ARIZONA STATE UNIVERSITY

CSE 434, SLN 70516– **Computer Networks** — Fall 2022

Lab #3 Solution

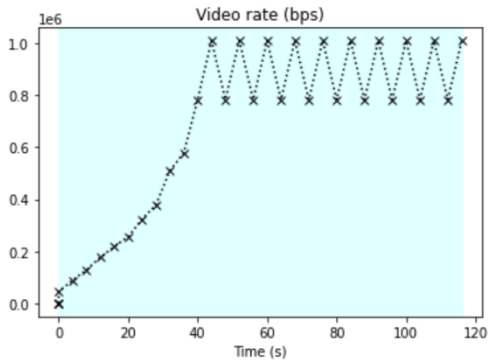
1 Adaptive Video using DASH

Exercise 1.1: *Experiment: Constant Bit Rate*

Follow the instructions in the section *Experiment: Constant Bit Rate*. For the lab report, use the Python notebook to visualize the logs produced from running the adaptive video streaming experiment with a constant bit rate. Include a snap shot that shows the video rate as a function of time. (It should look similar to the figure in that section.)

Did your video experience any time when the video was rebuffering and the playback was frozen? If yes, annotate the time intervals in your figure. (You may even want to play back the video on your own computer.)

The graph below depicts the DASH client's throughput over time at a constant data rate of 1000Kbit/sec:



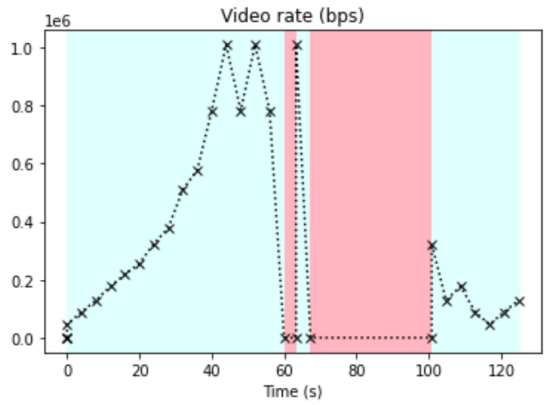
Despite conducting the experiment several times, the video to should not be rebuffed or freeze playing; as evidenced by the solid blue backdrop, the movie played constantly except for a brief initial loading phase.

Exercise 1.2: *Experiment: Constant Bit Rate with Interruption*

In Exercise 1.1, you may not have experienced any rebuffering so this experiment will force rebuffering by reducing the data rate. Now follow the instructions in the section *Experiment: Constant Bit Rate with Interruption*. For the lab report, once again use the Python notebook to visualize the logs produced from running the adaptive video streaming experiment with a constant bit rate, where you have reduced the data rate to force rebuffering. Include a snap shot that shows the video rate as a function of time.

Were you able to cause rebuffering? If yes, annotate time intervals in your figure. (You may even want to play back the video on your own computer.)

The graph below depicts the throughput of the DASH client utilizing the same constant bit rate, but this time, in the midst of the session, the network rate was reduced to a considerably lower 50Kbit/sec rate before being restored:



After adding the bottleneck of 50Kbit/sec to the router, video playback began to freeze. Because there was no bottleneck in place, the replay in the first 40 seconds of the session is similar to the first snapshot. Performing the rate-restricting command, slowed the video rate and produced freezing, as evidenced by the red backdrop. After restoring the previous pace, a minute later, the buffer refilled and the video resumed playing continuously. There was rebuffering from the 60s to the 100s, as seen by the red gaps in the screenshot.

Exercise 1.3: *Experiment: Mobile User*

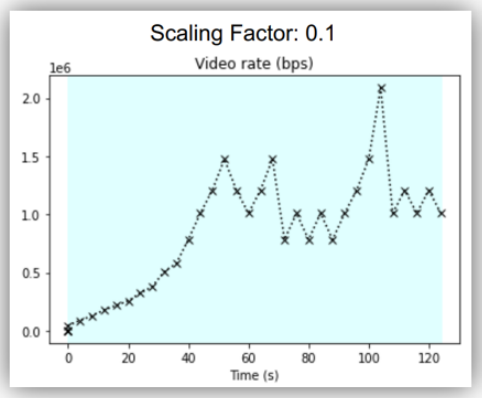
Follow the instructions in the section *Experiment: Mobile User* to experience adaptive video as a mobile user. This experiment uses network traces collected in the New York City metro area. With these traces, the data rate experienced by the DASH client in the experiment mimics the experience of traveling around NYC on bus, subway, or ferry.

