Cabrillo College

CIS 192 Linux Lab Exercise

Lab 5: Firewalls and Network Address Translation (NAT)

Fall 2011

Lab 5: Firewalls and Network Address Translation (NAT)

The purpose of this lab is to exercise the use of iptables to build a permissive firewall by selectively filtering packets based on protocol type. It also demonstrates how addresses may be translated from private addresses to public and vice versa as they pass in and out of the firewall. The goal of this lab is to allow internet access to the hosts in Rivendell, and to allow hosts in the CIS Lab only telnet access, and no other, to a single server in Rivendell. Elrond will act as the gateway/firewall between Rivendell and the CIS Lab.

Supplies

- Virtualization: VMware ESXi/vSphere (for VLab) or Workstation (for CIS Lab PCs)
- Centos VMs: Elrond and Arwen
- Ubuntu VMs: Frodo and Sauron
- Virtual networks: Rivendell/VMnet3 and Mordor/VMnet4

Preparation

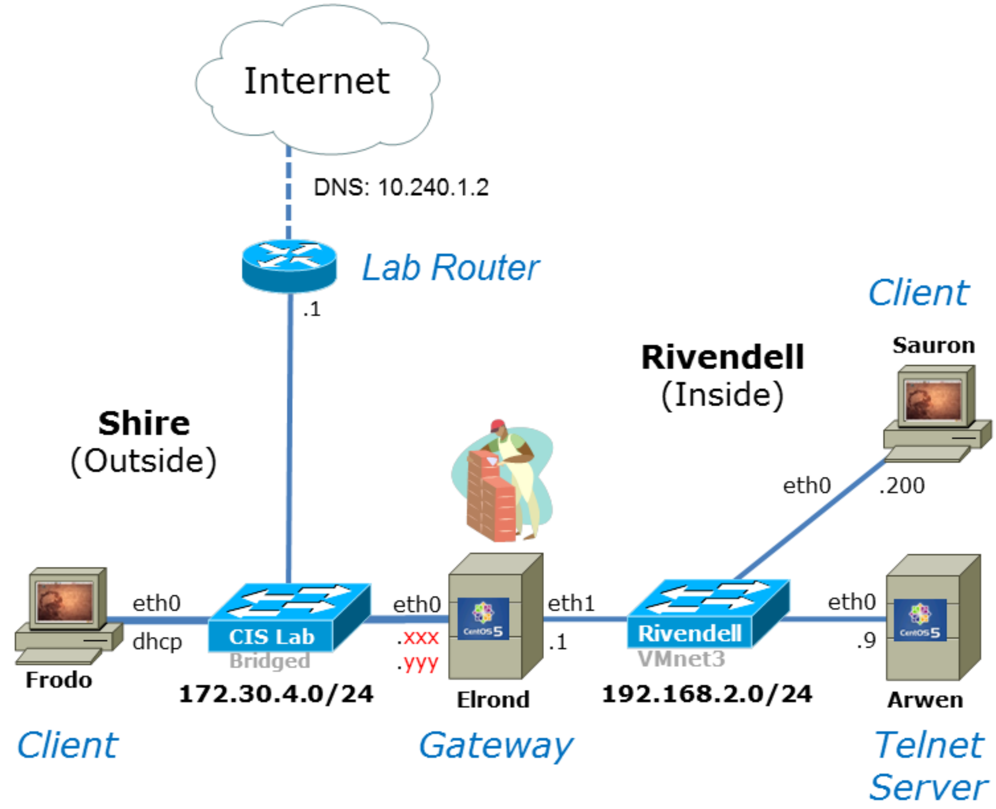
- Revert to the “Pristine” snapshot on all four VMs.
- On Opus, make a copy of the lab5 report template file in /home/cis192/depot in your home directory. Edit the header of this file with your own information and record all the information requested.

Forum

Use the forum to ask and answer questions, collaborate, and report any equipment issues. Post tips and any lessons learned when you have finished. Forum is at:
http://opus.cabrillo.edu/forum/viewforum.php?f=39

Background

Note that the setup shown below indicates that Elrond is the only host in Rivendell that will have access to the Internet. That is because Elrond has a network interface directly onto the 172.30.4.0 network. For the sake of this lab, we will treat the 172. IP addresses as if they were public and the 192. addresses as private. To the world outside of the firewall, your gateway provides the public address of 172.30.4.xxx.   The Rivendell telnet server will appear to have a public address of 172.30.4.yyy



Select static IP addresses .xxx and .yyy for Elrond from the static IP address table in the Appendix.

