CNT 4007C Computer Networks Fundamentals
Instructor: Prof. A. Helmy

Homework 4: Network Layer

Assigned: Nov. 26th, 2015. Due Date: Dec 7th, 2015 (to the TA)

Q1. What are the 2 most important network-layer functions in a datagram network, and what is the difference between them?

A1. Datagram-based network layer: forwarding; routing.
Forwarding is about moving a packet from a router's input link to the appropriate output link. Routing is about determining the end-to-routes between sources and destinations. Usually routing runs continuously to establish the routing tables before the packets are forwarded and in anticipation of packets.

Q2. How many columns does a bare-bones forwarding table in a MPLS-capable network have? What is the meaning of the values in each of these columns?
How many columns does a bare-bones forwarding table in a datagram network have? What is the meaning of the values in each of these columns?

A2. For a MPLS forwarding table, 4 columns, the columns are : Incoming Interface, Incoming flow label, Outgoing Interface, Outgoing flow label. For a datagram forwarding table, 2 columns, the columns are: Destination Address, Outgoing Interface. [Extra: the forwarding table can also include the address of the next hop. For multicast, a list of ourgoing interfaces are included, in addition to the incoming interface and the source address.]

Q3. An application generates chunks of 40 bytes of data every 20msec, and each chunk gets encapsulated in a TCP segment and then an IP datagram.
What percentage of each datagram will be overhead, and what percentage will be application data?

A3. Each TCP header is 20 bytes, and each IP header is 20 bytes, totaling 40 bytes of header. If the payload (i.e., application data) is 40 bytes, then the overhead is 50%.

Q4. What is the triangle routing problem in mobile IP? Suggest an improvement to it.

The triangle routing problem in mobile IP refers to the forwarding path of packets from a sending/correspondent node to the home agent, then to the mobile node (at the foreign network), instead of the direct path (from the correspondent node to the mobile node).
One way to improve it is through route optimization where the mobile node sends its new address to the correspondent host so that further packets are sent directly to the mobile node without going to the home agent. [or the home agent sends this new address to the correspondent host]

Q5. Detail two examples in which tunneling helps in mobile IP and IPv6 deployment.

A5. Tunneling (or packet encapsulation, also called IP-in-IP) can be used as follows:

- For Mobile IP, the home agent intercepts the packets destined to the mobile node, and tunnels them over to the foreign network (or foreign agent). For the encapsulated packet, the source address is the home agent and the destination address is the foreign agent (or the mobile node's new address). Then they are decapsulated when they arrive at the foreign network and are delivered to the mobile node. This serves as an effective forwarding scheme for mobile nodes
- For IPv6 deployment, tunneling can be used to forward IPv6 packets between IPv6 islands (in what is called the 6Bone [referring to the IPv6 backbone) and cross the 'sea' IPv4 routers. Since IPv4 routers cannot process IPv6 packets, the tunnel would encapsulate the IPv6 packets in IPv4 headers (that IPv4 routers can understand), for which the source and destination addresses refer to the end-points of the tunnel. [extra: This 'interoperability' scheme is essential in the 'gradual' deployment of any new technology, since the legacy systems cannot all change at the same time, and dealing with 'islands' of new technology' in 'sea' of old technology needs to be performed in the interim.]

Q6. How does IPv6 help in supporting mobility, compared to IPv4?

A6. In IPv4, only one address can be included in the header. Such address is used both for routing and for identifying the destination machine/process. The problem occurs when the machine moves, and hence needs to be assigned a new IP address that belongs to the new subnet. When the address changes, this also affects the identity of the machine, and most applications (such as file transfers, http sessions, etc.) need to be restarted with the new address.
In IPv6, using the next header option, two addresses can now be used, one for routing and the other for identity (for use in applications) to maintain the consistency of the connections.

Q7. Why is MPLS sometimes called layer 2.5?

A7. MPLS (multi-protocol layer switching) uses its own MPLS header (inserted between the MAC layer-2 header, and the IP layer-3 header), with its own MPLS label IDs. So it sits in between the 2 layers. Hence, the name layer 2.5.

