I – Introduction

David Pointcheval
Ecole normale supérieure/PSL, CNRS & INRIA

Outline

Cryptography

Outline

Old Encryption Mechanisms

Substitutions and permutations

Security relies on
the secrecy of the mechanism

Old Encryption Mechanisms

Scytale - Permutation

Substitutions and permutations

Security relies on

the secrecy of the mechanism

Old Encryption Mechanisms

Scytale - Permutation

Alberti's disk
Mono-alphabetical Substitution

Substitutions and permutations

Security relies on
the secrecy of the mechanism

Old Encryption Mechanisms

Scytale - Permutation

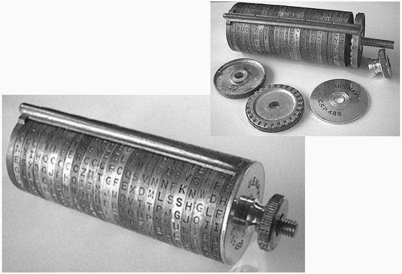
Substitutions and permutations

Security relies on

the secrecy of the mechanism

Alberti's disk

Mono-alphabetical Substitution



Wheel – M 94 (CSP 488)

Poly-alphabetical Substitution

Scytale - Permutation

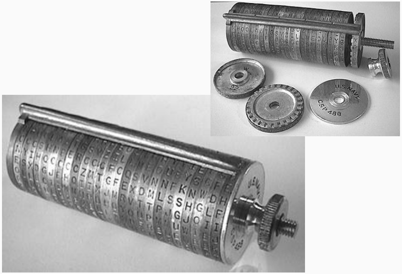
Substitutions and permutations

Security relies on

the secrecy of the mechanism

Alberti's disk

Mono-alphabetical Substitution



Wheel – M 94 (CSP 488)

Poly-alphabetical Substitution

Outline

Cryptography

Introduction

Kerckhoffs' Principles

Cryptographic Primitives

Symmetric Cryptography

Asymmetric Cryptography

Kerckhoffs' Principles (1)

La Cryptographie Militaire (1883)

Le système doit être matèriellement,
sinon mathématiquement, indéchiffrable

The system should be, if not theoretically unbreakable,
unbreakable in practice

$\longrightarrow$ If the security cannot be formally proven,
heuristics should provide some confidence.

Kerckhoffs' Principles (1)

La Cryptographie Militaire (1883)

Le système doit être matèriellement,
sinon mathématiquement, indéchiffrable

The system should be, if not theoretically unbreakable,
unbreakable in practice

$\longrightarrow$ If the security cannot be formally proven,
heuristics should provide some confidence.

Kerckhoffs' Principles (2)

La Cryptographie Militaire (1883)

Il faut qu'il n'exige pas le secret, et qu'il puisse sans inconvénient tomber entre les mains de l'ennemi

Compromise of the system should not inconvenience the correspondents

$\longrightarrow$ The description of the mechanism should be public

Kerckhoffs' Principles (2)

La Cryptographie Militaire (1883)

Il faut qu'il n'exige pas le secret, et qu'il puisse sans inconvénient tomber entre les mains de l'ennemi

Compromise of the system should not inconvenience the correspondents

$\longrightarrow$ The description of the mechanism should be public

Kerckhoffs' Principles (3)

La Cryptographie Militaire (1883)

La clef doit pouvoir en être communiquée et retenue sans le secours de notes écrites, et être changée ou modifiée au gré des correspondants

The key should be rememberable without notes and should be easily changeable

$\longrightarrow$ The parameters specific to the users (the key) should be short

Kerckhoffs' Principles (3)

La Cryptographie Militaire (1883)

La clef doit pouvoir en être communiquée et retenue sans le secours de notes écrites, et être changée ou modifiée au gré des correspondants

The key should be rememberable without notes and should be easily changeable

$\longrightarrow$ The parameters specific to the users (the key) should be short

Use of (Secret) Key

A shared information (secret key) between the sender and the receiver parameterizes the mechanism:

- Vigenère: each key letter tells the shift
- Enigma: connectors and rotors

Use of (Secret) Key

A shared information (secret key) between the sender and the receiver parameterizes the mechanism:

- Vigenère: each key letter tells the shift
- Enigma: connectors and rotors

Use of (Secret) Key

A shared information (secret key) between the sender and the receiver parameterizes the mechanism:

- Vigenère: each key letter tells the shift
- Enigma: connectors and rotors

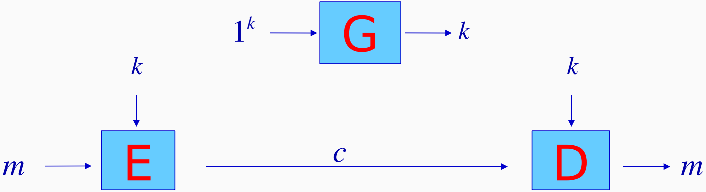
Use of (Secret) Key

A shared information (secret key) between the sender and the receiver parameterizes the mechanism:

- Vigenère: each key letter tells the shift
- Enigma: connectors and rotors

Security **looks** better: but broken (Alan Turing *et al.*)

Principles 2 and 3 define the concepts of symmetric cryptography:



Secrecy

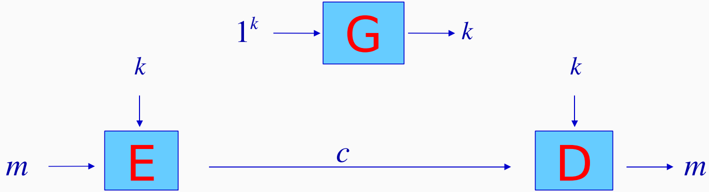
It is impossible/hard to recover m from c only (without k)

Security

It is heuristic only: 1st principle

Symmetric Encryption

Principles 2 and 3 define the concepts of symmetric cryptography:



Secrecy

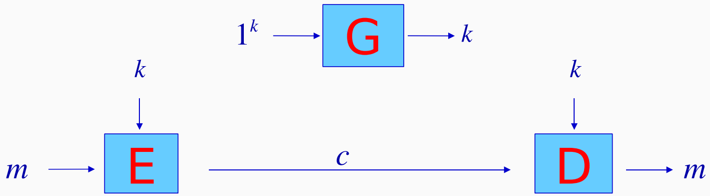
It is impossible/hard to recover m from c only (without k)

Security

It is heuristic only: 1st principle

Symmetric Encryption

Principles 2 and 3 define the concepts of symmetric cryptography:



Secrecy

It is impossible/hard to recover m from c only (without k)

Security

It is heuristic only: 1st principle

Perfect Secrecy?

Any security indeed vanished with statistical attacks!

Perfect Secrecy?

Any security indeed vanished with statistical attacks!

Perfect secrecy? Is it possible?

Perfect Secrecy?

Any security indeed vanished with statistical attacks!

Perfect secrecy? Is it possible?

Perfect Secrecy

The ciphertext does not reveal any (additional) information about the plaintext: no more than known before

- a priori information about the plaintext, defined by the distribution probability of the plaintext
- a posteriori information about the plaintext, defined by the distribution probability of the plaintext, given the ciphertext

Both distributions should be perfectly identical

One-Time Pad Encryption

Vernam's Cipher (1929)

- Encryption of $m \in \{0,1\}^n$ under the key $k \in \{0,1\}^n$:

$$
\begin{array}{rcl}
m = & \boxed{1 \quad 0 \quad 0 \quad 1 \quad 0 \quad 1 \quad 1} & \text{plaintext} \\
 & \oplus & \text{XOR } (+ \text{ modulo } 2) \\
k = & \boxed{1 \quad 1 \quad 0 \quad 1 \quad 0 \quad 0 \quad 0} & \text{key} = \text{random mask} \\
 & = & \\
c = & \boxed{0 \quad 1 \quad 0 \quad 0 \quad 0 \quad 1 \quad 1} & \text{ciphertext}
\end{array}
$$

- Decryption of $c \in \{0,1\}^n$ under the key $k \in \{0,1\}^n$:

$$c \oplus k = (m \oplus k) \oplus k = m \oplus (k \oplus k) = m$$

One-Time Pad Encryption

Vernam's Cipher (1929)

- Encryption of $m \in \{0,1\}^n$ under the key $k \in \{0,1\}^n$:

$$
\begin{array}{rcl}
m = & \boxed{1 \quad 0 \quad 0 \quad 1 \quad 0 \quad 1 \quad 1} & \text{plaintext} \\
& \oplus & \text{XOR } (+ \text{ modulo } 2) \\
k = & \boxed{1 \quad 1 \quad 0 \quad 1 \quad 0 \quad 0 \quad 0} & \text{key} = \text{random mask} \\
& = & \\
c = & \boxed{0 \quad 1 \quad 0 \quad 0 \quad 0 \quad 1 \quad 1} & \text{ciphertext}
\end{array}
$$

- Decryption of $c \in \{0,1\}^n$ under the key $k \in \{0,1\}^n$:

$$c \oplus k = (m \oplus k) \oplus k = m \oplus (k \oplus k) = m$$

One-Time Pad Encryption

Vernam's Cipher (1929)

