Examination
IK2218 Protocols and Principles of the Internet
EP2120 Internetworking

Date: 27 October 2015 at 14:00–18:00

a) ***No help material is allowed - You are not allowed to use dictionaries, books, or calculators!***
b) *You may answer questions in English or in Swedish.*
c) *Please answer each question on a separate page (not sheet).*
d) *Please write concise answers!*
e) *Put a mark in the table on the cover page for each question you have addressed.*
f) *The grading of the exam will be completed no later than 17 November 2015.*
g) *After grading, exams will be available for inspection online.*
h) *Deadline for written requests for grading review is 27 November 2015.*
i) *Course responsible IK2218 is Markus Hidell, phone 070-249 0252.*
j) *Course responsible EP2120 is György Dán, phone 08-790 4253.*

Important note!

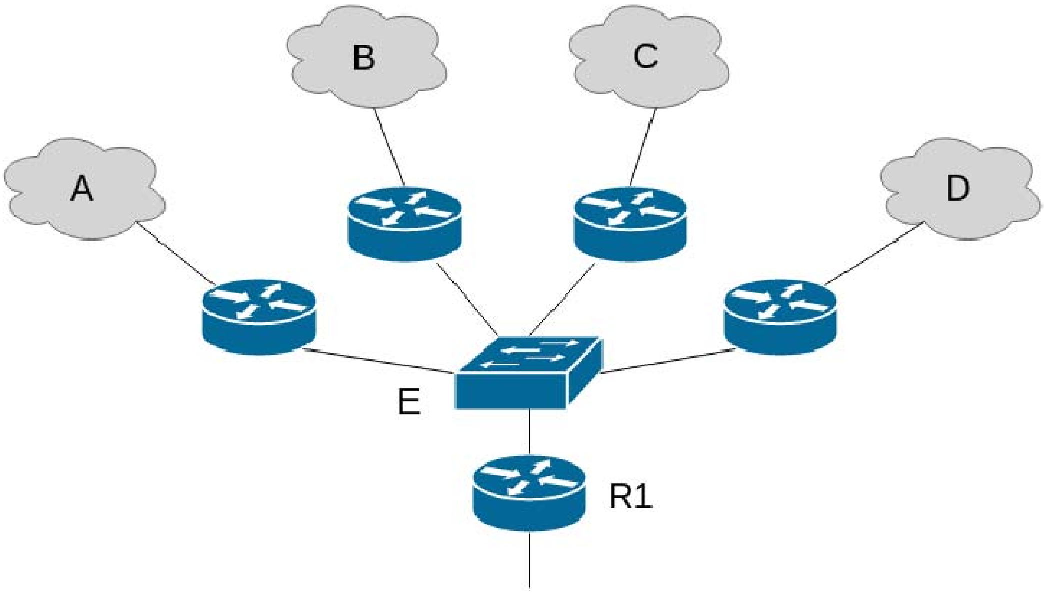
Your grade is F in any of these two cases:
- if you do not reach at least 10 (ten) points out of 20 for problems 1-4 or
- if you reach less than 30 points in total.

We advise you to start with problems 1-4.

Part I (Problems 1-4)

1. *IP and addressing (5p)*

Consider the network below, a routed network in an organization's enterprise network. The organization built a core network connected to a central router R1, and connected their edge/access routers with (long-haul) switched Ethernet (network E). Router R1 connects the enterprise network to the Internet through the Internet Service Provider. The access routers are connected to a set of local offices (networks A to D). All networks use Ethernet on the link layer.



Your task is to make an address allocation in the network by assigning a sub-block to each network A–D in the following way:

1. You need to use the address block 172.30.0.0/20 for address allocation.
2. The networks A–D require 1000 hosts each. Create a minimal block for each local office A through D. Start with the lowest address for network A.
3. There are no unnumbered point-to-point links: all Ethernet networks have a corresponding IP sub-network and all nodes (routers and hosts) have an IP address on each of their network interfaces. All nodes need to be reachable from any other host.

