# Perancangan *Filtering Firewall* Menggunakan *Iptables* Di Jaringan Pusat Teknologi Informasi Unsrat

Glend Sondakh.[(1)],Meicsy E. I. Najoan, ST., MT[(2)],Arie S. Lumenta, ST, MT.[(3)],
(1)Mahasiswa,(2)Pembimbing1,(3)Pembimbing2,
glend07sondakh@gmail.com[(1)]

JurusanTeknikElektro-FT.UNSRAT,Manado-95115

***Abstrak - Jaringan kampus unsrat yang dikelolah oleh Pusat Teknologi dan Informasi menyediakan layanan pertukaran informasi baik dari dalam (intranet) maupun dari luar (internet). Aliran informasi ini khususnya yang berasal dari luar, sangat rentan terhadap keamanan atau isi informasi yang tidak diinginkan. Untuk mengetahui masalah tersebut, pengelolah jaringan melakukan dengan cara-cara yang sifatnya pengamatan secara langsung dan melakukan buka tutup jika ada hal – hal yang dilihat sebagai suatu ancaman atau ada pengaksesan informasi yang tidak diinginkan. Hal ini sangat membutuhkan waktu dan tenaga, sehingga sangat perlu untuk dicari solusi yang tepat. Struktur dalam suatu jaringan komputer secara standart memiliki struktur 7 lapis model OSI (Open System Interconection) atau 7 layer OSI, dimana lapisan 3 (Network layer) dan lapisan 4 (Transport layer) dapat digunakan untuk melakukan tugas pengamanan dan penyaringan informasi (firewall). Perangkat lunak (software) IPTABLES yang merupakan software bawaan dari sistem operasi linux dapat digunakan sebagai firewall didalam suatu jaringan. Metode yang digunakan untuk melakukan pengamanan dan penyaringan informasi dengan perangkat IPTABLES adalah mekanisme filtering menggunakan Negative list dan Positive list.***

***Kata kunci : 7 OSI LAYER, Firewall, IPTABLES, , Mekanisme filtering Negative list, dan Positive list***

***Abstract - The campus network is managed by the Center UNSRAT and Information Technology provides information exchange services from both inside ( intranet ) and external ( Internet ) . The flow of this information , especially coming from the outside , is very vulnerable to the security or content of unwanted information . To investigate this problem, do the network management by means of direct observations in nature and do the lid if there is - it is seen as a threat or have access to information that is not desired. It is in dire need of time and effort , so it is necessary to look for a proper solution . The structure of a standard computer network having the structure of 7 layers of the OSI model ( Open Systems Interconection ) or 7 OSI layers , where the layer 3 ( network layer ) and Layer 4 ( transport layer ) can be used to perform security tasks and information filtering ( firewall ) . Software IPTABLES which is software built on Linux operating system can be used as a firewall in a network . The method used to do security and filtering information with the IPTABLES filtering mechanism is used Negative and Positive mailing list.***

***Keywords : 7 OSI LAYER, Firewall, IPTABLES, , Mekanisme filtering Negative list, dan Positive list.***

## I. PENDAHULUAN

Keamanan jaringan, PC (*Personal Computer*), server-server, dan perangkat komputer Anda yang lainnya memang merupakan faktor yang cukup penting untuk diperhatikan saat ini. Jika beberapa dekade yang lalu keamanan jaringan masih ditempatkan pada urutan prioritas yang rendah, namun akhir-akhir ini perilaku tersebut harus segera diubah. Pasalnya, kejahatan dengan menggunakan bantuan komputer, media komunikasi, dan perangkat elektronik lainnya meningkat sangat tajam belakangan. Hal ini sangat kontras dengan perkembangan kebutuhan perangkat komputer untuk kehidupan sehari-hari yang juga semakin meninggi. Tidak hanya di dalam kegiatan bisnis saja, kehidupan rumah tangga pun sudah sangat relevan jika dilengkapi dengan sebuah komputer. Maka dari itulah, mengapa keamanan jaringan komputer dan PC menjadi begitu penting untuk diperhatikan saat ini.

Pusat Teknologi dan Informasi (PTI) UNSRAT merupakan pusat penyedia jaringan internet untuk Universitas Sam Ratulangi. Dengan Semakin besarnya jaringan yang di buat oleh administrator jaringan PTI, maka keamanan jaringan PTI UNSRAT menjadi prioritas penting bagi administrator. Salah satu metode yang baik untuk keamanan jaringan adalah membuat sebuah *FIREWALL* yang bisa memfiltering paket-paket data (*trafic*).

Dengan latar belakang inilah maka penulis memilih penulisan skripsi ini dengan judul "*Perancangan Filtering Firewall Menggunakkan IP TABLES di Jaringan PTI UNSRAT"*, selain karena IPTABLES adalah perangkat bawaan linux, iptables lebih efisien sebagai firewall di dalam sebuah jaringan. Sehingga diharapkan skripsi ini menjadi referensi bagi *network administrator* atau *network engineer* pada Pusat Teknologi Informasi (PTI) Universitas Sam Ratulangi dalam mengamankan jaringan.

## II. LANDASANTEORI

### *A. TCP/IP*

TCP/IP (singkatan dari *Transmission Control Protocol*/*Internet Protocol*) adalah standar komunikasi data yang digunakan oleh komunitas internet dalam proses tukar-menukar data dari satu komputer ke komputer lain di dalam jaringan Internet. Protokol ini tidaklah dapat berdiri sendiri, karena memang protokol ini berupa kumpulan protokol (*protocol suite*). Protokol ini juga merupakan protokol yang paling banyak digunakan saat ini. Data tersebut diimplementasikan dalam bentuk perangkat lunak (*software*) di sistem operasi. Protokol ini juga bersifat *routable* yang berarti protokol ini cocok untuk menghubungkan sistem-sistem berbeda (seperti Microsoft Windows dan keluarga UNIX) untuk membentuk jaringan yang heterogen

Dalam TCP/IP terdapat 5 layer yang akan dijelaskan berikut.

*Physical Layer* (lapisan fisik) merupakan lapisan terbawah yang mendefinisikan besaran fisik seperti media komunikasi, tegangan, arus, dsb. Lapisan ini dapat bervariasi bergantung pada media komunikasi pada jaringan yang bersangkutan.