- Encryption of $m \in \{0,1\}^n$ under the key $k \in \{0,1\}^n$:

$$
\begin{array}{rcl}
m = & \boxed{1 \quad 0 \quad 0 \quad 1 \quad 0 \quad 1 \quad 1} & \text{plaintext} \\
 & \oplus & \text{XOR } (+ \text{ modulo } 2) \\
k = & \boxed{1 \quad 1 \quad 0 \quad 1 \quad 0 \quad 0 \quad 0} & \text{key} = \text{random mask} \\
 & = & \\
c = & \boxed{0 \quad 1 \quad 0 \quad 0 \quad 0 \quad 1 \quad 1} & \text{ciphertext}
\end{array}
$$

- Decryption of $c \in \{0,1\}^n$ under the key $k \in \{0,1\}^n$:

$$c \oplus k = (m \oplus k) \oplus k = m \oplus (k \oplus k) = m$$

One-Time Pad Encryption

Vernam's Cipher (1929)

- Encryption of $m \in \{0,1\}^n$ under the key $k \in \{0,1\}^n$:

$$\begin{array}{rcl} m = & \boxed{1 \quad 0 \quad 0 \quad 1 \quad 0 \quad 1 \quad 1} & \text{plaintext} \\ & \oplus & \text{XOR } (+ \text{ modulo } 2) \\ k = & \boxed{1 \quad 1 \quad 0 \quad 1 \quad 0 \quad 0 \quad 0} & \text{key} = \text{random mask} \\ & = & \\ c = & \boxed{0 \quad 1 \quad 0 \quad 0 \quad 0 \quad 1 \quad 1} & \text{ciphertext} \end{array}$$

- Decryption of $c \in \{0,1\}^n$ under the key $k \in \{0,1\}^n$:

$$c \oplus k = (m \oplus k) \oplus k = m \oplus (k \oplus k) = m$$

Which message is encrypted in the ciphertext $c \in \{0,1\}^n$?

One-Time Pad Encryption

Vernam's Cipher (1929)

- Encryption of $m \in \{0,1\}^n$ under the key $k \in \{0,1\}^n$:

$$
\begin{array}{rcl}
m = & \boxed{1 \quad 0 \quad 0 \quad 1 \quad 0 \quad 1 \quad 1} & \text{plaintext} \\
 & \oplus & \text{XOR } (+ \text{ modulo } 2) \\
k = & \boxed{1 \quad 1 \quad 0 \quad 1 \quad 0 \quad 0 \quad 0} & \text{key} = \text{random mask} \\
 & = & \\
c = & \boxed{0 \quad 1 \quad 0 \quad 0 \quad 0 \quad 1 \quad 1} & \text{ciphertext}
\end{array}
$$

- Decryption of $c \in \{0,1\}^n$ under the key $k \in \{0,1\}^n$:

$$c \oplus k = (m \oplus k) \oplus k = m \oplus (k \oplus k) = m$$

Which message is encrypted in the ciphertext $c \in \{0,1\}^n$?

For any candidate $m \in \{0,1\}^n$, the key $k = c \oplus m$ would lead to c

One-Time Pad Encryption

Vernam's Cipher (1929)

- Encryption of $m \in \{0,1\}^n$ under the key $k \in \{0,1\}^n$:

$$
\begin{array}{rcl}
m = & \boxed{1 \quad 0 \quad 0 \quad 1 \quad 0 \quad 1 \quad 1} & \text{plaintext} \\
& \oplus & \text{XOR } (+ \text{ modulo } 2) \\
k = & \boxed{1 \quad 1 \quad 0 \quad 1 \quad 0 \quad 0 \quad 0} & \text{key} = \text{random mask} \\
& = & \\
c = & \boxed{0 \quad 1 \quad 0 \quad 0 \quad 0 \quad 1 \quad 1} & \text{ciphertext}
\end{array}
$$

- Decryption of $c \in \{0,1\}^n$ under the key $k \in \{0,1\}^n$:

$$c \oplus k = (m \oplus k) \oplus k = m \oplus (k \oplus k) = m$$

Which message is encrypted in the ciphertext $c \in \{0,1\}^n$?

For any candidate $m \in \{0,1\}^n$, the key $k = c \oplus m$ would lead to c

$\Rightarrow$ no information about m is leaked with c!

Information Theory

Drawbacks

- The key must be as long as the plaintext
- This key must be used once only (one-time pad)

Theorem (Shannon – 1949)

To achieve perfect secrecy, A and B have to share a common string truly random and as long as the whole communication.

Thus, the above one-time pad technique is optimal. . .

Information Theory

Drawbacks

- The key must be as long as the plaintext
- This key must be used once only (one-time pad)

Theorem (Shannon – 1949)

To achieve perfect secrecy, A and B have to share a common string truly random and as long as the whole communication.

Thus, the above one-time pad technique is optimal...

Information Theory

Drawbacks

- The key must be as long as the plaintext
- This key must be used once only (one-time pad)

Theorem (Shannon – 1949)

To achieve perfect secrecy, A and B have to share a common string truly random and as long as the whole communication.

