De Linux Cipe+Masquerading mini-HOWTO

Anthony Ciaravalo, *acj@home.com*
Vertaler: *Reggy Ekkebus*, *reggy@zeelandnet.nl* v1.2, 21 april 1999

Het instellen van een VPN door het gebruik van Cipe op een linux masquerading firewall.

Inhoudsopgave

1 Introductie

Dit is de Linux Cipe+Masquerading mini-HOWTO. Het laat zien hoe je een Virtueel Prive netwerk kan maken tussen je LAN en andere Lan's door het gebruik van Cipe op linux masquerading firewall machines. Het laat ook een voorbeeld masquerading firewall configuratie zien.

1.1 Copyright

1.2 Disclaimer

Gebruik van de informatie en voorbeelden in dit document is op eigen risico. Er zijn veel security onderwerpen betrokken bij het verbinden over het het internet. Hoewel informatie versleuteld is, kan een niet goed geconfigureerde firewall resulteren in een inbraak. Voorzorgsmaatregelen kunnen worden genomen door het gebruik van cipe tussen twee machines, maar het is niet gegarandeerd 100% veilig. De auteur garandeert de informatie in dit document niet. Hoewel ik heel nauwkeurig ben geweest bij het schrijven van dit document, ben ik niet verantwoordelijk voor de problemen of schade opgelopen door acties gebaseerd op informatie in dit document.

1.3 Feedback

Zend vragen, commentaar, suggesties of correcties naar *acj@home.com*.

1.4 De files ophalen

Deze howto is gebaseerd op Cipe versie 1.0.1 en 1.2.0. Zie de referentie sectie voor een link naar de Cipe home page.

2 Firewall Configuratie

Deze howto veronderstelt dat je al kernel ondersteuning voor IP masquerading hebt. Zie de referentie voor informatie over hoe je je kernel moet configureren voor een linux firewall.

2.1 VPN Network Diagram

Deze setup gebruikt een star/hub configuratie. Het stelt een cipe connectie in van Machine A naar Machine B en een andere van Machine A naar Machine C.

```
                        Machine A
                     eth0: 192.168.1.1
                     eth1: echt ip 1
                    /               \
                   /                 \
          Machine B                  Machine C
        eth0: 192.168.2.1          eth0:192.168.3.1
        eth1: echt ip 2            eth1: echt ip 3
```

2.2 Een kleine referentie

```
     eth0 is het lokale netwerk (nep address)
     eth1 is het internet adres (echt address)

     Port A is elke geldige poort die je kiest
     Port B is elke andere geldige poort die je kiest

     Key A is elke goede sleutel die je kiest (lees cipe doc voor informatie)
     Key B is elke goede sleutel die je kiest
```

2.3 Toegevoegde notities over scripts en de VPN.

De ip-up scripts staan op het moment alleen klasse c verkeer toe door het cipe interface. Als je wilt dat machine B verbinding kan maken met Machine C dan moet je de bijbehorende ip-up en ip-down scripts aanpassen. Je moet de prpaddr en myaddr netmasks veranderen. Er zijn twee ip-up scripts, één voor ipchains en één voor ipfwadm. Hetzelfde geldt voor de ip-down scripts. Verander de desbetreffende inkomend, uitgaand en doorstuur cipe interface firewall rules netmask van /24 naar /16. Elke cipe firewall regel die je verandert in ip-up voor ipfwadm, moet je zeker weten dat je het ip-down script zo maakt dat het alles netjes van de lijst af haalt zodra het interface down gaat. Voor de ipchains file, alles wat je toegevoegd hebt, hoef je niet terug te vinden in de ip-down file, omdat het gewoon de hele gebruik gedefineerde chain verwijdert.

Je moet ook de netwerk route in de rc.cipe voor Machine B en C uncommenten dat voegt elkaars netwerk toe aan de route tabel.

3 Machine A Specifieke Configuratie

3.1 /etc/cipe/options.machineB

```
#uncomment er 1 hieronder
#naam voor cipe 1.0.x
#device          cip3b0
#naam voor cipe 1.2.x
device          cipcb0

# remote intern (nep) ip adres
ptpaddr         192.168.2.1
# mijn cipe (nep) ip adres
ipaddr          192.168.1.1
# mijn echte ip adres en cipe poort
me              (echt ip 1):(poort A)
# remote echt ip adres en cipe poort
peer            (echt ip 2):(poort A)
#unieke 128 bit sleutel
key             (Key A)
```

3.2 /etc/cipe/options.machineC

```
#uncomment er 1 hieronder
#naam voor cipe 1.0.x
#device          cip3b1
#naam voor cipe 1.2.x
device          cipcb1


# remote intern (nep) ip adres
ptpaddr         192.168.3.1
# mijn cipe (nep) ip adres
ipaddr          192.168.1.1
# my real ip adres and cipe port
me              (real ip 1):(port B)
# remote echt ip adres en cipe poort
peer            (real ip 3):(port B)
#unieke 128 bit sleutel
key             (Key B)
```

3.3 /etc/rc.d/rc.cipe

```
!#/bin/bash
#rc.cipe  3/29/1999
#Zend vragen of commentaar naar acj@home.com.

#Setup script path
PATH="/bin:/sbin:/usr/bin:/usr/sbin:/usr/local/bin:/usr/local/sbin"

#Optie file namen in cipe directory voor cipe interfaces
options="options.machineB options.machineC"

#Haal automatisch de optie filenamen uit de cipe directory
#options=`/bin/ls /etc/cipe/options.*`

#Uncomment er 1 hieronder voor de cipe module naam
#cipemod="cip3b"        #for cipe 1.0
cipemod="cipcb"        #for cipe 1.2

#Check voor cipe module en laad als dat nog niet gedaan is
grep $cipemod /proc/modules >/dev/null
if [ "$?" = "1" ]; then
        echo cipe module laden.
        modprobe $cipemod
        if [ "$?" = "1" ]; then
                echo Error bij het laden van de cipe module...exit
                exit
        fi
else
        echo Cipe module is al geladen.
fi

#Verwijder bestaande cipe interfaces
cipeif=`cat /proc/net/dev | cut -f1 -d: | grep $cipemod`

if [ "$cipeif" != "" ]; then
        echo Bestaande cipe interface(s) aan het verwijderen.
        for i in $cipeif; do
                ifconfig $i down
        done
fi

#Setup cipe interfaces
echo -n "De cipe interface(s) aan het instellen: "
for config in $options; do
        echo -n $config" "
        ciped -o $config
done
echo
echo

#Routes toevoegen voor andere remote netwerken via cipe interface(s)
#route add -net x.x.x.x netmask x.x.x.x gw x.x.x.x
```

3.4 Gateway

Alle machines op network 192.168.1.0 moeten 192.168.1.1 hebben als gateway. Als dat niet zo is zal het niet werken.

