I – Basics

David Pointcheval
Ecole normale supérieure, CNRS & INRIA

IACR-SEAMS School
Cryptographie: Foundations and New Directions
November 2016 – Hanoi – Vietnam

Outline

1 Cryptography
   ■ Introduction
   ■ Formal Notations

2 Provable Security
   ■ Definition
   ■ Computational Assumptions
   ■ Some Reductions

3 Public-Key Encryption
   ■ One-Wayness
   ■ Indistinguishability

4 Conclusion

Secrecy of Communications

One ever wanted to communicate secretly

With the all-digital world, security needs are even stronger
What Does Secrecy Mean?

Shannon provides a definition of secrecy:

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

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

Outline

1 Cryptography
   - Introduction
   - Formal Notations

2 Provable Security

3 Public-Key Encryption

4 Conclusion
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

Secrecy of the private key $sk \Rightarrow$ secrecy of communications
Because of $pk$, perfect secrecy is definitely impossible!

Outline

1 Cryptography
2 Provable Security
   - Definition
   - Computational Assumptions
   - Some Reductions
3 Public-Key Encryption
4 Conclusion

What is a Secure Cryptographic Scheme/Protocol?

- Public-key encryption:
  Secrecy of the private key $sk \Rightarrow$ secrecy of communications
- What does mean secrecy?
  → Security notions have to be formally defined
- How to guarantee above security claims for concrete schemes?
  → Provable security

Provable Security

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

hard instance $\Rightarrow$ solution
General Method

Computational Security Proofs

In order to prove the security of a cryptographic scheme/protocol, one needs:
- a formal security model (security notions)
- acceptable computational assumptions (hard problems)
- a reduction: if one can break the security notions, then one can break the hard problem

Outline

1 Cryptography
2 Provable Security
   - Definition
   - Computational Assumptions
   - Some Reductions
3 Public-Key Encryption
4 Conclusion

Integer Factoring

[Lenstra-Verheul 2000]

Integer Factoring

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

<table>
<thead>
<tr>
<th>Year</th>
<th>Required Complexity</th>
<th>$n$ bitlength</th>
</tr>
</thead>
<tbody>
<tr>
<td>before 2000</td>
<td>64</td>
<td>768</td>
</tr>
<tr>
<td>before 2010</td>
<td>80</td>
<td>1024</td>
</tr>
<tr>
<td>before 2020</td>
<td>112</td>
<td>2048</td>
</tr>
<tr>
<td>before 2030</td>
<td>128</td>
<td>3072</td>
</tr>
<tr>
<td></td>
<td>192</td>
<td>7680</td>
</tr>
<tr>
<td></td>
<td>256</td>
<td>15360</td>
</tr>
</tbody>
</table>

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

Integer Factoring

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

<table>
<thead>
<tr>
<th>Digits</th>
<th>Date</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>129</td>
<td>April 1994</td>
<td>Quadratic Sieve</td>
</tr>
<tr>
<td>130</td>
<td>April 1996</td>
<td>Algebraic Sieve</td>
</tr>
<tr>
<td>140</td>
<td>February 1999</td>
<td></td>
</tr>
<tr>
<td>155</td>
<td>August 1999</td>
<td></td>
</tr>
<tr>
<td>160</td>
<td>April 2003</td>
<td></td>
</tr>
<tr>
<td>200</td>
<td>May 2005</td>
<td>512 bits</td>
</tr>
<tr>
<td>232</td>
<td>December 2009</td>
<td>768 bits</td>
</tr>
</tbody>
</table>
### Integer Factoring Variants

**RSA**

[Rivest-Shamir-Adleman 1978]

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

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

**Flexible RSA**

[Baric-Pfitzmann and Fujisaki-Okamoto 1997]

- Given \( n = pq \) and \( y \in \mathbb{Z}_n^* \)
- Find \( x \) and \( e > 1 \) such that \( y = x^e \mod n \)

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

### Discrete Logarithm

**Discrete Logarithm Problem**

- Given \( G = \langle g \rangle \) a cyclic group of order \( q \), and \( y \in G \)
- Find \( x \) such that \( y = g^x \)