*Network Access Layer* mempunyai fungsi yang mirip dengan *Data Link layer* pada OSI. Lapisan ini mengatur penyaluran data *frame-frame* data pada media fisik yang digunakan secara

handal. Lapisan ini biasanya memberikan servis untuk deteksi dan koreksi kesalahan dari data yang ditransmisikan. Beberapa contoh protokol yang digunakan pada lapisan ini adalah X.25 jaringan publik, *Ethernet* untuk jaringan *Etehernet*, AX.25 untuk jaringan Paket Radio dsb.

*Internet Layer* mendefinisikan bagaimana hubungan dapat terjadi antara dua pihak yang berada pada jaringan yang berbeda seperti *Network Layer* pada OSI. Pada jaringan Internet yang terdiri atas puluhan juta host dan ratusan ribu jaringan lokal, lapisan ini bertugas untuk menjamin agar suatu paket yang dikirimkan dapat menemukan tujuannya di mana pun berada.

*Transport Layer* mendefinisikan cara-cara untuk melakukan pengiriman data antara *end to end host* secara handal. Lapisan ini menjamin bahwa informasi yang diterima pada sisi penerima adalah sama dengan informasi yang dikirimkan pada pengirim.

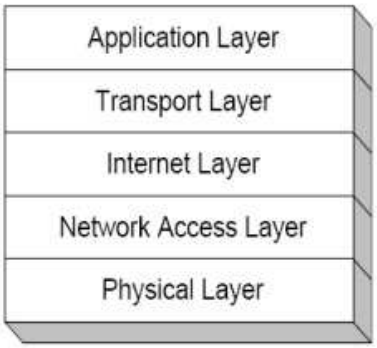
*Application Layer* merupakan lapisan terakhir dalam arsitektur TCP/IP yang berfungsi mendefinisikan aplikasi-aplikasi yang dijalankan pada jaringan. Karena itu, terdapat banyak protokol pada lapisan ini, sesuai dengan banyaknya aplikasi TCP/IP yang dapat dijalankan. Contohnya adalah SMTP (*Simple Mail Transfer Protocol*) untuk pengiriman e-mail , FTP (*File Transfer Protoco*l) untuk transfer *file*, HTTP (*Hyper Text Transfer Protocol*) untuk aplikasi web, NNTP (*Network News Transfer Protocol*) untuk distribusi news group dan lain-lain. Setiap aplikasi pada umumnya menggunakan protocol TCP dan IP, sehingga keseluruhan keluarga protokol ini dinamai dengan TCP/IP.

Lapisan-lapisan tersebut dapat digambarkan seperti yang disajikan pada gambar 1.

### *B. OSI (Open System Interconnectioan) Layer*

Model referensi jaringan terbuka *OSI* atau *OSI Reference Model for open networking* adalah sebuah model arsitektural jaringan yang dikembangkan oleh badan *International Organization for Standardization (ISO)* di Eropa pada tahun 1977. *OSI* sendiri merupakan singkatan dari *Open System Interconnection.* Model ini disebut juga dengan model "Model tujuh lapis OSI" (*OSI seven layer model*).

Struktur tujuh lapis model OSI, bersamaan dengan protocol data unit pada setiap lapisan. Pada *protocol* model OSI standar, *protocol* dibagi menjadi 7 lapisan/layer, yaitu *Physical Layer*, *Data Link Layer*, *Network Layer*, *Transport Layer*, *Session Layer*, *Presentation Layer*, dan *Application Layer* seperti pada Gambar 2.




Gambar 1. *Layer TCP/IP*

### *C. Jaringan Komputer*

Jaringan memiliki banyak arti, tetapi kata jaringan yang digunakan di sini berada dalam lingkup studi teknologi informasi yang memiliki definisi sebagai kumpulan dua atau lebih komputer yang masing-masing berdiri sendiri dan terhubung melalui sebuah teknologi. Dengan menghubungkan dua atau lebih komputer, memungkinkan pengaksesan data, pertukaran file, dan komunikasi satu sama lain (Halin, 1996). Hubungan antar komputer tersebut tidak terbatas pada kabel tembaga (*copper cable*) saja, tetapi juga dapat melalui kabel serat kaca (*fiber optic*), gelombang mikro, gelombang *infrared*, dan juga melalui satelit (Tanenbaum, 2003).

Jenis-jenis topologi jaringan adalah *Bus Topology*, *Ring Topology*, *Star Topology*, *Extended Star Topology*, *Hierarchial Topology*, *Mash Topology*

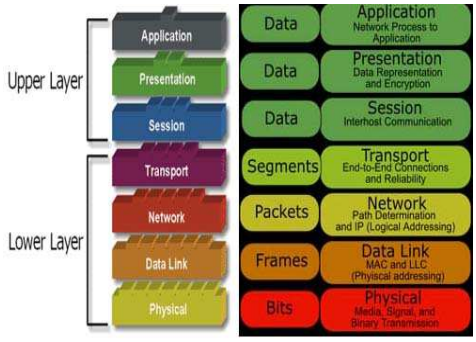
### *D. Firewall*

Firewall adalah sebuah sistem pengaman, jadi firewall bisa berupa apapun baik hardware maupun software. Firewall dapat digunakan untuk memfilter paket-paket dari luar dan dalam jaringan di mana ia berada. Jika pada kondisi normal semua orang dari luar jaringan anda dapat bermain-main ke komputer anda, dengan firewall semua itu dapat diatasi dengan mudah.

Firewall yang sederhana biasanya tidak memiliki kemampuan melakukan filtering terhadap paket berdasarkan isi dari paket tersebut. Sebagai contoh, firewall tidak memiliki kemampuan melakukan filtering terhadap e-mail bervirus yang Anda download atau terhadap halaman web yang tidak pantas untuk dibuka. Yang bisa dilakukan firewall adalah melakukan blokir terhadap alamat IP dari mail server yang mengirimkan virus atau alamat halaman web yang dilarang untuk dibuka. Dengan kata lain, firewall merupakan sistem pertahanan yang paling depan untuk jaringan Anda.

### *E. IPTABLES*

Iptables mengizinkan user untuk mengontrol sepenuhnya jaringan melalui paket IP dengan system *LINUX* yang diimplementasikan pada kernel Linux. Sebuah kebijakan atau Policy dapat dibuat dengan iptables sebagai polisi lalulintas jaringan. Sebuah policy pada iptables dibuat berdasarkan sekumpulan peraturan yang diberikan pada kernel untuk mengatur setiap paket yang datang. Pada iptable ada istilah yang disebut dengan Ipchain yang merupakan daftar aturan bawaan dalam Iptables. Ketiga chain tersebut adalah *INPUT*, *OUTPUT* dan *FORWARD*.