4 Machine B specifieke Configuratie

4.1 /etc/cipe/options.machineA

```
#uncomment er 1 hieronder
#naam voor cipe 1.0.x
#device          cip3b0
#naam voor cipe 1.2.x
device          cipcb0

# remote intern (nep) ip adres
ptpaddr         192.168.1.1
# mijn cipe (nep) ip adres
ipaddr          192.168.2.1
# mijn echte ip adres en cipe poort
me              (echt ip 1):(poort A)
# remote echt ip adres en cipe poort
peer            (echt ip 2):(poort A)
#unieke 128 bit sleutel
key             (Key A)
```

4.2 /etc/rc.d/rc.cipe

```
!#/bin/bash
#rc.cipe  3/29/1999
#Zend vragen of commentaar naar acj@home.com.

#Setup script path
PATH="/bin:/sbin:/usr/bin:/usr/sbin:/usr/local/bin:/usr/local/sbin"

#Optie file namen in cipe directory voor cipe interfaces
options="options.machineA"

#Haal automatisch de optie filenamen uit de cipe directory
#options=`/bin/ls /etc/cipe/options.*`

#Uncomment er 1 hieronder voor de cipe module naam
#cipemod="cip3b"        #for cipe 1.0
cipemod="cipcb"        #for cipe 1.2

#Check voor cipe module en laad als dat nog niet gedaan is
grep $cipemod /proc/modules >/dev/null
if [ "$?" = "1" ]; then
        echo cipe module laden.

        modprobe $cipemod
        if [ "$?" = "1" ]; then
```

```
                echo Error bij het laden van de cipe module...exit
                exit
        fi
else
        echo Cipe module is al geladen.
fi

#Verwijder bestaande cipe interfaces
cipeif=`cat /proc/net/dev | cut -f1 -d: | grep $cipemod`

if [ "$cipeif" != "" ]; then
        echo Bestaande cipe interface(s) aan het verwijderen.
        for i in $cipeif; do
                ifconfig $i down
        done
fi

#Setup cipe interfaces

echo -n "De cipe interface(s) aan het instellen: "
for config in $options; do
        echo -n $config" "
        ciped -o $config
done
echo
echo

#Routes toevoegen voor andere remote netwerken via cipe interface(s)
#route add -net x.x.x.x netmask x.x.x.x gw x.x.x.x

#route naar machine C network
#route add -net 192.168.3.0 netmask 255.255.255.0 gw 192.168.3.1
```

4.3 Gateway

Alle machines op het netwerk 192.168.2.0 moet 192.168.2.1 als gateway hebben. Anders zal het niet werken.

5 Machine C Specifieke Configuratie

5.1 /etc/cipe/options.machineA

```
#uncomment er 1 hieronder
#naam voor cipe 1.0.x
#device          cip3b0
#naam voor cipe 1.2.x
device          cipcb0

# remote intern (nep) ip adres
ptpaddr         192.168.1.1
# mijn cipe (nep) ip adres
ipaddr          192.168.3.1
# mijn echte ip adres en cipe poort
me              (echt ip 1):(poort A)
```

```
# remote echt ip adres en cipe poort
peer            (echt ip 2):(poort A)
#unieke 128 bit sleutel
key             (Key A)
```

5.2 /etc/rc.d/rc.cipe

```
!#/bin/bash
#rc.cipe  3/29/1999
#Zend vragen of commentaar naar acj@home.com.

#Setup script path
PATH="/bin:/sbin:/usr/bin:/usr/sbin:/usr/local/bin:/usr/local/sbin"

#Optie file namen in cipe directory voor cipe interfaces
options="options.machineA"

#Haal automatisch de optie filenamen uit de cipe directory
#options=`/bin/ls /etc/cipe/options.*`

#Uncomment er 1 hieronder voor de cipe module naam
#cipemod="cip3b"        #voor cipe 1.0
cipemod="cipcb"        #voor cipe 1.2

#Check voor cipe module en laad als dat nog niet gedaan is
grep $cipemod /proc/modules >/dev/null
if [ "$?" = "1" ]; then
        echo cipe module laden.
        modprobe $cipemod
        if [ "$?" = "1" ]; then
                echo Error bij het laden van de cipe module...exit
                exit
        fi
else
        echo Cipe module is al geladen.
fi

#Verwijder bestaande cipe interfaces
cipeif=`cat /proc/net/dev | cut -f1 -d: | grep $cipemod`

if [ "$cipeif" != "" ]; then
        echo Bestaande cipe interface(s) aan het verwijderen.
        for i in $cipeif; do
                ifconfig $i down
        done
fi

#Setup cipe interfaces
echo -n "De cipe interface(s) aan het instellen: "
for config in $options; do
        echo -n $config" "
        ciped -o $config
done
echo
```

```
echo

#Routes toevoegen voor andere remote netwerken via cipe interface(s)
#route add -net x.x.x.x netmask x.x.x.x gw x.x.x.x
#route naar machine B netwerk
#route add -net 192.168.2.0 netmask 255.255.255.0 gw 192.168.2.1
```

5.3 Gateway

Alle machines op het 192.168.3.0 netwerk moet 192.168.3.1 als gateway hebben. Anders werkt het niet.

