
SYSC 3303 Real-Time Concurrent Systems

The TFTP Protocol (RFC 1350) Part 1

- Copyright © 2000-2003 D.L. Bailey, 2004 L.S. Marshall Systems and Computer Engineering, Carleton University
- revised February 26th, 2004

Trivial File Transfer Protocol

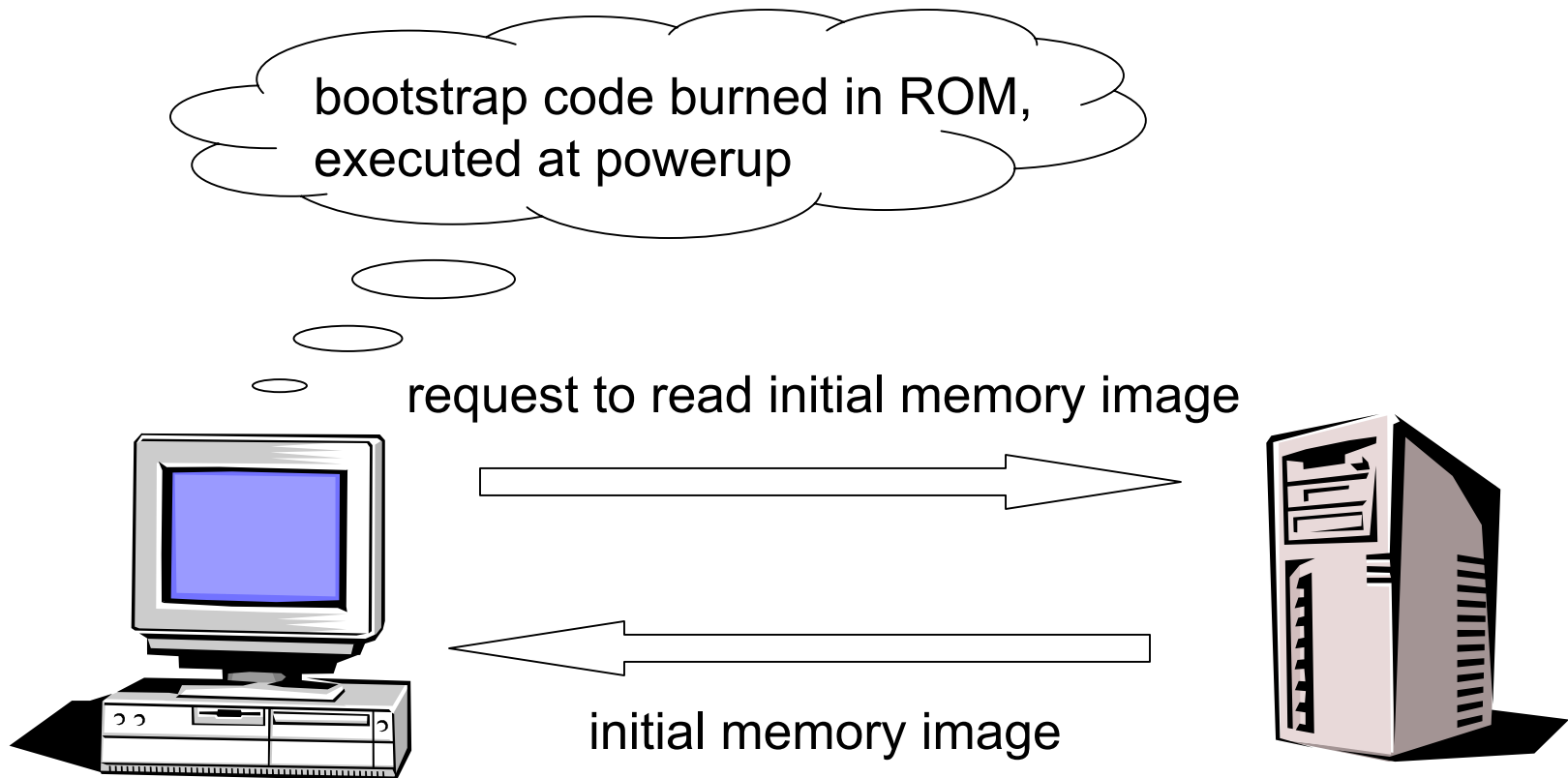
- Internet Protocol Standard STD 33
- Provides inexpensive, unsophisticated file transfer between a client and a server across the Internet
- Described by RFC (Request for Comments) 1350, July 1992
 - an earlier document describing TFTP Revision 2 (RFC 783, June 1981) is critiqued in RFC 1123, *Requirements for Internet Hosts -- Application and Support*, October 1989, and lead to the revisions found in RFC 1350

