Dr. Linux

ADSL Test Cases

If your ADSL link goes on strike, you can use simple tools that are available on any system to discover the cause of the problem using only your own hardware. BY MICHAEL RENNER

Question: My ADSL link suddenly quit working, I can't access any websites. How do I go about troubleshooting the problem?

Dr. Linux: Keep it simple: Are you sure you typed the URL correctly? Double-check the spelling. Is the webserver reachable? Try pinging the server to find out. After eliminating possible issues like these, you have to assume that you have a major network problem. Let's look at how to troubleshoot network problems using only standard Linux tools.

Let's start with the basics –

Being able to reach a working nameserver is also important. The */etc/resolv.conf* should contain the address of your nameserver.

If you can *ping* an IP address, but not the hostname assigned to this address, the nameserver may be unreachable or down.

This kind of error condition is typically short-lived, so you might just like to sit it out. However, it might be a good idea to obtain details of all your provider's nameservers and add these to your */etc/resolv.conf* file manually.

An ADSL modem with status LEDs is a big help. If your ADSL modem has LEDs, check the *Link* LED first to ensure that the hardware connection to your PC's NIC is working.

The *Sync* LED tells you the status of the link to the access concentrator. If a quick visual check of the LED proves inconclusive, there are a few software diagnosis tools you can use.

The protocol that ADSL uses "PPP over Ethernet" (PPPoE) requires both an Ethernet interface and a PPP device. The easiest way to check these interfaces is to run *ifconfig* in the command line.

If the ADSL link is up, the output should be similar to the following in Box 1:

The two interfaces, *eth0* and *ppp0*, are plainly visible in this output. The NIC, *eth0*, does not need an IP address, but *ppp0* needs both an IP address of its own and one for the other side of the connection (peer).

In our example, the peer address is *217.5.98.33*. The access concentrator automatically supplies you with an IP address when you access the Internet. In other words, you are assigned an address from an address pool – but not necessarily the same address – each time you establish a connection.

If the *ppp0* interface does not have an IP address, this could mean that the access concentrator node has not assigned an address for some reason.

Question: Both the *ppp0* interface and the IP address for the access point at the other side of the connection seem to be okay, but I can't ping the address. Also, I don't have an *eth0* interface. Instead *ifconfig* tells me that I have an interface called *tun0*. What does this additional interface mean?

Dr. Linux: If you have an ADSL modem with a USB port, it will use a virtual port like *tun0*, or *tap0*, rather than an Ethernet interface. This explains the missing Ethernet interface.

There may be a simple explanation for the fact that you can't ping the peer interface. There is an unfortunate tendency for providers to neglect the non-clickable part of the Internet and inhibit pings to their access concentrators.

The good news is that Linux does provide useful troubleshooting tools for this case, as much of the activity is logged in logfiles below */var/log/*. However, different distributions tend to log different events, as specified by the settings in *syslog.conf*.

Box 1:ifconfig

```
renner@hyaden:~$ /sbin/ifconfig
eth0      Link encap:Ethernet  HWaddr 00:60:08:BA:82:99
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:34154326 errors:23 dropped:0 overruns:20 frame:23
          TX packets:41041342 errors:0 dropped:0 overruns:0 carrier:0
          collisions:16329 txqueuelen:100
          RX bytes:1710005681 (1.5 GiB)  TX bytes:3907524253 (3.6 GiB)
          Interrupt:5 Base address:0x320

ppp0      Link encap:Point-to-Point Protocol
          inet addr:217.230.30.17  P-t-P:217.5.98.33
Mask:255.255.255.255
          UP POINTOPOINT RUNNING NOARP MULTICAST  MTU:1492  Metric:1
          RX packets:501327 errors:0 dropped:0 overruns:0 frame:0
          TX packets:635315 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:3
          RX bytes:69141770 (65.9 MiB)  TX bytes:258409218 (246.4 MiB)
```

