EEC-484/584
Computer Networks

Lecture 19

Wenbing Zhao

wenbing@ieee.org

(Lecture nodes are based on materials supplied by
Dr. Louise Moser at UCSB and Prentice-Hall)

Outline

- TCP slow start and congestion avoidance
- The application layer
  - Domain name systems
  - Electronic mail

Internet Congestion Control Algorithm

- Three parameters
  - Receiver's window size
  - Sender's congestion window size
  - Threshold, initially 64KB
- When timeout occurs,
  - Threshold is set to half **current** congestion window size
    - To be accurate, threshold should be set to half of the amount of outstanding data in the network for this connection
  - Congestion window size is set to one max segment size
- Use slow start algorithm but stop when threshold is reached, then increase congestion window size linearly (congestion avoidance)

Slow Start Algorithm

- When connection is established, sender initializes congestion window size (*cwnd*) to size of max segment (*mss*) in use on connection, it then sends one max segment
- If segment is acked before timer goes off, it increases congestion window size by max segment size (*mss*), it then sends two segments
- For each ACK received that acknowledges new data, *cwnd* is increased by one *mss* until timeout occurs or receiver's window size is reached
  - Doubling each *RTT* => exponential growth in *cwnd* during slow start

Congestion Avoidance

- When threshold is reached (typically when *cwnd* > *threshold*), *cwnd* is increased by one *mss* for every *RTT*
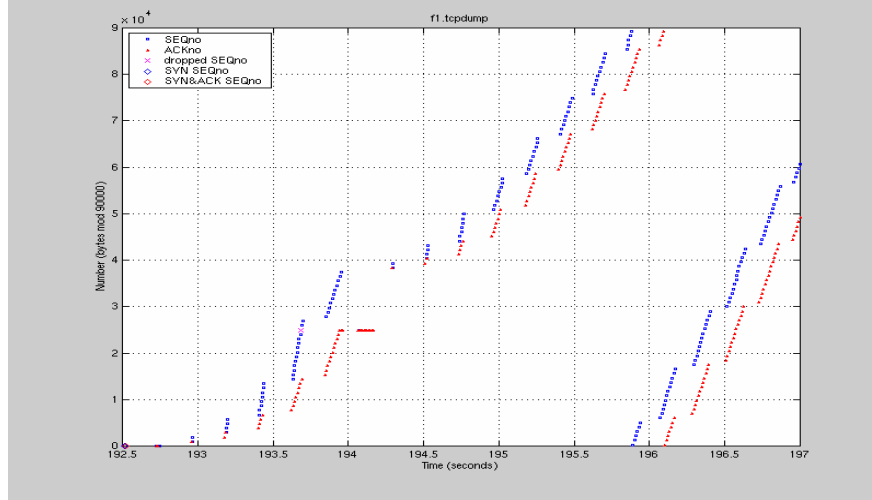- A practical implementation uses the following formula to update cwnd:

  cwnd (new) = *cwnd* + *mss*mss/cwnd*

- For more information on slow start and congestion avoidance, see the TCP congestion control RFC 2581: http://www.ietf.org/rfc/rfc2581.txt

TCP Congestion Control
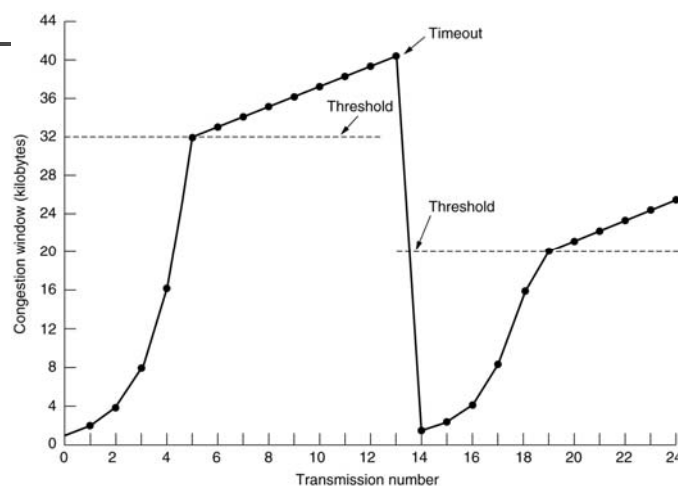
An example of the Internet congestion algorithm

