HOMEWORK 1

BY: Muppidi Harshavardhan Reddy (mreddy2)
Srinivas Parasa (sparasa)

1) Basic Linux network verification tasks.

(1) Interfaces:

By running the command ip addr we can get to know about the interfaces attached as shown:

```
ece792@ece792-Standard-PC-i440FX-PIIX-1996:~$
ece792@ece792-Standard-PC-i440FX-PIIX-1996:~$ ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: ens3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000
    link/ether 52:54:00:06:55:66 brd ff:ff:ff:ff:ff:ff
    inet 192.168.124.5/24 brd 192.168.124.255 scope global dynamic ens3
       valid_lft 3088sec preferred_lft 3088sec
    inet6 fe80::faaa:649d:f116:3492/64 scope link
       valid_lft forever preferred_lft forever
3: ens4: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000
    link/ether 52:54:00:5f:78:57 brd ff:ff:ff:ff:ff:ff
    inet6 fe80::38ea:7a42:16b4:78f4/64 scope link
       valid_lft forever preferred_lft forever
4: ens5: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000
    link/ether 52:54:00:1e:f4:66 brd ff:ff:ff:ff:ff:ff
    inet 192.168.123.185/24 brd 192.168.123.255 scope global dynamic ens5
       valid_lft 2560sec preferred_lft 2560sec
    inet6 fe80::76b2:71db:5811:bb4b/64 scope link
       valid_lft forever preferred_lft forever
ece792@ece792-Standard-PC-i440FX-PIIX-1996:~$
```

(2) Routing Table

By running the command route -n we can get to know about the interfaces attached as shown:

```
ece792@ece792-Standard-PC-i440FX-PIIX-1996:~$
ece792@ece792-Standard-PC-i440FX-PIIX-1996:~$ route -n
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
0.0.0.0         192.168.124.1   0.0.0.0         UG    100    0        0 ens3
0.0.0.0         192.168.123.1   0.0.0.0         UG    101    0        0 ens5
169.254.0.0     0.0.0.0         255.255.0.0     U     1000   0        0 ens3
192.168.123.0   0.0.0.0         255.255.255.0   U     100    0        0 ens5
192.168.124.0   0.0.0.0         255.255.255.0   U     100    0        0 ens3
ece792@ece792-Standard-PC-i440FX-PIIX-1996:~$
ece792@ece792-Standard-PC-i440FX-PIIX-1996:~$
```

(3) DNS

The DNS could be know from viewing the resolv.conf file in the /etc directory which gives the following content:

```
ece792@ece792-Standard-PC-i440FX-PIIX-1996:~$
ece792@ece792-Standard-PC-i440FX-PIIX-1996:~$
ece792@ece792-Standard-PC-i440FX-PIIX-1996:~$ cat /etc/resolv.conf
# Dynamic resolv.conf(5) file for glibc resolver(3) generated by resolvconf(8)
#     DO NOT EDIT THIS FILE BY HAND -- YOUR CHANGES WILL BE OVERWRITTEN
nameserver 127.0.1.1
ece792@ece792-Standard-PC-i440FX-PIIX-1996:~$
```