Based on your address allocation, provide the answers to the following:

a) What is the longest prefix length that you could consider for the networks A–D? What is the corresponding netmask? (1p)

b) For each of the networks A–D give the network address in CIDR notation and the directed broadcast address! What could be a possible IP address for router R1's interface on network E? Motivate your answer. (1p)

c) Are the addresses allocated to networks A–D routable on the public Internet? Do you need any particular functionality in R1 to be able to communicate from a host in network A with the rest of the Internet? If yes, what? Motivate your answer. (1p)

d) Consider that you have an IPv6 network, in which all links have an MTU of 2 MB. You would like to make use of the large MTU offered by the link layer technology. What support is there in IPv6 that you can rely on? (1 p)

e) What is the difference between link-local and unique-local addresses in IPv6? How do they relate to private addresses used in IPv4? (1p)

SOLUTION:

a) The prefix length should be /22 or shorter, the corresponding netmask is 255.255.252.0.

b) *With the prefix length of 22 bits, each network can accommodate up to 1022 hosts.*

Network	Network address	Broadcast address
A	172.30.0.0/22	172.30.3.255
B	172.30.4.0/22	172.30.7.255
C	172.30.8.0/22	172.30.11.255
D	172.30.12.0/22	172.30.15.255

Router R1IP address could be any IP address in the block allocated to network E, except the network address and the directed broadcast address. Note that the /20 block is insufficient for allocating a block to network E, as the four /22 blocks cover the entire /20 block. While one could come up with some ugly workaround, it is best to allocate a block of addresses outside of the /20 block to network E.

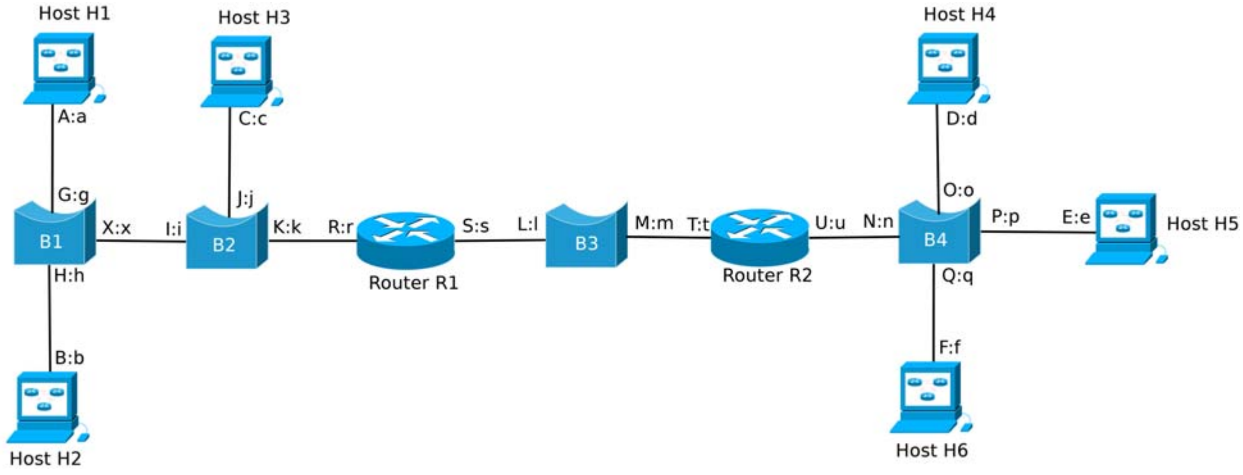
c) No, those are private IP addresses. Router R1 would need to have NAT functionality.

d) The IPv6 Hop-by-hop extension header can contain the Jumbo payload option, which allows one to send datagrams of length up to 4GB.

e) *IPv6 link-local addresses are valid only for addressing on a single link, while a unique local address can be used within a site very much like private addresses in IPv4. Every IPv6 enabled host has a link-local address, but they do not have to have a unique local address.*

2. *Delivery and address resolution (5p)*

Consider an IPv4 network consisting of 6 hosts, 4 bridges and 2 routers shown in the figure. Hosts H1 to H6 have one interface each. B1 to B4 are learning bridges. R1 and R2 are routers with appropriate forwarding tables. All ARP caches and the bridges' learning tables are empty.



a) Identify the subnets of the network. Observe that bridges B1, B2, B3, B4 have MAC and