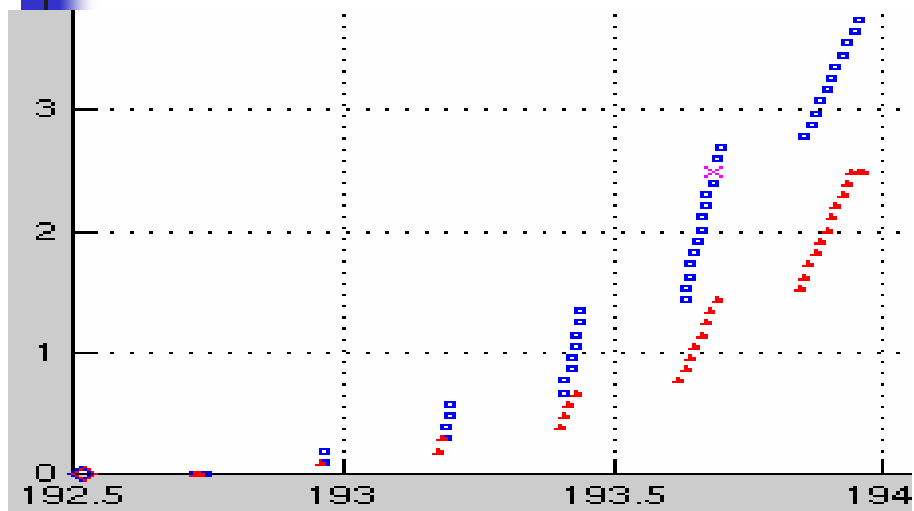# TCP Congestion Control

20 November 2005   EEC484/584   Wenbing Zhao

TCP Congestion Control

20 November 2005   EEC484/584   Wenbing Zhao

Exercise Problem #3

- Suppose that the TCP congestion window is set to 18 KB and a timeout occurs. How big will the window be if the next four transmission bursts are all successful? Assume that the maximum segment size is 1 KB.

Exercise Problem #3

Segment#	Action			Variables	
	Send	Receive	Comment	cwnd	ssthresh
			Timeout	18KB	-
			Retransmit	1024	9216
1	1:1025(1024)			1024	9216
2		ACK 1025	Slow start	2048	9216
3	1025:2049(1024)			2048	9216
4	2049:3073(1024)			2048	9216
5		ACK 2049	Slow start	3072	9216
6		ACK 3073	Slow start	4096	9216
7	3073:4097(1024)			4096	9216
8	4097:5121(1024)			4096	9216
9	5121:6145(1024)			4096	9216
10	6145:7169(1024)			4096	9216
11		ACK 4097	Slow start	5120	9216
12		ACK 5121	Slow start	6144	9216
13		ACK 6145	Slow start	7168	9216
14		ACK 7169	Slow start	8192	9216

Exercise Problem #3

15	7169:8193(1024)			8192	9216
16	8193:9217(1024)			8192	9216
17	9217:10241(1024)			8192	9216
18	10241:11265(1024)			8192	9216
19	11265:13313(1024)			8192	9216
20	12289:14337(1024)			8192	9216
21	14337:15361(1024)			8192	9216
22	15361:16385(1024)			8192	9216
23		ACK 8193	Slow start	9216	9216
24		ACK 9217	Slow start	10240	9216
25		ACK 10241	Cong. Avoid. New cwnd = cwnd + mss*mss/cwnd	10342	9216
26		ACK 11265	Cong. Avoid.	10443	9216
27		ACK 13313	Cong. Avoid.	10543	9216
28		ACK 14337	Cong. Avoid.	10642	9216
29		ACK 15361	Cong. Avoid.	10740	9216
30		ACK 16385	Cong. Avoid.	10837	9216

Correction to HW3 Problem 1
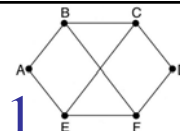
- Thanks Okechukwu Mezu for pointing this out!

- Going via *B* gives (**11**, **6**, 14, 18, 12, **8**).
- Going via *D* gives (19, 15, 9, **3**, 12, 13).
- Going via *E* gives (12, 11, 8, 14, **5**, 9).
- Taking the minimum for each destination except *C* gives (11, 6, 0, 3, 5, 8).
- The outgoing lines are (*B*, *B*, –, *D*, *E*, *B*).

