Quelles garanties avec la cryptographie ?

David Pointcheval
Ecole normale supérieure, CNRS & INRIA
Collège de France
27 avril 2011

Cryptography
Provable Security
Security of Signatures
Security of Encryption

Outline

1 Cryptography
2 Provable Security
3 Security of Signatures
4 Security of Encryption

Security of Communications

One ever wanted to exchange information securely
With the all-digital world, security needs are even stronger... 
In your pocket
But also at home

Cryptography

3 Historical Goals
- Confidentiality: The content of a message is concealed
- Authenticity: The author of a message is well identified
- Integrity: Messages have not been altered

between a sender and a recipient, against an adversary.
Also within groups, with insider adversaries
Cannot address availability, but should not affect it!
First Encryption Mechanisms

The goal of encryption is to hide a message

Substitutions and permutations

Security relies on the secrecy of the mechanism

⇒ How to widely use them?

Alberti’s disk

Mono-alphabetical Substitution

Wheel – M 94 (CSP 488)

Poly-alphabetical Substitution

David Pointcheval – ENS/CNRS/INRIA Collège de France 5/40

Use of a (Secret) Key

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

Enigma:
choice of the connectors and the rotors

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

⇒ Security analysis is required

David Pointcheval – ENS/CNRS/INRIA Collège de France 6/40

Modern Cryptography

Secret Key Encryption

One secret key only shared by Alice and Bob:
this is a common parameter for both E and D

Public Key Cryptography [Diffie-Hellman – 1976]

Bob’s public key is used by Alice as a parameter to E
Bob’s private key is used by Bob as a parameter to D

DES and AES

Still substitutions and permutations, but considering various classes of attacks (statistic)

DES: Data Encryption Standard

“Broken” in 1998 by brute force: too short keys (56 bits)!

⇒ No better attack granted a safe design!

New standard since 2001: Advanced Encryption Standard

Longer keys: from 128 to 256 bits

Criteria: Security arguments against many attacks

What does security mean?
Practical Secrecy

Perfect Secrecy vs. Practical Secrecy

- 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 $\Rightarrow$ information theory
- In practice: adversaries are limited in time/power $\Rightarrow$ complexity theory

We thus model all the players (the legitimate ones and the adversary) as Probabilistic Polynomial Time Turing Machines:

- computers that run programs

What is a Secure Cryptographic Scheme?

- What does security mean?
  $\Rightarrow$ Security notions have to be formally defined
- How to guarantee above security claims for concrete schemes?
  $\Rightarrow$ Provable security

Provable Security

- if an adversary is able to break the cryptographic scheme
- then one can break a well-known hard problem

Symmetric Cryptography

The secrecy of the key guarantees the secrecy of communications

Asymmetric Cryptography

The secrecy of the private key guarantees the secrecy of communications

General Method

Computational Security Proofs

To prove the security of a cryptographic scheme, one needs

- a formal security model (security notions)
- a reduction: if one (Adversary) can break the security notions, then one (Simulator + Adversary) can break a hard problem
- acceptable computational assumptions (hard problems)

Proof by contradiction
### Integer Factoring

#### Records

<table>
<thead>
<tr>
<th>Digits</th>
<th>Date</th>
<th>Bit-Length</th>
</tr>
</thead>
<tbody>
<tr>
<td>130</td>
<td>April 1996</td>
<td>431 bits</td>
</tr>
<tr>
<td>140</td>
<td>February 1999</td>
<td>465 bits</td>
</tr>
<tr>
<td>155</td>
<td>August 1999</td>
<td>512 bits</td>
</tr>
<tr>
<td>160</td>
<td>April 2003</td>
<td>531 bits</td>
</tr>
<tr>
<td>200</td>
<td>May 2005</td>
<td>664 bits</td>
</tr>
<tr>
<td>232</td>
<td>December 2009</td>
<td>768 bits</td>
</tr>
</tbody>
</table>

#### Complexity

- $768$ bits → $2^{64}$ op.
- $1024$ bits → $2^{80}$ op.
- $2048$ bits → $2^{112}$ op.
- $3072$ bits → $2^{128}$ op.
- $7680$ bits → $2^{192}$ op.
- $15360$ bits → $2^{256}$ op.

