



UNIVERSITY OF DHAKA

CSE:3111 COMPUTER NETWORKING LAB

Lab Report

**Title: Lab exercises on LAN configuration
and troubleshooting tools**

Diponker Roy
Roll: 28
Reg: 2020315638

Submitted to:

Dr. Md. Abdur Razzaque
Dr. Muhammad Ibrahim
Dr. Md. Redwan Ahmed Rizvee
Dr. Md. Mamun Or Rashid

Submitted on: 2021-08-31

Introduction

Provide an overview of the importance of LAN configuration and troubleshooting in maintaining a healthy network. State the objectives of the lab exercises and mention the tools you will be using.

Methodology

Detail the specific steps taken during the lab exercises, including the configuration changes made, the devices involved, and the scenarios tested.

LAN configuration exercises

0.1 PING

Using the `ping` command, test the connectivity between two devices. When the ping is successful, record the round-trip time (RTT) and the number of packets lost. When the ping is unsuccessful, record the error message. Repeat the test with different packet sizes and record the results.

Command Execution

To use PING, open the terminal and execute the following command:
ping < destinationHost >

Sample Output

Here is a sample output of a PING command:

```
user@user-Inspiron-3501:~/Desktop/latex$ ping facebook.com
PING facebook.com (157.240.1.35) 56(84) bytes of data:
64 bytes from edge-star-mini-shv-01-ccul.facebook.com (157.240.1.35): icmp_seq=1 ttl=54 time=10.3 ms
64 bytes from edge-star-mini-shv-01-ccul.facebook.com (157.240.1.35): icmp_seq=2 ttl=54 time=9.38 ms
64 bytes from edge-star-mini-shv-01-ccul.facebook.com (157.240.1.35): icmp_seq=3 ttl=54 time=16.5 ms
64 bytes from edge-star-mini-shv-01-ccul.facebook.com (157.240.1.35): icmp_seq=4 ttl=54 time=11.1 ms
64 bytes from edge-star-mini-shv-01-ccul.facebook.com (157.240.1.35): icmp_seq=5 ttl=54 time=14.0 ms
```

Analysis

In the sample output, the PING command sends ICMP (Internet Control Message Protocol) packets to the specified host (in this case, `google.com`) and receives responses. The output includes information such as the round-trip time (rtt) for each packet and overall statistics.

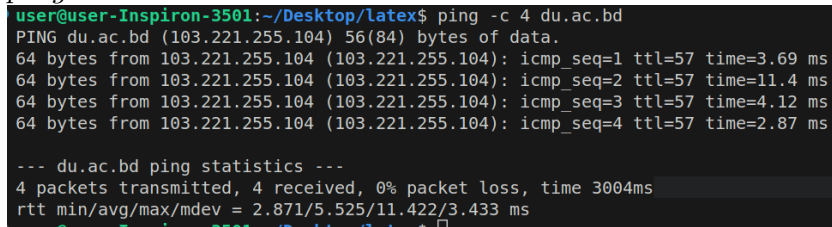
By analyzing the PING output, you can assess the network connectivity and performance to the specified destination.

Limiting the number of PING requests

By default, the PING command sends an unlimited number of requests to the destination. To limit the number of requests, use the `-c` option:

Sample Output

ping -c < num > < destinationHost >



```
user@user-Inspiron-3501:~/Desktop/latex$ ping -c 4 du.ac.bd
PING du.ac.bd (103.221.255.104) 56(84) bytes of data:
64 bytes from 103.221.255.104 (103.221.255.104): icmp_seq=1 ttl=57 time=3.69 ms
64 bytes from 103.221.255.104 (103.221.255.104): icmp_seq=2 ttl=57 time=11.4 ms
64 bytes from 103.221.255.104 (103.221.255.104): icmp_seq=3 ttl=57 time=4.12 ms
64 bytes from 103.221.255.104 (103.221.255.104): icmp_seq=4 ttl=57 time=2.87 ms

--- du.ac.bd ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3004ms
rtt min/avg/max/mdev = 2.871/5.525/11.422/3.433 ms
```

Analysis

In the sample output, the PING command sends 4 ICMP packets to the specified host (in this case, `du.ac.bd`) and receives responses. The output includes information such as the round-trip time (rtt) for each packet and overall statistics.