Trivial File Transfer Protocol

- Much simpler than the standard Internet file transfer protocol (FTP)
- Unlike FTP, TFTP provides no login or access control mechanisms
 - TFTP server processes should have limited rights, so that they do not violate the security of the server host's file system

Application: Bootstrapping Diskless Workstations

- A diskless workstation uses TFTP to download an O/S from a server over a network



Protocol Overview

- Client sends request to server to read or write a file
- If server grants request, a connection is established and file is transferred in 512-byte blocks encapsulated in data packets (1 block/data packet)
- Each data packet must be acknowledged by an acknowledgement packet before the next data packet can be sent
- A block of data < 512 bytes terminates the transfer
- Reliable transfer is enforced by a simple timeout/retransmit protocol

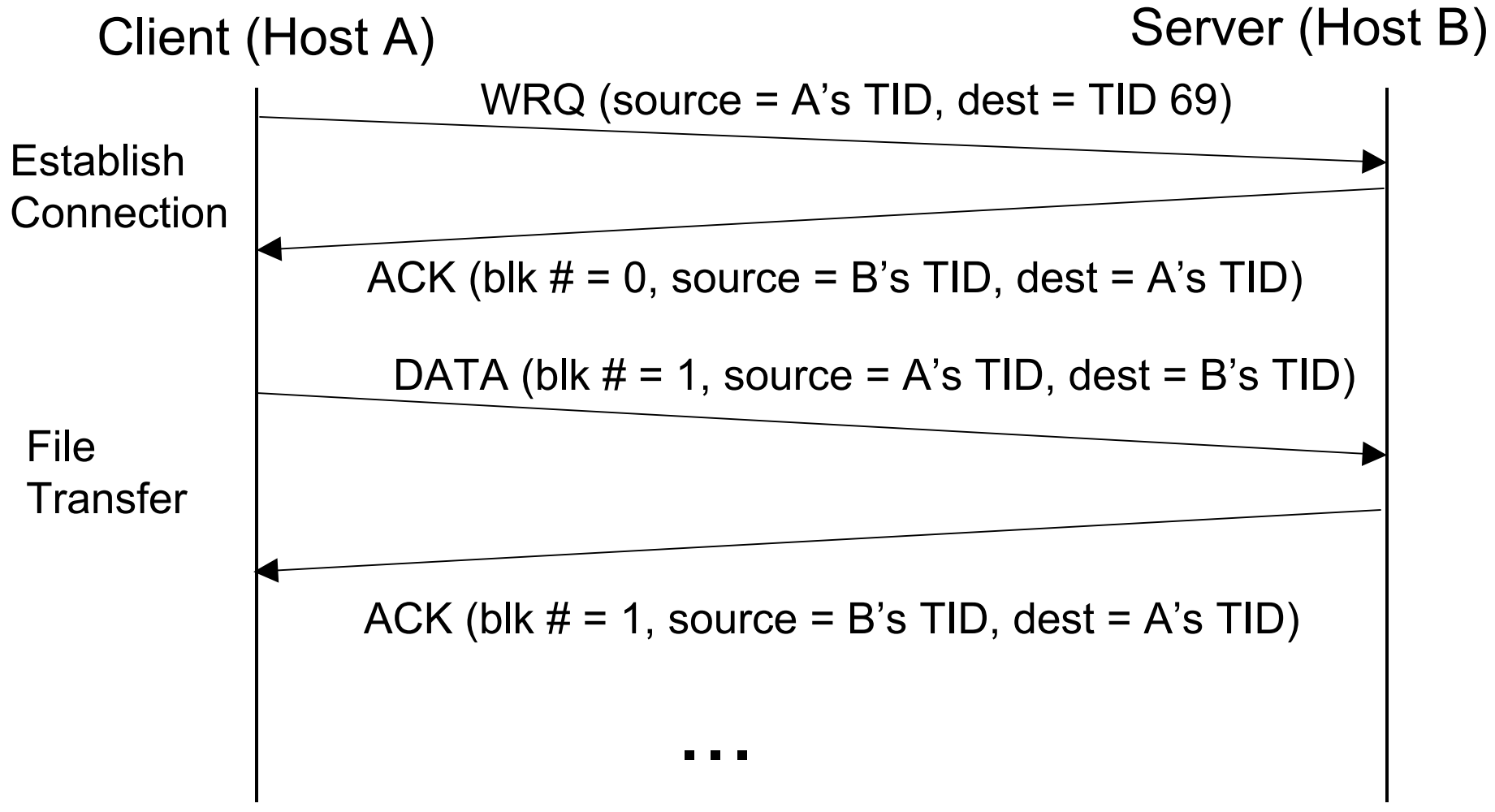