### One-Way Functions

**F(1k)** generates a function $f : X \rightarrow Y$

- From $x \in X$, it is easy to compute $y = f(x)$
- Given $y \in Y$, it is hard to find $x \in X$ such that $y = f(x)$

#### RSA Problem

[Rivest-Shamir-Adleman 1978]

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

This problem is hard without the prime factors $p$ and $q$.

It becomes easy with them: if $d = e^{-1} \mod \varphi(n)$, then $x = y^d \mod n$

This problem is assumed as hard as integer factoring:

the prime factors are a trapdoor to find solutions

⇒ trapdoor one-way permutation

### Signature

**Goal:** Authentication of the sender

- $k_s$ to $S$
- $k_v$ to $V$
- $S$ to $m$ to $\sigma$ to $V$

- $0/1$
EUF − NMA: Security Game

\[ k_y \xleftarrow{} G \xrightarrow{} k_z \]

\[ m \sigma \xleftarrow{} A \]

\[ \forall k_y, m, \sigma \]

**Succ\text{\textsuperscript{euf}}_{\text{SG}}(A) = \Pr[(k_s, k_y) \leftarrow G(); (m, \sigma) \leftarrow A(k_y) : \forall (k_y, m, \sigma) = 1\]

should be negligible.

\[ A \text{ knows the public key only } \Rightarrow \text{No-Message Attack (NMA)} \]

EUF − KMA

**One-Way Function**

- \( G(1^k): f \xleftarrow{} F(1^k) \) and \( x \xleftarrow{} X \), set \( y = f(x) \), \( k_s = x \) and \( k_v = (f, y) \)
- \( S(x, m) = k_s = x \)
- \( \forall'(f, y), m, x' \) checks whether \( f(x') = y \)

Under the one-wayness of \( F \), \( \text{Succ}\text{\textsuperscript{euf−nma}}(A) \) is small.

But given one signature, one can “sign” any other message! Signatures are public! \( \Rightarrow \text{Known-Message Attacks (KMA)} \)

The adversary has access to a list of messages-signatures.

EUF − CMA

**One-Way Functions**

- \( G(1^k): f \xleftarrow{} F(1^k) \) and \( x \xleftarrow{} X \), set \( y = f(x) \), \( k_s = x \) and \( k_v = (f, y) \)
- \( S(x, m) = k_s = x \)
- \( \forall'(f, y), m, x' \) checks whether \( f(x') = y \)

Under the one-wayness of \( F \), \( \text{Succ}\text{\textsuperscript{euf−cma}}(A) \) is small.

With the signature of \( m = 0^k \), I cannot forge any other signature.

With the signatures of \( m = 0^k \) and \( m' = 1^k \), I learn \( \bar{x} \): the secret key

Messages can be under the control of the adversary!

\( \Rightarrow \text{Chosen-Message Attacks (CMA)} \)

The adversary has access to any signature of its choice:

Chosen-Message Attacks (oracle access):

\[ \text{Succ}\text{\textsuperscript{euf−cma}}(A) = \Pr[(k_s, k_v) \leftarrow G(); (m, \sigma) \leftarrow A_S(k_s, m); (k_v, m, \sigma) = 1 \]

\[ \forall i, m \neq m_i \wedge \forall (k_v, m, \sigma) = 1 \]
The **RSA Signature**

The RSA signature scheme RSA is defined by

- \( G(1^k) \): \( p \) and \( q \), two random primes, and an exponent \( v \)
  \( n = pq, k_s \leftarrow s = v^{-1} \mod \varphi(n) \) and \( k_v \leftarrow (n, v) \)
- \( S(k_s, m) \): the signature is \( \sigma = m^s \mod n \)
- \( V(k_v, m, \sigma) \) checks whether \( m = \sigma^v \mod n \)

**Theorem (The Plain RSA is not EUF – NMA)**

The plain RSA signature is not secure at all!

