9. More Authentication and Trust

COMP6441 • KC Notes

9.1 Data, Control and Authentication

- **Data and control** is a very general idea and plays a large role in trust
    - Sending emails is considered **data**, and if abused, may **control your account** through links, attachments
    - **Authentication** of emails: susceptible to impersonation, fake website, man in the middle attack
- The problem is **to authenticate a website that people trust**
    - **Horton's Principle: "I meant what I said, and I said what I meant"**
    - We need to make sure **we are validating <u>the correct thing</u>**
    - What you think **what it means**, may not be the same as **what you see**

9.2 Two Authentication Approaches

1. **<u>Centralised Authority</u>**: a bank that gives you a signature, and you trust this bank
    - Based on command and control
    - Becomes a central point of failure
2. **<u>Decentralised, peer based Web of Trust</u>**: More people have their say and you trust these peers
    - Example: Pretty Good Privacy (PGP)

- Relies on **public key infrastructure (PKI)** – to talk to me, you only need a public key. To decrypt a message, you need a private key.
    - Man in the middle attack: **you encrypt with the wrong public key**, or person sends you the wrong public key
    - **X509** is the standard for public key certificates.

9.3 Certificate Authority (CA)

- Is an online and trusted
    - To sign via PKI, a company encrypts their certificate with **a private key** – people can **check this certificate with the public key**
    - Web browsers **come with preloaded public keys** from the CAs. All browsers do is **check if it matches**.
- However:
    - Anyone can become a CA, or **edit/create a certificate in your phone**
    - Why do CAs check and prove authentication? For **money**, e.g. people pay the cheapest CA, and CAs can pay to a browser
    - Malicious governments could place dodgy certificates and man-in-the-middle everything
    - **Horton's Principle**: you are authenticating **the URL/email address**, NOT the **actual company**. So, gitson-security.com may be a spoof of gitson.com, but a CA will authenticate it as owned by 'Gitson Security'.

9.4 Forward Secrecy

- **Perfect secrecy/Perfect Forward Secrecy**: if they learn your shared secret, **everything in the past is safe** and past messages and data is protected
    - This is usually not the case
    - E.g. S/Key: knowing the password means you know all the passwords

9.4.1 KEY DISTRIBUTION PROBLEM

- **Key Distribution Problem**: Distributing keys in public or over the internet is difficult as it must be done face to face. PKI solves this as a person can easily create a key pair and only share the public key.
- PKI is **more expensive** but solves the key distribution problem
- So, usually PKI is used to obtain a session key
    - A **slow, asymmetric** process, then a **fast symmetric** process for a period of time
    - Once a period of time is over, you can send another disposable public key
    - Obtaining a session key only unlocks a small part, **past data is protected**