Netfilter - Iptables Guide v1.2

Writer:Jin Chul, Park
2001.05.11 ROK GMT+9
black@hackerslab.org
http://www.netfilter.or.kr

This Document is to assist linux administrators
Admins are required to set security policy rule
It's recommended to set default rule to target drop
Please see following :
You need to perform kernel compile first before you run this program

Required package : iptables-last-version source || src.rpm .rpm
Kernel netfilter Option check [Y]

/etc/rc.d/init.d/iptables
Check the runlevel /etc/rc.d/init.d/iptables **Chkconfig list | grep iptables** **iptables 0:off 1:off 2:on 3:on 4:on 5:on 6:off** **chkconfig --level 2345 ipchains off** **chkconfig –level 2345 iptables on**
Iptables module list
[root@www init.d]# lsmod Module Size Used by iptable_mangle 2048 0 (autoclean) (unused) iptable_nat 23856 0 (autoclean) (unused) ip_conntrack 25376 1 (autoclean) [iptable_nat] iptable_filter 2080 0 (autoclean) (unused) ip_tables 13984 5 [iptable_mangle iptable_nat iptable_filter]

Add iptables to start init.d to run mode 2345

Use lsmod command to make sure iptables modules are loaded correctly

Examples)

Suppose you operate a webserver, following policy is applicable:

Composition iptables(for web server)

/etc/sysconfig/iptables
***filter**
:INPUT DROP [0:0]
:FORWARD DROP [0:0]
:OUTPUT ACCEPT [0:0]
-A INPUT -i lo -j ACCEPT
-A INPUT -d any_source_ip -p udp -m udp --dport 53 -j ACCEPT
-A INPUT -s any_source_ip -p udp -m udp --sport 53 -j ACCEPT
-A INPUT -d any_source_ip -p tcp -m tcp ! --tcp-flags SYN,RST,ACK SYN -j ACCEPT
-A INPUT -s any_source_ip -d destination_host_ip -p tcp -m tcp --dport 3306 -j DROP
-A INPUT -s admin_source_ip -d destination_host_ip -p tcp -m tcp --dport 22 -j ACCEPT
-A INPUT -s any_source_ip -d destination_host_ip -p tcp -m tcp --dport 80 -j ACCEPT
COMMIT

/etc/sysconfig/iptables example)
***mangle**
:PREROUTING ACCEPT [124:6512]
:OUTPUT ACCEPT [66:4305]
COMMIT
***nat**
:PREROUTING ACCEPT [0:0]
:POSTROUTING ACCEPT [1:132]
:OUTPUT ACCEPT [1:132]
-A POSTROUTING -o eth1 -j MASQUERADE
COMMIT
***filter**
:INPUT ACCEPT [124:6512]
:FORWARD ACCEPT [0:0]
:OUTPUT ACCEPT [66:4305]
COMMIT

Iptables start & policy listing

```
[root@www init.d]# ./iptables stop
$Resetting built-in chains to the default ACCEPT policy:   [  OK  ]
[root@www init.d]# ./iptables start
$Flushing all current rules and user defined chains:       [  OK  ]
$Clearing all current rules and user defined chains:       [  OK  ]
$Applying iptables firewall rules:                           [  OK  ]
                                                                  [  OK  ]
[root@www init.d]#
```

```
[root@www init.d]# iptables -L -n
Chain INPUT (policy DROP)
target     prot  opt  source               destination
ACCEPT     all   --   0.0.0.0/0            0.0.0.0/0
ACCEPT     tcp   --   0.0.0.0/0            0.0.0.0/0          tcp flags:!0x16/0x02
ACCEPT     udp   --   0.0.0.0/0            0.0.0.0/0          udp dpt:53
ACCEPT     udp   --   0.0.0.0/0            0.0.0.0/0          udp spt:53
DROP       tcp   --   0.0.0.0/0            0.0.0.0/0          tcp dpt:3306
ACCEPT     tcp   --   0.0.0.0/0            0.0.0.0/0          tcp dpt:22
[root@www init.d]#
```

