ECE568 笔记汇总

- Cryptography - Block Ciphers
- Cryptography - Ciphers
- Cryptography - Hashes, MACs, and Digital-Signitures
- Cryptography - Public-Key Cryptography
- Cryptography - Stream Ciphers

Table of Contents

Cryptography

- protecting **stored** and **transmitted data**
  - *Confidentiality* - Secrecy of data; provided by algorithms called ciphers
  - *Integrity* - Trustworthiness of data; provided by Hashes; no corruption/modification
  - *Authentication* - Allows a machine to prove origin of data; **signatures** & MAC
  - *Non-repudiation* - Prevents a principle from denying they performed an action; from help from **trusted third party**

Ciphers

- Algorithm that obfuscates info so that it seems random to anyone who does not possess special info (**key**)
  - Based on **trapdoor one-way** functions
    - one-way - computational-easy encoding, but difficult decoding
      - never proven to exist
      - **Factoring** $z = x \cdot y; z = ?$
      - **Discrete log** $z = (x^y \mod m = xxx; z, x, m, y = ?, y = (log_x z) \mod m$
    - trapdoor - decoding becomes easy ONLY when receiver has the **key**

Kerckhoff's Principle

- The security of any given encryption system must depend only on the secrecy of the key **K** and not the secrecy of algorithm
  - #SecurityByObsecurity
  - algorithms can be RE-ed and hard to change, compiled into software & wired into circuits
  - Mifare

Shift Ciphers

- Substitution ciphers
- Easily broken with **cryptanalysis**
  - Weakness - every letter in PT always gets encrypted to the **same** letter in CT
  - Attacker can perform **frequency analysis** on the CT to identify and decode common letters, then match against common English words to recover PT and thus the key
  - Doesn't hide frequency info b/c every PT letter always encrypts to same CT letter
  - Use **Polyalphabetic Cipher**/Periodic ciphers
    - Attacker needs to guess the period (N=15); harder for large N
    - Enigma Machine

Alphabet: ABCDE

Key #1: BEDAC
Key #2: ACBDE
Key #3: DACBE

E("BED") = EEB
E("ABACADA") = BCDDABB

Attacking a cipher

- Target: plain-text corresponding to a cipher-text or the key
  - **Brute-force attack** - try all possible keys
  - **Cryptanalysis** - sample plaintext-cipher-text pairs
    - More PT-CT pairs needed to break a cipher, the stronger it is
    - Pick PT and get corresponding CT, adaptively select PT to help break cipher
      - In English, letter E appears 13% of the time
      - In CT, if letter X appears 13% of the time, fair guess it's E

One-Time Pad / Vernam Cipher

- Special type of polyalphabetic cipher that never repeats
  - random substitution for every character
  - key is same length as the message encrypted; CT created by XOR of PT and key
  - CT is PT with randomly flipped bits
- Theoretically unbreakable
  - CT only attack - impossible to break
  - Known PT attack - weak; just XOR CT with PT to reveal the key; so key is not supposed to repeat
  - Chosen CT-PT - weak
- Disadvantages....
  - 100% overhead
  - each key can only be used **ONCE**
  - key must be sent separately
  - cipher is **malleable** - bit flip in CT flips only 1 bit in PT, needs integrity check to avoid tampering

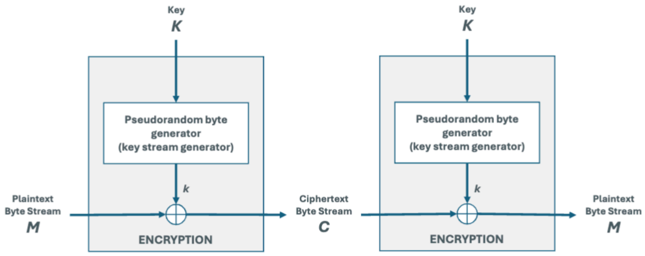
Stream Ciphers & Block Ciphers

Practical Ciphers

1. Fixed length keys, shorter than message
2. Efficient to encrypt and decrypt
3. one-way trapdoor (computationally difficult to decrypt; but CPU capacity evolve, difficulty shift)
4. *2 types of ciphers: Symmetric key and Public (asymmetric key)*

Symmetric Key Ciphers

- Stream Ciphers - **simple, fast, more performant**
  - Similar to OTP; a key is used to generate pseudo-random sequence of bits
  - PT encrypted 1 bit a time, useful for *streaming applications*
  - Suffer from synchronization problems; if bits are lost, entire system may be corrupted



- Block Ciphers - **more common**
  - Encrypt a block of PT at a time (64bits or multiple)
  - PT is divided into blocks and each is encrypted separately (last block might need to be padded)