0.2 TRACEROUTE

Using the `tracert` command, test the connectivity between two devices. When the traceroute is successful, record the RTT and the number of hops. When the traceroute is unsuccessful, record the error message. Repeat the test with different packet sizes and record the results.

Command Execution

To use TRACEROUTE, open the terminal and execute the following

command: *traceroute < destinationHost >*

```
user@user-Inspiron-3501:~/Desktop/latex$ traceroute google.com
traceroute to google.com (64.233.170.101), 30 hops max, 60 byte packets
 1  lab707-11 (10.42.0.1)  12.698 ms  12.657 ms  12.720 ms
 2  gateway (10.33.3.62)  31.030 ms  31.011 ms  30.994 ms
 3  10.33.4.2 (10.33.4.2)  12.834 ms  12.884 ms  13.294 ms
 4  103.221.254.33 (103.221.254.33)  13.386 ms  13.488 ms  13.582 ms
 5  163.47.38.93 (163.47.38.93)  15.051 ms  15.034 ms  15.019 ms
 6  100.100.1.45 (100.100.1.45)  15.003 ms  13.105 ms  13.064 ms
 7  * 100.100.0.97 (100.100.0.97)  4.487 ms  4.448 ms
 8  100.211.102.109 (100.211.102.109)  5.236 ms  5.222 ms  5.205 ms
 9  123.49.8.49 (123.49.8.49)  5.308 ms  5.662 ms  5.638 ms
10  142.250.169.168 (142.250.169.168)  53.420 ms  53.405 ms  53.390 ms
11  * * *
12  100.170.240.225 (100.170.240.225)  53.748 ms  100.170.254.225 (100.170.254.225)  50.767 ms  100.170.240.225 (100.170.240.225)  50.717 ms
13  100.170.240.242 (100.170.240.242)  58.973 ms  204.595 ms  204.548 ms
14  216.239.49.74 (216.239.49.74)  204.531 ms  * 216.239.50.192 (216.239.50.192)  204.042 ms
15  100.170.226.83 (100.170.226.83)  204.482 ms  142.251.231.182 (142.251.231.182)  204.011 ms  142.251.231.184 (142.251.231.184)  204.45 ms
0 ms
16  100.170.234.203 (100.170.234.203)  204.435 ms  142.251.247.201 (142.251.247.201)  203.963 ms  142.250.232.149 (142.250.232.149)  203.963 ms
```

Number of hops is 30. It indicates the time to live (TTL) of the packet. The TTL value is decremented by one each time the packet is forwarded by a router. When the TTL value reaches zero, the packet is discarded and an ICMP error message is sent to the source. The source can use the ICMP error message to determine the IP address of the router that discarded the packet. The source can then use the IP address to determine the router name.

Limiting the number of hops

By default, the TRACEROUTE command sends packets with a TTL value of 1 and increments the TTL value by 1 for each subsequent packet. To limit the number of hops, use the *-m* option:

Sample Output

traceroute -m < num > < destinationHost >

```
user@user-Inspiron-3501:~/Desktop/latex$ traceroute -m 5 du.ac.bd
traceroute to du.ac.bd (103.221.255.104), 5 hops max, 60 byte packets
 1  gateway (192.168.0.1)  2.182 ms  2.246 ms  2.353 ms
 2  10.92.68.1 (10.92.68.1)  3.949 ms  5.566 ms  5.676 ms
 3  10.40.40.101 (10.40.40.101)  6.799 ms  7.407 ms  7.526 ms
 4  10.0.5.17 (10.0.5.17)  7.635 ms  7.735 ms  7.844 ms
 5  103.161.216.92 (103.161.216.92)  8.774 ms  9.560 ms  10.002 ms
```

We use 5 hops. It indicates the time to live (TTL) of the packet. The TTL value is decremented by one each time the packet is forwarded by a router.

