossible is Nothing

“Tidak ada yang tidak mungkin”

Gusti iku cedhak tanpa senggolan, adoh tanpa wangenan

Nrimo Ing Pandum

KATA PENGANTAR

Alhamdulillahirobbil ‘alamin. Puji Syukur penulis panjatkan kepada Allah SWT yang telah memberikan nikmat yang tiada batas karena dengan restu-Nya pelaksanaan dan penyusunan skripsi yang berjudul “Implementasi *Honeypot* sebagai pendeteksi *malware* pada layanan *cloud computing*” dapat diselesaikan sebagai persyaratan menyelesaikan Sarjana Strata Satu (SI) jurusan Teknik Informatika, Fakultas Sain dan Teknologi Universitas Islam Negeri Yogyakarta. Shalawat dan salam senantiasa tercurah kepada nabiullah Muhammad SAW, yang telah membawa risalah keislaman dari zaman jahiliyah menuju zaman islamiyah.

Pelaksanaan dan penyusunan skripsi ini tidak terlepas dari bantuan dan dukungan berbagai pihak. Oleh karena, ucapan terimakasih penulis sampaikan kepada:

1. Ibu yang tercinta, almarhum Bapak yang tersayang, Kedua kakak yang perhatian serta segenap keluarga yang telah memberikan dorongan moral dan material.
2. Bapak Prof. Drs. H. Akh. Minhaji, M. A., Ph.D., selaku Dekan Fakultas Sains & Teknologi UIN Sunan Kalijaga.
3. Bapak Agus Mulyanto, M.Kom, selaku Ketua Program Studi Teknik Informatika Fakultas Sains & Teknologi UIN Sunan Kalijaga.
4. Bapak Dr. H. Imam Riadi, M.Kom selaku Dosen Pembimbing yang telah membantu terselesaikannya skripsi ini.

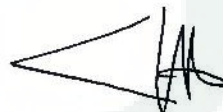
5. Bapak Sumarsono, M.Kom dan Bapak Arief Ikhwan W, M.Cs selaku dosen penguji yang telah memberikan masukan-masukan agar tugas akhir saya menjadi lebih baik
6. Bapak Bambang Sugiantoro, M.T. CompTIA selaku Dosen Pembimbing akademik yang selalu mendampingi dan membimbing semenjak awal kuliah
7. Bapak Agung Fatwanto, Ph. D selaku ketua PTIPD UIN Sunan Kalijaga dan segenap pengurus, khususnya mas R. Gatra yang telah memberikan dorongan moral dan material.
8. Bapak Drs. H. Rahmad Basuki, SH, MT selaku kepala sekolah SMK 2 Pengasih yang telah memberikan izin untuk penelitian penulis.
9. Pak Awan, Mas Yusuf dan Mas Munawir selaku pengurus laboratorium UIN Sunan Kalijaga yang telah memberikan izin tempat untuk penelitian penulis.
10. Fava dan Gallant yang sabar membimbing dan memotivasi untuk tidak menyerah
11. Sahabat-sahabat seperjuangan dalam mengerjakan skripsi di Lab, khususnya Feri, Faisal, Hanan dan Fajar yang saling memberi semangat dan ilmunya,
12. Sahabat-sahabat Kost Sugeng khususnya Syafi'i, Fanni, Akhyar, Yosi, Dina, Hanif, Yazid, Amir, Fuad yang selalu memberikan motivasi dan semangat. Terimakasih atas ilmu serta suka dan dukanya.
13. Sahabat-sahabat seperjuangan Teknik Informatika (2010) yang tidak bisa disebutkan satu per satu. Semoga kebersamaan ini senantiasa diertakan. Terimakasih atas sharing ilmunya.

14. Sahabat seperjuangan SMA khususnya Rasyid, Uswa, Bima, Widi, Hestu, Finan, Zella, Viali dan Ardi. Terimakasih atas inspirasi, semangat dan masukannya.
15. Bapak Nur, Bapak Suwono dan Bapak Didit selaku satpam UIN Sunan Kalijaga yang selalu membukakan pintu Laboratorium Terpadu setiap pagi
16. Ulfa Dina yang selalu memberikan semangat dan doa.
17. Semua pihak yang tidak bisa sebutkan satu per satu, terima kasih atas segala bantuannya.

Semoga Allah SWT memberikan balasan yang berlipat ganda dari segala kebaikan yang telah dilakukan. Penulis menyadari bahwa skripsi ini sangatlah jauh dari nilai kesempurna. Oleh karena itu, kritik dan saran yang membangun penulis harapkan sebagai masukan kearah yang lebih baik lagi.

Demikian penulis berharap agar laporan penelitian ini dapat bermanfaat bagi kita semua.

Yogyakarta, 29 Oktober 2014



Arya Ervan Leoresta
NIM.10650062

HALAMAN PERSEMBAHAN

Allah SWT, yang selalu melimpahkan banyak rahmat dan kenikmatan-Nya sehingga skripsi ini dapat terselesaikan dengan lancar

*Nabi akhiruzzaman Muhammad SAW,
semoga shalawat senantiasa terhatur kepadamu.*

*Almarhum Ayahanda Sukardi dan Ibunda Siti Asmunia, terimakasih atas
bimbingan moral dan spiritualnya selama ini.
Semoga kalian berdua selalu dijunjung tinggi haknya di dunia maupun
di akhirat.*

*Kedua kakak, Erna dan Erwin yang senantiasa
Menyemangati untuk terus maju.*

*Almamater tercinta Teknik Informatika
UIN Sunan Kalijaga Yogyakarta.
Inilah yang bisa aku torehkan sebagai
cinderamata hasil pembelajaranku.*

DAFTAR ISI

HALAMAN JUDUL	i
HALAMAN PENGESAHAN	ii
HALAMAN PERSETUJUAN SKRIPSI	iii
HALAMAN PERNYATAAN KEASLIAN SKRIPSI.....	iv
MOTTO	v
KATA PENGANTAR	vi
HALAMAN PERSEMBAHAN	ix
DAFTAR ISI.....	x
DAFTAR GAMBAR.....	xii
DAFTAR TABEL	xiii
DAFTAR LIST	xiv
INTISARI	xv
ABSTRACT	xvi
BAB I PENDAHULUAN	
1.1 Latarbelakang masalah.....	1
1.2 Rumusan masalah.....	2
1.3 Batasan masalah	2
1.4 Tujuan penelitian.....	3
1.5 Manfaat penelitian.....	3
1.6 Keaslian penelitian	3
BAB II TINJAUAN PUSTAKA.....	4
2.1 Tinjauan pustaka	4
2.2 Landasan teori	7
2.2.1 Jaringan komputer	7
2.2.2 Topologi Jaringan Komputer.....	14
2.2.3 Keamanan Komputer.....	18
2.2.4 Honeypot	21
2.2.4.1 Kategori Honeypot.	23
2.2.4.2 Tingkat Keterlibatan Honeypot	25
2.2.4.3 Perbandingan Karakteristik Honeypot	31
2.2.4.4 Penempatan Honeypot Pada Jaringan	32
2.2.5 Cloud Computing	33
2.2.6 Malware.....	38
2.2.7 Dionaesa.....	40
2.2.8 DionaesaFR.....	42
2.2.9 Proxmox VE.....	42
2.2.10 Ubuntu	44
2.2.11 Database	45
2.2.12 Shorewall.....	45
2.2.13 Black Box Testing.....	46
2.2.14 Stress Testing	46
BAB III METODE PENELITIAN	47
3.1 Studi Pendahuluan.....	47
3.2 Pengumpulan Data	47

3.2.1 Studi Literatur	47
3.2.2 Observasi.....	47
3.3 Kebutuhan Perangkat Penulis.....	48
3.2.1. Pendekatan sisi perangkat keras (Hardware).....	48
3.2.2. Pendekatan sisi perangkat lunak (Software)	48
3.4 Langkah kerja penelitian	49
3.4.1. Perancangan Jaringan dan Sistem	49
3.4.2. Pengujian Sistem	51
BAB IV HASIL DAN PEMBAHASAN	53
4.1 Analisa kebutuhan sistem.....	53
4.2 Perancangan Jaringan dan Sistem	54
4.2.1 Perancangan Topologi.....	54
4.2.2 Perancangan Jaringan.....	57
4.2.3 Perancangan Sistem.....	57
4.2.3.1 Perancangan Virtualisasi Server.....	58
4.2.3.2 Perancangan Firewall	58
4.3 Implementasi Rancangan Sistem	58
4.3.1. Instalasi Hypervisor Proxmox VE.....	58
4.3.2. Instalasi VM Sistem Operasi Ubuntu 12.04.....	60
4.3.3 Konfigurasi Jaringan	63
4.3.4 Instalasi dan Konfigurasi Honeypot Dionaera.....	64
4.3.5 Instalasi DionaeraFR	68
4.3.6 Membangun Shorewall	70
4.4 Pengujian sistem.....	74
4.4.1. Pengujian Fungsionalitas.....	75
4.4.2. Pengujian Stress Test.....	81
BAB V KESIMPULAN	90
5.1 Kesimpulan.....	90
5.2 Saran.....	91
DAFTAR PUSTAKA	92
LAMPIRAN	
LAMPIRAN A	95
LAMPIRAN B.....	100
LAMPIRAN C	104
LAMPIRAN D	114
LAMPIRAN E.....	141

DAFTAR GAMBAR

Gambar 2.1 Jaringan peer to peer	10
Gambar 2.2 Jaringan Client Server.....	11
Gambar 2.3 Local Area Network (LAN).....	12
Gambar 2.4 Metropolitan Area Network (MAN).....	12
Gambar 2.5 Wide Area Network (WAN).....	13
Gambar 2.6 Topologi Bus	14
Gambar 2.7 Topologi Ring	15
Gambar 2.8 Topologi Star	16
Gambar 2.9 Topologi Tree	17
Gambar 2.10 Topologi Mesh.....	18
Gambar 2.11 Sistem Topologi Honeypot Tunggal.....	23
Gambar 2.12 Low Interaction Honeypot.....	26
Gambar 2.13 Medium Interaction Honeypot.....	29
Gambar 2.14 High Interaction Honeypot	30
Gambar 2.15 Penempatan Honeypot	32
Gambar 2.16 Logo Proxmox	42
Gambar 2.17 Logo Ubuntu	44
Gambar 2.18 Logo Shorewall.....	45
Gambar 3.1 Rancangan Sistem Honeypot.....	50
Gambar 4.1 Topologi Sebelum Diterapkan Honeypot	55
Gambar 4.2 Topologi Jaringan Penelitian	56
Gambar 4.3 Create Virtual Machine.....	61
Gambar 4.4 Install Ubuntu	62
Gambar 4.5 Konfigurasi Shorewall <i>Interfaces</i>	71
Gambar 4.6 Konfigurasi <i>Zone</i>	72
Gambar 4.7 Konfigurasi Policy	73
Gambar 4.8 Konfigurasi <i>rules</i>	74
Gambar 4.9 Web Interface DionaeaFR	71
Gambar 4.10 Data Malware.....	77
Gambar 4.11 Data Detail Malware.....	77
Gambar 4.12 Port Dionaea	78
Gambar 4.13 Service Dionaea.....	78
Gambar 4.14 Negara Penyerang.....	79
Gambar 4.15 IP Negara Penyerang	79
Gambar 4.16 Diagram Waktu Penyerangan	80
Gambar 4.17 Hasil Stress	82
Gambar 4.18 Ping Jaringan	83
Gambar 4.19 Scanning Port.....	84
Gambar 4.20 Grafik Persentase dari hasil Pengujian Fungsional	88
Gambar 4.21 Grafik Persentasi dari hasil Pengujian Stress Test	89