Q8. Can label numbers be reused in MPLS? Why?

A8. Yes, MPLS labels can be reused as they have only a local significance (they are used in conjunction with the input interface) on that link or LAN. The label can potentially change from one link to another and can be reused. This is unlike IP addresses that are globally unique and generally do not change from one link to another [extra: except in tunnels, or NAT boxes]

Q9. Mention two examples in which IP addresses can be reused, explaining the reasons for their reuse.

A9. In general an IP address should be globally unique, except in certain cases:
1- In home and enterprise networks, the used IP addresses are local to that LAN and are not advertised outside the network. A NAT boxes handles all external communications. This serves to provide flexibility for the enterprise or home to add or remove devices without worrying about external routing issues. This also helps the limit IP address space problem since a campus with many thousands of machines can use local addresses inside and only a handle of globally-unique IP addresses. These local IP addresses (say 10.0.0.x or 192. 168.1.x) can be reused in other enterprises and homes.

2- In anycast applications, where multiple distributed servers are holding content from a content distributor (for movies, news, or other content), the different servers can use the same IP address [i.e., it is re-used for the various servers], and their addresses can be advertised externally (even through BGP). In those particular applications, getting to 'one of the *n* servers' is sufficient, and this can provide load balancing as well as redirection to the nearest server that can provide low response delay.

[Note: for DHCP lease expiration, or re-assignment of IP addresses, this can be considered re-use but at a different time, and not simultaneous reuse. Partial credit can be given for that if the 'reuse at different times' case is clarified in the answer.]

Q10. Compare and contrast link-state and distance-vector routing algorithms.

A10. Link state algorithms (used in OSPF): Computes the least-cost path between source and destination using complete, global knowledge about the network. Distance-vector routing (used in RIP): The calculation of the least-cost path is carried out in an iterative, distributed manner. A node only knows the neighbor to which it should forward a packet in order to reach a given destination along the least-cost path, and the cost of that path from itself to the destination.

Q11. Compare and contrast the advertisements used by RIP and OSPF.

A11. With OSPF, a router periodically broadcasts routing information to all other routers in the AS, not just to its neighboring routers. This routing information sent by a router has one entry for each of the router's neighbors; the entry gives the distance from the router to the neighbor. A RIP advertisement sent by a router contains information about all the networks in the AS, although this information is only sent to its neighboring routers.

Q12. Fill in the blank: "RIP advertisements announce the number of hops to various destinations. BGP updates, on the other hand, announce the _____ to the various destinations".

A12. "sequence of ASs on the routes" or "AS Path"

Q13. Complete the following sentence in ***three*** different ways, each time using one of the following words: ‘vector’, ‘domain’, ‘gateway’

“RIP is a _____ _____ protocol, while BGP is a _____ _____ protocol.”

A13.
RIP is a distance _vector_ protocol, while BGP is a path_(or AS Path)_vector__ protocol.

RIP is an intra-domain_ protocol, while BGP is a inter-domain__protocol.

RIP is an interior_ gateway__ protocol, while BGP is a _border (or exterior) gateway_ protocol.

Q14. Suppose you purchase a wireless router and connect it to your cable modem. Also suppose that your ISP dynamically assigns your connected device (that is, your wireless router) one IP address. Also suppose that you have five PCs at home that use 802.11 to wirelessly connect to your wireless router. How are IP addresses assigned to the five PCs? Does the wireless router use NAT? Why or why not?

A14. Typically the wireless router includes a DHCP server. DHCP is used to assign IP addresses to the 5 PCs and to the router interface. Yes, the wireless router also uses NAT as it obtains only one IP address from the ISP.

Q15. Consider the network setup in the figure below. Suppose that the ISP instead assigns the router the address 126.13.89.67 and that the network address of the home network is 192.168/16.

a. Assign addresses to all interfaces in the home network.

b. Suppose each host has two ongoing TCP connections, all to port 80 at host 128.119.40.86. Provide the six corresponding entries in the NAT translation table.

