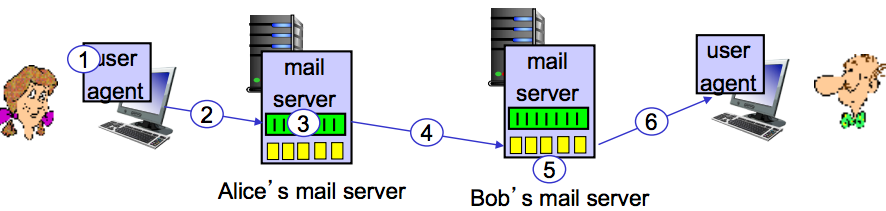
Video recording not available this week, only voice so notes are shortened.

The Electronic Mail system is composed of:

* **Mail User Agent (MUA)**
  + Program / client that allows you to send and receive email messages. E.g. Outlook, Gmail etc.
* **Mail Servers**
  + Mailbox: Contains incoming messages for the user
  + Message Queue of outgoing (to be sent) mail messages. SMTP Protocol in-between mail servers to send messages.
* **Simple Mail Transfer Protocol (SMTP)**
  + Uses TCP to reliably transfer email messages from client to server PORT 25
  + Direct transfer: sending server to receiving server
  + Three phases of transfer: **(1) handshaking (greeting),** **(2) transfer of messages**, **(3) closure**
  + Command / Response interaction (like HTTP, FTP)
    - Commands: ASCII text
    - Response: Status code and phrase
  + Messages must be in 7-bit ASCII

Email scenario

1. Alice uses MUA / UA to compose a message to Bob
2. Alice’s UA sends message to her mail server, mail is placed in a queue  
   (NOTE: Communication happens between user and mail server, UA’s don’t communicate directly)
3. Client side of SMTP opens TCP connection with Bob’s mail server
4. SMTP client sends Alice’s message over TCP connection to Bob’s mail server.
5. Bob’s mail server places messages in Bob’s mailbox
6. Bob uses his UA to read the message.



**Spear Phishing**

* Directed at specific individuals or companies.
* Attackers may gather personal info (social engineering) about their targets to increase probability of success

**Clone Phishing**

* A type of phishing where a legit and previously delivered email containing an attachment or link has had its content and recipient addresses taken and used to create an almost identical or cloned email.
* The attachment / link is replaced with a malicious version the sent from an email address spoofed to appear to come from the original sender.

**SMTP summary**

* SMTP uses persistent connections
* SMTP requires message (header and body) to be 7bit ASCII
* SMTP server uses CRLF.CRLF to determine the end of message.
* Comparison with HTTP: HTTP = Pull / SMTP = Push
  + **PULL**: Someone loads info on a web server and users use HTTP to pull info from their server at their convenience. TCP connection is initiated by the machine that wants to receive the file.
  + **PUSH**: The sending email server pushes the file to the receiving mail server. TCP is initiated by the machine that wants to send the file.
* Both have ASCII command/response interaction, status codes
* HTTP: each object encapsulated in its own response msg.
* SMTP: multiple objects sent in multipart message.

If SMTP only allows 7bit ASCII, how do we send pictures / videos / files via. Email?

* Base64 encoding makes it possible to send all kinds of data via. email.
* Base64 is designed to an efficient way to send binary data as a string.

Mail access protocols

* Mail access protocol handles retrieval from the server.
* **POP**: Post Office Protocol: authorisation, download
* **IMAP**: Internet Mail Access Protocol: more features, including manipulation of stored messages on server.
* **HTTP(S)**: Gmail, Yahoo etc.

DNS: Domain Name System