School of

Engineering and Computer Science

Te Kura Mātai Pūkaha, Pūrorohiko

CYBR 171 T1 2023

Ngā whakapūtanga o Te Haumaru rorohiko
Cybersecurity Fundamentals

Modern Cryptography II
Key Distribution and Diffie Hellman

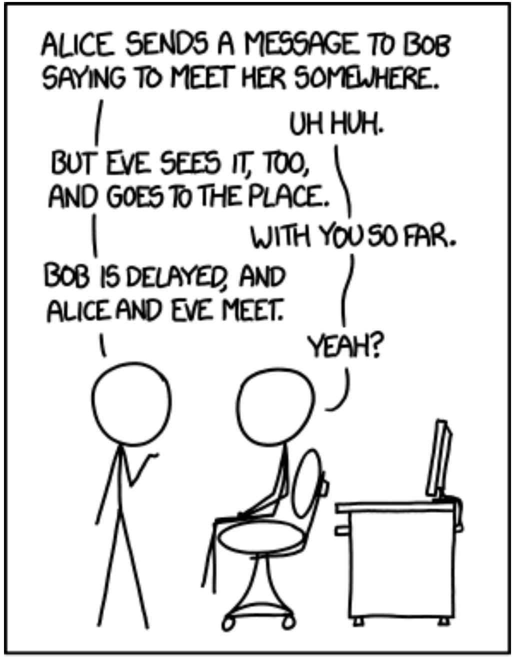
What are we going to cover

- Key distribution problem
- Diffie-Hellman
- MiTM attacks
- Fingerprinting
- Wrap up and what's next

PART I:
Key Distribution Problem

Alice, Bob and Eve

- Cryptographers use the fictional characters “**Alice**” and “**Bob**” exchanging messages.
- An eavesdropper “**Eve**” is usually listening, there is also “Mallory” and “**Trudy**”.



I'VE DISCOVERED A WAY TO GET COMPUTER SCIENTISTS TO LISTEN TO ANY BORING STORY.

https://xkcd.com/1323/

Key Distribution Problem

- Alice wants to send an encrypted message to Bob across an "**unsafe**" network.
  - A real-world example of this is a public wifi network where anyone can connect
- Eve is listening on the network, so Alice can't send the key as well.
- Alice can't simply **encrypt the key** because we have the same problem again, how to send that encryption key.
- Alice could always **walk it across to Bob** but that might be physically impossible or cumbersome whenever a key needs to be changed.

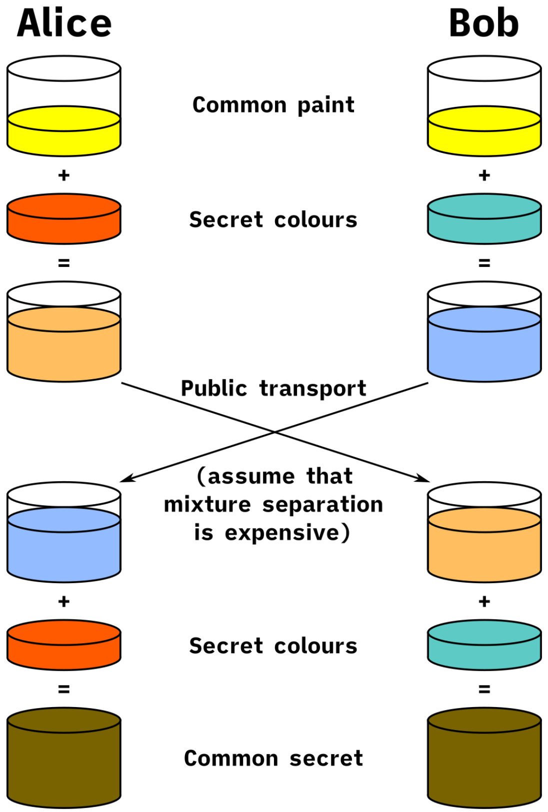
Diffie-Helman Key Exchange (DHKE)

- Diffie & Hellman with Merkle (1976).
- 1st Public-key cryptography scheme.
- Alice and Bob want to compute shared secret key.
- The secret key is never sent over the network.
- Requires **hard-to-reverse** method of combining elements.
- Basis of Transport Layer Security that secures **SSH** and **HTTPS**

