

**Electronic Mail**

INFO 341

---

---

---

---

---

---

---

---

**Objectives**

- What are email system components?
- What is SMTP? What port does it run on?
- What are the components of a proper email address? What RFC would you use to find this answer?
- What is one problem of SMTP?
- What is IMAP & POP? What port do they run on?
- What is one difference between POP & IMAP?
- What are two ways of transferring sending a binary file as an attachment?

---

---

---

---

---

---

---

---

**Electronic Mail**

- Most successful Internet application
- Used to transmit messages from one person to another or from one person to many people
  - Initially these were text only messages
  - Today these messages make include pictures, advanced text markup (like different fonts and colors) or file attachments (like Word or Excel files)
- On the Internet, most email systems are built on a client, server, protocol architecture and follow Internet standards
- Some organizations use "proprietary" email systems that incorporate additional features like calendaring or workflow, we'll talk about them a bit at the end

---

---

---

---

---

---

---

---

## Email Clients, Servers, Protocols

- The client is the email program for the end user
  - pine, elm, mail, xmail on Unix
  - Eudora, PC Pine, Netscape, Outlook, Outlook Express
- In the case of email the 'system' is a collection of servers and protocols. Each is responsible for a different part of the process.
- In general, the mail sending and receiving function is handled by SMTP (Simple Mail Transfer Protocol)
- Another service and protocol is used to retrieve messages from the user's mailbox so they can be displayed, deleted, etc. Those protocols are POP (Post Office Protocol) and IMAP (Internet Mail Access Protocol)

---

---

---

---

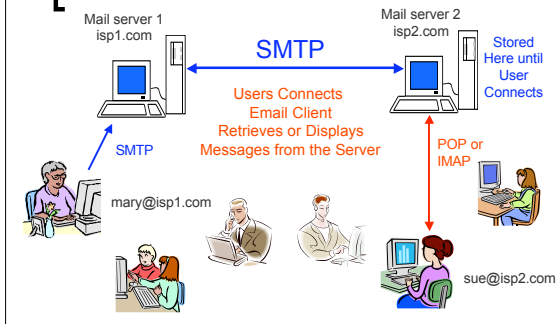
---

---

---

---

## Common Email delivery scenario



---

---

---

---

---

---

---

---

## Mail Delivery

- There are variations on the scenario and we will discuss some of them later
- First let's look closer at the pieces

---

---

---

---

---

---

---

---

## The Email Client

- This is the software the user uses to send and receive email. (Pine, Outlook Express, etc.)
- Gives you your user-interface and feature set
- Clients may be Unix based, PC or Mac based, dumb-terminal/character applications, GUI programs, or Web based
- In order to send mail, they all must implement SMTP

---

---

---

---

---

---

---

---

## Email Client Features

- Common
  - alias and alias-lists, (address books)
  - folders
  - forwarding, carbon copies (cc:/bcc:), reply
  - file inclusion, attachments
  - signature file
- Almost common
  - Security (PGP, S/MIME, possible technologies but not widely used)
  - Authentication (Digital certificates possible technologies but again not widely used)

---

---

---

---

---

---

---

---

## Email Addresses

- Internet standard (RFC 822, RFC 2822)
  - userid@host.domain
    - such as dwmc@u.washington.edu
  - Some variations are possible like:
    - David.McDonald@xyz.com
    - 12345.65362@xyz.com
- In the "old days" many people in education used something called "Bitnet"
  - Addresses looked like mcdonald@washington or
  - McDonald@washington.bitnet
  - Used a store and forward mechanism to deliver mail
- In the commercial world there were competing alternatives like MCI Mail, CompuServe — each had it's own address format and couldn't talk to the other!

---

---

---

---

---

---

---

---

## Finding Email Addresses

- Usually you just asked!
- Were some crude old tools like:
  - finger (UNIX)
  - Netfind
- Recently have appeared servers on the web that let you look for email addresses.
  - www.bigfoot.com, people.yahoo.com
  - Often support a protocol called LDAP (Lightweight Directory Access Protocol) that allows you to query them inside your mail client

---

---

---

---

---

---

---

---

## Email Clients - Unix Pine

- You can try text/command line Pine
- On Dante type 'pine' <enter>
  - "C" for compose a message
  - Fill in "To", "Subject"
  - Return takes you into message body
  - Ctrl-X to send it (see bottom of screen)

---

---

---

---

---

---

---

---

## Pine Supports

- Inbox
- Deleting Messages
- Using Folders
- Reply forward
- Address book
  
- All of these are common email client functions

---

---

---

---

---

---

---

---

## How about the original?

- Pine is a sophisticated email program for Unix (developed at UW)
- Before Pine many used the standard Unix mail program called "mail", on Dante there is a similar program called "mailx"
- Try it by typing:
  - mailx userid@host.domain
  - ctrl-D to end a message
- Notice how limited you are in editing.
- Obviously "mail" didn't have the best UI. Another popular Unix mail program called "elm" became popular at many sites.