IP addresses. What purpose do these addresses serve for? (1p)

b) A process on Host H1 sends 200 bytes via UDP to a process on Host H6. Using the notation in the figure, show the contents of the learning tables and of the ARP caches after the datagram has been delivered. Assume that the process on Host H1 knows the IP address of Host H6, and that ARP snooping is used. (1p)

c) A process on Host H5 sends a message with 100 bytes via UDP to Host H2. Using the notation in the figure, show the new contents of the ARP caches and of the learning tables after the datagram has been delivered. Assume that Host H5 knows the IP address of Host H2 and that ARP snooping is used. (1p)

d) How different would the ARP caches and learning tables be in b) and in c) if ARP snooping was not used? (1 p)

e) What layer of the TCP/IP protocol stack does ARP belong to? Motivate your answer. (1p)

a) Subnet 1: A, B, C, R. Subnet 2: S, T. Subnet 3: D, E, F, U. Bridges B1 to B4 do not need MAC addresses and IP addresses, but if a bridge has an IP and a MAC address then it can be remotely managed.

b) Contents of the ARP caches are as follows.

H1: R-r
H2: A-a
H3: A-a
H4: U-u
H5: U-u
H6: U-u
B1: a-north, r-east
B2: a-west, r-east
B3: s-west, t-east
B4: u-west, f-south
R1: A-a, T-t
R2: S-s, F-f

c) New entries in the ARP tables are as follows

H2: R-r
H3: R-r
B1: b-south
B2: b-west
B4: e-east
R1: B-b
R2: E-e

d) Question b): the ARP caches at H2, H3, H4 and H5 would be empty (0.5p).
Question c): the ARP cache of H3 would not contain R-r as a new entry.

e) An ARP packet is encapsulated within a link-layer frame and thus ARP relies on the services of the link layer (point-to-point delivery). At the same time, an ARP packet has fields containing IP addresses and thus one may argue that it is a network-layer protocol. Note however, that ARP packets are not forwarded by routers, they are local to a physical network, and IPv4 uses the services of ARP. Therefore the protocol is typically classified in Layer 2, sometimes one refers to it as being in Layer 2.5.

3. *IP forwarding (5p)*

a) Suppose a router receives and forwards a packet. Later, due to a routing loop, the router receives the same packet again. How does the router treat the packet the second time it receives the packet? What will happen to the packet as it keeps on being forwarded in the routing loop? (1p)

A router has the IPv4 routing table shown below. Determine the next-hop address and the outgoing interface for the packets arriving to the router with destination addresses as given in points (b) – (e).

Destination	*Next-hop*	*Flag*	*Interface*
10.56.48.0/20	–	U	m0
172.29.73.0/26	–	U	m1
192.168.0.0/16	–	U	m2
10.65.11.1/32	172.29.73.1	UGH	m1
172.28.1.0/24	10.56.62.6	UG	m0
176.22.0.0/18	192.168.0.2	UG	m2
188.122.0.1/32	192.168.0.1	UGH	m2
0.0.0.0/0	10.56.58.1	UG	m0

b) 10.65.11.2 (1p)
c) 10.56.62.255 (1p)
d) 176.22.51.39 (1p)
e) 172.28.1.1 (1p)

Solution:
a) The router treats the packet the same way as any other packet, forwarding it according to its forwarding table. For each hop the TTL counter of the packet is decremented until it reaches 0, at which point the packet is dropped.

b) 10.56.58.1 on m0 (default route)
c) direct delivery on m0
d) 192.168.0.2 on m2
e) 10.56.62.6 on m0

4. *TCP (5p)*

a) You are designing connection teardown for a reliable connection oriented transport layer protocol, in which the sender retransmits every segment for which it has not received an acknowledgement. The current design is as follows. Given two processes A and B, if A wants to terminate the connection, it sends a segment with the FIN bit set, and in response to this message B sends a segment with the ACK bit set. If now B wants to terminate the connection, it sends a segment with the FIN bit set, which A acknowledges with a segment with the ACK bit set, and A then considers the connection terminated (and can free up all resources related to the connection). What can go wrong with this design? How does TCP connection teardown differ from the above? (2p)

