Transport Level Protocols and UDP

Tom Kelliher, CS 325

Mar. 10, 2008

1 Administrivia

Announcements

PingClient due Wednesday.

Assignment

Read 3.4–3.5.

From Last Time

Exam.

Outline

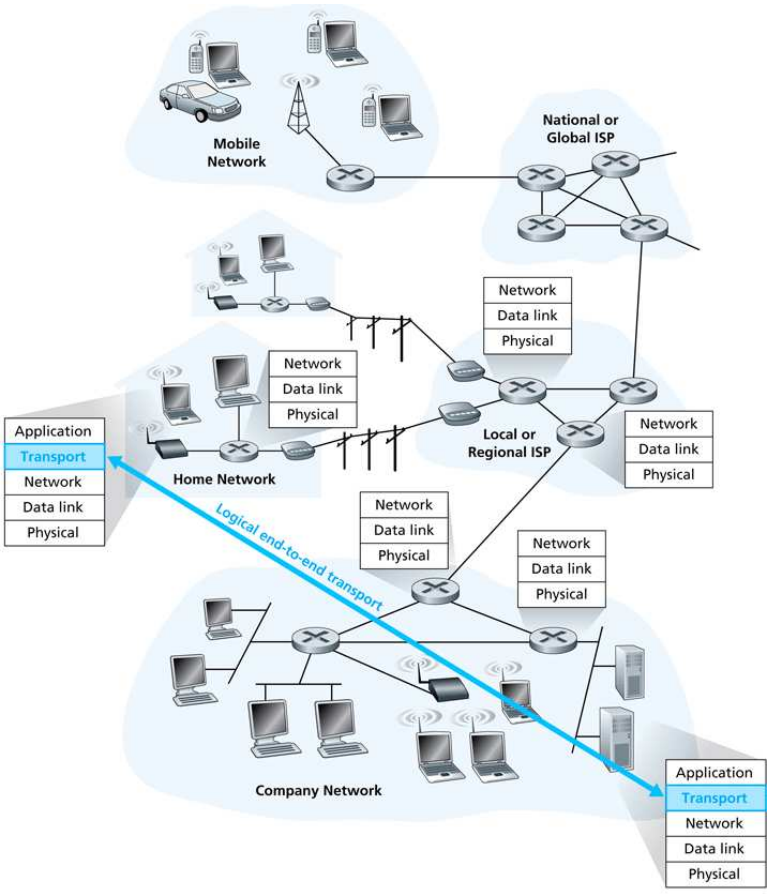
1. Transport level protocols introduction.
2. Multiplexing and demultiplexing.
3. UDP.

Coming Up

Reliable transport services and TCP.

2 Transport Level Protocols Introduction

Transport level protocols provide the illusion of a direct connection between hosts:



Terminology:

1. Segment: TCP.
2. Datagram: UDP and IP.

   IP: network-level protocol.

Things to note:

1. Network-level protocols provide host-to-host communications.

   USPS analogy — to your *house*.

   Only needs IP address.

2. Transport-level protocols provide process-to-process communications.

   USPS — To *you*.

3. IP is a best-effort delivery service. No guarantees: dropped, duplicated, mangled datagrams.

4. UDP adds process delivery and data error detection.

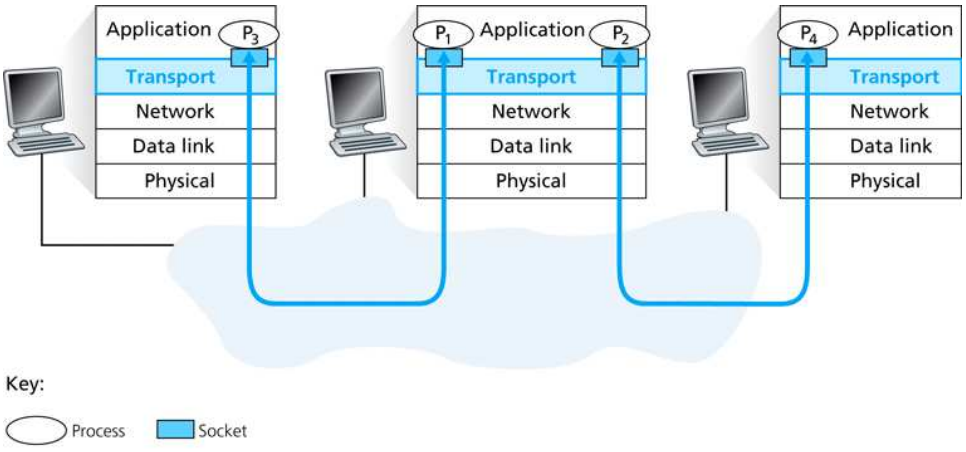
5. TCP adds reliability and congestion control.