---

---

---

---

---

---

---

---

## Delivering the message

- Once you have composed and "sent" an email message on a Unix based system, your message is usually placed in an outgoing mail queue folder
- A server daemon, checks that folder periodically and if it finds an outgoing message, it initiates an SMTP connection to the destination mail server
  - The most commonly used SMTP server software on Unix is probably "sendmail" although there are many other options
- If the destination machine is running and accepting SMTP connections, the message will be delivered immediately
- If the destination is not accepting SMTP incoming connections the message will be placed back into the outgoing mail folder and queued-up for another attempt later (usually 10-15 minutes)
- If the message cannot be delivered after some period of time (typically 3 to 5 days) it is returned to the sender as "undeliverable"

---

---

---

---

---

---

---

---

## SMTP

- Simple Mail Transfer Protocol
  - Originally defined (RFC 821) in 1982
  - A modified standard (RFC 2821) proposed in 2001
  - Defines the host to host mail transport and delivery protocol
- SMTP servers connect and listen on port 25
- SMTP servers may be the source or final destination
- SMTP servers may "relay" mail for other servers
  - Today mail relaying is often allowed in only limited situations. This prevents "spammers" from using other servers to distribute their email spam.

---

---

---

---

---

---

---

---

## [ SMTP Details ]

- SMTP is a text based protocol with commands like:

```
HELO (or EHLO in the newest version)
MAIL FROM:<user@host.domain>
RCPT TO:<user@host.domain>
DATA
QUIT
```

---

---

---

---

---

---

---

---

## [ Seeing the Protocol in Action ]

- The mailx program has a -v (for verbose) option that allows you to watch the SMTP protocol in action.
- Watch...

---

---

---

---

---

---

---

---

## [ More on SMTP ]

- The S in SMTP stands for SIMPLE
- SMTP needs add-ons for
  - security/privacy (PGP or S/MIME)
  - binary attachments (MIME)
- SMTP is only involved in the transfer of mail from one mail server to another. It has nothing to do with how you read the mail once it arrives.
- Other protocols retrieve mail from the server.
  - We'll discuss those later

---

---

---

---

---

---

---

---

### Where does mail go when it gets to the destination mail server?

- Remember that SMTP simply delivers mail. As it arrives a couple things have to happen.
  - First there is a check - does this mail need to be forwarded elsewhere? On Unix an `/etc/aliases` file allows an administrator to setup forwarding
  - On some systems the user can set up their own `.forward` file. In that the user enters the address they want to forward to.
  - One caution, if you forward, be careful not to create forwarding loops!

---

---

---

---

---

---

---

---

### If it isn't forwarded...

- Mail has to be placed somewhere so the user or the mail client can retrieve it later
- On Unix there is typically a special directory where incoming mail is stored. Often this is `/var/spool/mail/<login>`
- On PC based servers mail also usually just goes into a folder, or alternatively it may be added to a database for later retrieval.
- The precise delivery location on either platform depends on the server software that is running and how that mail server has been configured by the system administrator
- End users normally use a mail client to access the location through the file system, or use a protocol like to request messages from the server and display them to the user

---

---

---

---

---

---

---

---

### The entire mail delivery process

- User uses a mail *client* to compose
- That client turns the message over to a mail *server* that will try and deliver the mail
- The mail server uses the *SMTP protocol* to talk to another mail server at the destination
- The destination mail server also talks SMTP and it either accepts or forwards the mail elsewhere
- If mail stays at destination, the destination server stores the message waiting for the user to read it
- Destination user runs their mail client which checks their "inbox" for new mail, parses messages, and provides "nice" or "not so nice" UI.

---

---

---

---

---

---

---

---

## Back to the diagram...



Why do we need intermediary SMTP servers to deliver mail and additional protocols for retrieving and displaying messages from a mail server? Why not just connect directly to the destination machine to deliver the mail – our host/PC to their host/PC ?

---

---

---

---

---

---

---

---

## Mail exchange without an intermediary....

- Possible but....
  - What address do we use? Recall that many hosts have dynamic IP addresses.
  - What if the destination is not on or up?
  - What if the destination is not running an SMTP server that is "listening" for incoming connections?
  - How do we effectively queue messages up for later delivery since our own machine may be powered off on occasion?
  - What if we turn off our machine, how will we receive new mail?
- As a result of these and other issues, PC email clients are configured with an "SMTP Server" that will act as an intermediary for sending messages.

