ECE 435 – Network Engineering Lecture 7

Vince Weaver

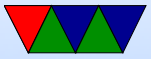
http://web.eece.maine.edu/~vweaver

vincent.weaver@maine.edu

8 February 2022

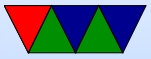
Announcements

- HW#3 was posted, due Friday. Encryption.



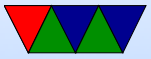
Encryption Problems

- Keys leaked (DVD/game console issues)
- poor random numbers used (Debian problem)
- differential cryptanalysis (start with similar plaintexts and see what patterns occur in output) [DES IBM/NSA story]
- Power/Timing analysis – note power usage or timing/cache/cycles when encryption going on, can leak info on key or algorithm
- Quantum computers



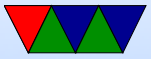
Trusting Trust

- When setting up an encrypted connection, how do you verify who is on the other side?
- How can you protect from man-in-the-middle attacks (MitM) where someone intercepts them downloading your public key, replaces with their own, then sits in the middle decrypting/re-encrypting in a transparent way?
- Some companies/countries will actually do this quite openly



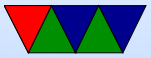
Key Signing Parties

- One way is to have get-togethers were friends sign each others keys
- If enough people do this, you can create a "chain of trust" where you can track someone's identity to someone you trust
- Linux kernel sorta tries this for git development
- Trouble for new people, or remote people, or people who don't travel much, or don't have many friends



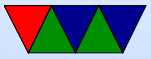
Certificate Authorities

- Certificate authority – an official organization that verifies identities
- Will sign a "certificate" saying who you say you are
- Operating Systems/Web-browsers will ship with a list of officially trusted Certificate Authorities
- Can hover over the lock symbol in URL bar to verify who signed for a website
- Hashed?
- Can be revoked



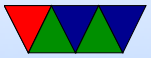
SSL/TLS

- Secure Socket Layer / Transport Layer Security
- Handshake protocol followed by key exchange
- Browser says hello, which hashes/algorithms it supports
- Server picks one and sends back
- Server then sends a certificate (signed by authority) saying who it is, and what its public key is
- Client verifies certificate (via the CA public key it has stored)
- client generates a random number, encrypts with servers



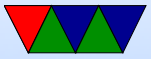
public key, sends to server, used as symmetric key

- What could go wrong, what if someone gets a hold of server private key? could decrypt past logged data.
- Could try Diffie-Hellman key exchange – random number plus unique session key prevents problems if server private key leaked



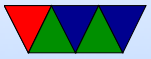
Diffie-Hellman (used by ssh)

- Both sides agree on large prime number
- Both sides agree on algorithm (AES?)
- Each side picks independently picks another secret prime number.
  This is not the authentication private key.
- The secret prime, AES, and shared prime are used to make a public key derived from the private key.
- The generated public key is shared
- The other side uses their own private key, the other side



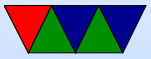
public key, and shared prime to figure out the shared secret key.

- This secret key is then used for symmetric encryption.
- Example on p812

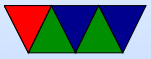


Other tools that use encryption

- How do you encrypt an e-mail, or a hard-drive, etc
- PGP – pretty good privacy
  OpenPGP RFC 4880
  Encrypt message with symmetric key, send along the key encrypted via asymmetric
  was illegal for a while (more than 40 bit encryption an exportable munition)
  people got RSA algorithm in perl tattoos
- GPG – free software replacement for PGP

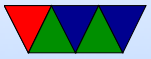


- Can also PGP sign a message. Not encrypted, but signed with your key to verify it was in fact sent by you. Takes hash of the input, then encrypts the hash with key. Also, downloads from servers (like debian)

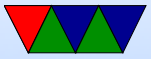


e-mail

- Been around since more or less start of networks
- ARPANET, Ray Tomlinson credited with first modern e-mail around 1971, decided to use '@' char
- UNIX mail, just a mail spool on your computer. Could use command line "mail" to send it. /var/spool/mail/username
- biff to interrupt you when mail came in (used to be exciting)
  origin of name

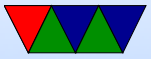


- mbox vs maildir. mbox format, tell each new e-mail via From:. So has to be escaped, you'll see this sometimes. Locking
- Want to send machine-to-machine e-mails. Various ways to do this. UUCP, etc.
- UUCP bang paths



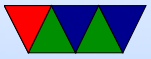
SMTP vs x.400

- As with OSI layer, the big formal ISO definition was made but the hacked-together SMTP won out.
- x.400 much better in many ways
  built-in security
  could tell you once e-mail was delivered
  can send binary files without hacks
- x.400 had horrible e-mail addresses
  C=country, A=administrator (like ISP?can be blank), P= Private Domain, etc



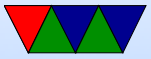
C=US;A=;P=UMaine;O=ECE;S=Weaver;G=Vince;

- x.400 actually used a lot in some situations. Microsoft exchange did for a while
- x.400 so complex that making a working setup was hard so people gave up and used SMTP



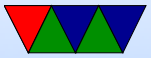
Internet e-mail

- Send/store, can wait on server (as opposed to an instant-message type system where both users have to be active)
- Compose message, send to outgoing server
- deliver to mailbox, collected
- user@host.network
- can often leave off subhost, looks up mailserver for domain via DNS

