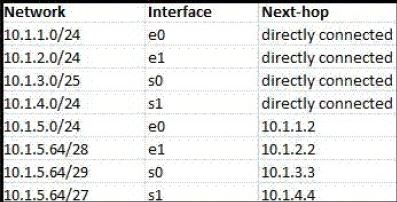
**Question 1:**



According to the routing table, where will the router send a packet destined for 10.1.5.65? Why?

Answer:

|  |  |
| --- | --- |
| 10 | 0000 1010 |
| 1 | 0000 0001 |
| 5 | 0000 0101 |
| 65 | 0100 0001 |

Other Networks

|  |  |
| --- | --- |
| 10 | 0000 1010 |
| 1 | 0000 0001 |
| 1 | 0000 0001 |
| 0 | 0000 0000 |

24

|  |  |
| --- | --- |
| 10 | 0000 1010 |
| 1 | 0000 0001 |
| 2 | 0000 0010 |
| 0 | 0000 0000 |

24

|  |  |
| --- | --- |
| 10 | 0000 1010 |
| 1 | 0000 0001 |
| 3 | 0000 0011 |
| 0 | 0100 0000 |

25

|  |  |
| --- | --- |
| 10 | 0000 1010 |
| 1 | 0000 0001 |
| 4 | 0000 0100 |
| 0 | 0100 0000 |

24

|  |  |
| --- | --- |
| 10 | 0000 1010 |
| 1 | 0000 0001 |
| 5 | 0000 0101 |
| 0 | 0100 0001 |

24

|  |  |
| --- | --- |
| 10 | 0000 1010 |
| 1 | 0000 0001 |
| 5 | 0000 0101 |
| 64 | 0100 0000 |

28

|  |  |
| --- | --- |
| 10 | 0000 1010 |
| 1 | 0000 0001 |
| 5 | 0000 0101 |
| 64 | 0100 0000 |

29 – This is the Best one

|  |  |
| --- | --- |
| 10 | 0000 1010 |
| 1 | 0000 0001 |
| 5 | 0000 0101 |
| 64 | 0100 0000 |

27

**Question 2:**

Classless Inter-domain Routing (CIDR) receives a packet with address 131.23.151.76. The router’s routing table has the following entries:

Prefix Output Interface Identifier

131.16.0.0/12 3

131.28.0.0/14 5

131.19.0.0/16 2

131.22.0.0/15 1

The identifier of the output interface on which this packet will be forwarded is \_\_\_\_\_\_. Why?

131.23.151.76 in binary is – (1000.0011).(0001.0111).(1001.0111).(0100.1100)

- 131.16.0.0/12 in binary: (1000.0011).(0001.0000) (12 bits)

- 131.28.0.0/14 in binary: (1000.0011).(0001.1100) (14 bits)

- 131.19.0.0/16 in binary: (1000.0011).(0001.0011) (16 bits)

- 131.22.0.0/15 in binary: (1000.0011).(0001.0110) (15 bits)

The longest matching prefix for the given packet's address (131.23.151.76) is 131.22.0.0/15 because it matches the first 15 bits of the packet's address. So, the packet will be forwarded on the output interface with identifier 1.

131.22.0.0/15 into 131.23.0.0/15

(1000.0011).(0001.0110) into (1000.0011).(0001.0111)

**Question 3:**

Consider the following routing table of a router.

| **PREFIX** | **NEXT HOP** |
| --- | --- |
| 192.24.0.0/18 | D |
| 192.24.12.0/22 | B |

Consider the following three IP addresses, what their next hop will be?

1. 192.24.6.0
2. 192.24.14.32
3. 192.24.54.0

Binary conversion:

PREFIX

192.24.0.0/18 – (1100.0000).(0001.1000)

192.24.12.0/22 - (1100.0000).(0001.1000).(0001.100)

1. 192.24.6.0:

In binary - (1100.0000).(0001.1000).(0000.0110)

Longest Prefix Match:

192.24.0.0/18 covers the first 18 bits (1100.0000).(0001.1000).(0000.0000), which matches IP address.

Next Hop: D

1. 192.24.14.32:

In binary – (1100.0000).(0001.1000).(0000.1110).(0010.0000)

Longest Prefix Match:

192.24.12.0/22 covers the first 22 bits (1100.0000).(0001.1000).(0000.1100), which matches IP address.

Next Hop: B

1. 192.24.54.0:

In binary - (1100.0000).(0001.1000).(0011.0110)

there is no exact match or longer prefix in the routing table for this IP address, so it depends on the router’s configuration.

**Question 4:**

Draw an TCP header. Capture packets using wireshark and explain the fields for a particular TCP packet captured. Try to explain the purpose of each field.

A screenshot of a computer

Description automatically generated

**Note: I can’t follow the TCP, won’t let me**

1. Source Port (16 bits): This field identifies the source application or process on the sender's side.
2. Destination Port (16 bits): This field identifies the destination application or process on the receiver's side.
3. Sequence Number (32 bits): This field is used to maintain the proper order of data segments and helps in detecting missing segments.
4. Acknowledgment Number (32 bits): If the ACK flag is set (Control Flags field, explained later), this field contains the sequence number of the next expected data byte.
5. Data Offset (4 bits): This field indicates the number of 32-bit words in the TCP header. It is used to locate the beginning of the data.
6. Reserved (6 bits): These bits are reserved for future use and should be set to zero.
7. Control Flags (6 bits):  
   These are various control flags used in TCP   
   including:   
    URG (Urgent): Indicates urgent data in the segment.

ACK (Acknowledgment): Acknowledges the receipt of data.

PSH (Push): Requests data to be pushed immediately to the application.

RST (Reset): Resets the connection.

SYN (Synchronize): Initiates a connection.

FIN (Finish): Closes the connection.

1. Window Size (16 bits): This field specifies the size of the sender's receive window, which indicates how much data can be sent before requiring an acknowledgment.
2. Checksum (16 bits): Used for error-checking to ensure the integrity of the TCP header and data.
3. Urgent Pointer (16 bits): If the URG flag is set, this field points to the last urgent data byte.
4. Options (variable): This field may contain various TCP options such as Maximum Segment Size (MSS), Window Scale, Timestamps, and others.

**Question 5:**

Draw an UDP header. Capture packets using wireshark and explain the fields for a particular UDP packet captured. Try to explain the purpose of each field.

UDP seems way more simple.

A screenshot of a computer

Description automatically generated

**Note: UDP Follow**

1. Source Port (16 bits): This field identifies the source application or process on the sender's side. It specifies the port number of the sender.
2. Destination Port (16 bits): This field identifies the destination application or process on the receiver's side. It specifies the port number of the intended recipient.
3. Length (16 bits): This field specifies the length of the UDP header and the UDP data in bytes. It includes the UDP header and the data. The minimum value for this field is 8 bytes (the length of the header itself).
4. Checksum (16 bits): The checksum field is optional but commonly used for error-checking. It allows the sender and receiver to verify the integrity of the UDP header and data.
5. Data (variable length): This field contains the actual data being sent by the UDP packet. The length of this field can vary, but it's determined by the value in the Length field.