Gambar 2.*Layer TCP/IP*

Sebuah rantai adalah aturan-aturan yang telah ditentukan. Setiap aturan menyatakan "jika paket memiliki informasi awal (*header*) seperti ini, maka inilah yang harus dilakukan terhadap paket". Jika aturan tersebut tidak sesuai dengan paket, maka aturan berikutnya akan memproses paket tersebut. Apabila sampai aturan terakhir yang ada, paket tersebut belum memenuhi salah satu aturan, maka kernel akan melihat kebijakan bawaan (*default*) untuk memutuskan apa yang harus dilakukan kepada paket tersebut. Ada dua kebijakan bawaan yaitu default *DROP* dan default *ACCEPT*.

## III. METODOLOGI PENELITIAN

### *A. Tempat dan Waktu Penelitian*

Dalam penelitian tugas ini, penulis mengambil tempat penelitian pada lokasi wisata kuliner di Kota Manado dan sekitarnya, Ruang Laboratorium Sistem Komputer Jurusan Teknik Elektro, dan rumah penulis dengan waktu penelitian sampai penulis memperoleh data-data guna penulisan tugas akhir ini.

### *B. Bahan dan Peralatan*

Dalam mengerjakan tugas akhir ini, penulis menggunakan menggunakan peralatan *Notebook* dengan spesifikasi *Intel Core i5 processor 430M (2,26 GHz, 1066 MHz FSB), NVIDIA GeForce 310M, 4 GB Memory DDR3 PC8500, 500 GB HDD, Windows 7 Operating System*

Komputer yang berfungsi sebagai *router* dan *firewal*l dengan spesifikasi *Intel Pentium D 3,0 GHz, HIS 1550GT GPU, 2 GB Memory DDR2, 160 GB HDD, 2 buah ethernet card*

### *C. Prosedur Penelitian*

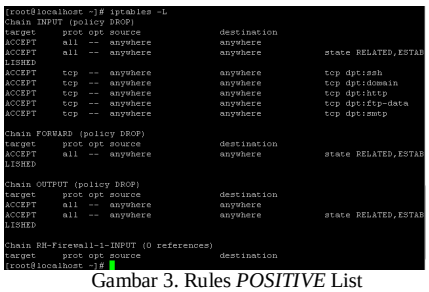
Terdapat tiga prosedur yang dilakukan dalam pembuatan aplikasi ini yang akan diuraikan.

Studi literatur merupakan prosedur untuk mendapatkan literatur / artikel tentang filtering firewall dengan IP Table, kemudian Mempelajari Sistem jaringan yang berjalan di PTI-Unsrat dan memahami pemahaman dari jaringan yang terpasang., setelah itu melakukan Evaluasi Data untuk mendapatkan data-data penggunaan jaringan atau konten-konten yang di akses oleh pengguna internet dan memisahkan konten yang bisa di akses dengan yang tidak perlu.

Perancangan Sistem, dalam hal ini membuat mekanisme filtering dan titik-titk yang akan ditempati firewall

Implementasi Sistem, Instalasi Perangkat Keras dan Perangkat Lunak dari sistem yang dirancang.

Melakukan monitoring jaringan untuk melihat bahwa filtering sudah berjalan dengan baik




Gambar 3. Rules *POSITIVE* List

### *D. Evaluasi Data*

Untuk mendapatkan informasi dari pengguna internet berupa konten yang diakses, diperoleh dengan cara klarifikasi dengan administrator jaringan PTI Unsrat tentang konten-konten yang sering di akses oleh pengguna internet PTI Unsrat, kemudian menyesuaikan dengan aturan-aturan yang diberlakukan di PTI Unsrat tentang konten-konten apa saja yang bisa di akses.

### *E. Perancangan Sistem*

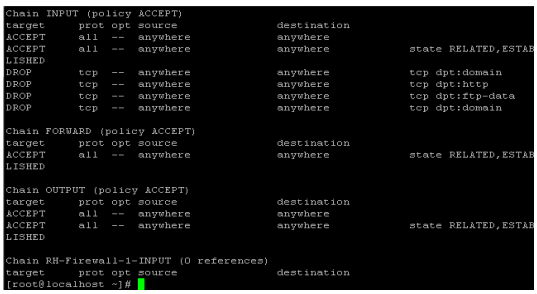
Dalam pembuatan tugas akhir ini penulis merancang sebuah komputer PC menjadi *firewall* dengan menggunakkan *IPTables* yang merupakan perangkat bawaan sistem operasi *Linux CentOs* 5.0. pertama yang dilakukan adalah dengan menginstal sistem operasi linux centos 5.0 di sebuah pc, kemudian aktifkan *iptables.*

### *F. Implementasi Sistem*

Ada dua cara iplementasi sistem di dalam *filtering firewall* yaitu instalasi *hardware* dan instalasi *software.*

Untuk Instalasi hardware digunakan spesifikasinya telah di paparkan dalam bagian 3.B yang telah di tunjukkan sebelumnya. Komputer akan berfungsi sebagai router dan Pc terpasang 2 buah ethernet card yang merupakan media untuk menghubungkan dua jaringan yang berbeda.

Untuk Intalasi *software* menggunakkan sistem operasi distro Linux CentOS, OS (*Operating System*) ini berfungsi sebagai seperangkat program yang mengelola sumber daya perangkat keras komputer, dan menyediakan layanan umum untuk aplikasi perangkat lunak. Konfigurasi jaringan instalasi membutuhkan alamat ip pada kedua buah ethernet card yang terpasang. Instalasi aplikasi menggunakkan *IPTABLES*, aplikasi ini berfungsi sebagai tools dalam sistem operasi linux yang berfungsi sebagai alat untuk melakukan filter (penyaringan) terhadap (*trafic*) lalulintas data. ataupun traffic yang sekedar melewati komputer kita. untuk Konfigurasi filtering, ada dua konfigurasi yaitu *positive list* dan *negative list*. untuk *Positive list* merupakan proses filtering dimana semua paket yang masuk akang di Drop dan hanya jalur-jalur atau port-port tertentu saja yang dibuka untuk bisa di akses, bentuk *POSITIVE* list ini disajikan pada gambar 3. Untuk *Negative list* merupakan proses filtering dimana semua jalur-jalur atau port kita buka, kemudian menutup satu persatu port-port yang penting dalam server kita seperti *DNS,WEB,FTP,SMTP*. Bentuk *negative list* disajikan pada gambar 4.