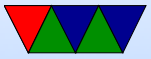


SMTP e-mail layout

- RFC 822/2822/5322
- Envelope first (RFC 821)
- Headers, blank line, body
- originally plain 7-bit ASCII, anything more needs MIME and other extensions
- Headers
  - To:
  - CC: (carbon copy)
  - BCC: (blind carbon copy)

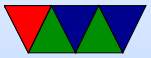


- Message-In:
- In-Reply-To:
- From: / Date: are required
- Reply-to:
- Received: (each transfer agent adds in)
- Return-path:
- Subject:
- X-* (optional extension, people get creative)

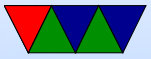


MIME

- Multipurpose Internet Mail Extensions (RFC-1341)
- How do you send Unicode/8-bit ASCII (accents) or Chinese/Japanese
- How do you attach audio/images?
- Backwards compatible
- Message headers:
  - MIME-Version:
  - Content-Description:
  - Content-Id:



  - Content-Transfer-Encoding:
  - Content-Type: (text/plain video/mpeg, etc)
- Encodings:
  - regular: 7-bit ASCII lines, each less than 1000 chars
  - Same, but 8-bit
  - base64 – groups of 24 bits broken into 4 6-bit parts, each a legal ASCII. A=0 B=1 then lower case digits, + /
  - quoted-printable – 7-bit ASCII but higher characters encoded with = sign (hex digits) equal sign =3D
  - multipart

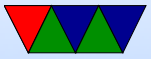


e-mail nettiquette

- 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.
- Eternal September / September that never ended!

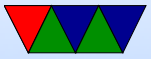
  sdate

  `Tue Sep 10388 12:25:42 AM EST 1993`



SMTP – simple mail transfer protocol

- RFC 821 in 1982
- connect port 25. Text. All commands 4 chars (no one remembers why)
  `S: 220 maine.edu SMTP service ready`
- `HELP`
- `HELO a.com`
  `S: 250 maine.edu says hello to a.com`
  There is an extended SMTP. You can detect by sending `EHLO` instead



- `MAIL FROM: <xyz@maine.edu>`

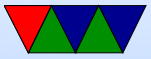
  `S: 250 sender ok`
- `RCPT TO: <abx@maine.edu>`

  `S: 250 recipient ok`
- `DATA`

  Put data. . on line by itself is end

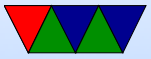
  `S: 250 message accepted`
- `QUIT`

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



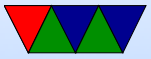
- 3xx = flow control problem
- 4xx failed
- 5xx error in command

- In theory supposed to keep retrying to send for up to 4days

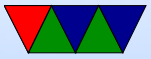


e-mail process

- Sender machine: MUA (mail user agent) sends by SMTP (simple mail transport protocol) to
- MSA (mail submission agent) which determines the destination to send to if not local.
- The MSA uses DNS to look up mail server for destination, then sends it to the receiving MTA (mail transfer agent)
- The final receiving MDA (mail delivery agent) puts into file/mailbox for user

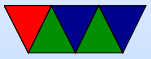


- Receive MUA on local machine via POP3/IMAP
- MUA – editor (optional) mutt/pine/thunderbird/outlook
  Often these days replaced by browser app
  can you use telnet as MUA?
- MTA – sendmail/qmail/postfix
  speaks SMTP. sendmail was standard, has more or less incomprehensible config setup
- MDA – fetchmail? deliver mail tp mailbox. Possibly just a single file, can also be series of directories
- MCA – retrieve e-mail via IMAP or POP



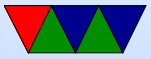
POP/IMAP

- POP (post office protocol) RFC 1939
  - download mail to local machine which handles
  - port 110
- IMAP (internet message access protocol) RFC 2060
  - manipulate mail on server
  - gmail can present as IMAP. tags are really imap "folders". Can actually download local (I do).



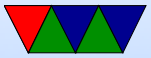
e-mail body

- 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?)



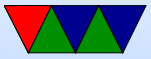
SPAM/other

- 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

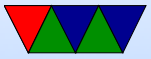


else's address at FROM? how can you catch this?)

- 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

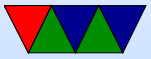


False positives?

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

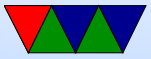
- Bayesian filtering – auto learning. Sometimes can see this in headers

- Vacation Messages
- Mailing lists



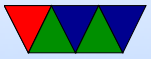
e-mail security

- 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



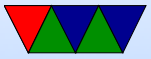
Can you run your own e-mail server

- Used to be possible/common
- Common e-mail servers: configuring them, specifically SendMail

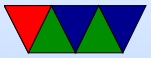


Other common protocols we won't cover

- Legacy (inetd): echo, chargen, discard, time, finger (.profile, .plan), qotd, systat, write, talk
  why no longer supported? security? lack of interest?
- Messaging:
  - IRC – internet relay chat
  - AIM/ICQ/MSN etc
  - unix talk/write
  - MUDs, talkers
- IPP – printer protocol (CUPS, lpd, jetdirect)



- backup software
- syslog
- telephony
  - skype
  - facetime
  - VOIP
  - ASTERISK
- ntp – network time protocol
- LDAP/Authentication
- Network Attached Storage/Fileservers
  - NFS



  - Samba/CIFS
  - andrewfs (afs)
- Databases: mysql
- Distributed/Torrent sites
- Distributed computing (SETI@Home)