II – Encryption

David Pointcheval
Ecole normale supérieure, CNRS & INRIA

MPRI – Paris

Outline

1 Basic Security Notions
   - Public-Key Encryption
   - Signatures

2 Game-based Proofs
   - Provable Security
   - Game-based Approach
   - Transition Hops

3 Advanced Security for Encryption
   - Advanced Security Notions
   - Cramer-Shoup Encryption Scheme
   - Generic Conversion

4 Conclusion

Public-Key Encryption

Goal: Privacy/Secrecy of the plaintext
### OW – CPA Security Game

**Succ\text{OW}^S(A) = \Pr[(sk, pk) \leftarrow \mathcal{K}(); m \xrightarrow{R} \mathcal{M}; c = E_{pk}(m) : A(pk, c) \rightarrow m]**

### IND – CPA Security Game

**Adv\text{ind-}\text{cpa}^S(A) = \Pr[b' = 1|b = 1] - \Pr[b' = 1|b = 0] = 2 \times \Pr[b' = b] - 1**

### Signature

**Goal: Authentication of the sender**
### Provable Security

One can prove that:
- if an adversary is able to break the cryptographic scheme
- then one can break the underlying problem (integer factoring, discrete logarithm, 3-SAT, etc)

Unfortunately
- Security may rely on several assumptions
- Proving that the view of the adversary, generated by the simulator, in the reduction is the same as in the real attack game is not easy to do in such a one big step
Outline

1. Basic Security Notions
2. Game-based Proofs
   - Provable Security
   - Game-based Approach
   - Transition Hops
3. Advanced Security for Encryption
4. Conclusion

Sequence of Games

Real Attack Game
The adversary plays a game, against a challenger (security notion)

Simulation
The adversary plays a game, against a sequence of simulators
Sequence of Games

Simulation
The adversary plays a game, against a sequence of simulators

Output
- The output of the simulator in Game 1 is related to the output of the challenger in Game 0 (adversary's winning probability)
- The output of the simulator in Game 3 is easy to evaluate (e.g. always zero, always 1, probability of one-half)
- The gaps (Game 1 ↔ Game 2, Game 2 ↔ Game 3, etc) are clearly identified with specific events

Outline
1. Basic Security Notions
2. Game-based Proofs
   - Provable Security
   - Game-based Approach
   - Transition Hops
3. Advanced Security for Encryption
4. Conclusion

Two Simulators
- perfectly identical behaviors [Hop-S-Perfect]
- different behaviors, only if event $E_v$ happens
  - $E_v$ is negligible [Hop-S-Negl]
  - $E_v$ is non-negligible [Hop-S-Non-Negl]
  and independent of the output in Game$_A$
  → Simulator B terminates in case of event $E_v$
Two Simulations

- Identical behaviors: \( \Pr[\text{Game}_A] - \Pr[\text{Game}_B] = 0 \)
- The behaviors differ only if \( \text{Ev} \) happens:
  - \( \text{Ev} \) is negligible, one can ignore it
  - Shoup's Lemma: \( \Pr[\text{Game}_A] - \Pr[\text{Game}_B] \leq \Pr[\text{Ev}] \)

\[
|\Pr[\text{Game}_A] - \Pr[\text{Game}_B]| \\
= \Pr[\text{Game}_A|\text{Ev}] \Pr[\text{Ev}] + \Pr[\text{Game}_A|\neg\text{Ev}] \Pr[\neg\text{Ev}] \\
- \Pr[\text{Game}_B|\text{Ev}] \Pr[\text{Ev}] - \Pr[\text{Game}_B|\neg\text{Ev}] \Pr[\neg\text{Ev}] \\
= (\Pr[\text{Game}_A|\text{Ev}] - \Pr[\text{Game}_B|\text{Ev}]) \times \Pr[\text{Ev}] \\
+ (\Pr[\text{Game}_A|\neg\text{Ev}] - \Pr[\text{Game}_B|\neg\text{Ev}]) \times \Pr[\neg\text{Ev}] \\
\leq |1 \times \Pr[\text{Ev}] + 0 \times \Pr[\neg\text{Ev}]| \leq \Pr[\text{Ev}]
\]

