Name_______________ Student ID____________________ Department/Year__________

3rd Examination

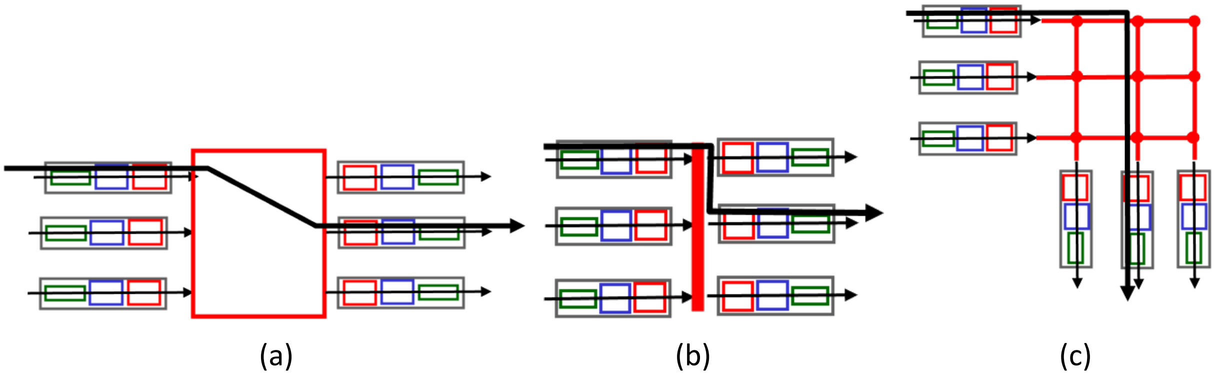
Introduction to Computer Networks (Online)
Class#: EE 4020, Class-ID: 901E31110
Spring 2022

10:20-12:10 Wednesday
June 01, 2022

Cautions

1. There are in total 100 points to earn. You have 90 minutes to answer the questions. Skim through all questions and start from the questions you are more confident with.
2. Use only English to answer the questions. Misspelling and grammar errors will be tolerated, but you want to make sure with these errors your answers will still make sense.

1. (ch42, 5pt) Below are 3 forms of switching fabric in a router.



(1) Which one is the memory switching fabric? (1pt)
(2) Which one is the bus switching fabric? (1pt)
(3) Which one is the crossbar switching fabric? (1pt)
(4) Which one is the slowest and why? (1pt)
(5) Which one is the fastest and why? (1pt)

Sample Solution:

(1) (a)
(2) (b)
(3) (c)
(4) (a) because writing/reading packets into/from memory is slow.
(5) (c) because packets can potentially travel in parallel paths thru the crossbar.

2. (ch42, 7pt) Forwarding table is also called routing table. One can issue a Unix system command – netstat – to show the network status of a machine. The -r flag gives particularly the status of the routing table. The -n flag suppresses the reverse DNS lookup and avoids acquiring the hostname of each IP address appearing in the routing table.

Now log on the PA workstation and create the exam3 directory if you have not yet done so. Go to the exam3 directory. Execute the following command. Look into netstat.txt. You should find two routing tables, “Internet:” for IPv4 packet lookup and “Internet6:” for IPv6. In each table there are columns of information, such as Destination, Gateway, Flags, Netif, and etc.

```
$ netstat -rn > netstat.txt
```

For this problem set, one must complete (1)-(2) to earn credits for (3)-(5).

(1) Execute the command above and leave the netstat.txt in your exam3 directory for grading. (2pt)

(2) Search the Web. Take a photo of the screen showing the description that tells what information the Netif column provides. (2pt)

(3) For a packet having destination address 127.0.0.1, which network interface will the packet go? (1pt)

(4) For a packet having destination address 140.112.42.220, which network interface will the packet go? (1pt)

(5) For a packet having destination address 192.168.0.220, which network interface will the packet go? (1pt)

Sample Solution:

(3) lo0

(4) en0

(5) en2

3. (ch43, 4pt) IPv4 addressing being hierarchical provides several benefits. Let's see if you can elaborate and explain the following four.

(1) Why does hierarchical addressing ease IP address management? (1pt)

(2) Why does hierarchical addressing save memory space? (1pt)

(3) Why does hierarchical addressing speed up forwarding? (1pt)

(4) Why does hierarchical addressing save network bandwidth for route advertisements? (1pt)

Sample Solution:

Whatever makes sense. Below are some possibilities.