DNS – The Domain Name System

- Hierarchical domain-based naming scheme and distributed database system for implementing it

- Maps symbolic names (ASCII strings) to network addresses, i.e., maps hostnames and email addresses to IP addresses

- Application program calls library procedure
  - Resolver whose input is name, output is IP address
  - Resolver sends query to local DNS server (both TCP and UDP can be used)
  - Local DNS server looks up name, returns IP address to resolver
  - Resolver returns IP address to caller

DNS – The Domain Name System

- **The DNS Name Space**
- **Resource Records**
- **Name Servers**

The DNS Name Space

- Internet divided into several hundred top-level domains

- Domains correspond to organizational boundaries, not physical networks

- Domains partitioned into subdomains, etc.

- To create new domain, need permission from manager above in tree

20 November 2005   EEC484/584   Wenbing Zhao

The DNS Name Space

- Generic names:
  - com – commercial   edu – education
  - gov – US government   int – certain international organizations
  - mil – US military   net – network providers
  - org – nonprofit organizations   biz, info, coop, pro, travel, …

20 November 2005   EEC484/584   Wenbing Zhao
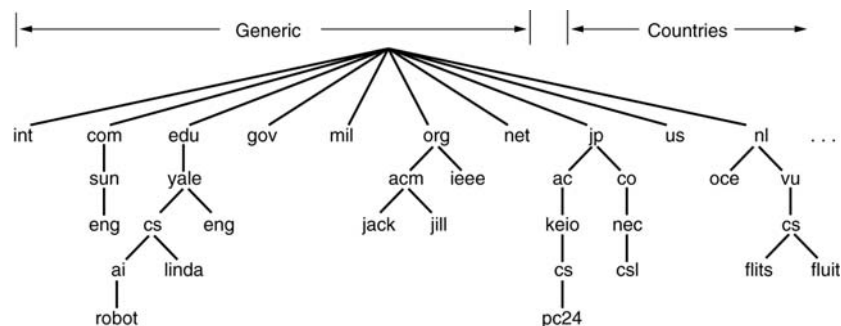
The DNS Name Space

- Each domain is named by the path upward from it to the unnamed root. The components are separated by period
  - E.g., eng.sun.com.
- Domain names can be absolute (end with period), or relative
- Domain names are case insentive
- Component names <= 63 chars
- Full path names <= 255 chars

Resource Records

- Every domain has set of resource records associated with it
- Resolver returns set of resource record
- Record:
  - Domain-name
  - Time_to_live – how stable (1 day, 1 min, etc.)
  - Type – what kind of record
  - Class – IN for Internet
  - Value – number, domain name, ASCII string

Resource Records

Type	Meaning	Value
SOA	Start of Authority	Parameters for this zone
A	IP address of a host	32-Bit integer
MX	Mail exchange	Priority, domain willing to accept e-mail
NS	Name Server	Name of a server for this domain
CNAME	Canonical name	Domain name
PTR	Pointer	Alias for an IP address
HINFO	Host description	CPU and OS in ASCII
TXT	Text	Uninterpreted ASCII text

- The principal DNS resource records types
  - CNAME: allow aliases to be created
  - PTR: primarily used to associate a domain name with an IP address to allow reverse lookups (IP => domain name)

