3. Risk and Key Cryptography

COMP6441 • KC Notes

3.1 Risk

- Humans struggle to **weigh up risk and put a price on it**
    - Richard could repair his laptop in 2 hours or in 1 hour, but will not think the amount of care and attention behind the curtain – only **cares about the cost**
- We are **not rational** about risk
- You can usually see a bad event occurring, not the chance of a bad event occurring
    - Humans only focus on **the outcome**
    - People are just lucky or unlucky, but in the end **there is the same amount of risk** for each person

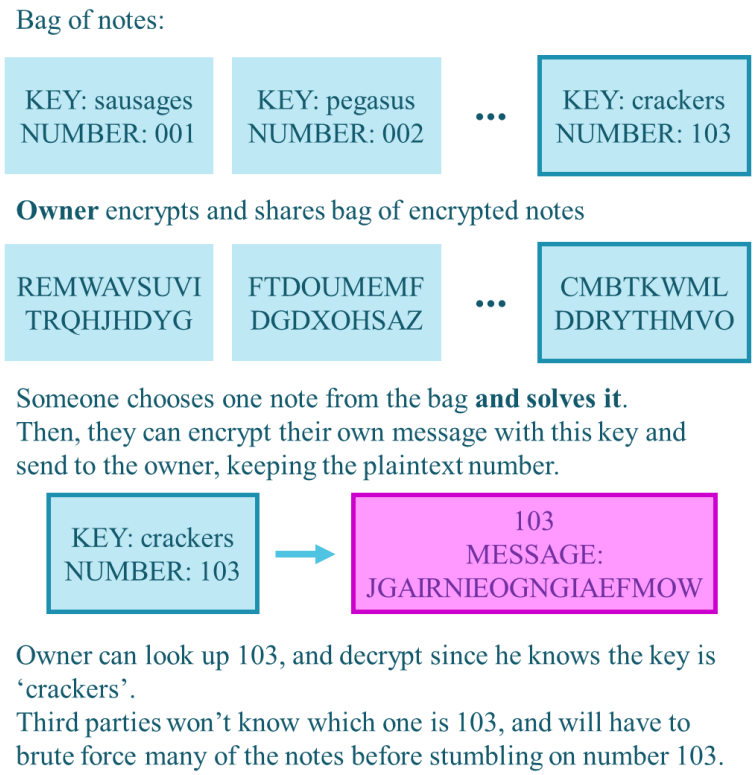
3.2 Low probability, high impact risks

- Humans are particularly bad at measuring risk when **the impact is catastrophic**, but it has **a very low probability** of occurring.
    - We usually shift this risk to another person (e.g. climate change)
- Examples: NASA gathering data on near-Earth objects, volcanoes, mass shootings, the San Andreas fault line, mass shooting
- We are getting good at measuring **risks in finance**, so we should apply ideas from finance
    - Both security and finance **protect something that might happen**
    - Little amount of past data in security, compared to finance
    - Companies struggle to know how much they need to invest in security
- In computing, low probability high impact risks are growing
    - **Companies storing everything in a central location** (single point of failure)
        - AWS, Google Drive, Gmail, Windows
    - This is a trade-off between benefits (speed, costs) and higher impact catastrophes

Kris Choy • 26 March 2017

3.3 Keys: Ralph Merkle

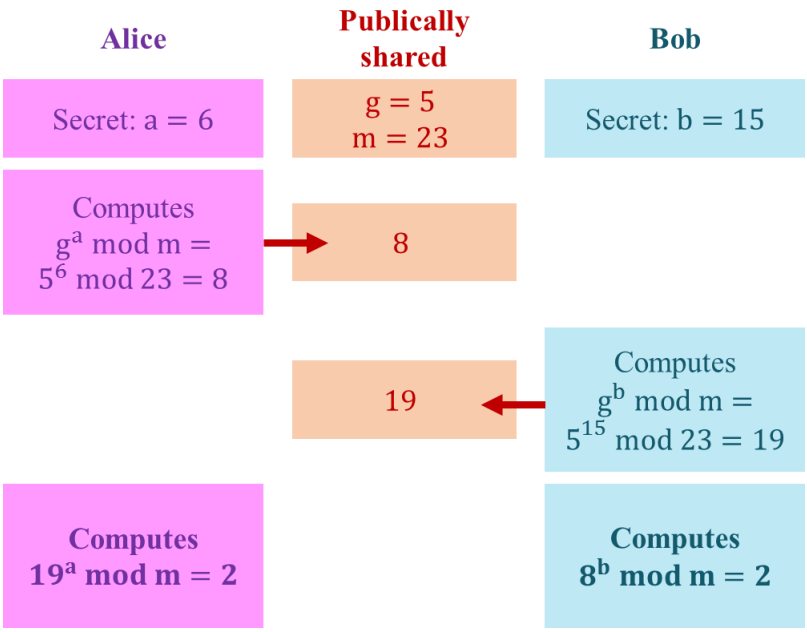
- With a large group of people, each pair of people need a secret key between them
- **Public key cryptography**: a public key for writing, a private key for reading messages
- **Ralph Merkle**:
    - Get a large bag of notes with different keys and a number. Encrypt both message and number, where the owner still knows the original message and number.
    - Share the encrypted messages.
    - Someone can **brute force and solve one note**, gaining one key and one number. They can then also encrypt, but **provide the number (e.g. 103) in plaintext**.
    - The owner can look up the number and decrypt using the key.
    - Third parties will have to **brute force and crack on average half the notes to find note number 103** in order to decrypt.



- Merkle's system relies on **asymmetry** – it takes much longer to decode every message than to solve one message.

3.4 Keys: Discrete Logarithm Problem and the Diffie-Hellman Key Exchange

- **Discrete Log Problem**: $g^a \ mod \ m = x$
  - Given g, m and x, **finding a is very difficult**.
  - We can utilise this difficulty to generate a private key
- **Diffie-Hellman Key Exchange**: how can you create a secret key between two people, without it being shared in public?
  1. Alice chooses a secret value **a**.
  2. Bob chooses a secret value **b**.
  3. Alice and Bob decide on public values **g and m**.
  4. Alice sends g^a mod m to Bob publicly.
  5. Bob sends g^b mod m to Alice publicly.
  6. Alice computes $(g^b \text{ mod } m)^a$ mod m = $\mathbf{g^{ab}}$ **mod m using her secret value a.**
  7. Bob computes $(g^a \text{ mod } m)^b$ mod m = $\mathbf{g^{ab}}$ **mod m using his secret value b.**
  8. Both now have a shared secret key, g^{ab} mod m!



- It is hard to get a and b, even when given all publically shared numbers.

3.5 Bits of security

- **Bits of security**: a measure of the amount of work needed to decipher a key.
  - Measured in bits (base 2)
- E.g. if we have to do $1000 \approx 2^{10}$ operations (e.g. 1000 'attempts' to decrypt) then it has 10 bits of security
- Exploiting space/time trade-offs (using more than one computer) halves the number of bits
- 128 bits of security is a good ballpark for the universe to end