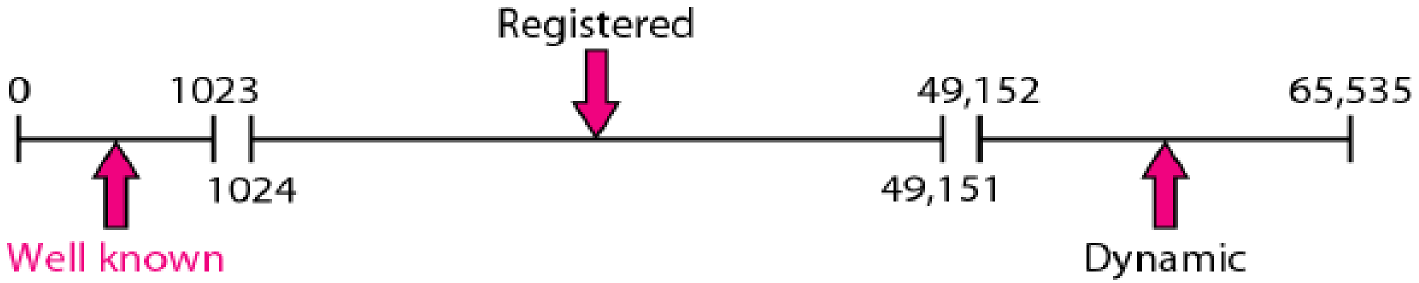
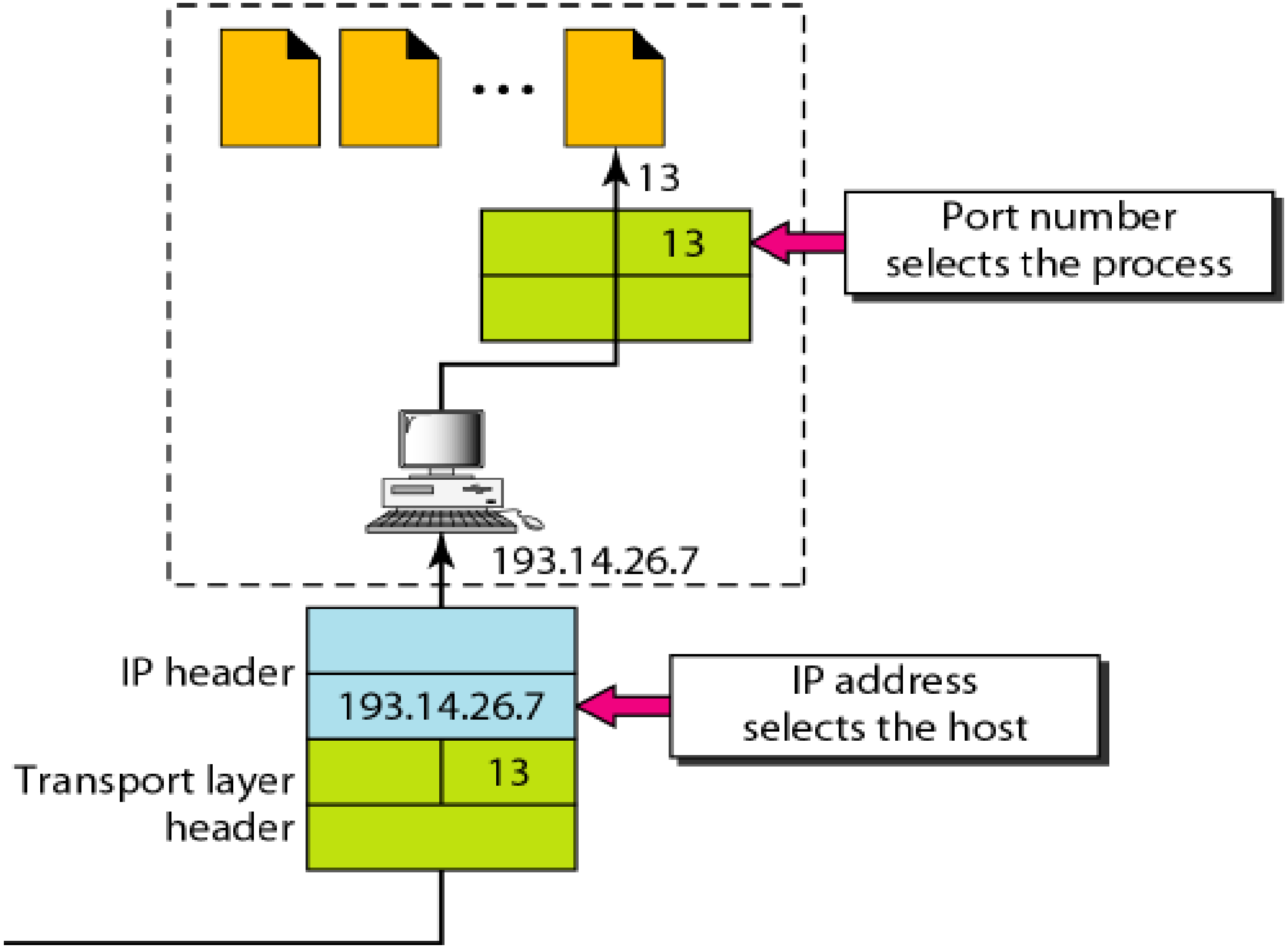
IANA
International Assigned Number Authority ranges