Consider two hosts, A and B, connected by a network running IPv6. The capacity of all links is 10Mbps and the round trip time is 200ms. The path MTU is known to be 1560 bytes. A process P_A on host A would like to transmit 30000 bytes to a process P_B on host B using TCP. TCP on the receiving host has a receiver window size limit of 9000 bytes, which it advertises during connection establishment. The sender uses a value of 65535 for *sshthresh* for congestion control. Delayed acknowledgements (two full sized segments) are used with a maximum delay of 200ms. The receiver can process the data as fast as they arrive.

b) What is bandwidth delay product of the channel? How big should the receiver window size be in order to be able to fully utilize the channel (not considering congestion control)? (1 p)

c) Consider that process P_B reads all data from the receive buffer as soon as they arrive. The active open is performed by A. The initial congestion window size is 3xMSS. How much time does it take to transmit the data from A to B including the connection establishment, until the last ACK is received by A? You can ignore the transmission times of the packets, but you should consider the impact of congestion and flow control. If a delayed ACK is to be sent at a time instant when a new segment arrives, the delayed ACK is sent first. Support your solution with a drawing of the segments sent, including the CWND, time sent, ACKed data, etc. (2p)

SOLUTION:

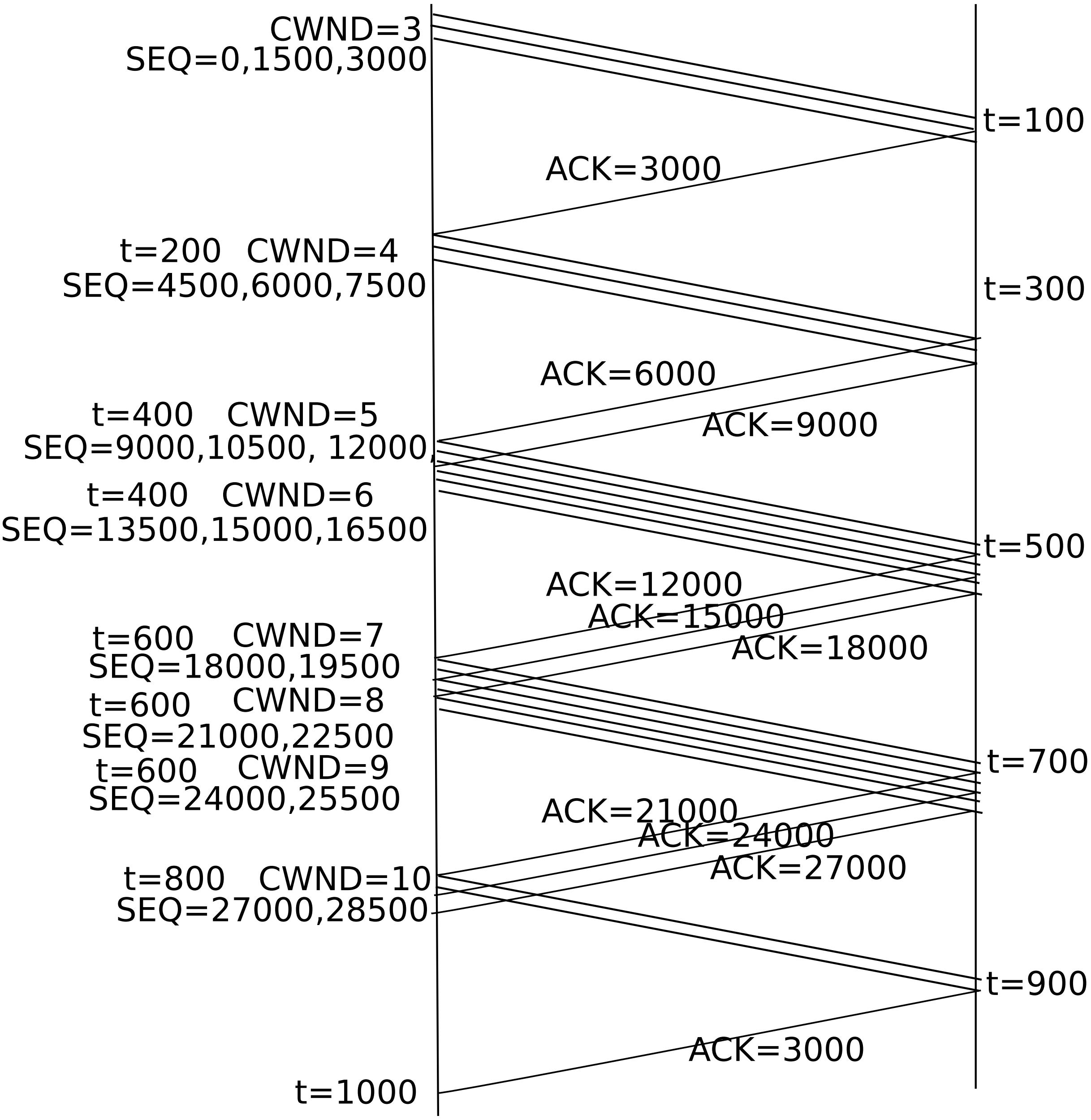
a) Consider the following scenario:

In this design A assumes that the segment containing the ACK will be delivered to B. If the segment is lost, B would retransmit the FIN segment, but A would ignore this and would never acknowledge the FIN segment.

In TCP connection teardown, A would set a timer (TIME_WAIT timer), and would keep on listening on the port for incoming segments until the timer expires.

b. The bandwidth delay product is $10*10^6*0.2s=2Mbit$. This is the receiver window size that one would need.

c. The MSS=1500bytes. The connection establishment takes 1RTT, data can be sent after that. In the figure time 0 corresponds to the first data segment sent, connection establishment is not shown. CWND is measured in terms of MSS, SEQ corresponds to the first byte of the segment, ACK is the next byte expected. In total the transmission takes 200+1000ms=1200ms. Observe that starting from t=600, CWND>RWND, and thus RWND is the limiting factor.



A slightly different solution is obtained if one assumes that the 2nd ACK (i.e., the first delayed ACK) is sent before the 4th segment arrives. That solution is of course correct if it is consistent otherwise (i.e., follows slow-start, rwnd, cwnd, etc).

Part II (Problems 5-1~~2~~1)

5. *Fragmentation and UDP (5p)*

a) In IPv4 the UDP checksum is calculated based on the pseudo-header, the UDP header and the UDP payload. The pseudo header is created using fields of the IP header. What is the purpose of including the pseudo-header in the checksum calculation? (1p)

b) What is the function of the More Fragments (MF) bit in the IPv4 datagram header? How does the destination host determine whether a fragment is missing and how does it reassemble the fragments in their proper order?(1p)

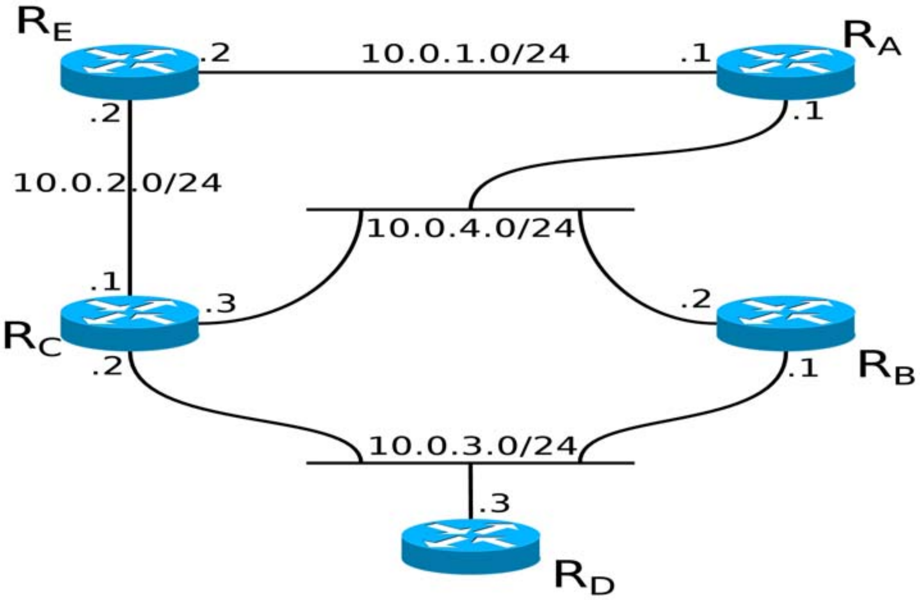
c) An application wants to transmit 2678 bytes of data via UDP from host A to host B via an IPv6 network. The UDP header is 8 bytes long. The path consists of two local area networks; the MTU of the first network is 1500 bytes, and the MTU of the second network is 1395 bytes. The path MTU is known at host A. How many IPv6 fragments arrive at host B? Give the fragment sizes, the fragmentation offset and the more fragments (MF) bit of all fragments.(3p)