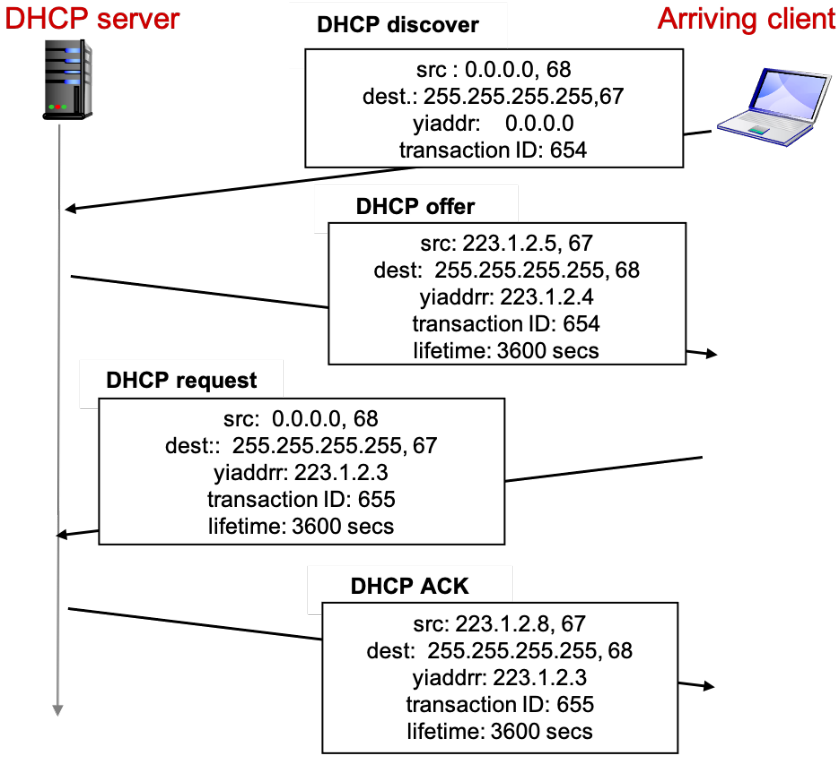
(1) The network admin can break the IP address space into subspaces easily using a longer prefix.

(2) Having 2^32 entries in the forwarding table vs. having few entries of prefixes. The latter reduces the # of entries and reduces memory requirement to keep the forwarding table.

(3) Having fewer entries in the forwarding table, the lookup process checks fewer entries. Checking 2^32 entries vs. few entries of prefixes. The latter is faster asymptotically.

(4) The routing protocol (BGP) advertises for each prefix (# of subnets) instead of each IP address (2^32). The traffic overhead is a lot lower in the former.

4. (ch43, 6pt) Check out the following DHCP message exchange. See if you can answer the related questions below.



DHCP discover
src : 0.0.0.0, 68
dest.: 255.255.255.255,67
yiaddr: 0.0.0.0
transaction ID: 654

DHCP offer
src: 223.1.2.5, 67
dest: 255.255.255.255, 68
yiaddrr: 223.1.2.4
transaction ID: 654
lifetime: 3600 secs

DHCP request
src: 0.0.0.0, 68
dest:: 255.255.255.255, 67
yiaddrr: 223.1.2.3
transaction ID: 655
lifetime: 3600 secs

DHCP ACK
src: 223.1.2.8, 67
dest: 255.255.255.255, 68
yiaddrr: 223.1.2.3
transaction ID: 655
lifetime: 3600 secs

(1) Tell the IP address of the DHCP server. (1pt)
(2) Tell the IP address for the arriving client. (1pt)
(3) Which layer protocol is DHCP and why? (2pt)
(4) Does DHCP send messages over TCP or UDP and why? (2pt)

Sample Solution:

(1) 223.1.2.5 or 223.1.2.8
(2) 223.1.2.4 or 223.1.2.3
(3) Application layer. To move complexity to the edge (much like DNS).
(4) UDP. Can’t start a TCP connection without the source IP address and to get IP address assigned ASAP (much like DNS).

5. (ch43, 4pt) The public opinion of NAT is mixed – love and hate.
(1) One of the reasons the public loves about NAT is that it is more secure. Why does NAT provide security? (2pt)
(2) One of the reasons the public hates about NAT is that it’s controversial. What’s so controversial about NAT? (2pt)

Sample Solution:

Whatever makes sense. Below are some possibilities.