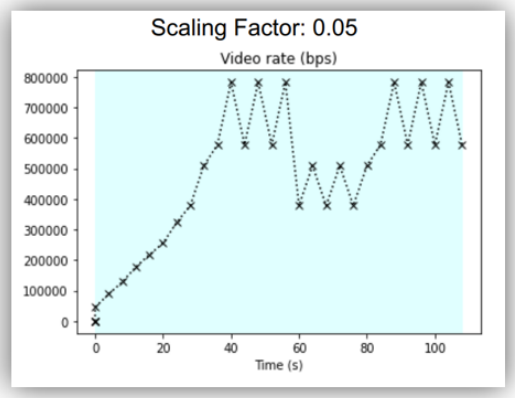
Select at least four (4) of the trace files to use and, for each, experiment with at least two (2) scaling factors. Plot the throughput as a function of time for each trace, and for the different scaling factors and include these in your lab report. (You should have a minimum of 8 figures.)

Describe your observations of your throughput for each trace. Is the throughput enough to stream the video? Can you see the impact of the scaling factor? Briefly explain.

The answer to this question will depend on the scaling factor that you choose but here we will give you general idea. We can use the trace files "Car/Car 1.csv," "Subway D Train/d1.csv," "Ferry/Ferry5.csv," and "Bus B57/bus57 1.csv" for the "Mobile User" experiment. We then measure the throughput for each of the trace files for around two minutes.

Car/Car_1.csv

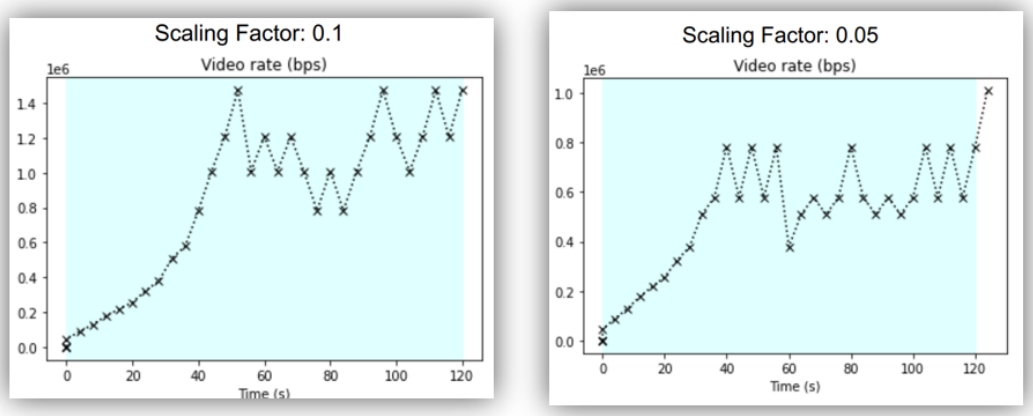




The graphs above show that the throughput was sufficient to stream the video with either a scaling factor

of 0.1 or 0.05, as there was no rebuffering in the midst of the playback, as indicated by the solid blue backdrop. The average throughput with a scaling factor of 0.1 was about 1Mbps, whereas the throughput with a scaling factor of 0.05 was around 500Kbps; this half of the throughput corresponds to the halving of the scaling factor.

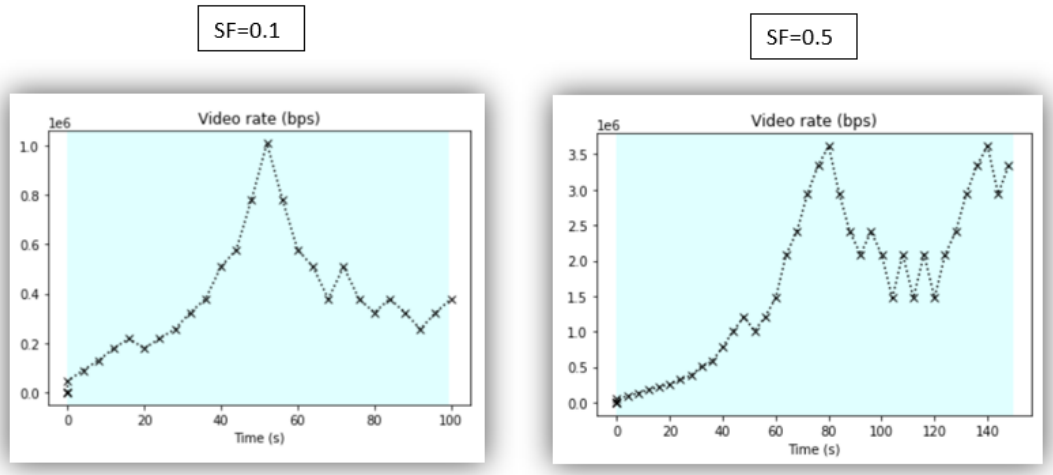
Subway_D_Train/d1.csv



The following graphs show that the throughput was sufficient to stream the video with either a scaling factor of 0.1 or 0.05, since no rebuffering occurred in the midst of the playback, as evidenced by the solid blue backdrop. With a scaling factor of 0.1, throughput steadied at an average of approximately 1.1Mbps, and with a scaling factor of 0.05, throughput stabilized at around 0.6Mbps, which corresponds to the shift in scaling factor. The throughput of both traces appears to change more than that of the vehicle trace above, which is consistent with the noisier dynamics of the train environment/trace.

