



Tips & Tricks

 [Click To Print](#)

Tips & Tricks

November 2007 • Vol.7 Issue 11

Page(s) 98-99 in print issue

Warm Up To Penguins

The Down & Dirty Details Of Linux Network Configuration

Most recent distributions of Linux will do everything but shine your shoes when it comes to Linux network configuration. If you're on a network with a DHCP server, your newly minted Linux system should come up already talking to the Internet and ready to party. And when you do need to make a manual change, GUI-based tools hold your hand and make sure you don't stub your toe.



Well, that's not how a pro configures a network. Leave those namby-pamby tools where they belong, in a child's toy box. Real geeks configure their networks from a console prompt. Blindfolded!

In all seriousness, there are situations when you need to punt the user-friendly GUIs and poke at your network from the command line. For one thing, the GUI doesn't show you all the information available about your Ethernet interfaces, nor does it allow you to set all the options. And GUI tools won't help you when you're trying to administer your system remotely via SSH. In this series, we'll take a look at the command-line networking tools that should be in your arsenal. Also, before we dive in, just a quick point of reference: In our illustrations, we'll be using a system with a broadband connection; dial-up requires a totally different set of tools.

■ Health Checks

If you've had a broadband connection for any appreciable amount of time, you've had to deal with a dropped connection. Let's say that you aren't able to open Web pages in Firefox. To start, you'd probably try seeing if you can communicate with your local router. Let's say that your Linux system's local IP is on 192.168.1.178 and your router's is 192.168.1.1.

The Ping command is your first friend. There are three possible things you could see when you try to ping the router. If your Ethernet interface is screwed up, you'll see something like this:

```
turner@ubuntu:~$ ping 192.168.1.1
```

```
connect: Network is unreachable
```

This tells you that there's no way that the system can find to talk to the 192.168.1.x network at all. In this case, you're going to need to use **ifconfig** and **route** to solve the problem, which we'll discuss later. You're more likely to see something like this:

```
turner@ubuntu:~$ ping 192.168.1.1
```

```
PING 192.168.1.1 (192.168.1.1) 56(84) bytes of data.
```

```
From 192.168.1.178 icmp_seq=1 Destination Host Unreachable From 192.168.1.178 icmp_seq=2 Destination Host Unreachable  
From 192.168.1.178 icmp_seq=3 Destination Host Unreachable
```

```
--- 192.168.1.1 ping statistics ---
```

```
5 packets transmitted, 0 received, +3 errors, 100% packet loss, time 4009ms, pipe 3
```



Most of the time, you should use a GUI-based network configuration tool if you can.

Your system will keep trying the Ping command until you press CTRL-C, which we did after three failed pings. The important things to see here are the error (Destination Host Unreachable) and the statistics at the bottom (100% packet loss). What this tells us is that our system sent the packets out over the network but never received a reply. This can mean any of the following: The local system's network interface is bad; the network cable isn't plugged in; there's a problem somewhere farther down in the wiring; or that the router itself is hosed (to use a technical term).

The third possibility is that you'll see something akin to this:

```
turner@ubuntu:~$ ping 192.168.1.1
PING 192.168.1.1 (192.168.1.1) 56(84) bytes of data.
64 bytes from 192.168.1.1: icmp_seq=1 ttl=127 time=6.67 ms
64 bytes from 192.168.1.1: icmp_seq=2 ttl=127 time=0.772 ms
```

```
64 bytes from 192.168.1.1: icmp_seq=3 ttl=127 time=0.577 ms
```

```
64 bytes from 192.168.1.1: icmp_seq=4 ttl=127 time=0.572 ms
```

```
64 bytes from 192.168.1.1: icmp_seq=5 ttl=127 time=0.829 ms
```

```
--- 192.168.1.1 ping statistics ---
```

```
5 packets transmitted, 5 received, 0% packet loss, time 4017ms rtt min/avg/max/mdev = 0.572/1.884/6.674/2.397 ms
```

Here, the packets were being received and echoed back. The only really useful number other than the 0% packet loss is the "time=" figure, which also appears in among the ping statistics. This tells you how long it took for the packet to reach the other device and return, which, in our case, was around 0.7 milliseconds in this case (discounting the first packet, which can take a bit longer). This is the kind of response you'd expect from a local network; when talking to hosts across the world, things can become quite a bit slower, as we'll see.

