```
#!/bin/sh
#
######################################################################
#
#   Name:           iptables_mail.sh
#   Author:         Chris Fedun 17/12/2016
#   Description:    Base IPTABLES Firewall Configuration for Mail Server
#   Based on:       http://cipherdyne.org/LinuxFirewalls/ch01/
#   Copyright (C) 2017  Christopher Fedun
#
#   This program is free software: you can redistribute it and/or modify
#   it under the terms of the GNU General Public License as published by
#   the Free Software Foundation, either version 3 of the License, or
#   (at your option) any later version.
#
#   This program is distributed in the hope that it will be useful,
#   but WITHOUT ANY WARRANTY; without even the implied warranty of
#   MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the
#   GNU General Public License for more details.
#
#   You should have received a copy of the GNU General Public License
#   along with this program.  If not, see <http://www.gnu.org/licenses/>.
######################################################################
#####Constants#####
IPTABLES=/sbin/iptables
IP6TABLES=/sbin/ip6tables
MODPROBE=/sbin/modprobe
#INT_NET=192.168.10.0/24
IFACE_INT=eth1
IFACE_EXT=eth0
INT_IP="$(ifconfig | grep -A 1 'eth1' | tail -1 | cut -d ':' -f 2 | cut -d ' ' -f 1)"
EXT_IP="$(ifconfig | grep -A 1 'eth0' | tail -1 | cut -d ':' -f 2 | cut -d ' ' -f 1)"
INT_MASK="$(ifconfig 'eth1' | sed -rn '2s/ .*:(.*)$/\1/p')"
EXT_MASK="$(ifconfig 'eth0' | sed -rn '2s/ .*:(.*)$/\1/p')"
INT_NET="$(ipcalc $INT_IP $INT_MASK | grep Network | awk '{print $2}')"
#DNS_SVR_IP=192.168.10.253
#WEB_SVR_IP=192.168.10.253
#EMAIL_SVR_IP=192.168.10.253
#CALL_MANAGER=192.168.10.252
Setup_dir='/root/initial_setup/mail/'

### Flush existing rules ###
echo "[+] Flushing existing iptables rules..."

$IPTABLES -F
$IPTABLES -F -t nat
$IPTABLES -X

### Load connection-tracking modules. ###
$MODPROBE ip_conntrack
$MODPROBE iptable_nat
$MODPROBE ip_conntrack_ftp
$MODPROBE ip_nat_ftp

##### CREATE LOG_DROP CHAIN #####
echo "[+] Setting up LOG_DROP chain..."
$IPTABLES -N LOG_DROP
$IPTABLES -A LOG_DROP -i $IFACE_INT ! -s  $INT_NET -j LOG --log-prefix "SPOOFED PKT "
$IPTABLES -A LOG_DROP -i $IFACE_INT ! -s  $INT_NET -j DROP
$IPTABLES -A LOG_DROP -p icmp -j LOG --log-prefix 'ICMP Block '
$IPTABLES -A LOG_DROP -p icmp -j DROP
$IPTABLES -A LOG_DROP -m conntrack --ctstate INVALID -j LOG --log-prefix "DROP INVALID " --log-ip-options --log-tcp-options
$IPTABLES -A LOG_DROP -m conntrack --ctstate INVALID -j DROP
$IPTABLES -A LOG_DROP -j LOG --log-prefix "DROP " --log-level 6 --log-ip-options --log-tcp-options
```

```
$IPTABLES -A LOG_DROP -j DROP

$IPTABLES -t mangle -N LOG_DROP
$IPTABLES -t mangle -A LOG_DROP -i $IFACE_INT ! -s  $INT_NET -j LOG --log-prefix "SPOOFED PKT "
$IPTABLES -t mangle -A LOG_DROP -i $IFACE_INT ! -s  $INT_NET -j DROP
$IPTABLES -t mangle -A LOG_DROP -p icmp -j LOG --log-prefix 'ICMP Block '
$IPTABLES -t mangle -A LOG_DROP -p icmp -j DROP
$IPTABLES -t mangle -A LOG_DROP -m conntrack --ctstate INVALID -j LOG --log-prefix "DROP INVALID " --log-ip-options --log-tcp-options
$IPTABLES -t mangle -A LOG_DROP -m conntrack --ctstate INVALID -j DROP
$IPTABLES -t mangle -A LOG_DROP -j LOG --log-prefix "DROP " --log-level 6 --log-ip-options --log-tcp-options
$IPTABLES -t mangle -A LOG_DROP -j DROP

##### INPUT chain #####
echo "[+] Setting up INPUT chain..."