6 Algemene Machine Configuratie

6.1 /etc/cipe/ip-up

6.1.1 Kernel 2.0, ipfwadm, cipe 1.0.x

```
#!/bin/bash
# ip-up <interface> <myaddr> <daemon-pid> <local> <remote> <arg>
#3/29/1999
#Een voorbeeld ip-up script voor de oudere 1.x 2.x kernels
#die ipfwadm gebruiken om de firewall en routes op te zetten,
#om je lokale class c netwerk te verbinden met een andere class c netwerk.

#De regels zijn geconfigureerd om spoofing en stuffed routing tussen de netwerken
#tegen te gaan. Er zijn extra security bevorderingen uit gecomment aan het
#einde van het script.
#Zend commentaar of vragen naar acj@home.com.

#------------------------------------------------------------------------
#Stel enkele script variabelen in.
device=$1               # het CIPE interface
me=$2                   # ons UDP address
pid=$3                  # het daemon proces ID
ipaddr=$4               # IP address van ons CIPE device
vptpaddr=$5             # IP address van het andere CIPE device
option=$6               # argument gegeven via opties

PATH="/bin:/sbin:/usr/bin:/usr/sbin:/usr/local/bin:/usr/local/sbin"

#comment/uncomment om kernel logging voor alle ongewenste toegangs pogingen
#aan/uit te zetten.
log="-o"

#------------------------------------------------------------------------
umask 022

# Een log voorbeeld
#echo "UP   $*" >> /var/adm/cipe.log

# Veel systeem willen de pid files
#echo $3 > /var/run/$device.pid
```

```
#------------------------------------------------------------------------

#voeg een route voor het andere cipe netwerk toe.
network=`expr $ptpaddr : '\([0-9]*\.[0-9]*\.[0-9]*\.\)'`0
route add -net $network netmask 255.255.255.0 dev $device

#moet een route voor de host toevoegen voor 2.0 kernels
route add -host $ptpaddr dev $device

#------------------------------------------------------------------------
#cipe interface inkomende firewall regels
#moet in de lijst gestopt worden in omgekeerde volgorde

#Weer alle andere inkomende pakketen naar het cipe interface
ipfwadm -I -i deny -W $device -S 0/0 -D 0/0 $log

#accepteer inkomende pakketen van remotenet naar localnet op het cipe interface
ipfwadm -I -i accept -W $device -S $ptpaddr/24 -D $ipaddr/24

#Accepteer inkomende pakketen van localnet naar remote net op het cipe interface
ipfwadm -I -i accept -W $device -S $ipaddr/24 -D $ptpaddr/24

#Weer inkomende pakketjes, cipe interface, claimend dat ze van het lokale net komen; log
ipfwadm -I -i deny -W $device -S $ipaddr/24 -D $ipaddr/24 $log

#------------------------------------------------------------------------
#cipe interface uitgaande firewall regels
#moet in de lijst gestopt worden in omgekeerde volgorde


#Weer alle uitgaande andere pakketjes van cipe interface
ipfwadm -O -i deny -W $device -S 0/0 -D 0/0 $log

#accepteer uitgaande pakketjes van het remotenet naar localnet op cipe interface
ipfwadm -O -i accept -W $device -S $ptpaddr/24 -D $ipaddr/24

#Accepteer uitgaande pakketen van localnet naar remote net op het cipe interface
ipfwadm -O -i accept -W $device -S $ipaddr/24 -D $ptpaddr/24

#Weer uitgaande pakketjes naar localnet van localnet, cipe interface; log
ipfwadm -O -i deny -W $device -S $ipaddr/24 -D $ipaddr/24 $log

#------------------------------------------------------------------------
#De forwarding is zo geconfigureerd zodat machines op je lokale netwerk niet worden
#gemaskeert naar het remote netwerk. Dit geeft betere toegangscontrole tussen
#de netwerken. moet in de lijst gestopt worden in omgekeerde volgorde

#Weer alle andere forwarding door het cipe interface; log
ipfwadm -F -i deny -W $device -S 0/0 -D 0/0 $log

#Accepteer forwarding van remotenet naar localnet op cipe interfaces
ipfwadm -F -i accept -W $device -S $ptpaddr/24 -D $ipaddr/24

#Accepteer forwarding van localnet naar remotenet op cipe interfaces
ipfwadm -F -i accept -W $device -S $ipaddr/24 -D $ptpaddr/24
```

```
#-------------------------------------------------------------------------
#Weet zeker dat forwarding in de kernel aanstaat. De kernel kan default forwarding
#uit hebben staan.
/bin/echo 1 > /proc/sys/net/ipv4/ip_forward

#-------------------------------------------------------------------------
#Optionele security verbeteringen - stel de standaard forwarding policy in op
#DENY of REJECT. Als je forwarding politiek DENY (Niet toelaten)/REJECT(afwijzen) is moet je de volgende
#regels toevoegen aan je standaard forward chain. Het is een goed idee om
#de default regel op DENY of REJECT te zetten.

#defineer machine interfaces
#localif="eth0"
#staticif="eth1"                ;cable modem gebruiker
#staticif="ppp0"                ;dialup gebruiker

#een erg slordige monier om het ip adres van de peer van de optie file te krijgen - een nieuw argement
# ip:port door gegeven aan het script zou erg handig zijn.
#beide regels moeten worden ge uncomment.
#peerfile=`grep $device /etc/cipe/options.* | cut -f1 -d:`
#peer=`grep peer $peerfile | cut -f1 -d: | awk '{print $2}'`

#moet peer ip adres loggen voor ip-down script
#echo $peer > /var/run/$device.peerip

#Accpteer forwarding van localnet naar remotenet op intern netwerk interface
#ipfwadm -F -i accept -W $localif -S $ipaddr/24 -D $ptpaddr/24

#Accpteer forwarding van remotenet naar localnet op intern netwerk interface
#ipfwadm -F -i accept -W $localif -S $ptpaddr/24 -D $ipaddr/24

#Accepteer forwarding op staticif van mij naar peer
#myaddr=`echo $me | cut -f1 -d:`
#ipfwadm -F -i accept -W $staticif -S $myaddr -D $peer

#-------------------------------------------------------------------------
#Andere optionele security verbeteringen
#Blok alle inkomende aanvragen van overal naar onze cipe udp
#poort behalve van onze peer udp poort

#Moet udp poort voor cipe interfaces vaststellen
#haal onze udp poort
#if [ "$option" = "" ]; then
#       myport=`echo $me | cut -f2 -d:`
#else
#       myport=$option
#fi

#haal remote udp poort -- peerfile variabel moet hierboven ingestelt zijn
#peerport=`grep peer $peerfile | cut -f2 -d:`

#moet peer udp poort loggen voor ip-down script
#echo $peerport > /var/run/$device.peerport
```

```
#haal ons ip adres
#myaddr=‘echo $me | cut -f1 -d:‘

#Verweer en log alle aanvragen op onze cipe udp poort, moet eerst worden ingestoken
#ipfwadm -I -i deny -P udp -W $staticif -S 0/0 -D $myaddr $myport $log

#Accepteer udp pakketen van peer op udp cipe poort naar mijn udp cipe poort
#ipfwadm -I -i accept -P udp -W $staticif -S $peer $peerport \
#-D $myaddr $myport

exit 0
```