International Assigned Number Authority ranges:



- It should be clear by now that the IP address and port number play different roles in selecting the final destination of data.
- The destination IP address defines the host among the different hosts in the world.
- After the host has been selected, the port number defines one of the processes on this destination host.


13
13
Port number selects the process
193.14.26.7
IP header
193.14.26.7
IP address selects the host
Transport layer header
13

Connectionless Vs connection-oriented services

Connectionless services

- In a connectionless services, the packet are sent from one party to another with no need for connection establishment or connection release.
- The packet are not numbered; they may be delayed or lost or may arrive out of sequence.
- There is no acknowledgment.
- One of the transport layer protocol in the Internet model, UDP, is connectionless protocol.

Connection-oriented services

- In Connection-oriented service, a connection is first established between the sender and the receiver.
- Data are transferred.
- At the end, the connection is released.
- Two of the transport layer protocol in the Internet model, TCP and SCTP, is connection-oriented protocol.

Reliable Vs Unreliable

- The transport layer service can be reliable or unreliable.
- If the application layer program need reliability, we use a reliable transport layer protocol by implementing flow and error control at the transport layer.
- This means a slower and more complex service.

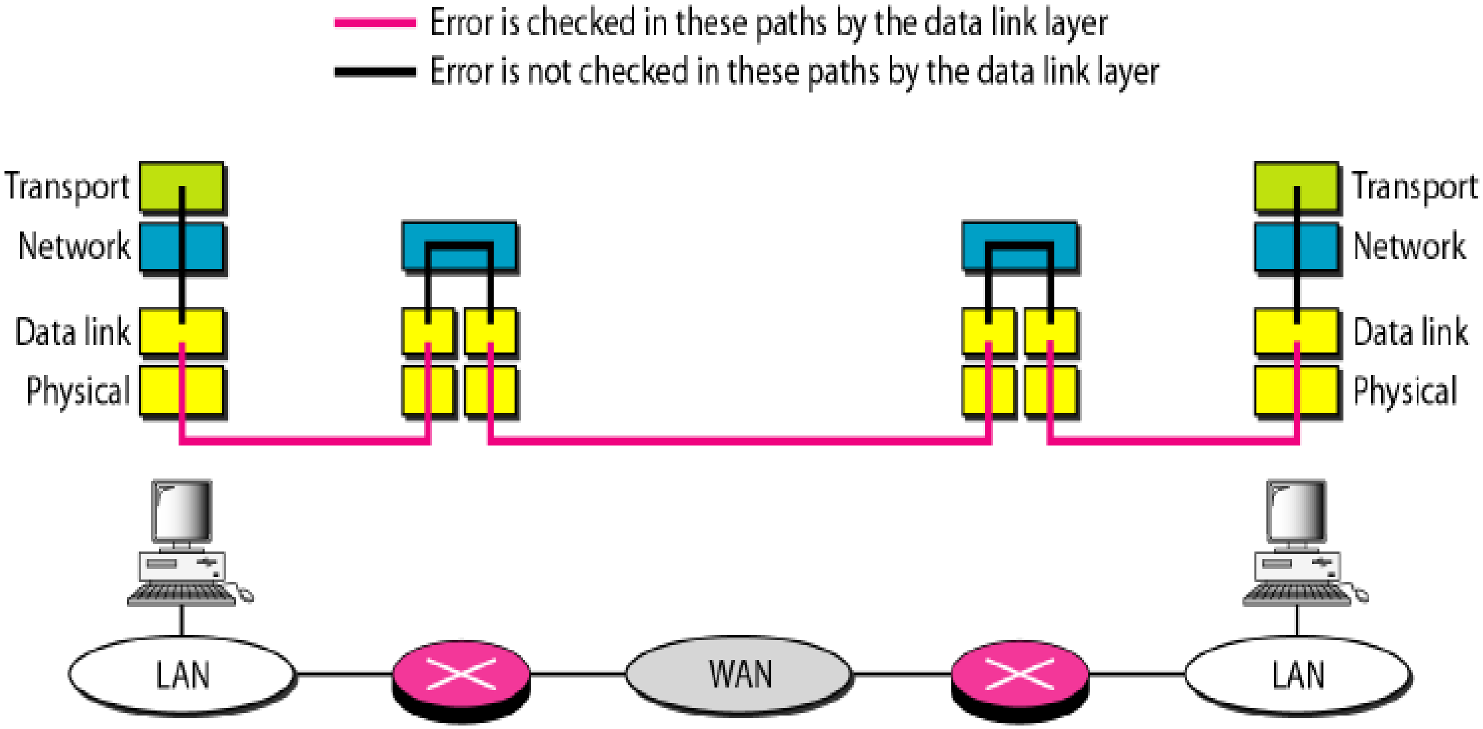
- On the other hand, if the application layer program does not need reliability because it uses its own flow and error control mechanism or it needs fast service or the nature of the service does nor demand flow and error control (real-time application), then an unreliable protocol can be used.