Gambar 4 Rules Negative List

*G. Konfigurasi IP Tables*

Sebelum konfigurasi ini dilakukan, perlu dilakukan pengecekan apakah kernel *distro linux* centos sudah mendukung atau terinstalnya iptables. Jika ada hasil yang ditampilkan maka distro linux sudah terinstal *IPTABLES*. Proses ini disajikan pada gambar 5.

Setelah itu dilakukan pengaktifan ipv4 *forwarding* agar paket dapat diteruskan oleh router melalui proses routing, tapi sebelum itu, command file /etc/sysctl.conf**:** dan menghapus tanda (#) pada file konfigurasi. Dan untuk pengaktifan IP *forwarding*, harus deberikan angka (1) pada net.ipv4.ipforward. Untuk melihat tahapan ini, disajikan pada gambar 6, gambar 7 dan gambar 8.

```
Installed Packages
Name       : iptables
Arch       : x86_64
Version    : 1.3.5
Release    : 9.1.el5
Size       : 661 k
Repo       : installed
Summary    : Tools for managing Linux kernel packet filtering capabilities.
URL        : http://www.netfilter.org/
License    : GPL
Description: The iptables utility controls the network packet filtering code in
           : the Linux kernel. If you need to set up firewalls and/or IP
           : masquerading, you should install this package.
```

Gambar 5. Tahapan instalisasi IPTables

```
[root@localhost ~]# nano /etc/sysctl.conf
```

Gambar 6. Tahapan mumbuka Commend File

```
 GNU nano 1.3.12          File: /etc/sysctl.conf                    Modified

# Kernel sysctl configuration file for Red Hat Linux
#
# For binary values, 0 is disabled, 1 is enabled.  See sysctl(8) and
# sysctl.conf(5) for more details.

# Controls IP packet forwarding
net.ipv4.ip_forward = 0

# Controls source route verification
net.ipv4.conf.default.rp_filter = 1

# Do not accept source routing
net.ipv4.conf.default.accept_source_route = 0
```

Gambar 7. Tahapan Menghapus "#" pada commend file

Selanjutnya kongfigurasi ip Masquarade. Ip Masquarade adalah salah satu fasilitas di linux yang memungkinkan komputer yang tidak memiliki nomor IP resmi dapat tersambung ke internet melewati komputer berbasis sistem operasi linux. Ip Masquarade dibutuhkan jika jaringan anda mempunyai nomor IP resmi yang lebih sedikit dari pada jumlah komputer yang ada. Berikut ini adalah command untuk mengaktiftkan ip masquarade:

Iptables –t nat –A *POSTROUTING* –o eth0 –j *MASQUARADE*. Untuk melihat tahapan pengaktifan ip Masquarade disajikan pada gambar 9 dan gambar 10.

```
 GNU nano 1.3.12          File: /etc/sysctl.conf

# Kernel sysctl configuration file for Red Hat Linux
#
# For binary values, 0 is disabled, 1 is enabled.  See sysctl(8) and
# sysctl.conf(5) for more details.

# Controls IP packet forwarding
net.ipv4.ip_forward = 1

# Controls source route verification
net.ipv4.conf.default.rp_filter = 1

# Do not accept source routing
net.ipv4.conf.default.accept_source_route = 0

# Controls the System Request debugging functionality of the kernel
kernel.sysrq = 0

# Controls whether core dumps will append the PID to the core filename
# Useful for debugging multi-threaded applications
kernel.core_uses_pid = 1

# Controls the use of TCP syncookies
net.ipv4.tcp_syncookies = 1

                [ Read 35 lines ]
^G Get Help  ^O WriteOut  ^R Read File ^Y Prev Page ^K Cut Text  ^C Cur Pos
^X Exit      ^J Justify   ^W Where Is  ^V Next Page ^U UnCut Text^T To Spell
```

Gambar 8. Aktivasi ip forwarding dengan mengisi nilai "1"

```
login as: root
root@192.168.254.111's password:
Last login: Tue Apr 23 13:49:24 2013 from 192.168.254.110
[root@localhost ~]# iptables -t nat -A POSTROUTING -o eth0 -j MASQUARADE
```

Gambar 9. Aktivasi ip masquarade

```
login as: root
root@192.168.254.111's password:
Last login: Tue Apr 23 13:49:24 2013 from 192.168.254.110
[root@localhost ~]# iptables -t nat -nL
Chain PREROUTING (policy ACCEPT)
target     prot opt source               destination

Chain POSTROUTING (policy ACCEPT)
target     prot opt source               destination
MASQUERADE  all  --  0.0.0.0/0            0.0.0.0/0

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination
[root@localhost ~]#
```

Gambar 10. Status IP masquerade yang telah active

*H. Konfigurasi DHCP server*

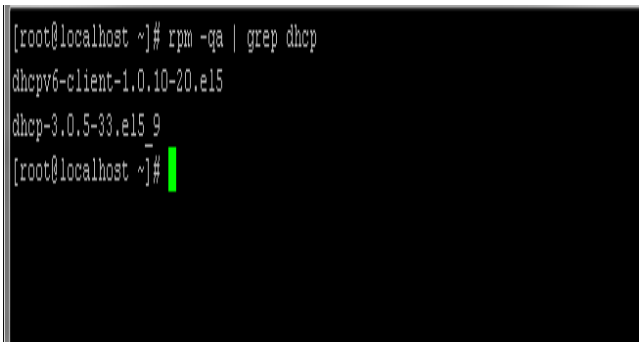
Langkah pertama untuk membuat dhcp *server* cek dahulu apakah *dhcp* sudah terinstall atau belum dengan *command* di bawah ini.

[root@localhost ] #rpm –qa | grep dhcp.

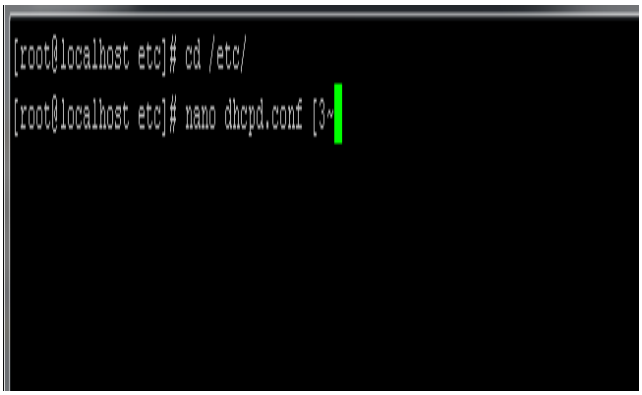
Jika sudah terinstal, buka konfigurasi *dhcpd*.conf. untuk tahapan ini disajikan pada gambar 11 dan gambar 12.

