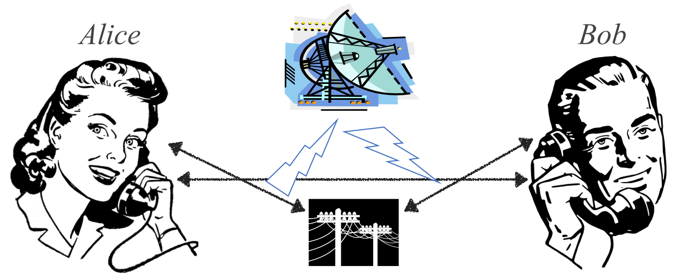
Cybersecurity: Introduction to Cryptography

Ozalp Babaoglu

ALMA MATER STUDIORUM – UNIVERSITA' DI BOLOGNA

Historic Motivation for Cryptography



Confidentiality: Private communication in a public environment (infrastructure)

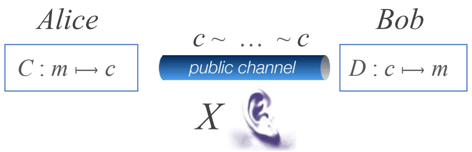
Goals

- Learn what problems can (and cannot) be solved using cryptography
- Become convinced that:
  - Using cryptography requires building a substantial (but easily overlooked) infrastructure
  - Designing a good crypto system is extremely difficult
  - Wide-spread use of cryptography requires overcoming legal and social barriers

Terminology

- *Plaintext*: the original message or data to be communicated that is fed as input to the encryption algorithm
- *Encryption algorithm (cipher)*: transforms the plaintext to make it unintelligible (private)
- *Secret key*: second input to the encryption algorithm that determines the exact transformations performed by the algorithm on the plaintext
- *Ciphertext*: transformed version of the plaintext produced as output of the encryption algorithm
- *Decryption algorithm*: inverse of the encryption algorithm that takes the ciphertext and the same secret key to reproduce the original plaintext

Communication Scenario



- $Alice\ (A)$: sender
- $Bob\ (B)$: receiver
- m : plaintext message
- C : encryption algorithm
- c : ciphertext
- D : decryption algorithm
- X : cryptanalyst

More Terminology

- ***Cryptography***: the design of secure and efficient ciphers (and their inverses)
- ***Cryptanalysis***: the process of attempting to discover the plaintext or key. The strategy used by the cryptanalyst depends on the nature of the encryption scheme and the available information
- ***Cryptology***: Cryptography + Cryptanalysis

Two Families of Cryptography

- ***Secret/Private-key*** (*symmetric*) Cryptography:
  - The keys used for encrypting and decrypting are the same (therefore must be shared by the sender and receiver)
  - The encryption and decryption functions C are D are often the same
- ***Public-key*** (*asymmetric*) Cryptography:
  - Different keys are used for encrypting and for decrypting
  - The encryption and decryption functions C are D are different

Cryptanalyst X

- Motives: curiosity, espionage, malice, …
- Roles
  - ***Passive***: is limited to eavesdropping (listening and recording)
  - ***Active***: can interject into the conversation or modify it
- What does the cryptanalyst know?
  - Encryption/Decryption algorithms (no “security through obscurity”)
  - All information that can be collected from observing past communication between A and B

Types of attack

- What the cryptanalyst knows in addition to the encryption/decryption algorithm determines the type of attack:
  - ***Brute-force attack***: nothing (other than the algorithms)
  - ***Ciphertext attack***: ciphertext as a collection of $c_1,..., c_n$
  - ***Known plaintext attack***: ciphertext plus one or more pairs (m_i, c_i)
  - ***Chosen plaintext attack***: ciphertext plus one or more pairs (m_i, c_i) where m_i is chosen by the cryptanalyst

