Setting up a Linux Home Gateway

Ramesh Panuganty

rameshpanuganty@myrealbox.com

This document is intended to help newbies to setup a linux based home gateway solution.

1. Introduction

1.1. Why am I interested in this document?

I played around several of the networking, security & firewall issues of linux systems for a company which I was working for. We had done a linux based satellite broadband access gateway aimed at SOHO and corporate IS departments. As a part of the setup done for enabling virtual private networks and interactive video-conferencing, I prepared this document (first version) for our internal use. I later thought of giving this out for public usage. Hope you appreciate the document.

1.2. Copyright

Copyright © 2001 Ramesh Panuganty. Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.1 or any later version published by the Free Software Foundation; with no Invariant Sections, with no Front-Cover Texts, and with no Back-Cover Texts. A copy of the license can be found at the Free Software Foundation (http://www.fsf.org/copyleft/fdl.html).

1.3. Disclaimer

The author does not accept any responsibility of any consequences that may result of the instructions in this document, in any manner. Use the examples and the other content at your risk. The author is also not responsible for any system outages or security issues that you may come across with your setup. Proceed with caution.

2. Theory

This document will try to help you in setting up a linux based home gateway. The procedure explained here will help you in getting a software based gateway instead of going for expensive hardware solutions.

At the end of the document, I will also try to briefly explain how you can secure your gateway from possible internet attacks.

The examples I would provided are based on Debian Linux - Woody distribution. However, it may be fairly easier if you are looking for other Linux distributions like RedHat, SuSe or Slackware.

After following the setup procedures you will be able to share your existing internet connection (PPP, DSL, cable modem or a T1 line) with other computers or devices in your home or office. Thereafter, your entire private network will be able to transparently access the internet with the single IP address.

3. Choices of implementation

3.1. Software Vs Hardware Solutions

There are several ways of achieving the home gateway solution and we selected the method of IP masquerading (Network Address Translation). This is the most simplest and optimum solution for a home gateway configuration. You may also do this with specialised kernels as done by Linux Router Project or Socks Gateway proxy project. There are other dedicated hardware solutions for the home gateway setup which come with pre-existing configurations.

3.2. Software Solutions in IP Masquerading

The IP Masquerading was done with ipfwadm in Linux 2.0 kernels, with ipchains in Linux 2.2 kernels and with iptables in Linux 2.4 kernels. There are obviously several advantages of using the newer versions due to the quality of support, improved implementations and enhanced configuration options. For example, ipchains introduced the concept of chains where as ipfwadm was based on a linear system, and is definitely much easier to administer. Ipchains also allows you to specify the IP protocol type and you may even replace or negate the rules.

Iptables is a stateful mechanism unlike the ipchains or ipfwadm. This allows you to track connections and permit incoming responses to outgoing requests without creating holes in the firewall. The connection tracking opens a specific, temporary hole for responses and only from the contacted server.

In this article, I will discuss only about ipchains and iptables. And I will try to explain all the tips & tricks for doing everything on your own.

4. Comparision of Alternatives

Linux 2.2 kernels will not have iptables (no official patches as yet). Though I will give examples for both ipchains and iptables, I would suggest you to start using iptables because of the greater flexibility and the reasons as explained above. With iptables, you can do stateful comparision of packets and also have enhanced set of rules. If you have a very old system with low memory (<8MB), suggest you to use only ipchains. Iptables keeps the states in RAM and may need a little more memory.

On 2.4 kernels, you may run either ipchains or iptables but not both at the same time. The rules specified by ipchains have precedence and the kernel even first attempts to load ipchains first. If you have ipchains modules installed, iptables may never start automatically.

If you decide to use iptables, you have a choice of several user interfaces for configuring your own firewall rules later-on. You may use knetfilter (GUI based), gshield, ferm, AGT or MM-Firewall. If you decide to use ipchains, you have a GUI tool gnome-lokkit to do the firewall configuration for your future firewall needs.