6.1.2 Kernel 2.1/2.2, ipchains, cipe 1.2.x

```
#!/bin/bash
# ip-up <interface> <myaddr> <daemon-pid> <local> <remote> <arg>
#3/29/1999
#Een voorbeeld ip-up script voor de nieuwere 2.1/2.2 kernels die ipchains
#gebruiken om routes en firewall regels in te stellen om je lokale class c netwerk
#met een ander class c netwerk te verbinden. Dit script creererd 3 gebruiker gedefineerde
#chains -input, output, and forward - voor elke cipe interface, gebaseerd op
#de interface naam. Het stelt dan een regel in in de ingebouwde input, output, en forward chains
#om de gebruiker gedefineerde chains te gebruiken. De regels zijn geconfigureerd om
#spoofing en stuffed routing tussen de netwerken tegen te gaan. Er zijn ook optionele
#security verbeteringen uit gecomment aan het einde van het script.
#Zend vragen of commentaar naar acj@home.com.

#---------------------------------------------------------------------------
#Stel enkele script variabelen in.
device=$1               # het CIPE interface
me=$2                   # ons UDP address
pid=$3                  # het daemon proces ID
ipaddr=$4               # IP address van ons CIPE device
vptpaddr=$5             # IP address van het andere CIPE device
option=$6               # argument gegeven via opties


PATH="/bin:/sbin:/usr/bin:/usr/sbin:/usr/local/bin:/usr/local/sbin"

#comment/uncomment om kernel loggen van niet gewilde toegangs pogingen
#uit/aan te zetten. Moet het zelfde zijn als ip-down script om de regels te
#verwijderen.
log="-l"

#---------------------------------------------------------------------------
umask 022

# Een log voorbeeld
#echo "UP   $*" >> /var/adm/cipe.log

# Veel systeem willen de pid files
#echo $3 > /var/run/$device.pid

#---------------------------------------------------------------------------
#voeg een route toe voor het remote cipe netwerk
```

```
network=`expr $ptpaddr : '\([0-9]*\.[0-9]*\.[0-9]*\.\)'`0
route add -net $network netmask 255.255.255.0 dev $device

#-------------------------------------------------------------------------
#Creeer een nieuwe ipchain voor cipe interface input regels
ipchains -N $device"i"

#flush alle regels in die chain (sanity flush)
ipchains -F $device"i"

#Weer inkomende pakketen, cipe interface, claimend dat ze van localnet komen; log
ipchains -A $device"i" -j DENY -i $device -s $ipaddr/24 -d $ipaddr/24 $log

#Accepteer inkomende pakketjes van localnet naar remotenet op cipe interface
ipchains -A $device"i" -j ACCEPT -i $device -s $ipaddr/24 -d $ptpaddr/24

#Accepteer inkomende pakketjes van remotenet naar localnet op cipe interface
ipchains -A $device"i" -j ACCEPT -i $device -s $ptpaddr/24 -d $ipaddr/24

#Weer alle andere inkomende pakketjes
ipchains -A $device"i" -j DENY -s 0/0 -d 0/0 $log

#-------------------------------------------------------------------------
#Creeer een nieuwe ipchain voor cipe interface output regels
ipchains -N $device"o"

#flush alle regels in die chain (sanity flush)
ipchains -F $device"o"

#Weer uitgaande pakketen van localnet naar localnet, cipe interface; log
ipchains -A $device"o" -j DENY -i $device -s $ipaddr/24 -d $ipaddr/24 $log

#Accepteer uitgaande pakketten van localnet naar remotenet op cipe interface
ipchains -A $device"o" -j ACCEPT -i $device -s $ipaddr/24 -d $ptpaddr/24

#Accepteer uitgaande pakketjes van remotenet naar localnet op cipe interface
ipchains -A $device"o" -j ACCEPT -i $device -s $ptpaddr/24 -d $ipaddr/24

#Weer alle andere uitgaande pakketjes
ipchains -A $device"o" -j DENY -s 0/0 -d 0/0 $log

#-------------------------------------------------------------------------
#De forwarding is zo geconfigureerd zodat machines op je lokale netwerk niet worden
#gemaskeert naar het remote netwerk. Dit geeft betere toegangscontrole tussen
#de netwerken.

#Creeer een nieuwe ipchain voor cipe interface forward regels
ipchains -N $device"f"

#flush alle regels in die chain (sanity flush)
ipchains -F $device"f"

#Accepteer forwarding van localnet naar remotenet op cipe interface
ipchains -A $device"f" -j ACCEPT -i $device -s $ipaddr/24 -d $ptpaddr/24
```

```
#Accepteer forwarding van remotenet naar localnet op cipe interface
ipchains -A $device"f" -j ACCEPT -i $device -s $ptpaddr/24 -d $ipaddr/24

#Wijger alle andere fowarding; log
ipchains -A $device"f" -j DENY -s 0/0 -d 0/0 $log

#-------------------------------------------------------------------------
#Weet zeker dat forwarding in de kernel aan staat. Nieuwe kernels hebben standaard
#forwarding uit staan.
/bin/echo 1 > /proc/sys/net/ipv4/ip_forward

#-------------------------------------------------------------------------
#Voeg de regels toe aan de hoofd input, outpud en forward chains om de nieuwe regels
#te activeren voor het cipe interface
ipchains -I input -i $device -j $device"i"
ipchains -I output -i $device -j $device"o"
ipchains -I forward -i $device -j $device"f"

#-------------------------------------------------------------------------
#Optionele security verbeteringen - stel de standaard forwarding policy in op
#DENY of REJECT. Als je forwarding politiek DENY (Niet toelaten)/REJECT(afwijzen) is moet
#je de volgende regels toevoegen aan je standaard forward chain. Het is een goed idee om
#de default regel op DENY of REJECT te zetten.

#defineer machine interfaces
#localif="eth0"
#staticif="eth1"                ;cable modem gebruiker
#staticif="ppp0"                ;dialup gebruiker

#een erg slordige monier om het ip adres van de peer van de optie file te krijgen - een
#nieuw argement
# ip:port door gegeven aan het script zou erg handig zijn.
#beide regels moeten worden ge uncomment.
#peerfile=`grep $device /etc/cipe/options.* | cut -f1 -d:`
#peer=`grep peer $peerfile | cut -f1 -d: | awk '{print $2}'`

#moet peer ip adres loggen voor ip-down script
#echo $peer > /var/run/$device.peerip

#Accpteer forwarding van localnet naar remotenet op intern netwerk interface
#ipchains -I forward -j ACCEPT -i $localif -s $ipaddr/24 -d $ptpaddr/24

#Accpteer forwarding van remotenet naar localnet op intern netwerk interface
#ipchains -I forward -j ACCEPT -i $localif -s $ptpaddr/24 -d $ipaddr/24

#Accepteer forwarding op staticif van mij naar peer
#myaddr=`echo $me | cut -f1 -d:`
#ipchains -I forward -j ACCEPT -i $staticif -s $myaddr -d $peer

#-------------------------------------------------------------------------
#Andere optionele security verbeteringen
#Blok alle inkomende aanvragen van overal naar onze cipe udp
#poort behalve van onze peer udp poort
```

```
#Moet udp poort voor cipe interfaces vaststellen
#haal onze udp poort

#if [ "$option" = "" ]; then
#       myport=`echo $me | cut -f2 -d:`
#else
#       myport=$option
#fi

#haal remote udp poort -- peerfile variabel moet hierboven ingestelt zijn
#peerport=`grep peer $peerfile | cut -f2 -d:`

moet peer udp poort loggen voor ip-down script
#echo $peerport > /var/run/$device.peerport

#haal ons ip adres
#myaddr=`echo $me | cut -f1 -d:`

#Weer en log alle aanvragen op onze cipe udp poort, moet eerst worden ingestoken
#ipchains -I input -j DENY -p udp -i $staticif -s 0/0 \
#-d $myaddr $myport $log

#Accepteer udp pakketen van peer op udp cipe poort naar mijn udp cipe poort
#ipchains -I input -j ACCEPT -p udp -i $staticif -s $peer $peerport \
# -d $myaddr $myport

#------------------------------------------------------------------------
# Stel spoofing protectie in kernel in als optionele security maatregel
#------------------------------------------------------------------------
#Waarom heb ik hier spoofprotectie voor elk device in de kernel?
#Denk dat ik paranoide ben.

if [ -e /proc/sys/net/ipv4/conf/all/rp_filter ]; then
        echo -n "Ip spoof bescherming instellen..."
        iface="/proc/sys/net/ipv4/conf/$device/rp_filter"
        echo 1 > $iface
        echo "gedaan."
else
        echo "Kan spoof protectie in de kernel niet instellen voor $device" \
                | mail -s"Security Waarschuwing: $device" root
        exit 1
fi

exit 0
```