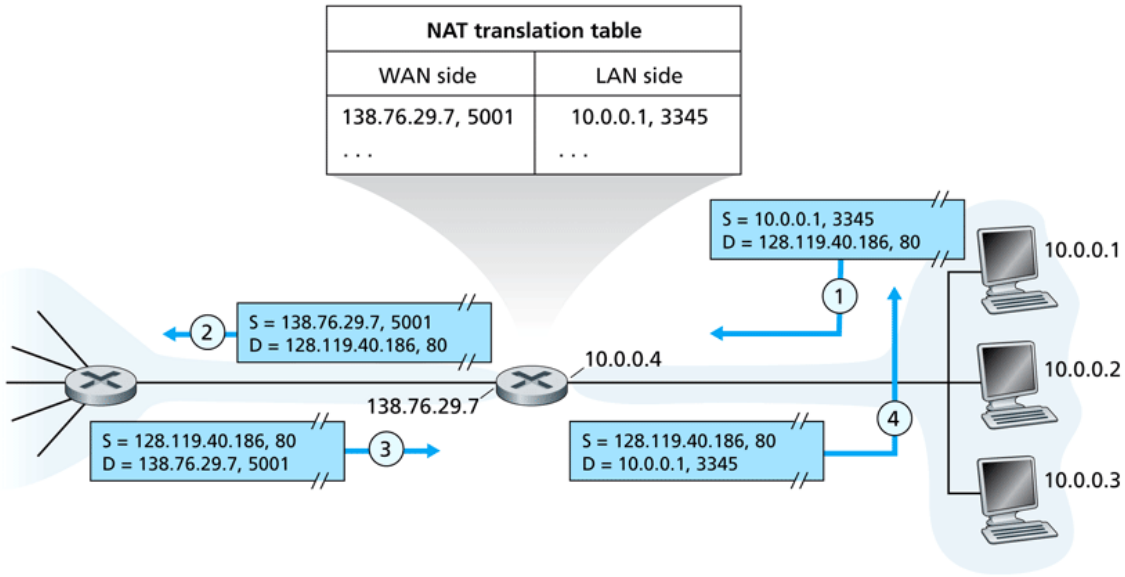


Figure 4.22 ♦ Network address translation

A15.

a) Home addresses: 192.168.0.1, 192.168.0.2, 192.168.0.3 with the router interface being 192.168.0.4

b)

NAT Translation Table

WAN Side	LAN Side
126.13.89.67, 4000	192.168.0.1, 3345
126.13.89.67, 4001	192.168.0.1, 3346
126.13.89.67, 4002	192.168.0.2, 3445
126.13.89.67, 4003	192.168.0.2, 3446
126.13.89.67, 4004	192.168.0.3, 3545
126.13.89.67, 4005	192.168.0.3, 3546

Q16. What is the '*rendezvous problem*' in multicast? How can it be solved? (mention three main approaches/algorithms to the solution along with the protocols that use them)

A16. senders do not know about receivers and receivers do not know about senders (mainly for scalability purposes).
The main approaches to solve the rendezvous problem include:

1. broadcast and prune of multicast packets (truncated reverse path broacast, DVMRP, PIM-DM)
2. broadcast of membership information (MOSPF=OSPF + membership DB, and the use of Dijsktra's algorithm for routing)
3. use of a core or rendezvous point that the senders and receivers' first-hop-routers know about (PIM-SM)

Q17. How does SDN extend the notion of matching to define firewalls and NAT boxes, while traditional longest-matching is limited?

A17. SDN allows the 'controller' to define the 'matching' rules (including exact or longest matches, and including matching on potentially multiple fields in the headers [MAC, IP, or transport headers]. It also allows various 'actions' to forward, block or change the header in ways that go beyond traditional longest-prefix matching.