3 Multiplexing and Demultiplexing

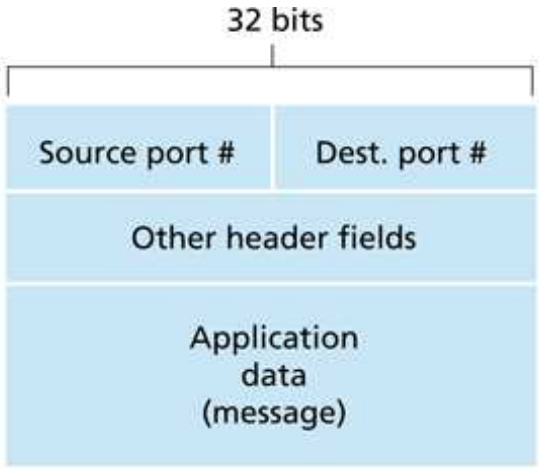
Multiplexing: Taking application data and adding necessary header information to ensure delivery to appropriate process on destination host. Multiple application streams (possibly to same destination host) *multiplexed* into network-level protocol.

Demultiplexing: Using header information to pass application data to correct process.

Illustration:



Use of port numbers to identify sockets:



1. Source port is also part of the "return" address.

2. Servers listen on "well known" port numbers. Clients use "any old" port number.

3. A process creates a socket; the OS then associates I/O on that socket to that process.

4. Only one process can open a socket on a given port, but...

   multiple TCP sockets (processes) might be using the same socket — the "main" process and forked children.

5. A UDP socket is identified by an ordered pair: (destination IP address, destination port).

   A single process receives all datagrams sent to a given port.

6. A TCP socket is identified by a four-tuple: (source IP address, source port, destination IP address, destination port).

   Multiple processes can receive segments sent to a given port.

   Example: Two SSH clients (one for Christine, one for Joe) on kingfisher (10.67.1.43) connecting to the SSH server (port 22) on phoenix (10.32.3.39).