Setup

Build the diagram above using the lab VMs.

1. Install the telnet-server package on Arwen:
   - **rpm -qa | grep telnet-server** to see if it is already installed
   - If not installed:
     - Cable Arwen's first interface to the lab network and use **dhclient -v eth0** to join the network.
     - Use **yum install telnet-server** to install the service.
     - Release the IP address with **dhclient -r -v eth0**
     - Configure **/etc/xinetd.d/telnet** and modify **disable = yes** line to **disable = no**
     - Start the Telnet service with **service xinetd start**
2. Review the commands/files used in previous labs/lessons to configure permanent interface settings, IP aliases, DNS settings, and IP forwarding.
3. Cable and permanently configure the interfaces using the diagram above. Note .yyy on Elrond is an alias.
4. Configure permanent IP forwarding on Elrond.
5. Turn off NetworkManager on the Ubuntu VMs with **service network-manager stop** so any changes we make do not get undone.
6. The default routes on Rivendell hosts should be the gateway (Elrond).
7. Add IP/name pairs to /etc/hosts files so you can use hostnames in addition to IP addresses.

8. Add a static route on Frodo so it can reach Rivendell hosts.
9. Configure 10.240.1.2 as the DNS server for the Rivendell VMs.
10. Restart network services with **service network restart** (CentOS VMs) and **/etc/init.d/networking restart** (Ubuntu VM).
11. Verify that Elrond can ping the Lab Router, Frodo, Sauron and Arwen.
12. Verify Frodo can ping Arwen.

Part I

In this step, you will disable the firewall on Elrond and open port 23 on Arwen.

1. On Elrond,
    a. Make a backup of the current firewall rules:
        cp /etc/sysconfig/iptables /etc/sysconfig/iptables.bak
    b. Disable the firewall by:
        iptables -F
        service iptables save
    c. Verify that the rules are flushed with ACCEPT as the policy
        iptables -L
2. On Arwen,
    a. show the firewall with **iptables -nL --line-numbers**
    b. Determine the line number, **n**, of the final "REJECT all" rule on the INPUT chain.
    c. On Arwen, open port 23 for incoming new Telnet connections by inserting a new rule at line **n**:
        iptables -I INPUT n -m state --state NEW -m tcp -p tcp --dport 23 -j ACCEPT
    d. Verify changes with the **iptables -nL** command. Both port 22 (SSH) and 23 (Telnet) should accept new connections.
    e. Make the firewall permanent with **service iptables save**
3. On Elrond, telnet into Arwen,
    a. Install telnet client with **yum install telnet**
    b. Telnet to Arwen with **telnet 192.168.2.9**
        (be patient, it may take some time before you see the login prompt)
    c. Login as cis192
    d. Use **exit** to end the session and get back to Elrond.

Part II

In this section we will filter out all packets to, from, and through Elrond's firewall, thus isolating the Rivendell network.

On Elrond,

1. List the current firewall settings and note the default policies:
    iptables -L
2. Now set the default policy on all three chains in the filter table to DROP:
    iptables -P INPUT DROP
    iptables -P FORWARD DROP
    iptables -P OUTPUT DROP
3. Verify the new policies with:
    iptables -L
4. Verify that no network traffic can enter, leave or pass through the firewall by:

- From the Telnet server (Arwen): **ping 192.168.2.1**
- From the CIS Lab client (Frodo): **ping 172.30.4.xxx**
- From the Gateway (Elrond): **ping 172.30.4.1** (the lab router)

Note these same pings worked earlier before setting the filter chain policies to DROP.

Part III

Now we will configure Elrond's firewall. Since we want to allow outside hosts to use our Telnet server, will allow only Telnet packets to be forwarded through our firewall from the outside world. In addition we will allow all packets generated within Rivendell to be forwarded to the outside world.