a) Adding the pseudo header allows the checksum to also protect against problems such are delivery of a message to the wrong destination and delivery of a message to the wrong protocol (e.g., UDP instead of TCP).

b) The last fragment of the original datagram has the More Fragments bit set to 0, whereas all other fragments have this flag bit set to 1. In order for the destination host to determine whether a fragment is missing and also to be able to reassemble the fragments in their proper order, the destination host uses the ID, the offset field, and the MF bit (the last fragment according to the offset should have MF=0).

c) In total 2678+8=2686 bytes of payload have to be transmitted. The path MTU is 1395 bytes. The IPv6 base header is 40 bytes, the fragmentation extension header is 8 bytes long. The IPv6 payload in every non-last fragment could be 1395-48=1347 bytes, but this is not divisible by 8, so it will have to be 1344 bytes. The last fragment's length does not have to be divisible by 8, hence it can carry up to 1347 bytes of payload. The IPv6 payload, offset and MF values are (all in bytes):
    1) 1344, 0, 1
    2) 1342, 1344, 0

6. *Routing (5p)*



In the IPv4 network shown in the figure all routers R_A-R_E run RIPv2 and all link metrics are 1. The addresses of the IPv4 networks and the associated interface addresses are given in the figure. Note that the letters R_A-R_E do not denote addresses. Assume an initial state for all routers, where only the addresses of the directly connected networks are present in the routing tables. The destinations in the network are the /24 prefixes. Assume also that all RIP implementations support Equal-cost-multi-path (ECMP). All routers implement split-horizon and poison reverse. When answering the questions below, express routes as 'destination, metric, next-hop'. If the destination is a directly connected network, the route is given as 'destination, metric, -'.

a) What is the initial routing state of R_C? (1p)

b) Assume that router R_C starts by sending a RIP response to its neighbors. What is the routing state of R_D after it has received the initial distance-vector from R_C? (1p)

c) Assume that the second event that happens in the network is that router R_B sends RIP responses to its neighbors. Which RIP response messages does R_B send, and which distance-vectors do they contain? You should indicate the source and the destination address of each RIP message, on which interface it is sent out (and to where) and which distance-vectors (destination-metric tuples) are contained in each message. (1p)

d) What are the routing states of R_C and R_D after they have received the distance-vector from R_B in the previous step (c)? (1p)

e) After the routing information has spread through the network and the routing states of each router has stabilized, will we find any occurrences of ECMP? If so, where? Why are we keeping track of multiple routes to the same destination if we know that their costs are equal?

a)

Destination	Metric	Next-hop	Interface
10.0.2.0/24	1	–	n
10.0.3.0/24	1	–	s
10.0.4.0/24	1	–	e

b)

Destination	Metric	Next-hop	Interface
10.0.3.0/24	1	–	n
10.0.2.0/24	2	10.0.3.2	n
10.0.4.0/24	2	10.0.3.2	n

c)

On the west interface, R_B sends a RIP response message with source address 10.0.4.2 and destination address 224.0.0.9. The distance-vector of this message using split-horizon with poison reverse is:

Destination	Metric
10.0.3.0/24	1
10.0.2.0/24	16 # A single tuple is provided for the ECMP route
(10.0.4.0/24	16 # RIP implementations may announce this network but it is not necessary since all connected routers have this as a directly connected network: it is accepted both to have this route and to omit it)

On the south interface, R_B sends a RIP response message with source address 10.0.3.1 and destination address 224.0.0.9. The distance-vector of this message using split-horizon with poison reverse is:

Destination	Metric
10.0.4.0/24	1
10.0.2.0/24	16

(10.0.3.0/24 16 # Same comment as above)

d)