(4) DHCP

The DHCP configurations are found in dhclient.conf file which is located in etc/dhcp/dhclient.conf , When viewed we can observe as below:

```
ece792@ece792-Standard-PC-i440FX-PIIX-1996:~$ cat /etc/dhcp/dhclient.conf
# Configuration file for /sbin/dhclient.
#
# This is a sample configuration file for dhclient. See dhclient.conf's
#       man page for more information about the syntax of this file
#       and a more comprehensive list of the parameters understood by
#       dhclient.
#
# Normally, if the DHCP server provides reasonable information and does
#       not leave anything out (like the domain name, for example), then
#       few changes must be made to this file, if any.
#

option rfc3442-classless-static-routes code 121 = array of unsigned integer 8;

send host-name = gethostname();
request subnet-mask, broadcast-address, time-offset, routers,
        domain-name, domain-name-servers, domain-search, host-name,
        dhcp6.name-servers, dhcp6.domain-search, dhcp6.fqdn, dhcp6.sntp-servers,
        netbios-name-servers, netbios-scope, interface-mtu,
        rfc3442-classless-static-routes, ntp-servers;

#send dhcp-client-identifier 1:0:a0:24:ab:fb:9c;
#send dhcp-lease-time 3600;
#supersede domain-name "fugue.com home.vix.com";
#prepend domain-name-servers 127.0.0.1;
#require subnet-mask, domain-name-servers;
timeout 300;
#retry 60;
#reboot 10;
#select-timeout 5;
#initial-interval 2;
#script "/sbin/dhclient-script";
#media "-link0 -link1 -link2", "link0 link1";
#reject 192.33.137.209;

#alias {
#  interface "eth0";
#  fixed-address 192.5.5.213;
#  option subnet-mask 255.255.255.255;
#}

#lease {
#  interface "eth0";
#  fixed-address 192.33.137.200;
#  medium "link0 link1";
#  option host-name "andare.swiftmedia.com";
#  option subnet-mask 255.255.255.0;
#  option broadcast-address 192.33.137.255;
#  option routers 192.33.137.250;
#  option domain-name-servers 127.0.0.1;
#  renew 2 2000/1/12 00:00:01;
#  rebind 2 2000/1/12 00:00:01;
#  expire 2 2000/1/12 00:00:01;
#}
ece792@ece792-Standard-PC-i440FX-PIIX-1996:~$
```

2) Basic Linux performance verification tasks:

1. CPU usage: Three reports of statistics for all processors at two second intervals. By running the command mpstat -P ALL 2 3 we can get the

```
ece792@ece792-Standard-PC-i440FX-PIIX-1996:~$
ece792@ece792-Standard-PC-i440FX-PIIX-1996:~$ mpstat -P ALL 2 3
Linux 4.10.0-28-generic (ece792-Standard-PC-i440FX-PIIX-1996)   09/18/2018      _x86_64_        (4 CPU)

01:43:44 PM  CPU    %usr   %nice    %sys %iowait    %irq   %soft  %steal  %guest  %gnice   %idle
01:43:46 PM  all    0.38    0.00    0.12    0.00    0.00    0.00    0.00    0.00    0.00   99.50
01:43:46 PM    0    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00  100.00
01:43:46 PM    1    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00  100.00
01:43:46 PM    2    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00  100.00
01:43:46 PM    3    1.99    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00   98.01

01:43:46 PM  CPU    %usr   %nice    %sys %iowait    %irq   %soft  %steal  %guest  %gnice   %idle
01:43:48 PM  all    0.38    0.00    0.13    0.00    0.00    0.00    0.00    0.00    0.00   99.50
01:43:48 PM    0    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00  100.00
01:43:48 PM    1    0.50    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00   99.50
01:43:48 PM    2    0.50    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00   99.50
01:43:48 PM    3    1.00    0.00    0.50    0.00    0.00    0.00    0.00    0.00    0.00   98.50

01:43:48 PM  CPU    %usr   %nice    %sys %iowait    %irq   %soft  %steal  %guest  %gnice   %idle
01:43:50 PM  all    0.25    0.00    0.25    0.00    0.00    0.00    0.00    0.00    0.00   99.50
01:43:50 PM    0    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00  100.00
01:43:50 PM    1    0.00    0.00    0.50    0.00    0.00    0.00    0.00    0.00    0.00   99.50
01:43:50 PM    2    0.00    0.00    0.50    0.00    0.00    0.00    0.00    0.00    0.00   99.50
01:43:50 PM    3    1.01    0.00    0.50    0.00    0.00    0.00    0.00    0.00    0.00   98.49

Average:     CPU    %usr   %nice    %sys %iowait    %irq   %soft  %steal  %guest  %gnice   %idle
Average:     all    0.33    0.00    0.17    0.00    0.00    0.00    0.00    0.00    0.00   99.50
Average:       0    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00  100.00
Average:       1    0.17    0.00    0.17    0.00    0.00    0.00    0.00    0.00    0.00   99.67
Average:       2    0.17    0.00    0.17    0.00    0.00    0.00    0.00    0.00    0.00   99.67
Average:       3    1.33    0.00    0.33    0.00    0.00    0.00    0.00    0.00    0.00   98.33
ece792@ece792-Standard-PC-i440FX-PIIX-1996:~$
```

As we can see from the above result CPU 0 is idle the most which is 100% in all the cases.

