Comprehension Questions of Task 3: Switch Configuration

1. On which layers of the ISO/OSI model does the ARP operate?

It operates in the second layer, also known as data link layer, as it is used to discover MAC addresses and similar. However, as it also works with internet layer addresses like IPv4 addresses to discover them, it can also be considered as operating within this third layer.

2. What does an end device do when it wants to send a packet to an IP address for which it has no entry in its ARP table?

In order for a device to send a packet to another, we need two things, both the recipient IP address and MAC address. As we have the IP address but not a MAC address in our ARP table, we send an ARP Broadcast message over the local network, requesting an answer for the already known IP address. The recipient then responds with its IP and MAC addresses, and may save our computers' information. We receive the information and we are ready to send the packet.

3. What does a switch do with a frame whose destination MAC address is unknown? How does it learn which MAC address is mapped to which port? When does this happen at the latest?

When a device sends a frame through a switch, the switch stores its MAC address and the port where it came in on its cache. Therefore, when a switch doesn't know the MAC address, it simply floods all of its ports except the one where the packet came from, expecting the recipient to answer the transmitter. When the recipient does that, that response goes through one of the switch ports, and that way, the switch is able to map the recipient's MAC address to one of its ports. With this, next time a packet is sent to that same recipient, the switch does not flood all of its ports but only sends the information through the cached one. The switch learns the MAC address when the response comes at the latest.

4. Does a switch keep the entries of its MAC address table forever? What problems could arise with this?

No, a switch does not keep the entries forever. For each MAC address in the table, the switch records a timestamp of when the information was learned. Each time the switch detects traffic from a MAC address that is in its table, it updates the timestamp. If the MAC address of a node is too old, the switch removes that MAC address from the table. This process ensures that the switch tracks only active MAC addresses on the network and that it is able to delete them from the table if they are no longer available or have changed its port.

5. Assume that the ARP tables of both devices (i.e., laptop and Raspberry) as well as the MAC address table of the switch are initially empty. Your laptop now sends 3 ping requests to the IP address of the Raspberry. Describe step by step which messages are sent from which host and where and when the information in the various tables is updated.

First, with all tables empty, the ping process on layer 3 on the laptop starts the ping request, which creates an ICMP Echo Request, with its own IP address as source and the destination IP address of the Raspberry which is in the same subnet. Then, it goes to layer 2, and the ARP process looks the IP address in the table. As it is not in the ARP table, the ARP process tries to send an ARP request for that IP address and buffers the packet. The laptop encapsulates the request into an Ethernet frame and sends it.

The switch then receives the frame and decapsulates it. The source MAC address does not exist in the MAC table of the switch, so it adds a new entry to the table. Since the port is in Learning state, the switch drops the frame. The ARP request in the laptop times out and the process drops this packet and the **first ping fails**.

The ping process starts a second request, with the same steps as in the first paragraph. Now, the source MAC address exists in the MAC table of the switch. The frame's destination MAC address is broadcast, and the switch processes the frame. Since it is in a broadcast, the switch sends out the frame to all ports except the receiving port.

The Raspberry receives the frame. As the MAC address destination matches (broadcast), the Raspberry decapsulates the frame, discovering is an ARP frame. The ARP process on the Raspberry processes it. As the request IP address matches the receiving port IP address, the frame is accepted. The Raspberry's ARP table is updated with the laptop's information. Finally, it replies to the request with the MAC address of the receiving port, encapsulating the frame and sending it back.

The switch now receives the frame from the Raspberry. As the source MAC address does not exist in the MAC table, the switch adds a new entry. Now the frame is unicast. The switch looks in its MAC table for the destination MAC address and finds it, therefore, sends the reply to the laptop.

The laptop receives the frame. The MAC address matches the receiving port's so the laptop decapsulates the frame. It is an ARP frame so the ARP process processes it. It is a reply, so the ARP table of the laptop is updated with Raspberry's information, and the ARP process sends the buffered packets waiting for this ARP reply, with the new MAC address obtained.

The frame is received by the switch, which now finds both the source and destination MAC addresses on its table. The switch forwards the frame through the correct port.

Raspberry receives the frame. It is for it so it is decapsulated. It is an ICMP packet, which the ICMP process processes, sending an Echo reply to the IP address received. The ARP looks into its table and finds a match for that IP address, and therefore, adds the destination MAC address to the frame. The reply is ready to be sent back to the laptop.

The returning frame is received again by the switch and is forwarded to the laptop. Both the MAC address and IP address match with those on the laptop, so the frame is decapsulated. The ICMP processes the Echo Reply message and the **Ping process returns successful**.

The Ping process start the third and last ping request. It is encapsulated in the same way in layer 3, but now, in layer 2, the ARP process finds the MAC address, and adds it to the frame. The frame now hops to the switch, which forwards it to the Raspberry. The Raspberry repeats the steps of the last successful ping, decapsulating, generating the Echo reply and sending it. The switch again forwards the frame, now back to the laptop, and **the ping is again received successfully**. A connection is stabilished.