Thus, the above one-time pad technique is optimal...

Practical Secrecy

Perfect Secrecy vs. Practical Secrecy

- No information about the plaintext m is in the ciphertext c without the knowledge of the key k
  $\Rightarrow$ information theory
  No information about the plaintext m can be extracted from the ciphertext c, even for a powerful adversary (unlimited time and/or unlimited power): perfect secrecy
- In practice: adversaries are limited in time/power
  $\Rightarrow$ complexity theory

Practical Secrecy

Perfect Secrecy vs. Practical Secrecy

- No information about the plaintext m is in the ciphertext c without the knowledge of the key k
  $\Rightarrow$ information theory
  No information about the plaintext m can be extracted from the ciphertext c, even for a powerful adversary (unlimited time and/or unlimited power): perfect secrecy
- In practice: adversaries are limited in time/power
  $\Rightarrow$ complexity theory

Practical Secrecy

Perfect Secrecy vs. Practical Secrecy

- No information about the plaintext m is in the ciphertext c without the knowledge of the key k
  $\Rightarrow$ information theory
  No information about the plaintext m can be extracted from the ciphertext c, even for a powerful adversary (unlimited time and/or unlimited power): perfect secrecy
- In practice: adversaries are limited in time/power
  $\Rightarrow$ complexity theory

Shannon also showed that combining appropriately permutations and substitutions can hide information: extracting information from the ciphertext is time consuming

Modern Symmetric Encryption: DES and AES

Combination of substitutions and permutations

DES (1977)
Data Encryption Standard

AES (2001)
Advanced Encryption Standard

Modern Symmetric Encryption: DES and AES

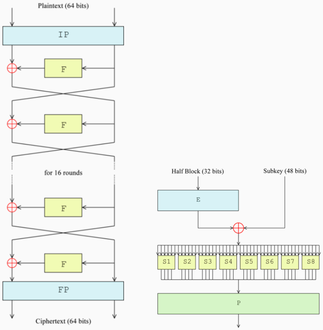
Combination of substitutions and permutations

DES (1977)
Data Encryption Standard

AES (2001)
Advanced Encryption Standard

Modern Symmetric Encryption: DES and AES

Combination of substitutions and permutations



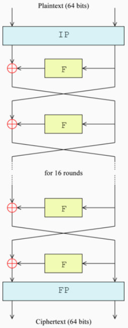
DES (1977)

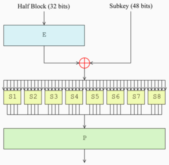
Data Encryption Standard

AES (2001)

Advanced Encryption Standard

Combination of substitutions and permutations





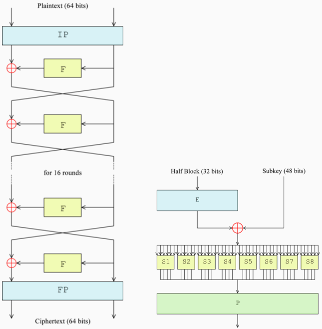
DES (1977)

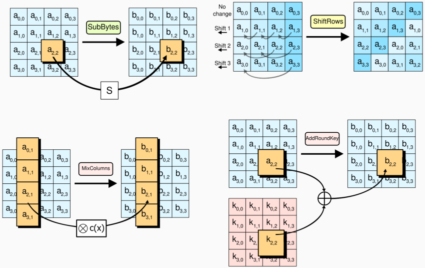
Data Encryption Standard

AES (2001)

Advanced Encryption Standard

Combination of substitutions and permutations





DES (1977)

Data Encryption Standard

AES (2001)

Advanced Encryption Standard

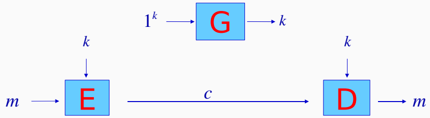
Outline

Symmetric Encryption: Formalism

Symmetric Encryption – Secret Key Encryption

One secret key only shared by Alice and Bob: this is a common parameter for the encryption and the decryption algorithms

This secret key has a symmetric capability

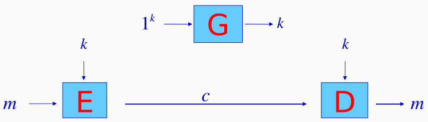


Symmetric Encryption: Formalism

Symmetric Encryption – Secret Key Encryption

One secret key only shared by Alice and Bob: this is a common parameter for the encryption and the decryption algorithms

This secret key has a symmetric capability



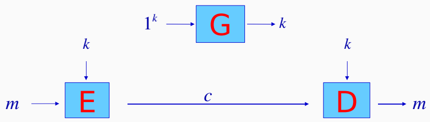
The secrecy of the key k guarantees the secrecy of communications but requires such a common secret key!

