

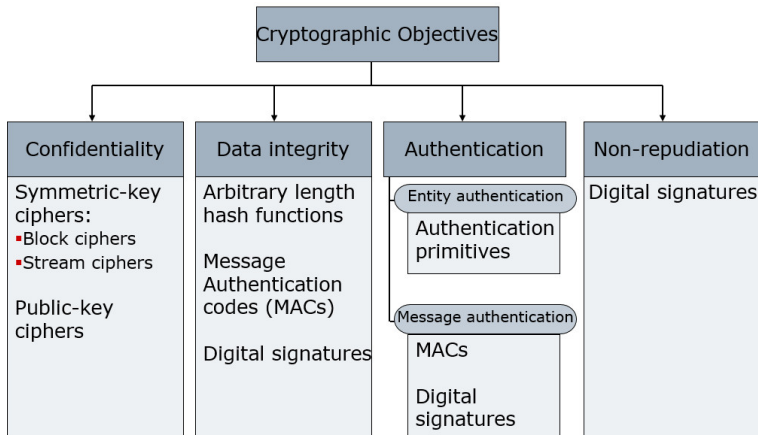
Foundations of Computer Security

Andrea Pintér-Husztí

Contents

- 1 Algorithmic control (Cryptography)
 - Encryption schemes in general
 - Attacks on encryption schemes

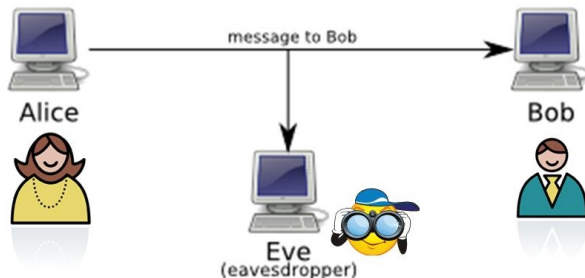
Algorithmic control (Cryptography)



Encryption schemes in general

Encryption schemes

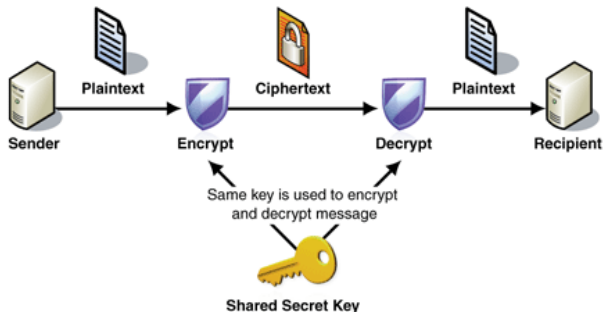
Alice would like to send a confidential message to Bob on a public channel.



- $\mathcal{P}$ Plaintext space : **finite** set of plaintexts
- $\mathcal{C}$ Ciphertext space: **finite** set of ciphertexts
- $\mathcal{K}$ Key space: **finite** set of possible keys
- $m \in \mathcal{P}$ plaintext
- $c \in \mathcal{C}$ ciphertext

Symmetric encryption schemes

The encryption key is the same as (bit by bit) the decryption key, or the decryption key can be calculated from the encryption key in polynomial time.

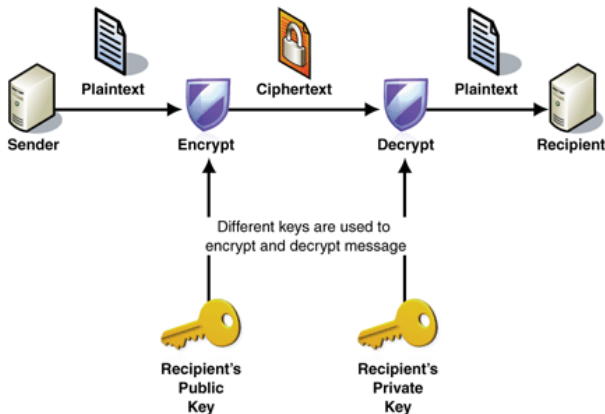


Asymmetric encryption schemes

The encryption key and the decryption key are different.

Calculating the decryption key from the encryption key is infeasible (intractable).

Infeasible: we do not know polynomial time algorithm for it



Comparison

	Symmetric	Asymmetric
Secrecy of the keys	keys are kept secret (K)	(PK, SK) public and secret key
Handling the keys	key exchange algorithms are needed	Public Key Infrastructure
Computational time	fast algorithms	slow algorithms
Size of messages	large size	small size
Examples	TDES, AES	RSA, ElGamal, elliptic curve encryption

Symmetric encryption scheme formal definition

Definíció

A triple $SE = (Key, Enc, Dec)$ is a symmetric encryption scheme, if

- *Key: a key-generation algorithm that outputs a key $K \in \mathcal{K}$ for a security parameter k .*
- *Enc: an encryption algorithm that takes as input a key $K \in \mathcal{K}$ and a plaintext message $m \in \mathcal{P}$ and outputs a ciphertext $c \in \mathcal{C}$.*

$$c = Enc_K(m)$$

- *Dec: a decryption algorithm that takes as input a key K and a ciphertext c and outputs a plaintext m .*

$$m = Dec_K(c)$$

Symmetric encryption scheme

- In most cases a random r is also the input of the encryption algorithm, hence the encryption algorithm is randomized.
- The decryption algorithm is deterministic.