PART II:
Diffie-Hellman

Key Exchange by Mixing Colours

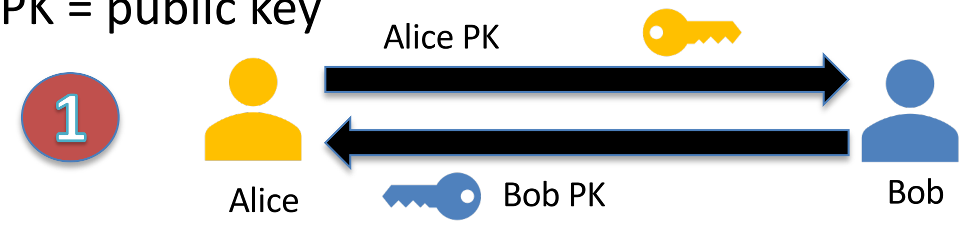
- **Assume**:
  - Two liquids of different colours
  - Mix colours to obtain a new colour.
  - No way to separate mixed colours out again.
- **Scenario**:
  - Alice and Bob agree on starting shared colour (it doesn't need to be secret).
  - Alice and Bob select a secret colour (not shared).
  - Alice and Bob mix their secret colour with mutually shared colours.



Source: https://en.wikipedia.org/wiki/Diffie–Hellman_key_exchange

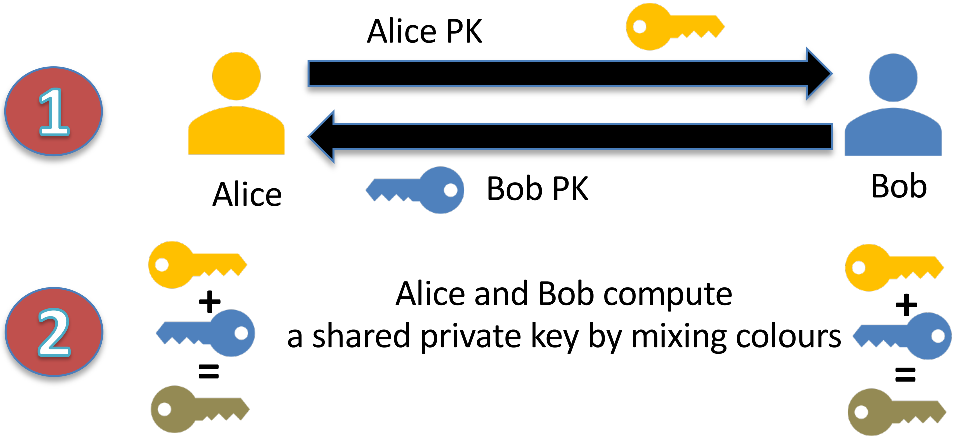
Diffie-Hellman protocol 1

- What does this look like in practice?