- In the Internet, there are three common different transport layer protocols:
- UDP: is connectionless and unreliable
- TCP and SCTP are connection oriented and reliable.

Question:

- If the data link layer is reliable and has a flow and error control, do we need this at the transport layer????

- Reliability at the data link layer is between two nodes (we need reliability between two ends).
- But the network layer in the internet model is unreliable, for this we need to implement reliability at the transport layer.


Error is checked in these paths by the data link layer
Error is not checked in these paths by the data link layer
Transport
Network
Data link
Physical
Transport
Network
Data link
Physical
LAN
WAN
LAN

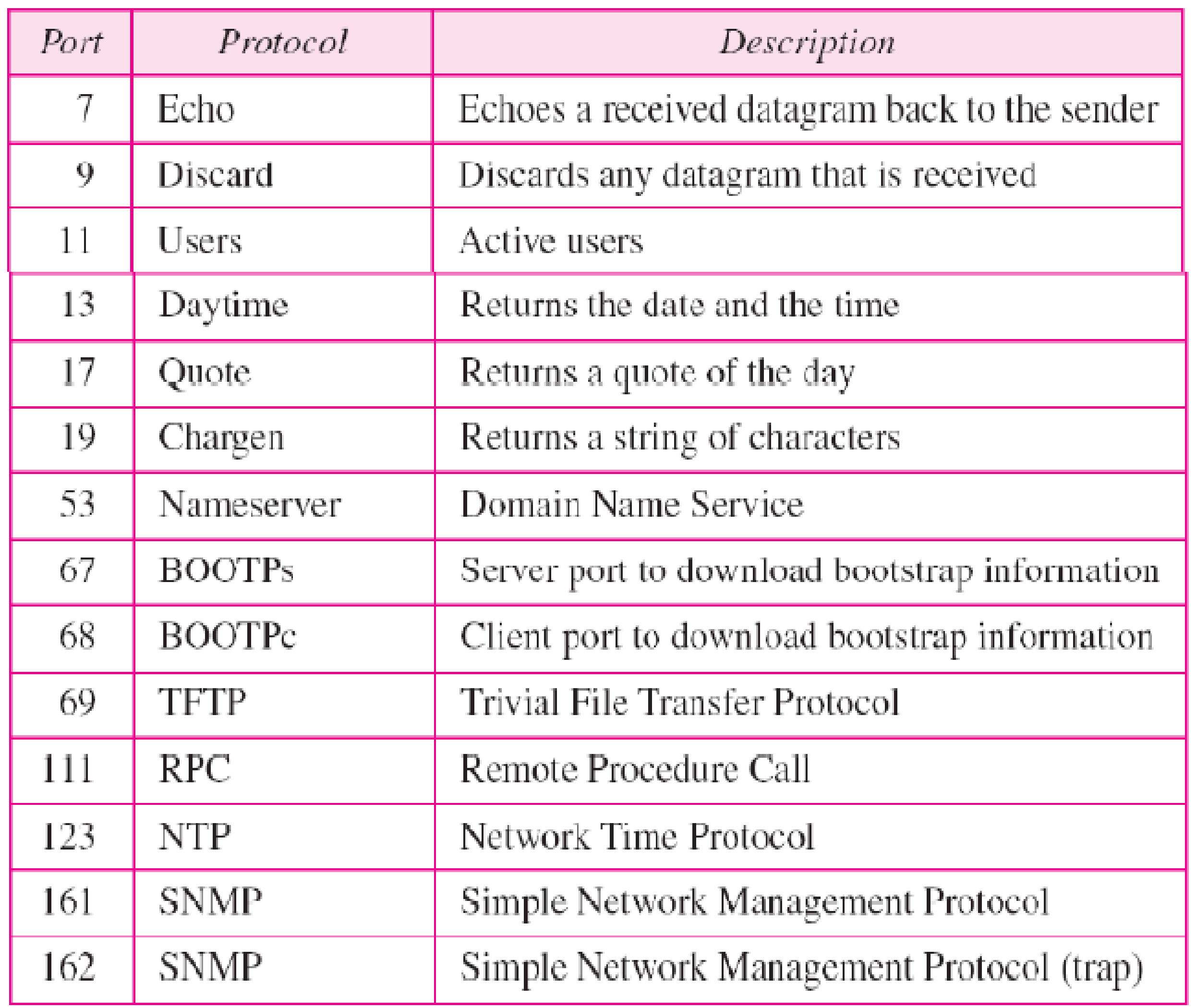
User datagram protocol
UDP

- The user Datagram Protocol (UDP) is called a connectionless, unreliable transport protocol.
- It does not add anything to the services of IP except to provide process-to-process communication instead of host-to- host communication
- It performs very limited error checking.

If UDP is powerless, why would a process want to use it???

- With the disadvantages come some advantages:
- UDP is very simple protocol using a minimum of overhead.
- If a process wants to send a small message and does not care much about reliability, it can use UDP
- Sending a small message by using UDP takes much less interaction between the sender and the receiver than using TCP.