Possible groups: \( G \in (\mathbb{Z}_p^*, \cdot) \), or an elliptic curve

**(Computational) Diffie Hellman Problem**

- Given \( G = \langle g \rangle \) 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)

### Success Probabilities

For any computational problem \( P \), we quantify the quality of an adversary \( A \) by its success probability in finding the solution:

\[
\text{Succ}^P(A) = \Pr[A(\text{instance}) \rightarrow \text{solution}]
\]

We quantify the hardness of the problem by the success probability of the best adversary within time \( t \):

\[
\text{Succ}(t) = \max_{|A| \leq t} \{\text{Succ}(A)\}
\]

Note that the probability space can be restricted:

- some inputs are fixed, and others only are randomly chosen

**Discrete Logarithm Problem**

We usually fix the group \( G = \langle g \rangle \) of order \( q \), \( X \) is randomly chosen:

\[
\text{Succ}^d_G(A) = \Pr_{x \sim \mathbb{Z}_q^*}[A(g^x) \rightarrow x]
\]

**Decisional Problem**

** (Decisional) Diffie Hellman Problem**

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

In such a case, the adversary is called a distinguisher (outputs 1 bit)

A good distinguisher should behave in significantly different manners according to the input distribution:

\[
\text{Adv}^{ddh}_G(A) = \Pr[A(X, Y, Z) = 1 | Z = g^{xy}] - \Pr[A(X, Y, Z) = 1 | Z \sim G]
\]

\[
\text{Adv}^{ddh}(t) = \max_{|A| \leq t} \{\text{Adv}^{ddh}_G(A)\}
\]
**Distribution Indistinguishability**

**Indistinguishabilities**

Let $D_0$ and $D_1$ be two distributions on a finite set $X$:

- $D_0$ and $D_1$ are perfectly indistinguishable if

$$\text{Dist}(D_0, D_1) = \sum_{x \in X} \left| \Pr_{a \in D_1}[a = x] - \Pr_{a \in D_0}[a = x] \right| = 0$$

- $D_0$ and $D_1$ are statistically indistinguishable if

$$\text{Dist}(D_0, D_1) = \sum_{x \in X} \left| \Pr_{a \in D_1}[a = x] - \Pr_{a \in D_0}[a = x] \right| = \text{negl}()$$

---

**Computational Indistinguishability**

Let $D_0$ and $D_1$ be two distributions on a finite set $X$, a distinguisher $A$ between $D_0$ and $D_1$:

- $A$’s advantage $\text{Adv}^{D_0, D_1}(A)$ is

$$\text{Adv}^{D_0, D_1}(A) = \Pr_{a \in D_1}[A(a) = 1] - \Pr_{a \in D_0}[A(a) = 1]$$

- The computational indistinguishability of $D_0$ and $D_1$ is

$$\text{Adv}^{D_0, D_1}(t) = \max_{|A| \leq t} \{ \text{Adv}^{D_0, D_1}(A) \}$$

**Theorem**

$$\forall t, \quad \text{Adv}^{D_0, D_1}(t) \leq \text{Dist}(D_0, D_1)$$

---

**Outline**

1. Cryptography
2. Provable Security
   - Definition
   - Computational Assumptions
   - Some Reductions
3. Public-Key Encryption
4. Conclusion

---

**DDH $\leq$ CDH $\leq$ DLP**

**CDH $\leq$ DLP**

Let $\mathcal{A}$ be an adversary against the DLP within time $t$, then we build an adversary $B$ against the CDH: given $X$ and $Y$, $B$ runs $\mathcal{A}$ on $X$, that outputs $x'$ (correct or not); then $B$ outputs $Y^{x'}$.

The running time $t'$ of $B$ is the same as $\mathcal{A}$, plus one exponentiation:

$$\text{Succ}_{\text{cdh}}^G(t') \geq \text{Succ}_{\text{cdh}}^G(B) = \Pr[B(X, Y) \rightarrow g^{xy} = Y^x] = \Pr[A(X) \rightarrow x] = \text{Succ}_{\text{dlp}}^G(\mathcal{A})$$

Taking the maximum on the adversaries $\mathcal{A}$:

$$\text{Succ}_{\text{cdh}}^G(t + \tau_{\exp}) \geq \text{Succ}_{\text{dlp}}^G(t)$$
Let $A$ be an adversary against the CDH within time $t$, we build an adversary $B$ against the DDH: given $X, Y$ and $Z$, $B$ runs $A$ on $(X, Y)$, that outputs $Z'$; then $B$ outputs 1 if $Z' = Z$ and 0 otherwise.