DAFTAR TABEL

Tabel 2.1 Tabel daftar penelitian.....	6
Tabel 2.2 Perbandingan Tingkat Keterlibatan Honeypot	31
Tabel 3.1 Daftar Pertanyaan Uji Fungsional	52
Tabel 4.1 Pembagian IP Adress pada Proxmox	57
Tabel 4.2 Daftar Penguji	85
Tabel 4.3 Daftar Pertanyaan Uji Fungsional	86
Tabel 4.4 Daftar Pertanyaan Stress Test.....	86



DAFTAR LIST

List 1. Konfigurasi Interfaces	60
List 2. Konfigurasi Alamat IP Ubuntu.....	63
List 3. Konfigurasi File Resolv.conf	64
List 4. Instalasi Dependencies Untuk Dionaea	65
List 5. Instal Dionaea	65
List 6. Dionaea Start	66
List 7. Cek Dionaea	66
List 8. Konfigurasi File Log	67
List 9. Konfigurasi File Logging	67
List 10. Instalasi Python Pip	68
List 11. Pip Instal Django	68
List 12. Instalasi Node.js	69
List 13. Instalasi DionaeaFR	69
List 14. Start DionaeaFR	70
List 15. Aktivasi Shorewall	71
List 16. Exploit SMB.....	76
List 17. Stress Tes.....	81
List 18. Scanning Port	83

IMPLEMENTASI HONEYPOT SEBAGAI PENDETEKSI MALWARE PADA LAYANAN CLOUD COMPUTING

Arya Ervan Leoresta

10650062

INTISARI

Keamanan menjadi konsentrasi dan fokus utama seiring dengan ketergantungan masyarakat akan teknologi informasi. Ancaman keamanan pun sangat beragam, mulai dari *virus* atau *worm*, *malware*, hingga penanaman *backdoor*. Hal ini dapat diminimalkan dengan memaksimalkan identifikasi celah keamanan sedini mungkin. Implementasi sistem *honeypot* merupakan solusi keamanan yang komprehensif meskipun *honeypot* hanya memiliki ruang lingkup deteksi dan respon, namun dapat merekam semua kemungkinan celah keamanan. Penelitian ini bertujuan untuk mendeteksi serangan *malware* menggunakan *Honeypot* pada layanan *cloud computing*.

Penelitian ini menggunakan metode pengumpulan data berupa studi literatur dan observasi langsung ke tempat penelitian. Penelitian menekankan pada implementasi *Honeypot* untuk mendeteksi serangan *malware* pada layanan *cloud computing*. Tahapan penelitian terbagi dalam beberapa langkah yaitu perancangan sistem dan jaringan, analisis dan perancangan sistem implementasi rancangan sistem serta pengujian sistem menggunakan metode *black box* dan *stress test*.

Honeypot dijalankan selama seminggu menggunakan IP publik berhasil menangkap *malware* yang menyusup ke sistem. Port 5060 yang sering digunakan untuk layanan *voice* dan media melalui Internet paling banyak mendapat serangan dengan memperoleh 1301 ancaman. Data yang didapatkan lainnya adalah Amerika Serikat, sebagai negara yang sering melakukan ancaman serangan dengan prosentase 36,31% dan berdasarkan waktu dalam satu hari, ancaman serangan paling banyak terjadi saat malam hari pada pukul 20.00-04.00 WIB. Hasil pengujian *stress testing* menggunakan aplikasi *stress*, Internet berjalan lancar dan server bekerja dengan baik dengan *load cpu* sebesar 99,91 %. Sistem *Honeypot* pada layanan *cloud computing* dapat berjalan dengan baik dan informasi *Honeypot* dapat digunakan *administator* jaringan untuk melakukan pencegahan ancaman serangan selanjutnya.

Kata kunci : *Honeypot*, *Cloud Computing*, *Dionaea*, *Malware*, *Port*

IMPLEMENTATION OF HONEYPOT TO DETECT MALWARE ATTACKS IN THE CLOUD COMPUTING SERVICES

Arya Ervan Leoresta

10650062

ABSTRACT

Security had become the main focus and public concentration along society's dependance on information technology. There were various security threat, starting from viruses or worms, malware, up to backdoor attacks. Security threats could be decreased by optimizing identification of security holes earliest. Honeypot system implementation was a comprehensive security solution. Despite it had only the scope of detection and response, it was able to record all possible security holes. This research aimed to detect malware attacks by using Honeypot towards cloud computing services.

This study uses data collection methods such as study of literature and observation. The study emphasizes on the implementation of the honeypot to detect malware attacks on cloud computing services. Stages of research is divided into several steps that the design of the system and network, analysis and prototyping, system design implementation and system testing using black box method and stress test.

Honeypot, which had been run for a week using public IP, successfully captured malware infiltrating into the system. Port 5060, which was commonly used for voice and media services by Internet, mostly gained 1301 threat attacks. The United States was the only country that frequently gave threat attack with percentage 36.31%. In addition, based on the time in each day, the attacks threat occurred mostly during the night, between 20:00 up to 4:00 WIB. Test results of stress testing using stress applications, the Internet runs smoothly and the server works well with cpu load by 99.91%. Honeypot systems on cloud computing services can run well and the information can be used honeypot network administrator to make the next attack threat prevention.

Keywords : *Honeypot, Cloud Computing, Dionaean, Malware, Port*

BAB I

PENDAHULUAN

1.1 Latar Belakang Masalah

Keamanan menjadi konsentrasi dan fokus utama seiring dengan ketergantungan masyarakat akan teknologi informasi. Ancaman keamanan pun sangat beragam, mulai dari *virus* atau *worm*, *malware*, penanaman *backdoor*, hingga penyerangan *denial of service* yang menyebabkan server akan mati dan tidak dapat beroperasi lagi sehingga otomatis tidak dapat memberikan pelayanan. Berbagai ancaman keamanan teknologi informasi dapat diminimalkan dengan memaksimalkan identifikasi celah keamanan sedini mungkin.

Kombinasi dan perpaduan keamanan *software* dan perangkat *hardware* merupakan solusi keamanan teknologi informasi yang komprehensif. Implementasi sistem *honeypot* merupakan solusi keamanan yang komprehensif meskipun *honeypot* hanya memiliki ruang lingkup deteksi dan respon, namun dapat merekam semua kemungkinan celah keamanan. *honeypot* adalah sistem yang dibuat seperti desain sistem aslinya agar sengaja diserang atau disusupi, sehingga informasinya dapat dieksplor untuk melakukan tindakan pencegahan selanjutnya.

Cloud computing atau komputasi awan merupakan pemanfaatan teknologi komputasi pihak ketiga, baik dalam bentuk *software* maupun *hardware* yang menyediakan layanan penyimpanan melalui sebuah jaringan Internet. Saat ini *cloud computing* menjadi tren baru, perusahaan dan masyarakat beramai-ramai

menggunakannya, mulai *file* musik sampai data perusahaan yang bersifat penting disimpan ke media penyimpanan *cloud*. Hal tersebut menimbulkan ancaman terhadap *cloud computing* seperti serangan *malware* semakin meningkat, sehingga kehadiran *honeypot* memberikan solusi dalam meningkatkan keamanan.

1.2 Rumusan Masalah

1. Bagaimana merancang sistem pendeteksi *malware* menggunakan *Honeypot*
2. Bagaimana implementasi sistem *Honeypot* pada *cloud computing* berbasis *Virtual Environment (Proxmox VE)*
3. Apakah sistem *Honeypot* menggunakan *Dionaea* dapat mendeteksi *malware* yang menyusup ke jaringan

1.3 Batasan Masalah

1. Penelitian ini hanya membahas implementasi *Honeypot* menggunakan *Dionaea*.
2. Sistem operasi yang digunakan adalah *Proxmox Virtual Environment 3.1*.
3. *Dionaea* digunakan untuk mendeteksi serangan *malware*.
4. Layanan *cloud computing* pada tingkat Iaas (*Infrastructure as service*).
5. Menggunakan aturan dalam *firewall* untuk mengarahkan serangan ke *Honeypot*
6. Pengujian serangan *malware* menggunakan *exploit* dari *tool* Metasploit
7. Data *malware* berasal dari *website* virustotal.com

1.4 Tujuan Penelitian

1. Merancang sistem pendeteksi *malware* menggunakan *Honeypot*
2. Mengimplementasi sistem *honeypot* pada *cloud computing* menggunakan *Proxmox Virtual Environment*
3. Mendeteksi *malware* menggunakan *Dionaea*

1.5 Manfaat Penelitian

1. Membantu *administrator* jaringan dalam mengantisipasi serangan ke *server*.
2. *Traffic* serangan *Malware* dapat digunakan untuk melakukan tindakan pencegahan selanjutnya.

1.6 Keaslian Penelitian

Penelitian umum terkait *Honeypot* sudah pernah dilakukan sebelumnya. Akan tetapi penelitian terdahulu masih belum ada yang pernah mengimplementasi *Honeypot* menggunakan *dionaea* pada layanan *cloud computing*. Pada penelitian ini penulis akan melakukan implementasi *Honeypot* untuk mendeteksi serangan *malware* pada layanan *cloud computing*, yang mana sejauh ini topik serupa belum pernah ada dilakukan sebelumnya, khususnya di UIN Sunan Kalijaga Yogyakarta.

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan hasil pengujian yang telah dilakukan, maka dapat diambil kesimpulan sebagai berikut:

1. Hasil dari penerapan *Honeypot* yang telah dilakukan adalah *port* 5060 sering mengalami serangan dengan mendapatkan 1301 serangan. *Port* 5060 merupakan layanan SIP yang sering digunakan untuk *voice* dan media melalui Internet. Hal ini menyebabkan *port* yang paling rentan untuk disusupi serangan *Trojan* dan *Rootkits*. Negara yang sering melakukan ancaman adalah Amerika Serikat (36,31%), China (31,98%), Indonesia (8,46%), Palestina Territory (7,84%), dan Brazil (5,15%). Penyerangan berdasarkan waktu dalam satu hari, ancaman yang paling banyak terjadi pada malam hari (pukul 20.00-04.00), lalu siang hari (pukul 12.00-20.00), kemudian pagi hari (pukul 04.00-12.00).
2. Implementasi *Honeypot* pada layanan *cloud computing* mendapatkan hasil bahwa *hardware* dapat berjalan normal dalam layanan virtualisasi dengan *load cpu* berada pada angka 99,91% berdasarkan pengujian menggunakan *stress test*.
3. Implementasi *Honeypot* sebagai pendeteksi *malware* pada layanan *cloud computing* berhasil dilaksanakan dengan tahapan penelitian yakni

perancangan jaringan, implementasi rancangan sistem dan pengujian sistem. Pada implementasinya sistem ini dapat mendeteksi *malware* dengan menjalankannya selama 7 hari.