/etc/init.d/iptables.sh for debian

```
#!/bin/sh -e
#
# init.d script for ipchains by Lenart Janos.
# modified for use with iptables by jordan@mjh.teddy-net.com
# changed by black@hackerslab.org <grep -v "^ *#" /etc/iptables.rule>
#
# This library is free software; you can redistribute it and/or
# modify it under the terms of the GNU Library General Public
# License as published by the Free Software Foundation; either
# version 2 of the License, or (at your option) any later version.
#
# This library 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
# Library General Public License for more details.
#
# $Id: iptables,v 1.10 2001/06/11 20:27:37 root Exp $
#
# leave this token to have it checked a maching script type
# MAGIC: echexeetifongah
#



self=`basename $0`
PATH=/usr/local/sbin:/usr/local/bin:/sbin:/bin:/usr/sbin:/usr/bin

# ------------------------------------------------------------------------
# Setting SAVEONCE to yes, the stop script won't cause the saving
# of the actual rules, only manual 'save' will do that.
# ------------------------------------------------------------------------
SAVEONCE=no

# Blurb string
SAVEFILEHEADER="#
# Automatically generated by '/etc/init.d/$self stop|save|store'
# from kernel's IP tables at '`date`' (last reboot?).
#
# Edit kernel's IP chains with the ipchains utility, and
# DO NOT EDIT THIS FILE. When you've finished you can run
# '/etc/init.d/self save' to update this file (if you like).
#
# If you want to restore your saved IP chains from this file
# execute '/etc/init.d/$self load'.
#"

# ------------------------------------------------------------------------
# rc installation settings
# ------------------------------------------------------------------------
```

```
# start & stop mode according to runlevel
Stag=10 ; Srunlevel_list="S 2 3"
Ktag=79 ; Krunlevel_list="0 6"

# where is the location of this script
source=/etc/fwb/bin/$self

# ----------------------------------------------------------------------
# sanilty check
# ----------------------------------------------------------------------
test -x /sbin/iptables || exit 0
test -x /sbin/iptables-restore || exit 0
test -x /sbin/iptables-save || exit 0

# ----------------------------------------------------------------------
# start/stop methods
# ----------------------------------------------------------------------

clear () {
    echo 0 >/proc/sys/net/ipv4/ip_forward
    iptables -L -n |awk '/Chain/ {printf "iptables -F %s\n", $2;}'|/bin/sh
    iptables -X
    iptables -P OUTPUT   ACCEPT
    iptables -P INPUT    ACCEPT
    iptables -P FORWARD ACCEPT
    iptables -F
    iptables -t nat -F
     echo -n " current-rules-flushed"
}

save () {
      echo -n "Saving IP tables:"
       TEMPFILE="`tempfile`"
      {
          echo "${SAVEFILEHEADER}"
           /sbin/iptables-save
```

```
            echo
          echo "# ip forwarding state saved by $0 on `date`"
            echo "# IPFORWARD=`cat /proc/sys/net/ipv4/ip_forward`"
      } >>"${TEMPFILE}"

      echo  -n " iptables-save"
      [  -f /etc/iptables.save ] &&
            mv /etc/iptables.save /etc/iptables.previous
      cat  "${TEMPFILE}"  >/etc/iptables.save
      rm  "${TEMPFILE}"
}

start () {
      echo  -n "Restoring IP tables: "
       clear
      grep  -v "^ *#" /etc/iptables.rule | /sbin/iptables-restore
      IPFORWARD=`awk  -F= '$1 ~ /^# *IPFORWARD$/{print $2;exit}' /etc/iptables.save`
      echo  ${IPFORWARD:-0} >/proc/sys/net/ipv4/ip_forward
      echo  -n " iptables Succeed !!!"
}

stop () {
    if [ "${SAVEONCE}" != "yes" ]
     then
            save
     else
            echo -n "Clearing IP tables: "
     fi
     clear
}

restart () {
    stop ; echo .
    start ; echo .
}
```

```
install () {
    # sanity check
    [ -f "$source" ] ||  {
        echo $self: No installation source \"$source\" - aborting
        exit 1
    }
    [ -h /etc/init.d/$self ] && rm -f /etc/init.d/$self
    [ -f /etc/init.d/$self ] && {
        echo $self: Cannot install rcfile $self - aborting
        exit 1
    }
    (set -x; ln -s $source /etc/init.d/$self)
    chmod +x                    /etc/init.d/$self
    # ----
    for n in $Srunlevel_list \# ; do (
        case $n in
            \#) continue ;;
        [0-9Ss]) RCD=rc$n.d ;;
            *) RCD=rc.$n
        esac
        ls -d                   /etc/$RCD >/dev/null 2>&1 || continue
        rm -f                     /etc/$RCD/[KS]??$self
        set -x
        ln -s ../init.d/$self /etc/$RCD/S$Stag$self
    ) done
    for n in $Krunlevel_list \# ; do (
        case $n in
            \#) continue ;;
        [0-9Ss]) RCD=rc$n.d ;;
            *) RCD=rc.$n
        esac
        ls -d                   /etc/$RCD >/dev/null 2>&1 || continue
        rm -f                     /etc/$RCD/[KS]??$self
        set -x
        ln -s ../init.d/$self /etc/$RCD/K$Ktag$self
    ) done
```

```
}

uninstall () {
    [  -h /etc/init.d/$self ] && (
          set  -x
          rm  -f /etc/init.d/$self
     )
    for n in $Srunlevel_list $Krunlevel_list \# ; do (
          case  $n  in
               \#) continue ;;
         [0-9Ss]) RCD=rc$n.d ;;
                *) RCD=rc.$n
           esac
          ls  -d /etc/$RCD/[KS]??$self >/dev/null 2>&1 || continue
          set  -x
          rm  -f /etc/$RCD/[KS]??$self
    ) done
}

# ---------------------------------------------------------------------------
# main
# ---------------------------------------------------------------------------

case $1 in
start|stop|save|restart|install|uninstall)
       $1
      echo  .
       ;;
load|restore|reload|force-reload)
       stert
      echo  .
       ;;
store)
       save
      echo  .
       ;;
```

```
*)
      echo "Usage: $0 {start|load|restore|reload|force-reload|stop|save|store|restart}" >&2
      exit 1
esac
exit 0
```

If you have any question contact to black@hackerslab.org
Thank you :)

http://www.hackerslab.org

http://www.netfilter.or.kr **GPL License**