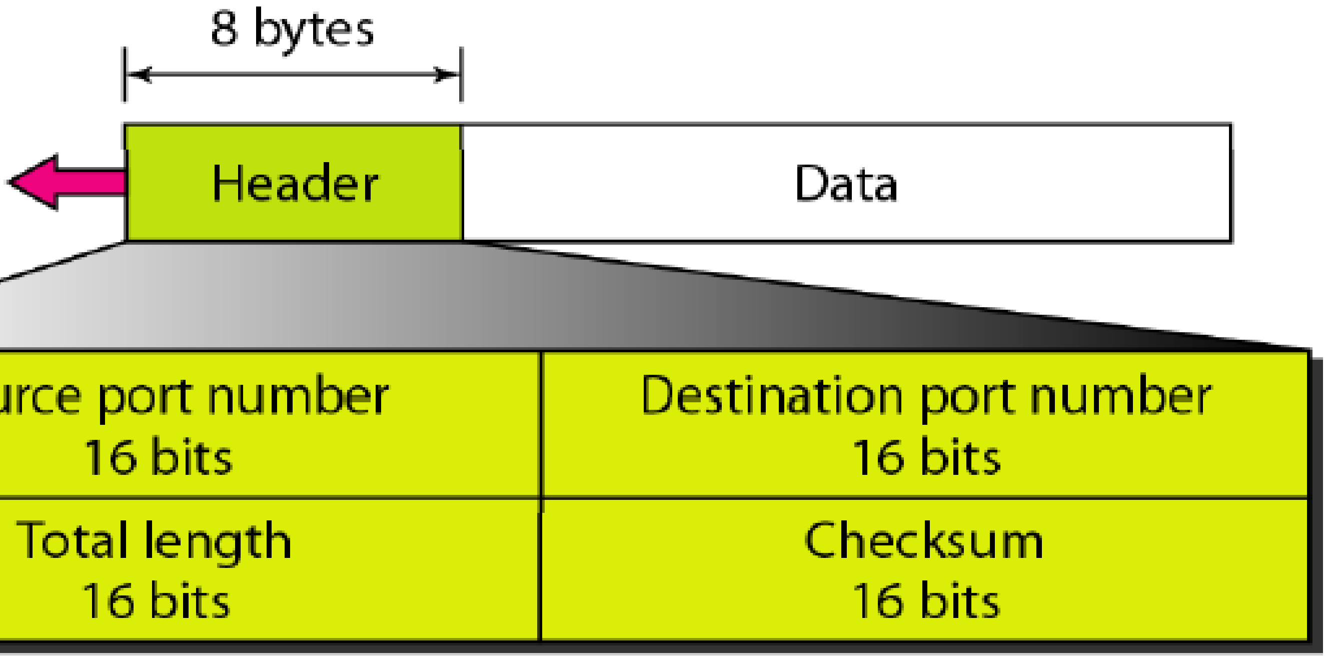
Well-known ports for UDP



Port	*Protocol*	*Description*
7	Echo	Echoes a received datagram back to the sender
9	Discard	Discards any datagram that is received
11	Users	Active users
13	Daytime	Returns the date and the time
17	Quote	Returns a quote of the day
19	Chargen	Returns a string of characters
53	Nameserver	Domain Name Service
67	BOOTPs	Server port to download bootstrap information
68	BOOTPc	Client port to download bootstrap information
69	TFTP	Trivial File Transfer Protocol
111	RPC	Remote Procedure Call
123	NTP	Network Time Protocol
161	SNMP	Simple Network Management Protocol
162	SNMP	Simple Network Management Protocol (trap)

USER DATAGRAM

UDP packets have a fixed size of 8 bytes. •



Source port number 16 bits	Destination port number 16 bits
Total length 16 bits	Checksum 16 bits

Source port number

- this is the port number used by the process running on the source host.
- It is 16 bits long, which means that the port number can range from 0 to 65535.
- If the source host is the client (a client sending a request), the port number is an ephemeral port number requested by the process and chosen by the UDP software running on the source host.
- If the source is the server (a server sending a response) the port number is a well-known port number.

destination port number

- this is the port number used by the process running on the destination host.
- It is 16 bits long.
- If the destination host is the server (a client sending a request), the port number well-known port number.
- If the destination host is the client (a server sending a response), the port number is an ephemeral port number.
- In this case, the server copies the ephemeral port number it has received in the request packet.

Length

- this is a 16 bit field that defines the total length of the user datagram,(header + data). The 16 bit can define a total length of 0 to 65535 bytes.
- However, the total length needs to be much less, because a UDP user datagram is stored in an IP datagram with a total length of 65535 Bytes.