(1) Machines from outside the NAT box won't be able to reach the machines inside running on the fake addresses.

(2) NAT changes the IP address and port number fields in a packet. This operation crosses two layers violating the end-to-end argument, where we're supposed to keep the protocols in one layer to keep the protocol stack modular.
Or, P2P protocols that send messages directly from peer to peer (using each other's IP addresses) will not work when peers are behind NAT. The P2P protocol designer can no longer design independently of other layers. This violates the layered architecture of the Internet's system software stack and slows down the development/evolution of protocols in each layer.

6. (ch43, 5pt) The main difference between IPv4 and IPv6 is the address space. The address field is expanded from 32-bit to 128-bit long. Let's examine the impact of the much larger address space to the benefits of hierarchical addressing.

(1) Compared to IPv4, will IPv6 address management be easier? Why or why not? (1pt)
(2) Compared to IPv4, will IPv6 save more memory space? Why or why not? (1pt)
(3) Compared to IPv4, will IPv6 speed up forwarding more? Why or why not? (1pt)
(4) Compared to IPv4, will IPv6 save more network bandwidth? Why or why not? (1pt)
(5) IPv6 has been out for decades. The adoption rate is still low. Why so slow? (1pt)

Sample Solution:

Whatever makes sense. Below are some possibilities.

(1) The same. The technique of using a longer prefix is the same.
Easier, more addresses to allocate to subgroups in an organization.
Harder, more addresses to track.

(2) The same as long as the number of prefixes remain the same.
Higher as the number of prefixes are potentially higher.
Relative to IPv6 having 2^128 entries in the worst case, the memory saving rate of IPv6 is potentially higher.

(3) Slower. The prefixes are potentially longer and taking more time to match per entry.
Depends. Each subnet might just be bigger when the prefix length and # of prefixes remain the same.

(4) Similar to the answer to (2)

(5) Reconfiguring the routers requires effort. Unless there's significant incentive

(customer complaints or $$) in exchange of the effort, salarymen normally don’t move the task higher in the todo list.

7. (ch44, 4pt) Below is the flow table entry format.

Switch Port	MAC src	MAC dst	Eth type	VLAN ID	IP Src	IP Dst	IP Prot	TCP sport	TCP dport	Action	Priority

Configure the entry(s) to achieve each of the functions below.

(1) Traditional forwarding: forward packets going to subnet 140.112.42.0/24 via network interface 1. (1pt)

(2) Generalized forwarding: forward packets from subnet 192.168.0.0/24 to subnet 140.112.42.0/24 port 80 via network interface 2. (1pt)

(3) Firewall: drop packets from subnet 192.168.0.0/24 to any web servers. Forward other packets from subnet 192.168.0.0/24 out via network interface 3. (1pt)

(4) NAT: translate packets with source host 192.168.0.11 source port 52000 to 140.112.42.220 and 12000. Also, translate packets with destination host 140.112.42.220 destination port 12000 back to 192.168.0.11 and 52000. (1pt)

Sample Solution:

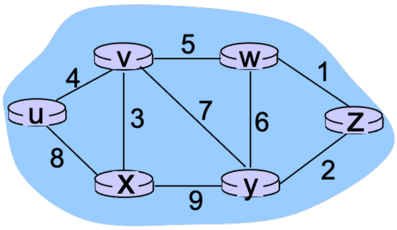
(1) Matching rule: IP Dst = 140.112.42.*
Action: forward(1)

(2) Matching rule: IP Src=192.168.0.* && IP Dst=140.112.42.* && TCP dport=80
Action: forward(2)

(3) Entry#1--
Matching rule: IP Src=192.168.0.* && TCP dport=80
Action: drop
Priority: high
Entry #2--
Matching rule: IP Src=192.168.0.*
Action: forward(3)
Priority: low

(4) Entry#1--
Matching rule: IP Src=192.168.0.11 && TCP sport=52000
Action: IP Src=140.112.42.220, TCP sport=12000
Entry #2--
Matching rule: IP Dst=140.112.42.220 && TCP dport=12000
Action: IP Dst=192.168.0.11 TCP dport=52000

8. (ch52, 6pt) Run Dijkstra algorithm on the following graph.



