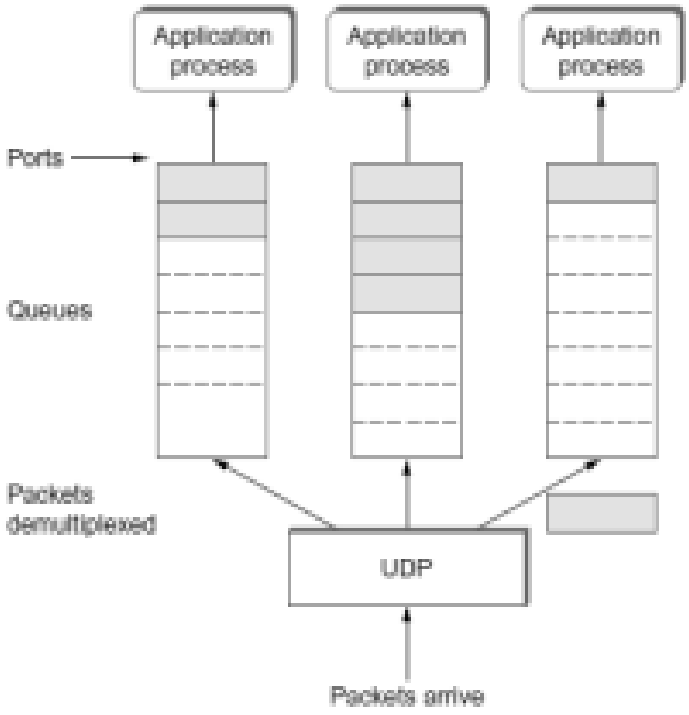
User Datagram Protocol (UDP) Basics:

- UDP is a straightforward way for processes to communicate.
- It adds process-to-process communication to basic IP services.
- UDP is minimalistic and doesn't have many extra features.
- It's like a direct line for processes on the same computer.
- UDP doesn't guarantee message order, flow control, or reliability.
- Best for quick, small, and non-critical messages.
- Requires less back-and-forth between sender and receiver.
- Processes use "ports" to find each other indirectly.

UDP Ports:

- Ports are like addresses for processes (servers/clients).
- Servers listen for messages at specific, well-known ports.
- Some well-known UDP ports include 7 (Echo), 53 (DNS), 111 (RPC), 161 (SNMP), and more.
- Messages are sorted and directed using <port, host> pairs.
- Ports work like message queues.
- Incoming messages are added to the end of the queue.
- If the queue is full, messages are discarded.
- Reading a message removes it from the queue.
- When an app wants a message, it takes one from the front.
- If the queue is empty, the process waits.



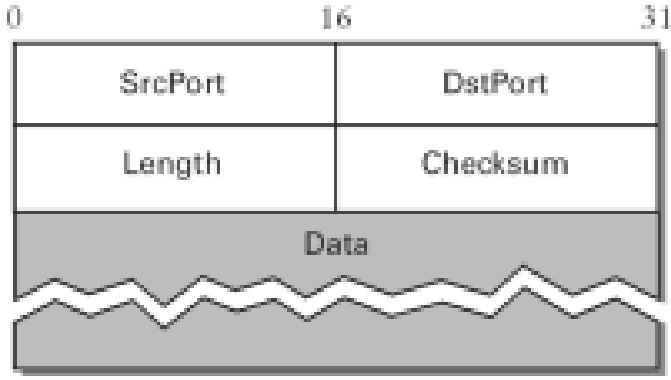
UDP Packet Structure:

- UDP packets consist of an 8-byte header with four 2-byte fields.

Source Port Number:

- 16-bit address used by the sender's process.

- Clients use temporary ports, while servers use well-known ports.



Destination Port Number:

- 16-bit address used by the receiver's process.

- Servers use well-known ports, clients copy temporary ones from requests.

UDP Packet Fields (Continued):

- Length: Total size of the UDP Packet, including header and data, from 0 to 65,535 bytes.

- Checksum: Error-checking using a checksum. It looks at the UDP header, message contents, and a "pseudoheader" including IP header data.

- Data: Carries the actual information, with variable size.

UDP Services:

- Process-to-Process Communication: UDP connects processes using "socket addresses" combining IP addresses and port numbers.

- Connectionless Services: No connection setup or termination. Each message stands alone, even from the same source to the same destination.

- Flow Control: UDP doesn't provide it, so processes handle it if needed.

- Error Control: Except for the checksum, UDP doesn't provide error control. The sender can't tell if a message was lost or duplicated.

Error Control in UDP:

- UDP doesn't handle errors, so processes using UDP must manage error control.

Checksum in UDP:

- UDP checks for errors using a checksum, considering the UDP header, message, and a pseudoheader. The pseudoheader includes data from the IP header.

- Optionally, the sender can skip calculating the checksum. In this case, the checksum field is filled with all 0s before sending. If the sender calculates it and gets all 0s, it changes to all 1s before sending.

Applications of UDP:

- Used for network management processes like SNMP.

- Applied in route updating protocols like RIP.

- Suitable for multicasting, where one message goes to many.

- Fits processes like TFTP with internal flow and error control.

- Useful for simple request-response communication, even with minimal error control.

- Common in real-time applications needing consistent message delivery times.

- DHCP makes it easy to set up and take care of computers in a network.

- DHCP comes from an older protocol called BOOTP.

- Ethernet addresses are like unique IDs given by the manufacturer.

- IP addresses need to be unique on the network but also fit its design.

- Normally, you can set up IP info for a computer yourself.

Drawbacks of Manual Configuration:

1. Manually setting up all the computers in a big network is a lot of work.

2. Manual setup can often lead to mistakes.

- Ensuring every host gets the correct network number and unique IP addresses is essential.

- To overcome these challenges, automated configuration methods are needed.

- Dynamic Host Configuration Protocol (DHCP) is a primary solution.

- DHCP's main goal is to reduce manual configuration for hosts.

- When a new computer joins a network, DHCP provides all the necessary information for seamless integration.

- DHCP operates using a client/server model.

- DHCP clients request IP addresses from DHCP servers.

- At least one DHCP server is needed per administrative domain.

- DHCP servers act as centralized repositories for host configuration data.

- DHCP servers maintain a pool of available IP addresses for on-demand allocation.

- A new or connected host broadcasts a DHCPDISCOVER message to a special IP address (255.255.255.255), received by all hosts and routers.

- DHCP uses relay agents, with at least one on each network.

- Relay agents forward DHCPDISCOVER messages to DHCP servers and send the server's response back to the requesting client.

DHCP Message Format:

- DHCP messages use a protocol called User Datagram.