

# CS355: Cryptography

Lecture 2: Attacks against symmetric ciphers; models to evaluate security.

# Secret-key vs. Public-key Cryptography

---

- Secret-key cryptography (a.k.a. symmetric cryptography)
  - encryption & decryption use the same key
  - key must be kept secret
  - key distribution is very difficult
- Public-key cryptography (a.k.a. asymmetric cryptography)
  - encryption key different from decryption key
  - cannot derive decryption key from encryption key
  - higher cost than symmetric cryptography

# Some Goals of Modern Cryptography

---

- Pseudo-random number generation
- Non-repudiation: Digital signatures
- Zero-knowledge proof
- E-voting
- Secret sharing

# Example: Cellular Networks Authentication

---

- Focus:
  - Make sure the client is billed for the service
  - **Provide authentication, confidentiality and anonymity of the communication**
- Assumptions
  - There is a long-term relationship between the client and the network operator (home network) in the form of a contract
  - The long-term relationship is represented by a long-term secret key shared by the client and network and serving as basis for identification

# Cellular Networks Authentication

---

- SIM (Subscriber Identity Module): secret PIN (personal identification number) and the long term secret key
- Storing the key on the SIM allows the portability of the service from one phone to another
- Authentication is based on a challenge response protocol - this is where cryptography plays its role

# A Symmetric Cipher

---

- A Cipher ( $K$ ,  $P$ ,  $C$ ,  $E$ ,  $D$ )
  - $K$ : the key space
  - $P$ : the plaintext space
  - $C$ : the ciphertext space
  - $E: K \times P \rightarrow C$ : the encryption function
  - $D: K \times C \rightarrow P$ : the decryption function
    - Given a key  $K$  and a plaintext  $P$ ,  
 $D(K, E(K, P)) = P$

# Rules of the Game

---

1. Overcome the adversary only by means of protocols
2. Protocol designs are made public, only keys are secret (Kerckhoff's principle 1883)
  - **security by obscurity does not work**  
(there are many examples, WEP, voting machines)

# How Do You Know a Cipher is Secure?

---

- Show that under the considered attack model, security goals are NOT achieved (break it)
- Show that under the considered attack model, security goals are achieved (evaluate/prove)

# Breaking Ciphers...

---

- There are different methods of breaking a cipher, depending on:
  - the type of information available to the attacker
  - the interaction with the cipher machine
  - the computational power available to the attacker



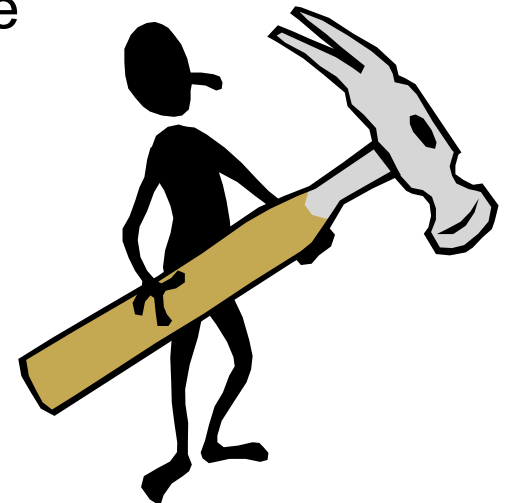
# Breaking Ciphers...

---

- **Ciphertext-only attack:**

- The cryptanalyst knows **only the ciphertext**. Sometimes the language of the plaintext and the cipher are also known.
- The goal is to find the plaintext and the key.

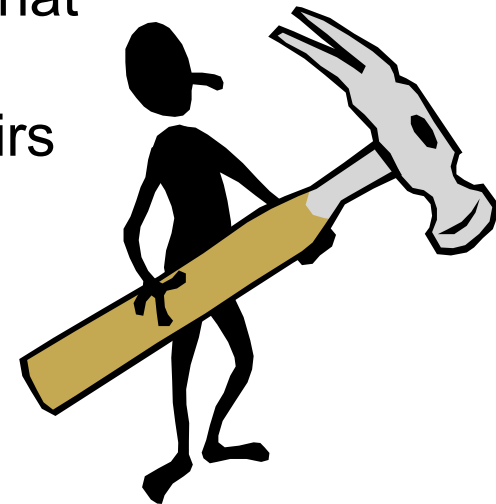
- **NOTE: any encryption scheme vulnerable to this type of attack is considered to be completely insecure.**



