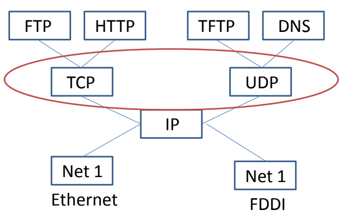
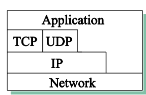
Fundamentals of Computer Networks ECE 478/578

Lecture #19: Transport Layer
Instructor: Loukas Lazos
Dept of Electrical and Computer Engineering
University of Arizona

The Internet Architecture





FTP: File Transfer Protocol
HTTP: Hypertext Transport Protocol
TFTP: Trivial File Transfer Protocol
DNS: Domain Name System

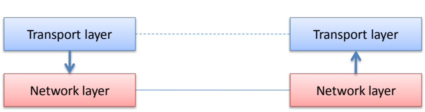
TCP: Transmission Control Protocol
UDP: User Datagram Protocol
IP: Internet Protocol

2

End-to-End Protocols

Responsible for the realization of host-to-host packet delivery

Services of the transport layer
- Guaranteed message delivery (end-to-end)
- Ordered delivery
- Rejection of duplicate messages
- Messages of arbitrary length
- Congestion control to handle network overloading
- Running of multiple application processes at the same host



3

Why do we need a Transport Layer?

Limitations of the network layer

- Can drop messages (routers may incurr buffer overflow etc.)
- Can reorder messages
- Deliver duplicate copies of the same message
- Limit messages to finite size (eg. IP size is limited)

Network layer is intended to hide the different technologies on a link level and provide a transparent end to end routing service

Transport layer provides end-to-end transport service to the application layer

4

Internet Transport Protocols

Datagram messaging service (UDP)

- "Best-effort" IP end-to-end service

Reliable, in-order delivery (TCP)

- Connection set-up
- Discarding of corrupted packets
- Retransmission of lost packets
- Flow control
- Congestion control (What is the difference?)

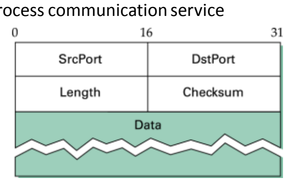
Remote Procedure Call (RPC)

- Request/Reply Service

5

User Datagram Protocol (UDP)

Process-to-process communication service



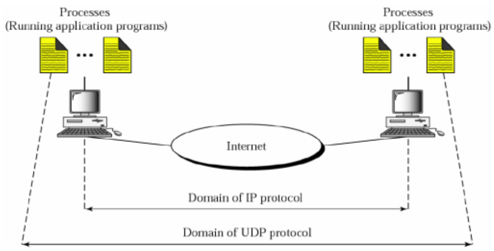
Processes are identified based on incoming ports (sockets in Windows)

16 bits for each field yields 64K different identifiers

<IP, port> combination allows de-multiplexing at receiving host

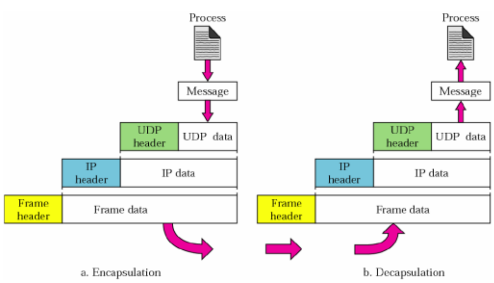
6

IP vs. Transport



7

The Encapsulation Process



8

Port Discovery

Use well-publicized ports for different services

- DNS uses to port 53
- Email uses port 25
- HTTP uses port 80

Use one port as a "port-mapper" service

- Call 411 to learn the port of any other process
- Allows for dynamic allocation of ports to different services
- Allows for the assignment of ports to newly created services

9

Multiplexing and De-multiplexing

Host may be running multiple processes at the same time

These processes

- Generate multiple messages for the same host
- Generate multiple messages for multiple hosts

Transport layer multiplexing

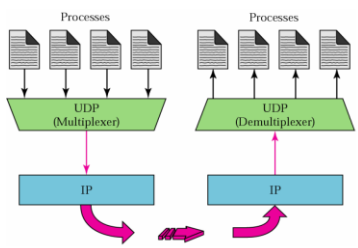
- Multiplex messages from multiple processes
- Break down messages to segments and pass to network layer

Transport layer de-multiplexing

- Reassemble messages at the receiving host and pass to the communication processes

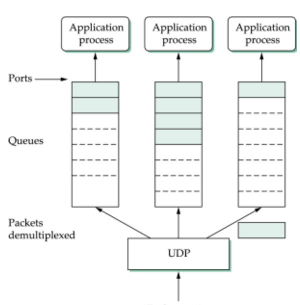
10

Multiplexing and De-multiplexing



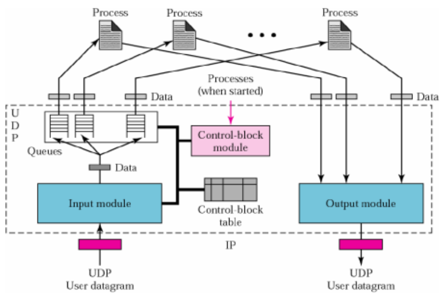
11

UDP Message Queue



12

Example of UDP



13

Control Block Module

Responsible for the management of the Control Block Table

- Request for a port for a new starting process
- Update the control block table

Example

State	Process ID	Port Number	Queue Number
In Use	2,569	52,010	34
In Use	4,759	52,011	
Free	-	-	-
In Use	7,489	80	26
Free	-	-	-

14

Input Module

Receive UDP packet from IP layer

Look up control block table to map port

If a queue exists, push packet to the corresponding queue

If not, allocate queue for the new process

State	Process ID	Port Number	Queue Number
In Use	2,569	52,010	34
In Use	4,759	52,011	**45**
Free	-	-	-
In Use	7,489	80	26
Free	-	-	-

15

Output Module

Receive data from the application layer
Create a UDP packet and send it.

16

Advantages of UDP

Control over what data is sent and when
- As soon as an application process writes into the socket
- … UDP will package the data and send the packet

No delay for connection establishment
- UDP just sends messages without contacting the host first
- Pays off when host is expecting messages anyway

Stateless connection
- No allocation of buffers, parameters, sequence #s, etc.
- … making it easier to handle many active clients at once (think of servers)

Small packet header overhead
- UDP header is only eight-bytes long

17

Disadvantages of UDP

"Best effort" networking
- No guarantee delivery of messages to destination host, no ordered delivery

No congestion control
- No adaptation to the congestion conditions of the network

Suppresses TCP flows
- In case of congestion TCP flows will back off while UDP will stay on the same rate
- Can be used as an attack method (UDP flooding attack)

18

Applications Utilizing UDP

Simple query protocols like Domain Name System
- Delay for connection establishment is too large
- Queries are small and UDP adds a small overhead (header)
- Easier to have application retransmit if needed
- Usually may fit within a UDP packet so no out-of-order danger

Multimedia Applications
- Retransmitting lost/corrupted packets is not worthwhile
- By the time the packet is retransmitted, it's too late
- E.g., telephone calls, video conferencing, gaming
- Certain loss is acceptable since Voice, picture, etc are still discernable

19