6.2 /etc/cipe/ip-down

6.2.1 Kernel 2.0, ipfwadm, cipe 1.0.x

```
#!/bin/bash

# ip-down <interface> <myaddr> <daemon-pid> <local> <remote> <arg>
#3/29/1999
#Een voorbeeld ip-down script voor de oudere 1.x 2.x kernels die ipfwadm gebruiken
#verwijderd de regels die ingesteld zijn om je klasse c netwerk met het andere klasse c netwerk
```

```
#te verbinden.

#--------------------------------------------------------------------------
#Stel enkele script variabelen in.
device=$1               # het CIPE interface
me=$2                   # ons UDP address
pid=$3                  # het daemon proces ID
ipaddr=$4               # IP address van ons CIPE device
vptpaddr=$5             # IP address van het andere CIPE device
option=$6               # argument gegeven via opties

PATH="/bin:/sbin:/usr/bin:/usr/sbin:/usr/local/bin:/usr/local/sbin"

#comment/uncomment om kernel logging voor alle ongewenste toegangs pogingen
#aan/uit te zetten. moet het zelfde zijn als in het ip-down script om de regels
#te verwijderen.
log="-o"

#--------------------------------------------------------------------------
umask 022

# Een log voorbeeld
#echo "UP   $*" >> /var/adm/cipe.log

# Verwijder de pid file
# rm -f /var/run/$device.pid

#--------------------------------------------------------------------------
#cipe interface inkomende firewall regels

#verwijder (Weer alle andere inkomende pakketen naar het cipe interface)
ipfwadm -I -d deny -W $device -S 0/0 -D 0/0 $log

#verwijder (accepteer inkomende pakketen van remotenet naar localnet op het cipe interface)
ipfwadm -I -d accept -W $device -S $ptpaddr/24 -D $ipaddr/24

#verwijder (accepteer inkomende pakketen van localnet naar remote net op het cipe interface)
ipfwadm -I -d accept -W $device -S $ipaddr/24 -D $ptpaddr/24

#verwijder (Weer inkomende pakketjes, cipe interface, claimend dat ze van het lokale net komen; log)
ipfwadm -I -d deny -W $device -S $ipaddr/24 -D $ipaddr/24 $log

#--------------------------------------------------------------------------
#cipe interface uitgaande firewall regels

#verwijder (Weer alle uitgaande andere pakketjes van cipe interface)
ipfwadm -O -d deny -W $device -S 0/0 -D 0/0 $log

#verwijder (accepteer uitgaande pakketjes van het remotenet naar localnet op cipe interface)
ipfwadm -O -d accept -W $device -S $ptpaddr/24 -D $ipaddr/24

#verwijder (accepteer uitgaande pakketen van localnet naar remote net op het cipe interface)
ipfwadm -O -d accept -W $device -S $ipaddr/24 -D $ptpaddr/24

#verwijder (verweer uitgaande pakketjes naar localnet van localnet, cipe interface; log)
```

```
ipfwadm -O -d deny -W $device -S $ipaddr/24 -D $ipaddr/24 $log

#-----------------------------------------------------------------------
#cipe interface forwarding firewall regels

#verwijder (Weer alle andere forwarding door het cipe interface; log)
ipfwadm -F -i deny -W $device -S 0/0 -D 0/0 $log

#verwijder (accepteer forwarding van remotenet naar localnet op cipe interfaces)
ipfwadm -F -i accept -W $device -S $ptpaddr/24 -D $ipaddr/24

#verwijder (accepteer forwarding van localnet naar remotenet op cipe interfaces)
ipfwadm -F -i accept -W $device -S $ipaddr/24 -D $ptpaddr/24

#-----------------------------------------------------------------------
#Optionele security verbeteringen - stel de standaard forwarding policy in op
#DENY of REJECT. Als je forwarding politiek DENY (Niet toelaten)/REJECT(afwijzen) is moet je de volgende
#regels toevoegen aan je standaard forward chain. Het is een goed idee om
#de default regel op DENY of REJECT te zetten.

#defineer machine interfaces
#localif="eth0"
#staticif="eth1"                ;cable modem gebruiker
#staticif="ppp0"                ;dialup gebruiker

#een erg slordige monier om het ip adres van de peer van de optie file te krijgen - een nieuw argement
# ip:port door gegeven aan het script zou erg handig zijn.
#beide regels moeten worden ge uncomment.
#peerfile=`grep $device /etc/cipe/options.* | cut -f1 -d:`
#peer=`grep peer $peerfile | cut -f1 -d: | awk '{print $2}'`

#moet peer ip adres loggen voor ip-down script
#echo $peer > /var/run/$device.peerip

#verwijder (accpteer forwarding van localnet naar remotenet op intern netwerk interface)
#ipfwadm -F -d accept -W $localif -S $ipaddr/24 -D $ptpaddr/24

#verwijder (accpteer forwarding van remotenet naar localnet op intern netwerk interface)
#ipfwadm -F -d accept -W $localif -S $ptpaddr/24 -D $ipaddr/24

#verwijder (accepteer forwarding op staticif van mij naar peer)
#myaddr=`echo $me | cut -f1 -d:`
#ipfwadm -F -d accept -W $staticif -S $myaddr -D $peer

#-----------------------------------------------------------------------
#Andere optionele security verbeteringen
#Blok alle inkomende aanvragen van overal naar onze cipe udp
#poort behalve van onze peer udp poort

#Moet udp poort voor cipe interfaces vaststellen
#haal onze udp poort
#if [ "$option" = "" ]; then
#       myport=`echo $me | cut -f2 -d:`
#else
#       myport=$option
```

```
#fi

#haal remote udp poort -- peerfile variabel moet hierboven ingestelt zijn
#peerport=`grep peer $peerfile | cut -f2 -d:`

#moet peer udp poort loggen voor ip-down script
#echo $peerport > /var/run/$device.peerport

#haal ons ip adres
#myaddr=`echo $me | cut -f1 -d:`

#verwijder (Weer en log alle aanvragen op onze cipe udp poort, moet eerst worden ingestoken)
#ipfwadm -I -d deny -P udp -W $staticif -S 0/0 -D $myaddr $myport $log

#verwijder (accepteer udp pakketen van peer op udp cipe poort naar mijn udp cipe poort)
#ipfwadm -I -d accept -P udp -W $staticif -S $peer $peerport \
#-D $myaddr $myport

exit 0
```

