System Administration and IP Services

TCP/IP Networking Exercises

June 9, 2013

1. Check your network configuration

Use these commands to list all the network interfaces on your machine, and to show the configuration of one specific interface.

```
$ ifconfig -a
$ ifconfig eth0
```

Do you see an IP address on your network card? It should look like this:

```
eth0      Link encap:Ethernet  HWaddr 52:54:8e:12:66:49
          inet addr:10.10.0.x  Bcast:10.10.0.255  Mask:255.255.255.0
```

"inet addr" is your machine's IP address, and "HWaddr" is your machine's hardware address (MAC address).

If you needed to configure your network card, these would be the commands:

```
$ sudo ifconfig eth0 10.10.0.x/24
$ sudo route add default gw 10.10.0.254
```

However, as you are already connected over ssh don't do this or you may end up breaking your network connection to your machine.

2. netstat

Look at your forwarding table:

```
$ netstat -rn
```

What do you notice? Is the default gateway configured? How do you know? Review the presentation if you are not sure. What is your default gateway? On what network interface is your default gateway valid for?

Here's another way to look at your forwarding table:

```
$ ip route
```

3. ping

Let's ping the default gateway:

```
$ ping 10.10.0.254
```

(Stop it with CTRL+C)

Let's ping something outside, on the Internet. For example, nsrc.org

```
$ ping nsrc.org
```

Do you get an answer ?

If not, check:

- That you have a gateway configured
- That in the file /etc/resolv.conf there is an entry for "nameserver"
- Do you notice anything about the response time? How far away is nsrc.org?

Verify 10.10.0.254 is configured as your default gateway:

```
$ netstat -rn
```

Now, remove your default gateway:

```
$ sudo route delete default
```

Check that it's gone

```
$ netstat -rn
```

How can you be sure that the default gateway is no longer configured? Now, try to ping the local NOC machine.

```
$ ping 10.10.0.250
```

Now let's ping a machine outside our network (nsrc.org):

```
$ ping nsrc.org
```

The ip address of nsrc.org is 128.223.157.19

```
$ ping 128.223.157.19
```

What do you observe?

What is the consequence of removing the default gateway?

Re-establish the default gateway:

```
 $ sudo route add default gw 10.10.0.254
```

Check that the default gateway is enabled again by pinging nsrc.org:

```
 $ ping nsrc.org
```

4. traceroute

Traceroute to nsrc.org

```
$ traceroute nsrc.org
```

Try again, this time with the -n option:

```
$ traceroute -n nsrc.org
```

Observe the difference with and without the '-n' option. Do you know what it is? If you are trying to debug a network problem, which might be preferable?

5. tcpdump

Run tcpdump on your system:

```
$ sudo tcpdump -n -i eth0 icmp
```

(Note the use of the "icmp" keyword to limit viewing ICMP traffic)

Ask the instructor(s) or your neighbor to ping your machine, and look at your screen.

Repeat the exercise, this time using flag "-e" to show the layer 2 (ethernet) headers on each packet.

```
 $ sudo tcpdump -n -e -i eth0 icmp
```

Can you see the source and destination MAC addresses in each frame?

Repeat again, this time showing verbose decode (-v) and full hex dump (-X) and capturing all 1500 bytes of each packet (-s1500)

```
 $ sudo tcpdump -nvX -s1500 -i eth0 icmp
```

If you have time, read "man pcap-filter" to find out about the tcpdump filtering language, and try some other filters to include or exclude particular types of traffic. e.g.

```
 $ sudo tcpdump -n -i eth0 'not tcp port 22'
```

If you have a Windows laptop you can install Wireshark to see traffic going in and out of your network interfaces.

6. arp

The ARP table remembers which machine's MAC address owns each IP address on your local network. Examine it like this:

```
$ arp -an
```

Based on the commands you've seen before, what do you think the "-n" flag does? To see if you're right, try it without:

```
$ arp -a
```

Try sending some test ARP packets using a tool called “arping”:

```
$ sudo arping -I eth0 10.10.0.254
```

If the command doesn’t exist then install the package like this:

```
$ sudo apt-get install arping
```

arping is particularly useful for debugging problems where two machines have been configured with the same IP address, because you will see the ARP responses from both.