13 The Cisco Troubleshooting Methodology - Lab Exercise

This lab tests your network troubleshooting skills.

Troubleshoot Connectivity to DNS Server

1) R3 has just been configured as a server for the Flackbox.lab domain. Members of staff have complained that DNS is not working.

2) From R1, use Telnet to check if the DNS service appears operational on R3

```
R1#telnet 10.10.20.1 53
Trying 10.10.20.1, 53 ...
% Destination unreachable; gateway or host down
```

3) When you have verified that DNS is not working, troubleshoot and fix the problem. Note that there may be more than one issue causing the problem.

There is more than one way to troubleshoot the issue. A suggested workflow is shown below.

The first two questions to ask when troubleshooting a problem are:

1. Was it working before? If so, has something changed which could cause the problem? This will usually direct you to the cause.

This question is not particularly useful for our example as the DNS server has just been brought online for the first time.

2. Is the problem affecting everybody or just one particular user? If it's affecting just one user, the likelihood is that the problem is at their end.

In this case the problem is affecting all users, so the problem is likely on the server end or with the network.

The error message when we tried to Telnet was 'Destination unreachable', so it looks like a connectivity issue.

Ping from R1 to R3.

```
R1#ping 10.10.20.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.10.20.1, timeout is 2
seconds:
UUUUU
Success rate is 0 percent (0/5)
```

The ping fails at the network layer so there is little point in checking the DNS service at higher layers until we fix this problem.

Rather than checking connectivity hop by hop, we can possibly save a little time by using traceroute.

```
R1#traceroute 10.10.20.1

Type escape sequence to abort.
Tracing the route to 10.10.20.1

  1 10.10.10.2 64 msec 60 msec 60 msec
  2 10.10.10.2 !H  !H  !H
```

The traceroute got as far as R2, which lets us know that R1 has the correct route to get to R3, and the problem is probably between R2 and R3.

R2 has an interface connected to the 10.10.20.0/24 network, so we don't need to check it has a route to R3. We do need to check that the interface is up though.

```
R2#sh ip int brief
Interface                  IP-Address      OK? Method Status                Protocol
FastEthernet0/0            10.10.10.2      YES NVRAM  up                    up
FastEthernet0/1            unassigned      YES NVRAM  administratively down down
FastEthernet1/0            10.10.20.2      YES NVRAM  administratively down down
FastEthernet2/0            unassigned      YES NVRAM  administratively down down
FastEthernet3/0            unassigned      YES NVRAM  administratively down down
```

There's the problem – FastEthernet1/0 facing R3 is administratively shutdown. Let's fix it.

```
R2(config)#interface f1/0
R2(config-if)#no shut
```

Next we'll try pinging from R1 to R3 again to verify we fixed connectivity.

```
R1#ping 10.10.20.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.10.20.1, timeout is 2
seconds:
.!!!!
Success rate is 80 percent (4/5), round-trip min/avg/max =
1/1/1 ms
```

That looks better. Next we'll verify DNS is working.

```
R1#ping R3.flackbox.lab

Translating "R3.flackbox.lab"...domain server (10.10.10.1)
% Unrecognized host or address, or protocol not running.
```

The error message tells us the problem if we take the time to really read it – R1 is using 10.10.10.1 as its DNS server, but the correct address is 10.10.20.1.

We fix that next.

```
R1(config)#ip name-server 10.10.20.1
```

Then test again.

```
R1#ping R3.flackbox.lab

Translating "R3.flackbox.lab"...domain server (10.10.10.1)
(10.10.20.1)
% Unrecognized host or address, or protocol not running.
```

The error message helps us again. First, we forgot to remove the incorrect DNS server entry. Fix that first.

```
R1(config)#no ip name-server 10.10.10.1
```

We know we have connectivity and the DNS server configured correctly on R1, so the problem looks like it's on R3.

From R1, let's see if the DNS service is running on it.

```
R1#telnet 10.10.20.1 53
Trying 10.10.20.1, 53 ...
% Connection refused by remote host
```

It looks like DNS isn't running on R3. Let's check on R3.

```
R3#sh run | include dns
R3#
```

The DNS server command is missing. We fix that next.

```
R3(config)#ip dns server
```

And then check the rest of the DNS configuration.

```
R3#sh run
Building configuration...
ip domain name flackbox.lab
ip host R1 10.10.10.1
ip host R2 10.10.10.2
ip host R3 10.10.20.1
ip host R1.flackbox.lab 10.10.10.1
ip host R2.flackbox.lab 10.10.10.2
ip host R3.flackbox.lab 10.10.20.1
ip name-server 10.10.20.1
!
ip dns server
!
truncated
```

That looks better. Time to test it from R1 again. We try telnet to port 53 first.

```
R1#telnet 10.10.20.1 53
Trying 10.10.20.1, 53 ... Open

[Connection to 10.10.20.1 closed by foreign host]
```

'Open' shows that R3 is listening on port 53. The final check is to verify by pinging by FQDN.

```
R1#ping R3.flackbox.lab
Translating "R3.flackbox.lab"...domain server (10.10.20.1)
[OK]

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.10.20.1, timeout is 2
seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max =
1/1/4 ms
```

That’s the problem solved.

To summarise the issues: port FastEthernet1/0 was shut down on R2, R1 was using the wrong IP address for the DNS server, and the DNS service was not running on R3.

Problems in the real world are usually caused by just one error rather than three as in this case. This can still occur though, particularly when a new service is being deployed.