**Proof.**

Choose a random \( \sigma \in \mathbb{Z}_n^* \), and set \( m = \sigma^v \mod n \).

By construction, \( \sigma \) is a valid signature of \( m \)

---

**Full-Domain Hash Signature**

The FDH-RSA signature scheme is defined by

- \( G(1^k) \): \( p \) and \( q \), two random primes, and an exponent \( v \)
  \( n = pq, k_s \leftarrow s = v^{-1} \mod \varphi(n) \) and \( k_v \leftarrow (n, v) \)
- \( H \) is a hash function onto \( \mathbb{Z}_n^* \)
- \( S(k_s, m) \): the signature is \( \sigma = H(m)^s \mod n \)
- \( V(k_v, m, \sigma) \) checks whether \( H(m) = \sigma^v \mod n \)

**Theorem (Security of the FDH-RSA)**

The FDH-RSA is EUF – CMA under appropriate assumptions on \( H \), and assuming the RSA problem is hard.

---

**FDH-RSA Security**

Initial reduction: \( T \approx q_H \times t \) (where \( q_H \) is number of Hashing queries \( \approx 2^{60} \))

- \( k = 1024 \) \( (2^{80}) \) \( t < 2^{80} \) \( T < 2^{140} \) \( \times \)
- \( k = 2048 \) \( (2^{112}) \) \( t < 2^{80} \) \( T < 2^{140} \) \( \times \)
- \( k = 3072 \) \( (2^{128}) \) \( t < 2^{80} \) \( T < 2^{140} \) \( \times \)

\( \implies \) large modulus required!

---

**Improved Security**

By exploiting the random self-reducibility of RSA: \( (xr)^6 = x^6r^6 \mod n \)

\( \implies \) Improved reduction: \( T \approx q_S \times t \) (where \( q_S \) is the number is Signing queries \( \leq 2^{30} \))

With \( k = 2048 \) and \( t < 2^{80} \), one gets \( T < 2^{110} \) \( \checkmark \)

(Best algorithm in \( 2^{112} \))
RSA-PSS (PKCS #1 v2.1) [Bellare-Rogaway – Eurocrypt ’96]

- $m$ is the message to encrypt
- $r$ is the additional randomness to make encryption probabilistic

After the transformation, $w \parallel s \parallel t$ goes in the plain RSA

Theorem (EUF-CMA Security) [Bellare-Rogaway – Eurocrypt ’96]

RSA-PSS is EUF-CMA secure under the RSA assumption

Security reduction between EUF – CMA and the RSA assumption:

$$T \approx t$$

$$\Rightarrow 1024\text{-bit RSA moduli provide } 2^{80} \text{ security}$$

Public-Key Encryption

Goal: Privacy/Secrecy of the plaintext

OW − CPA: Security Game

The RSA Encryption [Rivest-Shamir-Adleman 1978]

- $G(1^k)$: $p$ and $q$, two random primes, and an exponent $e$
  $$n = pq, \ sk \leftarrow d = e^{-1} \mod \varphi(n) \text{ and } pk \leftarrow (n, e)$$
- $\mathcal{E}(pk, m) = c = m^e \mod n$; $\mathcal{D}(sk, c) = m = c^d \mod n$

$$\mathcal{G}(sk, c_1 || c_2) : m_1 = \mathcal{D}(sk, c_1), m_2 = c_2, \text{ output } m = m_1 || m_2$$

OW – CPA Too Weak

- $\mathcal{G}' = \mathcal{G}$; $\mathcal{E}'(pk, m = m_1 || m_2) = \mathcal{E}(pk, m_1) || m_2 = c_1 || c_2$
- $\mathcal{D}'(sk, c_1 || c_2)$: $m_1 = \mathcal{D}(sk, c_1), m_2 = c_2$, output $m = m_1 || m_2$

If $(\mathcal{G}, \mathcal{E}, \mathcal{D})$ is OW − CPA: then $(\mathcal{G}', \mathcal{E}', \mathcal{D}')$ is OW − CPA too

But this is clearly not enough: **half or more of the message leaks!**
OW – CPA: Is it Enough?