Kemudian buka file konfigurasi dhcpd.conf. dengan menggunakan *command* di bawah ini.

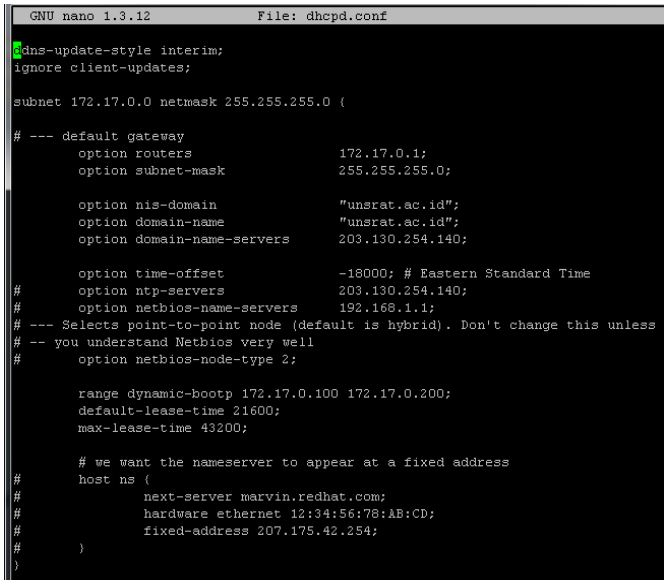
[root@localhost etc] #nano dhcpd.conf




Gambar 11. Fitur DHCP yang telah terinstall




Gambar 12. Membuka file konfigurasi DHCP




Gambar 13. Setting konfigurasi file DHCP

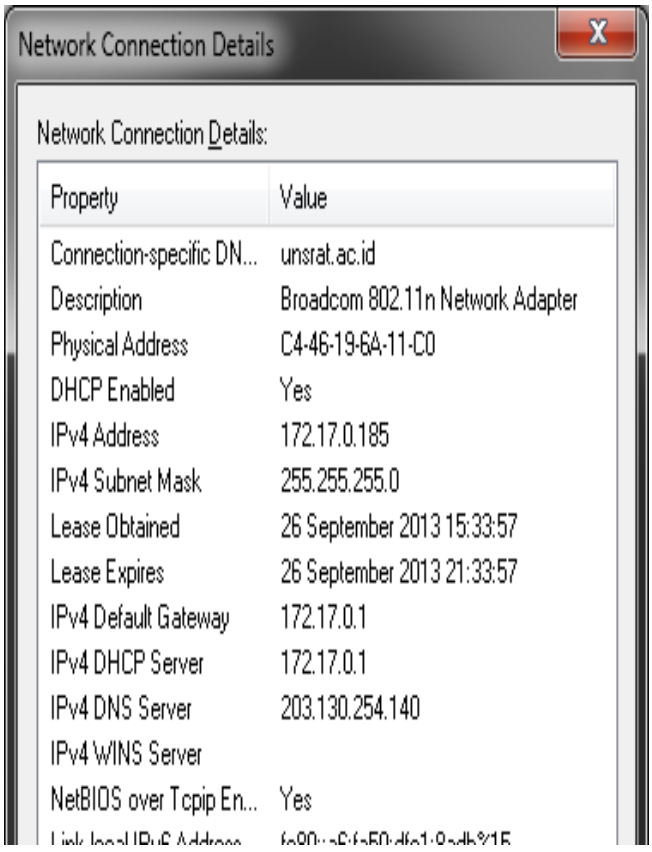
Jika konfigurasi sudah sesuai, kemudian simpan file dengan nama dhcpd.conf. untuk tahapan ini disajikan pada gambar 13.

Selanjutnya aktifkan aplikasi dhcpd dengan menggunakan commend di bawah ini.

*[root@localhost dhcp] #service dhcpd start*

Setelah *dhcp server* telah aktif kita akan menguji apakah komputer server *dhcp* bekerja dengan baik atau tidak. Untuk tahapan ini disajikan pada gambar 14. Kita dapat melihat contoh computer client yang telah mendapatkan IP *automatic* dari server yang disajikan pada gambar 15.


Gambar 14. Fitur DHCP yang telah terinstall




Gambar 15. Computer client yang telah mendapatkan IP automati

## IV. HASIL DAN PEMBAHASAN

Di dalam sebuah layanan jaringan terdapat layanan-layanan penting seperti *WEB,DNS,SSH* dan *MAIL*, layanan-layanan jaringan tersebut memiliki jalur-jalur atau portport tertentu. Tiap-tiap jalur atu port memiliki nomor jalur atau nomor port misalnya, *WEB* : port tcp 80,443, *MAIL* : port 25 / 110, DNS : port tcp/udp 54, *SSH* : port 22

Ada dua jenis metode mekanisme filtering yaitu *NEGATIVE LIST* dan *POSITIVE LIST* dimana kedua duanya memiliki konfigurasi masing-masing.

### *A. Negative List*

Mekanisme filtering dengan menggunakan metode *Negative* list adalah dimana secara *DEFAULT* semua port dibuka baru kemudian satu persatu di tutup port yang diinginkan. Aturan atau rules iptables menggunakkan mekanisme *NEGATIVE* list

```
Chain INPUT (policy ACCEPT)
target     prot opt source               destination
ACCEPT     all  --  anywhere             anywhere
ACCEPT     all  --  anywhere             anywhere            state RELATED,ESTAB
LISHED
DROP       tcp  --  anywhere             anywhere            tcp dpt:domain
DROP       tcp  --  anywhere             anywhere            tcp dpt:http
DROP       tcp  --  anywhere             anywhere            tcp dpt:ftp-data
DROP       tcp  --  anywhere             anywhere            tcp dpt:domain

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination
ACCEPT     all  --  anywhere             anywhere            state RELATED,ESTAB
LISHED

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination
ACCEPT     all  --  anywhere             anywhere
ACCEPT     all  --  anywhere             anywhere            state RELATED,ESTAB
LISHED

Chain RH-Firewall-1-INPUT (0 references)
target     prot opt source               destination
[root@localhost ~]#
```

Gambar 16. mekanisme filtering menggunakkan NEGATIVE list

```
[root@localhost ~]# iptables -A FORWARD -p tcp --dport 80 -j DROP
```