TFTP Packet Types

- Read request (RRQ) (Figure 5-1 in RFC 1350)
- Write request (WRQ) (Figure 5-1)
- Data (DATA) (Figure 5-2)
- Acknowledgement (ACK) (Figure 5-3)
- Error (ERROR) (Figure 5-4)

Transfer IDs

- When a client prepares a request it selects a random integer *Transfer ID* (TID) for itself
- When the server grants the request and establishes the connection it selects a random TID for itself
- A source TID and destination TID are associated with every TFTP packet
- TIDs are not stored in the TFTP packets
- Instead, when TFTP is implemented on top of UDP, the TIDs are used as the source and destination ports for UDP datagrams

Write File from Client to Server



Write File from Client to Server (cont'd)

Client (Host A)

Server (Host B)

...

DATA (blk # = $n-1$, source = A's TID, dest = B's TID)

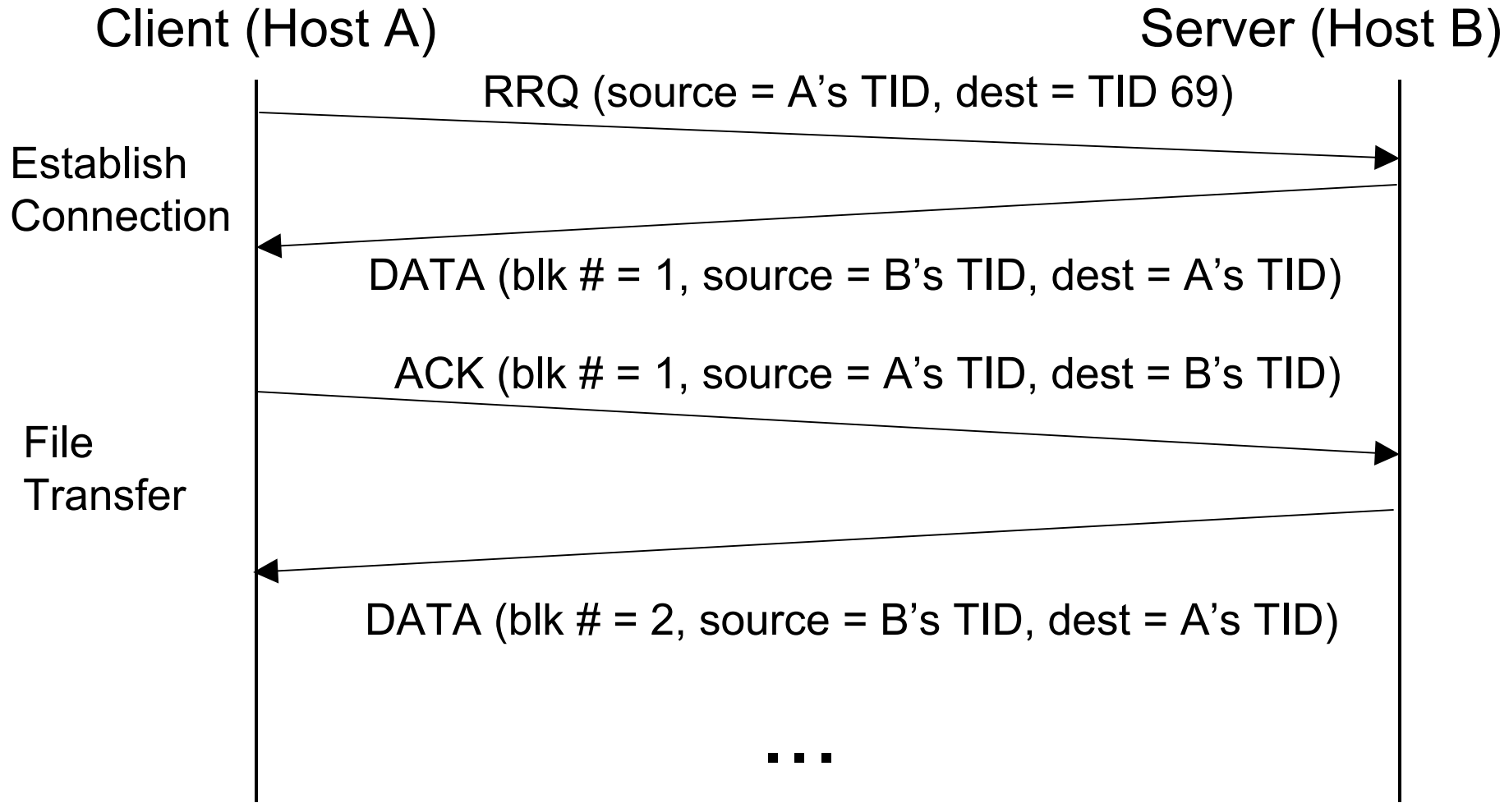
ACK (blk # = $n-1$, source = B's TID, dest = A's TID)