M15_1.csv

Changing scale factor of 0.05 to 0.5 in next experiments.

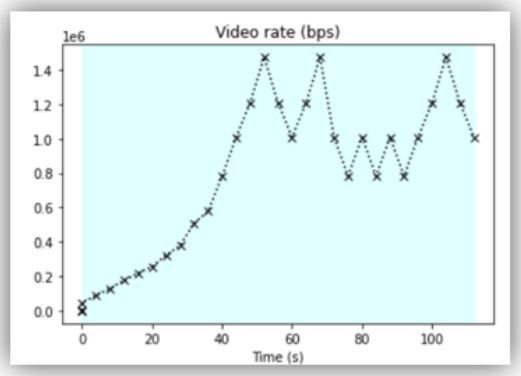


Throughput is sufficient to stream video without buffering in SF of 0.1. The throughput peaks at about 1.0e6 bps, which is about 3.5 times less than the 0.5 scale. Throughput ramps up almost exponentially until

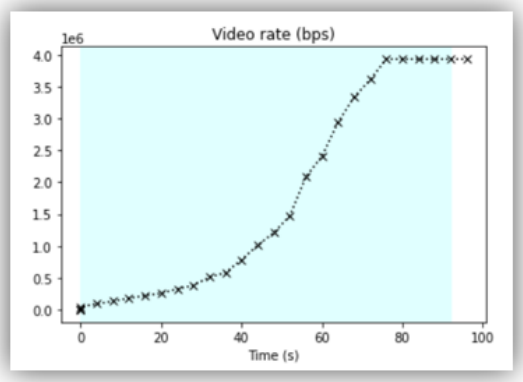
its peak, then begins to taper off a large amount to almost half of its peak, then stays around 4e6 bps, which is four times lower than the tapered off constant post-peak throughput of 1.75 Mbps given by the scale of 0.5.

Car_1.csv

SF=0.1



SF=0.5



It is sufficient to stream the movie without buffering. Throughput climbs exponentially to a high of 1.4 Mbps, then tapers down to fluctuate about 0.9 Mbps, then returns to the top after around 20 seconds of oscillations. The peak of the 0.1 scale is around 1.4 Mbps, which is approximately 2.5 times smaller than the high of the 0.5 scale, which is approximately 4.0 Mbps. After the first high, the 0.1 scale decreases and eventually recovers, while the 0.5 scale remains bouncing about the peak for an extended period of time (30 seconds)

2 Static Routing on the Racks

These solutions were obtained on a different rack than in BYENG 217, with interfaces named eth0 and eth1; otherwise, the results should be comparable to yours.

2.1 Configuring a Linux PC as an IP Router

Exercise 2.1 Network Setup

Use the saved data to answer the following questions:

- What is the output on PC A when the ping commands are issued?

◊ **Solution:**

```
ping -c 5 10.0.1.21
```

```
PING 10.0.1.21 (10.0.1.21) 56(84) bytes of data.
64 bytes from 10.0.1.21: icmp_seq=1 ttl=64 time=0.538 ms
64 bytes from 10.0.1.21: icmp_seq=2 ttl=64 time=0.280 ms
64 bytes from 10.0.1.21: icmp_seq=3 ttl=64 time=0.278 ms
64 bytes from 10.0.1.21: icmp_seq=4 ttl=64 time=0.278 ms
```

```
64 bytes from 10.0.1.21: icmp_seq=5 ttl=64 time=0.277 ms

--- 10.0.1.21 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4024ms
rtt min/avg/max/mdev = 0.277/0.330/0.538/0.104 ms
```

`ping -c 5 10.0.2.1` and `ping -c 5 10.0.3.41`

```
connect: Network is unreachable
```

- Which packets, if any, are captured by Wireshark?

