---
layout: notes
title: Email Security
scribe: George Gelinas

- Neither Snow Nor Rain Nor MITM
- Lots of layers of email security
- PG
    - PGP message begins with ----Begin PGP Signed Message----
    - When we send an email that is PGP encrypted
        - To and From and Subject is not encrypted
        - Only the body is encrypted
        - Unlock the security key for the certificate, everything is visible except for body of the text
        - NTP attack is the subject of the email
        - PGP encryption leaks the subject
        - Entering passphrase to open the email
        - It is an RSA encrypted message
    - Matt has message m and encrypt the message under Sharon's public key
    - 4906 bits long for RSA, message can be longer than 4906
    - Take the message and encrypt it under a session key
    - RSA encryption under public key of Sharon of session key of message
    - Session key is encrypted in RSA and then encrypt the message
- Object level security
    - All you are protecting with this is the message
    - If you are signing the message
        - Sigma = RSA_skmatt(ciphertext)
        - I would also have the public key of matt
        - And I have the secret key
- You learn each other keys by going to key exchange parties and exchange physically in person
- Choose my own pair of keys
    - Make a key for myself
    - Then write messages and sign it with Omars key
- Other way to get keys
    - Key servers,
        - Verify with Matt, over the phone and through text
        - Look through key server and find emails with keys and use that key to send encrypted messages
        - Grab that name get the public key from
        - Attacks
            - Compromise the key server
            - Upload false public keys on the key server
- Not used on a large scale

- Use the wrong key, the adversary can decrypt it
- SMTP Protocol
    - Alice and Bob
    - Send an email to their mail server (Alice)
    - Mail server of bu.edu
    - MTA (Mail transfer agent)
    - One protocol that you speak with the mail server and bob has one as well
    - Connecting over webmail
    - Speak to gmail, we use HTTPS
    - TLS protects connection form your browser to gmail
    - Two mta servers have to communicate with each other
    - Every web mail provider provides secure connection
    - Connection between the two mail servers
        - When we have a mail, we need to figure out how to find each other
        - Use DNS in email is tied up in each other
        - When this guy gets this email towards bob.
        - MX record is for mail transfer
        - Find the MX record for the destination.com
        - MX has the host name for the server
        - Simple mail transfer protocol (SMTP)
        - We have this destination.com
        - Then query for the A record
        - Now he can put it inside a packet and the destination is the other mail server
        - Send the mail to the ip and it will deliver it to Bob
        - Worry about DNS manipulation
- Fraudulent DNS responses
    - They did an internet wide scan
    - Went around and identify DNS servers on the internet
    - Send DNS packets to get a response
    - See what kind of MX from popular mail providers
    - Put the query for gmail.com, see if answer is a real server or see if it was something else
    - Did it for 5 different mail providers
    - DNS
        - Each one of these zones have a key
        - There would be a zone signing that would sign all of the DNS records
        - What happens is that you would probably have a key signing key
        - Hash of the key signing key, put the hash in the parent zone
        - Parent has its own key
        - That key is used to sign everything in the zone
        - No certificates but there is chaining up to the roots
- DNS Seq
    - When we have a computer, it does not ask nameservers
    - Go through a resolver, tell it to ask the queries for us

    - DNS seq is applied to the recursive resolver and public internet
    - DNS seq will prevent man in the middle if the recursive resolver is honest
    - Using a recursive resolver that has been manipulated, then it could be giving you the wrong answer
    - So the computer to the recursive resolver is not secure
- If I am expecting TLS with gmail, someone give a wrong TLS response. The browser will not connect
- Email is a soft fail, but in email if the connection fail, we continue to speak but unencrypted
- Soft-hard fail is typical for TLS
- Soft fail for email (SMTP)
- Retrofitting is putting something onto something already there
- STARTTLS
    - First message is TCP handshake
    - Then this guy says 220 ready,
    - Then source mail server sends EHLO
    - If the destination mail server wants to speak in an encrypted way then it will STARTTLS
    - Now that the encryption is secure then communication happens
    - Three message negotiation, handshake
    - If the source does not want to do encryption, it will not accept the STARTTLS and then the communications will be unencrypted
    - Encrypt all of the email
    - There is this negotiation, a lot of MTA don't speak in encryption
    - There will be a soft fail if the source server does not want to speak in TLS
    - There is a mode of operation of TLS where both sides have certificates
    - In this setting both sides will have certificates
    - It is important that each side knows that they are talking to each other
    - Attack
        - Man in the middle
        - Message was intercepted for STARTTLS and replaced with X's
        - Downgrade attack
- They actually found it happening
- Looking at which SMTP servers for ip prefixes and see if they have this type of behavior of stripping STARTTLS
- A lot of the devices were cisco for tampering
- One of the problems in TLS is a downgraded attack in TLS
- There is a technology called HSTS
    - Make a conection with gmail
    - Succeeds cause no one is messing with my connection
    - In the future, we have to speak TLS and if it is not then we have a hard fail for a period of time like a year or few months
- Most MTA's do not check for certificates
- Transport security of email is insecure
- SPF, DLIM, DMARC

- When this source server is sending to the destination server, it is going to add the DKIM signature
- Signature is signed by the secret key of the source server
- DKIM query to DNS to find the key for source.com and it is returning the RSA key
- As long as no one messes with DNS query, then we can verify that it was validly signed
- It only provides integrity and authentications
- We have to rely on DNS and DNS could be vulnerable if it gives false keys
- DMARC
    - Tells you if the signature value fails. It will respond the policy is reject.
    - Different DMARC policies you can have
    - If it fails you can set up to not receive the message
    - Not a lot of places have DMARC policy, people are very afraid to reject emails