Definitions, Notation

- Encryption function
  - $C_k(m) = c$ "encryption of m with key k"
- Decryption function
  - $D_k(c) = m$ "decryption of c with key k"
- D_k is the *mathematical inverse* of C_k:
  - $D_k(C_k(m)) = m$
  - Sometimes we require that they be also *commutative*

$$D_k(C_k(m)) = C_k(D_k(m)) = m$$

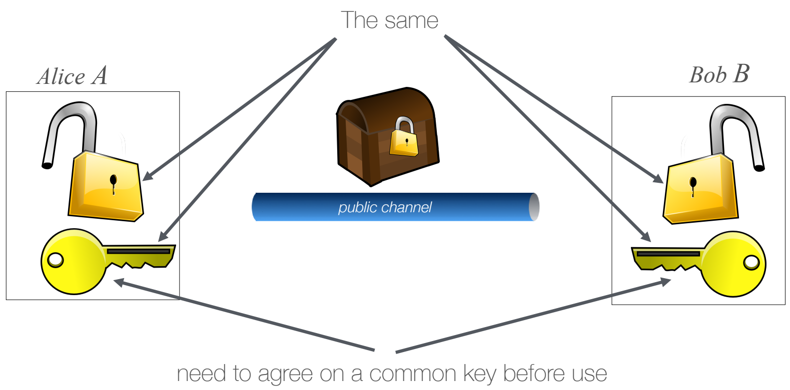
Can we build a "perfect" cipher?

- A "perfect" cipher would guarantee secrecy (confidentiality) always
- Need to distinguish between
  - ***Perfect secrecy***: Confidentiality always guaranteed
  - ***Computational secrecy***: Confidentiality guaranteed only if we limit the resources available to the adversary (cryptanalyst)
- "Resources" can be computing power, time, memory, communication bandwidth, etc.

Can we build a "perfect" cipher?

- As long as information (messages) are composed from a finite alphabet and have finite length (n bits), perfect secrecy is impossible to achieve
- Brute force method — for each letter in the ciphertext, the adversary can always "guess" the corresponding plaintext letter to see if the resulting text "makes sense"
- Thus, ***perfect secrecy*** is not possible
- The best we can aim for is ***computational secrecy***: can we encrypt information such that confidentiality is guaranteed in the presence of an adversary with limited resources?

Secret-Key (Symmetric) Cryptography
A Metaphor



Secret-Key (Symmetric) Cryptography

- The two parties must share a common secret key before use
  - Agreeing on the common key cannot use the public channel
  - Has to be based on an "out-of-band" communication channel that is used only once
  - The "out-of-band" channel can be slow and expensive
  - The high cost of the "out-of-band" channel can be amortized over many subsequent exchanges using public channel
- The secret key space must be very large:
  - To prevent "brute-force attacks" (try all possible keys)
  - Necessary but not sufficient

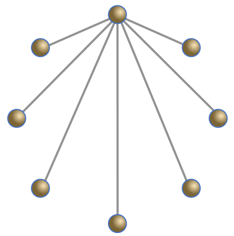
Secret-Key Ciphers

- Some well-know "secret key" (symmetric) ciphers:
  - *Data Encryption Standard* (DES)
  - *Triple-DES*
  - *Blowfish*
  - *International Data Encryption Algorithm* (IDEA)
  - *Advanced Encryption Standard* (AES)

Secret-Key Ciphers

- The functions for encrypting and decrypting are interchangeable
- Sender and receiver:
  - Both know a common secret key k
  - Both can encrypt or decrypt
  - Both promise to keep k a secret
- What are the defects of secret-key cryptography?
  - Requires a common secret key
  - For confidential communication between $O(n)$ parties, requires $O(n^2)$ secret keys