  ◊ **Solution:**

```
No.  Time       Source            Destination       Protocol Info
 1 0.000000   00:04:75:ac:87:22   ff:ff:ff:ff:ff:ff         ARP     Who has 10.0.1.21?  Tell
              10.0.1.11
 2 0.000209   00:04:75:ad:04:cd   00:04:75:ac:87:22    ARP     10.0.1.21 is at 00:04:75:ad:04:cd
 3 0.000221   10.0.1.11           10.0.1.21          ICMP    Echo (ping) request
 4 0.000470   10.0.1.21           10.0.1.11          ICMP    Echo (ping) reply
 5 1.011533   10.0.1.11           10.0.1.21          ICMP    Echo (ping) request
 6 1.011797   10.0.1.21           10.0.1.11          ICMP    Echo (ping) reply
 7 2.012161   10.0.1.11           10.0.1.21          ICMP    Echo (ping) request
 8 2.012424   10.0.1.21           10.0.1.11          ICMP    Echo (ping) reply
 9 3.011534   10.0.1.11           10.0.1.21          ICMP    Echo (ping) request
10 3.011793   10.0.1.21           10.0.1.11          ICMP    Echo (ping) reply
11 4.011572   10.0.1.11           10.0.1.21          ICMP    Echo (ping) request
12 4.011835   10.0.1.21           10.0.1.11          ICMP    Echo (ping) reply
13 4.999708   00:04:75:ad:04:cd   00:04:75:ac:87:22    ARP     Who has 10.0.1.11?  Tell
              10.0.1.21
14 4.999728   00:04:75:ac:87:22   00:04:75:ad:04:cd    ARP     10.0.1.11 is at 00:04:75:ac:87:22
```

- Do you observe any ARP or ICMP packets? If so, what do they indicate?

  ◊ **Solution:**

  Yes. ARP and ICMP packets between PC A and PC B are observed. In the captured data, PC A sends an ARP request to get the MAC address of PC A and PC B replies to its ARP request. Once, PC A obtains MAC address of PC B, PC A sends ICMP request which is the actually `ping` packet. PC B replies to this query. Because five `ping` commands issue, there are five pairs of ICMP request and reply.

- Which destinations are not reachable? Explain.

  ◊ **Solution:**

  Destinations 10.0.2.1 and 10.0.3.41 are unreachable, this is because PC A realized these two hosts are not in the same subset as itself (by using the network prefix), which implies that direct delivery was impossible, and since there was no gateway for these two networks in PC A's routing table (and there was also no default gateway), so PC A concluded these two hosts are unreachable. Hence the message network unreachable.

Exercise 2.2 Configuring a Linux PC as an IP Router

- Include the saved output of the routing table. Explain the entries in the routing table and discuss the values of the fields for each entry.

  ◊ **Solution:**

  Routing table on PC A

```
Kernel IP routing table
Destination     Gateway         Genmask         Flags   MSS Window  irtt   Iface
10.0.1.0        0.0.0.0         255.255.255.0   U         0 0          0 eth0
10.0.2.0        10.0.1.21       255.255.255.0   UG        0 0          0 eth0
10.0.3.0        10.0.1.21       255.255.255.0   UG        0 0          0 eth0
127.0.0.0       0.0.0.0         255.0.0.0       U         0 0          0 lo
```

  Routing table on PC B

```
Kernel IP routing table
Destination     Gateway         Genmask         Flags   MSS Window  irtt   Iface
10.0.1.0        0.0.0.0         255.255.255.0     U         0 0          0 eth0
10.0.2.0        0.0.0.0         255.255.255.0     U         0 0          0 eth1
10.0.3.0        10.0.2.1        255.255.255.0     UG        0 0          0 eth1
127.0.0.0       0.0.0.0         255.0.0.0         U         0 0          0 lo
```

  Routing table on PC D

```
Kernel IP routing table
Destination     Gateway         Genmask         Flags      MSS Window  irtt  Iface
10.0.1.0        10.0.3.1        255.255.255.0   UG          0 0          0 eth0
10.0.2.0        10.0.3.1        255.255.255.0   UG          0 0          0 eth0
10.0.3.0        0.0.0.0         255.255.255.0   U           0 0          0 eth0
127.0.0.0       0.0.0.0         255.0.0.0       U           0 0          0 lo
```

- Explain the entries in the routing table and discuss the values of the fields for each entry.
  - The first field is the destination, which can be a specific host, a subnet, the loopback, or a default route (anything else).
  - The second field is the gateway to forward packets intended for the destination.
  - The third field is the netmask for the corresponding destination, with 255.255.255.255 for a host and 0.0.0.0 for a default route.
  - The fourth field is the flags (U = route is up; G = use gateway).
  - The fifth field is the Maximum Segment Size (bytes) for TCP packets.
  - The sixth field is the Windows size for TCP connections over this route.
  - The seventh field is the initial route trip time for TCP connections over this route.
  - The eighth field is the interface in which this route is connected to.

2.4 Configuring a Cisco Router

Exercise 2.3 Configuring IP Interfaces on a Cisco Router

- Include the output from Step 3 in your lab report.