5.2 Saran

Berdasarkan penelitian yang telah dilakukan penulis ada kelebihan, namun juga banyak kekurangan yang masih membutuhkan saran-saran untuk mendukung kesempurnaan,. Saran tersebut diantaranya sebagai berikut:

1. Penelitian kedepan waktu pengujian lebih lama, agar data yang diperoleh lebih banyak dan variasi.
2. Penelitian selanjutnya menggunakan *Honeypot* keterlibatan tinggi, agar mendapat serangan yang lebih *advance*.
3. Membuat program untuk mengarahkan *Honeypot* dalam layanan *cloud computing*, agar lebih maksimal dalam mengalihkan penyerang dari *server* utama.
4. Mempunyai IP publik khusus untuk mendapatkan informasi serangan yang maksimal

Daftar Pustaka

- Agriyanto, S. (2004). *Teknik Mengenali Penyerang Sistem Komputer dan Internet dengan Honeypots*, Paper, ITB, Bandung, <http://budi.ihsan.co.id/courses/el7010/2003/sakri-report.pdf>, diakses pada 29 Oktober 2013.
- Aneldi, W. (2012). *Honeypot*. Paper, Prodi Teknologi Informasi Universitas Kristen Maranatha Bandung, Bandung.
- Arief, M. (2012). *Implementasi Honeypot Dengan Menggunakan Dionaea Di Jaringan Hotspot Fizz*. Skripsi. Politeknik Tekom Bandung, Bandung.
- Brwon, S, dkk. (2012). *Honeypots in The Cloud*. Skripsi. University of Winconsin Madison, Madison.
- Budi, S. (2013). *Implementasi Cluster Pada Web Server Berbasis Cloud Computing*. Skripsi, Prodi Teknik Informatika Universitas Islam Negeri Sunan Kalijaga Yogyakarta, Yogyakarta.
- Gollman, D. (2006). *Computer Security*. Chichester: John Willey & Sons, Inc.
- <http://dionaea.carnivore.it/>, diakses pada 1 Oktober 2013.
- <http://indonesiancloud.com/id/content/cloud-computing>, diakses pada 29 Oktober 2013.
- <http://www.catatanteknisi.com/2011/12/pengertian-jenis-jenis-malware.html>, diakses pada 1 November 2013.
- Komputer, Wahana. (2003). *Konsep jaringan komputer dan pengembangannya*. Jakarta: Salemba Infotek.
- L. Spitzner. (2002). *Honeypots: Tracking Hackers*, Boston.
- Mell, P., and Grance, T. (2011). *The NIST Definition of Cloud Computing. Computer Security Division Information Technology Laboratory National Institute of Standards and Technology. Gaithersburg USA* , Sumber: <http://www.nist.gov/itl/cloud/upload/cloud-def-v15.pdf>.
- Mustofa, M, dkk. (2013). *Penerapan Sistem Honeypot dan IDS Pada Jaringan Nirkabel (Hotspot)*. Skripsi. Prodi Teknik Informatika Universitas Ahmad Dahlan Yogyakarta, Yogyakarta.

- Oetomo, B. S. D. (2003). *“Konsep dan perancangan jaringan komputer”*. Yogyakarta: Penerbit Andi
- Prakoso, S. (2005). *Jaringan Komputer Linux: Konsep, Dasar, Aplikasi, Keamanan, dan Penerapan*. Yogyakarta: ANDI.
- Pressman, R. S. (1997). *Rekayasa Perangkat Lunak*. Yogyakarta: Andi.
- Provos, Niels & Thorsten Holz. (2007). *Virtual Honeypots: From Botnet Tracking to Intrusion Detection*. Boston : Pearson Education, Inc.
- Pujianto, A. (2014). *Implementasi Traffic Shapping Dengan Metode CBQ Pada Jaringan Komputer Pondok Pesantren AL-Jailani Yogyakarta*. Skripsi. Prodi Teknik Informatika Universitas Islam Negeri Sunan Kalijaga Yogyakarta, Yogyakarta.
- Purbo, O. W. (2011). *Petunjuk Praktisi Cloud Computing Menggunakan Open Source*. Jakarta : Opensource.
- Purbo, O. W. (2008). *Keamanan Jaringan Internet*. Jakarta: PT Elex Media Komputindo.
- Rahardjo, B. (2005). *Keamanan Sistem Informasi Berbasis Internet*. Bandung: Insan Infonesia.
- Saputra, R (2012). *Implementasi Shorewall Pada Yayasan Iba Palembang*. Skripsi, Sekolah. Tinggi Manajemen Informatika dan Komputer Palcomtech Palembang, Palembang.
- Setiawan, I. (2012). *IDS, Dngap dan Honeypot*. Paper. Prodi Teknik Informatika Universitas Soedirman Purwokerto, Purwokerto.
- Sugianto, Masim Vavai. (2011). *Panduan Virtualisasi & Cloud Computing pada Sistem Linux*. Sumber: <http://vavai.com/wp-content/uploads/2011/01/Implementasi-Virtualisasi-Cloud-Computing-pada-Sistem-Linux.pdf>
- Sunardi, A dkk. (2012). *Implementasi dan Evaluasi Honeypot Dionaea dan Glastopf di ID-SIRTII*. Skripsi. Prodi Sistem Komputer Universitas Bina Nusantara Jakarta, Jakarta.
- Syafrizal, M. (2005). *Pengantar Jaringan Komputer*. Yogyakarta: Penerbit Andi.
- Thomas, T. M. (2004). *Network Security First-Step*. Boston: Pearson Education, Inc, publishing as Cisco Press.

Unswagati. (2010). *Keamanan Jaringan Komputer*. http://unswagati-crb.ac.id/component/option,com_phocadownload/Itemid,73/download,55/id,11/view,category/. Tanggal akses 10 Desember 2013.

Widodo, T (2011). *Analisis Log Honeypot Sebagai Pendukung Keamanan Jaringan*. Skripsi. Prodi Teknik Informatika Universitas Islam Negeri Sunan Kalijaga Yogyakarta, Yogyakarta.

Zulkarnaen, D. (2012). *Implementasi Honeypot Sebagai Alat Bantu Deteksi Keamanan Jaringan Pada Kantor Pengawasan Dan Pelayanan Bea dan Cukai Tipe A2 Palembang*. Skripsi. Fakultas Teknik Informatika STMIK Palcomtech Palembang, Palembang.

LAMPIRAN A

KODE SUMBER (SOURCE CODE) KONFIGURASI SERVER DIONAEA

Nama File : Interfaces

Lokasi : Ubuntu 12.04 Server Edition

This file describes the network interfaces available on your system

and how to activate them. For more information, see interfaces(5).

The loopback network interface

auto lo

iface lo inet loopback

The primary network interface

auto eth0

iface eth0 inet static

address 192.168.100.3

netmask 255.255.255.0

network 192.168.100.0

broadcast 192.168.100.255

gateway 192.168.100.1

dns-nameservers 8.8.8.8

dns-search lab.jarkom.uin

Nama File : Resolv.conf

Lokasi : Ubuntu 12.04 Server Edition

Dynamic resolv.conf(5) file for glibc resolver(3) generated by resolvconf(8)

DO NOT EDIT THIS FILE BY HAND -- YOUR CHANGES WILL BE

DAFTAR RIWAYAT HIDUP



Nama : Arya Ervan Leoresta
Tempat, tanggal lahir : Magelang, 30 Juli 1992
Jenis Kelamin : Laki-laki
Agama : Islam
Alamat Asal : Sekaran RT 02 / 04 Banyurojo, Mertoyudan, Magelang
No. HP : 08994563861
Email : aryaervanleoresta19@gmail.com

Riwayat Pendidikan :

1. SD Kartika XV-1 Panca Arga (1998-2004)
2. SMP Negeri 7 Magelang (2004-2007)
3. SMA Negeri 1 Mertoyudan (2007-2010)
4. S1 Teknik Informatika UIN Sunan Kalijaga Yogyakarta (2010-2014)

OVERWRITTEN

nameserver 8.8.8.8

search lab.jarkom.uin

Nama File : Dionaea.conf

Lokasi : Ubuntu 12.04 Server Edition

```
logging = {
    default = {
        // file not starting with / is taken relative to LOCALESTATEDIR (e.g.
/opt/dionaea/var)

        file = "log/dionaea.log"

        levels = "error"

        domains = "*"
    }

    errors = {
        // file not starting with / is taken relative to LOCALESTATEDIR (e.g.
/opt/dionaea/var)

        file = "log/dionaea-errors.log"

        levels = "error"

        domains = "*"
    }
}
```

Nama Konfigurasi : liblcfg

Lokasi : Ubuntu 12.04 Server Edition

```
git clone git://git.carnivore.it/liblcfg.git liblcfg
```

```
cd liblcfg/code
```

```
autoreconf -vi
```

```
./configure --prefix=/opt/dionaea
```

```
make install
```

Nama Konfigurasi : libemu

Lokasi : Ubuntu 12.04 Server Edition

```
git clone git://git.carnivore.it/libemu.git libemu
```

```
cd libemu
```

```
autoreconf -vi
```

```
./configure --prefix=/opt/dionaea
```

```
make install
```

Nama Konfigurasi : libnl

Lokasi : Ubuntu 12.04 Server Edition

```
apt-get install libnl-3-dev libnl-genl-3-dev libnl-nf-3-dev libnl-route-3-dev
```

Nama Konfigurasi : libev

Lokasi : Ubuntu 12.04 Server Edition

```
wget http://dist.schmorp.de/libev/Attic/libev-4.04.tar.gz
```

```
tar xzf libev-4.04.tar.gz
```

```
cd libev-4.04
```



```
./configure --prefix=/opt/dionaea
```

```
make install
```

Nama Konfigurasi : python

Lokasi : Ubuntu 12.04 Server Edition

```
wget http://www.python.org/ftp/python/3.2.2/Python-3.2.2.tgz
```

```
tar xzf Python-3.2.2.tgz
```

```
cd Python-3.2.2/
```

```
./configure --enable-shared --prefix=/opt/dionaea --with-computed-gotos \
```

```
--enable-ipv6 LDFLAGS="-Wl,-rpath=/opt/dionaea/lib/ -L/usr/lib/x86_64-  
linux-gnu/"
```

```
make
```

```
make install
```

Nama Konfigurasi : cython

Lokasi : Ubuntu 12.04 Server Edition

```
wget http://cython.org/release/Cython-0.15.tar.gz
```

```
tar xzf Cython-0.15.tar.gz
```

```
cd Cython-0.15
```

```
/opt/dionaea/bin/python3 setup.py install
```

Nama Konfigurasi : libpcap

Lokasi : Ubuntu 12.04 Server Edition

```
wget http://cython.org/release/Cython-0.15.tar.gz
```

```
tar xzf Cython-0.15.tar.gz
```

```
cd Cython-0.15
```

```
/opt/dionaea/bin/python3 setup.py install
```

Nama Konfigurasi : libpcap

Lokasi : Ubuntu 12.04 Server Edition

```
git clone git://github.com/rep/dionaea.git
```

```
cd dionaea
```

```
autoreconf -vi
```

Nama Konfigurasi : dionaea

Lokasi : Ubuntu 12.04 Server Edition

```
./configure --with-lcfg-include=/opt/dionaea/include/ \
```

```
--with-lcfg-lib=/opt/dionaea/lib/ \
```

```
--with-python=/opt/dionaea/bin/python3.2 \
```

```
--with-cython-dir=/opt/dionaea/bin \
```

```
--with-udns-include=/opt/dionaea/include/ \
```

```
--with-udns-lib=/opt/dionaea/lib/ \
```

```
--with-emu-include=/opt/dionaea/include/ \
```

```
--with-emu-lib=/opt/dionaea/lib/ \
```

```
--with-gc-include=/usr/include/gc \
```

```
--with-ev-include=/opt/dionaea/include \
```

```
--with-ev-lib=/opt/dionaea/lib \
```

```
--with-nl-include=/opt/dionaea/include \
```

```
--with-nl-lib=/opt/dionaea/lib/ \
```

```
--with-curl-config=/usr/bin/ \
```

```
--with-pcap-include=/opt/dionaea/include \
```

```
--with-pcap-lib=/opt/dionaea/lib/
```

```
make
```

```
make install
```

LAMPIRAN B

KODE SUMBER (SOURCE CODE) KONFIGURASI DIONEAFR

Nama Konfigurasi : Python pip

Lokasi : Ubuntu 12.04 Server Edition

```
apt-get installpython-pip python-netaddr
```

```
pip install Django
```

```
pip install pygeoip
```

```
pip install django-pagination
```

```
pip install django-tables2
```

```
pip install django-compressor
```

```
pip install django-htmlmin
```

Nama Konfigurasi : [django-tables2-simplefilter](#)

Lokasi : Ubuntu 12.04 Server Edition

```
cd/opt/
```

```
wget https://github.com/benjiec/django-tables2-simplefilter/archive/master.zip -O  
django-tables2-simplefilter.zip
```

```
unzip django-tables2-simplefilter.zip
```

```
mvdjango-tables2-simplefilter-master/ django-tables2-simplefilter/
```

```
cddjango-tables2-simplefilter/
```

```
python setup.pyinstall
```