1. FORWARD chain: Allow all necessary packets supporting established connections to pass through:
   iptables -A FORWARD -m state --state ESTABLISHED,RELATED -j ACCEPT
2. FORWARD chain: Allow new connections initiated from inside our firewall to propagate through the firewall to the outside world:
   iptables -A FORWARD -s 192.168.2.0/24 -d 0/0 -m state --state NEW -j ACCEPT
3. FORWARD chain: Allow packets from the outside destined for our Telnet server to pass through the firewall:
   iptables -A FORWARD -s 0/0 -d 192.168.2.9 -m state --state NEW -p tcp --dport 23 -j ACCEPT *(all on one line)*
4. OUTPUT chain: For completeness we should also allow our firewall to output packets:
   iptables -A OUTPUT -m state --state NEW,ESTABLISHED,RELATED -j ACCEPT
5. INPUT chain: Allow return traffic from any connections initiated on Elrond and accept any new incoming connections from our internal Rivendell network:
   iptables -A INPUT -m state --state ESTABLISHED,RELATED -j ACCEPT
   iptables -A INPUT -i eth1 -s 192.168.2.0/24 -d 192.168.2.1 -m state --state NEW -j ACCEPT
6. Verify that Arwen and Sauron can now ping the firewall.
7. Verify that Frodo can telnet into Arwen but not ping Arwen or Elrond.

Part IV

Now we will provide NAT (Network Address Translation) to allow all hosts within Rivendell to access the Internet, and allow all hosts outside the firewall to access our Telnet server through a "public" address of 172.30.4.yyy

1. Allow any packets destined to 172.30.4.yyy to be translated to 192.168.2.9
   iptables -t nat -A PREROUTING -i eth0 -d 172.30.4.yyy -j DNAT --to-destination 192.168.2.9
2. Now allow for the translation of packets from our Telnet server to this pseudo-public address:
   iptables -t nat -A POSTROUTING -o eth0 -s 192.168.2.9 -j SNAT --to-source 172.30.4.yyy
3. And finally, allow all other hosts in Rivendell to have their private addresses translated to the public address of our firewall:
   iptables -t nat -A POSTROUTING -o eth0 -s 192.168.2.0/24 -j SNAT --to-source 172.30.4.xxx
4. Verify that any Rivendell host can now surf the Internet, and that any CIS Lab host can access the Telnet server via the public address of 172.30.4.yyy. Note that you can't ping the Telnet server from that same host in the CIS Lab.
5. Save your firewall with **service iptables save**

Part V

Part of maintaining a secure firewall is monitoring attempts to contact or pass through the firewall. This may be done using the LOG action on the firewall.

1. Add the following line near the top of the RULES section in */etc/rsyslog.conf* file:
   ```
   kern.info                              /var/log/iptables
   ```
2. Create this log file in the var/log directory:
   ```
   > /var/log/iptables
   ```
3. Restart the system logging daemon:
   ```
   service rsyslog restart
   ```
4. Add the following two lines to the filter table:
   ```
   iptables -A INPUT -j LOG --log-level info  --log-prefix "iptables INPUT: "
   iptables -A FORWARD -j LOG --log-level info  --log-prefix "iptables FORWARD: "
   ```
5. To view the entries added to the log file, run the following command on your Gateway while you ping or otherwise try to attack Rivendell from the CIS Lab network:
   ```
   tail -f /var/log/iptables
   ```
   See if you can collect both log types, input and forward.

   When you are finished viewing the log activity, use Ctrl-C to break out of the **tail** command.
6. Make your new firewall permanent with:
   ```
   service iptables save
   ```

Congratulations! You have created a secure network in Rivendell with all machines having access to the Internet!

To turn in

Record the following in your lab05 report:

1. On Frodo, use the script command or Copy & Paste to record a telnet login session from Frodo to Arwen via Elrond.
2. On Sauron, record the route to Opus with output from:
   ```
   mtr -c2 --report  opus.cabrillo.edu
   ```
3. On Arwen, record your telnet and firewall configuration with output from :
   ```
   cat /etc/xinetd.d/telnet
   cat /etc/sysconfig/iptables
   ```
4. On Elrond, record your interfaces, firewall/NAT rules, and iptables log with output from:
   ```
   ifconfig
   cat /etc/sysconfig/iptables
   cat /var/log/iptables
   ```

Check your work for completeness then submit as many times as you wish up until the due date deadline. Remember, **late work is not accepted**, so start early, plan ahead for things to go wrong and use the forum to ask questions.

```
cp lab05 /home/rsimms/turnin/lab05.$LOGNAME
```

Grading rubric (30 points)

2 points for complete submittal to the turnin directory
2 points for complete header including time spent and station info
2 points for unique IP addresses for Elrond’s eth0 interface

2 points for correct trace to Opus on Sauron
2 points for correct telnet login to Arwen from Frodo

4 points for correctly configuring telnet on Arwen
4 points for correctly configuring firewall on Arwen

4 points for correctly configuring the Elrond firewall rules
4 points for correctly configuring Elrond NAT rules
2 points for input log entries on Elrond caused by Frodo
2 points for forward log entries on Elrond caused by Frodo