---

---

---

---

---

---

---

---

## PC Email Client Protocols

- How does the email get to the PC email client
- Back to our diagram...

---

---

---

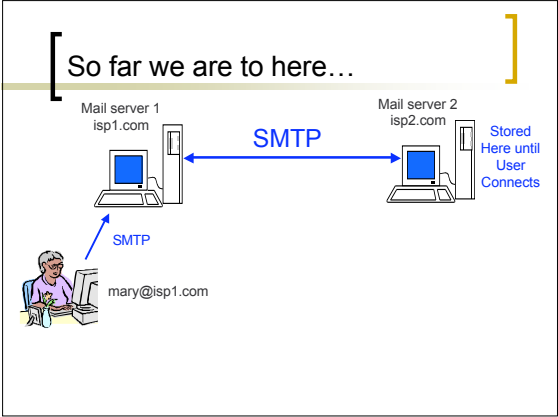
---

---

---

---

---




---

---

---

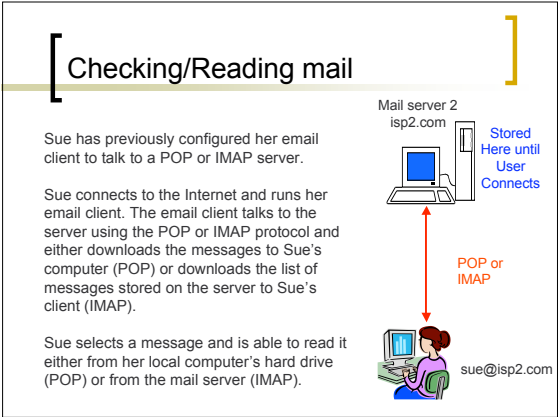
---

---

---

---

---




---

---

---

---

---

---

---

---

**[ POP and IMAP ]**

- POP - Post Office Protocol (RFC 1939, STD 0053)
- IMAP - Internet Message Access Protocol (RFC 2060, RFC 2061)
- Both provide a mechanism for a mail client to read or display email messages stored on a server

---

---

---

---

---

---

---

---

## More Specifics - POP and IMAP

- POP - Post Office Protocol (version 3 mostly widely used today, often called POP3).
  - Essentially store, copy, delete. Mail is stored on the server until the client connects and then is downloaded to the client. Mail is then usually deleted from the server.
  - Simple protocol and widely used. Many clients available including the "King of POP" - Eudora. Also widely supported by Outlook Express, Netscape Mail
  - Most ISP's suggest you use a POP client for mail. Good for them in that you connect and then the mail is removed from their server
  - Runs on port 110

---

---

---

---

---

---

---

---

## POP Protocol Commands

- Again a simple text based protocol
- Commands like:
  - USER username
  - PASS password (note this might be in clear text!)
  - STAT (to see how many messages)
  - LIST (to see the size of those messages)
  - RETR message# (retrieve that message)
  - DELE message# (delete that message)
  - QUIT

---

---

---

---

---

---

---

---

## Same deal as before...

- Like with SMTP you can telnet to the port to interact with the POP server directly
- However at UW the POP server does not support passwords in clear text
- So, here we need to configure a POP client and tell it a secure connection is required on a UW server.

---

---

---

---

---

---

---

---

## POP Weaknesses

- Bad for users that use multiple machines. Why?
- Can't selectively retrieve message or body parts. What if there is a big attachment and you have a SLOW connection?
- But...Since mail is downloaded to your machine, if you carry around a laptop computer everywhere you go or only use one computer, POP can be nice as you can read and compose messages while "offline"

---

---

---

---

---

---

---

---

## IMAP4 Internet Message Access Protocol

- Defined in RFC 2060, RFC 2061 and several other additions. Most work done right here at UW by Mark Crispin and others in CAC
- Folders and Messages can be stored on the server or in local folders on your machine.
- Much better for people using multiple machines than POP.
- Can selectively copy messages from the server to the local client based on many criteria.
- IMAP 4 has a mechanism to work "off-line" and then resynchronize the changes when you reconnect.
- IMAP servers use port 143

---

---

---

---

---

---

---

---

## IMAP Protocol Specifics

- MUCH more complex protocol to implement as a result not as many clients as POP
- Text based protocol like SMTP, and POP3 – possible to telnet to the port

Uses commands like:

```
01 LOGIN userid password
02 SELECT INBOX
03 FETCH 1:1 BODY[]
04 LOGOUT
```

Again password was in "clear text". Many (including UW) now require IMAP over SSL connections using port 993

---

---

---

---

---

---

---

---

## Extending Email System

- Calendaring/Scheduling
- Attachments
- HTML Email
- Web Clients

---

---

---

---

---

---

---

---

## Non "Standard" Email

- While SMTP, POP and IMAP are great standards, they are addressing just a portion of what people may want to do with email
  - Already noted the address book problem. But there is more..
  - What if you want to "recall" a message you already sent?
  - What if you want to do workflow through your email system?
  - What if you want to do shared calendaring and scheduling in email?
  - These protocols have no mechanisms in them for that type of functionality.