  - Note PK = public key



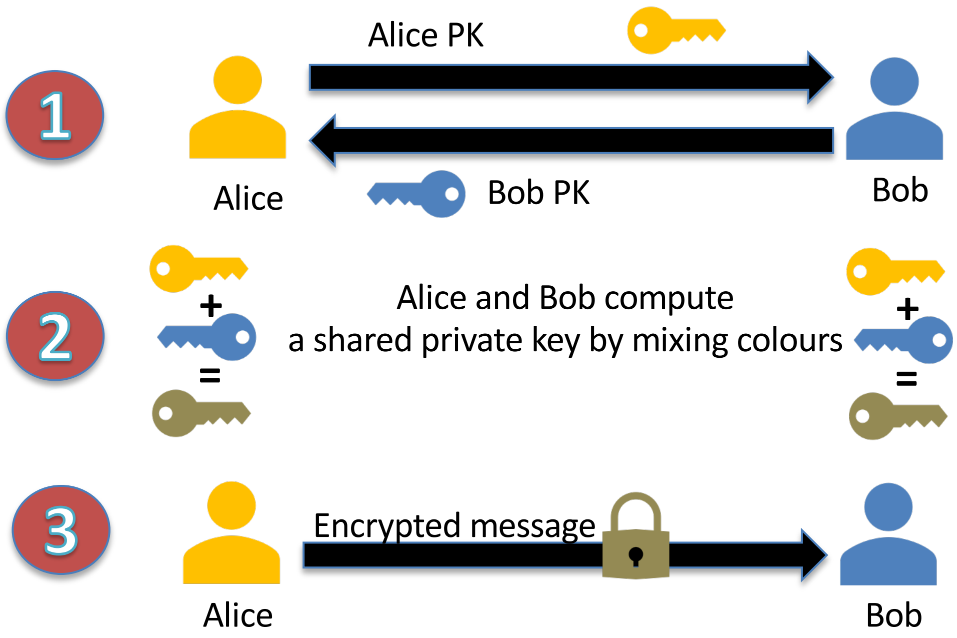
Diffie-Hellman protocol 2

- What does this look like in practice?



Diffie-Hellman protocol 3

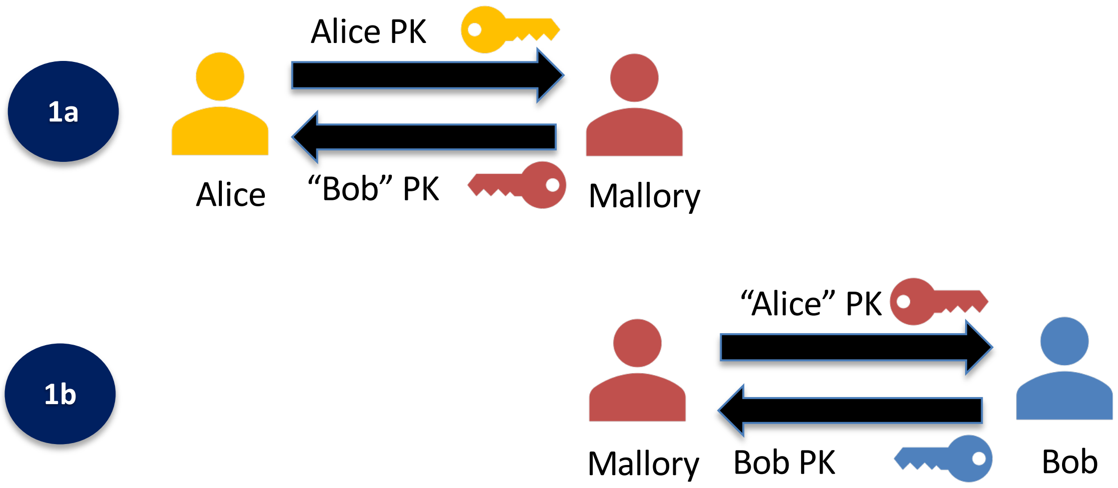
- What does this look like in practice?



PART II:
Man-in-the-Middle Attack (MitM)

MitM attack on Diffie-Hellman

- Mallory-in-the-middle, man-in-the-middle, monster-in-the-middle, machine-in-the-middle, monkey-in-the-middle or person-in-the-middle.
  - Mallory intercepts communications between Alice and Bob
  - Aim is to be able to eavesdrop on secret messages and optionally change them

