

CS9668

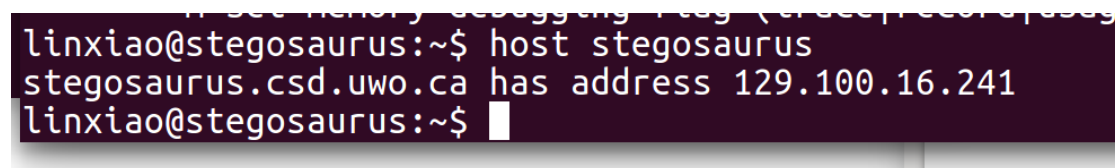
Assignment 1

Linxiao Wang
250888611

October 3, 2018

PROBLEM 1

Command: host stegosaurus

A terminal window with a dark background and light-colored text. The prompt is 'linxiao@stegosaurus:~\$'. The command entered is 'host stegosaurus'. The output is 'stegosaurus.csd.uwo.ca has address 129.100.16.241'. The prompt is followed by a cursor.

```
linxiao@stegosaurus:~$ host stegosaurus
stegosaurus.csd.uwo.ca has address 129.100.16.241
linxiao@stegosaurus:~$
```

Figure 0.1: Screen shot of “host stegosaurus”

- Symbolic name: stegosaurus.csd.uwo.ca
- IP address: 129.100.16.241 or 10000001 01100100 00010000 11110001
- Class: B. Since the IP address in binary starts with “10”, it means class B is assigned to the computers in the University’s network.
- Prefix: 129.100 or 10000001 01100100
- There can be $2^{16} = 65536$ computers in this network.

PROBLEM 2

I wrote a program that can generate a list of random IP addresses in the network 129.100.

```
1 #include <stdio.h>
2 #include <stdlib.h>
3 #include <time.h>
4
5 int main(){
6     time_t t;
7     srand((unsigned) time(&t));
8     FILE* output = fopen("list.txt", "w");
9     for(int i = 0; i < 100; i++){
10         fprintf(output, "129.100.%d.%d\n", rand()%256, rand()%256);
11     }
12     return 0;
13 }
```

Listing 1: C code for generating random IP addresses

I used the code 1 to generate a list of 100 IP addresses. Then I wrote a script 2 to **ping** all these IP, and counted how many of them were up.

```
1 #!/bin/bash
2 # Program name: pingip.sh
3 date
4 TEMPFILE=/tmp/tempfile.tmp
5 echo 0 > $TEMPFILE
6 cat list.txt | while read output
7 do
8     ping -c 1 -w 5 "$output">/dev/null
9     if [ $? -eq 0 ]; then
10         echo "node $output is up"
11         counter=$((cat $TEMPFILE) + 1)
12         echo $counter > $TEMPFILE
13     else
14         echo "node $output is down"
15     fi
16 done
17 cat $TEMPFILE
18 }
```

Listing 2: Script for ping all the IP

The complete list of IP is long so the following lists the ones that correspond to actual computer.
linxiao@stegosaurus:/data/d1/student/linxiao/CS9868 ./pingip.sh Thu Sep 27 14:13:39 EDT 2018
node 129.100.252.82 is up
node 129.100.177.10 is up
node 129.100.92.225 is up
node 129.100.75.165 is up
node 129.100.6.219 is up
node 129.100.176.110 is up

node 129.100.234.251 is up
node 129.100.76.255 is up
node 129.100.132.28 is up
node 129.100.85.22 is up
node 129.100.61.221 is up
node 129.100.225.23 is up

There are 12 out of 100 IPs responded to the ping command. So I assume the ratio of IPs that are used in the network is 12%. So my estimation of the size of the network is $65536 \times 12\% \approx 7864$ computers.

PROBLEM 3

Figure 0.2 shows the screenshot of the result of command “ip addr show”. My IP address in the CIDR notation is 129.100.18.146/22.

My IP address in binary is: 10000001 01100100 00010010 10010010/22

Network number: 10000001 01100100 000100

Computer number: 1010010010

There can be $2^{10} = 1024$ computers in the subnetwork to which my computer is connected.

```
linxiao@triceratops:~$ ip addr show
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: enp0s31f6: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000
    link/ether 6c:4b:90:1c:4d:ad brd ff:ff:ff:ff:ff:ff
    inet 129.100.18.146/22 brd 129.100.19.255 scope global dynamic enp0s31f6
        valid_lft 8794sec preferred_lft 8794sec
    inet6 fe80::5403:1fee:66bd:e05c/64 scope link
        valid_lft forever preferred_lft forever
3: wlp2s0: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc mq state DOWN group default qlen 1000
    link/ether 1c:4d:70:85:70:82 brd ff:ff:ff:ff:ff:ff
linxiao@triceratops:~$
```