Nama Konfigurasi : [PySubnetTree](#)

Lokasi : Ubuntu 12.04 Server Edition

```
cd/opt/
```

```
git clone https://github.com/bro/pysubnettree.git
cdpysubnettree/
python setup.pyinstall
```

Nama Konfigurasi : Node.js

Lokasi : Ubuntu 12.04 Server Edition

```
cd/opt/
wget http://nodejs.org/dist/v0.8.16/node-v0.8.16.tar.gz
tarxzvf node-v0.8.16.tar.gz
cdnode-v0.8.16
./configure
make
Makeinstall
```

Nama Konfigurasi : [LESS](#)

Lokasi : Ubuntu 12.04 Server Edition

```
npm install -g less
```

Nama Konfigurasi : DionaeaFR

Lokasi : Ubuntu 12.04 Server Edition

```
cd/opt/
wget https://github.com/RootingPuntoEs/DionaeaFR/archive/master.zip -O
DionaeaFR.zip
unzip DionaeaFR.zip
Mv DionaeaFR-master/ DionaeaFR
```

Nama Konfigurasi : GeoIP dan GeoLite**Lokasi : Ubuntu 12.04 Server Edition**

```
cd/opt/
```

```
wget http://geolite.maxmind.com/download/geoip/database/GeoLiteCity.dat.gz
```

```
wget
```

```
http://geolite.maxmind.com/download/geoip/database/GeoLiteCountry/GeoIP.dat.  
gz
```

```
gunzip GeoLiteCity.dat.gz
```

```
gunzip GeoIP.dat.gz
```

```
mvGeoIP.dat DionaeaFR/DionaeaFR/static
```

```
mvGeoLiteCity.dat DionaeaFR/DionaeaFR/static
```

Nama Konfigurasi : Start DionaeFR**Lokasi : Ubuntu 12.04 Server Edition**

```
cd/opt/DionaeaFR/
```

```
python manage.py collectstatic#type yes when asked
```

```
python manage.py runserver 0.0.0.0:8000
```


LAMPIRAN C

KODE SUMBER (SOURCE CODE) PROXMOX

Nama File : Interfaces

Lokasi : Proxmox VE

```
# network interface settings

auto lo

iface lo inet loopback

#auto eth0

#iface eth0 inet dhcp

auto eth0

iface eth0 inet static
    address 10.10.200.253
    netmask 255.255.255.0
    network 10.10.200.0
    broadcast 10.10.200.255
    gateway 10.10.200.254
    post-up echo 1 > /proc/sys/net/ipv4/conf/eth0/proxy_arp

#dns-* options are implemented by the resolvconf package, if installed

#auto eth1

#iface eth1 inet manual

auto vmbr0

iface vmbr0 inet static
    address 192.168.100.1
    netmask 255.255.255.0
```

```
bridge_ports none  
bridge_stp off  
bridge_fd 0  
auto eth1  
iface eth1 inet static  
    address 192.168.10.1  
    netmask 255.255.255.0  
    network 192.168.10.0  
    broadcast 192.168.10.255
```

Nama File : Resolv.conf

Lokasi : Proxmox VE

```
nameserver 8.8.8.8
```

Nama File : Shorewall-Interfaces

Lokasi : Proxmox VE

```
#  
# Shorewall version 4.0 - Sample Interfaces File for one-interface configuration.  
# Copyright (C) 2006 by the Shorewall Team  
#  
# This library is free software; you can redistribute it and/or  
# modify it under the terms of the GNU Lesser General Public  
# License as published by the Free Software Foundation; either  
# version 2.1 of the License, or (at your option) any later version.  
#
```

See the file README.txt for further details.

#-----

For information about entries in this file, type "man shorewall-interfaces"

#####

FORMAT 1

#####

#ZONE	INTERFACE	BROADCAST	OPTIONS
net	eth0	detect	dhcp,nosmurfs,routefilter
loc	eth1	detect	nosmurfs,routefilter
dmz	vmbr0	detect	nosmurfs,bridge

Nama File : Shorewall-Zones

Lokasi : Proxmox VE

#

Shorewall version 4.0 - Sample Zones File for one-interface configuration.

Copyright (C) 2006 by the Shorewall Team

#

This library is free software; you can redistribute it and/or

modify it under the terms of the GNU Lesser General Public

License as published by the Free Software Foundation; either

version 2.1 of the License, or (at your option) any later version.

#

See the file README.txt for further details.

#-----

For information about entries in this file, type "man shorewall-zones"

#####

#ZONE	TYPE	OPTIONS	IN	OUT
#		OPTIONS		OPTIONS
fw	firewall			
net	ipv4			
loc	ipv4			
dmz	ipv4			

Nama File : Shorewall-Policy

Lokasi : Proxmox VE

```
#
# Shorewall version 4.0 - Sample Policy File for one-interface configuration.
# Copyright (C) 2006 by the Shorewall Team
#
# This library is free software; you can redistribute it and/or
# modify it under the terms of the GNU Lesser General Public
# License as published by the Free Software Foundation; either
# version 2.1 of the License, or (at your option) any later version.
#
# See the file README.txt for further details.
#-----
# For information about entries in this file, type "man shorewall-policy"
#####
#####
```

#SOURCE	DEST	POLICY	LOG LEVEL	LIMIT:BURST
---------	------	--------	-----------	-------------

From Firewall Policy

fw	net	ACCEPT	info	
fw	dmz	ACCEPT	info	
fw	fw	ACCEPT		
fw	loc	ACCEPT		

From DMZ Policy

dmz	fw	ACCEPT	info	
dmz	net	ACCEPT	info	
dmz	dmz	ACCEPT		
dmz	loc	ACCEPT		

From NET Policy

net	fw	REJECT	info	
net	dmz	REJECT	info	
net	loc	REJECT		

From LOC Policy

loc	fw	ACCEPT	info	
loc	dmz	ACCEPT	info	
loc	loc	ACCEPT		
loc	net	ACCEPT		

The FOLLOWING POLICY MUST BE LAST

all all REJECT info

Nama File : Shorewall-Rules

Lokasi : Proxmox VE

```
#
# Shorewall version 4.0 - Sample Rules File for one-interface configuration.
# Copyright (C) 2006 by the Shorewall Team
#
# This library is free software; you can redistribute it and/or
# modify it under the terms of the GNU Lesser General Public
# License as published by the Free Software Foundation; either
# version 2.1 of the License, or (at your option) any later version.
#
# See the file README.txt for further details.
#-----
-----
# For information on entries in this file, type "man shorewall-rules"
#####
#####
#####

#ACTION      SOURCE      DEST      PROTO DEST  SOURCE
ORIGINAL     RATE      USER/  MARK  CONNLIMIT  TIME
HEADERS      SWITCH

#
#          PORT  PORT(S)  DEST  LIMIT
GROUP

#SECTION ALL
```


#SECTION ESTABLISHED

#SECTION RELATED

SECTION NEW

Drop Ping from the "bad" net zone.. and prevent your log from being flooded..

Ping(ACCEPT) all fw

Permit all ICMP traffic FROM the firewall TO the net zone

ACCEPT fw net icmp

web server

HTTP(DNAT) net dmz:192.168.100.2

HTTPS(DNAT) net dmz:192.168.100.2

HTTPS(DNAT) net dmz:192.168.100.1 tcp 8006

DNAT net dmz:192.168.100.2 tcp 8888

DNAT net dmz:192.168.100.2 tcp 8081

ssh access

DNAT net dmz:192.168.100.1:22 tcp 1111

DNAT net dmz:192.168.100.3:22 tcp 3333

DNAT net dmz:192.168.100.2:22 tcp 2222

```
DNAT      net      dmz:192.168.100.5:22  tcp  5555
DNAT      net      dmz:192.168.100.10:22 tcp  1010
```

Honeypot

```
ACCEPT    net      all          tcp  445
#DNAT     net      dmz:192.168.100.3  tcp
21,80,443,42,135,445,1433,3306,5060,5061
#DNAT     net      dmz:192.168.100.3  udp  69
DNAT      net      dmz:192.168.100.3
```

Nama File : Shorewall.conf

Lokasi : Proxmox VE

```
#####
#####

#
# Shorewall Version 4 -- /etc/shorewall/shorewall.conf
#
# For information about the settings in this file, type "man shorewall.conf"
#
# Manpage also online at
http://www.shorewall.net/manpages/shorewall.conf.html

#####
#####

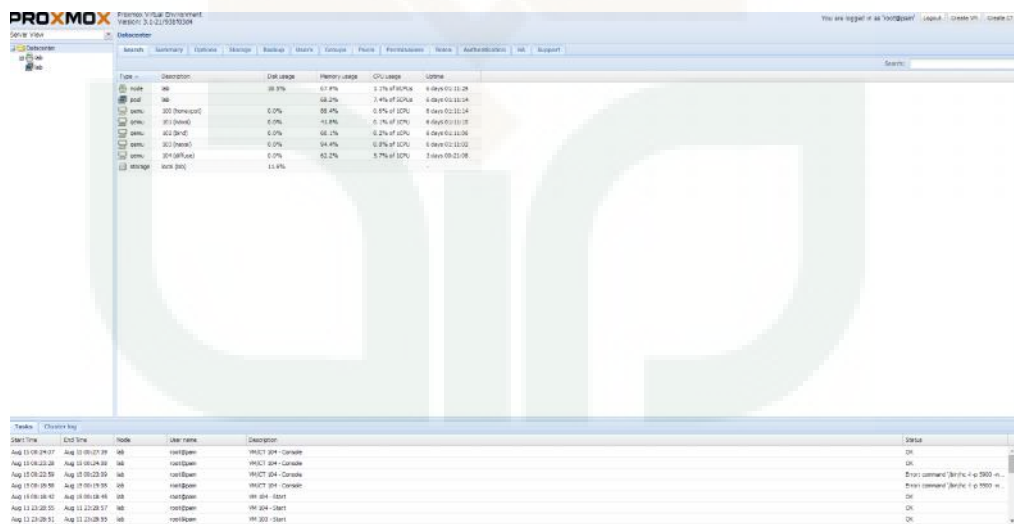
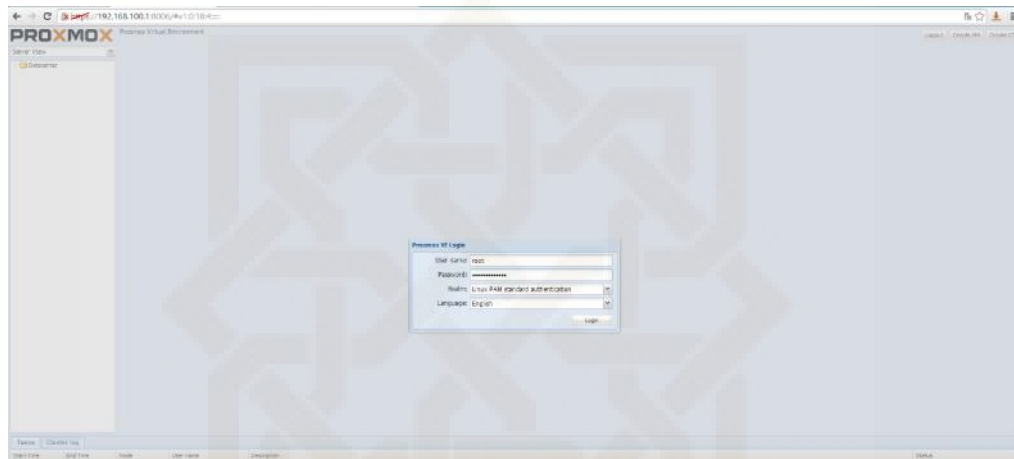
#          S T A R T U P   E N A B L E D

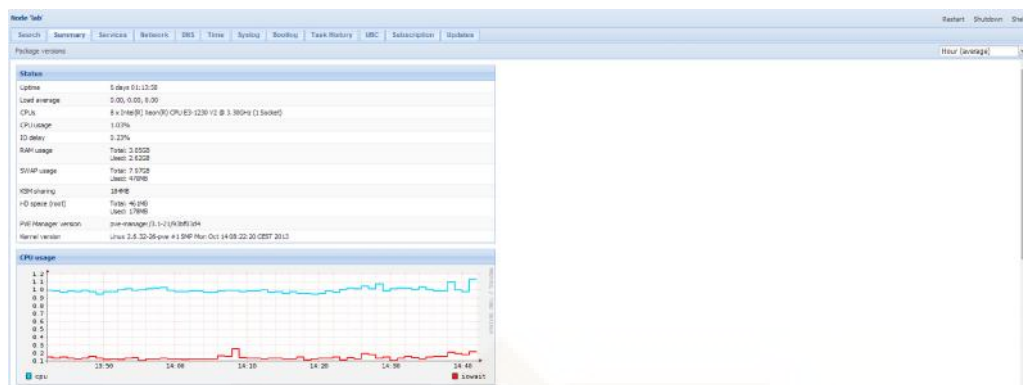
#####
#####
```