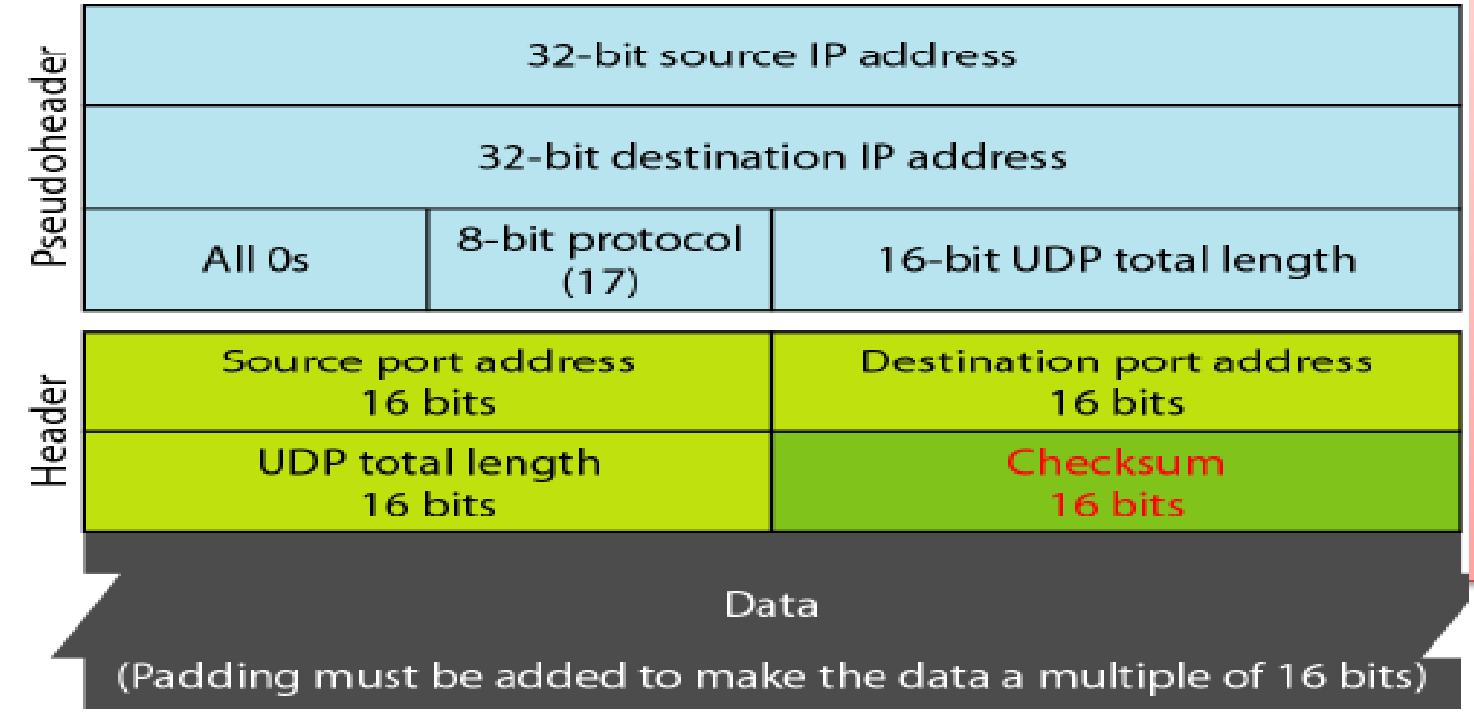
- The length field in a UDP user datagram is actually not necessary. Why???
- A user datagram is encapsulated in an IP datagram.
- There is a field in the IP datagram that defines the total length.
- There is another field in the IP datagram that defines the length of the header.
- By subtract the second field from the fist, we deduce the length of the UDP.

Checksum

- the UDP checksum calculation is different from the one for IP. For UDP, the checksum includes three sections:
- **Pseudoheader**
- **UDP header**
- **Data coming from application layer**

Pseudoheader

- is the part of the header of the IP packet in which the user datagram to be encapsulated with some fields filled with 0's.



Pseudoheader

32-bit source IP address

32-bit destination IP address

All 0s

8-bit protocol (17)