When the TTL value reaches zero, the packet is discarded and an ICMP error message is sent to the source. The source can use the ICMP error message to determine the IP address of the router that discarded the packet. The source can then use the IP address to determine the router name.

0.3 IFCONFIG

Using the `ifconfig` command, display the IP address, subnet mask, and default gateway for a device. Record the results. The “`ifconfig`” command with no arguments will display all the active network interface configuration details that includes their assigned IP addresses, netmasks, and other relevant information.

Command Execution

To use IFCONFIG, open the terminal and execute the following command:

ifconfig

```
Processing triggers for man-db (2.10.2-1) ...
user@user-Inspiron-3501:~/Desktop/latex$ ifconfig
enp3s0: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500
        ether 60:18:95:3c:fe:15 txqueuelen 1000 (Ethernet)
        RX packets 0 bytes 0 (0.0 B)
        RX errors 0 dropped 0 overruns 0 frame 0
        TX packets 0 bytes 0 (0.0 B)
        TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1000 (Local Loopback)
    RX packets 131280 bytes 43458757 (43.4 MB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 131280 bytes 43458757 (43.4 MB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

wlp4s0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.0.103 netmask 255.255.255.0 broadcast 192.168.0.255
    inet6 fe80::2518:1e47:622c:18f8 prefixlen 64 scopeid 0x20<link>
    ether 90:0f:0c:a9:a6:2b txqueuelen 1000 (Ethernet)
    RX packets 75625 bytes 48184680 (48.1 MB)
    RX errors 0 dropped 1 overruns 0 frame 0
    TX packets 67381 bytes 21699389 (21.6 MB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

Analysis

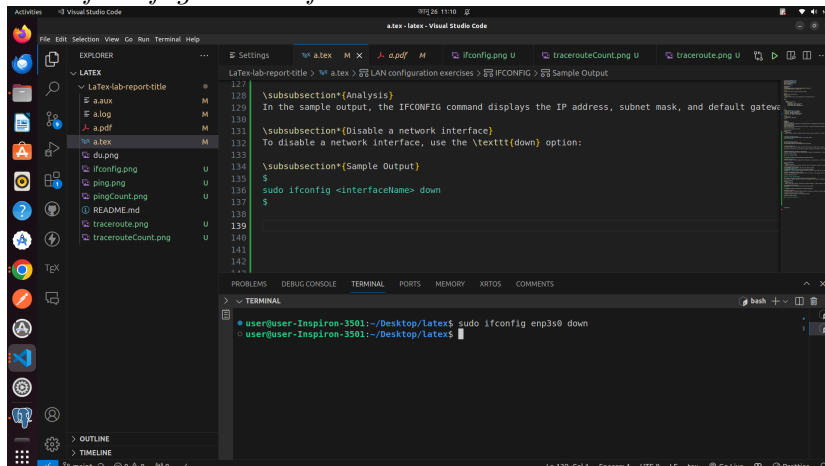
In the sample output, the IFCONFIG command displays the IP address, subnet mask, and default gateway for the device. The output also includes information such as the MAC address, MTU, and the number of packets transmitted and received.

Disable a network interface

To disable a network interface, use the **down** option:

Sample Output

sudo ifconfig <interfaceName> down



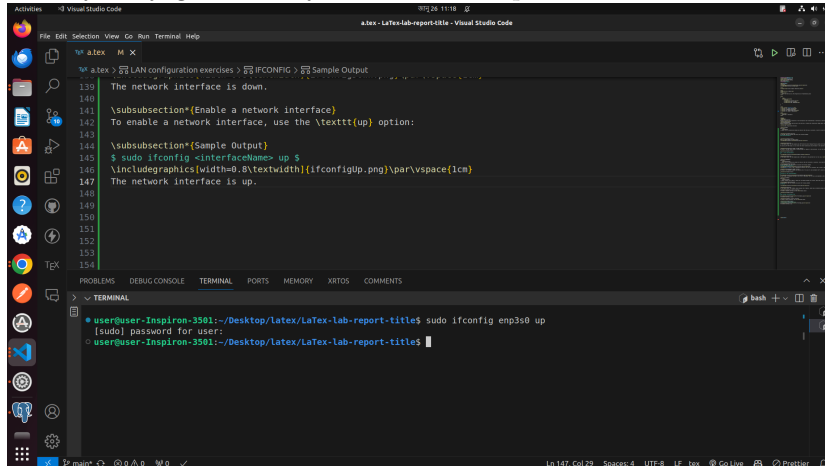
The network interface is down.

Enable a network interface

To enable a network interface, use the **up** option:

Sample Output

sudo ifconfig < interfaceName > up



The network interface is up.

0.4 ARP

Using the `arp` command, display the ARP cache for a device. Record the results.

Command Execution

To use ARP, open the terminal and execute the following command: `arp`