6.2.2 Kernel 2.1/2.2, ipchains, cipe 1.2.x

```
#!/bin/sh
# ip-down <interface> <myaddr> <daemon-pid> <local> <remote> <arg>
#3/29/1999
#Een voorbeeld ip-down script voor de nieuwere 2.1/2.2 kernels die
#ipchains gebruiken om de firewall regels te verwijderen die je hebt
#aangemaakt bij het verbinden van je klasse c netwerk met het andere
#klasse c netwerk. Optionele security verwijdering is ook uit gecomment aan
#einde van het script.
#Zend vragen of commentaar naar acj@home.com.

#--------------------------------------------------------------------------
#Stel enkele script variabelen in.
device=$1               # het CIPE interface
me=$2                   # ons UDP address
pid=$3                  # het daemon proces ID
ipaddr=$4               # IP address van ons CIPE device
vptpaddr=$5             # IP address van het andere CIPE device
option=$6               # argument gegeven via opties


PATH="/bin:/sbin:/usr/bin:/usr/sbin:/usr/local/bin:/usr/local/sbin"

#comment/uncomment om kernel loggen van niet gewilde toegangs pogingen
#uit/aan te zetten. Moet hetzelfde zijn als ip-down script om de regels te
#verwijderen.
log="-l"

#--------------------------------------------------------------------------
umask 022

# Een log voorbeeld
#echo "UP   $*" >> /var/adm/cipe.log
```

```
# Verwijder de pid file
# rm -f /var/run/$device.pid

#-----------------------------------------------------------------------
#verwijder regels van main input, output en foward chains voor cipe interface
ipchains -D input -i $device -j $device"i"
ipchains -D output -i $device -j $device"o"
ipchains -D forward -i $device -j $device"f"

#-----------------------------------------------------------------------
#flush alle regels in cipe interface input chain
ipchains -F $device"i"
#verwijder cipe interface input chain
ipchains -X $device"i"

#-----------------------------------------------------------------------
#flush alle regels in cipe interface output chain
ipchains -F $device"o"
#verwijder cipe interface output chain
ipchains -X $device"o"

#-----------------------------------------------------------------------
#flush alle regels in cipe interface forward chain
ipchains -F $device"f"
#verwijder cipe interface forward chain
ipchains -X $device"f"

#-----------------------------------------------------------------------
#Verwijder optionele security verbeteringen

#haal peer adres
#peer=`cat /var/run/$device.peerip`

#defineer machine interfaces
#localif="eth0"
#staticif="eth1"                ;cable modem gebruiker
#staticif="ppp0"                ;dialup gebruiker

#haal ons ip adres
#myaddr=`echo $me |cut -f1 -d:`

#verwijder (accepteer forwarding van localnet naar remotenet op intern netwerk
#interface)
#ipchains -D forward -j ACCEPT -i $localif -s $ipaddr/24 -d $ptpaddr/24

#verwijder (accepteer forwarding van remotenet naar localnet op intern netwerk
#interface)
#ipchains -D forward -j ACCEPT -i $localif -s $ptpaddr/24 -d $ipaddr/24

#verwijder (accepteer forwarding van staticif van mij naar peer)
#ipchains -D forward -j ACCEPT -i $staticif -s $myaddr -d $peer

#verwijder peer ip file
#rm /var/run/$device.peerip
```

```
#-----------------------------------------------------------------------
#Verwijder andere optionele security verbeterings regels

#haal peer udp port
#peerport=‘cat /var/run/$device.peerport‘

#haal onze udp port
#if [ "$option" = "" ]; then
#        myport=‘echo $me | cut -f2 -d:‘
#else
#        myport=$option
#fi

#verwijder (Weer en log alle aanvragen naar de cipe udp poort moet eerst ingestoken worden)
#ipchains -D input -j DENY -p udp -i $staticif -s 0/0 \
#-d $myaddr $myport $log

#verwijder (accepteer udp packets van peer op udp cipe poort naar mijn udp cipe poort)
#ipchains -D input -j ACCEPT -p udp -i $staticif -s $peer $peerport \
#-d $myaddr $myport

#verwijder peer poort file
#rm /var/run/$device.peerport

#-----------------------------------------------------------------------

exit 0
```

7 Voorbeeld masquerading firewall scripts

7.1 Kernel 2.0, ipfwadm

```
#!/bin/sh
#04/04/1999
#voorbeeld rc.firewall script voor de 2.0 kernels die ipfwadm gebruiken
#Ik kan geen vol vetrouwen geven voor dit script. Ik heb het een paar
#jaar geleden gevonden en heb wat aanpassingen gemaakt.
#Zend vragen of commentaar naar acj@home.com.

#--------------------------------------------------------------------
#Variabelen
#--------------------------------------------------------------------

#lokaal ethernet interface
localip=
localif=eth0

#statisch ethernet interface
staticip=
staticif=eth1

PATH="/bin:/sbin:/usr/bin:/usr/sbin:/usr/local/bin:/usr/local/sbin"

#--------------------------------------------------------------------
```

```
#Inkomende Firewall politiek
#-------------------------------------------------------------------
#flush inkomende firewall politiek
/sbin/ipfwadm -I -f

#stel inkomende firewall politiek standaard op deny (niet toelaten)
/sbin/ipfwadm -I -p deny

#-------------------------------------------------------------------

#lokaal interface, lokale machines, overal naar toegaand is toegestaan
/sbin/ipfwadm -I -a accept -V $localip -S $localip/24 -D 0.0.0.0/0

#remote interface, claimend dat het van het lokale netwerk komt (IP spoofing) verweren en loggen
/sbin/ipfwadm -I -a deny -V $staticip -S $localip/24 -D 0.0.0.0/0 -o

#remote interface, elke ,die gaan staticipen is goed
/sbin/ipfwadm -I -a accept -V $staticip -S 0.0.0.0/0 -D $staticip/32

#loopback interface is goed
/sbin/ipfwadm -I -a accept -V 127.0.0.1 -S 0.0.0.0/0 -D 0.0.0.0/0

#alle andere inkomende dingen worden gestopt en gelogd
/sbin/ipfwadm -I -a deny -S 0.0.0.0/0 -D 0.0.0.0/0 -o

#-------------------------------------------------------------------
#Outgaande Firewall politiek
#-------------------------------------------------------------------

#flush uitgaande firewall politiek
/sbin/ipfwadm -O -f

#stel uitgaande firewall policy in op 'niet toegestaan'
/sbin/ipfwadm -O -p deny

#-------------------------------------------------------------------

#lokaal interface, elke bron gaand naar local net is goed
/sbin/ipfwadm -O -a accept -V $localip -S 0.0.0.0/0 -D $localip/24

#uitgaand naar localnet op static interface, stuffed routing, niet toegestaan
/sbin/ipfwadm -O -a deny -V $staticip -S 0.0.0.0/0 -D $localip/24 -o

#uitgaand van lokaalnetwerk op static interface, stuffed masquerading, niet toegestaan
/sbin/ipfwadm -O -a deny -V $staticip -S $localip/24 -D 0.0.0.0/0 -o

#uitgaand naar lokaal netwerk op static interface, stuffed masquerading, niet toegestaan
/sbin/ipfwadm -O -a deny -V $staticip -S 0.0.0.0/0 -D $localip/24 -o

#elke andere uitgaande op remote interface is toegestaan
/sbin/ipfwadm -O -a accept -V $staticip -S $staticip/32 -D 0.0.0.0/0

#loopback interface is goed
/sbin/ipfwadm -O -a accept -V 127.0.0.1 -S 0.0.0.0/0 -D 0.0.0.0/0
```

```
#alle andere uitgaande pakketten zijn niet toegestaan en worden gelogd
/sbin/ipfwadm -O -a deny -S 0.0.0.0/0 -D 0.0.0.0/0 -o

#-----------------------------------------------------------------------
#Forwarding firewall politiek
#-----------------------------------------------------------------------

#flush forwarding politiek
/sbin/ipfwadm -F -f

#stel forwarding politiek standaard in op 'niet toegestaan'
/sbin/ipfwadm -F -p deny

#masquerade van localnet op local interface naar overal
/sbin/ipfwadm -F -a masquerade -W $staticif -S $localip/24 -D 0.0.0.0/0

#alle andere forwarding is niet toegestaan
/sbin/ipfwadm -F -a deny -S 0.0.0.0/0 -D 0.0.0.0/0

exit 0
```