STARTUP_ENABLED=Yes

Nama File : Web Interfaces Proxmox

Lokasi : Proxmox VE





LAMPIRAN D

KODE SUMBER (SOURCE CODE) SHOREWALL WEB INTERFACE

Nama File : Home.php

Lokasi : Proxmox VE

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <meta name="description" content="">
    <meta name="author" content="">
    <title>Home</title>
    <link rel="shortcut icon" href="../../assets/ico/favicon.ico">
    <!-- Bootstrap core CSS -->
    <link href="assets/css/bootstrap.min.css" rel="stylesheet">
    <link href="assets/css/font-awesome.min.css" rel="stylesheet">
    <!-- Custom styles for this template -->
    <link href="assets/css/arcustom.css" rel="stylesheet">
  </head>
  <body>
    <!-- navbar / top menu -->
    <div class="navbar navbar-inverse navbar-default navbar-fixed-top" role="navigation">
      <div class="container">
```

```

<div class="navbar-header">

    <button type="button" class="navbar-toggle"
data-toggle="collapse" data-target=".navbar-collapse">

        <span class="sr-only">Toggle
navigation</span>

        <span class="icon-bar"></span>

        <span class="icon-bar"></span>

        <span class="icon-bar"></span>

    </button>

    <a class="navbar-brand" href="#"><img src =
assets/img/logoo.png> Shorewall Web Interface</a>

</div>

<div class="navbar-collapse collapse">

    <ul class="nav navbar-nav">

        <li class="active"><a href="home.php">
<i class ="fa fa-home"></i> Home</a></li>

        <li><a href="rules.php"> <i class ="fa fa-
fire"></i> Rules</a></li>

        <li><a href="policy.php"> <i class ="fa
fa-globe"></i> Policy</a></li>

        <li><a href="zones.php"> <i class ="fa
fa-cog"></i> Zones</a></li>

        <li><a href="interfaces.php"> <i class
="fa fa-bolt"></i> Interfaces</a></li>

    </ul>

    <ul class="nav navbar-nav navbar-right">

        <li class="dropdown">

            <a href="#" class="dropdown-
toggle" data-toggle="dropdown"><i class="fa fa-user"></i> User <b
class="caret"></b></a>

            <ul class="dropdown-menu">

                <li><a

```



```

href="akun.html"><i class="fa fa-smile-o"></i> Info Akun</a></li>

<li><a href="logout"><i
class="fa fa-sign-out"></i> Logout</a></li>

</ul>

</li>

</ul>

</div>

</div>

</div>

<!-- end of menu -->

<!-- header -->

<div class="jumbotron">

  <div class="container text-center">

    <h1>Hello, user!</h1>

  </div>

</div>

<div style="text-align:center;"><img src =
assets/img/logo.png align = "center"></div>

<!-- end of content body -->

<!-- footer -->

<div id="footer">

  <div class="container">

    <p class="text-center">&copy; created by Arya
Ervan Leoresta & middot; 2014 </a></p>

  </div>

</div>

<!-- end of footer -->

```

```

        <script src="assets/js/jquery.min.js"></script>

        <script src="assets/js/bootstrap.min.js"></script>

    </body>

</html>

```

Nama File : interfaces.php

Lokasi : Proxmox VE

```

<?php
$txt_file = file_get_contents('/etc/shorewall/interfaces');
$rows = explode("\n", $txt_file);
?>

<!DOCTYPE html>

<html lang="en">
    <head>

        <meta charset="utf-8">

        <meta http-equiv="X-UA-Compatible" content="IE=edge">

        <meta name="viewport" content="width=device-width, initial-scale=1">

        <meta name="description" content="">

        <meta name="author" content="">

        <link rel="shortcut icon" href="../../assets/ico/favicon.ico">

        <!-- Bootstrap core CSS -->

        <link href="assets/css/bootstrap.min.css" rel="stylesheet">

        <link href="assets/css/font-awesome.min.css" rel="stylesheet">

        <!-- Custom styles for this template -->

        <style text="text/css">

            body {

```

```

        min-height: 2000px;

        padding-top: 70px;

        padding-bottom: 40px;

    }

</style>

</head>

<body>

    <!-- navbar / top menu -->

    <div class="navbar navbar-inverse navbar-default navbar-fixed-top" role="navigation">

        <div class="container">

            <div class="navbar-header">

                <button type="button" class="navbar-toggle"
data-toggle="collapse" data-target=".navbar-collapse">

navigation</span>                <span class="sr-only">Toggle

                <span class="icon-bar"></span>

                <span class="icon-bar"></span>

                <span class="icon-bar"></span>

            </button>

            <a class="navbar-brand" href="#"><img src =
assets/img/logoo.png> Shorewall Web Interface</a>

        </div>

        <div class="navbar-collapse collapse">

            <ul class="nav navbar-nav">

                <li><a href="home.php"> <i class ="fa
fa-home"></i> Home</a></li>

                <li><a href="rules.php"> <i class ="fa fa-
fire"></i> Rules</a></li>

                <li><a href="policy.php"> <i class ="fa
fa-globe"></i> Policy</a></li>

```

```

fa-cog"></i> Zones</a></li>

        <li class="active"><a
href="interfaces.php"> <i class ="fa fa-bolt"></i> Interfaces</a></li>

        <li class="dropdown">

            <a href="#" class="dropdown-
toggle fa fa-legal" data-toggle="dropdown"> Actions <b class="caret"></b></a>

            <ul class="dropdown-menu">

                <li><a
href="interfaces_export.php">Edit</a></li>

                <li class="divider"></li>

                <li><a
href="policy_restart.php">Restart</a></li>

            </ul>

        </li>

    </ul>

    <ul class="nav navbar-nav navbar-right">

        <li class="dropdown">

            <a href="#" class="dropdown-
toggle" data-toggle="dropdown"><i class="fa fa-user"></i> User <b
class="caret"></b></a>

            <ul class="dropdown-menu">

                <li><a
href="akun.html"><i class="fa fa-smile-o"></i> Info Akun</a></li>

                <li><a href="logout"><i
class="fa fa-sign-out"></i> Logout</a></li>

            </ul>

        </li>

    </ul>

</div>

</div>

```

```

</div>

<!-- end of menu -->

<!-- content body -->

<div class="container">

    <ol class="breadcrumb">

        <li><a href="home.html"><i class="fa fa-home"></i>
Home</a></li>

        <li class="active">Interfaces</li>

    </ol>

    <div class="row">

        <!-- content-->

        <div class="col-md-12">

            <table class="table table-hover table-bordered">

                <!-- <thead> -->

                <tr><td>ZONE</td><td>INTERFACE</td><td>BROADCAST</td><td>OPTIONS</t
d></tr>

                <!-- </thead> -->

                <?php
foreach($rows as $row => $data) {

    //get row data

    if(($rows[$row] != null) &&
((substr($data,0,1) != "#"))) {

        $row_data[] = preg_split('/\s+/', $data);

    }else{

        continue;

    };

    // echo "<tr>";

```

```

// foreach ($row_data as $row_data_size =>
$row_data_string) {

//      if ($row_data_string!="") {
//      echo "<td>" .

$row_data[$row_data_size] . '</td>';

//      }else{
//      echo "<td></td>";
//      }

//}
// echo "</tr>";

}
?>

<?php

//print_r($row_data);

foreach ($row_data as $k => $v)

{

echo "<tr>";

for ($i=0; $i <

sizeof($row_data[0]); $i++) {

if

(array_key_exists($i, $row_data[$k])) {

echo "<td>" .

$row_data[$k][$i] . "</td>";

}else{

echo "<td> -

</td>";

}

}

```

```

    }
    echo "</tr>";
}

```

```

?>
<!--
<tbody>
    <tr>
        <td></td>
    </tr>
</tbody> -->
</table>
</div>
</div>
<!-- end of content body -->
<!-- footer -->
<div id="footer">
    <div class="container">
        <p class="text-center">&copy; created by Arya Ervan
Leoresta &middot; 2014 </a></p>
    </div>
</div>
<!-- end of footer -->
<script src="assets/js/jquery.min.js"></script>
<script src="assets/js/bootstrap.min.js"></script>
</body>
</html>

```


Nama File : policy.php

Lokasi : Proxmox VE

```
<?php
$txt_file = file_get_contents('/etc/shorewall/policy');
$rows = explode("\n", $txt_file);
?>
<!DOCTYPE html>
<html lang="en">
    <head>
        <meta charset="utf-8">
        <meta http-equiv="X-UA-Compatible" content="IE=edge">
        <meta name="viewport" content="width=device-width, initial-scale=1">
        <meta name="description" content="">
        <meta name="author" content="">
        <link rel="shortcut icon" href="../../assets/ico/favicon.ico">
        <!-- Bootstrap core CSS -->
        <link href="assets/css/bootstrap.min.css" rel="stylesheet">
        <link href="assets/css/font-awesome.min.css" rel="stylesheet">
        <!-- Custom styles for this template -->
        <style text="text/css">
            body {
                min-height: 2000px;
                padding-top: 70px;
                padding-bottom: 40px;
            }
```

```

</style>

</head>

<body>

<!-- navbar / top menu -->

<div class="navbar navbar-inverse navbar-default navbar-fixed-top"
role="navigation">

    <div class="container">

        <div class="navbar-header">

            <button type="button" class="navbar-toggle"
data-toggle="collapse" data-target=".navbar-collapse">

                <span class="sr-only">Toggle
navigation</span>

                <span class="icon-bar"></span>

                <span class="icon-bar"></span>

                <span class="icon-bar"></span>

            </button>

            <a class="navbar-brand" href="#"><img src =
assets/img/logoo.png> Shorewall Web Interface</a>

        </div>

        <div class="navbar-collapse collapse">

            <ul class="nav navbar-nav">

                <li><a href="home.php"> <i class ="fa
fa-home"></i> Home</a></li>

                <li><a href="rules.php"> <i class ="fa fa-
fire"></i> Rules</a></li>

                <li class="active"><a href="policy.php">

                    <i class ="fa fa-globe"></i> Policy</a></li>

                <li><a href="zones.php"> <i class ="fa
fa-cog"></i> Zones</a></li>

                <li><a href="interfaces.php"> <i class
="fa fa-bolt"></i> Interfaces</a></li>

```

```

<li class="dropdown">
    <a href="#" class="dropdown-
toggle fa fa-legal" data-toggle="dropdown"> Actions <b class="caret"></b></a>
    <ul class="dropdown-menu">
        <li><a
href="policy_export.php">Edit</a></li>
        <li class="divider"></li>
        <li><a
href="policy_restart.php">Restart</a></li>
    </ul>
</li>
</ul>
<ul class="nav navbar-nav navbar-right">
    <li class="dropdown">
        <a href="#" class="dropdown-
toggle" data-toggle="dropdown"><i class="fa fa-user"></i> User <b
class="caret"></b></a>
        <ul class="dropdown-menu">
            <li><a
href="akun.html"><i class="fa fa-smile-o"></i> Info Akun</a></li>
            <li><a href="logout"><i
class="fa fa-sign-out"></i> Logout</a></li>
        </ul>
    </li>
</ul>
</div>
</div>
<!-- end of menu -->
<!-- content body -->

```

```

<div class="container">

    <ol class="breadcrumb">

        <li><a href="home.html"><i class="fa fa-home"></i>
Home</a></li>

        <li class="active">Policy</li>

    </ol>

    <div class="row">

        <!-- content-->

        <div class="col-md-12">

            <table class="table table-hover table-bordered">

                <tr><td>SOURCE</td><td>DEST</td><td>POLICY</td><td>LOG</td><td>LEVEL<
/td><td>LIMIT:BURST</td></tr>

                <?php
                foreach($rows as $row => $data) {
                    //get row data
                    if(($rows[$row] != null) &&
((substr($data,0,1) != "#"))) {

                        $row_data[] = preg_split('/\s+/', $data);
                    }else{
                        continue;
                    };

                    // echo "<tr>";

                    // foreach ($row_data as $row_data_size =>
$row_data_string) {

                        //          if ($row_data_string!="") {
                        //          echo "<td>" .