5. Hardware Requirements

If you are connecting to the ISP on ethernet, you need two ethernet cards on your gateway system. You may either install two single-port NIC cards in PCI slots, or select a dual-port NIC card. If you have a NIC slot built-in on your motherboard, you may go for a USB NIC device, instead of going for a second PCI slot. Choice is yours.

You need a NIC card one each on all your workstations (may be PCI, ISA, USB, wireless or built-int).

You need 10/100 network hub to connect all your workstations to the gateway system. Use a Cat5 cable for all connectivity reasons. If you are using uplink port on the hub, use a crossover cable and leave the adjacent normal port empty. Otherwise, if are using a normal port, use a straight-through network cable. If you have all similar cables avoid using the 'uplink' port on the network hub.

6. Kernel configuration

6.1. If using iptables

Iptables is actually a user interface tool and depends on the kernel implementation called netfilter (http://netfilter.samba.org/). Netfilter includes support only for IPv4 and IPv6, and does not filter any other protocols. Hence if your system should run something like IPX, remember that the protocols other than IPv4 and IPv5 are not going to be filtered according to the iptables rules. User kerne 2.4.18 or above, if possible to have all the new features of netfilter.

Install the iptables software on your system (apt-get install iptables). Once you know that your kernel is configured with netfilter support, you need not worry about it at all. Just remember that iptables need the kernel support from netfilter.

Check if your kernel is configured for supporting iptables. Though most distributions include this support by default, do this quick test as root.

```
bash# modprobe ip_tables
bash# lsmod | grep ip_tables
```

If any of the above commands give an error or ip_tables doesn't show up in module listing, you must enable these options in the kernel configuration using make menuconfig or make xmenuconfig

- Code maturity-level options for development and/or incomplete code/drivers
- Network packet filtering in Networking options.
- IP: Netfilter Configuration iin Networking options

select all these options as modules.

6.2. If using ipchains

Install the desired package on your system (apt-get install ipchains). Have these modules loaded using "insmod" in the startup script for iptables ip_tables, ip_nat_ftp and ip_conntrack_ftp. To check your system startup configuration for iptables or ipchains, cross check with rcconf on debian or chkconfig on RedHat. Select the appropriate module to load on startup. Be sure to select only one of the two.

For 2.4 kernels, follow the instructions as listed for iptables in the above section.

For 2.2 kernels,

- Select "Network Firewalls", "IP: defragment", "IP: masquerading", "IP: ICMP masquerading " and "IP: Firewalling", "IP: ipportfw masq support " and "IP: TCP syncookie support" in networking options.

7. Select your Private IP Addresses

To have your private network talk to each other and also for the home gateway setup to work, you need to assign IP addresses to all your system. Though you may assign any IP address to your private systems, you are not advised to do so. You are expected to use a specific set of IP addresses which have been reserved by IANA for private networks.

Table 1. Private IP Address assignments from RFC1918 (http://www.faqs.org/rfcs/rfc1918.html)

IP Address Class	IP Address From	IP Address To	CIDR Notation	Subnet Mask
Class A	10.0.0.0	10.255.255.255	10.0.0.0/8	255.0.0.0
Class B	172.16.0.0	172.31.255.255	172.16.0.0/16	255.255.0.0
Class C	192.168.0.0	192.168.255.255	192.168.0.0/24	255.255.255.0

8. Network Settings on Gateway

8.1. For PPP Connections

You will be assigned an IP address on connecting to your ISP over PPP and you need a ethernet connection to your private subnet. This means you need to configure your eth0 interface. For Debian users, update the file /etc/network/interfaces for IP configuration.

```
auto eth0
iface eth0 inet static
address 192.168.1.1
network 192.168.1.0
netmask 255.255.255.0
broadcast 192.168.1.255
```

You may also think of configuring the dial-on-demand, if you wish to automatically make a call whenever one of your other desktops tries to access internet. For enabling this feature, you would need to add options in /etc/ppp/options. You may also need to do "echo 1 > /proc/sys/net/ipv4/ip_dynaddr" to stop any program errors, till the connection is made.