The routing state of R_C does not change:

Destination	Metric	Next-hop	Interface
10.0.2.0/24	1	–	n
10.0.3.0/24	1	–	s
10.0.4.0/24	1	–	e

The routing state of R_D is updated by adding a second route to the 10.0.4.0/24 network:

Destination	Metric	Next-hop	Interface
10.0.3.0/24	1	–	n
10.0.2.0/24	2	10.0.3.2	n
10.0.4.0/24	2	10.0.3.2	n
10.0.4.0/24	2	10.0.3.1	n

e)

Yes, for example from Router R_D to the network 10.0.4.0/24.

Multiple paths could for example be used for load balancing when forwarding many packets, however, in practice it is difficult to take advantage of. See for example RFC 2991 for a discussion about the concerns around ECMP.

7. Web and HTTP (7 p)

You are fetching a Web page from a Web server. The Web page consists of two objects, an image and an audio sample. Every HTTP response from the server contains the following header lines:

```
Connection: close
Cache-Control: max-age=120
```

From these lines we can see that the Web server does not support persistent connections, but it allows the response to be reused (cached) for the next 120 seconds. Your organization installs a Web proxy, which you have to use in order to access any Web site. Unlike the Web server, the proxy supports persistent connections. Assume that the round-trip time (two-way delay) between your computer and the proxy is T_P, and that the round-trip time between the proxy and the server is T_S. Everything else in this system is incredibly fast, so you can ignore all processing and transmission delays.

a) The proxy and your computer have recently been started, so all caches are empty. How long time does it take from that you click on the link to get the Web page, until the Web page can be presented on your computer? (2 p)

b) You are eager to see if there are any updates on the Web page (which there aren't), so impatiently you reload the page after just a few seconds, by clicking on the link again (and forcing your browser to flush any caches). How long time does it take now? (1 p)

c) You are so fascinated by this Web page that you decide to view it in three different browser windows at the same time, so you will have three TCP connections open to the proxy. How many TCP port numbers are you using then on your computer, and how many TCP port numbers are in use on the proxy? (1 p)

d) Assuming that both your computer and the proxy are using the standard socket API (Application Programming Interface), and that this is the only communication going on at the moment, how many sockets are open on your computer and the proxy? (1 p)

e) Next you turn your attention to a Web site that uses cookies. What does the proxy do with the HTTP requests and responses that are exchanged between your computer and this Web site? Does the proxy cache them or not? Explain your answer. (2 p)

Solution

a) Between client and proxy: one round-trip time to set up the TCP connection, one to get the main object, and then one for each object on the page. $4T_P$ total. Between proxy and server, there is one TCP connection per object, plus the round-trip time to get the object, so $6T_S$. Hence, it the total time is $4T_P + 6T_S$.

b) This time the proxy has everything in its cache, so it can respond directly to your request. Hence, the time it takes is $4T_P$. ($3T_P$ if the TCP connection between client and proxy is still open – both answers would be correct.)

c) Three port numbers on your computer (one for each connection), and one on the server (the well-known port number where the proxy is providing HTTP communication service, port 80 most likely). If the connection between proxy and server is also taken into consideration (which is not a requirement), there will be another port in use on the proxy.

d) Three sockets on your computer (one per connection) and four on the proxy (one per connection, and one listen socket where the proxy is accepting new connections) For the connection with the server, yet another socket is used on the proxy.

e) The proxy shouldn't cache a response that contains a cookie ("Set-Cookie:" header), neither should it cache a response to a request with a cookie ("Cookie:" header). The reason is that those responses are, most likely, customized for a specific user.

As usual, in practice there are more subtleties involved and there are header fields to control caching with respect to cookies. There are situations where it may in fact be desirable to cache cookies – for instance, a server may want a cookie to be shared by all users, or a proxy could maintain separate caches for different users. An answer along those lines must be properly explained and motivated.

