Module 15
(TCP in More Detail)

• In this module, we find out more detail about TCP packets. We see what many of the fields in the packet represent and see how TCP connections are opened (with a three-way handshake) and closed (with a four-way handshake).
TCP Source Port

- Source Port (16 bits)
- Ports are *conceptual* locations (really just numbers).
- The source port is associated with the program on the local machine.
TCP Destination Port

- Destination port (16 bits)
- The destination port is associated with the program being contacted on the remote machine.
TCP Sequence Number

- The TCP Sequence Number identifies which packet this is in the TCP stream. Packets may arrive out-of-order and must be reassembled in order.
TCP Acknowledgment Number

- The Acknowledgment number is the sequence number of the next packet expected to be received. It asserts that all packets with lower sequence numbers have been received.
TCP Data Offset

- The data offset is the size of the TCP header in 32 bit words (also the offset to where the packet data begins).
TCP Flags

- Flags identify any control information communicated by a packet.
TCP Flags

- NS
- CWR
- ECE
- URG
- ACK
- PSH
- RST
- SYN
- FIN
TCP Checksum

- Checksum for verifying Header and Data (leaving Checksum = 0).
Connection Establishment

• To establish a connection on a TCP port, a *three-way handshake* is used:
  
  1. Client sends a packet with FLAGS==SYN message to the server with sequence number A (chosen randomly).
  2. Server replies with a packet with FLAGS==SYN|ACK and acknowledgment field = A+1 and Sequence number B (chosen randomly).
Connection Termination

• Connection Termination involves a four-way handshake:
  1. Machine A sends packet with FLAGS==FIN
  2. Machine B sends packet with FLAGS==ACK
  3. Machine B sends packet with FLAGS==FIN
  4. Machine A sends packet with FLAGS==ACK

• Packets 2 and 3 can be combined (FIN|ACK) for a 3-way disconnect.

• If one side has sent FIN but the other has not, the connection is said to be half-open.