

## C2107 Tutorial 3 (Password, 2FA, a bit of PKC)

Chang E.-C., School of Computing, NUS

September 16, 2025

1. Number of 5-words combinations approximately:  $2^{10} \times 2^{10} \times 2^{10} \times 2^{10} \times 2^{10} = 2^{50}$ . Entropy is 50.

Answer: 5

- *Remarks:* There was an inconsistency on how many bits were required to prevent online dictionary attack between RFC4086 and Q1's recommendation. In the updated version of Tutorial 3 (v1), the recommendation was changed to 49 to be consistent with RFC.

2. Smaller than 0.5 in order to be more accepting than the server room.  
Recall again that when threshold is 0, the system accepts everyone; and when it is 1, the system accepts no one, i.e. rejects all.

### 3. (Token)

- (a) Consider a malicious PC who performs the following. When the user instructs PC to carry out a transaction, say  $T$ , the malicious PC submits another transaction  $T'$  (for e.g., transfer the money to the attacker's account) to the bank. The malicious PC then asks the user for the OTP, and submits the OTP to complete the transaction.

- In the **Token** setting, the user would not notice that  $T$  is being replaced by  $T'$ .
- In the **SMS** setting, the full detail of the transaction is included in the SMS. Assuming that user's phone is not compromised, and the user is attentive, the user can detect that the transaction  $T'$  is different from the original  $T$ .

The attack scenario is realistic. (1) The PC could be infected by malware; (2) The PC to an attacker, e.g. Internet Cafe or airport computer (3) Phishing attack, where the user is being tricked to visit a fake website. The fake website plays the role of the fake PC. To study this setting, we need to know more about DNS and https, which will be covered later in network security.

- (b) Note that here, the laptop/PC is not compromised.
  - In the SMS setting, an attacker can sniff the SMS (that is the assumption) and see the detail transaction. So, there is a leak of information.
  - In the Token setting, no additional channel for the attacker to get the transaction detail.

- (c)
    - The attack is same as the attack on Token. We first assume that the PC is malicious. Next, carry out the attack as described earlier. This attack works on M1 but not M2.
    - Display partial information. For e.g. *“You have requested to transfer \$50,000 from account ending with 3-23 to the account ending with 4-A2, enter OTP: 132373”*.
4. Let's us write  $c = \langle v, c_0 \rangle$  where  $v$  is the IV. Note that  $c_0$  is a 33-byte sequence.

Mallory generates

$$c' = \langle v, c_0 \oplus (0_{21} \| d_4 \| 0_8) \rangle$$

where

- $0_{21}$  is a 21-byte sequence and each byte has value 0 (i.e. a null character);
- $d_4$  is a 4-byte sequence where value of each byte is  $d$ , and  $d$  is determined in the following way. Let  $A_0$  and  $A_9$  be ASCII representation of character 0 and 9 respectively. The byte  $d = A_0 \oplus A_9$ , in other words,  $d$  is the different between 0 and 9. Also note that  $A_0 \oplus d$  is  $A_9$ .
- Likewise,  $0_8$  is a 8-byte sequence with value 0.

(Remark: The notation  $\|$  refers to string/sequence concatenation.)

Note that the IV in  $c$  and  $c'$  are the same. Decryption of  $c'$  will give the intended

*“Ux gives Uy \$9999 dollars”?*

Interestingly, Mallory still doesn't know  $x$  and  $y$ . So, even if confidentiality is preserved, Mallory can still compromised integrity.

5. Consider a scenario where an attacker has physical access of the phone and able to unlock it. In such scenario, the attacker by viewing the SMS message, would able to know which account this phone is linked to. Next, the attacker reset the password of those accounts. The OTP would be sent to the phone. The attacker can now complete the password reset process. Note that here, the phone become the single point of failure!
6. Smaller exponent leads to faster computation of exponentiation.  
Google website use DSA.