### State tracking rules ###
$IPTABLES -A INPUT -m conntrack --ctstate INVALID -j LOG_DROP
$IPTABLES -A INPUT -m conntrack --ctstate ESTABLISHED,RELATED -j ACCEPT


### ACCEPT rules ###
#$IPTABLES -A INPUT -i $IFACE_INT -p tcp -s $INT_NET --dport 22 -m conntrack --ctstate NEW -j ACCEPT
$IPTABLES -A INPUT -p tcp --dport 22 -m conntrack --ctstate NEW -j ACCEPT
$IPTABLES -A INPUT -p tcp --dport 21 -m conntrack --ctstate NEW -j ACCEPT
$IPTABLES -A INPUT -p tcp --dport 25 -m conntrack --ctstate NEW -j ACCEPT
$IPTABLES -A INPUT -p tcp --dport 43 -m conntrack --ctstate NEW -j ACCEPT
$IPTABLES -A INPUT -p tcp --dport 80 -m conntrack --ctstate NEW -j ACCEPT
$IPTABLES -A INPUT -p tcp --dport 443 -m conntrack --ctstate NEW -j ACCEPT
$IPTABLES -A INPUT -p tcp --dport 53 -m conntrack --ctstate NEW -j ACCEPT
$IPTABLES -A INPUT -p udp --dport 53 -m conntrack --ctstate NEW -j ACCEPT

##### To enable possible MYSQL communication between servers #####
$IPTABLES -A INPUT -i $IFACE_INT -p udp --dport 3306 -m conntrack --ctstate NEW -j ACCEPT
$IPTABLES -A INPUT -i $IFACE_INT -p tcp --dport 3306 -m conntrack --ctstate NEW -j ACCEPT
# SMTP and SMTPS #
$IPTABLES -A INPUT -p tcp --dport 25 -m conntrack --ctstate NEW -j ACCEPT
$IPTABLES -A INPUT -p tcp --dport 465 -m conntrack --ctstate NEW -j ACCEPT
$IPTABLES -A INPUT -p tcp --dport 587 -m conntrack --ctstate NEW -j ACCEPT
# IMAP and IMAPS #
$IPTABLES -A INPUT -p tcp --dport 143 -m conntrack --ctstate NEW -j ACCEPT
$IPTABLES -A INPUT -p tcp --dport 993 -m conntrack --ctstate NEW -j ACCEPT
# POP3 and POP3S #
$IPTABLES -A INPUT -p tcp --dport 110 -m conntrack --ctstate NEW -j ACCEPT
$IPTABLES -A INPUT -p tcp --dport 995 -m conntrack --ctstate NEW -j ACCEPT
$IPTABLES -A INPUT -p icmp --icmp-type echo-request -j ACCEPT

### Default INPUT LOG rule ###
$IPTABLES -A INPUT ! -i lo -j LOG_DROP

### Make sure that loopback traffic is accepted ###
$IPTABLES -A INPUT -i lo -j ACCEPT

##### FORWARD chain #####
##### to allow posssible VPN Configuration or Load Balencing#####
echo "[+] Setting up FORWARD chain..."

### State tracking rules ###
$IPTABLES -A FORWARD -m conntrack --ctstate INVALID -j LOG_DROP
$IPTABLES -A FORWARD -m conntrack --ctstate ESTABLISHED,RELATED -j ACCEPT
$IPTABLES -A FORWARD -i $IFACE_INT -o $IFACE_EXT -j ACCEPT
### Default FORWARD LOG rule ###
$IPTABLES -A FORWARD ! -i lo -j LOG_DROP
```

```
##### OUTPUT chain #####
echo "[+] Setting up OUTPUT chain..."