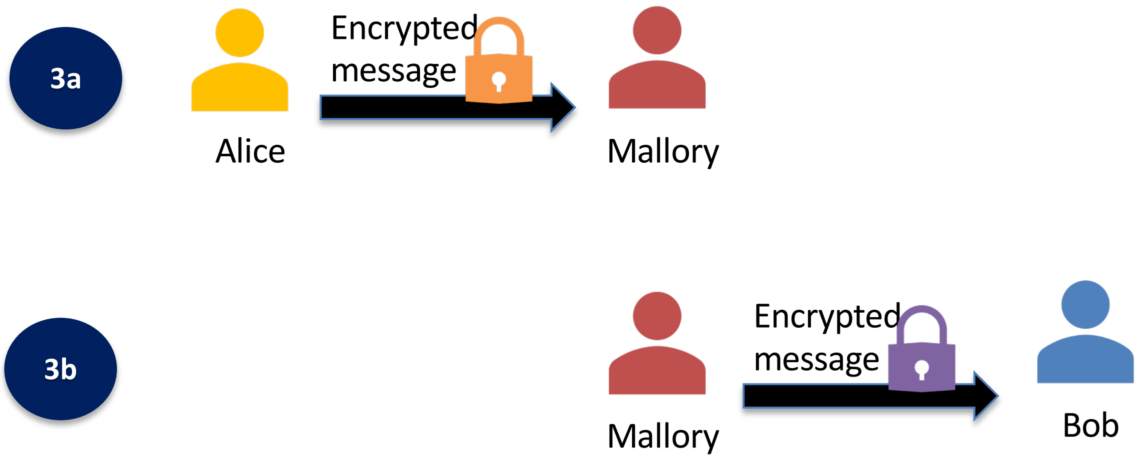


MitM attack on Diffie-Hellman (cont.)

- Mallory-in-the-middle, man-in-the-middle, monster-in-the-middle, machine-in-the-middle, monkey-in-the-middle or person-in-the-middle.
  - Mallory intercepts communications between Alice and Bob
  - Aim is to be able to eavesdrop on secret messages and optionally change them
  - Abbreviate public key to PK and secret key to SK

MitM attack on Diffie-Hellman (cont.)

- Alice encrypts message using her key shared with “Bob”.
- Mallory “Bob” decrypts the message and reads or modifies it.
- Mallory “Alice” encrypts the message using their key shared with Bob.
- Bob decrypts message using his key shared with “Alice”



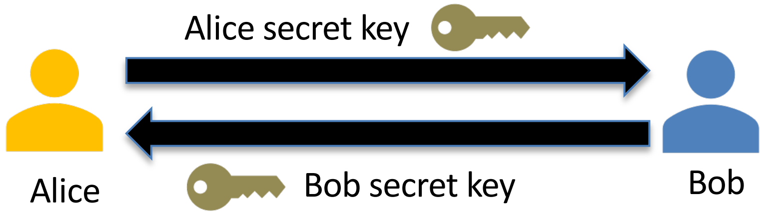
MitM attack on Diffie-Hellman (cont.)

- Alice and Bob believe they are talking securely with each other

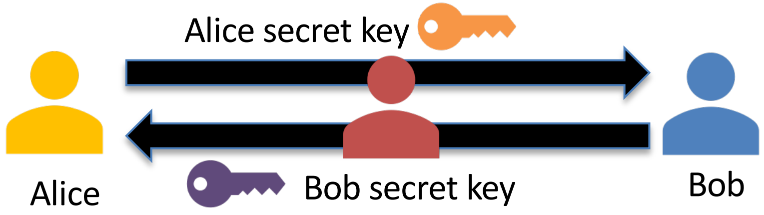
- Mallory is able to secretly eavesdrop or modify their communications

Detecting MiTM

- Alice and Bob compare their secret keys.
  - Same? Trust the keys, no MiTM is happening.



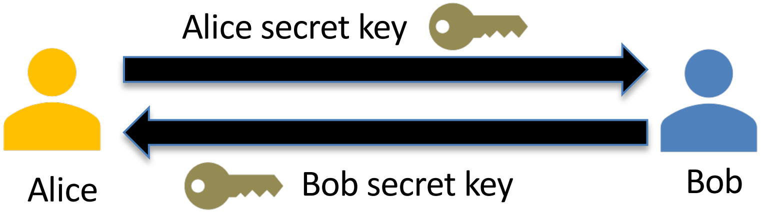
  - Different? Dob't trust the keys, MiTM happening.



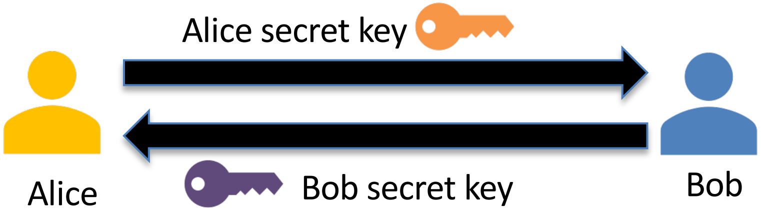
Detecting MiTM (cont.)

- Alice and Bob compare their secret keys.
  - Same? Trust the keys, no MiTM is happening.

Problem?



  - Different? Don't trust the keys, MiTM happening.