16-bit UDP total length

Header

Source port address 16 bits

Destination port address 16 bits

UDP total length 16 bits

Checksum 16 bits

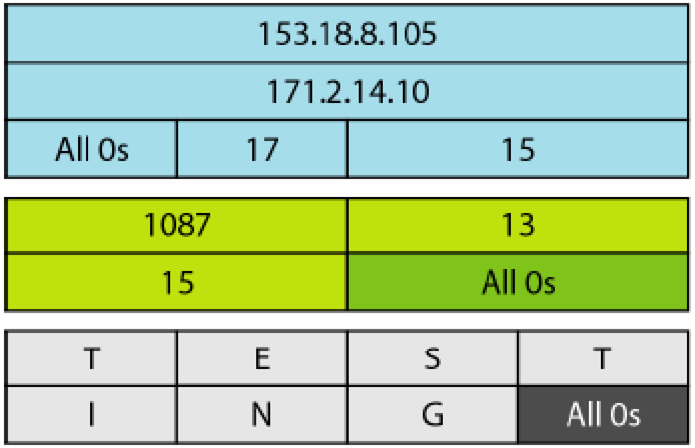
Data

(Padding must be added to make the data a multiple of 16 bits)

- If the checksum does not include the pseudoheader, a user datagram may arrive safe and sound.
- However, if the IP header is corrupted, it may be delivered to the wrong host.
- The protocol field is added to ensure that the packet belongs to the UDP, and not to other transport later protocols.
-

- If a process can use either UDP or TCP, the destination port number can be the same.
- The value of the protocol field for UDP is 17.
- If this value is changed during transmission, the checksum calculation at the receiver will detect it and UDP drops the packet.
- It is not delivered to the wrong protocol.