8. DNS Resolution (8 p)

You wonder what the name server is for the domain `duck.soup`. You send a DNS request from your regular computer connected to KTH's network, for instance using dig:

```
dig ns duck.soup
```

As a result, you receive a response with the following information:

```
duck.soup.              3280  IN    NS      rufus.duck.soup.
```

a) Your dig command will trigger a number of DNS requests and responses to be sent. Describe each request/response transaction that takes place in the DNS system in order to answer your question. For each message, specify the sender and the receiver, and explain what the message contains. Assume that all caches in the system are empty. You do not need to specify the content of the individual fields in the DNS messages, it is sufficient that you briefly characterize their content. Make a drawing, for instance. (3 p)

b) Your solution to a) probably involves communication with several name servers. In order to send DNS messages to those name servers, their IP addresses need to be determined. Explain for each of the name servers involved how its IP address is determined during the process of answering your question. (3 p)

Next you try to send your question to a specific name server, namely `horse.feathers`:

```
dig ns duck.soup @horse.feathers
```

c) You find that the response does not contain the answer to your question. Why not? Explain! (2 p)

Solution

a)

1) Request from your computer to the local name server
2) Request from the local name server to a root server
3) Response from the root server to the local name server, with the top-level domain server for “soup.”
4) Request from the local name server to the top-level domain server for “soup.”
5) Response from the top-level domain server for “soup.”, with the authoritative name server for “duck.soup.”.
6) Response from the local name server to your computer, with the authoritative name server for “duck.soup.”.

b) The IP address for the local name server is in your computers IP configuration, which your computer received through DHCP. The IP address for the root server is in the root zone file, which has been downloaded to the local name server. The IP address for the top-level domain server for “soup.” is in the response from the root server to the local name server, as a glue record.

c) The “horse.feathers.” server does not know the answer, and most likely it does not provide recursive name resolution, so it will not find out the answer for you. It will send a response, but it won’t contain the answer. Just answering “it doesn’t know” will not give any points. (Some have answered that the queried domain is not given as an FQDN, and therefore the lookup will be relative to the current domain (kth.se perhaps) and fail. Dig will actually turn the request into an FQDN, which you would have learned from using dig in the labs. However, it is a clever observation, and considered a valid solution.)

9. Autoconfiguration and SLAAC (5 p)

Briefly explain how IPv6 *Stateless Address Auto-Configuration* (SLAAC) works, and why DHCP still may have a role even if SLAAC is used.

Solution

The client first generates a Link Local Address (LLA), which is an address that is not globally routed, and probes the subnet to ensure it is not currently in use. It then combines the global unicast prefix of the organization and the subnet prefix of the subnet as provided by gateway routers in Router Advertisements (RA) with its LLA to generate a unique global address.¨

Using SLAAC, a host will only get an IP address, but in many cases we also want information about local resources like DNS server, printer server, time server etc., which a DHCP server can provide.

10. IPsec (5 p)

Describe outbound (for outgoing packets) IPsec processing in terms of explaining the interplay between IPsec protection, security parameter index (SPI), security policy database, security association (SA), and the SA database.

Solution

First, the security policy database (SPD) is consulted to check if the outgoing packet should be protected with IPsec or not. The SPD can be seen as a filter table, and it provides a pointer to the correct security association (SA) entry in the SA database (SAD). The SA will then provide the kind of protection, algorithms, keys, etc that are needed for the IPsec protection. The SA will also provide the SPI, which is included in the IPsec header of the outgoing packet.

11. Firewalls (5 p)

Firewalls can be placed in a number of different places, providing different protection. Give at least three examples of places where deploying firewalls is motivated, and explain the motivation for placing them there.

Solution

1. On the host itself, i.e., a *personal firewall.* Common in operating systems.
   Placed here to protect the host itself, and to regulate access to and from the host. Because you may still want to be able to access, e.g., shared filesystems etc, on the LAN, this firewall is often more permissive than other firewalls.
2. Between the host and the rest of the Internet. This is the classic firewall, protecting the LAN from outside threats and especially access to services used on the LAN which are not meant to be accessed from the outside, such as printers, file servers etc. It may also serve to limit access from the LAN to the outside (e.g., corporate firewalls limiting access for the employees, national firewalls preventing access to pages belonging to political dissenters).

Between departments and LANs inside a company. Sometimes done because some departments are more sensitive (payroll), but it is also a good practice in general to compartmentalize access, since this means that even if an attacker gets into a corporate LAN, they do not have unlimited access.