ECE 435 – Network Engineering
Lecture 19

Vince Weaver

http://web.eece.maine.edu/~vweaver

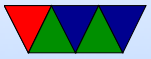
vincent.weaver@maine.edu

14 November 2016

Announcements

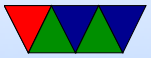
- I replied to all of the project ideas I received
- `http://www.theregister.co.uk/2016/11/14/its_2016_and_a_ping_of_death_can_still_be_a_thing/`

  New ping of death, take down routers by sending a lot of ICMP Type 3, Code 3 packets (dest unreach, port unreach)
- e-mail reply-all loops. Currently one ongoing in England/NHS



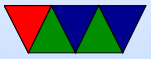
e-mail nettiquette

- Signature, 4 lines 76(?) chars (why?)
- No top-posting!
- Quoting
- Linux kernel rules. Text only. No attachments. No MIME. no line-wrapping. Include patch as text.



SMTP – simple mail transfer protocol

- connect port 25. Text. All commands 4 chars (no one remembers why)
  `S: 220 maine.com SMTP service ready`
- `HELP`
- `HELO a.com`
  `S: 250 maine.com says hello to a.com`
  There is an extended SMTP. You can detect by sending `EHLO` instead
- `MAIL FROM: <xyz@maine.edu>`



`S: 250 sender ok`

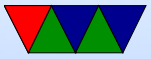
- `RCPT TO: <abx@maine.edu`

  `S: 250 recipient ok`
- `DATA`

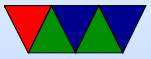
  Put data. . on line by itself is end

  `S: 250 message accepted`
- `QUIT`

  `S: 221 maine.edu closing connection`
- Respond with 3-digit code
  - 2xx = successful
  - 3xx = flow control problem

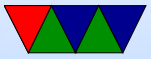


  - 4xx failed
  - 5xx error in command
- In theory supposed to keep retrying to send for up to 4days



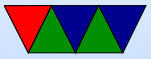
POP/IMAP

- POP (post office protocol) – download mail to local machine which handles
  port 110
- IMAP (internet message access protocol) – manipulate mail on server
  gmail is IMAP. tags are really imap "folders". Can actually download local (I do).



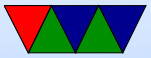
e-mail body

- What happens if try to start line with “From”? Try it.
- Useful to check headers for things like SPAM, phishing attacks
- Signatures (4 lines/80col?)



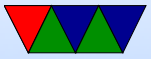
SPAM/other

- In early days, "open relays" if an e-mail came in the server would take mail from anyone and try to deliver it to anyone. Not a good idea (spammers)
- Origin of term SPAM?
  First commercial spam March 5, 1994 Law Firm, Green Card Spam
- Spam/Virus filtering (joke of getting viruses via e-mail)
- procmail sorting
- mail spoofing (What's to stop you from putting someone

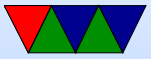


else's address at FROM? how can you catch this?)

- SPAM countermeasures
  - On the sysadmin side, make sure systems are secure. Many ISPs block outgoing port 25
  - SPF records in DNS, say which machines in your network are allowed to send e-mail. Downside, if user has bought a domain and uses it but the ISP doesn't support SPF.
  - Not posting your e-mail, intentionally mixing up your e-mail so address harvesters have trouble getting it. Downside? Things like + in e-mail address?



- Challenge/response. Need to ACK before e-mail goes through. No one likes this.
- DNS black lists, lists of known spamming sources
  Some people block whole countries or all cable-modem connections
- Strict SMTP implementations. Spammers don't always implement their mail senders well.
- Greylisting – delay delivery of the mail by a few minutes (with a 400 response). Most legitimate servers will retry, a lot of spam software doesn't bother.
- Filtering, blocking keywords/all-caps

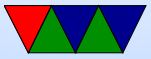


False positives?

e-mails with chunks of books in them, crazy characters

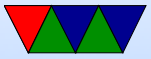
- Bayesian filtering – auto learning. Sometimes can see this in headers

- Vacation Messages
- Mailing lists



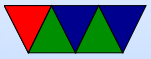
e-mail security

- SSL encrypted connection to SMTP server (usually plain text) SSMTP
- SMTP end to end still unencrypted
- Can use PGP (pretty good privacy) to encrypt e-mails, practically no one does this

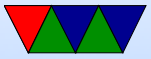


The World Wide Web (history)

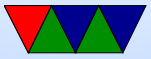
- Before: getting files via ftp (or e-mail/ftp gateways!), could search with archie (archive w/o the V, not comics related)
- gopher: university of Minnesota, 1991. search with jughead/veronica
  Why fail? UMN tried to charge license fee, much more restricted file format than html.
- World-Wide-Web: Tim-Berners Lee, CERN, Initial Proposal 1989, first text-based prototype 1991



- Marc Anderson UIUC worked on graphical browser, Mosaic, 1993
- Anderson went on to form Netscape Communications 1994. Webserver software, made Navigator ("mozilla") relatively cheap/free to drive uptake of web servers.
- Microsoft Internet Explorer. Licensed version of Mosaic. 1995 (as add on to Win95). MS paid percentage royalties to Spyglass Mosaic, so what happened when they gave it away for free?
- Browser wars.
- Netscape bought by AOL in 1998

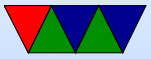


- By 2000, IE had over 80% due to bundling with windows, famous lawsuit
- Gap between IE6 and IE7 of 5 years (2001 to 2006)
- Netscape released firefox as open source in 2004
- Safari/Webkit browsers based off of KDE browser
- Google Chrome took over the lead around 2012 or so
- Standards fight. ACID test.



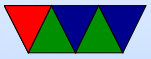
WWW, HTML HTTP

- Standards, World Wide Web Consortium (W3C) 1994



Browsers

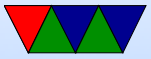
- chrome
- internet explorer
- firefox
- safari
- midori
- lynx/links



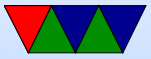
HTML

- HTML – hyper text markup language
- How different from XML?
- Hypertext (documents that can link to each other) actually proposed by Vannevar Bush in 1945
- Simplest form, just a text file with some extra commands specified in angle brackets, and usually a closing tag with a / in it. Case insensitive (though supposed to use lowercase these days).

```
<html>
```

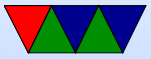


```
<head><title>ECE435 Test</title></head>
<body>
<center><h1>ECE435 Test</h1></center>

<hr>

This is a test.
<img src="image.png" alt="alternate text">

<br>Line Break
<!-- Comment -->
```

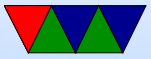


```
<p> Paragraph
<b>Bold</b> <i>Italic</i>

<a href="other.html">A link to another page</a>

</body>
</html>
```

- Tables also easy to do.
- Various HTML versions
- Marquee tag, Blink Tag. Frames.
- CSS (cascading style sheets)



- "view source"
- Originally idea was no formatting, web browser should automatically display simple text in a way to best be displayed on your local machine
  Publishers/graphics designers got a hold of it and that's where all the pixel perfect positioning stuff came in
- Submitting back to the website, HTML forms
- Dynamic content – cgi-bin programs. Write a program that takes input as environment vars, output as standard out sent to the requesting browser.
  Can write in any program. Typically was things like perl,



I often did this in C or even Fortran

- Dynamic content – SSI (server side includes)
- Server extensions (such as PHP) more commonly used (with security issues)
- Dynamic content – Javascript horror. Client side, code runs on your computer rather than on the server.