$row_data[$row_data_size] . '</td>';

                        //      }else{

```

```

//          echo "<td></td>";
//      }
//}
// echo "</tr>";

}
?>

<?php

//print_r($row_data);

foreach ($row_data as $k => $v)
{
    echo "<tr>";
    for ($i=0; $i < 6; $i++) {
        if
(array_key_exists($i, $row_data[$k])) {
            echo "<td>" .
$row_data[$k][$i] . "</td>";
        }else{
            echo "<td> -
</td>";
        }
    }
    echo "</tr>";
}

?>

<!--

```

```

        <tbody>
            <tr>
                <td></td>
            </tr>
        </tbody> -->
    </table>
</div>
</div>
<!-- end of content body -->
<!-- footer -->
<div id="footer">
    <div class="container">
        <p class="text-center">&copy; created by Arya Ervan
        Leoresta & middot; 2014 </a></p>
    </div>
</div>
<!-- end of footer -->
<script src="assets/js/jquery.min.js"></script>
<script src="assets/js/bootstrap.min.js"></script>
</body>
</html>

```

Nama File : rules.php

Lokasi : Proxmox VE

```
<?php
```

```
$txt_file = file_get_contents('/etc/shorewall/rules');
```

```
$rows = explode("\n", $txt_file);
```

```

?>

<!DOCTYPE html>

<html lang="en">

  <head>

    <meta charset="utf-8">

    <meta http-equiv="X-UA-Compatible" content="IE=edge">

    <meta name="viewport" content="width=device-width, initial-scale=1">

    <meta name="description" content="">

    <meta name="author" content="">

    <link rel="shortcut icon" href="../../assets/ico/favicon.ico">

    <!-- Bootstrap core CSS -->

    <link href="assets/css/bootstrap.min.css" rel="stylesheet">

    <link href="assets/css/font-awesome.min.css" rel="stylesheet">

    <!-- Custom styles for this template -->

    <style text="text/css">

      body {

        min-height: 2000px;

        padding-top: 70px;

        padding-bottom: 40px;

      }

    </style>

  </head>

  <body>

    <!-- navbar / top menu -->

    <div class="navbar navbar-inverse navbar-default navbar-fixed-top"
role="navigation">

      <div class="container">

        <div class="navbar-header">

```



```

        <button type="button" class="navbar-toggle"
data-toggle="collapse" data-target=".navbar-collapse">

        <span class="sr-only">Toggle

navigation</span>

        <span class="icon-bar"></span>

        <span class="icon-bar"></span>

        <span class="icon-bar"></span>

    </button>

    <a class="navbar-brand" href="#"><img src =
assets/img/logoo.png> Shorewall Web Interface</a>

</div>

<div class="navbar-collapse collapse">

    <ul class="nav navbar-nav">

        <li><a href="home.php"> <i class ="fa
fa-home"></i> Home</a></li>

        <li class="active"><a href="rules.php">
<i class ="fa fa-fire"></i> Rules</a></li>

        <li><a href="policy.php"> <i class ="fa
fa-globe"></i> Policy</a></li>

        <li><a href="zones.php"> <i class ="fa
fa-cog"></i> Zones</a></li>

        <li><a href="interfaces.php"> <i class
="fa fa-bolt"></i> Interfaces</a></li>

        <li class="dropdown">

            <a href="#" class="dropdown-
toggle fa fa-legal" data-toggle="dropdown"> Actions <b class="caret"></b></a>

            <ul class="dropdown-menu">

                <li><a
href="rules_export.php">Edit</a></li>

                <li class="divider"></li>

                <li><a
href="policy_restart.php">Restart</a></li>

```

```

        </ul>
    </li>
</ul>
<ul class="nav navbar-nav navbar-right">
    <li class="dropdown">
        <a href="#" class="dropdown-
toggle" data-toggle="dropdown"><i class="fa fa-user"></i> User <b
class="caret"></b></a>
        <ul class="dropdown-menu">
            <li><a
href="akun.html"><i class="fa fa-smile-o"></i> Info Akun</a></li>
            <li><a href="logout"><i
class="fa fa-sign-out"></i> Logout</a></li>
        </ul>
    </li>
</ul>
</div>
</div>
<!-- end of menu -->
<!-- content body -->
<div class="container">
    <ol class="breadcrumb">
        <li><a href="home.html"><i class="fa fa-home"></i>
Home</a></li>
        <li class="active">Rules</li>
    </ol>
    <div class="row">
        <!-- content-->
        <div class="col-md-12">

```

```

<table class="table table-hover table-bordered">

    <tr><td>ACTION</td><td>SOURCE</td><td>DEST</td><td>PROTO</td><td>DE
ST</td><td>SOURCE</td>

    <td>ORIGINAL</td><td>RATE</td><td>USER</td><td>MARK</td><td>CONLIMI
T</td><td>TIME</td><td>HEADERS</td>

    <td>SWITCH</td></tr>

    <?php
    foreach($rows as $row => $data) {
        //get row data
        if(($rows[$row] != null) &&
((substr($data,0,1) != "#"))) {

            $row_data[] = preg_split('/\s+/', $data);
        }else{
            continue;
        };

        // echo "<tr>";

        // foreach ($row_data as $row_data_size =>
$row_data_string) {

            //         if ($row_data_string!="") {
            //         echo "<td>" .

$row_data[$row_data_size] . '</td>';

            //     }else{
            //         echo "<td></td>";
            //     }

            // }

            // echo "</tr>";

```

```

    }
    ?>

<?php

    //print_r($row_data);

    foreach ($row_data as $k => $v)
    {
        echo "<tr>";
        for ($i=0; $i < 14; $i++) {
            if
(array_key_exists($i, $row_data[$k])) {
                echo "<td>" .
$row_data[$k][$i] . "</td>";
            }else{
                echo "<td> -
</td>";
            }
        }
        echo "</tr>";
    }
    ?>

<!--
<tbody>
    <tr>
        <td></td>
    </tr>
</tbody> -->

```

```

        </table>

    </div>

</div>

<!-- end of content body -->

<!-- footer -->

<div id="footer">

    <div class="container">

        <p class="text-center">&copy; created by Arya Ervan
        Leoresta & middot; 2014 </a></p>

    </div>

</div>

<!-- end of footer -->

<script src="assets/js/jquery.min.js"></script>

<script src="assets/js/bootstrap.min.js"></script>

</body>

</html>

```

Nama File : zones.php

Lokasi : Proxmox VE

```

<?php
$txt_file = file_get_contents('/etc/shorewall/zones');
$rows = explode("\n", $txt_file);

?>

<!DOCTYPE html>

<html lang="en">

    <head>

        <meta charset="utf-8">

```

```

<meta http-equiv="X-UA-Compatible" content="IE=edge">

<meta name="viewport" content="width=device-width, initial-scale=1">

<meta name="description" content="">

<meta name="author" content="">

<link rel="shortcut icon" href="../../assets/ico/favicon.ico">

<!-- Bootstrap core CSS -->

<link href="assets/css/bootstrap.min.css" rel="stylesheet">

<link href="assets/css/font-awesome.min.css" rel="stylesheet">

<!-- Custom styles for this template -->

<style text="text/css">

    body {

        min-height: 2000px;

        padding-top: 70px;

        padding-bottom: 40px;

    }

</style>
</head>
<body>

    <!-- navbar / top menu -->

    <div class="navbar navbar-inverse navbar-default navbar-fixed-top"
role="navigation">

        <div class="container">

            <div class="navbar-header">

                <button type="button" class="navbar-toggle"
data-toggle="collapse" data-target=".navbar-collapse">

                    <span class="sr-only">Toggle
navigation</span>

                    <span class="icon-bar"></span>

                    <span class="icon-bar"></span>

```

```

        <span class="icon-bar"></span>

    </button>

    <a class="navbar-brand" href="#"><img src =
assets/img/logoo.png> Shorewall Web Interface</a>

</div>

<div class="navbar-collapse collapse">

    <ul class="nav navbar-nav">

        <li><a href="home.php"> <i class ="fa
fa-home"></i> Home</a></li>

        <li><a href="rules.php"> <i class ="fa fa-
fire"></i> Rules</a></li>

        <li><a href="policy.php"> <i class ="fa
fa-globe"></i> Policy</a></li>

        <li class="active"><a href="zones.php">
<i class ="fa fa-cog"></i> Zones</a></li>

        <li><a href="interfaces.php"> <i class
="fa fa-bolt"></i> Interfaces</a></li>

        <li class="dropdown">

            <a href="#" class="dropdown-
toggle fa fa-legal" data-toggle="dropdown"> Actions <b class="caret"></b></a>

            <ul class="dropdown-menu">

                <li><a
href="zones_export.php">Edit</a></li>

                <li class="divider"></li>

                <li><a
href="policy_restart.php">Restart</a></li>

            </ul>

        </li>

    </ul>

    <ul class="nav navbar-nav navbar-right">

        <li class="dropdown">

```

```

<a href="#" class="dropdown-
toggle" data-toggle="dropdown"><i class="fa fa-user"></i> User <b
class="caret"></b></a>

<ul class="dropdown-menu">

    <li><a
href="akun.html"><i class="fa fa-smile-o"></i> Info Akun</a></li>

    <li><a href="logout"><i
class="fa fa-sign-out"></i> Logout</a></li>

</ul>

</li>

</ul>

</div>

</div>

</div>

<!-- end of menu -->

<!-- content body -->

<div class="container">

    <ol class="breadcrumb">

        <li><a href="home.html"><i class="fa fa-home"></i>
Home</a></li>

        <li class="active">Zones</li>

    </ol>

    <div class="row">

        <!-- content-->

        <div class="col-md-12">

            <table class="table table-hover table-bordered">

                <!-- <thead> -->

                <tr><td>ZONE</td><td>TYPE</td><td>OPTIONS</td><td>IN
OPTIONS</td><td>OUT OPTIONS</td></tr>

```



```

<!-- </thead> -->

<?php
foreach($rows as $row => $data) {
    //get row data
    if(($rows[$row] != null) &&
    ((substr($data,0,1) != "#"))) {

        $row_data[] = preg_split('/\s+/', $data);
    }else{
        continue;
    };

    // echo "<tr>";
    // foreach ($row_data as $row_data_size =>
$row_data_string) {
        //      if ($row_data_string!="") {
        //          echo "<td>" .
$row_data[$row_data_size] . '</td>';

        //      }else{
        //          echo "<td></td>";
        //      }
    // }
    // echo "</tr>";

