CS2107

Introduction to Information Security

AY2022/23 Semester 2

Notes by Jonathan Tay

Last updated on January 12, 2023

Contents

Part I

Encryption

Symmetric-key encryption uses the same key k for encryption and decryption.

Any plaintext x that is encrypted to produce a ciphertext $c = E_k(x)$ must also be decryptable to recover the plaintext $D_k(E_k(x)) = x$ — (**correctness**).

It must also be difficult or impossible to derive useful information of the key or the plaintext from the ciphertext — (**security**).

attack models: information

Given access to an **oracle** with varying levels of information, adversaries can conduct several models of attack:

- **ciphertext only (COA)**:
  only ciphertexts and properties of the plaintext are known
- **known-plaintext (KPA)**:
  ciphertexts of corresponding plaintexts are known
- **chosen-plaintext (CPA)**:
  ciphertexts can be derived from arbitrary plaintexts
- **chosen-ciphertext (CCA2)**:
  plaintexts can be derived from arbitrary ciphertexts

attack models: goals

Adversaries may have varying goals:

- **total break**: find the encryption key
- **partial break**: decrypt a ciphertext or obtain information about a plaintext
- **indistinguishability**: distinguish ciphertexts of different plaintexts better than random chance

1 Classical Ciphers

substitution cipher

Construct a **substitution table** used as the key from some permutation of all possible characters used in the plaintext.

Replace each character in the plaintext with the corresponding character in the substitution table to produce the ciphertext.

Substitution ciphers are vulnerable to various attack models:

- **general**: brute-force exhaustive search on all keys
- **KPA**: substitution table reconstruction by comparison of characters
- **COA**: frequency analysis of characters between ciphertext and plaintext

permutation ciphers

Group characters of the plaintext into blocks of equal size, then rearrange the characters in each block according to a permutation key.

Extremely vulnerable under KPA and COA attacks as the permutation key can be easily reconstructed, especially if the plaintext language is known.

Note that applying substitution or permutation ciphers several times alone in succession is equivalent to applying it once — the ciphertext is not any more secure!

one-time pad

Generate a key of equal bit length as the plaintext, then XOR them together to encrypt. XOR the ciphertext with the key to decrypt:

correctness:

1. c = x XOR k
2. (x XOR k) XOR k = x XOR (k XOR k) = x XOR 0 = x

security: Even if an adversary obtains a ciphertext-plaintext pair and derives the key (trivial), the key is useless as it is not re-used.

Exhaustive search does not work on OTPs because no meaningful information about the plaintext can be learnt (**perfect secrecy**) — every decrypted plaintext is equally probable.