Symmetric Encryption: Formalism

Symmetric Encryption – Secret Key Encryption

One secret key only shared by Alice and Bob: this is a common parameter for the encryption and the decryption algorithms

This secret key has a symmetric capability



The secrecy of the key k guarantees the secrecy of communications but requires such a common secret key!

How can we establish such a common secret key?
Or, how to avoid it?

Asymmetric Encryption: Intuition

[Diffie-Hellman 1976]

Secrecy

- The recipient only should be able to open the message
- No requirement about the sender

Why would the sender need a secret key to encrypt a message?

Secrecy

- The recipient only should be able to open the message
- No requirement about the sender

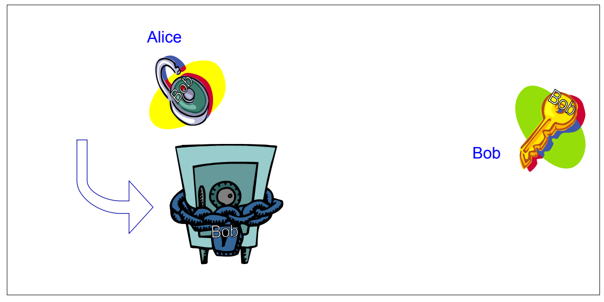
Why would the sender need a secret key to encrypt a message?

Asymmetric Encryption: Intuition [Diffie-Hellman 1976]

Secrecy

- The recipient only should be able to open the message
- No requirement about the sender

Why would the sender need a secret key to encrypt a message?



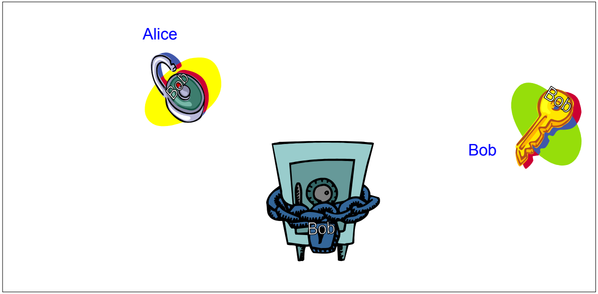
Asymmetric Encryption: Intuition

[Diffie-Hellman 1976]

Secrecy

- The recipient only should be able to open the message
- No requirement about the sender

Why would the sender need a secret key to encrypt a message?



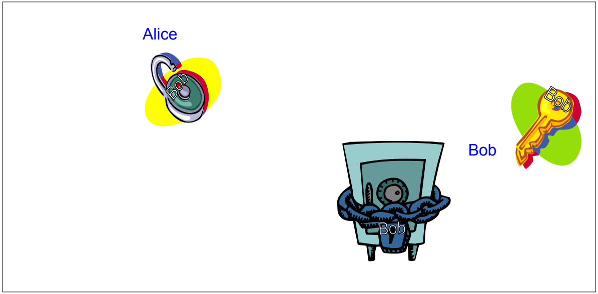
Asymmetric Encryption: Intuition

[Diffie-Hellman 1976]

Secrecy

- The recipient only should be able to open the message
- No requirement about the sender

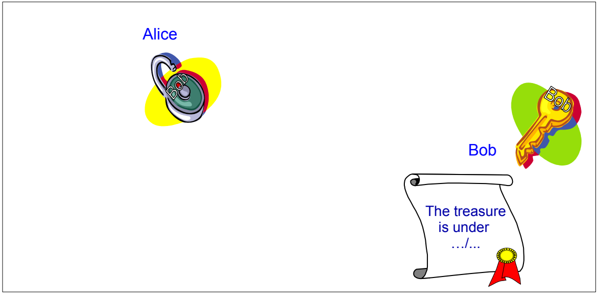
Why would the sender need a secret key to encrypt a message?



Secrecy

- The recipient only should be able to open the message
- No requirement about the sender

Why would the sender need a secret key to encrypt a message?

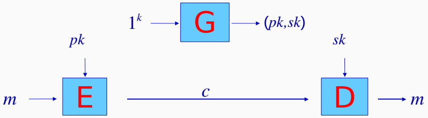


Asymmetric Encryption: Formalism

Public Key Cryptography – Diffie-Hellman (1976)

- Bob's public key is used by Alice as a parameter to encrypt a message to Bob
- Bob's private key is used by Bob as a parameter to decrypt ciphertexts

Asymmetric cryptography extends the 2nd principle:

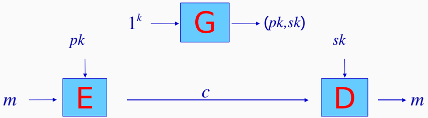


Asymmetric Encryption: Formalism

Public Key Cryptography – Diffie-Hellman (1976)