Make sure that the default route is set to ppp0.

Restart the network settings on your system by

8.2. For Ethernet connections (cable-mode, DSL or T1)

In this mode, you need two ethernet interfaces on the linux gateway system, one to connect to internet and the other to connect to your private network hub. For example, if eth0 connects to the Internet and eth1 connects to the LAN, your network configuration should be set as (example for Debian Woody /etc/network/interfaces)

```
auto eth0
iface eth0 inet dhcp

auto eth1
iface eth1 inet static
address 192.168.1.1
network 192.168.1.0
netmask 255.255.255.0
broadcast 192.168.1.255
```

If you have a static IP address from your ISP, set the values for eth0 also similar to those of eth1 and change dhcp to static.

Restart the network settings on your system by

```
bash#  /etc/init.d/networking restart
```

9. IP Masquerade Configuration

9.1. For PPP Connections

Create the file /etc/gateway.rules with the following initial ruleset,

9.1.1. If using ipchains

```
/sbin/ipchains -M -S 7200 10 160
/sbin/ipchains -P input ACCEPT
/sbin/ipchains -P output ACCEPT
/sbin/ipchains -F input
/sbin/ipchains -F output
/sbin/ipchains -F forward
/sbin/ipchains -P forward DENY
/sbin/ipchains -A forward -i ppp0 -s 192.168.1.0/24 -j MASQ
```

9.1.2. If using iptables

```
/sbin/iptables -F
/sbin/iptables -t nat -F
/sbin/iptables -t mangle -F #ignore if you get an error here
/sbin/iptables -X #deletes every non-builtin chain in the table

/sbin/iptables -A INPUT -m state --state ESTABLISHED,RELATED -j ACCEPT
/sbin/iptables -A INPUT -m state --state NEW -i ! ppp0 -j ACCEPT
# only if both of the above rules succeed, use
/sbin/iptables -P INPUT DROP

/sbin/iptables -A FORWARD -i ppp0 -o eth0 -m state --state ESTABLISHED,RELATED -j ACCEPT
/sbin/iptables -A FORWARD -i eth0 -o ppp0 -j ACCEPT

/sbin/iptables -t nat -A POSTROUTING -o ppp0 -j MASQUERADE

/sbin/iptables -A FORWARD -i ppp0 -o ppp0 -j REJECT
```

9.2. For Ethernet connections (cable-mode, DSL or T1)

I am assuming that eth0 refers to the external interface and eth1 refers to the internal interface.

9.2.1. If using ipchains

For users connecting to external network on ethernet & using ipchains:

```
/sbin/ipchains -M -S 7200 10 160
/sbin/ipchains -P input ACCEPT
/sbin/ipchains -P output ACCEPT
/sbin/ipchains -F input
/sbin/ipchains -F output
/sbin/ipchains -F forward
/sbin/ipchains -P forward REJECT

# use this line if you have a dynamic IP address (on DHCP or BOOTP)
# configured from your ISP
/sbin/ipchains -A input -j ACCEPT -i eth0 -s 0/0 67 -d 0/0 68 -p udp

/sbin/ipchains -P forward DENY
/sbin/ipchains -A forward -i eth0 -s 192.168.1.0/24 -j MASQ
```

9.2.2. For iptables Users

For users connecting to external network on ethernet & using iptables:

```
/sbin/iptables -F
/sbin/iptables -t nat -F
/sbin/iptables -t mangle -F #ignore if you get an error here
/sbin/iptables -X #deletes every non-builtin chain in the table
```

```
/sbin/iptables -A INPUT -m state --state ESTABLISHED,RELATED -j ACCEPT
/sbin/iptables -A INPUT -m state --state NEW -i ! eth0 -j ACCEPT
# only if both of the above rules succeed, use
/sbin/iptables -P INPUT DROP

/sbin/iptables -A FORWARD -i eth0 -o eth1 -m state --state ESTABLISHED,RELATED -j ACCEPT
/sbin/iptables -A FORWARD -i eth1 -o eth0 -j ACCEPT

# use this line if you have a static IP address from your ISP
# replace your static IP with x.x.x.x
/sbin/iptables -t nat -A POSTROUTING -o eth0 -j SNAT --to x.x.x.x

# use this line only if you have dynamic IP address from your ISP
/sbin/iptables -t nat -A POSTROUTING -o eth0 -j MASQUERADE

/sbin/iptables -A FORWARD -i eth0 -o eth0 -j REJECT
```