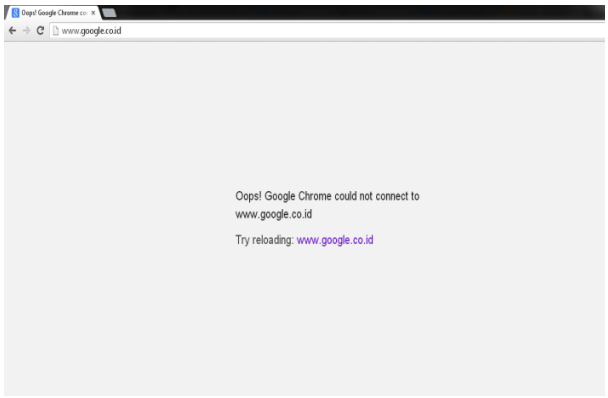
Gambar 17. Command rule iptables untuk menutup port 80

```
LISHED
ACCEPT     all  --  172.17.0.2           0.0.0.0/0
LOG        all  --  0.0.0.0/0            0.0.0.0/0           limit: avg 2/min bu
rst 2 LOG flags 0 level 4 prefix `** INPUT DROP **'

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination
ACCEPT     all  --  0.0.0.0/0            0.0.0.0/0           state RELATED,ESTAB
LISHED
LOG        all  --  0.0.0.0/0            0.0.0.0/0           limit: avg 2/min bu
rst 2 LOG flags 0 level 4 prefix `** FORWARD DROP **'
DROP       tcp  --  0.0.0.0/0            0.0.0.0/0           tcp dpt:80

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination
ACCEPT     all  --  0.0.0.0/0            0.0.0.0/0           state RELATED,ESTAB
LISHED
ACCEPT     all  --  0.0.0.0/0            172.17.0.2
LOG        all  --  0.0.0.0/0            0.0.0.0/0           limit: avg 2/min bu
rst 2 LOG flags 0 level 4 prefix `** OUTPUT DROP **'
```

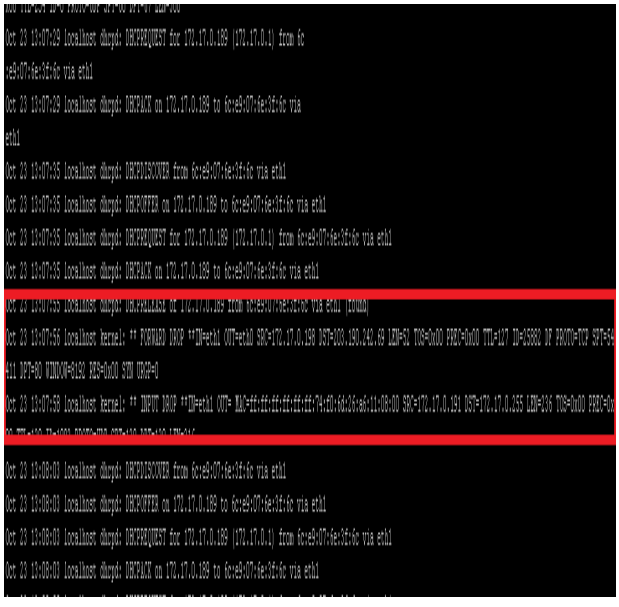
Gambar 18. Hasil setelah menginput command rule untuk menutup port 80

dimana semua akses perbolehkan lalu kemudian menutup port-port penting agar tidak bisa di akses dari luar atau sembarang orang. Aturan ini disajikan pada gambar 16.

Untuk konfigurasi *Negative* list, yang harus dilakukan terlebi dahulu adalah membuka atura iptables, kemudian menutup port-port seperti *WEB* (port 80) yang disajikan pada gambar 17, gambar 18, gambar 19, dan gambar 20. *MAIL* (port 25/110) yang disajikan pada gambar 21, gambar 22. Gambar 23, gambar 24. *DNS* (port 53) yang berfungsi untuk merubah sebuah alamat ip ke alamat domain. Untuk penutupan DNS disajikan pada gambar 25, gambar 26, dan gambar 27. *SSH* (port 22) yang merupakan sebuah protokol untuk meremote suatu komputer server dan bisa juga dijadikan sebagai alternatif untuf *FTP transfer file* dan *SSH* juga mendukung tunneling *forwarding* tcp port. Untuk penutupan *SSH* disajikan pada gambar 28, gambar 29, dan gambar 30. Penutupan port-port ini dilakukan dengan cara menambahkan commend *DROP* pada *commend* file port tersebut.




Gambar 19. Menunjukkan bahwa setelah menutup akses port 80 (http), klien tidak bisa lagi mengakses sebuah situs



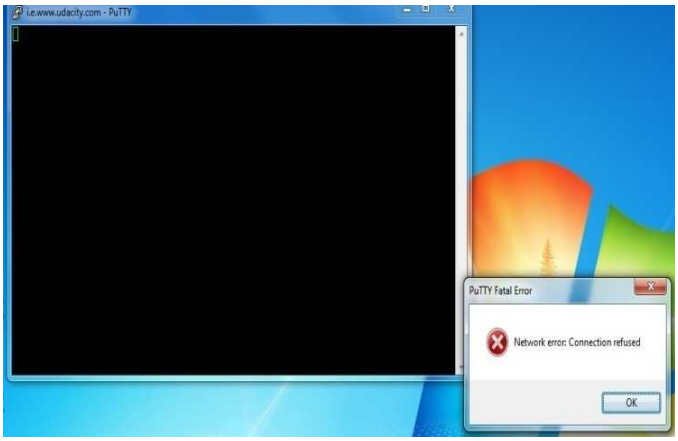
Gambar 20. Hasil logging file dari port 80, dimana dikatakan bahwa sebuah koneksi client akan mengakses port 80 atau http tapi di tolak atau di DROP oleh rules IPTABLES

```
[root@localhost ~]# iptables -A INPUT -p tcp --dport 110 -j DROP
[root@localhost ~]# iptables -A INPUT -p tcp --dport 25 -j DROP
```