- \( \text{Ev} \) is non-negligible and independent of the output in \( \text{Game}_A \), Simulator B terminates in case of event \( \text{Ev} \)

Two Simulations

- Identical behaviors: \( \Pr[\text{Game}_A] - \Pr[\text{Game}_B] = 0 \)
- The behaviors differ only if \( \text{Ev} \) happens:
  - \( \text{Ev} \) is negligible, one can ignore it
  - \( \text{Ev} \) is non-negligible and independent of the output in \( \text{Game}_A \), Simulator B terminates in case of event \( \text{Ev} \)

Event \( \text{Ev} \)

- Either \( \text{Ev} \) is negligible, or the output is independent of \( \text{Ev} \)
- For being able to terminate simulation B in case of event \( \text{Ev} \), this event must be efficiently detectable
- For evaluating \( \Pr[\text{Ev}] \), one re-iterates the above process, with an initial game that outputs 1 when event \( \text{Ev} \) happens
Two Distributions

Pr[Game_A] - Pr[Game_B] ≤ Adv(D^Oracles)

For identical/statistically close distributions, for any oracle:

Pr[Game_A] - Pr[Game_B] = Dist(Distrib_A, Distrib_B) = negl()

For computationally close distributions, in general, we need to exclude additional oracle access:

Pr[Game_A] - Pr[Game_B] ≤ Adv(Distrib(t))

where t is the computational time of the distinguisher

Outline

1. Basic Security Notions
2. Game-based Proofs
3. Advanced Security for Encryption
   - Advanced Security Notions
   - Cramer-Shoup Encryption Scheme
   - Generic Conversion
4. Conclusion

Public-Key Encryption

Goal: Privacy/Secrecy of the plaintext
**IND − CPA Security Game**

\[
\begin{aligned}
&b \in \{0,1\} \\
r \text{ random}
\end{aligned}
\]

\[
\begin{aligned}
&k_e \\
&G
\end{aligned}
\]

\[
\begin{aligned}
&m_0 \\
&m_1
\end{aligned}
\]

\[
\begin{aligned}
&c^* \\
&m_b
\end{aligned}
\]

\[
\begin{aligned}
&b' \overset{?}{=} b
\end{aligned}
\]

The adversary cannot get any information about a plaintext of a specific ciphertext (validity, partial value, etc)

**Malleability**

Semantic security (ciphertext indistinguishability) guarantees that no information is leaked from \(c\) about the plaintext \(m\). But it may be possible to derive a ciphertext \(c'\) such that the plaintext \(m'\) is related to \(m\) in a meaningful way:

- ElGamal ciphertext: \(c_1 = g^r\) and \(c_2 = m \times y^r\)
- Malleability: \(c_1' = c_1 = g^r\) and \(c_2' = 2 \times c_2 = (2m) \times y^r\)

From an encryption of \(m\), one can build an encryption of \(2m\), or a random ciphertext of \(m\), etc.

**Non-Malleability: NM − CPA Security Game**

\[
\begin{aligned}
&m^*, m' \leftarrow \mathcal{D} \\
r \text{ random}
\end{aligned}
\]

\[
\begin{aligned}
&k_e \\
&G
\end{aligned}
\]

\[
\begin{aligned}
&m^* \\
&m_r
\end{aligned}
\]

\[
\begin{aligned}
&c^* \\
&m^* \leftarrow \mathcal{E}
\end{aligned}
\]

\[
\begin{aligned}
&m = \mathcal{D}(c)
\end{aligned}
\]

The adversary cannot get any information about a plaintext of a specific ciphertext (validity, partial value, etc)