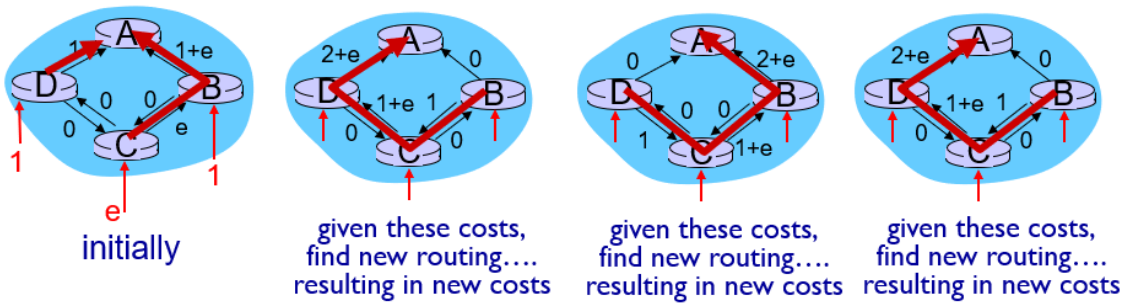
Examples of using SDN to implement NAT or firewalls are given in Ch4 – 77. For example, a firewall that blocks ssh (port 22) can be easily defined as 'match port = 22, action: drop/block/deny'. A NAT example would match on IP and port number, with action to change the header in the way defined for NAT functionality.

Q18. Do you see a layer violation in BGP peering? Explain.

A18. BGP peering between different border router (operating at the network layer, layer-3), uses TCP which is a transport layer (layer-4) protocol that should be operating end-to-end. Yes, there is a layer violation since layer-(n+1) usually uses layer-n protocols and services and not the other way around.

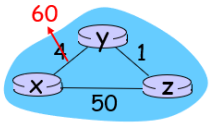
Q19. Provide example of route oscillation in link state routing (e.g., Dijkstra's algorithm).

A19. From the lecture slides Ch5 – 12, when using the link traffic (or congestion) as a routing metric, oscillations can occur as follows: [check slides for animations]



Q20. Explain the 'count-to-infinity' problem and why it happens in distance-vector routing algorithms (like Bellman-Ford in OSPF), then suggest an improvement. Mention a limitation of your suggestion.

A20. The problem occurs when using a metric that changes dynamically, such as traffic (or congestion) which in turn may reflect in changes in the routing, which triggers change of traffic, that triggers route changes, and so on and so forth … this is explained in the lecture (Ch5 - 22) with the following example:



This example needs 44 iterations of routing updates to converge, as 'y' thinks it can go through 'z' to 'x'. [see the lecture for further details]

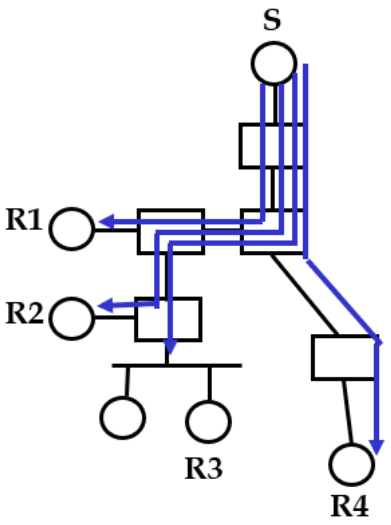
One way to improve this situation is by using 'poisoned reverse' messages in the routing exchange, to tell upstream nodes (y in this case) not to use their downstream nodes (z in this case) to reach a destination (x in this case). So z's routing distance vector update to 'y' will contain a metric of 'infinity' to reach 'x'. This eliminates this routing loops that contain y, z and x as in the above configuration. It has a limitation of not working as well when the routing loop is larger and contains more than 2 nodes as in the above example.

Q21. What is the advantage of using multicast over unicast? Give example and provide a quantitative analysis (i.e., numbers) to solidify your argument (you can use a drawing of the network topology).

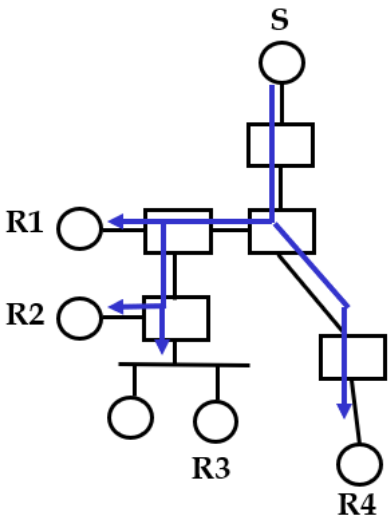
A21. IP multicast is used for efficient point-to-multipoint (and multi-point to multi-point) communication, to provide a more scalable solution (with lower overhead) than multiple unicasts. For multiple unicasts the same packet can traverse the same links multiple times since the different unicast streams are routed independently. Multicast establishes a distribution tree that caries every packet only once, with its branches reaching all the receivers (group members) at the leaves.