Secret-Key Cryptography



Each pair must have a *different* secret key

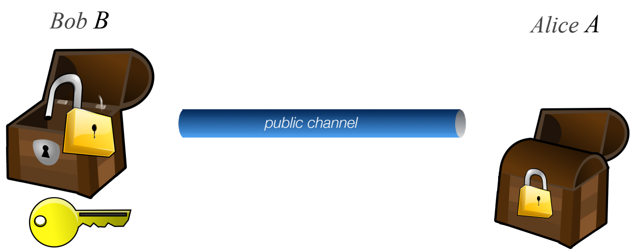
- Consider n parties
- For any one receiver, there can be $(n-1)$ senders
  - thus we need $(n-1)$ secret keys
- For n possible receivers and n possible senders
  - we need $n(n-1)/2$ which is $O(n^2)$ secret keys

Public-Key (Asymmetric) Cryptography

- Is it possible to exchange information confidentially without having to first agree on a key?
- The year 1976 is a turning point! (Diffie-Hellman and Merkle)
- Huge consequences:
  - from a theoretical point of view
  - from an economical point of view (basis of modern e-commerce on the Internet)

Public-Key (Asymmetric) Cryptography A Metaphor

- Is it possible to exchange information confidentially without having to first agree on a key?



Asymmetric or Public-Key Cryptography

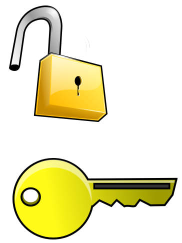
- Goal: break the symmetry between encrypting and decrypting
- Who knows how to encrypt must ***not*** know how to decrypt
- The key k divided in two parts $k[priv]$, $k[pub]$:
  - The key k generated by the destination
  - $k[priv]$ kept secret by the destination and used to decrypt
  - $k[pub]$ made public by the destination and used by *everyone* to encrypt messages for the destination
  - It must be (computationally) difficult to go from $k[pub]$ to $k[priv]$
- Concept of "one-way trap-door" function

One-way Trapdoor Functions

- A function is “One-way Trapdoor” if it is
  - easy to compute,
  - difficult to invert in general,
  - easy to invert if you know some additional information

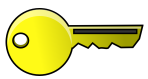
One-way Trapdoor Functions

- A lock that is open:
  - is easy to close,
  - difficult to open,
  - unless you have the key



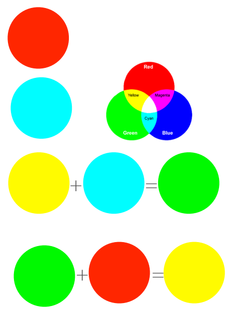
One-way Trapdoor Functions

- A sidewalk mailbox:
  - easy to drop a letter,
  - difficult to remove a letter,
  - unless you have the key



One-way Trapdoor Function Physical Metaphor

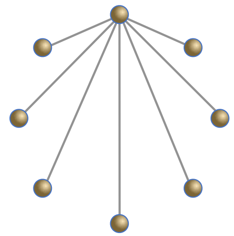
- Alice picks a random color (keeps it private)
- Alice determines its complement color and sends it to Bob
- Bob “encodes” his message using the complement color and sends the result to Alice
- Alice “decodes” the message by mixing her private color to the color received from Bob



One-way Trapdoor Functions

- Let p and q be two prime numbers:
  - given p and q, it is easy to compute $n=pq$
  - given n, it is difficult to compute p (or q)
  - unless you know q ($p=n/q$)

Public-Key Cryptography



- Consider n parties
- For any one receiver, there can be $(n-1)$ senders
  - but they all use the ***same*** public key of the receiver
- For n possible senders and n possible receivers, it suffices to have just n public keys (and n private keys)

Properties of Modern Cryptography

- For modern cryptography, ***confidentiality*** is not the only property that is required
- Modern uses of cryptography need three additional properties:
  - *Integrity*: The receiver must be able to determine if the received message has been tampered with (modified, replaced)
  - *Authentication*: The receiver must be able to ascertain that the message was sent by the presumed sender
  - *Non-repudiation*: The sender must not be able to refute having sent the message. The receiver must be able to convince a third party (judge) that the received message was indeed sent by the sender (obtained through *digital signatures*)