◊ **Solution:**

Output of show interface:

```
router1#show interface
FastEthernet0/0 is up, line protocol is up
  Hardware is AmdFE, address is 0007.50d0.9d21 (bia 0007.50d0.9d21)
  Internet address is 10.0.2.1/24
  MTU 1500 bytes, BW 100000 Kbit, DLY 100 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation ARPA, loopback not set
  Keepalive set (10 sec)
  Half-duplex, 100Mb/s, 100BaseTX/FX
  ARP type: ARPA, ARP Timeout 04:00:00
  Last input 00:13:25, output 00:00:00, output hang never
  Last clearing of "show interface" counters never
  Queueing strategy: fifo
  Output queue 0/40, 0 drops; input queue 0/75, 0 drops
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
     239 packets input, 20309 bytes
     Received 9 broadcasts, 0 runts, 0 giants, 0 throttles
     0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored
     0 watchdog
     0 input packets with dribble condition detected
     16432 packets output, 1129006 bytes, 0 underruns(0/0/0)
     0 output errors, 0 collisions, 1 interface resets
     0 babbles, 0 late collision, 0 deferred
     0 lost carrier, 0 no carrier
     0 output buffer failures, 0 output buffers swapped out
Serial0/0 is administratively down, line protocol is down
  Hardware is DSCC4 Serial
  MTU 1500 bytes, BW 2048 Kbit, DLY 20000 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation HDLC, loopback not set
  Keepalive set (10 sec)
  Last input never, output never, output hang never
  Last clearing of "show interface" counters 1d19h
  Queueing strategy: fifo
  Output queue 0/40, 0 drops; input queue 0/75, 0 drops
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
     0 packets input, 0 bytes, 0 no buffer
     Received 0 broadcasts, 0 runts, 0 giants, 0 throttles
     1 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 1 abort
     0 packets output, 0 bytes, 0 underruns
     0 output errors, 0 collisions, 1 interface resets
     0 output buffer failures, 0 output buffers swapped out
     0 carrier transitions
     DCD=down  DSR=down  DTR=down  RTS=down  CTS=down

FastEthernet0/1 is up, line protocol is up
  Hardware is AmdFE, address is 0007.50d0.9d22 (bia 0007.50d0.9d22)
```

```
  Internet address is 10.0.3.1/24
  MTU 1500 bytes, BW 100000 Kbit, DLY 100 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation ARPA, loopback not set
  Keepalive set (10 sec)
  Half-duplex, 100Mb/s, 100BaseTX/FX
  ARP type: ARPA, ARP Timeout 04:00:00
  Last input 00:13:54, output 00:00:03, output hang never
  Last clearing of "show interface" counters never
  Queueing strategy: fifo
  Output queue 0/40, 0 drops; input queue 0/75, 0 drops
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
     230 packets input, 17925 bytes
     Received 4 broadcasts, 0 runts, 0 giants, 0 throttles
     0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored
     0 watchdog
     0 input packets with dribble condition detected
     16375 packets output, 1124730 bytes, 0 underruns(0/0/0)
     0 output errors, 0 collisions, 1 interface resets
     0 babbles, 0 late collision, 2 deferred
     0 lost carrier, 0 no carrier
     0 output buffer failures, 0 output buffers swapped out
Serial0/1 is administratively down, line protocol is down
  Hardware is DSCC4 Serial
  MTU 1500 bytes, BW 2048 Kbit, DLY 20000 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation HDLC, loopback not set
  Keepalive set (10 sec)
  Last input never, output never, output hang never
  Last clearing of "show interface" counters never
  Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 0
  Queueing strategy: weighted fair
  Output queue: 0/1000/64/0 (size/max total/threshold/drops)
     Conversations  0/0/256 (active/max active/max total)
     Reserved Conversations 0/0 (allocated/max allocated)
     Available Bandwidth 1536 kilobits/sec
  5 minute input rate 0 bits/sec, 0 packets
```

Output of show running-config:

```
!
version 12.1
no service single-slot-reload-enable
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
!
hostname Router1
!
logging rate-limit console 10 except errors
enable secret 5 $1$PhxU$UM0qKAbeZYXfvk4IgOkjV/
!
ip subnet-zero
!
!
no ip finger
```

```
no ip domain-lookup
!
!
!
!
interface FastEthernet0/0
 ip address 10.0.2.1 255.255.255.0
 duplex auto
 speed auto
 no cdp enable
!
interface Serial0/0
 no ip address
 shutdown
 no fair-queue
 clockrate 2000000
 no cdp enable
!
interface FastEthernet0/1
 ip address 10.0.3.1 255.255.255.0
 duplex auto
 speed auto
 no cdp enable
!
interface Serial0/1
 no ip address
 shutdown
 clockrate 2000000
 no cdp enable
!
ip classless
no ip http server
!
no cdp run
!
line con 0
 transport input none
line aux 0
line vty 0 4
!
end
```

Exercise 2.4 Setting Static Routing Table Entries on a Cisco Router

- Include the saved output of the routing table from Steps 1 and 2. Explain the fields of the routing table entries of the Cisco router. Explain how the routing table has changed from Step 1 to Step 3.