Listing 1: Logfile */tmp/pppoe.log*

```
rp-pppoe-3.3
12:26:22.988 SENT PPPoE Discovery (8863) PADI sess-id 0 length 4
SourceAddr 00:60:08:ba:82:99 DestAddr ff:ff:ff:ff:ff:ff
01 01 00 00                                      ....

12:26:23.044 RCVD PPPoE Discovery (8863) PADO sess-id 0 length 38
SourceAddr 00:90:1a:10:14:77 DestAddr 00:60:08:ba:82:99
01 02 00 0a 53 54 47 58 31 32 2d 65 72 78 01 01   ....STGX12-erx..
00 00 01 04 00 10 6a d3 82 2b 33 69 29 6b 5c 5f   ......j..+3i)k\_
8f 4d fe 0a 5b 70                                 .M..[p

12:26:23.047 SENT PPPoE Discovery (8863) PADR sess-id 0 length 24
SourceAddr 00:60:08:ba:82:99 DestAddr 00:90:1a:10:14:77
01 01 00 00 01 04 00 10 6a d3 82 2b 33 69 29 6b   ........j..+3i)k
5c 5f 8f 4d fe 0a 5b 70                           \_.M..[p

12:26:28.050 SENT PPPoE Discovery (8863) PADR sess-id 0 length 24
SourceAddr 00:60:08:ba:82:99 DestAddr 00:90:1a:10:14:77
01 01 00 00 01 04 00 10 6a d3 82 2b 33 69 29 6b   ........j..+3i)k
5c 5f 8f 4d fe 0a 5b 70                           \_.M..[p

12:26:38.053 SENT PPPoE Discovery (8863) PADR sess-id 0 length 24
SourceAddr 00:60:08:ba:82:99 DestAddr 00:90:1a:10:14:77
01 01 00 00 01 04 00 10 6a d3 82 2b 33 69 29 6b   ........j..+3i)k
5c 5f 8f 4d fe 0a 5b 70                           \_.M..[p
```

Let's look at a typical Debian installation. The *debug* and *daemon.log* files are important here, as the *pppoe* program uses them to store messages. In case of error, these files should throw some light on the possible causes.

Watch out for messages generated by the access concentrator dialog, such as the exchange of account credentials (including the password dialog), and messages indicating that the connection has been established, or lost unexpectedly.

If the logs contain unusual messages, you should consider changing the debug level in the */etc/ppp/options* file, to provide additional data on the issue.

If you do not discover any suspicious entries, you should take a look at the *kernel.log*, *messages*, and *syslog* files next. The Linux kernel uses these files to log data for the PPP device and the network, amongst other things.

You might be looking at a scenario where the connection to your ADSL provider works fine, but a badly configured firewall is filtering and dropping packets. Packets required by an external application can also fall foul of a firewall.

Generally speaking, it makes sense to compare the current messages with messages logged when the system was still working to see if any obvious reasons for failure exist.

If the logs do not provide any real clues to the problem, you can perform some additional tests, for example, using *pppoe* to check the availability of the access concentrator. Enter the following command to do so:

```
hyaden:~# pppoe -I eth0 -A
Access-Concentrator: STGX12-erx
Got a cookie: 6a d3 82 2b 33 69
 29 6b 5c 5f 8f 4d fe 0a 5b 70
------------------------------
AC-Ethernet-Address:
00:90:1a:10:14:77
------------------------------
```

Messages such as *no such interface* or *pppoe: Timeout waiting for PADO packets* would indicate connection problems that prevent you from reaching the access concentrator. You can use the following test to make sure that the access concentrator is actually working – you will need root privileges to run the test:

```
hyaden:~# pppoe -I eth0 -T 20
-D /tmp/pppoe.log
```

If a file is written, it should be similar to Listing 1 – provided the link to the access concentrator is up.

If these tests all worked out, there might be a routing problem. To check this, first use *route* to ensure that the PPP interface is the default route (0.0.0.0) for all connections as in Box 2.

This output shows a direct link to the IP address 217.5.98.33 as the first entry. In the line below this, there is a 0.0.0.0 entry for all other IP addresses that will route packets to the network via the gateway. The last line shows that the *ppp0* interface is used in both cases.

If you also have a LAN connection that uses a second interface card, there will be another column that will look something like this:

```
192.168.1.0  0.0.0.0
255.255.255.0   U     0      0
      0 eth1
```

This entry sends any IP packets destined for the local network to the *eth1* interface, that is to a second network adapter.

Unfortunately, if all of these tests still proved negative, there is very little more you can do, using standard tools. You could always try to access your provider's status page to see if they know of any current issues, although this may involve you using a dial-up modem, friend on another ISP or just the telephone. ■

Box 2: Finding the route

```
hyaden:~# route -n
Kernel IP routing table
Destination     Gateway Genmask Flags   Metric  Ref     Use     Iface
217.5.98.33     0.0.0.0 255.255.255.255 UH      0       0       0
ppp0
0.0.0.0 217.5.98.33     0.0.0.0 UG      0       0       0       ppp0
```