# Breaking Ciphers (2)

---

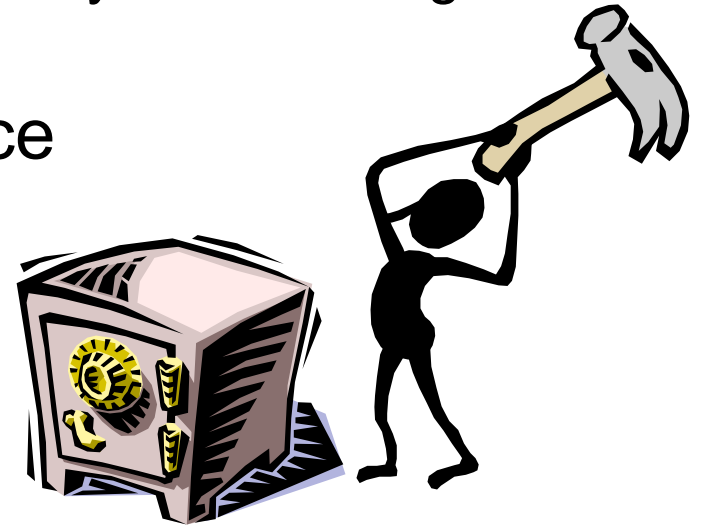
- **Known-plaintext attack:**
  - The cryptanalyst knows **one or several pairs of ciphertext and the corresponding plaintext.**
  - The goal is to find the key used to encrypt these messages or a way to decrypt any new messages that use that key.
  - How does the cryptanalyst get the pairs of ciphertext and plaintext?



# Breaking Ciphers (3)

---

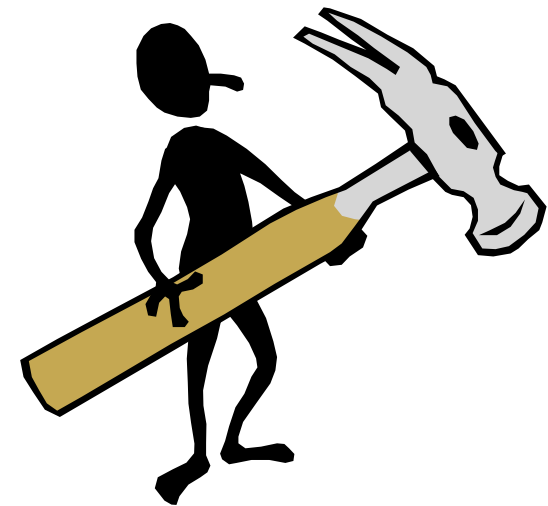
- **Chosen-plaintext attack**
  - The cryptanalyst **can choose a number of messages and obtain the ciphertexts for them**
  - The goal is to deduce the key used in the other encrypted messages or decrypt any new messages using that key.
- It can be **adaptive**, the choice of plaintext depends on the ciphertext received from previous requests.



# Breaking Ciphers (4)

---

- **Chosen-ciphertext attack**  
Similar to the chosen-plaintext attack, but the cryptanalyst **can choose a number of ciphertexts and obtain the plaintexts.**
- It can also be **adaptive** The choice of ciphertext may depend on the plaintext received from previous requests.



# How Do You Know a Cipher is Secure?

---

- Show that under the considered attack model, security goals are NOT achieved (break it)
- Show that under the considered attack model, security goals are achieved (evaluate/prove)

# Models for Evaluating Security

---

- **Unconditional (information-theoretic) security**
  - **Assumes that the adversary has unlimited computational resources.**
  - Plaintext and ciphertext modeled by their distribution
  - Analysis is made by using probability theory.
  - For encryption systems: **perfect secrecy**, observation of the ciphertext provides no information to an adversary.

# Models for Evaluating Security (2)

---

- **Provable security:**
  - Prove security properties based on assumptions that it is difficult to solve a well-known and supposedly difficult problem (example: computation of discrete logarithms, factoring).

# Models for Evaluating Security (3)

---

- **Computational security (practical security)**
  - Measures the amount of computational effort required to defeat a system using the best-known attacks.
  - Sometimes related to the hard problems, but no proof of equivalence is known.

# Models for Evaluating Security (4)

---

- **Ad hoc security (heuristic security):**
  - Variety of convincing arguments that every successful attack requires more resources than the ones available to an attacker.
  - Unforeseen attacks remain a threat.
  - **THIS IS NOT A PROOF**