User Datagram Protocol (UDP)

Kameswari Chebrolu

Recap

- Transport layer provides logical communication between processes
- Internet supports a few transport layer protocols
  - UDP, TCP, RPC, RTP
- UDP: ‘bare bones’ transport protocol

User Datagram Protocol

- Provides Mux/Demux capability over best-effort network layer service
- UDP segments can be lost, duplicated, delivered out of order to applications
- Connectionless: no handshaking between UDP sender, receiver
- Each UDP segment handled independently of others

Why used?

- No connection establishment (which can add delay)
  - DNS uses UDP
- Simple: no connection state at sender, receiver
  - A server can support more clients
- Small segment header: Less overhead per packet

Why used?

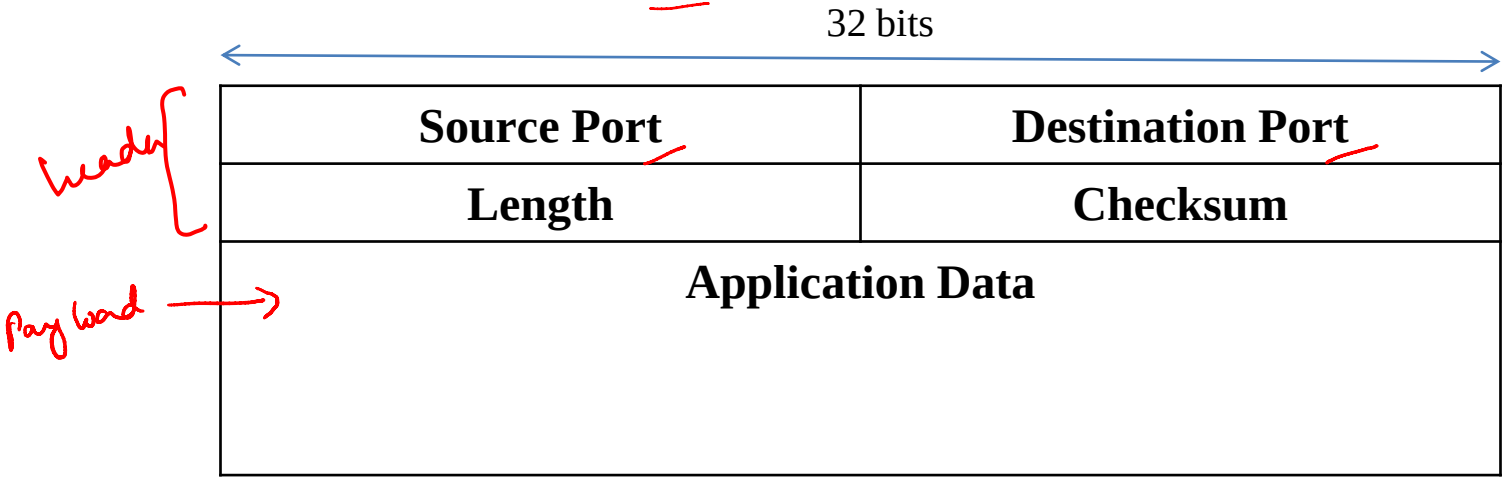
- No congestion control: UDP can blast away as fast as desired
- No retransmission delays: Useful for real-time applications like VoIP, online games
- Want additional features? Applications have to implement them themselves

Example Protocols

- DHCP
- RIP
- DNS
- SNMP (Simple Network Management Protocol)
  - Used for managing nodes (switches, routers, printers, servers etc) on IP networks