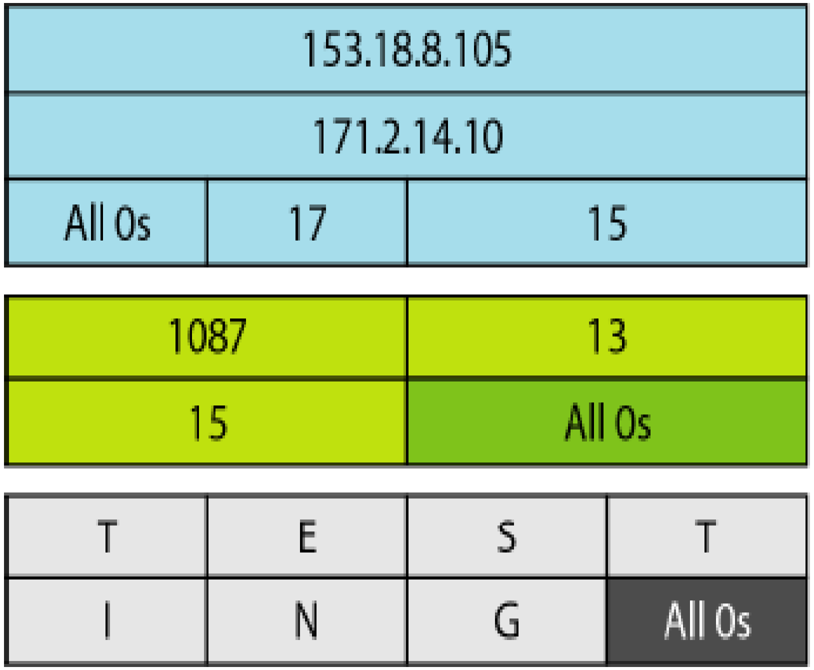
Example



<table>
<tr><td colspan="4">153.18.8.105</td></tr>
<tr><td colspan="4">171.2.14.10</td></tr>
<tr><td>All 0s</td><td>17</td><td colspan="2">15</td></tr>
</table>

<table>
<tr><td>1087</td><td>13</td></tr>
<tr><td>15</td><td>All 0s</td></tr>
</table>

<table>
<tr><td>T</td><td>E</td><td>S</td><td>T</td></tr>
<tr><td>I</td><td>N</td><td>G</td><td>All 0s</td></tr>
</table>



153.18.8.105			
171.2.14.10			
All 0s	17	15	
1087		13	
15		All 0s	
T	E	S	T
I	N	G	All 0s

10011001 00010010 → 153.18
00001000 01101001 → 8.105
10101011 00000010 → 171.2
00001110 00001010 → 14.10
00000000 00010001 → 0 and 17
00000000 00001111 → 15
00000100 00111111 → 1087
00000000 00001101 → 13
00000000 00001111 → 15
00000000 00000000 → 0 (checksum)
01010100 01000101 → T and E
01010011 01010100 → S and T
01001001 01001110 → I and N
01000111 00000000 → G and 0 (padding)

10010110 11101011 → Sum
01101001 00010100 → Checksum

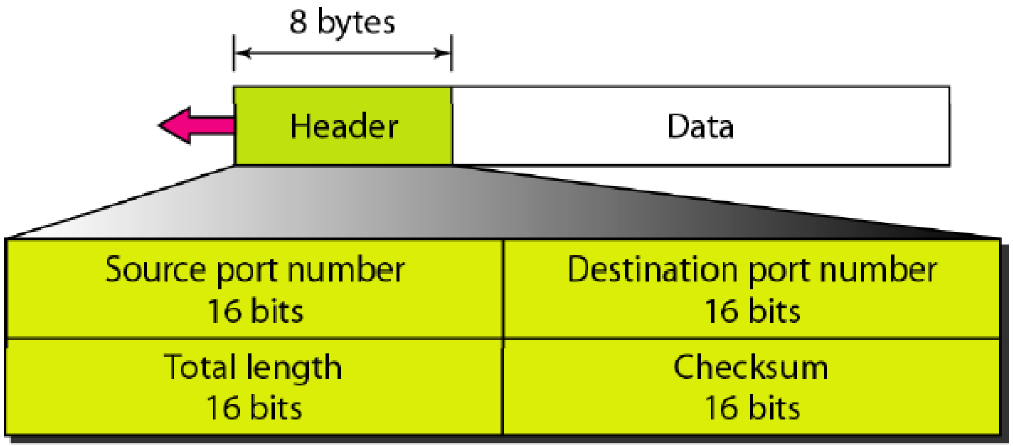
USE of UDP

- UDP is suitable for a process that requires simple request response communication with little concern for flow and error control. It is not usually used for a process such as FTP that needs to send bulk data.
- UDP is suitable for a process with internal flow and error control mechanism. Trivial File Transfer Protocol (TFTP).