2. Memory usage: 3 reports of MEM statistics for every active task in the system at two second intervals.
For testing purposes we have stressed the system and ran the command pidstat -r 2 3 and observed the following results as follows:

```
ece792@ece792-Standard-PC-i440FX-PIIX-1996:~$
ece792@ece792-Standard-PC-i440FX-PIIX-1996:~$ pidstat -r 2 3
Linux 4.10.0-28-generic (ece792-Standard-PC-i440FX-PIIX-1996)   09/21/2018      _x86_64_        (4 CPU)

04:38:55 PM   UID       PID  minflt/s  majflt/s     VSZ     RSS   %MEM  Command
04:38:57 PM     0       900     19.90      0.00  363928   45120   0.18  Xorg
04:38:57 PM  1000     24813     31.84      0.00    7712    2156   0.01  pidstat

04:38:57 PM   UID       PID  minflt/s  majflt/s     VSZ     RSS   %MEM  Command
04:38:59 PM     0       900     20.00      0.00  363928   45120   0.18  Xorg
04:38:59 PM  1000     24813      6.50      0.00    7712    2420   0.01  pidstat

04:38:59 PM   UID       PID  minflt/s  majflt/s     VSZ     RSS   %MEM  Command
04:39:01 PM     0       900     21.00      0.00  363928   45120   0.18  Xorg

Average:      UID       PID  minflt/s  majflt/s     VSZ     RSS   %MEM  Command
Average:        0       900     20.30      0.00  363928   45120   0.18  Xorg
Average:     1000     24813     12.81      0.00    7712    2332   0.01  pidstat
ece792@ece792-Standard-PC-i440FX-PIIX-1996:~$
```

As we can see from the observation that the most memory intensive task is by Xorg with PID 900.

3) Basic Linux tasks, use of tools:

After setting up the server and then connecting it from the client and varying the packet size starting with 100Bytes to 6400Bytes. We observe the following results:

On the Client Side:

```
^Cece792@ece792-Standard-PC-i440FX-PIIX-1996:~$ iperf -c 192.168.124.5 -t 10 -l 100
------------------------------------------------------------
Client connecting to 192.168.124.5, TCP port 5001
TCP window size: 2.50 MByte (default)
------------------------------------------------------------
[  3] local 192.168.124.5 port 44868 connected with 192.168.124.5 port 5001
[ ID] Interval       Transfer     Bandwidth
[  3]  0.0-10.0 sec  65.4 MBytes  54.9 Mbits/sec
ece792@ece792-Standard-PC-i440FX-PIIX-1996:~$ iperf -c 192.168.124.5 -t 10 -l 200
------------------------------------------------------------
Client connecting to 192.168.124.5, TCP port 5001
TCP window size: 2.50 MByte (default)
------------------------------------------------------------
[  3] local 192.168.124.5 port 44870 connected with 192.168.124.5 port 5001
[ ID] Interval       Transfer     Bandwidth
[  3]  0.0-10.0 sec   131 MBytes   110 Mbits/sec
ece792@ece792-Standard-PC-i440FX-PIIX-1996:~$ iperf -c 192.168.124.5 -t 10 -l 400
------------------------------------------------------------
Client connecting to 192.168.124.5, TCP port 5001
TCP window size: 2.50 MByte (default)
------------------------------------------------------------
[  3] local 192.168.124.5 port 44872 connected with 192.168.124.5 port 5001
[ ID] Interval       Transfer     Bandwidth
[  3]  0.0-10.0 sec   262 MBytes   219 Mbits/sec
ece792@ece792-Standard-PC-i440FX-PIIX-1996:~$ iperf -c 192.168.124.5 -t 10 -l 800
------------------------------------------------------------
Client connecting to 192.168.124.5, TCP port 5001
TCP window size: 2.50 MByte (default)
------------------------------------------------------------
[  3] local 192.168.124.5 port 44874 connected with 192.168.124.5 port 5001
[ ID] Interval       Transfer     Bandwidth
[  3]  0.0-10.0 sec   525 MBytes   440 Mbits/sec
ece792@ece792-Standard-PC-i440FX-PIIX-1996:~$ iperf -c 192.168.124.5 -t 10 -l 1600
------------------------------------------------------------
Client connecting to 192.168.124.5, TCP port 5001
TCP window size: 2.50 MByte (default)
------------------------------------------------------------
[  3] local 192.168.124.5 port 44876 connected with 192.168.124.5 port 5001
[ ID] Interval       Transfer     Bandwidth
[  3]  0.0-10.0 sec  1.02 GBytes   880 Mbits/sec
ece792@ece792-Standard-PC-i440FX-PIIX-1996:~$ iperf -c 192.168.124.5 -t 10 -l 3200
------------------------------------------------------------
Client connecting to 192.168.124.5, TCP port 5001
TCP window size: 2.50 MByte (default)
------------------------------------------------------------
[  3] local 192.168.124.5 port 44878 connected with 192.168.124.5 port 5001
[ ID] Interval       Transfer     Bandwidth
[  3]  0.0-10.0 sec  2.05 GBytes  1.76 Gbits/sec
ece792@ece792-Standard-PC-i440FX-PIIX-1996:~$ iperf -c 192.168.124.5 -t 10 -l 6400
------------------------------------------------------------
Client connecting to 192.168.124.5, TCP port 5001
TCP window size: 2.50 MByte (default)
------------------------------------------------------------
[  3] local 192.168.124.5 port 44880 connected with 192.168.124.5 port 5001
[ ID] Interval       Transfer     Bandwidth
[  3]  0.0-10.0 sec  4.10 GBytes  3.52 Gbits/sec
```

On the Server Side:

```
ece792@ece792-Standard-PC-i440FX-PIIX-1996:~$ iperf -s
------------------------------------------------------------
Server listening on TCP port 5001
TCP window size: 85.3 KByte (default)
------------------------------------------------------------
[  4] local 192.168.124.5 port 5001 connected with 192.168.124.5 port 44868
[ ID] Interval       Transfer     Bandwidth
[  4]  0.0-10.0 sec  65.4 MBytes  54.8 Mbits/sec
[  5] local 192.168.124.5 port 5001 connected with 192.168.124.5 port 44870
[  5]  0.0-10.0 sec   131 MBytes   110 Mbits/sec
[  4] local 192.168.124.5 port 5001 connected with 192.168.124.5 port 44872
[  4]  0.0-10.0 sec   262 MBytes   219 Mbits/sec
[  5] local 192.168.124.5 port 5001 connected with 192.168.124.5 port 44874
[  5]  0.0-10.0 sec   525 MBytes   440 Mbits/sec
[  4] local 192.168.124.5 port 5001 connected with 192.168.124.5 port 44876
[  4]  0.0-10.0 sec  1.02 GBytes   879 Mbits/sec
[  5] local 192.168.124.5 port 5001 connected with 192.168.124.5 port 44878
[  5]  0.0-10.0 sec  2.05 GBytes  1.76 Gbits/sec
[  4] local 192.168.124.5 port 5001 connected with 192.168.124.5 port 44880
[  4]  0.0-10.0 sec  4.10 GBytes  3.52 Gbits/sec
```

From the above results we can observe the throughputs for various sizes as follows:
100B- 54.8Mbits/sec
200B- 110Mbits/sec
400B- 219Mbits/sec
800B- 440Mbits/sec
1600B- 879Mbits/sec
3200B- 1.76Gbits/sec
6400B- 3.52Gbits/sec

From this we can observe that as the size doubles the throughput also doubles. This also means that the system bandwidth capacity can easily handle these message sizes.
This is because IP stack traversal and operations are done over packet objects. As overhead decreases with smaller packets and increases as the packet size increase which results in the throughput increasing as we have seen.