Start from including node u to N’. Develop the table until the algorithm terminates. Step 0 is the initialization step.

Step	N'	D(v),p(v)	D(w),p(w)	D(x),p(x)	D(y),p(y)	D(z),p(z)
0	u					
1						
2						
3						
4						
5						

(1) Tell N’ and (D,p) pairs for node v, w, x, y, z in step 0. (1pt)

(2) Tell N’ and (D,p) pairs for node v, w, x, y, z in step 1. (1pt)

(3) Tell N’ and (D,p) pairs for node v, w, x, y, z in step 2. (1pt)

(4) Tell N’ and (D,p) pairs for node v, w, x, y, z in step 3. (1pt)

(5) Tell N’ and (D,p) pairs for node v, w, x, y, z in step 4. (1pt)

(6) Tell N’ and (D,p) pairs for node v, w, x, y, z in step 5. (1pt)

Sample Solution:

(1) N’={u}, (4,u) (inf) (8,u) (inf) (inf)

(2) N’={uv}, (-) (9,v) (7,v) (11,v) (inf)

(3) N’={uvx}, (-) (9,v) (-) (11,v) (inf)

(4) N’={uvxw}, (-) (-) (-) (11,v) (10,w)

(5) N’={uvxwz}, (-) (- (-) (11,v) (-)

(6) N’={uvxwzy}, (-) (-) (-) (-) (-)

9. (ch52, 6pt) Continue from Problem 8. To generate the graph for Dijkstra computation, each node sends the link state reports.

(1) Tell the link state reports sent by node u. (1pt)

(2) Tell the link state reports sent by node v. (1pt)

(3) Tell the link state reports sent by node w. (1pt)

(4) Tell the link state reports sent by node x. (1pt)

(5) Tell the link state reports sent by node y. (1pt)

(6) Tell the link state reports sent by node z. (1pt)

Sample Solution:

(1) (v,4)(x,8)

(2) (u,4)(w,5)(x,3)(y,7)

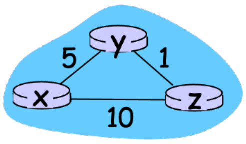
(3) (v,5)(y,6)(z,1)

(4) (u,8)(v,3)(y,9)

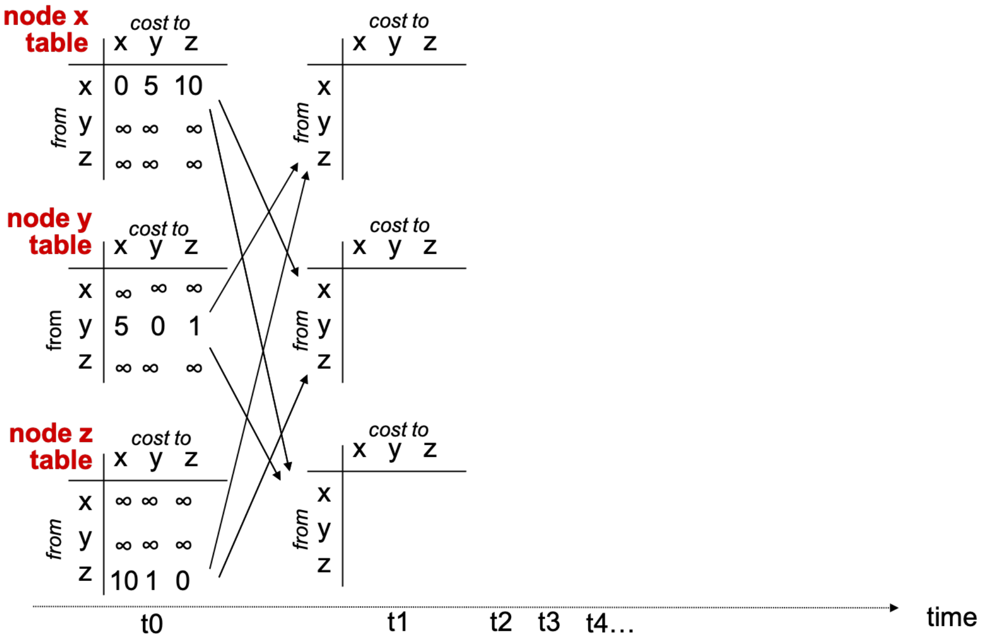
(5) (v,7)(w,6)(x,9)(z,2)

(6) (w,1)(y,2)

10. (ch53, 4pt) Consider a simple network below. Develop the full DV tables at node x, y, and z until the DV routing algorithm converges (i.e., no more changes to propagate further).