- Bob's public key is used by Alice as a parameter to encrypt a message to Bob
- Bob's private key is used by Bob as a parameter to decrypt ciphertexts

Asymmetric cryptography extends the 2nd principle:



The secrecy of the private key *sk* guarantees the secrecy of communications

Outline

Main Symmetric Primitives

- Encryption:
  - block-cipher
  - stream-cipher
- Authentication:
  - MAC: Message Authentication Codes
  - AEAD: Authenticated Encryption (with Associated Data)
- Integrity:
  - hash functions

Hash Functions: Collision-Resistance

Cryptographic Hash Function

A hash function generates a (constant-length) output from any input

To be used as a fingerprint of the file input

Collision: $m \neq m'$ such that $H(m) = H(m')$.

Properties of Hash Functions

- One-wayness (First Preimage):

  given $h = H(x)$, hard to find x' such that $h = H(x')$

- Second Preimage:

  given $x, h = H(x)$, hard to find $x' \neq x$ such that $h = H(x')$

- Collision-Resistance: hard to find $x \neq x'$ such that $H(x) = H(x')$

Hash Functions: Collision-Resistance

Cryptographic Hash Function

A hash function generates a (constant-length) output from any input

To be used as a fingerprint of the file input

Collision: $m \neq m'$ such that $H(m) = H(m')$.

Properties of Hash Functions

- One-wayness (First Preimage):

  given $h = H(x)$, hard to find x' such that $h = H(x')$

- Second Preimage:

  given $x, h = H(x)$, hard to find $x' \neq x$ such that $h = H(x')$

- Collision-Resistance: hard to find $x \neq x'$ such that $H(x) = H(x')$

Hash Functions: Collision-Resistance

Cryptographic Hash Function

A hash function generates a (constant-length) output from any input

To be used as a fingerprint of the file input

Collision: $m \neq m'$ such that $H(m) = H(m')$.

Properties of Hash Functions

- One-wayness (First Preimage):

  given $h = H(x)$, hard to find x' such that $h = H(x')$

- Second Preimage:

  given $x, h = H(x)$, hard to find $x' \neq x$ such that $h = H(x')$

- Collision-Resistance: hard to find $x \neq x'$ such that $H(x) = H(x')$

Hash Functions: Collision-Resistance

Cryptographic Hash Function

A hash function generates a (constant-length) output from any input

To be used as a fingerprint of the file input

Collision: $m \neq m'$ such that $H(m) = H(m')$.

Properties of Hash Functions

- One-wayness (First Preimage):
  given $h = H(x)$, hard to find x' such that $h = H(x')$
- Second Preimage:
  given $x, h = H(x)$, hard to find $x' \neq x$ such that $h = H(x')$
- Collision-Resistance: hard to find $x \neq x'$ such that $H(x) = H(x')$

Hash Functions: Collision-Resistance

Cryptographic Hash Function

A hash function generates a (constant-length) output from any input

To be used as a fingerprint of the file input

Collision: $m \neq m'$ such that $H(m) = H(m')$.

Properties of Hash Functions

- One-wayness (First Preimage):

  given $h = H(x)$, hard to find x' such that $h = H(x')$
- Second Preimage:

  given $x, h = H(x)$, hard to find $x' \neq x$ such that $h = H(x')$
- Collision-Resistance: hard to find $x \neq x'$ such that $H(x) = H(x')$

Generic attack: birthday paradox against collision-resistance

(the output must be at least 256-bit long)

Asymmetric Cryptography

Outline

Integer Factoring

- Given $n = pq$
- Find p and q

Year	Required Complexity	n bitlength
before 2000	64	768
before 2010	80	1024
before 2020	112	2048
before 2030	128	3072
	192	7680
	256	15360

Note that the reduction may be lossy: extra bits are then required

Integer Factoring Records

Integer Factoring

- Given $n = pq$
- Find p and q

Digits	Date	Details
129	April 1994	Quadratic Sieve
130	April 1996	Algebraic Sieve
140	February 1999	
155	August 1999	512 bits
160	April 2003	
200	May 2005	
232	December 2009	768 bits

Integer Factoring Variants

RSA [Rivest-Shamir-Adleman 1978]

- Given $n = pq$, e and $y \in \mathbb{Z}_n^\star$
- Find x such that $y = x^e \bmod n$

Note that this problem is hard without the prime factors p and q, but becomes easy with them: if $d = e^{-1} \bmod \varphi(n)$, then $x = y^d \bmod n$

Flexible RSA [Baric-Pfitzmann and Fujisaki-Okamoto 1997]

- Given $n = pq$ and $y \in \mathbb{Z}_n^\star$
- Find x and $e > 1$ such that $y = x^e \bmod n$