20 November 2005   EEC484/584   Wenbing Zhao

Resource Records

```
; Authoritative data for cs.vu.nl
cs.vu.nl.        86400  IN  SOA    star boss (952771,7200,7200,2419200,86400)
cs.vu.nl.        86400  IN  TXT    "Divisie Wiskunde en Informatica."
cs.vu.nl.        86400  IN  TXT    "Vrije Universiteit Amsterdam."
cs.vu.nl.        86400  IN  MX     1 zephyr.cs.vu.nl.
cs.vu.nl.        86400  IN  MX     2 top.cs.vu.nl.

flits.cs.vu.nl.  86400  IN  HINFO  Sun Unix
flits.cs.vu.nl.  86400  IN  A      130.37.16.112
flits.cs.vu.nl.  86400  IN  A      192.31.231.165
flits.cs.vu.nl.  86400  IN  MX     1 flits.cs.vu.nl.
flits.cs.vu.nl.  86400  IN  MX     2 zephyr.cs.vu.nl.
flits.cs.vu.nl.  86400  IN  MX     3 top.cs.vu.nl.
www.cs.vu.nl.    86400  IN  CNAME  star.cs.vu.nl
ftp.cs.vu.nl.    86400  IN  CNAME  zephyr.cs.vu.nl

rowboat                 IN  A      130.37.56.201
                        IN  MX     1 rowboat
                        IN  MX     2 zephyr
                        IN  HINFO  Sun Unix

little-sister           IN  A      130.37.62.23
                        IN  HINFO  Mac MacOS

laserjet                IN  A      192.31.231.216
                        IN  HINFO  "HP Laserjet IIISi" Proprietary
```
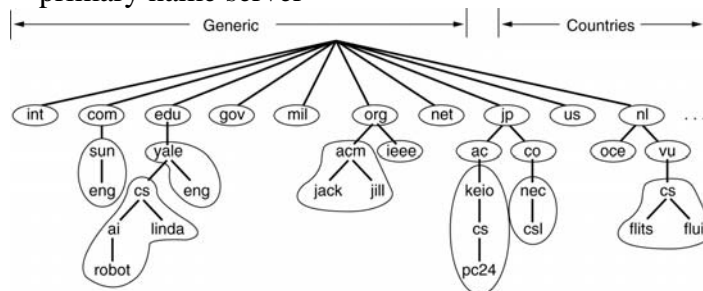
A portion of a possible DNS database for *cs.vu.nl*

Name Servers

- DNS name space divided into non-overlapping zones

- Each zone contains part of tree and name servers holding info about zone
  - One primary name server gets its info off disk
 - One or more non-primary name servers get info from primary name server

---

Name Servers

- Root servers maintain the IP addresses of all the top-level servers
  - There are about a dozen root servers spread around the world

- The IP addresses of the root servers are stored in a system configuration file. They are loaded into the DNS cache when the DNS server is booted

- A name servers knows the IP addresses of all its children

Recursive Query

- When resolver has query about domain names, it passes query to one of local name servers
- If domain is in jurisdiction of name server, it returns the authoritative resource record
- If domain is remote, name server sends query message to top-level name server for domain requested using recursive query
- The records returned might be cached along the reverse path
  - A cached record might be out-of-dated and will be timed out
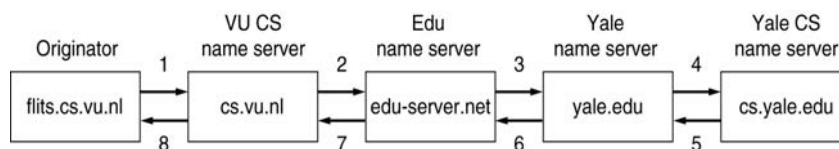
20 November 2005   EEC484/584   Wenbing Zhao

Electronic Mail

- **Architecture and Services**
- **The User Agent**
- **Message Formats**
- **Message Transfer**
- **Final Delivery**

20 November 2005   EEC484/584   Wenbing Zhao

Architecture and Services

- User agents – allow users to read and send email.
  - They are local programs that provide command-based, menu-based, or graphics for interacting with email system
  - User agent builds message, passes it to message transfer agent, which uses header fields to construct envelop
- Message transfer agents – move messages from source to destination
  - They are system daemons (software that runs in background) that move email through systems

Architecture and Services

- Five basic functions
  - Composition – creating and replying to messages
  - Transfer – moving messages from source to destination
    - Establish connection to destination
    - Output the message
    - Release connection
  - Reporting – telling source what happened to messages, e.g., sent, rejected, etc.
  - Displaying incoming messages so user can read them
  - Disposition – what destination does with message after receiving, e.g., throw away, save, etc.

Architecture and Services

- Most systems allow users to create **mailboxes** to store incoming email
- A key idea in email systems is the distinction between the **envelope** and its contents
  - The **envelope** encapsulates the message. It contains all the information needed for transporting the message, such as destination address, priority, and security level
  - The transport agents use the envelope for routing

Architecture and Services

Paper mail   Electronic mail

Sending Email

- To send an email message, a user must provide the message, the destination address, and possibly some other parameters

- Email address typically has the form user@dns-address

- Other email addresses are possible, such as X.400 addresses:
  - /C=US/ST=OHIO/L=CLEVELAND/PA=2121 EULICD AVE/CN=W ZHAO

Reading Email

An example display of the contents of a mailbox

#	Flags	Bytes	Sender	Subject
1	K	1030	asw	Changes to MINIX
2	KA	6348	trudy	Not all Trudys are nasty
3	K F	4519	Amy N. Wong	Request for information
4		1236	bal	Bioinformatics
5		104110	kaashoek	Material on peer-to-peer
6		1223	Frank	Re: Will you review a grant proposal
7		3110	guido	Our paper has been accepted
8		1204	dmr	Re: My student's visit