Please follow the derivation style as shown in the lecture. Assume the 3 nodes synchronously receive, compute, and send the DVs if there's any change. The initialization step at t0 is illustrated below. The first iteration happens at t1, the 2nd iteration at t2, and so on so forth.



node x table

from \ cost to	x	y	z
x	0	5	10
y	∞	∞	∞
z	∞	∞	∞

node y table

from \ cost to	x	y	z
x	∞	∞	∞
y	5	0	1
z	∞	∞	∞

node z table

from \ cost to	x	y	z
x	∞	∞	∞
y	∞	∞	∞
z	10	1	0

(1) Complete development of all nodes' DV tables until the DV routing algorithm converges. Show the Bellman-Ford computation to earn credits. (3pt)

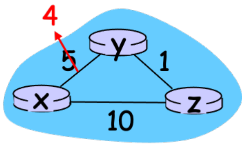
(2) Tell to which iteration (the n in the final tn) the DV tables converge. To earn credits for (2), one must complete (1) correctly. (1pt)

Sample Solution:

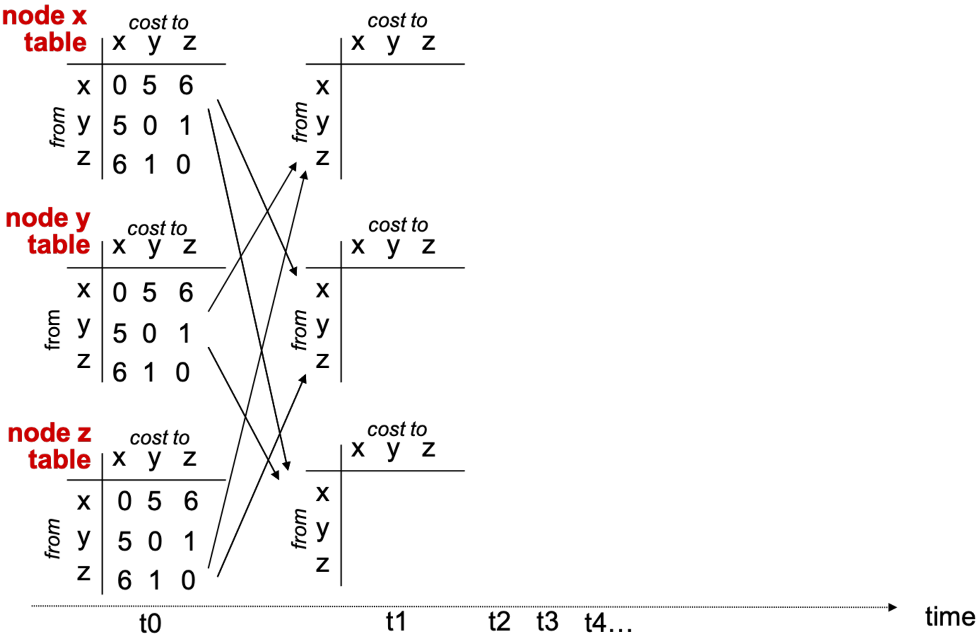
(1) will leave this for you to solve

(2) t2

11. (ch52, 4pt) Good news travel fast. Let's consider the following change to the network. Develop the full DV tables at node x, y, and z until the DV routing algorithm converges (i.e., no more changes to propagate further).



Please follow the derivation style as shown in the lecture. Assume also the 3 nodes synchronously receive, compute, and send the DVs if there's any change. The tables before the change are those shown at t0. The first iteration at t1 is triggered by the link cost change, the 2nd iteration at t2 is triggered by receiving the subsequent DV reports, and so on so forth.



node x table

from \ cost to	x	y	z
x	0	5	6
y	5	0	1
z	6	1	0

node y table

from \ cost to	x	y	z
x	0	5	6
y	5	0	1
z	6	1	0

node z table

from \ cost to	x	y	z
x	0	5	6
y	5	0	1
z	6	1	0

(1) Complete development of all nodes' DV tables until the DV routing algorithm converges. Show the Bellman-Ford computation to earn credits. (3pt)