For a “yes/no” answer or “sell/buy” order, one bit of information may be enough for the adversary! How to model that no bit of information leaks?

Perfect Secrecy vs. Computational Secrecy

- **Perfect secrecy**: the distribution of the ciphertext is **perfectly** independent of the plaintext
- **Computational secrecy**: the distribution of the ciphertext is **computationally** independent of the plaintext

Idea: No adversary can distinguish a ciphertext of $m_0$ from a ciphertext of $m_1$. Probabilistic encryption is required!

ElGamal Encryption

[ElGamal 1985]

The ElGamal Encryption ($\mathcal{E_\mathcal{G}}$)

- $\mathcal{G}(1^k)$: $\mathcal{G} = \langle g \rangle$ of order $q$, $sk = x \overset{R}{\leftarrow} \mathbb{Z}_q$ and $pk \leftarrow y = g^x$
- $\mathcal{E}(pk, m, r) = (c_1 = g^r, c_2 = y^r m)$
- $D(sk, (c_1, c_2)) = c_2 / c_1^x$

The ElGamal encryption is IND – CPA, under the DDH assumption

Decisional Diffie-Hellman Problem

For $\mathcal{G} = \langle g \rangle$ of order $q$, and $x, y \overset{R}{\leftarrow} \mathbb{Z}_q$,

- Given $X = g^x$, $Y = g^y$ and $Z = g^z$, for either $z \overset{R}{\leftarrow} \mathbb{Z}_q$ or $z = xy$
- Decide whether $z = xy$

This problem is assumed hard to decide in appropriate groups $\mathcal{G}$!

IND – CPA: Security Game

Let $A$ be an adversary against $\mathcal{E_\mathcal{G}}$: $B$ is an adversary against DDH:

- $A$ gets $pk \leftarrow X$ from $B$, and outputs $(m_0, m_1)$
- $B$ sets $c_1 \leftarrow Y$
- $B$ chooses $b \overset{R}{\leftarrow} \{0,1\}$, sets $c_2 \leftarrow Z \times m_b$, and sends $c = (c_1, c_2)$
- $B$ receives $b'$ from $A$ and outputs $d = (b' = b)$

$2 \times \Pr[b' = b] - 1$

$= \text{Adv}_{\mathcal{E_\mathcal{G}}\text{–} \text{IND\text{–} CPA}}(A)$, if $z = xy$

$= 0$, if $z \overset{R}{\leftarrow} \mathbb{Z}_q$
ElGamal is IND – CPA: Proof

As a consequence,

1. \( 2 \times \Pr[b' = b | z = xy] - 1 = \mathsf{Adv}_{\mathcal{G}}^{\text{IND-CPA}}(\mathcal{A}) \)
2. \( 2 \times \Pr[b' = b | z \leftarrow \mathbb{Z}_q^*] - 1 = 0 \)

If one subtracts the two lines:

\[
\mathsf{Adv}_{\mathcal{G}}^{\text{IND-CPA}}(\mathcal{A}) = 2 \times \left( \Pr[d = 1 | z = xy] - \Pr[d = 1 | z \leftarrow \mathbb{Z}_q^*] \right) = 2 \times \mathsf{Adv}_{\mathcal{G}}^{\text{ddh}}(B) \leq 2 \times \mathsf{Adv}_{\mathcal{G}}^{\text{ddh}}(t)
\]

The ElGamal Encryption

\[\mathcal{G}(1^k): G = \langle g \rangle \text{ of order } q, \ sk = x \leftarrow \mathbb{Z}_q \text{ and } pk \leftarrow y = g^x\]

\[\mathcal{E}(pk, m, r) = (c_1 = g^r, c_2 = y^r m); \mathcal{D}(sk, (c_1, c_2)) = c_2 / c_1^x\]

IND – CPA: Is it Enough?