Message Formats – RFC 822

Header	Meaning
To:	E-mail address(es) of primary recipient(s)
Cc:	E-mail address(es) of secondary recipient(s)
Bcc:	E-mail address(es) for blind carbon copies
From:	Person or people who created the message
Sender:	E-mail address of the actual sender
Received:	Line added by each transfer agent along the route
Return-Path:	Can be used to identify a path back to the sender

RFC 822 header fields related to message transport

Message Formats – RFC 822

Header	Meaning
Date:	The date and time the message was sent
Reply-To:	E-mail address to which replies should be sent
Message-Id:	Unique number for referencing this message later
In-Reply-To:	Message-Id of the message to which this is a reply
References:	Other relevant Message-Ids
Keywords:	User-chosen keywords
Subject:	Short summary of the message for the one-line display

Some header fields used by the user agents or human recipients

MIME –
Multipurpose Internet Mail Extensions

- Problems with international languages:
  - Languages with accents (French, German)
  - Languages in non-Latin alphabets (Hebrew, Russian)
  - Languages without alphabets (Chinese, Japanese)
  - Messages not containing text at all (audio or images)
- Solutions
  - Add structure to message body
  - Define encoding rules for non-ASCII messages

MIME – Encoding Binary Messages

- **Base64 encoding** or **ASCII armor:**
  - Groups of 24 bits are broken up into four 6-bit units, with each unit being sent as a legal ASCII character
  - The coding is "A" for 0, "B" for 1, and so on, followed by the 26 lower-case letters, the ten digits, and finally + and / for 62 and 63, respectively
  - The == and = sequences indicate that the last group contained only 8 or 16 bits, respectively
  - Carriage returns and line feeds are ignored,
- **Quoted-printable encoding:** For messages that are almost entirely ASCII but with a few non-ASCII characters
  - This is just 7-bit ASCII, with all the characters above 127 encoded as an equal sign followed by the character's value as two hexadecimal digits

MIME

- RFC 822 headers added by MIME

Header	Meaning
MIME-Version:	Identifies the MIME version
Content-Description:	Human-readable string telling what is in the message
Content-Id:	Unique identifier
Content-Transfer-Encoding:	How the body is wrapped for transmission
Content-Type:	Type and format of the content

MIME

- The MIME types and subtypes defined in RFC 2045

Type	Subtype	Description
Text	Plain	Unformatted text
	Enriched	Text including simple formatting commands
Image	Gif	Still picture in GIF format
	Jpeg	Still picture in JPEG format
Audio	Basic	Audible sound
Video	Mpeg	Movie in MPEG format
Application	Octet-stream	An uninterpreted byte sequence
	Postscript	A printable document in PostScript
Message	Rfc822	A MIME RFC 822 message
	Partial	Message has been split for transmission
	External-body	Message itself must be fetched over the net
Multipart	Mixed	Independent parts in the specified order
	Alternative	Same message in different formats
	Parallel	Parts must be viewed simultaneously
	Digest	Each part is a complete RFC 822 message

MIME

From: elinor@abcd.com
To: carolyn@xyz.com
MIME-Version: 1.0
Message-Id: <0704760941.AA00747@abcd.com>
Content-Type: multipart/alternative; boundary=qwertyuiopasdfghjklzxcvbnm
Subject: Earth orbits sun integral number of times

This is the preamble. The user agent ignores it. Have a nice day.

--qwertyuiopasdfghjklzxcvbnm
Content-Type: text/enriched

Happy birthday to you
Happy birthday to you
Happy birthday dear <bold> Carolyn </bold>
Happy birthday to you

--qwertyuiopasdfghjklzxcvbnm
Content-Type: message/external-body;
        access-type="anon-ftp";
        site="bicycle.abcd.com";
        directory="pub";
        name="birthday.snd"

content-type: audio/basic
content-transfer-encoding: base64
--qwertyuiopasdfghjklzxcvbnm--

- A multipart message containing enriched and audio alternatives

---

Message Transfer System

- Establishes transport connection from source machine to destination machine and transfers the message
- SMTP – Simple Mail Transfer Protocol