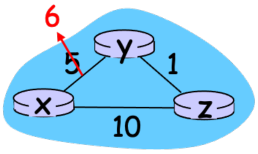
(2) Tell to which iteration (the n in the final tn) the DV tables converge. To earn credits for (2), one must complete (1) correctly. (1pt)

Sample Solution:

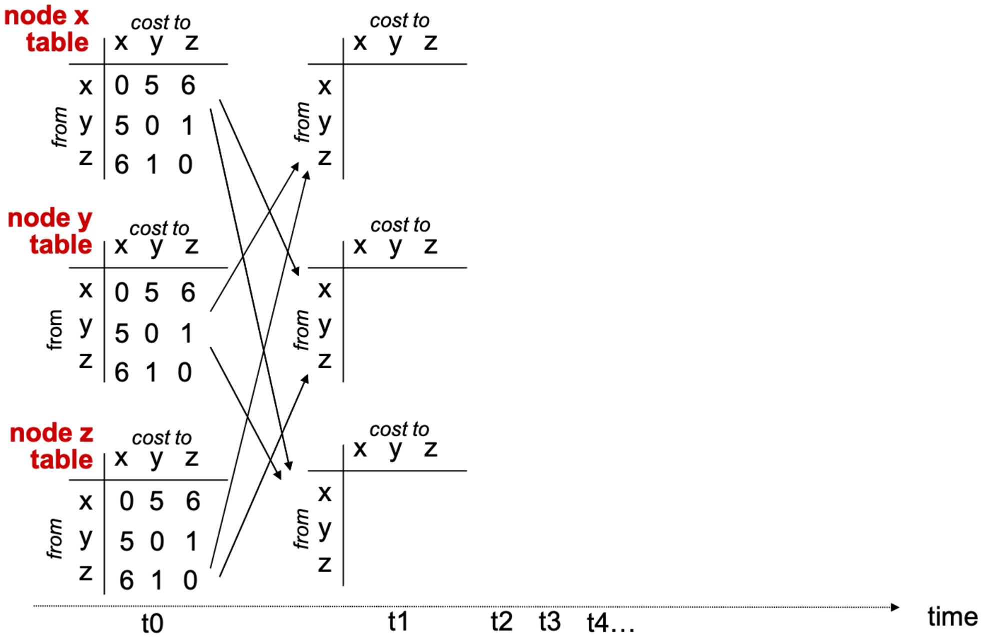
(1) will leave this for you to solve

(2) t3

12. (ch52, 4pt) Bad news travel slow. Let's consider the following change to the network. Develop the full DV tables at node x, y, and z until the DV routing algorithm converges (i.e., no more changes to propagate further).



Again, use the derivation style shown in the lecture. Assume the 3 nodes synchronously receive, compute, and send the DVs if there's any change. The tables before the change are those shown at t0. The first iteration at t1 is triggered by the link cost change, the 2nd iteration at t2 is triggered by the subsequent DV changes, and so on so forth.



node x table

from \ cost to	x	y	z
x	0	5	6
y	5	0	1
z	6	1	0

node y table

from \ cost to	x	y	z
x	0	5	6
y	5	0	1
z	6	1	0

node z table

from \ cost to	x	y	z
x	0	5	6
y	5	0	1
z	6	1	0

(1) Complete development of all nodes' DV tables until the DV routing algorithm converges. Show the Bellman-Ford computation to earn credits. (2pt)

(2) Tell to which iteration (the n in the final tn) the DV tables converge. To earn credits for (2), one must complete (1) correctly. (1pt)

(3) Compare the result to Problem 11. Does the case of bad news in Problem 12 travel slower than the case of good news in Problem 11? To earn credits for (3), one must complete Problem 11 and earlier parts of Problem 12 correctly. (1pt)

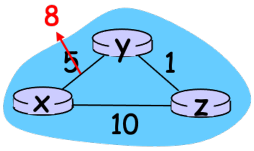
Sample Solution:

(1) will leave this for you to solve

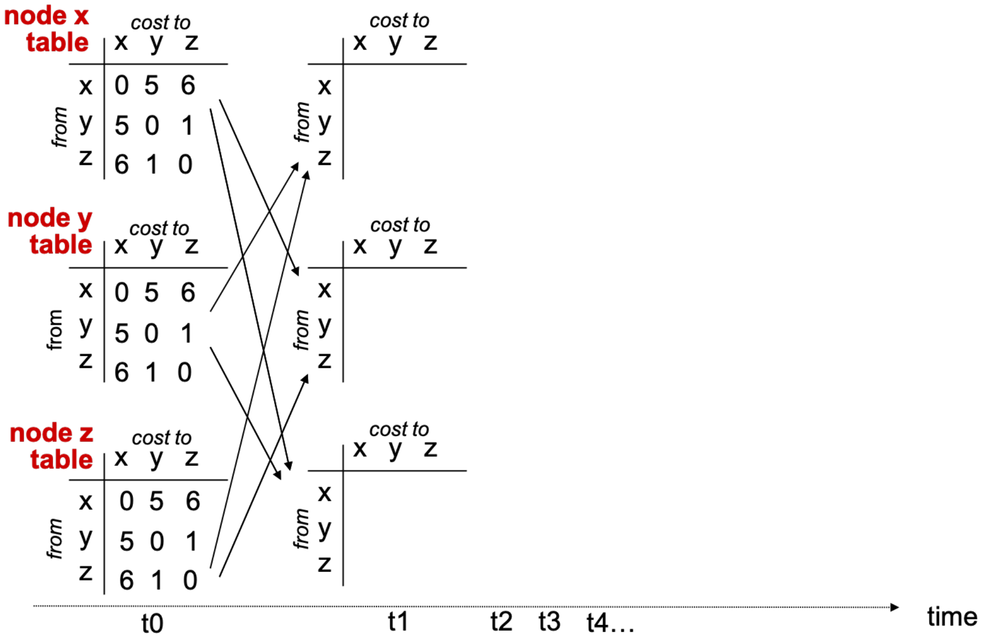
(2) t3

(3) no

13. (ch52, 6pt) Maybe it's more worse news travel slower. Let's consider the following change to the network. Develop the full DV tables at node x, y, and z until the DV routing algorithm converges (i.e., no more changes to propagate further).



Use the same derivation style shown in the lecture. Assume the 3 nodes synchronously receive, compute, and send the DVs if there's any change. The tables before the change are those shown at t0. The first iteration at t1 is triggered by the link cost change, the 2nd iteration at t2 is triggered by the subsequent DV changes, and so on so forth.



node x table

from \ cost to	x	y	z
x	0	5	6
y	5	0	1
z	6	1	0

node y table

from \ cost to	x	y	z
x	0	5	6
y	5	0	1
z	6	1	0

node z table

from \ cost to	x	y	z
x	0	5	6
y	5	0	1
z	6	1	0

(1) Complete development of all nodes' DV tables until the DV routing algorithm converges. Show the Bellman-Ford computation to earn credits. (3pt)

(2) Tell to which iteration (the n in the final tn) the DV tables converge. To earn credits for (2), one must complete (1) correctly. (1pt)

(3) Compare the result to Problem 12. Does the case of bad news in Problem 13 travel slower than the case of bad news in Problem 12? To earn credits for (3), one must

complete Problem 12 and earlier parts of Problem 13 correctly. (1pt)

(4) Project the iteration to converge when the link cost is changed to 7 and 9? To earn credits for (4), one must complete Problem 12 and earlier parts of Problem 13 correctly. (1pt)

Sample Solution:

(1) will leave this for you to solve

(2) t5

(3) yes

(4) t3 for link cost change to 7, t5 for link cost change to 9.

14. (ch54, 4pt) Recall the term – hot potato routing.

(1) What does hot potato routing do? (2pt)

(2) Packets might travel a longer inter-AS path with hot potato routing. Why does the Internet implement hot potato routing anyway? (2pt)

Sample Solution:

(1) In ch54, hot potato routing means passing of data (via intra-AS routing) to the nearest BGP gateway router regardless of the inter-AS path length.

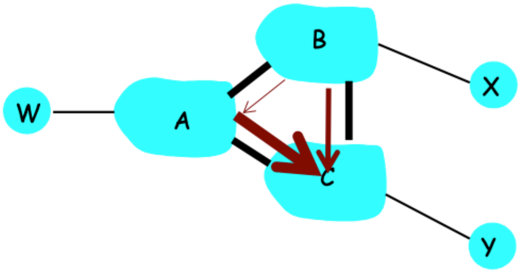
(2) Multiple possibilities. For example,

- A longer inter-AS path might not be longer delay route anyway. One might as well get the data out to the next AS as soon as possible.