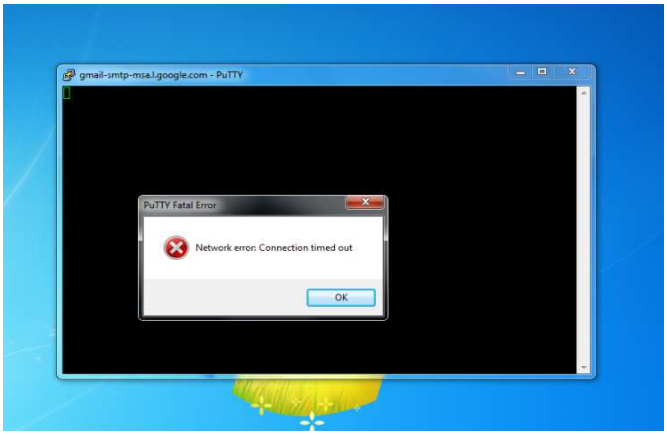
Gambar 21. Command rule untuk menutup port smtp (25)

```
[root@localhost ~]# iptables -L
Chain INPUT (policy ACCEPT)
target     prot opt source               destination
ACCEPT     all  --  anywhere             anywhere            state RELATED,ESTABLISHED
ACCEPT     all  --  172.17.0.2           anywhere

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination
ACCEPT     all  --  anywhere             anywhere            state RELATED,ESTABLISHED
DROP       tcp  --  anywhere             anywhere            tcp dpt:smtp
DROP       tcp  --  anywhere             anywhere            tcp dpt:pop3
LOG        all  --  anywhere             anywhere            limit: avg 2/min burst 2 LOG

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination
ACCEPT     all  --  anywhere             anywhere            state RELATED,ESTABLISHED
ACCEPT     all  --  anywhere             172.17.0.2

Chain LOGGING (0 references)
target     prot opt source               destination
```

Gambar 22. Hasil setelah menambahkan command rules




Gambar 23. Hasil dari menutup nomor port tujuan 25 yaitu Mail server



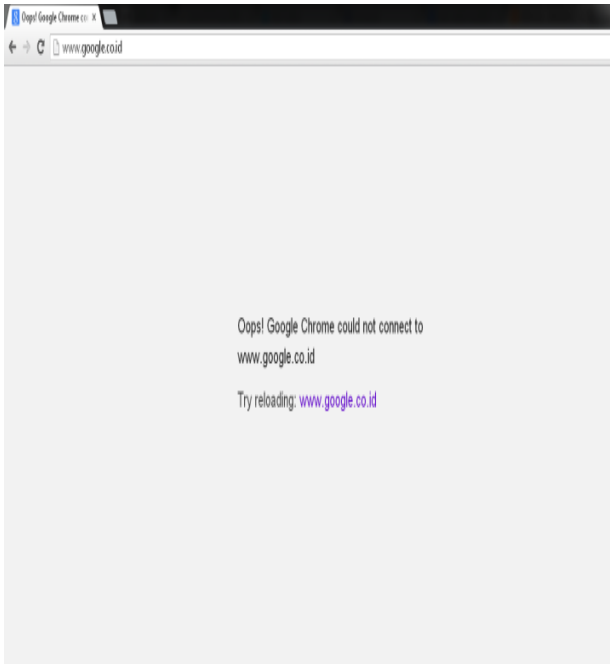

Gambar gambar 24. Hasil bahwa rules berjalan dengan baik untuk tidak mengizinkan mengakses mail server google menggunakkan jaringan PTI

```
target     prot opt source               destination
[root@localhost ~]# iptables -A FORWARD -p udp -s 172.17.0.198 --dport 53 -j DROP
```

Gambar 25. Konfigurasi aturan untuk menutup port 53

```
Chain INPUT (policy ACCEPT)
target     prot opt source               destination
ACCEPT     all  --  anywhere             anywhere            state RELATED,ESTABLISHED
ACCEPT     all  --  172.17.0.2           anywhere
LOG        all  --  anywhere             anywhere            limit: avg 2/min burst 2 LOG level warning prefix `** INPUT DROP **'

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination
ACCEPT     all  --  anywhere             anywhere            state RELATED,ESTABLISHED
LOG        all  --  anywhere             anywhere            limit: avg 2/min burst 2 LOG level warning prefix `** FORWARD DROP **'
DROP       udp  --  172.17.0.198         anywhere            udp dpt:domain

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination
ACCEPT     all  --  anywhere             anywhere            state RELATED,ESTABLISHED
ACCEPT     all  --  anywhere             172.17.0.2
LOG        all  --  anywhere             anywhere            limit: avg 2/min burst 2 LOG level warning prefix `** OUTPUT DROP **'
```

Gambar 26. Hasil setelah menambahkan rule untuk menutup port 53 ke dalam aturan-aturan iptables bisa dilihat di dalam chain FORWARD



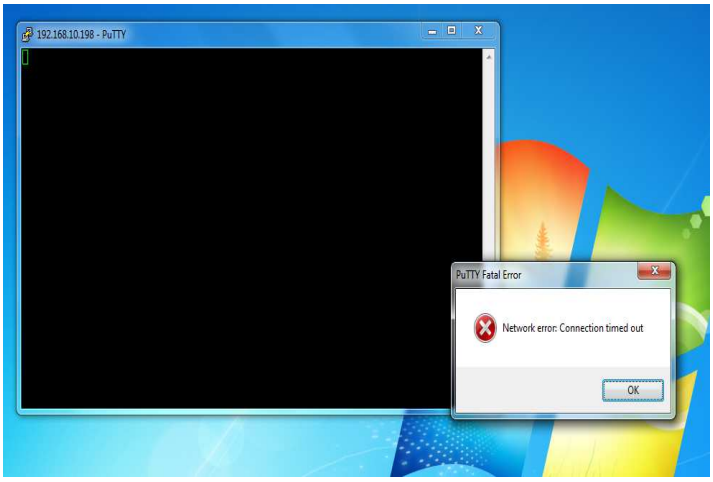

Gambar 27. Klien tidak bisa melakukan browsing ke sebuah situs

```
[root@localhost ~]# iptables -A INPUT -p tcp --dport 22 -j DROP
```

Gambar 28.Rule untuk menutup port *SSH*

```
target     prot opt source               destination
[root@localhost ~]# iptables -A INPUT -p tcp --dport 22 -j DROP
[root@localhost ~]# iptables -L
Chain INPUT (policy DROP)
target     prot opt source               destination
ACCEPT     all  --  anywhere             anywhere            state RELATED,ESTABLISHED
ACCEPT     all  --  172.17.0.2           anywhere
LOG        all  --  anywhere             anywhere            limit: avg 2/min burst 2 LOG level warning prefix `
DROP       tcp  --  anywhere             anywhere            tcp dpt:ssh

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination
ACCEPT     all  --  anywhere             anywhere            state RELATED,ESTABLISHED
LOG        all  --  anywhere             anywhere            limit: avg 2/min burst 2 LOG level warning prefix `

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination
ACCEPT     all  --  anywhere             anywhere            state RELATED,ESTABLISHED
ACCEPT     all  --  anywhere             172.17.0.2
LOG        all  --  anywhere             anywhere            limit: avg 2/min burst 2 LOG level warning prefix `
```