### State tracking rules ###
$IPTABLES -A OUTPUT -m conntrack --ctstate INVALID -j LOG_DROP
$IPTABLES -A OUTPUT -m conntrack --ctstate ESTABLISHED,RELATED -j ACCEPT

### ACCEPT rules for allowing NEW connections out. ###
$IPTABLES -A OUTPUT -p tcp --dport 21 -m conntrack --ctstate NEW -j ACCEPT
$IPTABLES -A OUTPUT -p tcp --dport 22 -m conntrack --ctstate NEW -j ACCEPT
$IPTABLES -A OUTPUT -p tcp --dport 43 -m conntrack --ctstate NEW -j ACCEPT
$IPTABLES -A OUTPUT -p tcp --dport 80 -m conntrack --ctstate NEW -j ACCEPT
$IPTABLES -A OUTPUT -p tcp --dport 443 -m conntrack --ctstate NEW -j ACCEPT
$IPTABLES -A OUTPUT -p tcp --dport 53 -m conntrack --ctstate NEW -j ACCEPT
$IPTABLES -A OUTPUT -p udp --dport 53 -m conntrack --ctstate NEW -j ACCEPT

##### To enable possible MYSQL communication between servers #####
$IPTABLES -A OUTPUT -o $IFACE_INT -p udp --dport 3306 -m conntrack --ctstate NEW -j ACCEPT
$IPTABLES -A OUTPUT -o $IFACE_INT -p tcp --dport 3306 -m conntrack --ctstate NEW -j ACCEPT
# SMTP and SMTPS #
$IPTABLES -A OUTPUT -p tcp --dport 25 -m conntrack --ctstate NEW -j ACCEPT
$IPTABLES -A OUTPUT -p tcp --dport 465 -m conntrack --ctstate NEW -j ACCEPT
$IPTABLES -A OUTPUT -p tcp --dport 587 -m conntrack --ctstate NEW -j ACCEPT
# IMAP and IMAPS #
$IPTABLES -A OUTPUT -p tcp --dport 143 -m conntrack --ctstate NEW -j ACCEPT
$IPTABLES -A OUTPUT -p tcp --dport 993 -m conntrack --ctstate NEW -j ACCEPT
# POP3 and POP3S #
$IPTABLES -A OUTPUT -p tcp --dport 110 -m conntrack --ctstate NEW -j ACCEPT
$IPTABLES -A OUTPUT -p tcp --dport 995 -m conntrack --ctstate NEW -j ACCEPT
$IPTABLES -A OUTPUT -p icmp --icmp-type echo-request -j ACCEPT

### Default OUTPUT LOG rule ###
$IPTABLES -A OUTPUT ! -o lo -j LOG_DROP

### Make sure that loopback traffic is accepted. ###
$IPTABLES -A OUTPUT -o lo -j ACCEPT

# POSTROUTING rule
$IPTABLES -t nat -A POSTROUTING -s $INT_NET -o $IFACE_EXT -j MASQUERADE

##### Forwarding #####
echo "[+] Enabling IP forwarding..."
echo 1 > /proc/sys/net/ipv4/ip_forward

### Setting chain policy settings to DROP. ###
$IPTABLES -P INPUT DROP
$IPTABLES -P OUTPUT DROP
$IPTABLES -P FORWARD DROP
### This policy does not handle IPv6 traffic except to DROP it. ###
echo "[+] Disabling IPv6 traffic..."

$IP6TABLES -P INPUT DROP
$IP6TABLES -P OUTPUT DROP
$IP6TABLES -P FORWARD DROP

##### Basic DDos Prevention #####
bash -x $Setup_dir\ddos_protection_my_build.sh

exit

# END  :)
### EOF ###
```