Useful UNIX Tools

This document describes the development tools available on the networking lab machines, as well as tools that you can use to explore the local network and the Internet. Your code will be expected to compile with these tools and work on the lab machines. If you decide to do development somewhere else, make sure you use the same versions of the tools. This is especially important in the case of GCC, especially if you are using C++. All of these tools are included with most Linux distributions or are quite easy to install.

Help

The "man" command provides access to manual pages. "man gcc" will show you the man page for the gcc compiler. "man –k packet" will find man pages that have something to do with packets. "man man" will tell you how to use man. Some help in Linux is in GNU's texinfo format. Run "info" to see all the installed info documents. You can also via man and info pages graphically using "gnome-help-browser".

Editors

An editor such as XEmacs ("xemacs"), GNU Emacs ("emacs"), vi, or others is essential. All are installed on the lab machines. Both Emacs editors integrate with make, GCC, and GDB to form a relatively powerful integrated development environment.

GNU Make

Make is the core tool for build management in Minet and many other programs. Make is covered in a separate document, and the GNU Make manual is available from the web page.

CVS

CVS is a very useful tool when working in groups. It allows each team member to have a local copy of the code while still managing a global version. This enables integration and coordination when people are modifying the same file.

GCC Development Tools

The lab machines use GCC 4.4.4. If you use another machine, make sure that your code compiles with this compiler version. Any GCC 4.x.x version should work, but be prepared for surprises. The lab machines also include GDB.

Promiscuous mode and the Berkeley Packet Filter

Your kernel must be able to run the Ethernet card in promiscuous mode and it must have Berkeley Packet Filter support enabled. Most Linux kernels permit this if you are root or you are running a binary that is suid root or via sudo. On the lab machines, we have set up things up so that you'll be able to run Minet and the other various tools described below.

Libpcap

Libpcap is an interface that program can use to access an Ethernet card running in promiscuous mode. Make sure you have version 0.4 or later. To find out your version, you can use rpm. You should see the following output:

```
$ rpm -q -a | grep libpcap
libpcap-0.6.2-12
```

Libnet

Libnet is an interface for injecting raw Ethernet packets into the network. You should have version 1.1 or later.

Tcpdump and Wireshark

Tcpdump is a program for printing the traffic that your Ethernet card sees in a human readable form. If you are running the Ethernet card in promiscuous mode, then you can see all of the traffic that passes by. Running tcpdump without arguments will spew all the traffic that can be seen. You can supply tcpdump with a packet filter to limit what is printed to what you are specifically interested in. For example,

```
$ /usr/sbin/tcpdump src scratchy and tcp
```

will show you TCP traffic originating from the host "scratchy."

Wireshark is a graphical version of tcpdump that provides an easy to use interface as well as advanced inspection and filtering features. You will find these programs essential to debugging your code.

Ifconfig

/sbin/ifconfig –a gives you information about the interfaces in a machine.

Route

/sbin/route will show you the routing table of the machine. The Minet projects will be configured to operate on the private lab network, consisting of the 192.168.0.0 subnet.

Ping

Ping is a program that you can use to see if the TCP/IP stack on a remote machine is actually functioning, provided that the stack supports ICMP. Minet is pingable.

Traceroute

Traceroute lets you discover the path that an IP packet takes from your local machine to some remote host, provided the intervening routers support ICMP. For example, here is the route from one Northwestern machine to www.wisc.edu:

```
$ /usr/sbin/traceroute www.wisc.edu
traceroute to www.wisc.edu (144.92.104.37), 30 hops max, 38 byte packets
 1  birl-idf-eth-3.nwu.edu (129.105.100.170)  0.818 ms  0.715 ms  0.729 ms
 2  lev-mdf-5-gig-3-0-1.nwu.edu (129.105.253.53)  0.958 ms  0.874 ms  0.795 ms
 3  lev-mdf-rtr-1.nwu.edu (129.105.253.238)  1.556 ms  1.290 ms  1.281 ms
 4  lev-mdf-rtr-2.nwu.edu (199.249.169.65)  1.941 ms  1.984 ms  2.804 ms
 5  206.220.243.35 (206.220.243.35)  8.204 ms  7.681 ms  7.440 ms
 6  r-macc.net.wisc.edu (144.92.128.129)  7.977 ms  8.865 ms  8.772 ms
 7  gopher.adp.wisc.edu (144.92.104.37)  9.245 ms  8.048 ms  9.038 ms
```

Netcat

Netcat, or "nc", is a tool for sending or receiving data using UDP or TCP. It is very handy for testing servers and things like the Minet protocol stack.

Nslookup, Dig, and Whois

Nslookup is a command-line and interactive interface to the DNS system. You can use it to find mappings of hostnames to IP address. Dig is similar in principle, but it presents much more detailed information from DNS. It is very useful in figuring out how various name server tricks are played. For fun, you can use dig to figure out how Akamai caching works. Whois allows you to query for detailed information on top-level domains. For example, you can use whois to find out who owns flibbertygibbet.com.