Gambar 29. Hasil setelah menambahkan Command rule




Gambar 30. Klien tidak diperbolehkan untuk megakses atau melakukan tunneling ke nomor port tujuan 22

```
[root@localhost ~]# iptables -L
Chain INPUT (policy DROP)
target     prot opt source               destination
ACCEPT     all  --  anywhere             anywhere
ACCEPT     all  --  anywhere             anywhere            state RELATED,ESTAB
LISHED
ACCEPT     tcp  --  anywhere             anywhere            tcp dpt:ssh
ACCEPT     tcp  --  anywhere             anywhere            tcp dpt:domain
ACCEPT     tcp  --  anywhere             anywhere            tcp dpt:http
ACCEPT     tcp  --  anywhere             anywhere            tcp dpt:ftp-data
ACCEPT     tcp  --  anywhere             anywhere            tcp dpt:smtp

Chain FORWARD (policy DROP)
target     prot opt source               destination
ACCEPT     all  --  anywhere             anywhere            state RELATED,ESTAB
LISHED

Chain OUTPUT (policy DROP)
target     prot opt source               destination
ACCEPT     all  --  anywhere             anywhere
ACCEPT     all  --  anywhere             anywhere            state RELATED,ESTAB
LISHED
```

Gambar 31. Mekanisme filtering *POSITIVE* list

```
Chain INPUT (policy DROP)
target     prot opt source               destination
ACCEPT     all  --  anywhere             anywhere
ACCEPT     all  --  anywhere             anywhere
LISHED
ACCEPT     tcp  --  anywhere             anywhere
ACCEPT     tcp  --  anywhere             anywhere
ACCEPT     tcp  --  anywhere             anywhere
ACCEPT     tcp  --  anywhere             anywhere
ACCEPT     tcp  --  anywhere             anywhere

Chain FORWARD (policy DROP)
target     prot opt source               destination
ACCEPT     all  --  anywhere             anywhere
LISHED

Chain OUTPUT (policy DROP)
target     prot opt source               destination
ACCEPT     all  --  anywhere             anywhere
ACCEPT     all  --  anywhere             anywhere
LISHED
```

Gambar 32. Semua Port telah di tutup

### B. *Positive List*

Mekanisme *filtering* menggunakan metode *POSITIVE* list adalah dimana secara *DEFAULT* semua port kita tutup dan baru kemudian satu persatu kita buka port yang diinginkan. Untuk mekanisme *filtering* dari *positive* list disajikan pada gambar 31.

semua trafik yang masuk ke dalam chain iptables di *DROP* yang artinya semua trafik dalam bentuk apapun dan dari protokol manapun tidak diizinkan atau di blokir, kecuali alamat-alamat bernomor port tujuan yang kita inginkan untuk bisa diakses seperti 22(*SSH*), 53(*DNS*), 80(*HTTP*), 25(*SMTP*).

Konfigurasi Command Rule *POSITIVE* list, Pertama merubah semua chain didalam iptables menjadi DROP kemudian membuka satu persatu port-port yang dibutuhkan di dalam layanan jaringan PTI UNSRAT. Untuk bisa mengakses layanan-layanan penting dalam jaringan makan kita harus membuka satu persatu layanan tersebut. untuk melihat port yang sudah ditutup disajikan pada gambar 32. Mekanisme konfigurasi dari *POSITIVE* list hampir sama dengan *negative* list, akan tetapi untuk *POSITIVE* list bukan ditambahkan *Command DROP* pada *command* file port tetapi ditambahkan commend *ACCEPT* untuk membuka port-port penting yang diinginkan.

## V. PENUTUP

### A. *Kesimpulan*

Sesuai dengan hasil yang telah dilakukan di atas, dalam perancangan *filtering firewall* menggunakkan *IPTABLES* di jaringan PTI UNSRAT bahwa *filtering* menggunakkan metode positif list lebih aman dilakukan karna secara default semua port ditutup, tapi untuk konfigurasinya cukup sulit untuk dilakukan. Sedangkan untuk metode *Negatif* list sangat mudah untuk di lakukan karena semua port secara default kita buka, lalu kemudian kita tutup portnya satu-satu, tetapi kelemahannya dari metode ini adalah bisa terjadi kemungkinan kelalaian tidak menutup port.

### B. *Saran*

Untuk jaringan seperti PTI UNSRAT lebih mudah menggunakkan metode *positif* list karna lebih mudah untuk dilakukan dan lebih aman dibandingkan dengan menggunakkan metode *negatif* list.

## DAFTAR PUSTAKA


[1] A. P.Jiwa, Agus, *Penggunaan Firewall Untuk Menjaga Keamanan Sistem Jaringan Komputer,* Artikel Populer , IlmuKomputer, 2009.

[2] A. Raharja, A.Yunianto, dkk., *Open Source Campus Agreement, Modul Pelatihan*, Pengenalan Linux, 2001.

[3] Archlinux, IPTABLES, 2013, Tersedia di: https://wiki.archlinux.org/index.php/iptables.

[4] F. Pryianto, PanduanPraktis *Firewall* Dengan IP*Tables*, 2008, Tersedia di: http://linux2.arinet.org.

[5] Forum Pengguna Linux Indonesia [Online], Tersedia di: http://linux.or.id/view-topic-[Share]-Setting-Iptables-Pada-Ubuntu.html.

[6] G. P., Faiz, *Firewall* Dengan Menggunakan IP*Tables*, 2005, Tersedia di : http://purwakarta.org.

[7] R. Angenendt, IPTables Configuration, 2009, tersedia di : http://wiki.centos.org/HowTos/Network/IPTables.

[8] L. Adhi, *Desain Dan Implementasi Firewall Dengan Layer 7 Filter Pada Jaringan Teknik Elektro*, Universitas Diponegoro, 2011.

[9] M. Pudja, *Debian GNU/Linux, Konfigurasi Debian Server Teknik Komputer dan Jaringan*, Al-Mansyurin Informatika *Team*, Mojokerto, 2011.