8. Identity and Authentication

COMP6441 • KC Notes

8.1 Intro to Authentication

- When a computer program attempts to authenticate, their **only interaction is through 1's and 0's** – it is like looking only at a screen in front of you
    - **Authorisation**: whether someone has the permissions to access something
    - **Authentication**: verifying that someone is who they say they are
- Authentication can be done with **context or a pattern**
    - Must be used in plain sight
    - Is asymmetric – for example, a shared secret for each pair
        - Secrets **may not be tamper evident**
    - Should prevent replay attacks in case it is eavesdropped and altered
    - **Challenge response**: a way of authenticating you are speaking to the correct person
- The launch codes to US nuclear weapons were 00000000 in all locations
    - **Type 1/Type 2 errors** involved in launching
        - Either: having the military capable of immediately responding to a threat
        - Or: having another party launch the weapons easily, and relying on obscurity

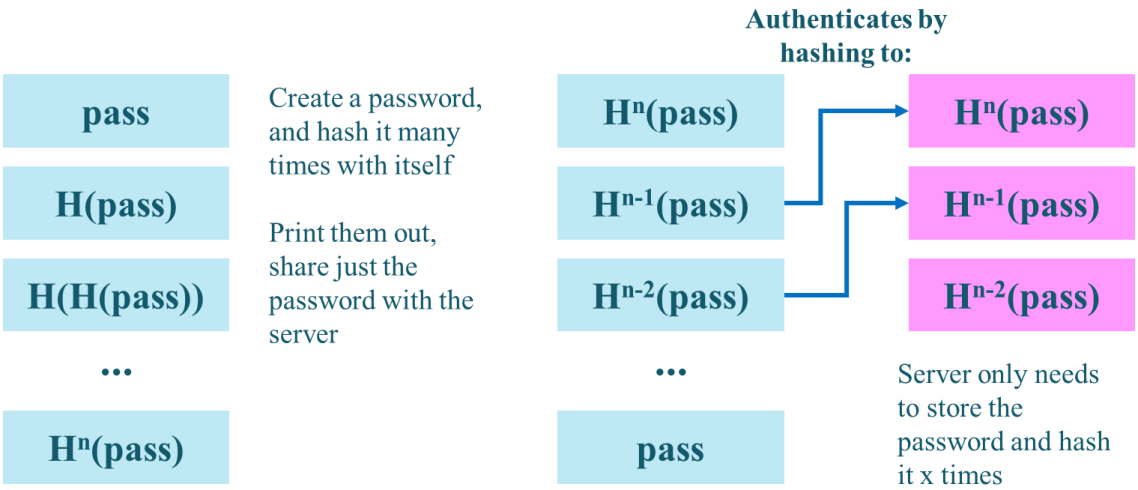
8.2 Identity and Authentication

- **We authenticate with something we <u>know</u>, something we <u>have</u> and something we <u>are</u>.**
    - **Know**: a secret, but can be **leaked**
    - **Have**: a card, license or phone, but can be **forged**
        - A card also becomes **information** and becomes something you know
    - **Are**: fingerprint, way you walk
        - Fingerprints are **digitised and stored (knowledge)**. Also, it is difficult to change your fingerprints and can be lifted or read easily.
- **2 Factor Authentication**: if you compromise one factor, you may not be able to compromise the other.
    - E.g. Key logger can get your password (know) but not your 2FA app on phone (have)
- Identity theft **costs 2.2 billion every year**
    - Hard to change birthday, address or name
    - Driver's license: $500, Passport: $5,000

8.3 Authentication Protocols

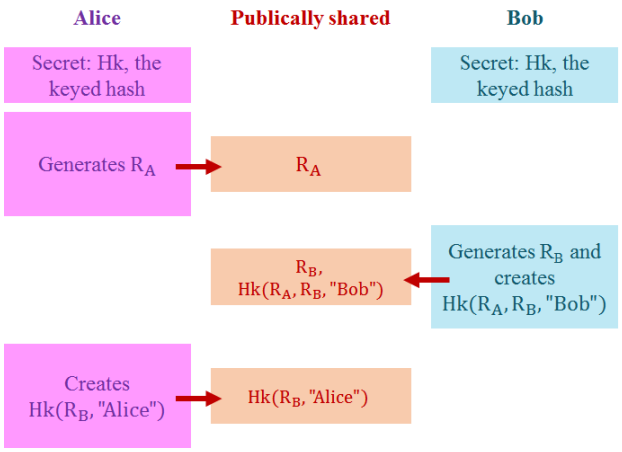
8.3.1 S/KEY

- **S/Key**: a password that is hashed many times – **prevents replay attacks**.
    - When we need to authenticate, we use the **second last hash** which will hash to the last hash. The next time we authenticate, we use the **third last hash**, etc.



8.3.2 SKID ALGORITHM

- **Secret Key IDentification protocol** (SKID): a way of authentication that relies on a secret **keyed hash Hk()**.
    - Alice sends random number R_A
    - Bob sends back random number R_B, and the key-hashed string $R_A + R_B$ + "Bob"
    - Alice sends back key-hashed string R_B + "Alice"



- We don't send just Hk(R_B, "Bob") to **prevent replay attack** (R_A is a nonce, just R_B could be easily replayed)
- We must send "Bob" **to prevent reflection attack** – we could be talking to ourselves without us knowing

8.4 Time of Check, Time of Use

- If there is a difference between **time of check and time of use**, it can cause a security vulnerability
    - If there is a time between checking if a user can **access a file**, and opening the file:
        - We can **change the filename before it opens but after it checks**
    - **Plane tickets** – they check your identity at security but at the plane, the boarding pass can be exchanged and you can board someone else's flight