The ElGamal Encryption

- \( \mathcal{G}(1^k): G = \langle g \rangle \text{ of order } q, \ sk = x \leftarrow \mathbb{Z}_q \text{ and } pk \leftarrow y = g^x \)
- \( \mathcal{E}(pk, m, r) = (c_1 = g^r, c_2 = y^r m); \mathcal{D}(sk, (c_1, c_2)) = c_2 / c_1^x \)

Private Auctions

All the players \( P_i \) encrypt their bids \( c_i = \mathcal{E}(pk, b_i) \) for the authority; the authority opens all the \( c_i \); the highest bid \( b_i \) wins

- IND – CPA guarantees privacy of the bids
- Malleability: from \( c_i = \mathcal{E}(pk, b_i) \), without knowing \( b_i \), one can generate \( c' = \mathcal{E}(pk, 2b_i) \): an unknown higher bid!

IND – CPA does not imply Non-Malleability

RSA-OAEP (PKCS #1 v2.1)

The RSA encryption is OW – CPA, under the RSA assumption, but even not IND – CPA: need of randomness and redundancy

- \( m \) is the message to encrypt
- \( r \) is the additional randomness to make encryption probabilistic
- 00 . . . 00 is redundancy to be checked at decryption time

After the transformation, \( X \parallel Y \) goes in the plain RSA

Theorem (NM vs. CCA) [Bellare-Desai-Pointcheval-Rogaway – Crypto '98]

The chosen-ciphertext security implies non-malleability

\( \Rightarrow \) the highest security level

Theorem (IND-CCA Security) [Fujisaki-Okamoto-Pointcheval-Stern – Crypto '01]

RSA-OAEP is IND-CCA secure under the RSA assumption
RSA-OAEP Security Proof  [Fujisaki-Okamoto-Pointcheval-Stern – Crypto ’01]

More precisely, to get information on \( m \), encrypted in \( c = f(X \parallel Y) \), one must have asked \( \mathcal{H}(X) \implies \) partial inversion of \( f \)

For RSA: partial inversion and full inversion are equivalent (but at a computational loss)

---

RSA-OAEP Security

With provable security, one can precisely get:

- the security games one wants to resist against any adversary
- the security level, according to the resources of the adversary

But, it is under some assumptions:

- the best attacks against famous problems (integer factoring, etc)
- no leakage of information excepted from the given oracles

Cryptographers’ goals are thus

- to analyze the intractability of the underlying problems
- to define realistic and strong security notions (games)
- to correctly model the leakage of information (oracle access)
- to design schemes with tight security reductions

Implementations and uses must satisfy the constraints!

---

REACT-RSA Security  

[Okamoto-Pointcheval – CT-RSA ’01]

Game 0

Adversary running time \( t \)

Algorithm running time \( T = f(t) \)

If there is an adversary that distinguishes, within time \( t \), the two ciphertexts with overwhelming advantage (close to 1), one can break RSA within time \( T \approx 2t + 3q_H^2 k^3 \)

(where \( q_H \) is number of Hashing queries \( \approx 2^{60} \))

\[
\begin{align*}
k &= 1024 \quad (2^{80}) & t &< 2^{80} & T &< 2^{152} \quad \times \\
k &= 2048 \quad (2^{112}) & t &< 2^{80} & T &< 2^{155} \quad \times \\
k &= 3072 \quad (2^{128}) & t &< 2^{80} & T &< 2^{158} \quad \times
\end{align*}
\]

large modulus: \( > 4096 \) bits!