---

---

---

---

---

---

---

---

## "Proprietary" Servers/Clients

- There are many products available today that add these missing features. Most notable are:
  - Microsoft Exchange (they claim 100 million licenses sold to businesses)
  - Lotus Notes
  - Novell Groupwise
- Typically these products address the scheduling/calendaring problem and some do workflow
  - Unfortunately, no standard has emerged yet that everyone agrees on for calendaring, so when you use these products you are often limited to exchanging data with others that use the same product
- While these products are often considered "proprietary" they all still talk SMTP for sending/receiving mail, and most can act as a POP or IMAP server so you can use a "standard" client if you wish. You lose the added functionality if you use a "standard" client

---

---

---

---

---

---

---

---

## [ This is a problem... ]

- Clearly, from the user perspective the need for a calendaring/scheduling protocol is very strong but...
- Current market leaders are not so interested in a standard

---

---

---

---

---

---

---

---

## [ Email Attachments ]

- SMTP was originally designed just for the delivery of text messages
- Sometimes people want to send more just text messages
  - they want to send binary files (e.g. graphics, spreadsheets, music, video clips, or word processing files)
- The original version specification of SMTP did not allow this as it only deals with text. (Revisions have been proposed to the protocol to permit transmission of binary files but this change could take years before it is propagated everywhere)

---

---

---

---

---

---

---

---

## [ Sending Binary Files in Email ]

- We send binary attachments in email all the time, so what is happening to make it work?
- In the old days (7 years ago or so....)
  - Use a utility for converting binary to ascii. On Unix there is a program called uuencode that does this.
  - Mail the ascii file to the recipient. The recipient saves the file and runs uudecode to convert it back to binary.
- Today...
  - Use a MIME-compliant mailer

---

---

---

---

---

---

---

---

## [ uuencode uudecode ]

- Utilities to convert from ascii to binary and back
- Steps for using uuencode/uudecode on Dante
  - % uuencode demo.doc demo.encode
  - include demo.encode in your email ...
  - receiver saves emailed file as "attach.encode"
  - % uudecode attach.encode
- One other little issue skipped, was how the demo.doc file got to the Unix machine. How?

---

---

---

---

---

---

---

---

## [ MIME ]

- Multipurpose Internet Mail Extensions (RFC 1521, RFC 1522)
- MIME eliminates the need for uuencode/uudecode utilities
  - Email is "automatically" broken into parts and each part identified by its content type
    - E.g. Content-type: image/gif, Content-type: text/html
    - These same MIME types are used by HTTP servers to help describe the content of a web page to a web browser
  - When used with Email system, MIME compliant clients encode binary files into a text representation using base64 (different than uuencode)
  - MIME also allows email clients to supports multiple character sets
- MIME-compliant mail clients
  - Pine, Outlook Express, Netscape, Outlook, Eudora – but.....

---

---

---

---

---

---

---

---

## [ Using a MIME Mailer ]

- When using a MIME compliant mailer the process of sending binary attachments should be transparent to the user
  - No uuencode, decode
  - Just indicate attachment name and the client email program should take care of everything
- If using a Unix based, MIME compliant mailer and you want to send a PC file you need to get the file to the Unix machine first. How?

---

---

---

---

---

---

---

---

## Using HTML in email

- Many clients allow messages that include HTML
- Let the user use different fonts, colors, graphics etc.
- Is this a good idea?
- What do you think an HTML marked up message with an included graphic would look like in Unix Pine?

---

---

---

---

---

---

---

---

## Web Mail Front Ends

- Several services now offer a Web based email program
  - [www.hotmail.com](http://www.hotmail.com), [www.yahoo.com](http://www.yahoo.com)
- Advantages:
  - Great for people who are mobile, simple to use, if there is a Web browser you have a mail client
- Disadvantages:
  - Generally not as feature rich as stand-alone clients, you see ads, can be slow, generally not reliable
  - Why do many students insist on forwarding mail from a reliable service like UW to something as unreliable as HotMail?
- Of course UW has "WebPine" at <http://webpine.washington.edu>

---

---

---

---

---

---

---

---

## Summary

- Email related protocols
  - SMTP
  - IMAP
  - POP
  - MIME
- Clients, related tools, protocols
  - mail, pine, mailx
  - Eudora, Outlook, Netscape Mail, HotMail, YahooMail
  - uuencode, uudecode
  - LDAP

---

---

---

---

---

---

---

---