```
user@user-Inspiron-3501:~/Desktop/latex/LaTeX-lab-report-title$ arp
Address          HWtype  HWaddress      Flags Mask    Iface
_gateway         ether    e0:1c:fc:a6:80:c6  C             wlp4s0
169.254.169.254   (incomplete)
_gateway         ether    e0:1c:fc:a6:80:c6  C             enp3s0
user@user-Inspiron-3501:~/Desktop/latex/LaTeX-lab-report-title$
```

Analysis

In the sample output, the ARP command displays the ARP cache for the device. The output includes information such as the IP address, MAC address, and type of each entry.

RARP

Using the `rarp` command, display the RARP cache for a device. Record the results. The Reverse Address Resolution Protocol (RARP) is a networking protocol that is used to map a physical (MAC) address to an Internet Protocol (IP) address. It is the reverse of the more commonly used Address Resolution Protocol (ARP), which maps an IP address to a MAC address.

Command Execution

To use RARP, open the terminal and execute the following command:

rarp

```
user@user-Inspiron-3501:~/Desktop/latex/LaTeX-lab-report-title$ rarp
Usage: rarp -a                list entries in cache.
       rarp -d <hostname>    delete entry from cache.
       rarp [<HW>] -s <hostname> <hwaddr> add entry to cache.
       rarp -f                add entries from /etc/ethers.
       rarp -V                display program version.

<HW>=Use '-H <hw>' to specify hardware address type. Default: ether
List of possible hardware types (which support ARP):
ash (Ash) ether (Ethernet) ax25 (AMPR AX.25)
netrom (AMPR NET/ROM) rose (AMPR ROSE) arcnet (ARCnet)
dlci (Frame Relay DLCI) fddi (Fiber Distributed Data Interface) hippi (HIPPI)
irda (IrLAP) x25 (generic X.25) eui64 (Generic EUI-64)
```

Analysis

In the sample output, the RARP command displays the RARP cache for the device. The output includes information such as the IP address, MAC address, and type of each entry.

NSLOOKUP

Using the `nslookup` command, display the DNS cache for a device. Record the results.

Command Execution

To use NSLOOKUP, open the terminal and execute the following command:

nslookup


```

user@user-Inspiron-3501:~/Desktop/latex/LaTeX-lab-report-title$ nslookup google.com
Server:      127.0.0.53
Address:     127.0.0.53#53

Non-authoritative answer:
Name:   google.com
Address: 142.250.182.14
Name:   google.com
Address: 2404:6800:4007:819::200e

```

Analysis

In the sample output, the NSLOOKUP command displays the DNS cache for the device. The output includes information such as the IP address, hostname, and type of each entry. nslookup followed by the domain name will display the “A Record” (IP Address) of the domain. Use this command to find the address record for a domain. It queries domain name servers and gets the details.

Using type any

To display all the information in the DNS cache, use the `set type=any` option:

Sample Output

nslookup - type = any < domainName >

```

user@user-Inspiron-3501:~/Desktop/latex/LaTeX-lab-report-title$ nslookup -type=any google.com
Server:      127.0.0.53
Address:     127.0.0.53#53