- Alternatively, shorter intra-AS path on the other hand is usually shorter delay route. And as the packets arrive at the next AS, that AS will forward it to the nearest BGP gateway router, which is likely the shortest delay route possible to the subsequent AS.

- One guarantees the an optimal route locally. Alternatively, traffic in parts of an AS can be sent to different BGP gateway. A way to balance traffic load.

15. (ch54, 6pt) Below is a network we've seen in the quiz. A, B, and C are ISPs and w, x, and y are customer networks. C is the dominant ISP having the largest customer base. A is the 2nd largest and B is a late comer. The $$ flows are indicated by the arrows in brown color. The thicker the arrow, the higher the $$ volume.



Same as in Quiz#50, y is constantly sending data to x. Instead of negotiating with other ISPs a better price to pass traffic through each other, let’s try configuring the BGP path export policy to orient the traffic such that the cost is lower or the profit is higher.

(1) Suppose you are hired by B. How would you configure the export policy of AS path “Bx” to A vs. C? And why does this lower the cost for B? (2pt)

(2) Suppose you are hired by A. How would you configure the export policy of AS path “ABx” to C? And why does this lower the cost or raise the profit for A? (2pt)

(3) Suppose you are hired by C. How would you configure the export policy of AS path “CBx” vs. “CABx” to y? And why does this raise the profit for C? (2pt)

Sample Solution:

Solutions will be accepted when your arguments make sense.

16. (PA, 12pt) Please go on the PA workstation. At your home directory, create the exam3 subdirectory if you haven't done so. You will work under the exam3 directory for this problem set. You will be implementing a Web server with varying functionalities, so find a port number that's not in use and test your server on the port only.

    (1) Develop exam3-p16-1.go such that it works as a Web server – (1) accepting HTTP requests for files and (2) returning the file requested, assuming the file is always there. (2pt)

    (2) Develop exam3-p16-2.go such that it works as a Web server – (1) accepting HTTP requests for files, (2) returning the file requested if the file is found or returning "Nope, there is no such file here." if the file is not found. (3pt)

    (3) Move the exam3.key and exam3.cer file in your home directory to exam3 directory. (1pt)

    (4) Develop exam3-p16-3.go such that it extends exam3-p16-2.go to allow HTTPS requests. Use the exam3.key and exam3.cer for the secure Web server. (3pt)

    (5) Develop exam3-p16-4.go such that it extends exam3-p16-3.go and the Web server returns "Hello to you, too!" when the client requests with this URL – https://<server IP>:<port#>/hello. (3pt)

Sample Solution:

Whatever that works.

17. (PA, 13pt) Please go on the PA workstation and work under the exam3 directory for this problem set. Move the exam3.key and exam3.cer file in your home directory to exam3 directory if you have not yet done so.
    1) Develop exam3-p17-1.go such that it connects and then closes the connection to the secure servers running on port 11991-11993 sequentially. (2pt)
    2) Develop exam3-p17-2.go such that it connects, sends “delay\n”, and then closes the connection to the secure servers running on port 11991-11993 sequentially. (2pt)
    3) Develop exam3-p17-3.go such that it connects, sends “delay\n”, reads a line of message, prints the message, and then closes the connection to the secure servers running on port 11991-11993 sequentially. (2pt)
    4) Develop exam3-p17-4.go such that it extends exam3-p17-3.go, finds the secure server that gives the lowest delay and prints the port number of the server. (2pt)
    5) Develop exam3-p17-5.go such that it connects only to the secure server with the lowest delay (from running exam3-p17-4.go), sends “bandwidth\n”, reads a line of message, prints the message, and then closes the connection to the lowest delay server. (1pt)
    6) Develop exam3-p17-6.go such that it connects only to the secure server with the lowest delay, sends “resolutions\n”, reads multiple lines until a line that reads exactly “.\n”, prints all the lines, and then closes the connection to the server. (2pt)
    7) Develop exam3-p17-7.go such that it connects only to the secure server with the lowest delay, requests a video (e.g., “1040p\n”) whose bandwidth requirement is just below the bandwidth to the server (known by running your exam3-p17-5.go). (2pt)

Sample Solution:

Whatever that works.