Figure 0.2: ip addr show

PROBLEM 4

Figure 0.3 shows the result of maximum number of hops that I can find, the destination is the host of the website “www.gnu.org” the IP address of the host is “208.118.235.148”. Following

list shows the possible location of each server that I got from IP2Location.

```
linxiao@triceratops:~$ traceroute www.gnu.org
traceroute to www.gnu.org (208.118.235.148), 30 hops max, 60 byte packets
 1  vlan16-pub.edge-serv.netmgmt.uwo.pri (129.100.16.1)  0.606 ms  0.570 ms  0.575 ms
 2  edge-serv.core-uwo.netmgmt.uwo.pri (172.29.102.5)  0.564 ms  0.561 ms  0.557 ms
 3  uwo-core-ssb.netmgmt.uwo.pri (172.29.102.18)  1.416 ms  1.409 ms  1.401 ms
 4  199.71.2.185 (199.71.2.185)  1.399 ms  1.660 ms  1.639 ms
 5  172.16.1.101 (172.16.1.101)  2.545 ms  2.541 ms  2.533 ms
 6  172.16.1.2 (172.16.1.2)  2.226 ms  2.159 ms  2.152 ms
 7  te0-1-1-4.ccr31.yyz02.atlas.cogentco.com (38.104.251.1)  5.433 ms  5.430 ms  5.729 ms
 8  be2993.ccr21.cle04.atlas.cogentco.com (154.54.31.225)  12.442 ms  12.408 ms  12.404 ms
 9  be2878.ccr21.alb02.atlas.cogentco.com (154.54.26.130)  24.238 ms  24.246 ms  24.241 ms
10  be3599.ccr31.bos01.atlas.cogentco.com (66.28.4.238)  27.548 ms  27.398 ms  27.382 ms
11  ibr02-te-0-0-6.bst03.twdx.net (208.118.224.197)  32.437 ms  32.427 ms  32.050 ms
12  bbr02-ae-4-300.bos01.twdx.net (198.160.63.4)  32.421 ms  32.307 ms  32.297 ms
13  dcr01-be-10200.bsn05.twdx.net (198.160.62.185)  32.267 ms  32.179 ms  31.929 ms
14  FREE-SOFTWA.dcr01.bsn05.twdx.net (185.134.180.134)  31.911 ms  31.708 ms  31.689 ms
15  wildebeest.gnu.org (208.118.235.148)  32.209 ms  31.994 ms  31.977 ms
linxiao@triceratops:~$
```

Figure 0.3: Trace route result

- 1 129.100.16.1 London,ON,Canada
- 2 172.29.102.5 private IP address
- 3 172.29.102.18 private IP address
- 4 199.71.2.185 London,ON,Canada,N6A 5B7
- 5 172.16.1.101 private IP address
- 6 172.16.1.2 private IP address
- 7 38.104.251.1 Atlanta,Georgia,US,30301
- 8 154.54.31.225 Cleveland,Ohio,US,44101
- 9 154.54.26.130 Albany,New York,US,12201
- 10 66.28.4.238 Miami,Florida,US,33010
- 11 208.118.224.197 Boston,Massachusetts,US,02110
- 12 198.160.63.4 Boston,Massachusetts,US,02210
- 13 198.160.62.185 Boston,Massachusetts,US,02210
- 14 185.134.180.134 Somerville,Massachusetts,US,02143
- 15 208.118.235.148 Boston,Massachusetts,US,02110

PROBLEM 5

PART 1

If A chose a number between 1 to 3, there are two possible choice for B that is different from A. So the possibility of not having the second collision is $\frac{2}{3}$.

PART 2

For the first $k - 1$ rounds, the possibility of having a collision in each round is always $\frac{1}{3}$, so the possibility of having collision in the all $k - 1$ rounds is $(\frac{1}{3})^{k-1}$. In the k -th round, the possibility of not having a collision is $\frac{2}{3}$. So the possibility of exactly k rounds are needed before one of the computers can transmit is

$$(\frac{1}{3})^{k-1} \times \frac{2}{3} = \frac{2}{3^k}$$

PROBLEM 6

PROBLEM 7

Table 0.1 gives the routing table for router R_1 .

Destination	Nest hop
132.32.16.10	deliver direct
164.80.22.31	deliver direct
192.10.4.2	192.10.4.16
129.1.44.12	129.1.7.12
194.8.11.55	deliver direct
196.3.7.4	196.3.7.18

Table 0.1: Routing table for R_1 .

PROBLEM 8

Network 1 packet:

1. [**packet header:** MAC address of A, MAC address of R_1 , control bits of Network 1
packet data: {**datagram header:** IP address of A, IP address of B, rest of header. **datagram data:** 500 bytes}]

Network 2 packets:

1. [**packet header:** MAC address of R_1 , MAC address of R_2 , control bits of Network 2
packet data: {**datagram header:** IP address of A, IP address of B, rest of header. **datagram data:** 350 bytes}]
2. [**packet header:** MAC address of R_1 , MAC address of R_2 , control bits of Network 2

packet data: {**datagram header:** IP address of A, IP address of B, rest of header. **datagram data:** 150 bytes}}

Network 3 packets:

1. [**packet header:** MAC address of R_2 , MAC address of B, control bits of Network 3
packet data: {**datagram header:** IP address of A, IP address of B, rest of header. **datagram data:** 350 bytes}]
2. [**packet header:** MAC address of R_2 , MAC address of B, control bits of Network 2
packet data: {**datagram header:** IP address of A, IP address of B, rest of header. **datagram data:** 150 bytes}]