Definíció

The symmetric encryption scheme $SE = (Key, Enc, Dec)$ provides correct decryption, if $\forall m \in \mathcal{P}$ and $\forall K \in \mathcal{K}$

$$Dec_K(Enc_K(m)) = m.$$

Asymmetric encryption scheme formal definition

Definíció

A triple $AE = (Key, Enc, Dec)$ is an asymmetric encryption scheme, if

- *Key*: a key-generation algorithm that for a security parameter k outputs a key pair (PK, SK) , where PK is the public and SK is the secret key.
- *Enc*: an encryption algorithm that takes as input a public key PK and the a plaintext message $m \in \mathcal{P}$ and outputs a ciphertext $c \in \mathcal{C}$.

$$c = Enc_{PK}(m)$$

- *Dec*: a decryption algorithm that takes as input a secret key SK and a ciphertext c and outputs a plaintext m .

$$m = Dec_{SK}(c)$$

Asymmetric encryption scheme

- In most cases a random r is also the input of the encryption algorithm, hence the encryption algorithm is randomized.
- The decryption algorithm is deterministic.
- The output of the key-generation algorithm gives the sets $\mathcal{P}, \mathcal{C}, \mathcal{K}$.

Definíció

The asymmetric encryption scheme $AE = (Key, Enc, Dec)$ provides correct decryption, if $\forall m \in \mathcal{P}$ and $\forall (PK, SK) \in \mathcal{K}$

$$Dec_{SK}(Enc_{PK}(m)) = m.$$

Attacks

The goal of the adversary:

- to get the secret decryption key
- to get the plaintext for a given ciphertext

Attack scenarios:

- Ciphertext Only Attack (passive attack)
- Known Plaintext Attack (passive attack)
- Chosen Plaintext Attack
 - Non-adaptive (passive attack)
 - Adaptive (active attack)
- Chosen Ciphertext Attack
 - Non-adaptive (passive attack)
 - Adaptive (active attack)

COA

Ciphertext Only Attack The adversary has a list of ciphertexts (encryption is made by the same key):

$Enc_K(m_1)$	$Enc_{PK}(m_1)$
$Enc_K(m_2)$	$Enc_{PK}(m_2)$
$\vdots$	$\vdots$
$Enc_K(m_n)$	$Enc_{PK}(m_n)$

KPA

Known Plaintext Attack The adversary has a list of plaintext, ciphertext pairs (encryption is made by the same key):

$(m_1, Enc_K(m_1))$	$(m_1, Enc_{PK}(m_1))$
$(m_2, Enc_K(m_2))$	$(m_2, Enc_{PK}(m_2))$
$\vdots$	$\vdots$
$(m_n, Enc_K(m_n))$	$(m_n, Enc_{PK}(m_n))$

The goal is to get the plaintext for a ciphertext not listed.

Chosen Plaintext Attack The adversary has a list of plaintext, ciphertext pairs, where the plaintexts are chosen by the adversary. (encryption is made by the same key)

$(m_1, Enc_K(m_1))$	$(m_1, Enc_{PK}(m_1))$
$(m_2, Enc_K(m_2))$	$(m_2, Enc_{PK}(m_2))$
$\vdots$	$\vdots$
$(m_n, Enc_K(m_n))$	$(m_n, Enc_{PK}(m_n))$

Non-adaptive case: the adversary chooses the plaintexts before receiving the ciphertexts

Adaptive case: the adversary chooses the next plaintext according to the ciphertexts received

The goal is to get the plaintext for a ciphertext not listed.

CCA

Chosen Ciphertext Attack The adversary has a list of plaintext, ciphertext pairs, where the ciphertexts are chosen by the adversary. (encryption is made by the same key)

$(c_1, Dec_K(c_1))$	$(c_1, Dec_{SK}(c_1))$
$(c_2, Dec_K(c_2))$	$(c_2, Dec_{SK}(c_2))$
$\vdots$	$\vdots$
$(c_n, Dec_K(c_n))$	$(c_n, Dec_{SK}(c_n))$

Non-adaptive case: the adversary chooses the ciphertexts before receiving the plaintexts

Adaptive case: the adversary chooses the next ciphertext according to the plaintexts received

The goal is to get the plaintext for a ciphertext not listed.

Security issues

- Perfect secrecy: the adversary has **unbounded** computational power.
- Computational secrecy: the adversary has **bounded** computational power, uses polynomial time algorithms
- Kerckhoffs' intention was that an encryption scheme should be designed so as to be secure even if an adversary knows the details of all the component algorithms of the scheme, as long as the adversary doesn't know the key being used.