Examples include 'sender to multiple receivers' applications, such as IPTV, replication in CDNs, data centers and clusters, broadcast lectures, or multimedia teleconferencing, collaborative on-line learning, telemedicine, etc.

The example below shows a reduction from 18 flows over links to 9 flows, to deliver packets from the source to the same set of receivers. A saving of 50% for this example.



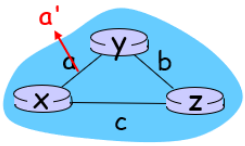
Multiple unicasts
(18 link traversals)



Multicast
(9 link traversals)

Q22. [Extra points] Consider the network topology in the figure below with nodes *x*, *y*, *z*. Assume that distance-vector routing is used with Bellman-Ford algorithm. The link cost between *x* and *y* changes from *a* to *a'* , where *a'* >> *a*, and *a'* > *c*. Focus on nodes *y* and *z*, their routing table updates, and messages exchanged between them.

I. Derive an equation for the number of iterations needed for the routing to converge as a function of the link costs *a*, *a'*, *b* and *c*.



II. Suggest an improvement that would speed up the convergence and prevent looping in the above case. Explain your solution and how it works.

A22. I. Consider just before the increase in cost from "a" to "a' " for the link between y and x, at time t_0 the cost of getting from y to x was "a" and from z to x was "a + b". That is t_0: Dz(x) = b + a.

Right after the increase in cost to "a' ", at t_1, y discovers the increase and compares it to z's cost to get to x (a+b), plus y's cost to get to z (b), at t_1: Dy(x) = min(a+2b, a')=a+2b

The next iteration, at t_2 y advertises its distance vector to z, informing it with the new cost to get to x, so z updates its cost to a+2b plus 'b' (its cost to get to y), so long as it is lower than 'c'

at t_2: Dz(x)=min(a+3b, c) = a+3b, and sends the update to y

at t_3: Dy(x) = a + 4b

and so on… at iteration n, time t_n: Dy(x) = a + (n+1) b

This occurs until the cost equals/exceeds 'c' at which point z realizes that its direct path to x costs less than the cost through y. This can be obtained when c = a + (n+1) b

Or n = (c-a)/b – 1 , to converge on the new path

Example if a=4, b=1, c=50, n=(50-4)/1 – 1 = 45 (i.e., on the 45^{th} iteration no more updates will be done, so 44 iterations of updates are needed before stability).

Q23. [Extra points] Mention the two main reasons why broadcast-and-prune multicast does not scale**.**

A23. First, the broadcast of the packets (in case of DVMRP), or membership (in case of MOSPF), over all the links in the network [initially and periodically] does not scale with the increase in number of links in the network.
Second, because the broadcasts are sent to everyone in the network (members and non-members in the multicast group), those who do not want the packets need to maintain 'prune state' off-tree [i.e., on parts of the tree that do not lead to members]. For sparse groups, the number of routers in the network not leading to members (i.e., off-tree) is high, thus incurring overhead that does not scale well.

Q24. [Extra points] Mention two reasons why an optimal *RP* placement is not used (or needed) in PIM-SM**.**

A24. First, the optimal placement of a node (the rendezvous point, *RP*, in this case) in a network with respect to a group of members has been proven to be NP-complete problem (the Steiner tree problem). Add to that the dynamics of the membership – any node can join or leave at any time –, which can lead to changes of the placement (potentially disrupting packet forwarding and delivery).
Second, the fact that the RP is used mainly for rendezvous (as a meeting point) and then PIM-SM switches to the shortest path tree between the sender and the receivers. Hence, the protocol side-steps the NP-complete problem altogether. [extra: the criteria for choosing the RP is connectivity, reachability and availability].