DATA (blk # = n , source = A's TID, dest = B's TID)
(Data packet contains 0..511 bytes of data)

ACK (blk # = n , source = B's TID, dest = A's TID)

Transfer
Last Block
of File

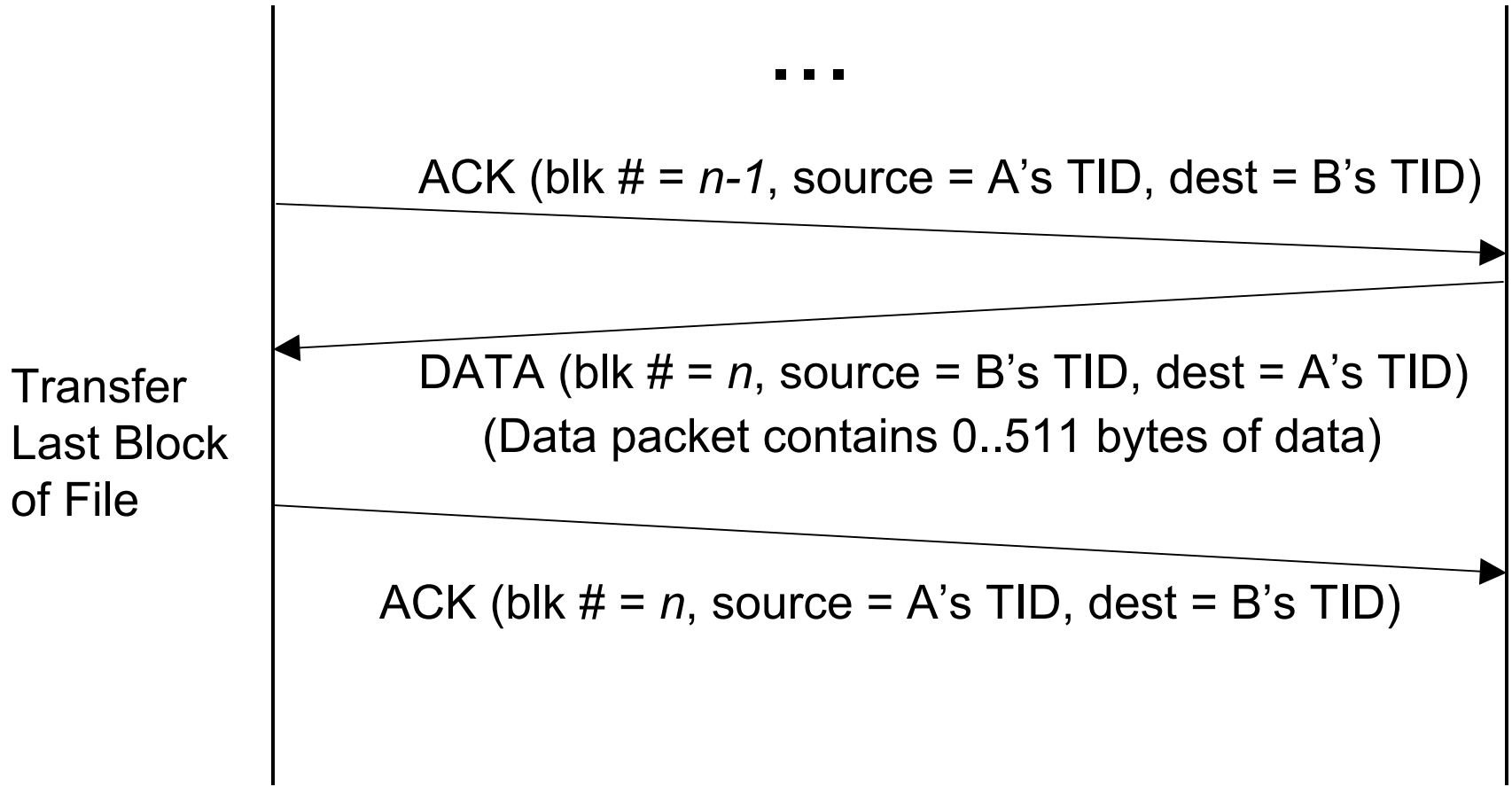
Read File from Server to Client



Read File from Server to Client (cont'd)

Client (Host A)

Server (Host B)