7. Linux example of connection status (produced with `netstat -t -p -e -a`):

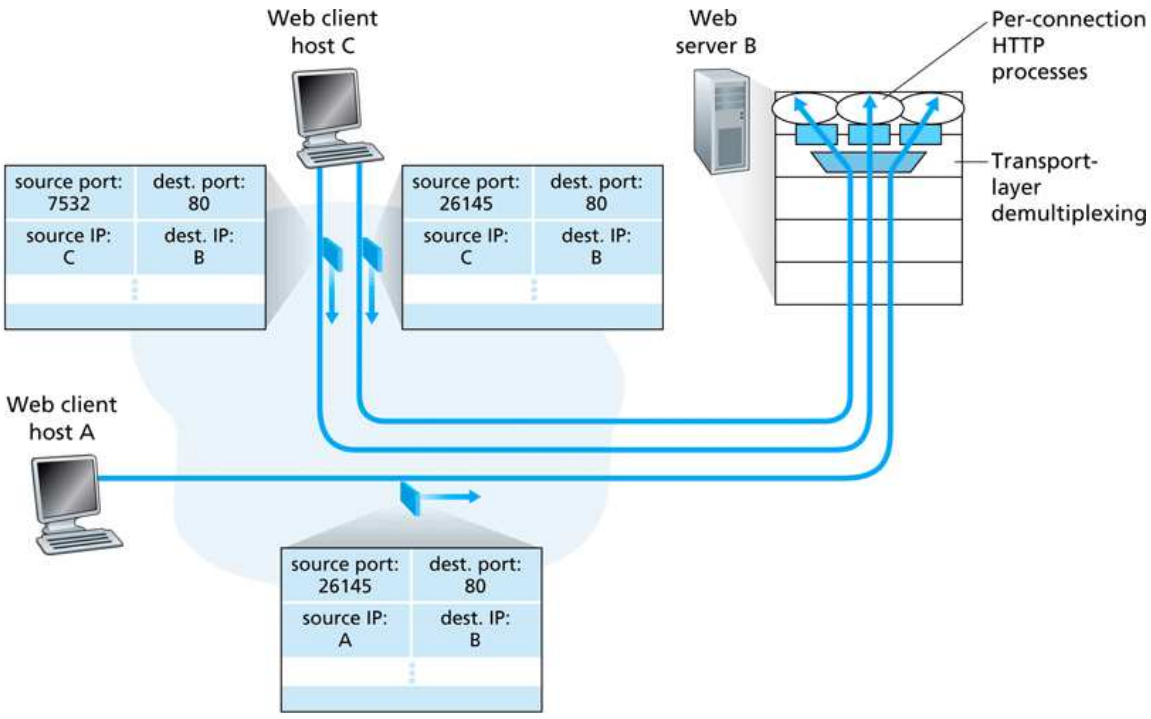
   ```
   Active Internet connections (servers and established)
   ```

```
Proto Recv-Q Send-Q Local Address               Foreign Address
State       User       Inode      PID/Program name

tcp        0      0 *:printer                   *:*
LISTEN      root       328119     16106/xinetd

tcp        0      0 *:5252                      *:*
LISTEN      smmsp      328461     16305/milter-greyli

tcp        0      0 *:netbios-ssn               *:*
LISTEN      root       328886     16491/smbd

tcp        0      0 localhost.localdomain:783   *:*
LISTEN      root       328561     16348/spamd.pid

tcp        0      0 *:sunrpc                    *:*
LISTEN      root       327576     15887/portmap

tcp        0      0 *:10000                     *:*
LISTEN      root       328048     16085/beremote

tcp        0      0 *:auth                      *:*
LISTEN      root       328118     16106/xinetd

tcp        0      0 phoenix.goucher.edu:domain  *:*
LISTEN      named      328000     16055/named

tcp        0      0 localhost.localdomai:domain *:*
LISTEN      named      327998     16055/named

tcp        0      0 *:ipp                       *:*
LISTEN      root       644033     25518/cupsd

tcp        0      0 *:postgres                  *:*
LISTEN      postgres   589356     19320/postmaster

tcp        0      0 *:824                       *:*
LISTEN      root       327612     15906/rpc.statd

tcp        0      0 *:smtp                      *:*
LISTEN      root       328522     16328/sendmail: acc

tcp        0      0 localhost.localdomain:rndc  *:*
LISTEN      named      328003     16055/named
```

```
tcp        0      0 localhost.lo:x11-ssh-offset *:*
LISTEN      kelliher   433569     32014/0

tcp        0      0 *:xfs                       *:*
LISTEN      xfs        328807     16482/xfs

tcp        0      0 *:microsoft-ds              *:*
LISTEN      root       328885     16491/smbd

tcp        0      0 phoenix.goucher.edu:5252    bluebird.goucher.edu:54880
ESTABLISHED smmsp      419692     16305/milter-greyli

tcp        0      0 phoenix.goucher.edu:39161   bluebird.goucher.edu:5252
ESTABLISHED smmsp      433791     16305/milter-greyli

tcp        0      0 *:5252                      *:*
LISTEN      smmsp      328460     16305/milter-greyli

tcp        0      0 *:http                      *:*
LISTEN      root       328689     16417/httpd

tcp        0      0 *:ssh                       *:*
LISTEN      root       328027     16077/sshd

tcp        0      0 *:postgres                  *:*
LISTEN      postgres   589355     19320/postmaster

tcp        0      0 localhost:x11-ssh-offset    *:*
LISTEN      kelliher   433570     32014/0

tcp        0      0 *:https                     *:*
LISTEN      root       328691     16417/httpd

tcp        0      0 *:xfs                       *:*
LISTEN      xfs        328806     16482/xfs

tcp        0      0 phoenix.goucher.edu:http    lj511775.crawl.yahoo.:36548
TIME_WAIT   root       0          -

tcp        0      0 phoenix.goucher.edu:ssh     bluebird.goucher.edu:46079
ESTABLISHED root       433527     32012/sshd: kellihe

tcp        0      0 phoenix.goucher.edu:http    crawl-8.cuill.com:38700
TIME_WAIT   root       0          -
```

```
tcp        0      0 phoenix.goucher.edu:http    crawl-8.cuill.com:49221
TIME_WAIT   root       0          -

tcp        0      0 phoenix.goucher.edu:http    ppp-69-218-215-238.dsl:1278
FIN_WAIT2   apache     660717     25594/httpd
```