The running time of $B$ is the same as $A$: and $\text{Adv}_{\text{ddh}}^G(t)$ is greater than $\text{Adv}_{\text{cdh}}^G(A)$:

\[
\text{Adv}_{\text{ddh}}^G(B) = \text{Pr}[B \rightarrow 1 | Z = g^{xy}] - \text{Pr}[B \rightarrow 1 | Z \leftarrow G] = \text{Pr}[A(X, Y) \rightarrow Z | Z = g^{xy}] - \text{Pr}[A(X, Y) \rightarrow Z | Z \leftarrow G] = \text{Pr}[A(X, Y) \rightarrow g^{xy}] - \text{Pr}[A(X, Y) \rightarrow Z | Z \leftarrow G] = \text{Succ}_{\text{cdh}}^G(A) - 1/q
\]

Taking the maximum on the adversaries $A$:

\[
\text{Adv}_{\text{ddh}}^G(t) \geq \text{Succ}_{\text{cdh}}^G(t) - 1/q
\]

### OW − CPA

#### One-Wayness

For a public-key encryption scheme $S = (K, E, D)$, without the secret key $sk$, it should be computationally impossible to recover the plaintext $m$ from the ciphertext $c$:

\[
\text{Succ}_{\text{ow}}^S(A) = \text{Pr}[(sk, pk) \leftarrow K(); m \leftarrow M; c = E_{pk}(m) : A(pk, c) \rightarrow m] \text{ should be negligible}
\]

#### Chosen-Plaintext Attacks

In the public-key setting, the adversary has access to the encryption key (the public key), and thus can encrypt any plaintext of its choice: chosen-plaintext attack.
**OW − CPA Security Game**

The ElGamal encryption scheme \( E_G \) is defined, in a group \( G = \langle g \rangle \) of order \( q \), for \( m \in G \):

1. \( K_c(G, g, q) : x \overset{R}{\leftarrow} \mathbb{Z}_q \), and \( sk \leftarrow x \) and \( pk \leftarrow y = g^x \)
2. \( E_{pk}(m) : r \overset{R}{\leftarrow} \mathbb{Z}_q \), \( c_1 \leftarrow g^r \) and \( c_2 \leftarrow y^r \times m = pk^r \times m \)
   Then, the ciphertext is \( c = (c_1, c_2) \)
3. \( D_{sk}(c) \) outputs \( c_2 / c_1^x = c_2 / c_1^{sk} \)

**Theorem (ElGamal is OW − CPA)**

\[
\text{Succ}_{E_G}^{\text{OW−CPA}}(t) \leq \text{Succ}_{G}^{\text{CDH}}(t)
\]

---

**ElGamal is OW − CPA: Proof**

Let \( A \) be an adversary against \( E_G \), we build an adversary \( B \) against CDH: let us be given a CDH instance \( (X, Y) \):

- \( A \) gets \( pk \leftarrow X \) from \( B \)
- \( B \) sets \( c_1 \leftarrow Y \)
- \( B \) chooses \( c_2 \overset{R}{\leftarrow} G \) (this implicitly defines \( m^* = c_2 / \text{CDH}(X, Y) \)), and sends \( c = (c_1, c_2) \)
- \( B \) receives \( m \) from \( A \) and outputs \( c_2 / m \)
- \( \text{Pr}[m = m^*] = \text{Succ}_{E_G}^{\text{OW−CPA}}(A) \)
  \( = \text{Pr}[c_2 / m = c_2 / m^*] = \text{Pr}[c_2 / m = \text{CDH}(X, Y)] \leq \text{Succ}_{G}^{\text{CDH}}(t) \)

---

**Outline**

1. Cryptography
2. Provable Security
3. Public-Key Encryption
   - One-Wayness
   - Indistinguishability
4. Conclusion
Is OW–CPA 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?