Non-authoritative answer:
Name:   google.com
Address: 2404:6800:4007:819::200e
google.com    nameserver = ns4.google.com.
google.com    nameserver = ns2.google.com.
google.com    nameserver = ns1.google.com.
google.com    nameserver = ns3.google.com.
google.com    rdata_65 = 1 . alpn="h2,h3"
Name:   google.com
Address: 142.250.182.14

Authoritative answers can be found from:
ns1.google.com internet address = 216.239.32.10
ns2.google.com internet address = 216.239.34.10
ns3.google.com internet address = 216.239.36.10
ns4.google.com internet address = 216.239.38.10
ns1.google.com has AAAA address 2001:4860:4802:32::a
ns2.google.com has AAAA address 2001:4860:4802:34::a
ns3.google.com has AAAA address 2001:4860:4802:36::a
ns4.google.com has AAAA address 2001:4860:4802:38::a

```

There are also available types of DNS records. The most common ones are:

- A: Address record
- AAAA: IPv6 address record
- CNAME: Canonical name record
- MX: Mail exchange record
- NS: Name server record
- PTR: Pointer record
- SOA: Start of authority record
- TXT: Text record

NETSTAT

Using the `netstat` command, display the active TCP connections for a device. Record the results. The `netstat` command is like a special tool in Linux that helps you understand and check things about how your computer connects to the internet. It can tell you about the connections your computer is making, the paths it uses to send information, and even some technical details like how many packets of data are being sent or received. In simple terms, it's like a window that shows you what's happening with your computer and the internet. This article will help you learn how to use `netstat`, exploring different ways to get specific information and giving you a better idea of what's going on behind the scenes.

Command Execution

To use NETSTAT, open the terminal and execute the following command:
netstat

```

unix 3 [ ] STREAM CONNECTED 38518
unix 9 [ ] DGRAM CONNECTED 22566 /run/systemd/journal/socket
unix 3 [ ] STREAM CONNECTED 767163
unix 3 [ ] STREAM CONNECTED 749286
unix 3 [ ] STREAM CONNECTED 742230
unix 3 [ ] STREAM CONNECTED 736969 /run/user/1000/gvfsd/socket-63SR30wd
unix 3 [ ] STREAM CONNECTED 721234
unix 2 [ ] DGRAM CONNECTED 154793
unix 3 [ ] STREAM CONNECTED 61813
unix 3 [ ] STREAM CONNECTED 51779 /run/user/1000/at-spi/bus
unix 3 [ ] STREAM CONNECTED 32440 /run/systemd/journal/stdout
unix 3 [ ] STREAM CONNECTED 34892 /run/dbus/system_bus_socket
unix 3 [ ] STREAM CONNECTED 30623 /run/systemd/journal/stdout
unix 3 [ ] STREAM CONNECTED 766219
unix 3 [ ] STREAM CONNECTED 728037 /run/user/1000/pulse/native
unix 3 [ ] STREAM CONNECTED 58348 @/tmp/.X11-unix/X0
unix 3 [ ] STREAM CONNECTED 59787
unix 3 [ ] STREAM CONNECTED 43061
unix 3 [ ] STREAM CONNECTED 33575 /run/user/1000/bus
unix 3 [ ] STREAM CONNECTED 23057
unix 3 [ ] STREAM CONNECTED 767166
unix 3 [ ] STREAM CONNECTED 762703
unix 3 [ ] STREAM CONNECTED 72006
unix 3 [ ] SEQPACKET CONNECTED 61799
unix 3 [ ] STREAM CONNECTED 37726 @/tmp/.X11-unix/X1
unix 3 [ ] STREAM CONNECTED 38080 /run/systemd/journal/stdout
unix 3 [ ] STREAM CONNECTED 23444
unix 3 [ ] STREAM CONNECTED 16175
unix 3 [ ] STREAM CONNECTED 142454
unix 3 [ ] STREAM CONNECTED 58991
unix 3 [ ] STREAM CONNECTED 49594
unix 3 [ ] STREAM CONNECTED 41813
unix 3 [ ] STREAM CONNECTED 15666

```

Analysis

In the sample output, the NETSTAT command displays the active TCP connections for the device. The output includes information such as the protocol, local address, foreign address, and state of each connection.