- UDP is a suitable transport protocol for multicasting.
- UDP is used for management process such as SNMP
- UDP is used for some route updating protocols such RIP

Example

- Show the entries for the header of a UDP user datagram that carries a message from a TFTP client to a TFTP server.
- Fill the checksum field 0s.
- Choose an appropriate ephemeral port number and the correct well-known port number.
- The length of data is 40 bytes.
- Show the UDP packet.



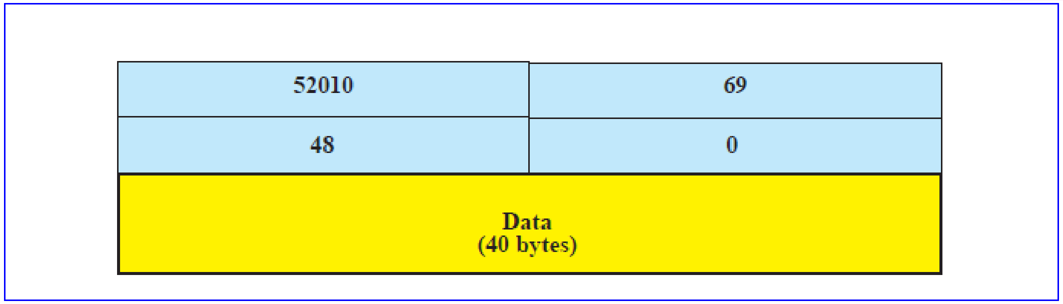
8 bytes

Header

Data

Source port number 16 bits	Destination port number 16 bits
Total length 16 bits	Checksum 16 bits

Port	*Protocol*	*Description*
7	Echo	Echoes a received datagram back to the sender
9	Discard	Discards any datagram that is received
11	Users	Active users
13	Daytime	Returns the date and the time
17	Quote	Returns a quote of the day
19	Chargen	Returns a string of characters
53	Nameserver	Domain Name Service
67	BOOTPs	Server port to download bootstrap information
68	BOOTPc	Client port to download bootstrap information
69	TFTP	Trivial File Transfer Protocol
111	RPC	Remote Procedure Call
123	NTP	Network Time Protocol
161	SNMP	Simple Network Management Protocol
162	SNMP	Simple Network Management Protocol (trap)

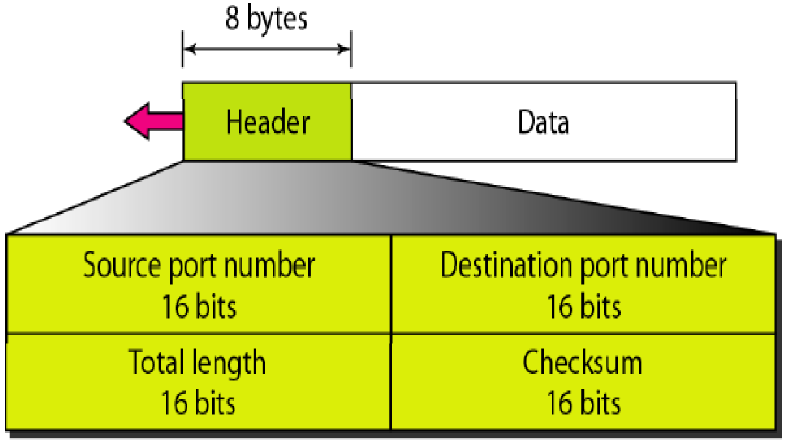


52010	69
48	0
Data **(40 bytes)**	

Example

- The following is a dump of a UDP header in hexadecimal format: 06 32 00 0D 00 1C E2 17
- What is the source port number?
- What is the destination port number?
- What is the total length of the user datagram?
- What is the length of the data?
- Is the packet directed from a client to a server or vice versa?

What is the client process? •



- a. Port number **1586**
- b. Port number **13**
- c. **28** bytes
- d. **20** bytes (28 – 8 byte header)
- e. ***From a client to a server***
- f. ***Daytime***