7.2 Kernel 2.1/2.2, ipchains

```
#!/bin/sh
#04/04/1999
#voorbeeld rc.firewall script voor de nieuwere 2.1/2.2 kernels die ipchains
#gebruiken welke gebruik gedefineerde chains maakt voor elk interface.
#Er zijn firewall regels voor spoofing protectie welke onnodig zijn sinds
#de nieuwere kernels spoofing protectie aan kunnen hebben staan. Je kunt
#denken dat dit dan een beetje overbodig is.
#Zend vragen of commentaar naar acj@home.com.

#------------------------------------------------------------------
#Variabelen
#------------------------------------------------------------------

#lokaal ethernet interface
localip=
localif=eth0

#statisch ethernet interface
staticip=
staticif=eth1

#loopback interface
loopback=lo

PATH="/bin:/sbin:/usr/bin:/usr/sbin:/usr/local/bin:/usr/local/sbin"

#------------------------------------------------------------------
#Flush ingebouwde input, output, and forward ipchains; stel de standaard politiek
#in. Goed politiek is om alle pakketjes te wijgeren, vooral bij het instellen
#van chains
#------------------------------------------------------------------
```

```
#stel inkomende firewall politiek default op 'niet toegestaan'
ipchains -P input DENY

#flush inkomende firewall politiek
ipchains -F input

#---------------------------------------------------------------------

#stel uitgaande firewall politiek default op 'niet toegestaan'
ipchains -P output DENY

#flush uitgaande firewall politiek
ipchains -F output

#---------------------------------------------------------------------

#stel forwarding firewall politiek default op 'niet toegestaan'
ipchains -P forward DENY

#flush forwarding firewall politiek
ipchains -F forward

#---------------------------------------------------------------------
#flush alle politieken  -overbodig voor algemene politiek, maar flusht ook
#gebruik gedifineerde politieken
#ipchains -F

#Verwijder alle gebruiker gedefineerde politieken - je kan er voor kiezen om
#dit niet te doen
#ipchains -X

#---------------------------------------------------------------------
#Inkomende Firewall Politiek
#---------------------------------------------------------------------

#maak een nieuwe input chain voor static ethernet interface
ipchains -N $staticif"-i"

#flush all regels in chain (sanity flush)
ipchains -F $staticif"-i"

#blok inkomende SYN pakketjes op alle poorten op staticif en log
#dit kan een beetje grof zijn, maar het kan handig zijn in de toekomst
#ipchains -A $staticif"-i" -j DENY -p tcp -y -i $staticif -s 0/0 \
#-d $staticip : -l

#remote interface, claimend te komen van lokale machine (IP spoofing) niet toestaan en loggen
ipchains -A $staticif"-i" -j DENY -i $staticif -s $localip/16 -d 0/0 -l

#remote interface, elke bron, naar staticip adres is goed
ipchains -A $staticif"-i" -j ACCEPT -i $staticif -s 0/0 -d $staticip/32

#alle inkomende pakketjes worden afgewezen en gelogd
ipchains -A $staticif"-i" -j DENY -s 0/0 -d 0/0 -l
```

```
#--------------------------------------------------------------------

#maak een nieuwe input chain voor lokaal ethernet interface
ipchains -N $localif"-i"

#flush alle regels in chain (sanity flush)
ipchains -F $localif"-i"

#lokaal interface, lokale machines, gaand naar overal is goed
ipchains -A $localif"-i" -j ACCEPT -i $localif -s $localip/24 -d 0/0

#alle andere inkomende pakketjes worden afgewezen en gelogd
ipchains -A $localif"-i" -j DENY -s 0/0 -d 0/0 -l

#--------------------------------------------------------------------

#maak een nieuwe input chain voor loopback interface
ipchains -N $loopback"-i"

#flush alle regels in chain (sanity flush)
ipchains -F $loopback"-i"

#loopback interface is goed
ipchains -A $loopback"-i" -j ACCEPT -i $loopback -s 0/0 -d 0/0

#alle andere inkomende pakketjes worden afgewezen en gelogd
ipchains -A $loopback"-i" -j DENY -s 0/0 -d 0/0 -l

#-------------------------------------------------------------------------
#Forwarding firewall politiek
#-------------------------------------------------------------------------

#maak een nieuwe forward chain voor static ethernet interface
ipchains -N $staticif"-f"

#flush alle rules in chain (sanity flush)
ipchains -F $staticif"-f"

#masquerade van localnet op static interface naar overal
ipchains -A $staticif"-f" -j MASQ -i $staticif -s $localip/24 -d 0/0

#alle ander forwarding is verboden en wordt gelogd
ipchains -A $staticif"-f" -j DENY -s 0/0 -d 0/0 -l

#--------------------------------------------------------------------

#maak een nieuwe forward chain voor local ethernet interface
ipchains -N $localif"-f"

#flush alle regels in chain (sanity flush)
ipchains -F $localif"-f"

#alle ander forwarding is verboden en wordt gelogd
ipchains -A $localif"-f" -j DENY -s 0/0 -d 0/0 -l
```

```
#----------------------------------------------------------------------

#maak een nieuwe forward chain voor loopback interface
ipchains -N $loopback"-f"

#flush alle regels in chain (sanity flush)
ipchains -F $loopback"-f"

#alle ander forwarding is verboden en wordt gelogd
ipchains -A $loopback"-f" -j DENY -s 0/0 -d 0/0 -l


#----------------------------------------------------------------------
#Uitgaande Firewall Politiek
#----------------------------------------------------------------------

#maak een nieuwe output chain voor static ethernet interface
ipchains -N $staticif"-o"

#flush alle regels in chain (sanity flush)
ipchains -F $staticif"-o"

#uitgaand naar localnet op remote interface(stuffed routing) niet toegestaan & log
ipchains -A $staticif"-o" -j DENY -i $staticif -s 0/0 -d $localip/24 -l

#outgaand van local net op remote interface, stuffed masquerading, 'niet toestaan'
ipchains -A $staticif"-o" -j DENY -i $staticif -s $localip/24 -d 0/0 -l

#alle andere dingen op remote interface zijn goed
ipchains -A $staticif"-o" -j ACCEPT -i $staticif -s $staticip/32 -d 0/0

#alle andere uitgaande pakketjes zijn verboden en worden gelogd
ipchains -A $staticif"-o" -j DENY -s 0/0 -d 0/0 -l

#----------------------------------------------------------------------

#maak een nieuwe output chain voor local ethernet interface
ipchains -N $localif"-o"

#flush alle regels in chain (sanity flush)
ipchains -F $localif"-o"

#lokaal interface, elke bron gaand naar local net is toegestaan
ipchains -A $localif"-o" -j ACCEPT -i $localif -s 0/0 -d $localip/24

#alle andere uitgaande pakketjes zijn verboden en worden gelogd
ipchains -A $localif"-o" -j DENY -s 0/0 -d 0/0 -l

#----------------------------------------------------------------------

#maak een nieuwe output chain voor loopback interface
ipchains -N $loopback"-o"

#flush alle regels in chain (sanity flush)
ipchains -F $loopback"-o"
```

```
#loopback interface is toegestaan
ipchains -A $loopback"-o" -j ACCEPT -i $loopback -s 0/0 -d 0/0

#alle andere uitgaande pakketjes zijn verboden en worden gelogd
ipchains -A $loopback"-o" -j DENY -s 0/0 -d 0/0 -l

#-------------------------------------------------------------------------
#weet zeker dat forwarding in de kernel aan staat
#-------------------------------------------------------------------------

/bin/echo 1 > /proc/sys/net/ipv4/ip_forward

#-------------------------------------------------------------------------
#Voeg pointers toe aan ingebouwde chains om de gebruiker gedifineerde chains
#te activeren verander de volgorde om te optimaliseren voor een interface
#-------------------------------------------------------------------------

#voeg local interface input chain toe
ipchains -A input -i $localif -j $localif"-i"

#voeg static interface input chain toe
ipchains -A input -i $staticif -j $staticif"-i"

#voeg loopback interface input chain toe
ipchains -A input -i $loopback -j $loopback"-i"

#------------------------------------------------------------------------

#voeg local interface output chain toe
ipchains -A output -i $localif -j $localif"-o"

#voeg static interface output chain toe
ipchains -A output -i $staticif -j $staticif"-o"

#voeg loopback interface output chain toe
ipchains -A output -i $loopback -j $loopback"-o"

#------------------------------------------------------------------------

#voeg local interface forward chain toe
ipchains -A forward -i $localif -j $localif"-f"

#voeg static interface forward chain toe
ipchains -A forward -i $staticif -j $staticif"-f"

#voeg loopback interface forward chain toe
ipchains -A forward -i $loopback -j $loopback"-f"

#--------------------------------------------------------------------
#Super Paranoide check --- ook als staat de default politiek op niet toestaan
#blok alle pakketjes op elk interface
#--------------------------------------------------------------------

#alle andere inkomende pakketjes zijn niet toegestaan en worden gelogd
```

```
ipchains -A input -j DENY -s 0/0 -d 0/0 -l

#alle andere uitgaande pakketjes zijn niet toegestaan en worden gelogd
ipchains -A output -j DENY -s 0/0 -d 0/0 -l

#alle andere forwarding is niet toegestaan en worden gelogd
ipchains -A forward -j DENY -s 0/0 -d 0/0 -l

exit 0
```

