|  |  |  |
| --- | --- | --- |
| Bochs User Manual | | |
| [Prev](http://docs.google.com/smp.html) | Chapter 8. Tips and Techniques | [Next](http://docs.google.com/config-tuntap.html) |

8.10. Setting Up Networking in DLX Linux

All Bochs binaries come with a working version of DLX Linux. This section describes how to configure networking in Bochs and enable it within DLX Linux. First you must add a ne2k line in your bochsrc file. Then, when you boot the DLX Linux image, you need to type some Linux commands to set up an IP address, a network route, and a name server.

When you have an ne2k line in your bochsrc file, Bochs will emulate a network card called an NE2000. Below are some examples of valid ne2k lines for various operating systems. Choose the one that's closest to what you need, add it to your bochsrc file, and edit the values if necessary.

# sample for Mac OS X  
 ne2k: ioaddr=0x300, irq=9, mac=b0:c4:20:00:00:00, ethmod=fbsd, ethdev=en0  
 # sample for FreeBSD  
 ne2k: ioaddr=0x300, irq=9, mac=b0:c4:20:00:00:00, ethmod=fbsd, ethdev=xl0  
 # sample for Linux  
 ne2k: ioaddr=0x300, irq=9, mac=b0:c4:20:00:00:00, ethmod=linux, ethdev=eth0  
 # sample for Windows  
 ne2k: ioaddr=0x300, irq=9, mac=00:c4:3B:00:C3:00, ethmod=win32, ethdev=NE2000

You see the pattern. Usually you won't need to change the I/O address, IRQ number, or MAC address. The ethmod value depends on your host operating system, and it must be either null, fbsd (for FreeBSD or OpenBSD), linux, or win32. The ethdev setting is the name of the network interface on your system, and is also OS-dependent. On UNIX systems you can get the name of the network interface by running **ifconfig**. (Don't choose the loopback interface.) On Windows systems, the correct ethdev setting is not always obvious, so we provide a utility called **niclist** to list the names of network interfaces to use. When you run **niclist**, it will suggest an ne2k line which is a very good first try.

Next, if you are on a UNIX machine you will need to become the root user. Since bochs is sending and receiving raw network packets, you need to be root to use the network device. To allow normal users to do this would be a security problem.

Now run Bochs to boot DLX Linux. Press enter a few times to accept the default configuration choices. This tells Bochs read the configuration file and then begin. DLX Linux should boot in the Bochs window, and you should see that Linux detects the NE2000 card. Eventually it gets to a login prompt.

ne.c:v1.10 9/23/94 Donald Becker (becker@cesdis.gsfc.nasa.gov)  
 NE\*000 ethercard probe at 0x300: b0 c4 20 00 00 00  
 eth0: NE2000 found at 0x300, using IRQ 9.

At the login prompt, type "root" to log in as root. Then type the ifconfig and route commands to set up networking. The exact IP numbers in the example won't work for you; you must choose an IP configuration that is legal on your network.

dlx login: root  
 Linux 1.3.89.  
 dlx:~# ifconfig eth0 192.168.0.99 # set bochs IP address  
 dlx:~# route add -net 192.168.0.0 # first 3 numbers match IP  
 dlx:~# route add default gw 192.168.0.1 # your gateway to the net  
 dlx:~# \_

**Note:** The bochs IP address must be an unused IP address on your network. If you duplicate someone else's IP address, your network will become very confused.

Finally, the network is ready and you can test it out with ping, telnet, or ftp to various machines by their numerical IP address. Keep in mind that for all UNIX host platforms, Bochs networking cannot talk to the host machine. That means the host machine can't be the gateway either. You need another physical machine on the network that bochs can talk to. On Win32 this restriction does not apply.

**Note:** When you have a working network configuration, you can make DLX Linux recreate the same settings the next time you boot. Just add the ifconfig and route commands to the end of /etc/rc.d/rc.inet1. I won't try to describe how to use the **vi** editor in this limited amount of space...

To configure a name server, set up /etc/resolv.conf with the IP address of your name server as shown.

dlx:~# echo 'nameserver 192.168.0.1' > /etc/resolv.conf

|  |  |  |
| --- | --- | --- |
| [Prev](http://docs.google.com/smp.html) | [Home](http://docs.google.com/index.html) | [Next](http://docs.google.com/config-tuntap.html) |
| Simulating a Symmetric Multiprocessor (SMP) Machine | [Up](http://docs.google.com/howto.html) | Configuring and using a tuntap network interface |