Problems with direct comparison

- **Send secret keys in plaintext**
  - Defeats the purpose of the protocol

- **Encrypt the secret key**
  - Alice and Bob decrypt and compare keys
  - Mallory manipulates the keys to appear the same

Out-of-band communication

- Out-of-band refers to the communication channel

- Alice phones Bob
  - Compare the secret keys
  - Mallory cannot intercept phone calls
  - Mallory cannot manipulate keys

- Secure method but is it practical?
  - Is there a way to avoid sharing the whole key?

PART III:
Fingerprinting

Cryptographic Hash

- Hash function

  *output = hash(**input**)*

  - Input: Data of **arbitrary** size
  - Output: Data of **fixed** size
- Fingerprint, digest, hash value or hash of the input
- Four properties:
  - **Same** input = **same** output
  - Can't workout input from knowing output (**one way**)
  - Two different inputs ≠ same output (**collision**)
  - **Small** change to input = **large** change to the output

Examples

- MD5
  - 128-bit fingerprint
- SHA family
  - SHA-1
    - 160 bits
    - Collisions possible
  - SHA-2 (current)
    - 224, 256, 384 or 512 bits
  - SHA-3 (new)
    - Arbitrary bits, fundamentally different inside

Detection of MitM using Fingerprinting

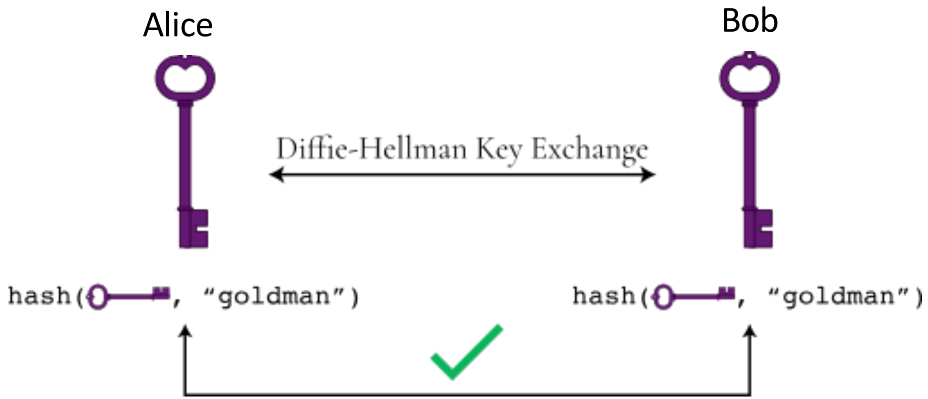
- Two **in-band** methods:
  - Weak password
  - Key comparison in voice calls

- Both are variations of the in-band method.

- Both are **limited** but are good examples of the design of security protocols.