Both problems are assumed as hard as integer factoring:
the prime factors are a trapdoor to find solutions

Integer Factoring Variants

RSA [Rivest-Shamir-Adleman 1978]

- Given $n = pq$, e and $y \in \mathbb{Z}_n^\star$
- Find x such that $y = x^e \bmod n$

Note that this problem is hard without the prime factors p and q, but becomes easy with them: if $d = e^{-1} \bmod \varphi(n)$, then $x = y^d \bmod n$

Flexible RSA [Baric-Pfitzmann and Fujisaki-Okamoto 1997]

- Given $n = pq$ and $y \in \mathbb{Z}_n^\star$
- Find x and $e > 1$ such that $y = x^e \bmod n$

Both problems are assumed as hard as integer factoring:
the prime factors are a trapdoor to find solutions

Integer Factoring Variants

RSA
[Rivest-Shamir-Adleman 1978]

- Given $n = pq$, e and $y \in \mathbb{Z}_n^\star$
- Find x such that $y = x^e \bmod n$

Note that this problem is hard without the prime factors p and q, but becomes easy with them: if $d = e^{-1} \bmod \varphi(n)$, then $x = y^d \bmod n$

Flexible RSA
[Baric-Pfitzmann and Fujisaki-Okamoto 1997]

- Given $n = pq$ and $y \in \mathbb{Z}_n^\star$
- Find x and $e > 1$ such that $y = x^e \bmod n$

Both problems are assumed as hard as integer factoring:
the prime factors are a trapdoor to find solutions

Discrete Logarithm

Discrete Logarithm Problem

- Given $\mathbb{G} = \langle g \rangle$ a cyclic group of order q, and $y \in \mathbb{G}$
- Find x such that $y = g^x$

Possible groups: $\mathbb{G} \in (\mathbb{Z}_p^\star, \times)$, or an elliptic curve

(Computational) Diffie Hellman Problem

- Given $\mathbb{G} = \langle g \rangle$ a cyclic group of order q, and $X = g^x$, $Y = g^y$
- Find $Z = g^{xy}$

The knowledge of x or y helps to solve this problem (trapdoor)

Discrete Logarithm

Discrete Logarithm Problem

- Given $\mathbb{G} = \langle g \rangle$ a cyclic group of order q, and $y \in \mathbb{G}$
- Find x such that $y = g^x$

Possible groups: $\mathbb{G} \in (\mathbb{Z}_p^\star, \times)$, or an elliptic curve

(Computational) Diffie Hellman Problem

- Given $\mathbb{G} = \langle g \rangle$ a cyclic group of order q, and $X = g^x$, $Y = g^y$
- Find $Z = g^{xy}$

The knowledge of x or y helps to solve this problem (trapdoor)

Decisional Problem

(Decisional) Diffie Hellman Problem

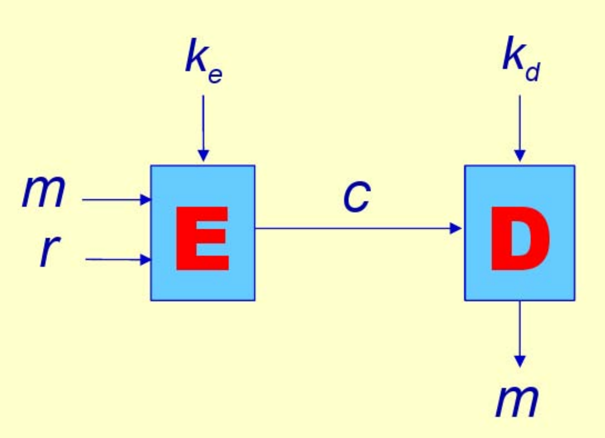
- Given $\mathbb{G} = \langle g \rangle$ a cyclic group of order q, and $X = g^x$, $Y = g^y$, as well as a candidate $Z \in \mathbb{G}$
- Decide whether $Z = g^{xy}$

The adversary is called a distinguisher (outputs 1 bit).
A good distinguisher should behave in significantly different manners according to the input distribution:

$$\begin{aligned}\mathbf{Adv}_{\mathbb{G}}^{\mathsf{ddh}}(\mathcal{A}) = & \Pr[\mathcal{A}(X, Y, Z) = 1 | Z = g^{xy}] \\ & - \Pr[\mathcal{A}(X, Y, Z) = 1 | Z \stackrel{R}{\leftarrow} \mathbb{G}]\end{aligned}$$

