Computer Security
CS 426
Lecture 3

Cryptography: One-time Pad, Information Theoretical Security, and Stream CIphers

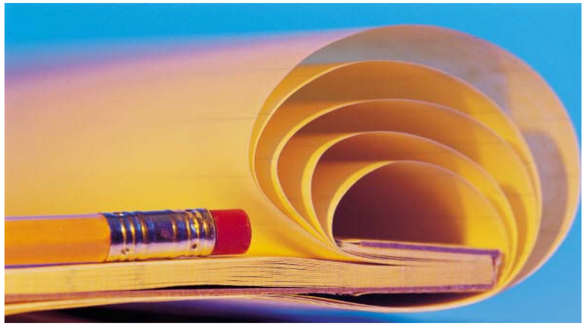
One-Time Pad

- Fix the vulnerability of the mono-alphabetical substitution cipher by encrypting letters in different locations differently

- Key is a random string that is at least as long as the plaintext
- Encryption is similar to shift cipher
- Invented by Vernam in the 1920s

One-Time Pad

Let Z_m ={0,1,…,m-1} be the alphabet.



Plaintext space = Ciphtertext space = Key space = $(Z_m)^n$

The key is chosen uniformly randomly

Plaintext $X = (x_1\ x_2\ \dots\ x_n)$

Key $K = (k_1\ k_2\ \dots\ k_n)$

Ciphertext $Y = (y_1\ y_2\ \dots\ y_n)$

$e_k(X) = (x_1+k_1\ \ x_2+k_2\ \dots\ x_n+k_n) \bmod m$

$d_k(Y) = (y_1-k_1\ \ y_2-k_2\ \dots\ y_n-k_n) \bmod m$

How Good is One-Time Pad?

- Intuitively, it is secure …
- The key is random, so the ciphertext is completely random

Shannon (Information-Theoretic) Security

- Basic Idea: Ciphertext should reveal no "information" about Plaintext
  - More precisely, given any ciphertext, each plaintext (of the same length) is equally likely
  - When keys are uniformly chosen in a cipher, the cipher has Shannon security iff. the number of keys encrypting M to C to be the same for any (M,C)
- We also say such a scheme has perfect secrecy.
  - secure even if the adversary has unbounded computation resources
- One-time pad has perfect secrecy (Proof?)

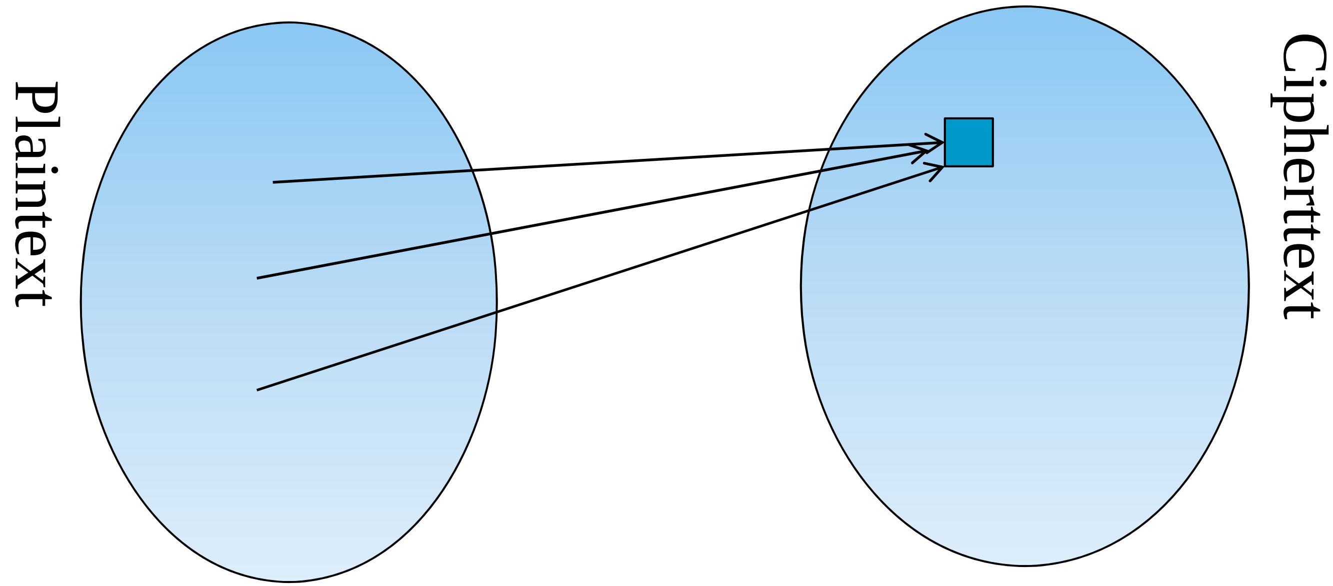
Key Randomness in One-Time Pad

- One-Time Pad uses a very long key, what if the key is not chosen randomly, instead, texts from, e.g., a book are used as keys.
  - this is not One-Time Pad anymore
  - this does not have perfect secrecy
  - this can be broken
  - How?
- The key in One-Time Pad should never be reused.
  - If it is reused, it is Two-Time Pad, and is insecure!
  - Why?

The "Bad News" Theorem for Perfect Secrecy

- Perfect secrecy $\Rightarrow$ key-length $\geq$ msg-length

Proof:



- Difficult to use in practice
- Why is this still useful, even though difficult?