■ **Ifconfig: Your Network Interface Friend**

Suppose that you get a "Network Is Unreachable" error. You should first make sure that you have a properly configured Ethernet interface. The command you use for this is **ifconfig**, which stands for interface configurator. The ifconfig output for a properly configured system might look something like this:

```
turner@ubuntu:~$ ifconfig

eth0 Link encap:Ethernet HWaddr 00:0C:29:E7:41:AA
inet addr:192.168.1.178 Bcast:192.168.1.255 Mask:255.255.255.0
inet6 addr: fe80::20c:29ff:fee7:41aa/64 Scope:Link

UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1

RX packets:1660 errors:0 dropped:0 overruns:0 frame:0
TX packets:520 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:1000

RX bytes:487892 (476.4 KiB) TX bytes:47399 (46.2 KiB)

Interrupt:177 Base address:0x1400
```

```
lo Link encap:Local Loopback
inet addr:127.0.0.1 Mask:255.0.0.0
inet6 addr: ::1/128 Scope:Host

UP LOOPBACK RUNNING MTU:16436 Metric:1

RX packets:486 errors:0 dropped:0 overruns:0 frame:0
TX packets:486 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:0

RX bytes:25912 (25.3 KiB) TX bytes:25912 (25.3 KiB)
```

You can see two network interfaces here. The “lo” interface is the loopback interface; it lets your system send and receive data to itself using the address 127.0.0.1. The “eth0” interface has the job of communicating with your local network. Let’s look at the useful information we can extract from it.

The HWaddr field is the Ethernet address (not the IP address) of the interface adapter. Every Ethernet interface device in the world should have a unique Ethernet hardware address. Sometimes a router needs to know this data, especially if you want to configure the router’s DHCP server to always give your system the same IP address, normally done on private subnets.

The next line tells you all about the local IP address assigned to this system. The really important one is the first section, which tells us that this system is configured to use the address 192.168.1.178. The remainder of the line gives you the broadcast address and subnet mask for the interface. For example, a class C subnet would have a broadcast address of 192.168.1.0 and a network mask of 255.255.255.0 (meaning that the first 24 bits of the address are used for determining the subnet, and the last 8 specify the address on the subnet).

The only other thing that you should look for is the word “UP,” two lines down. It tells us that the network interface is up and running. If you don’t see this, you can try starting the interface manually by typing **sudo ifconfig eth0 up**. If the command returns no output, it worked. It may also tell you that it can’t find the interface; in this case, you have a hardware problem. Looking in `/var/log/syslog` or typing **dmesg** at the console may give you some further information as to why it failed.

You can even do things like change your IP address. Typing **sudo ifconfig eth0 192.168.1.179 up** would change our local IP address from 192.168.1.179.

Note that you have to run the command as root (either by logging into the root account or by using sudo) to change an Ethernet interface, although you can ask for information as any user.

Let’s assume that you can successfully ping all the way to your router.

That leaves only a few reasons that your network could be down. Either some connection farther along is dead (such as between your ISP and the greater Internet), your DNS could be improperly configured, or your routing table could be wrong.

■ Next Steps

So what do you do if your network interface looks OK and you can ping your router, but you still can’t access the Internet? In the next column, we’ll look at some tools (traceroute, dig, and route) and a few files that can help you diagnose more complicated problems. We’ll also cover how you can make your fixes permanent. And in the final part of this series, you’ll get a peek at configuring an open-source wireless network. Until then, happy pinging! ■

by James Turner

Infinite Loop ROI: Zilch

Here's another example of negligence clashing with technology. Polish bus driver Leszek Wojcik was canned after he sent 38,000 text messages on a company cell phone in an attempt to win a \$36,000 prize. Things might have been OK if he had had permission to send 1,200 messages per day, instead of the 6 which he was actually allotted. The driver's comment regarding his fate: "Now I'm without work."

Source: cnews.canoe.ca/CNEWS/WeirdNews/2007/07/27/4373364-ap.html

Copyright © 2009 Sandhills Publishing Company U.S.A. All rights reserved.