**Additional Information**

More information modelled by oracle access

- reaction attacks: oracle which answers, on \(c\), whether the ciphertext \(c\) is valid or not
- plaintext-checking attacks: oracle which answers, on a pair \((m, c)\), whether the plaintext \(m\) is really encrypted in \(c\) or not (whether \(m = \mathcal{D}_{sk}(c)\))
- chosen-ciphertext attacks (CCA): decryption oracle (with the restriction not to use it on the challenge ciphertext) \(\rightarrow\) the adversary can obtain the plaintext of any ciphertext of its choice (excepted the challenge)
  - non-adaptive (CCA − 1)
  - adaptive (CCA − 2)

[Naor-Yung – STOC ’90]

[Rackoff-Simon – Crypto ’91]
The adversary can ask any decryption of its choice:
Chosen-Ciphertext Attacks (oracle access)

\[(sk, pk) \leftarrow \mathcal{K}(); (m_0, m_1, \text{state}) \leftarrow \mathcal{A}^D(pk);
\]
\[b \leftarrow \{0, 1\}; c = \mathcal{E}_{pk}(m_b); b' \leftarrow \mathcal{A}^D(\text{state}, c)\]

\[\text{Adv}^{\text{ind--cca}}_S(A) = \Pr[b' = 1|b = 1] - \Pr[b' = 1|b = 0] = 2 \times \Pr[b' = b] - 1\]

Outline

1. Basic Security Notions
2. Game-based Proofs
3. Advanced Security for Encryption
   - Advanced Security Notions
   - Cramer-Shoup Encryption Scheme
   - Generic Conversion
4. Conclusion

Cramer-Shoup Encryption Scheme

**Key Generation**

- \(G = \langle g \rangle, x \rangle \) group of order \(q\)
- \(sk = (x_1, x_2, y_1, y_2, z), \) where \(x_1, x_2, y_1, y_2, z \leftarrow \mathbb{Z}_q\)
- \(pk = (g_1, g_2, \mathcal{H}, c, d, h), \) where
  - \(g_1, g_2\) are independent elements in \(G\)
  - \(\mathcal{H}\) a hash function (second-preimage resistant)
  - \(c = g_1^{x_1} g_2^{x_2}, d = g_1^{y_1} g_2^{y_2}, \) and \(h = g_1^z\)

**Encryption**

\[u_1 = g_1^x, u_2 = g_2^y, e = m \times h^r, \quad v = c^d r^\alpha \] where \(\alpha = \mathcal{H}(u_1, u_2, e)\)
Cramer-Shoup Encryption Scheme vs. ElGamal

\[ u_1 = g_1^t, \ u_2 = g_2^t, \ e = m \times h^r, \ v = c^r d^\alpha \ \text{where} \ \alpha = H(u_1, u_2, e) \]

(u_1, e) is an ElGamal ciphertext, with public key \( h = g_1^z \)

Decryption

- since \( h = g_1^z, \ h^r = u_1^r \), thus \( m = e/u_1^r \)
- since \( c = g_1^{x_1} g_2^{x_2} \) and \( d = g_1^{y_1} g_2^{y_2} \)

\[ c' = g_1^{x_1} g_2^{x_2} = u_1^{x_1} u_2^{x_2} \quad d' = u_1^{y_1} u_2^{y_2} \]

One thus first checks whether

\[ v = u_1^{x_1+\alpha y_1} u_2^{x_2+\alpha y_2} \ \text{where} \ \alpha = H(u_1, u_2, e) \]

Security of the Cramer-Shoup Encryption Scheme

Theorem

The Cramer-Shoup encryption scheme achieves IND −CCA security, under the DDH assumption, and the second-preimage resistance of \( H \):

\[ \text{Adv}^\text{ind−cca}_{CS}(t) \leq 2 \times \text{Adv}^\text{ddh}_{G}(t) + \text{Succ}^H(t) + 3q_D/q \]

Let us prove this theorem, with a sequence of games, in which \( \mathcal{A} \) is an IND − CCA adversary against the Cramer-Shoup encryption scheme.