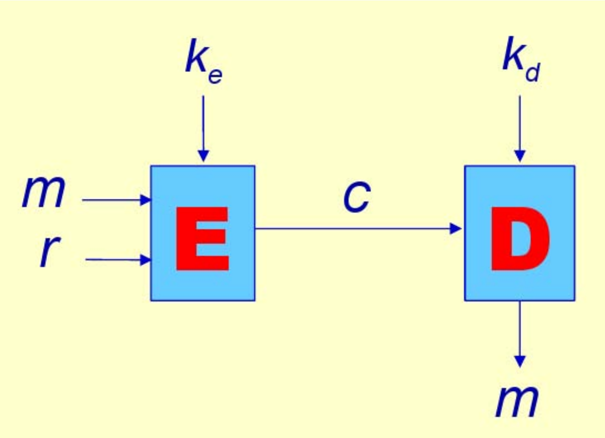
Outline

Public-Key Encryption



Goal: Privacy/Secrecy of the plaintext

Public-Key Encryption



Goal: Privacy/Secrecy of the plaintext

$\mathcal{RSA}$ Encryption

The RSA encryption scheme $\mathcal{RSA}$ is defined by

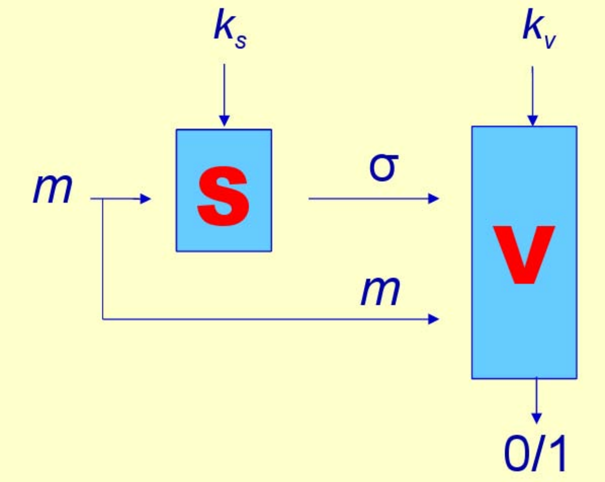
- $\mathcal{K}(1^k)$: p and q two random k-bit prime integers, and an exponent e (possibly fixed, or not): $sk \leftarrow d = e^{-1} \bmod \varphi(n)$ and $pk \leftarrow (n, e)$
- $\mathcal{E}_{pk}(m)$: the ciphertext is $c = m^e \bmod n$
- $\mathcal{D}_{sk}(c)$: the plaintext is $m = c^d \bmod n$

ElGamal Encryption

The ElGamal encryption scheme $\mathcal{EG}$ is defined,
in a group $\mathbb{G} = \langle g \rangle$ of order q

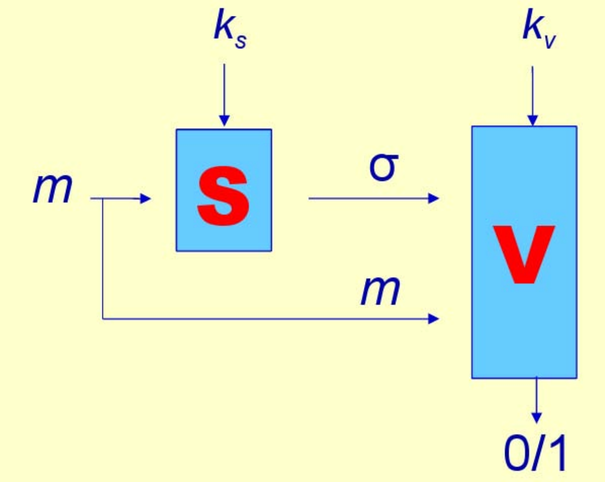
- $\mathcal{K}(\mathbb{G}, g, q)$: $x \stackrel{R}{\leftarrow} \mathbb{Z}_q$, and $sk \leftarrow x$ and $pk \leftarrow y = g^x$
- $\mathcal{E}_{pk}(m)$: $r \stackrel{R}{\leftarrow} \mathbb{Z}_q$, $c_1 \leftarrow g^r$ and $c_2 \leftarrow y^r \times m = pk^r \times m$.
  Then, the ciphertext is $c = (c_1, c_2)$
- $\mathcal{D}_{sk}(c)$ outputs $c_2/c_1^x = c_2/c_1^{sk}$

Signature



Goal: Authentication of the sender

Signature



Goal: Authentication of the sender

$\mathcal{RSA}$ Signature

The RSA signature scheme $\mathcal{RSA}$ is defined by

- $\mathcal{K}(1^k)$: p and q two random k-bit prime integers, and an exponent v (possibly fixed, or not): $sk \leftarrow s = v^{-1} \bmod \varphi(n)$ and $pk \leftarrow (n, v)$
- $\mathcal{S}_{sk}(m)$: the signature is $\sigma = m^s \bmod n$
- $\mathcal{V}_{pk}(m, \sigma)$ checks whether $m = \sigma^v \bmod n$