Connection-oriented demultiplexing:

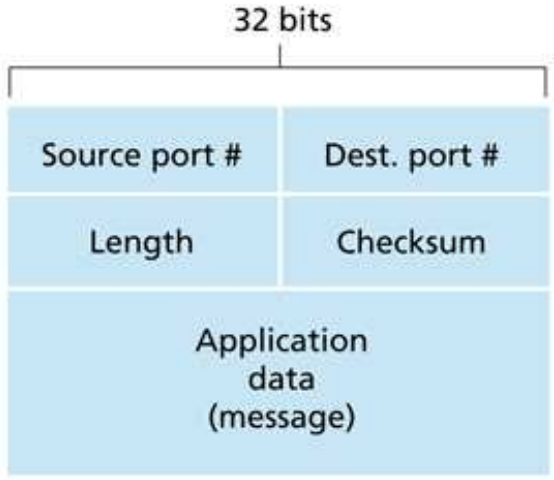


4 UDP

1. Little overhead.
2. No guarantees — application protocols must build in reliability.
3. Reliability not always needed — telephony.
4. Sometimes, getting through is more important: SNMP, DNS, RIP.
5. Sometimes, low overhead in a local networking environment matters most: NFS.

Application	Application-Layer Protocol	Underlying Transport Protocol
Electronic mail	SMTP	TCP
Remote terminal access	Telnet	TCP
Web	HTTP	TCP
File transfer	FTP	TCP
Remote file server	NFS	Typically UDP
Streaming multimedia	typically proprietary	UDP or TCP
Internet telephony	typically proprietary	UDP or TCP
Network management	SNMP	Typically UDP
Routing protocol	RIP	Typically UDP
Name translation	DNS	Typically UDP

UDP datagram format:



1. Header is eight bytes — low overhead.
2. Length is for entire datagram.
3. Checksum is computed over entire datagram and *pseudo-header*.
4. Pseudo-header fields:
   (a) Source and Destination IP address.

(b) IP protocol field.

Double-check that IP didn't route datagram to wrong host or transport-level protocol.

5. Importance of end-to-end checks: even if all links perform error checking, errors can and have been introduced in routers. Some link-level protocols (SLIP) do not perform error checking.

Open UDP sockets on phoenix:

```
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address               Foreign Address
State       User       Inode      PID/Program name

udp        0      0 *:who                       *:*
root       328156     16132/rwhod

udp        0      0 *:syslog                    *:*
root       327536     15859/syslogd

udp        0      0 phoenix.gouc:netbios-ns     *:*
root       328833     16495/nmbd

udp        0      0 *:netbios-ns                *:*
root       328828     16495/nmbd

udp        0      0 phoenix.gou:netbios-dgm     *:*
root       328834     16495/nmbd

udp        0      0 *:netbios-dgm               *:*
root       328829     16495/nmbd

udp        0      0 *:xdmcp                     *:*
root       329172     16578/gdm-binary

udp        0      0 *:818                       *:*
root       327602     15906/rpc.statd

udp        0      0 phoenix.goucher.edu:domain  *:*
named      327999     16055/named

udp        0      0 localhost.locald:domain     *:*
named      327997     16055/named
```

```
udp        0      0 *:821                       *:*
root       327609     15906/rpc.statd

udp        0      0 *:38749                     *:*
named      328001     16055/named

udp        0      0 *:sunrpc                    *:*
root       327575     15887/portmap

udp        0      0 localhost.localdomain:44528 localhost.localdomain:44528
ESTABLISHED postgres   589363     19320/postmaster

udp        0      0 *:ipp                       *:*
root       644034     25518/cupsd

udp        0      0 phoenix.goucher.edu:ntp     *:*
root       328166     16123/ntpd

udp        0      0 localhost.localdomain:ntp   *:*
root       328165     16123/ntpd

udp        0      0 *:ntp                       *:*
root       328163     16123/ntpd

udp        0      0 *:38750                     *:*
named      328002     16055/named

udp        0      0 *:ntp                       *:*
root       328164     16123/ntpd
```