}

?>

<?php

```

```

//print_r($row_data);
        foreach ($row_data as $k => $v)
{
        echo "<tr>";
        for ($i=0; $i < 5; $i++) {
                if
(array_key_exists($i, $row_data[$k])) {
                        echo "<td>" .
$row_data[$k][$i] . "</td>";
                }else{
                        echo "<td> -
</td>";
                }
        }
        echo "</tr>";
}

?>
<!--
<tbody>
        <tr>
                <td></td>
        </tr>
</tbody> -->
</table>
</div>
</div>
<!-- end of content body -->
<!-- footer -->

```

```
<div id="footer">
  <div class="container">
    <p class="text-center">&copy; created by Arya Ervan
    Leoresta &middot; 2014 </a></p>
  </div>
</div>
<!-- end of footer -->
<script src="assets/js/jquery.min.js"></script>
<script src="assets/js/bootstrap.min.js"></script>
</body>
</html>
```

LAMPIRAN E

LEMBAR ANGKET PENGUJIAN FUNGSIONAL DAN STRESS TEST IMPLEMENTASI HONEYPOT SEBAGAI PENDETEKSI MALWARE PADA LAYANAN CLOUD COMPUTING

LEMBAR ANGKET STRESS TEST
IMPLEMENTASI HONEYPOT SEBAGAI PENDETEKSI MALWARE
PADA LAYANAN CLOUD COMPUTING

Nama : Feni Sehyawan

Pekerjaan : Mahasiswa Uin 2010

Berilah tanda *Check* (v) pada pilihan yang menurut anda paling sesuai menggambarkan fungsi sistem. SS untuk **sangat setuju**, S untuk **setuju**, KS untuk **kurang setuju** dan TS untuk **tidak setuju** untuk pernyataan berikut :

No	Aspek	Pilihan			
		SS	S	KS	TS
1	Apakah hasil pengujian stress test sudah sesuai dengan harapan?	✓			
2	Apakah konsep implementasi Honeypot dalam <i>Cloud Computing</i> dapat berjalan sesuai harapan?	✓			

Saran :

Yogyakarta, 18 Agustus 2014

Penguji

Feni
Feni Sehyawan

LEMBAR ANGKET PENGUJIAN FUNGSIONAL
IMPLEMENTASI HONEYPOT SEBAGAI PENDETEKSI MALWARE
PADA LAYANAN CLOUD COMPUTING

Nama : Budi Lestari, M.Pd

Pekerjaan : Guru TIK SMK 2 Pengasih

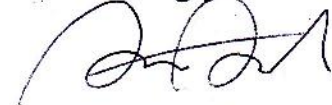
Berilah tanda *Check* (v) pada pilihan yang menurut anda paling sesuai untuk pernyataan berikut :

No	Pengujian	Pilihan	
		Ya	Tidak
1	Apakah skenario pengujian selama 7 hari, sistem Honeypot dapat menangkap <i>malware</i> ?	✓	
2	Apakah skenario pengujian selama 7 hari, sistem Honeypot dapat mendeteksi dan mengetahui <i>port</i> yang sering diserang?	✓	
3	Apakah skenario pengujian selama 7 hari, sistem Honeypot dapat mengetahui negara mana yang sering melakukan penyerangan?	✓	
4	Apakah skenario pengujian selama 7 hari, sistem Honeypot dapat mengetahui waktu penyerangan?	✓	

Saran :

Yogyakarta, 19 Agustus 2014

Penguji



Budi Lestari, M.Pd

LEMBAR ANGKET STRESS TEST
IMPLEMENTASI HONEYPOT SEBAGAI PENDETEKSI MALWARE
PADA LAYANAN CLOUD COMPUTING

Nama : Budi Lestari, M.Pd
Pekerjaan : Guru TKJ SMK 2 Pengasih

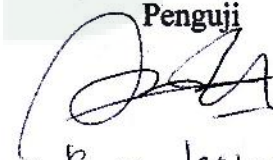
Berilah tanda *Check* (v) pada pilihan yang menurut anda paling sesuai menggambarkan fungsi sistem. SS untuk **sangat setuju**, S untuk **setuju**, KS untuk **kurang setuju** dan TS untuk **tidak setuju** untuk pernyataan berikut :

No	Aspek	Pilihan			
		SS	S	KS	TS
1	Apakah hasil pengujian stress test sudah sesuai dengan harapan?		✓		
2	Apakah konsep implementasi Honeypot dalam <i>Cloud Computing</i> dapat berjalan sesuai harapan?	✓			

Saran :

Yogyakarta , 19 Agustus 2014

Penguji



Budi Lestari, M.Pd

LEMBAR ANGKET PENGUJIAN FUNGSIONAL
IMPLEMENTASI HONEYPOT SEBAGAI PENDETEKSI MALWARE
PADA LAYANAN CLOUD COMPUTING

Nama : Setya Budi S. Kom

Pekerjaan : Alumni Mahasiswa UIN 2007

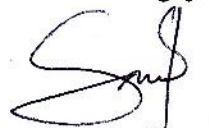
Berilah tanda *Check* (v) pada pilihan yang menurut anda paling sesuai untuk pernyataan berikut :

No	Pengujian	Pilihan	
		Ya	Tidak
1	Apakah skenario pengujian selama 7 hari, sistem Honeypot dapat menangkap <i>malware</i> ?	✓	
2	Apakah skenario pengujian selama 7 hari, sistem Honeypot dapat mendeteksi dan mengetahui <i>port</i> yang sering diserang?	✓	
3	Apakah skenario pengujian selama 7 hari, sistem Honeypot dapat mengetahui negara mana yang sering melakukan penyerangan?	✓	
4	Apakah skenario pengujian selama 7 hari, sistem Honeypot dapat mengetahui waktu penyerangan?	✓	

Saran :

Yogyakarta, 18 Agustus 2014

Penguji


Setya Budi

LEMBAR ANGKET STRESS TEST
IMPLEMENTASI HONEYPOT SEBAGAI PENDETEKSI MALWARE
PADA LAYANAN CLOUD COMPUTING

Nama : Setya Budi, S. Kom

Pekerjaan : Alumni Mahasiswa UIN 2007

Berilah tanda *Check* (v) pada pilihan yang menurut anda paling sesuai menggambarkan fungsi sistem. SS untuk **sangat setuju**, S untuk **setuju**, KS untuk **kurang setuju** dan TS untuk **tidak setuju** untuk pernyataan berikut :

No	Aspek	Pilihan			
		SS	S	KS	TS
1	Apakah hasil pengujian stress test sudah sesuai dengan harapan?	✓			
2	Apakah konsep implementasi Honeypot dalam <i>Cloud Computing</i> dapat berjalan sesuai harapan?		✓		

Saran :

Yogyakarta, 18 Agustus 2019

Penguji



Setya Budi

LEMBAR ANGKET PENGUJIAN FUNGSIONAL
IMPLEMENTASI HONEYPOT SEBAGAI PENDETEKSI MALWARE
PADA LAYANAN CLOUD COMPUTING

Nama : Ranyat Ys

Pekerjaan : Mahasiswa

Berilah tanda *Check* (v) pada pilihan yang menurut anda paling sesuai untuk pernyataan berikut :

No	Pengujian	Pilihan	
		Ya	Tidak
1	Apakah skenario pengujian selama 7 hari, sistem Honeypot dapat menangkap <i>malware</i> ?	✓	
2	Apakah skenario pengujian selama 7 hari, sistem Honeypot dapat mendeteksi dan mengetahui <i>port</i> yang sering diserang?	✓	
3	Apakah skenario pengujian selama 7 hari, sistem Honeypot dapat mengetahui negara mana yang sering melakukan penyerangan?	✓	
4	Apakah skenario pengujian selama 7 hari, sistem Honeypot dapat mengetahui waktu penyerangan?	✓	

Saran :

Yogyakarta, 12 Agustus 2014..

Penguji



RANYAT YS

LEMBAR ANGKET STRESS TEST
IMPLEMENTASI HONEYPOT SEBAGAI PENDETEKSI MALWARE
PADA LAYANAN CLOUD COMPUTING

Nama : Rasyid Y

Pekerjaan : Mahasiswa

Berilah tanda *Check* (v) pada pilihan yang menurut anda paling sesuai menggambarkan fungsi sistem. SS untuk **sangat setuju**, S untuk **setuju**, KS untuk **kurang setuju** dan TS untuk **tidak setuju** untuk pernyataan berikut :

No	Aspek	Pilihan			
		SS	S	KS	TS
1	Apakah hasil pengujian stress test sudah sesuai dengan harapan?		✓		
2	Apakah konsep implementasi Honeypot dalam <i>Cloud Computing</i> dapat berjalan sesuai harapan?	✓			

Saran :

Yogyakarta, 12 Agustus 2014

Penguji

Rasyid

Kory10

LEMBAR ANGKET PENGUJIAN FUNGSIONAL
IMPLEMENTASI HONEYPOT SEBAGAI PENDETEKSI MALWARE
PADA LAYANAN CLOUD COMPUTING

Nama : Raditya Habsero A.Md

Pekerjaan : Mahasiswa UGM

Berilah tanda *Check* (v) pada pilihan yang menurut anda paling sesuai untuk pernyataan berikut :

No	Pengujian	Pilihan	
		Ya	Tidak
1	Apakah skenario pengujian selama 7 hari, sistem Honeypot dapat menangkap <i>malware</i> ?	✓	
2	Apakah skenario pengujian selama 7 hari, sistem Honeypot dapat mendeteksi dan mengetahui <i>port</i> yang sering diserang?	✓	
3	Apakah skenario pengujian selama 7 hari, sistem Honeypot dapat mengetahui negara mana yang sering melakukan penyerangan?	✓	
4	Apakah skenario pengujian selama 7 hari, sistem Honeypot dapat mengetahui waktu penyerangan?	✓	

Saran :

Yogyakarta, 19 Agustus 2019

Penguji

Raditya Habsero, A.Md

LEMBAR ANGKET STRESS TEST
IMPLEMENTASI HONEYPOT SEBAGAI PENDETEKSI MALWARE
PADA LAYANAN CLOUD COMPUTING

Nama : Fadhiya habsono, A.Md
Pekerjaan : Mahasiswa UGM

Berilah tanda *Check* (v) pada pilihan yang menurut anda paling sesuai menggambarkan fungsi sistem. SS untuk **sangat setuju**, S untuk **setuju**, KS untuk **kurang setuju** dan TS untuk **tidak setuju** untuk pernyataan berikut :

No	Aspek	Pilihan			
		SS	S	KS	TS
1	Apakah hasil pengujian stress test sudah sesuai dengan harapan?	✓			
2	Apakah konsep implementasi Honeypot dalam <i>Cloud Computing</i> dapat berjalan sesuai harapan?	✓			

Saran :

Yogyakarta , 6 Agustus 2014

Penguji



Fadhiya Habsono, A. Md

LEMBAR ANGKET PENGUJIAN FUNGSIONAL
IMPLEMENTASI HONEYPOT SEBAGAI PENDETEKSI MALWARE
PADA LAYANAN CLOUD COMPUTING

Nama : Indrag Woro Widada

Pekerjaan : Mahasiswa

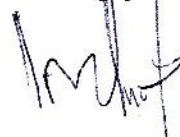
Berilah tanda *Check* (v) pada pilihan yang menurut anda paling sesuai untuk pernyataan berikut :

No	Pengujian	Pilihan	
		Ya	Tidak
1	Apakah skenario pengujian selama 7 hari, sistem Honeypot dapat menangkap <i>malware</i> ?	✓	
2	Apakah skenario pengujian selama 7 hari, sistem Honeypot dapat mendeteksi dan mengetahui <i>port</i> yang sering diserang?	✓	