SMTP

- Source machine (client) establishes TCP connection to port 25 of destination machine
    - Email daemon listens to this port, accepts incoming connections, copies messages from them into appropriate mailboxes
    - Error report returned to sender if message can't be delivered
    - Source machine waits for destination machine to reply
- Destination machine (server) sends line of text, giving its identity and telling whether or not it can receive mail
- If not, client releases connection and tries again later. If so, client announces users at source and destination machines. If user exists at destination machine, server tells client to send message
- Client sends, server acks
- Connection is released

20 November 2005   EEC484/584   Wenbing Zhao

Message Transfer

```
                  S: 220 xyz.com SMTP service ready
C: HELO abcd.com
                  S: 250 xyz.com says hello to abcd.com
C: MAIL FROM: <elinor@abcd.com>
                  S: 250 sender ok
C: RCPT TO: <carolyn@xyz.com>
                  S: 250 recipient ok
C: DATA
                  S: 354 Send mail; end with "." on a line by itself
C: From: elinor@abcd.com
C: To: carolyn@xyz.com
C: MIME-Version: 1.0
C: Message-Id: <0704760941.AA00747@abcd.com>
C: Content-Type: multipart/alternative; boundary=qwertyuiopasdfghjklzxcvbnm
C: Subject: Earth orbits sun integral number of times
C:
C: This is the preamble. The user agent ignores it. Have a nice day.
C:
C: --qwertyuiopasdfghjklzxcvbnm
C: Content-Type: text/enriched
C:
C: Happy birthday to you
C: Happy birthday to you
C: Happy birthday dear <bold> Carolyn </bold>
C: Happy birthday to you
C:
C: --qwertyuiopasdfghjklzxcvbnm
C: Content-Type: message/external-body;
C:      access-type="anon-ftp";
C:      site="bicycle.abcd.com";
C:      directory="pub";
C:      name="birthday.snd"
C:
C: content-type: audio/basic
C: content-transfer-encoding: base64
C: --qwertyuiopasdfghjklzxcvbnm
C: .
                  S: 250 message accepted
C: QUIT
                  S: 221 xyz.com closing connection
```
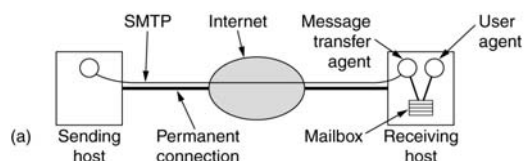
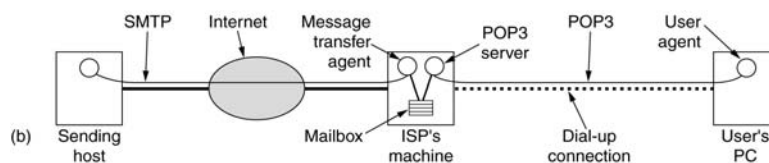- Transferring a message from *elinore@abc.com* to *carolyn@xyz.com*

Email Gateways

- Required when two machines use different transport protocols or different message formats
  - E.g., SMTP vs. X.400
- Gateway does conversion

Final Delivery

(a) Sending host — Permanent connection — SMTP — Internet — Message transfer agent — Mailbox — Receiving host — User agent

Sending and reading mail when the receiver has a permanent Internet connection and the user agent runs on the same machine as the message transfer agent

(b) Sending host — SMTP — Internet — Message transfer agent — Mailbox — ISP's machine — POP3 server — POP3 — Dial-up connection — User agent — User's PC

Reading e-mail when the receiver has a dial-up connection to an ISP

POP3 – Post Office Protocol

■ Fetch email from remote mailbox, store in user's local machine to be read later

  ➢ Example: using POP3 to fetch three messages

```
                              S: +OK POP3 server ready
          C: USER carolyn
                              S: +OK
          C: PASS vegetables
                              S: +OK login successful
          C: LIST
                              S: 1 2505
                              S: 2 14302
                              S: 3 8122
                              S: .
          C: RETR 1
                              S: (sends message 1)
          C: DELE 1
          C: RETR 2
                              S: (sends message 2)
          C: DELE 2
          C: RETR 3
                              S: (sends message 3)
          C: DELE 3
          C: QUIT
                              S: +OK POP3 server disconnecting
```

IMAP –
Interactive Mail Access Protocol

■ For users who use multiple machines

■ Email server maintains central repository that can be accessed from any machine