The Binary Version of One-Time Pad

Plaintext space = Ciphtertext space =

Keyspace = $\{0,1\}^n$

Key is chosen randomly

For example:

- Plaintext is 11011011
- Key is 01101001
- Then ciphertext is 10110010

Bit Operators

- Bit AND

  $0 \wedge 0 = 0$ $0 \wedge 1 = 0$ $1 \wedge 0 = 0$ $1 \wedge 1 = 1$

- Bit OR

  $0 \vee 0 = 0$ $0 \vee 1 = 1$ $1 \vee 0 = 1$ $1 \vee 1 = 1$

- Addition mod 2 (also known as Bit XOR)

  $0 \oplus 0 = 0$ $0 \oplus 1 = 1$ $1 \oplus 0 = 1$ $1 \oplus 1 = 0$

- Can we use operators other than Bit XOR for binary version of One-Time Pad?

Stream Ciphers

- In One-Time Pad, a key is a random string of length at least the same as the message
- Stream ciphers:
  - Idea: replace “rand” by “pseudo rand”
  - Use Pseudo Random Number Generator
  - PRNG: $\{0,1\}^s \rightarrow \{0,1\}^n$
    - expand a short (e.g., 128-bit) random seed into a long (e.g., 10^6 bit) string that “looks random”
  - Secret key is the seed
  - $E_{key}[M] = M \oplus PRNG(key)$

The RC4 Stream Cipher

- A proprietary cipher owned by RSA, designed by Ron Rivest in 1987.
- Became public in 1994.
- Simple and effective design.
- Variable key size (typical 40 to 256 bits),
- Output unbounded number of bytes.
- Widely used (web SSL/TLS, wireless WEP).
- Extensively studied, not a completely secure PRNG, but no known serious security flaw as a stream cipher

Pseudo Random Number Generator

- Useful for cryptography and for simulation
- The same seed gives the same output stream
  - why is this necessary for stream ciphers?
- Simulation requires uniform distributed sequences
- **Cryptographically secure pseudo-random number generator** requires unpredictable sequences
  - satisfies the "next-bit test“: given consecutive sequence of bits output (but not seed), next bit must be hard to predict
  - withstands “state compromise extensions”: given sequences from bits k+1 on, should be difficult to predict earlier bits
- Also useful for generating temporary keys, etc.

Properties of Stream Ciphers

- Typical stream ciphers are very fast
- Widely used, often incorrectly
  - Content Scrambling System (uses Linear Feedback Shift Registers incorrectly),
  - Wired Equivalent Privacy (uses RC4 incorrectly)
  - SSL (uses RC4, SSLv3 has no known major flaw)
- Are they secure?

Adversarial Models for Ciphers

- The language of the plaintext and the nature of the cipher are assumed to be known to the adversary.
- **Ciphertext-only attack**: The adversary knows only a number of ciphertexts.
- **Known-plaintext attack**: The adversary knows some pairs of ciphertext and corresponding plaintext.
- **Chosen-plaintext attack:** The adversary can choose a number of messages and obtain the ciphertexts
- **Chosen-ciphertext attack:** The adversary can choose a number of ciphertexts and obtain the plaintexts.

When would these attacks be relevant in wireless communications?

Security Properties of Stream Ciphers

- Under known plaintext, chosen plaintext, or chosen ciphertext, the adversary knows key stream (i.e., PRNG(key))
  - Security depends on PRNG
  - PRNG must be “unpredictable”

- Does not have perfect secrecy

- How to break a stream cipher?

Computational Security vs. Information Theoretical Security

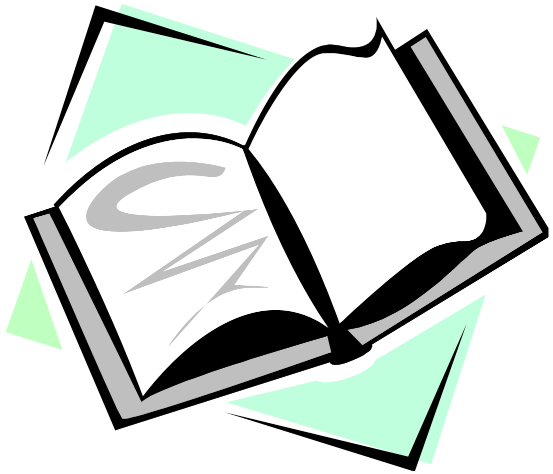
- If only having computational security, then can be broken by a brute force attack, e.g., enumerating all possible keys
  - Weak algorithms can be broken with much less time
- How to prove computational security?
  - Assume that some problems are hard (requires a lot of computational resources to solve), then show that breaking security means solving the problem
- Computational security is foundation of modern cryptography.

Real Weaknesses of Stream Ciphers

- If the same stream is used twice ever, then easy to break.
- Highly malleable
  - easy to change ciphertext so that plaintext changes in predictable, e.g., flip bits
  - which of the three properties is violated here?
- These are fundamental weaknesses of stream ciphers; they exist even if the PRNG is strong

Readings for This Lecture

- Required reading from wikipedia
  - One-Time Pad
  - Information theoretic security
  - Stream cipher
  - Pseudorandom number generator



Coming Attractions …

- Cryptography: Block ciphers, encryption modes, cryptographic functions