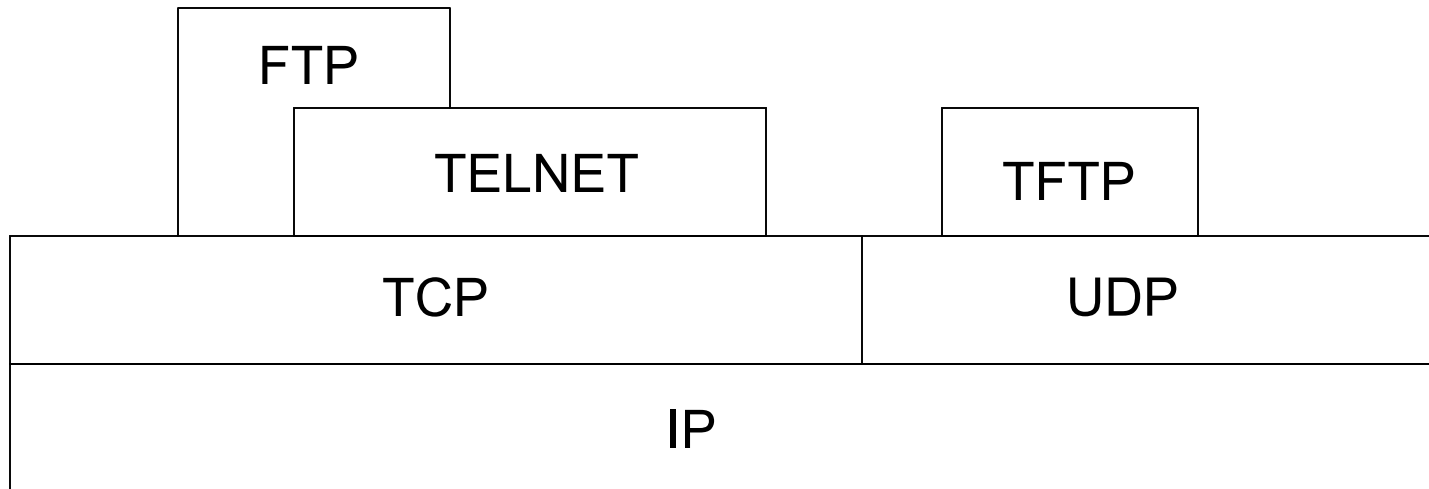


Acknowledgements

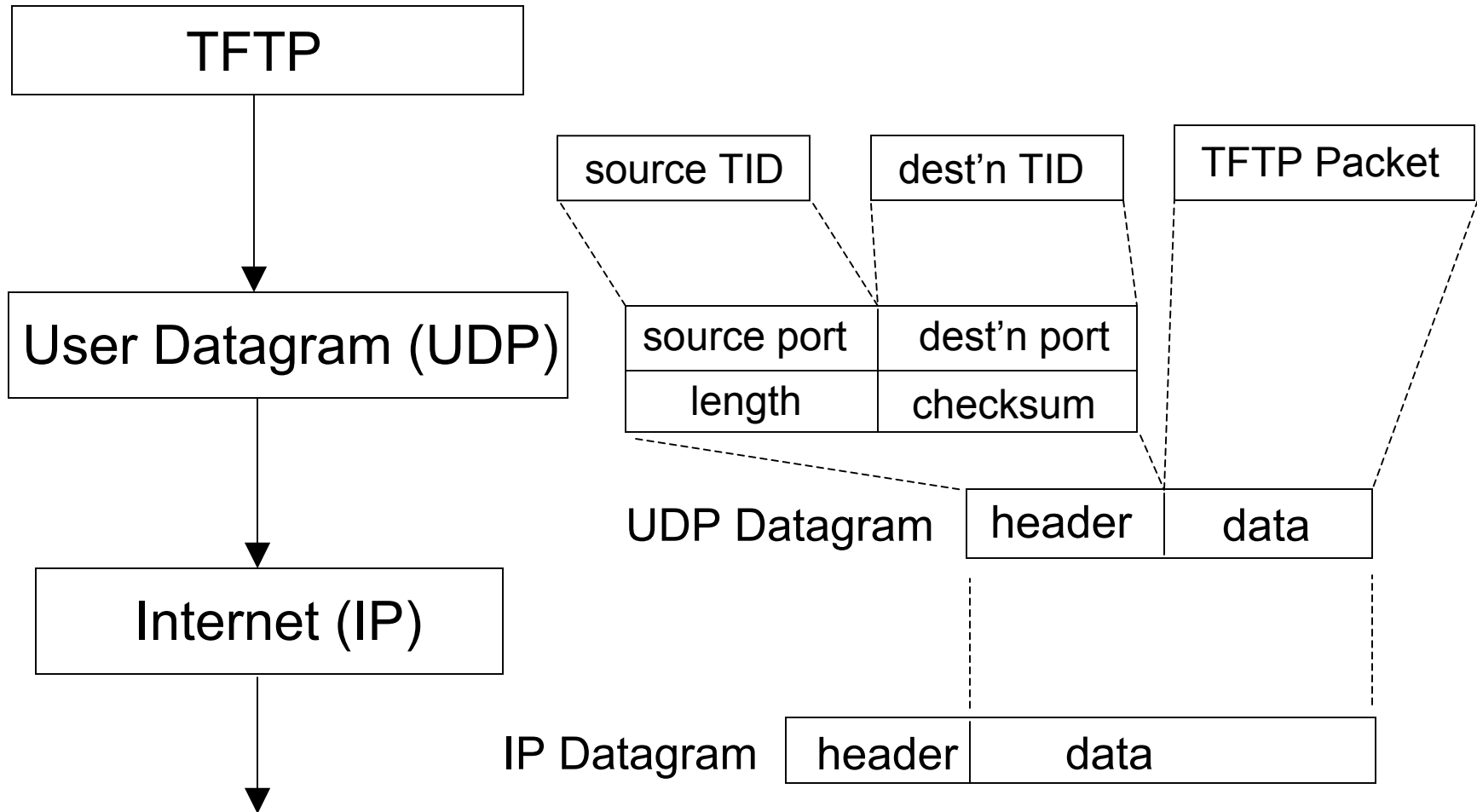
- Initial connection:
 - WRQ packets are acknowledged by ACK packets
 - RRQ packets are acknowledged by DATA packets
- File transfer:
 - DATA packets are acknowledged by ACK packets
 - ACK packets are acknowledged by DATA packets

Relationship to Other Internet Protocols

- TFTP is built on top of UDP (or any unreliable packet delivery system)



TFTP Built On UDP



Other Issues

- What happens if a packet is lost?
- What happens if a duplicate packet is received?
- When are ERROR packets sent?
- What happens if an ERROR packet is received?
- We'll discuss these later...