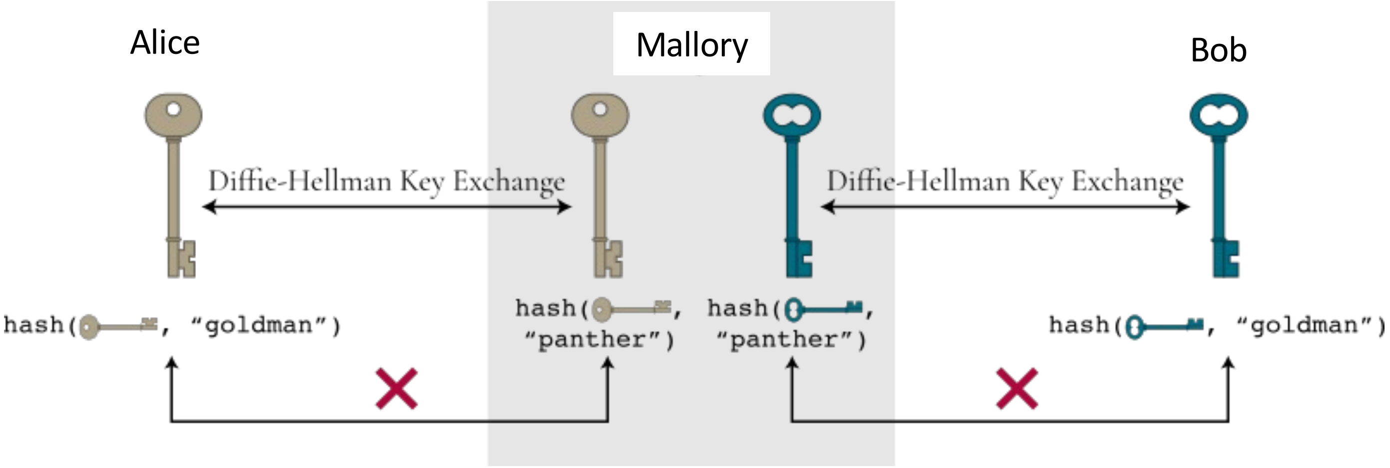
Weak Password

- Alice and Bob know the same weak password
  - Assume know it because of **prior knowledge**
  - Something they can guess (**hence weak**)
  - An example would be that they know which street Alice grew up on
- Hash computed with **password** and **key**



Detection of MitM Attack

- Mallory will generate different hashes because:
  - Mallory has different keys shared with Alice and Bob
  - Mallory doesn't know the password that is shared between Alice and Bob
- Alice and Bob will assume MitM attack



Key Comparison in Voice Calls

- Application - agree on a secret key to encrypt voice calls
- Send **hashes** over the same communications channel
- Method to check that no MitM is happening:
  - Alice and Bob **agree** on a secret key using Diffie-Hellman
  - Alice and Bob both **calculate** the hash of the secret key
  - **Convert** hash to **TWO** human readable words
    - E.g. Orange Banana
  - Alice reads out the first word to Bob
  - Bob reads out the second word to Alice
  - Alice & Bob should have the same pair of words

Detection of MitM Attack

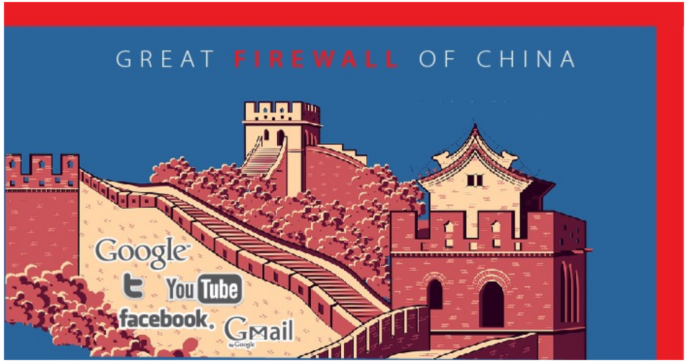
- Alice and “Bob” (**really Mallory**) agree on a secret key using Diffie-Hellman
- Alice now wants to check she is really talking to Bob
- Alice and “Bob” calculate a hash as two human-readable words
  - E.g. Kiwifruit Mango
- Alice reads the first word to “Bob” and waits for “Bob” to read the second word
- Alice and Bob will have different pairs of words if Mallory is in the middle

Why Can’t Mallory Pretend to be Bob?

- **First**, it relies on Alice recognising Bob’s voice.
  - When Mallory speaks, Alice knows it isn’t Bob.
  - This is a form of **biometric** (see next week).
- **Second**, Mallory won’t know the shared key if started impersonating Bob after the shared key was agreed.
  - Mallory won’t be able to provide the **correct second word**
  - Mallory would have been able to do so had Alice told “Bob” both words in the first place (which is why we don’t do this!)
- These are **two layers of countermeasures** put in place to try and detect a MitM

Real MitM attacks

- China blocked github Jan 2013
- China "re-opened it" later.
- Users reported fingerprints failing.
- China had mounted a MitM attack.



https://theprivacyblog.com/blog/censorship/china-launches-mitm-attack-on-github

PART IV:
Wrap up

Summary of Lecture

- Diffie-Hellman solves the key distribution problem across an unsafe network.
- MiTM attack on Diffie-Hellman possible.
- In-band detection techniques possible using fingerprinting.
- In-based detection techniques still assume some prior knowledge (password, voice patterns or shared secret key).
- Next lecture - *RSA algorithm avoids this problem by introducing different public and private keys for each party in a conversation.*

What’s up next

- More material in the next lecture.
  - RSA explained
  - Applications of RSA
  - Quantum and post-quantum