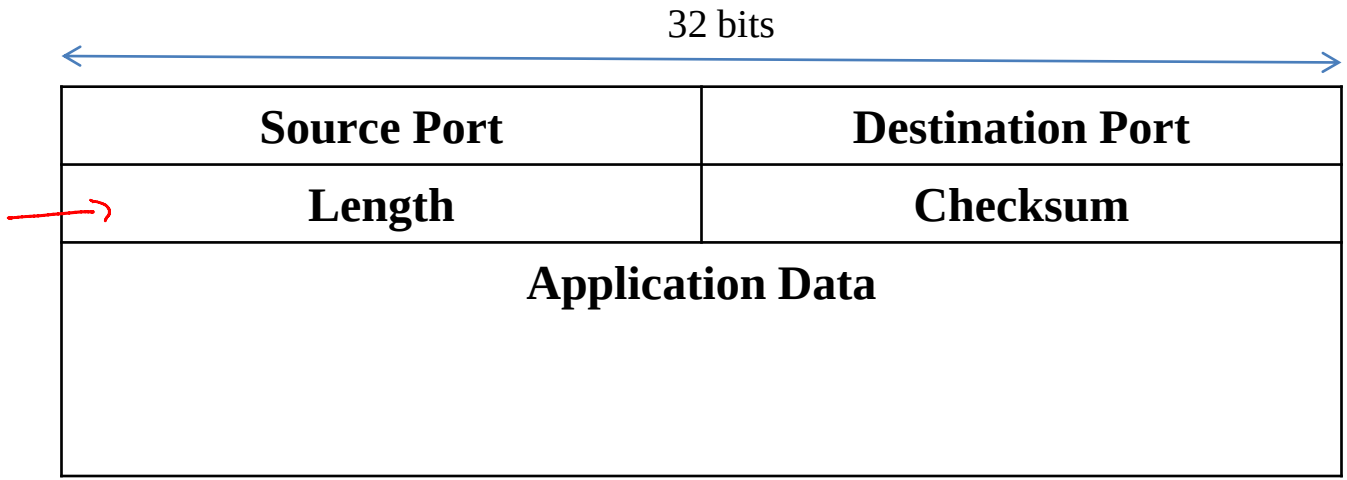
UDP Segment Format

- Source/Destination Port: Identifies sending/receiving process
  - Client: Ephemeral port; Server: Well-known port



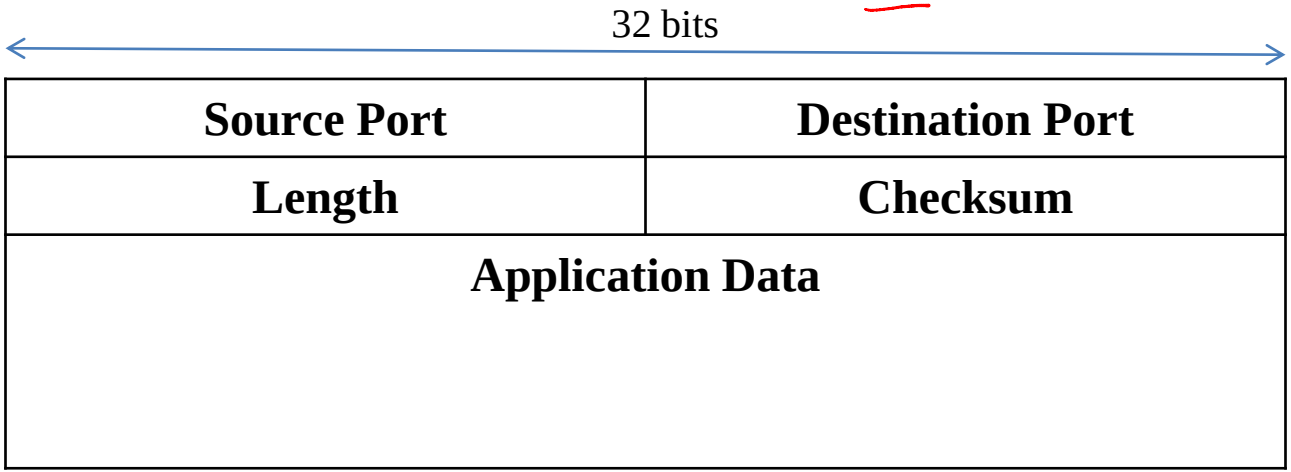
UDP Segment Format

- Length: Specifies the total length of the segment in bytes



Checksum

- Optional in IPv4, Compulsory in IPv6
- Ensures correctness of message
- Uses same algorithm as IP checksum



Checksum

- Calculated over UDP header, body and pseudoheader
  - Pseudoheader: three fields from IP (protocol number, source IP, destination IP) and UDP length field
  - Pseudoheader included to help verify if packet is indeed delivered to the right host

Summary

- UDP is a simple transport protocol
- Provides multiplexing/demultiplexing and simple error detection capability
- Finds good use in many protocols in spite of its simplicity