◊ **Solution:**

Routing table of Router A before static entry was added:

```
router1#show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       * - candidate default, U - per-user static route, o - ODR
       P - periodic downloaded static route
```

```
Gateway of last resort is not set
    10.0.0.0/24 is subnetted, 2 subnets
C      10.0.2.0 is directly connected, FastEthernet0/0
C      10.0.3.0 is directly connected, FastEthernet0/1
```

This output shows the routing table of the router A before a static route to the network 10.0.1.0/24 is added. We can see there are two routes, one route to "10.0.2.0" and the other route to "10.0.3.0". These two routes are both direct routes (denoted by the "C" at the beginning of the route entries) and are connected by the interfaces Ethernet0 and Ethernet1 respectively. Since the default gateway for this router is not set therefore, it says "Gateway of last resort is not set"

Routing Table of Router A:

```
router1#show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       * - candidate default, U - per-user static route, o - ODR
       P - periodic downloaded static route
Gateway of last resort is not set

    10.0.0.0/24 is subnetted, 3 subnets
C      10.0.2.0 is directly connected, FastEthernet0/0
C      10.0.3.0 is directly connected, FastEthernet0/1
S      10.0.1.0 [1/0] via 10.0.2.22
```

Static routing entry 10.0.1.0 is added. The subnet 10.0.1.0/24 has the next-hop 10.0.2.22. **S** indicates this entry is a static entry.

- The output indicates that there are 3 subnets, and each has a netmask of 24.
- The first field is the codes: C = connected; S = static. The first two entries are directly connected to local subnets, and the third entry is the static that has just been added.
- The second field is the destination (host, subnet, or default route), which has a description of whether it is directly connected or via a gateway, and the corresponding interface that it is connected to.

2.5 Finalizing And Exploring the Router Configuration

Exercise 2.5 Testing Routes with Traceroute

- Use the Wireshark output and the previously saved routing table to explain the operation of `traceroute`.

◊ **Solution:**

Traceroute output from PC A to PC D:

```
traceroute to 10.0.3.41 (10.0.3.41), 30 hops max, 38 byte packets
 1  10.0.1.21 (10.0.1.21)  1.968 ms  0.355 ms  0.202 ms
 2  10.0.2.1 (10.0.2.1)  1.642 ms  0.559 ms  0.475 ms
```

```
 3  10.0.3.41 (10.0.3.41)  0.442 ms  0.339 ms  0.380 ms
```

Routing table on PC A

```
Kernel IP routing table
Destination     Gateway         Genmask         Flags   MSS Window  irtt Iface
10.0.1.0        0.0.0.0         255.255.255.0   U         0 0          0 eth0
10.0.2.0        10.0.1.21       255.255.255.0   UG        0 0          0 eth0
10.0.3.0        10.0.1.21       255.255.255.0   UG        0 0          0 eth0
127.0.0.0       0.0.0.0         255.0.0.0       U         0 0          0 lo
```

Routing table on PC B

```
Kernel IP routing table
Destination     Gateway         Genmask         Flags   MSS Window  irtt Iface
10.0.1.0        0.0.0.0         255.255.255.0   U         0 0          0 eth0
10.0.2.0        0.0.0.0         255.255.255.0   U         0 0          0 eth1
10.0.3.0        10.0.2.1        255.255.255.0   UG        0 0          0 eth1
127.0.0.0       0.0.0.0         255.0.0.0       U         0 0          0 lo
```

Routing table on PC D

```
Kernel IP routing table
Destination     Gateway         Genmask         Flags   MSS Window  irtt Iface
10.0.1.0        10.0.3.1        255.255.255.0   UG        0 0          0 eth0
10.0.2.0        10.0.3.1        255.255.255.0   UG        0 0          0 eth0
10.0.3.0        0.0.0.0         255.255.255.0   U         0 0          0 eth0
127.0.0.0       0.0.0.0         255.0.0.0       U         0 0          0 lo
```

Routing table on Router A

```
Router1#show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, * - candidate default
       U - per-user static route, o - ODR

Gateway of last resort is not set

    10.0.0.0/24 is subnetted, 3 subnets
C      10.0.2.0 is directly connected, Ethernet0
C      10.0.3.0 is directly connected, Ethernet1
S      10.0.1.0 [1/0] via 10.0.2.22
```