10. Creating Startup Scripts

Create a script /etc/init.d/gateway

```
#! /bin/sh

# If no rules, do nothing.
[ -f /etc/gateway.rules ] || exit 0

case "$1" in
    start)
        echo -n "Turning on packet filtering:"

     /sbin/modprobe ip_masq_ftp #only if using ipchains
     /sbin/modprobe iptable_nat #only if using iptables
     /sbin/modprobe ipt_MASQUERADE #only if using iptables
        /sbin/ipchains-restore < /etc/ipchains.rules || exit 1

        echo 1 > /proc/sys/net/ipv4/ip_forward
     # for RedHat users, the above line is not needed if you have
     # FORWARD_IPV4=true in /etc/sysconfig/network file

     echo "1" > /proc/sys/net/ipv4/ip_dynaddr
     # the above option is for Dynamic IP users (DHCP,PPP or BOOTP)

        echo "."
        ;;
    stop)
        echo -n "Turning off packet filtering:"
        echo 0 > /proc/sys/net/ipv4/ip_forward
```

```
        /sbin/ipchains -F
        /sbin/ipchains -X
        /sbin/ipchains -P input ACCEPT
        /sbin/ipchains -P output ACCEPT
        /sbin/ipchains -P forward ACCEPT
        echo "."
        ;;
    *)
        echo "Usage: /etc/init.d/gateway {start|stop}"
        exit 1
        ;;
esac

exit 0
```

Give execute permissions to the startup scripts

```
bash# chmod 744 /etc/init.d/gateway
```

Use rcconf or chkconfig to create startup links for this file.

11. Network Settings on workstations

The values that need to be set on your workstations are,

```
static IP address   : 192.168.1.2 to 192.168.2.max-hosts
subnet Mask      : 255.255.255.0
Default gateway     : 192.168.1.1
Primary DNS Server  : <primary dns server as given by your ISP>
Secondary DNS Server: <secondary dns server as given by your ISP>
```

If you are linux clients, configure your network settings file. The file would be /etc/network/interfaces for Debian and /etc/sysconfig/network/ifcfg-eth0 for RedHat systems.

If you are using Windows clients, go to control panel, network settings, Properties, TCP-IP settings, static values to enter these values.

12. Activating the gateway

After doing all the settings as mentioned above, restart your networking with

```
bash# /etc/init.d/networking restart
```

Then Check for your routing table information for the default gateway. For users connecting with ethernet to the ISP, update your routing table entry for the gateway appropriately (as given by your ISP).

```
bash# route add -net 192.168.1.0 netmask 255.255.255.0 gw X.X.X.X dev eth1
```

13. How do I test the setup?

First ping your gateway system from one of your workstations. Then access a web site or an external system from your gateway and finally try to connect from your workstation directly.

If you can't go beyond your gateway, try printing your rules using "iptables -L" or "ipchains -L". See if they are okay as per the instructions. Check for other system settings and see if all the network interfaces are up in your workstations and look into their configurations. To view the established connections, do

```
bash# cat /proc/net/ip_conntrack
```

Some of the troubleshooting tips include checking for the ifconfig output on the home gateway. You should also see a loopback interface other than the ppp0-eth0/eth0-eth1. Check if network hub is powered-on and all the relevant lights are on in the network hub.

If you can ping the gateway's external IP address but not he hosts on internet, make sure you are not running 'gated' on your gateway.