8 Alles samen voegen

Dit is een voorbeeld rc.local script om alle te starten als je systeem opstart. Het voegt spoofing bescherming in de toe als je een 2.2 kernel hebt, stel masquerading firewall politiek in en start de cipe interface(s).

```
#!/bin/bash
#4/4/99
#een voorbeeld rc.local script
#Zend vragen of commentaar naar acj@home.com

echo

#Stel spoof protectie in de kernel is -- uit IPChains HOWTO door Paul Russell

#dit zijn alleen de nieuwere 2.1/2.2 kernels

#if [ -e /proc/sys/net/ipv4/conf/all/rp_filter ]; then
#   echo -n "Setting up IP spoofing protection..."
#   for f in /proc/sys/net/ipv4/conf/*/rp_filter; do
#       echo 1 > $f
#   done
#   echo "done."
#else
#   echo PROBLEMS SETTING UP IP SPOOFING PROTECTION.  BE WORRIED.
#   echo "CONTROL-D will exit from this shell and continue system startup."
#   echo
#   # Start a single user shell on the console
#   /sbin/sulogin $CONSOLE
#fi

echo

#Setup firewall policies
if [ -x /etc/rc.d/rc.firewall ]; then
        echo Setting up firewall packet filtering policies.
        echo
        . /etc/rc.d/rc.firewall
fi

#Start cipe interfaces
if [ -x /etc/rc.d/rc.cipe ]; then
        echo Starting VPN interfaces.
        . /etc/rc.d/rc.cipe
fi
```

```
    exit 0
```

9 Verbinding maken met de WAN

Nu moet je cipe interface draaiende zijn. Probeer te pingen naar machines op het andere netwerk(en). Als je niet kan pingen check dan het volgende op de firewall machine:

- Check of forwarding in de kernel aan staat.
- Doe een ifconfig om te kijken of je cipe interface up is.

```
cipcb0  Link encap:IPIP Tunnel  HWaddr
        inet addr:192.168.1.1  P-t-P:192.168.2.1  Mask:255.255.255.255
        UP POINTOPOINT NOTRAILERS RUNNING NOARP  MTU:1442  Metric:1
        RX packets:28163 errors:6 dropped:0 overruns:0 frame:6
        TX packets:29325 errors:0 dropped:0 overruns:0 carrier:0
        collisions:0 txqueuelen:100
```

- Check de route tabel voor een host invoer voor de andere cipe host op het cipe interface.

```
192.168.2.1    *      255.255.255.255 UH    0   0        0 cipcb0
```

- Check de route tabel voor een netwerk invoer voor het andere netwerk(en) op het cipe interface.

```
192.168.2.0    *      255.255.255.0   U     0   0        0  cipcb0
```

- Check de log files voor error berichten.

Als je andere machines achter je firewall geen toegang hebben tot machines achter de andere firewall kijk dan of je gateway goed is ingesteld op beide machines.

Zo snel je kunt pingen, ftp-en, telnetten, enz. naar de andere machines op het andere netwerk, dan is de volgende stap om de netwerken elkaar te laten zien om toegang te krijgen tot elkaars SAMBA browsen. Een aantal hints: lmhosts of wins servers zijn nodig, vertrouwde domeinen voor NT. Ik heb deze ingesteld maar dat is niet het doel van dit document (tot nu toe).

Als je het voorbeeld firewall masquerading script, dan zouden al je machines ook in staat moeten zijn om verbinding te maken met het internet. Als je dat niet kunt moet je de log files na kijken. Je kan ook tcpdump gebruiken om te zien wat er met de pakketjes gebeurt.

10 Referenties

10.1 Web Sites

Cipe Home Page `<http://sites.inka.de/~bigred/devel/cipe.html>`

Masq Home Page `<http://ipmasq.cjb.net>`

Samba Home Page `<http://samba.anu.edu.au>`

Linux HQ `<http://www.linuxhq.com>` —goede site met veel linux info

10.2 Documentatie

cipe.info: info file zit bij cipe distributie

Firewall HOWTO, door Mark Grennan, markg@netplus.net

IP Masquerade mini-HOWTO, door Ambrose Au, ambrose@writeme.com

IPChains-Howto, door Paul Russell, Paul.Russell@rustcorp.com.au