This output indicates the route to reach PC D from PC A. The packets traverse PC A->PC B->Router A->PC D.

```
    No. Time      Source           Destination        Protocol   Info
     1 0.000000   10.0.1.11          10.0.3.41          UDP     Source port: 32771
            Destination port: 33435
     2 0.000254    10.0.1.21            10.0.1.11            ICMP     Time-to-live
            exceeded
     3 0.001235   10.0.1.11          10.0.3.41          UDP     Source port: 32771
            Destination port: 33436
     4 0.001424    10.0.1.21            10.0.1.11            ICMP     Time-to-live
```

```
           exceeded
     5 0.001613   10.0.1.11          10.0.3.41          UDP     Source port: 32771
           Destination port: 33437
     6 0.001798    10.0.1.21            10.0.1.11            ICMP     Time-to-live
           exceeded
     7 0.002043   10.0.1.11          10.0.3.41          UDP     Source port: 32771
           Destination port: 33438
     8 0.004075    10.0.2.1             10.0.1.11            ICMP     Time-to-live
           exceeded
     9 0.004452   10.0.1.11          10.0.3.41          UDP     Source port: 32771
           Destination port: 33439
    10 0.006447    10.0.2.1             10.0.1.11            ICMP     Time-to-live
           exceeded
    11 0.006631   10.0.1.11          10.0.3.41          UDP     Source port: 32771
           Destination port: 33440
    12 0.008946    10.0.2.1             10.0.1.11            ICMP     Time-to-live
           exceeded
    13 0.009175   10.0.1.11          10.0.3.41          UDP     Source port: 32771
           Destination port: 33441
    14 0.012204    10.0.3.41            10.0.1.11            ICMP     Destination
           unreachable
    15 0.012650   10.0.1.11          10.0.3.41          UDP     Source port: 32771
           Destination port: 33442
    16 0.013575    10.0.3.41            10.0.1.11            ICMP     Destination
           unreachable
    17 0.013759   10.0.1.11          10.0.3.41          UDP     Source port: 32771
           Destination port: 33443
    18 0.014700    10.0.3.41            10.0.1.11            ICMP     Destination
           unreachable
    19 4.999294     00:04:75:ac:87:22      00:04:75:ad:04:cd      ARP       Who has
           10.0.1.21?  Tell 10.0.1.11
    20 4.999486   00:04:75:ad:04:cd    00:04:75:ac:87:22    ARP     10.0.1.21 is at
           00:04:75:ad:04:cd
```

From the Wireshark output we can see that traceroute program generates one UDP packet to the destination each time. At the first iteration, traceroute sets the TTL field of the UDP packet to be 1. When the packet reach the first router in the route, the TTL field will be reduced by 1 and this gives a 0, since the packet cannot be forwarded, the router replies with an ICMP error message to the sender saying “Time-to-live exceeded”, so the sender now knows who the first router in the route is. The traceroute program then generates another UDP packet with TTL=1, again the ICMP error message will be sent back, traceroute program will then generate a third UDP packet, still with TTL=1, and ICMP error message will again be sent back. In each iteration, traceroute will send 3 UDP packets with the same TTL value and this results in 3 ICMP errors, in this way traceroute measures a router three times in a single iteration. In the second iteration traceroute will set the TTL field of the UDP packets to be 2, now with the same process the second hop in the route will be revealed during this iteration. Traceroute keep increasing the TTL value by 1 for each iteration until the final destination is reached. When the UDP packet generated by traceroute arrives at the final destination, the “Time-to-live exceeded” will not be generated since no more forwarding is needed. Therefore traceroute can no longer use the previous trick on the final destination. But, traceroute manages to force the destination to reply by setting the destination UDP port of the UDP packets to an unused port on the destination host, therefore whenever the UDP packets reach the final destination, the destination is forced to send back an ICMP “Destination unreachable” message. When traceroute receives this message, it knows the final destination has been reached and will not trigger any further iteration, in this way the whole route to a destination is revealed by traceroute.

Exercise 2.6 Observe MAC Addresses at a Router

- Determine the source and destination addresses in the Ethernet and IP headers for the ICMP Echo Request messages that were captured at PC A.

  ◊ **Solution:**

  The source address in the Ethernet is 00:04:75:ac:87:22. (The MAC address will differ depending on the System /Rack used.)

  The destination address in the Ethernet is 00:04:75:ad:04:cd. (The MAC address will differ depending on the System /Rack used.)

  The source address in the IP header for the ICMP Echo Request message is 10.0.1.11.

  The destination address in the IP header is 10.0.3.41

- Determine the source and destination addresses in the Ethernet and IP headers for the ICMP Echo Request messages that were captured at PC D.

  ◊ **Solution:**

  The addresses are 00:10:7b:81:8c:8f , 00:04:75:ac:88:5d, 10.0.1.11, 10.0.3.41 respectively. (The MAC address will differ depending on the System /Rack used.)

- Use your previous answers to explain how the source and destination Ethernet and IP addresses are changed when a datagram is forwarded by a router.