■ Unlike POP3, IMAP does not copy email to user's machines

A Comparison of POP3 and IMAP

Feature	POP3	IMAP
Where is protocol defined?	RFC 1939	RFC 2060
Which TCP port is used?	110	143
Where is e-mail stored?	User's PC	Server
Where is e-mail read?	Off-line	On-line
Connect time required?	Little	Much
Use of server resources?	Minimal	Extensive
Multiple mailboxes?	No	Yes
Who backs up mailboxes?	User	ISP
Good for mobile users?	No	Yes
User control over downloading?	Little	Great
Partial message downloads?	No	Yes
Are disk quotas a problem?	No	Could be in time
Simple to implement?	Yes	No
Widespread support?	Yes	Growing

20 November 2005   EEC484/584   Wenbing Zhao

Email Spoofing

- Email spoofing: the forgery of an e-mail header so that the message appears to have originated from someone or somewhere other than the actual source

- E-mail spoofing is possible because Simple Mail Transfer Protocol does not include an authentication mechanism
  - To send spoofed e-mail, senders insert commands in headers that will alter message information.
  - It is possible to send a message that appears to be from anyone, anywhere, saying whatever the sender wants it to say

20 November 2005   EEC484/584   Wenbing Zhao

Open Replay Problem

- An open relay is an SMTP email server that allows third-party relay of e-mail messages.

- By processing mail that is neither for nor from a local user, an open relay makes it possible for an unscrupulous sender to route large volumes of spam

Relaying cases
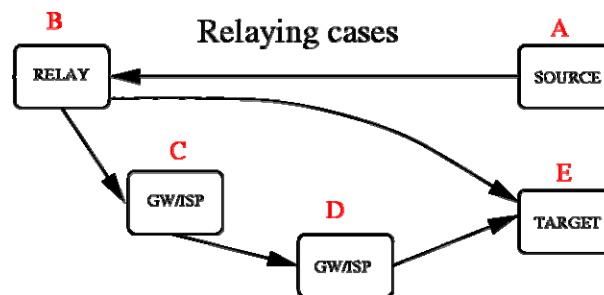
Relayed Mail Scenario

- Step 1 -- Source Host to Relay Host
  - R: 220 USC-ISIE.ARPA Simple Mail Transfer Service Ready
  - S: HELO MIT-AI.ARPA
  - R: 250 USC-ISIE.ARPA
  - S: MAIL FROM:<JQP@MIT-AI.ARPA>
  - R: 250 OK
  - S: RCPT TO:<@USC-ISIE.ARPA:Jones@BBN-VAX.ARPA>
  - R: 250 OK
  - S: DATA
  - R: 354 Start mail input; end with <CRLF>.<CRLF>
  - S: Date: 2 Nov 81 22:33:44
  - S: From: John Q. Public <JQP@MIT-AI.ARPA>
  - S: Subject: The Next Meeting of the Board
  - S: To: Jones@BBN-Vax.ARPA
  - S:
  - S: Bill:
  - S: The next meeting of the board of directors will be on Tuesday. John.
  - S: .
  - R: 250 OK
  - S: QUIT
  - R: 221 USC-ISIE.ARPA Service closing transmission channel

24

Relayed Mail Scenario

- Step 2 -- Relay Host to Destination Host
  - R: 220 BBN-VAX.ARPA Simple Mail Transfer Service Ready
  - S: HELO USC-ISIE.ARPA
  - R: 250 BBN-VAX.ARPA
  - S: MAIL FROM:<@USC-ISIE.ARPA:JQP@MIT-AI.ARPA>
  - R: 250 OK
  - S: RCPT TO:<Jones@BBN-VAX.ARPA>
  - R: 250 OK
  - S: DATA
  - R: 354 Start mail input; end with <CRLF>.<CRLF>
  - S: Received: from MIT-AI.ARPA by USC-ISIE.ARPA ; 2 Nov 81 22:40:10 UT
  - S: Date: 2 Nov 81 22:33:44
  - S: From: John Q. Public <JQP@MIT-AI.ARPA>
  - S: Subject: The Next Meeting of the Board
  - S: To: Jones@BBN-Vax.ARPA
  - S:
  - S: Bill:
  - S: The next meeting of the board of directors will be on Tuesday. John.
  - S: .
  - R: 250 OK
  - S: QUIT
  - R: 221 USC-ISIE.ARPA Service closing transmission channel