3	Apakah skenario pengujian selama 7 hari, sistem Honeypot dapat mengetahui negara mana yang sering melakukan penyerangan?	✓	
4	Apakah skenario pengujian selama 7 hari, sistem Honeypot dapat mengetahui waktu penyerangan?	✓	

Saran :

Yogyakarta, 12 Agustus 2019

Penguji



LEMBAR ANGKET STRESS TEST
IMPLEMENTASI HONEYPOT SEBAGAI PENDETEKSI MALWARE
PADA LAYANAN CLOUD COMPUTING

Nama : Indray Waro Wido

Pekerjaan : Mahasiswa

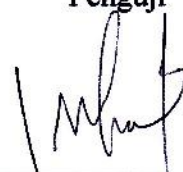
Berilah tanda *Check* (v) pada pilihan yang menurut anda paling sesuai menggambarkan fungsi sistem. SS untuk **sangat setuju**, S untuk **setuju**, KS untuk **kurang setuju** dan TS untuk **tidak setuju** untuk pernyataan berikut :

No	Aspek	Pilihan			
		SS	S	KS	TS
1	Apakah hasil pengujian stress test sudah sesuai dengan harapan?	✓			
2	Apakah konsep implementasi Honeypot dalam <i>Cloud Computing</i> dapat berjalan sesuai harapan?	✓			

Saran :

Yogyakarta, 12 Agustus 2019

Penguji



LEMBAR ANGKET PENGUJIAN FUNGSIONAL
IMPLEMENTASI HONEYPOT SEBAGAI PENDETEKSI MALWARE
PADA LAYANAN CLOUD COMPUTING

Nama : M. A. Lohief

Pekerjaan : Mahasiswa UIN 2009

Berilah tanda *Check* (v) pada pilihan yang menurut anda paling sesuai untuk pernyataan berikut :

No	Pengujian	Pilihan	
		Ya	Tidak
1	Apakah skenario pengujian selama 7 hari, sistem Honeypot dapat menangkap <i>malware</i> ?	✓	
2	Apakah skenario pengujian selama 7 hari, sistem Honeypot dapat mendeteksi dan mengetahui <i>port</i> yang sering diserang?	✓	
3	Apakah skenario pengujian selama 7 hari, sistem Honeypot dapat mengetahui negara mana yang sering melakukan penyerangan?	✓	
4	Apakah skenario pengujian selama 7 hari, sistem Honeypot dapat mengetahui waktu penyerangan?	✓	

Saran :

Yogyakarta, 18 Agustus 2019

Penguji

Luf
M. A. Lohief

LEMBAR ANGKET STRESS TEST
IMPLEMENTASI HONEYPOT SEBAGAI PENDETEKSI MALWARE
PADA LAYANAN CLOUD COMPUTING

Nama : M. A. Lahief

Pekerjaan : Mahasiswa UIN 2009

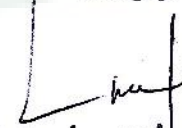
Berilah tanda *Check* (v) pada pilihan yang menurut anda paling sesuai menggambarkan fungsi sistem. SS untuk **sangat setuju**, S untuk **setuju**, KS untuk **kurang setuju** dan TS untuk **tidak setuju** untuk pernyataan berikut :

No	Aspek	Pilihan			
		SS	S	KS	TS
1	Apakah hasil pengujian stress test sudah sesuai dengan harapan?	✓			
2	Apakah konsep implementasi Honeypot dalam <i>Cloud Computing</i> dapat berjalan sesuai harapan?		✓		

Saran :

Yogyakarta, 18 Agustus 2019

Penguji


M. A. Lahief

LEMBAR ANGKET PENGUJIAN FUNGSIONAL
IMPLEMENTASI HONEYPOT SEBAGAI PENDETEKSI MALWARE
PADA LAYANAN CLOUD COMPUTING

Nama : Suzud Arifin

Pekerjaan : Mahasiswa Uin 2014

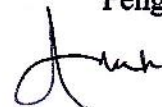
Berilah tanda *Check* (v) pada pilihan yang menurut anda paling sesuai untuk pernyataan berikut :

No	Pengujian	Pilihan	
		Ya	Tidak
1	Apakah skenario pengujian selama 7 hari, sistem Honeypot dapat menangkap <i>malware</i> ?	✓	
2	Apakah skenario pengujian selama 7 hari, sistem Honeypot dapat mendeteksi dan mengetahui <i>port</i> yang sering diserang?	✓	
3	Apakah skenario pengujian selama 7 hari, sistem Honeypot dapat mengetahui negara mana yang sering melakukan penyerangan?	✓	
4	Apakah skenario pengujian selama 7 hari, sistem Honeypot dapat mengetahui waktu penyerangan?	✓	

Saran :

Yogyakarta, 10 Agustus 2014

Penguji



Suzud Arifin

LEMBAR ANGKET STRESS TEST
IMPLEMENTASI HONEYPOT SEBAGAI PENDETEKSI MALWARE
PADA LAYANAN CLOUD COMPUTING

Nama : Suzul Arifin

Pekerjaan : Mahasiswa Uin 2014


Berilah tanda *Check* (v) pada pilihan yang menurut anda paling sesuai menggambarkan fungsi sistem. SS untuk **sangat setuju**, S untuk **setuju**, KS untuk **kurang setuju** dan TS untuk **tidak setuju** untuk pernyataan berikut :

No	Aspek	Pilihan			
		SS	S	KS	TS
1	Apakah hasil pengujian stress test sudah sesuai dengan harapan?	✓			
2	Apakah konsep implementasi Honeypot dalam <i>Cloud Computing</i> dapat berjalan sesuai harapan?	~			

Saran :

Yogyakarta, 18 Agustus 2014

Penguji


Suzul Arifin

LEMBAR ANGKET PENGUJIAN FUNGSIONAL
IMPLEMENTASI HONEYPOT SEBAGAI PENDETEKSI MALWARE
PADA LAYANAN CLOUD COMPUTING

Nama : Irham Sony

Pekerjaan : Mahasiswa UIN 2011

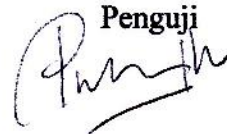
Berilah tanda *Check* (v) pada pilihan yang menurut anda paling sesuai untuk pernyataan berikut :

No	Pengujian	Pilihan	
		Ya	Tidak
1	Apakah skenario pengujian selama 7 hari, sistem Honeypot dapat menangkap <i>malware</i> ?	✓	
2	Apakah skenario pengujian selama 7 hari, sistem Honeypot dapat mendeteksi dan mengetahui <i>port</i> yang sering diserang?	✓	
3	Apakah skenario pengujian selama 7 hari, sistem Honeypot dapat mengetahui negara mana yang sering melakukan penyerangan?	✓	
4	Apakah skenario pengujian selama 7 hari, sistem Honeypot dapat mengetahui waktu penyerangan?	✓	

Saran :

Yogyakarta, 19 Agustus 2014

Penguji



Irham Sony

LEMBAR ANGKET STRESS TEST
IMPLEMENTASI HONEYPOT SEBAGAI PENDETEKSI MALWARE
PADA LAYANAN CLOUD COMPUTING

Nama : Irham Sony

Pekerjaan : Mahasiswa UIN 2011


Berilah tanda *Check* (v) pada pilihan yang menurut anda paling sesuai menggambarkan fungsi sistem. SS untuk **sangat setuju**, S untuk **setuju**, KS untuk **kurang setuju** dan TS untuk **tidak setuju** untuk pernyataan berikut :

No	Aspek	Pilihan			
		SS	S	KS	TS
1	Apakah hasil pengujian stress test sudah sesuai dengan harapan?	✓			
2	Apakah konsep implementasi Honeypot dalam <i>Cloud Computing</i> dapat berjalan sesuai harapan?	✓			

Saran :

Yogyakarta, 19 Agustus 2014

Penguji



Irham Sony

LEMBAR ANGKET PENGUJIAN FUNGSIONAL
IMPLEMENTASI HONEYPOT SEBAGAI PENDETEKSI MALWARE
PADA LAYANAN CLOUD COMPUTING

Nama : Rahmat Nur Faizin

Pekerjaan : Mahasiswa UIN 2010

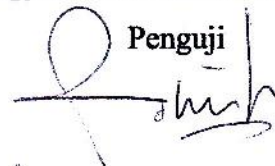
Berilah tanda *Check* (v) pada pilihan yang menurut anda paling sesuai untuk pernyataan berikut :

No	Pengujian	Pilihan	
		Ya	Tidak
1	Apakah skenario pengujian selama 7 hari, sistem Honeypot dapat menangkap <i>malware</i> ?	✓	
2	Apakah skenario pengujian selama 7 hari, sistem Honeypot dapat mendeteksi dan mengetahui <i>port</i> yang sering diserang?	✓	
3	Apakah skenario pengujian selama 7 hari, sistem Honeypot dapat mengetahui negara mana yang sering melakukan penyerangan?	✓	
4	Apakah skenario pengujian selama 7 hari, sistem Honeypot dapat mengetahui waktu penyerangan?	✓	

Saran :

Yogyakarta, 15 Agustus 2014

Penguji



Rahmat Nur Faizin

LEMBAR ANGKET STRESS TEST
IMPLEMENTASI HONEYPOT SEBAGAI PENDETEKSI MALWARE
PADA LAYANAN CLOUD COMPUTING

Nama : Rahmat Nur Faizin

Pekerjaan : Mahasiswa UIN 2010

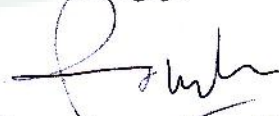
Berilah tanda *Check* (v) pada pilihan yang menurut anda paling sesuai menggambarkan fungsi sistem. SS untuk **sangat setuju**, S untuk **setuju**, KS untuk **kurang setuju** dan TS untuk **tidak setuju** untuk pernyataan berikut :

No	Aspek	Pilihan			
		SS	S	KS	TS
1	Apakah hasil pengujian stress test sudah sesuai dengan harapan?	✓			
2	Apakah konsep implementasi Honeypot dalam <i>Cloud Computing</i> dapat berjalan sesuai harapan?		✓		

Saran :

Yogyakarta , 19 Agustus 2010

Penguji


Rahmat Nur Faizin

LEMBAR ANGKET PENGUJIAN FUNGSIONAL
IMPLEMENTASI HONEYPOT SEBAGAI PENDETEKSI MALWARE
PADA LAYANAN CLOUD COMPUTING

Nama : Feri Seliyawan

Pekerjaan : Mahasiswa Ctin 2010

Berilah tanda *Check* (v) pada pilihan yang menurut anda paling sesuai untuk pernyataan berikut :

No	Pengujian	Pilihan	
		Ya	Tidak
1	Apakah skenario pengujian selama 7 hari, sistem Honeypot dapat menangkap <i>malware</i> ?	✓	
2	Apakah skenario pengujian selama 7 hari, sistem Honeypot dapat mendeteksi dan mengetahui <i>port</i> yang sering diserang?	✓	
3	Apakah skenario pengujian selama 7 hari, sistem Honeypot dapat mengetahui negara mana yang sering melakukan penyerangan?	✓	
4	Apakah skenario pengujian selama 7 hari, sistem Honeypot dapat mengetahui waktu penyerangan?	✓	

Saran :

Yogyakarta, 18 Agustus 2014

Penguji

Feri Seliyawan

DAFTAR RIWAYAT HIDUP



Nama : Arya Ervan Leoresta
Tempat, tanggal lahir : Magelang, 30 Juli 1992
Jenis Kelamin : Laki-laki
Agama : Islam
Alamat Asal : Sekaran RT 02 / 04 Banyurojo, Mertoyudan, Magelang
No. HP : 08994563861
Email : aryaervanleoresta19@gmail.com

Riwayat Pendidikan :

1. SD Kartika XV-1 Panca Arga (1998-2004)
2. SMP Negeri 7 Magelang (2004-2007)
3. SMA Negeri 1 Mertoyudan (2007-2010)
4. S1 Teknik Informatika UIN Sunan Kalijaga Yogyakarta (2010-2014)