Semantic Security / Indistinguishability

After having chosen two plaintexts \( m_0 \) and \( m_1 \), upon receiving the encryption of \( m_b \) (for a random bit \( b \)), it should be hard to guess which message has been encrypted:

\[
\begin{align*}
(sk, pk) & \leftarrow \mathcal{K}(); (m_0, m_1, \text{state}) \leftarrow \mathcal{A}(pk); \\
 b & \overset{\$}{\leftarrow} \{0, 1\}; c = \mathcal{E}_{pk}(m_b); b' \leftarrow \mathcal{A}(.c, \text{state})
\end{align*}
\]

\[
\text{Adv}^{\text{ind–cpa}}_S(\mathcal{A}) = \Pr[b' = 1|b = 1] - \Pr[b' = 1|b = 0]
\]

ElGamal Encryption

The ElGamal encryption scheme \( \mathcal{E}_G \) is defined, in a group \( \mathbb{G} = \langle g \rangle \) of order \( q \), for \( m \in \mathbb{G} \):

- \( \mathcal{K}(\mathbb{G}, g, q): x \overset{\$}{\leftarrow} \mathbb{Z}_q \), and \( sk \leftarrow x \) and \( pk \leftarrow y = g^x \)
- \( \mathcal{E}_{pk}(m): r \overset{\$}{\leftarrow} \mathbb{Z}_q, c_1 \leftarrow g^r \) 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 \( c_2/c_1^{sk} \)

**Theorem (ElGamal is IND – CPA)**

\[
\text{Adv}^{\text{ind–cpa}}_{\mathcal{E}_G}(t) \leq 2 \times \text{Adv}^{\text{ddh}}_G(t)
\]

ElGamal is IND – CPA: Proof

Let \( \mathcal{A} \) be an adversary against \( \mathcal{E}_G \), we build an adversary \( \mathcal{B} \) against DDH: let us be given a DDH instance \((X, Y, Z)\)

- \( \mathcal{A} \) gets \( pk \leftarrow X \) from \( \mathcal{B} \), and outputs \( (m_0, m_1) \)
- \( \mathcal{B} \) sets \( c_1 \leftarrow Y \)
- \( \mathcal{B} \) chooses \( b \overset{\$}{\leftarrow} \{0, 1\} \), sets \( c_2 \leftarrow Z \times m_b \), and sends \( c = (c_1, c_2) \)
- \( \mathcal{B} \) receives \( b' \) from \( \mathcal{A} \) and outputs \( d = (b' = b) \)
- \( 2 \times \Pr[b' = b] - 1 = \text{Adv}^{\text{ind–cpa}}_{\mathcal{E}_G}(t), \text{ if } Z = \text{CDH}(X, Y) \)
  \( = 0, \text{ otherwise} \)
ElGamal is IND – CPA: Proof

As a consequence,

\[ 2 \times \Pr[b' = b | Z = CDH(X, Y)] - 1 = \text{Adv}^{\text{ind-cpa}}_{\mathcal{E}_G}(A) \]

\[ 2 \times \Pr[b' = b | Z \overset{R}{\leftarrow} \mathcal{G}] - 1 = 0 \]

\[ \text{Adv}^{\text{ind-cpa}}_{\mathcal{E}_G}(A) = 2 \times \left( \Pr[d = 1 | Z = CDH(X, Y)] - \Pr[d = 1 | Z \overset{R}{\leftarrow} \mathcal{G}] \right) \]

\[ = 2 \times \text{Adv}^{\text{ddh}}_{\mathcal{G}}(B) \leq 2 \times \text{Adv}^{\text{ddh}}_{\mathcal{G}}(t) \]

\[ \text{Theorem (RSA is OW – CPA, but...)} \]

\[ \text{Succ}^{\text{OW-CPA}}_{\mathcal{RSA}}(t) \leq \text{Succ}^{\text{rsa}}(t) \]

A deterministic encryption scheme cannot be IND – CPA

Outline

1 Cryptography
   ■ Introduction
   ■ Formal Notations

2 Provable Security
   ■ Definition
   ■ Computational Assumptions
   ■ Some Reductions

3 Public-Key Encryption
   ■ One-Wayness
   ■ Indistinguishability

4 Conclusion

Global methodology for provable security:

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

We will apply this methodology

■ on advanced security notions for encryption
■ to signature schemes