  ◊ **Solution:**

  As illustrated above, the IP address of the packet does not change from hop to hop. The source and destination IP address of a packet will always be the originating host and the final destination host, respectively. However, the hardware (MAC) address does change. The hardware address of any intermediate node is not needed. Only the hardware address of the next hop is important. Therefore, ARP requests are sent in order to obtain the MAC address of the next hop. In our case for example, when PC A wishes to send a packet to PC D, PC A knows that only PC B can forward the packet from its routing table, so it uses ARP to find PCB's MAC address and then changes the destination MAC address of the Ethernet frame that encapsulates the IP packet to this MAC address. When PC B receives the Ethernet frame, it takes out the IP packet and inspects the destination IP address, PC B realizes from its routing table that this IP packet is to be forwarded to Router A so it again uses ARP to find the MAC address of Router A, now the IP packet is stuffed into a new Ethernet frame with source and destination MAC addresses being that of the PC B and Router A. When Router A receives the Ethernet frame, it will check the IP packet and then it will realize the destination is directly connected to itself. Router A will perform the final ARP lookup and stuff the IP packet to the Ethernet frame with the source and destination MAC addresses being that of itself and PC D respectively. Now PC D will receive the IP packet. Thus, as we see, Ethernet MAC addresses always change when the IP packet transits from one subnet to another, but the ***source and destination IP addresses never change***.

Exercise 2.7 Multiple Matches in the Routing Table

- Use the saved output to indicate the number of matches for each of the preceding IP addresses. Explain how PC A resolves multiple matches in the routing table. Include only relevant output data in your report to support your analysis of the data.

◊ **Solution:**

```
Kernel IP routing table
Destination     Gateway         Genmask         Flags  MSS Window  irtt Iface
10.0.3.9        10.0.1.81       255.255.255.255 UGH      0 0        0 eth0
10.0.1.0        0.0.0.0         255.255.255.0   U        0 0        0 eth0
10.0.2.0        10.0.1.21       255.255.255.0   UG       0 0        0 eth0
10.0.3.0        10.0.1.21       255.255.255.0   UG       0 0        0 eth0
10.0.0.0        10.0.1.71       255.255.0.0     UG       0 0        0 eth0
127.0.0.0       0.0.0.0         255.0.0.0       U        0 0        0 lo
```

The IP address in question is compared to each entry in the list in ascending order, and by performing a bit-wise AND with the corresponding netmask. A host-specific routing entry is always placed at the top, and a default routing entry is always placed at the bottom. The netmask of a host-specific entry is set to 255.255.255.255, and thus will give an exact match. Any network entry will have a netmask defined to match the network-prefix of the IP address. The default entry will have an IP address of 0.0.0.0 and netmask of 0.0.0.0. Since all bits are set to zero, any address being compared will match.

As we can see there are three matches for “10.0.3.9”, they are

“10.0.3.9/32” , “10.0.3.0/24” and “10.0.0.0/16”.

There are two matches for “10.0.3.14”, they are → “10.0.3.0/24” and “10.0.0.0/16”

There is one match for “10.0.4.1”, it is →“10.0.0.0/16”

There are 2 matches for each of the “10.0.1.61”,”10.0.1.71” and”10.0.1.81”. They are “10.0.1.0/24” and “10.0.0.0/16”.

1) When PC A pings 10.0.3.9, PC A first searches its routing table about the information how to reach 10.0.3.9. A match is found at the first line (gateway 10.0.1.81) and no further checking is needed. Then PC A tries to obtain the MAC address of 10.0.1.81 by sending ARP request. However, gateway 10.0.1.81 does not exist and no ARP reply is received. The ping command times-out after two successive ARP requests are sent. PC A decides the destination 10.0.3.9 is not reachable.

2) For pinging to 10.0.3.14, the first match is the entry for network 10.0.3.0. The matching process is stopped here and again, gateway 10.0.1.61 does not exist and no ARP reply is received. Also, the destination 10.0.3.14 is not reachable.

3) For pinging to 10.0.4.1, the match is found at the entry in the penultimate line for network 10.0.0.0 . Since gateway 10.0.1.71 also does not exist, no ARP reply is received. Not reachable.

Exercise 2.8 Default Routes

- What is the output on PC A when the `ping` command is issued?

  ◊ **Solution:**

```
[root@PC1 root]# ping -c 5 10.0.10.110
PING 10.0.10.110 (10.0.10.110) 56(84) bytes of data.
From 10.0.2.1 icmp_seq=1 Destination Host Unreachable
From 10.0.2.1 icmp_seq=2 Destination Host Unreachable
From 10.0.2.1 icmp_seq=3 Destination Host Unreachable
From 10.0.2.1 icmp_seq=4 Destination Host Unreachable
From 10.0.2.1 icmp_seq=5 Destination Host Unreachable

--- 10.0.10.110 ping statistics ---
5 packets transmitted, 0 received, +5 errors, 100% packet loss, time 4043ms
```

- Determine how far the ICMP Echo Request message travels?

  ◊ **Solution:** The ICMP Echo Request message travels among PC A, PC B and Router A.

- Which, if any, ICMP Echo Reply message returns to PC A?

  ◊ **Solution:** An ICMP error message (Type 3 – destination unreachable, Code 1 – host unreachable) is returned to PC A from Router A.