Real Attack Game

Game 0

Key Generation Oracle

\[ x_1, x_2, y_1, y_2, z \stackrel{R}{\rightarrow} \mathbb{Z}, \ g_1, g_2 \stackrel{R}{\rightarrow} \mathbb{G}; \ sk = (x_1, x_2, y_1, y_2, z) \]
\[ c = g_1^{x_1} g_2^{x_2}, \ d = g_1^{y_1} g_2^{y_2}, \ \text{and} \ h = g_1^z; \ \text{pk} = (g_1, g_2, H, c, d, h) \]

Decryption Oracle

If \( v = u_1^{x_1+\alpha y_1} u_2^{x_2+\alpha y_2} \ \text{where} \ \alpha = H(u_1, u_2, e) \): \( m = e/u_1^z \)

Proof: Invalid ciphertexts

- **Game 0**: use of the oracles \( \mathcal{K}, \mathcal{D} \)
- **Game 1**: we abort (with a random output \( b' \))

\[ \text{in case of bad (invalid) accepted ciphertext,} \]
\[ \text{where invalid ciphertext means } \log_{g_1} u_1 \neq \log_{g_2} u_2 \]

Event F

\( \mathcal{A} \) submits a bad accepted ciphertext

(note: this is not computationally detectable)

The advantage in **Game 1** is:

\[ \Pr_{\text{Game}_0}[b' = b|F] = \frac{1}{2} \]
\[ \Pr_{\text{Game}_1}[F] = \Pr_{\text{Game}_1}[b' = b|\neg F] = \Pr_{\text{Game}_0}[b' = b|\neg F] \]

\[ \Rightarrow \text{Hop-S-Negl: } \text{Adv}_{\text{Game}_1} \geq \text{Adv}_{\text{Game}_0} - \Pr[F] \]
Proof: Simulations

\[ \text{Game}_2: \text{we use the simulations} \]

**Key Generation Simulation**

\[ x_1, x_2, y_1, y_2, z_1, z_2 \overset{R}{\leftarrow} \mathbb{Z}_q, g_1, g_2 \overset{R}{\leftarrow} \mathbb{G}: \ sk = (x_1, x_2, y_1, y_2, z_1, z_2) \]

\[ c = g_1^{x_1} g_2^{x_2}, d = g_1^{y_1} g_2^{y_2}, \text{and} \ h = g_1^z g_2^{z_2}: \ pk = (g_1, g_2, \mathcal{H}, c, d, h) \]

Distribution of the public key: Identical

**Decryption Simulation**

\[ v = u_1^{x_1 + \alpha y_1} u_2^{x_2 + \alpha y_2} \text{ where } \alpha = \mathcal{H}(u_1, u_2, e): \ m = e/u_1^{z_1} u_2^{z_2} \]

Under the assumption of \( \neg F \), perfect simulation

\[ \Rightarrow \text{Hop-S-Perfect: } \text{Adv}_{\text{Game}_2} = \text{Adv}_{\text{Game}_1} \]

Proof: Computable Adversary

\[ \text{Game}_3: \text{we do no longer exclude bad accepted ciphertexts} \]

\[ \Rightarrow \text{Hop-S-Negl:} \]

\[ \text{Adv}_{\text{Game}_3} \geq \text{Adv}_{\text{Game}_2} - \text{Pr}[F] \geq \text{Adv}_{\text{Game}_2} - q_D/q \]

This is technical: to make the simulator/adversary computable.

Details: Shoup’s Lemma

\[ \text{Adv}_{\text{Game}_1} = 2 \times \text{Pr}_{\text{Game}_1}[b' = b] - 1 \]

\[ = 2 \times \text{Pr}_{\text{Game}_1}[b' = b|\neg F] \text{Pr}_{\text{Game}_1}[\neg F] + 2 \times \text{Pr}_{\text{Game}_1}[b' = b|F] \text{Pr}_{\text{Game}_1}[F] + \text{Pr}_{\text{Game}_0}[F] - 1 \]

\[ = 2 \times \text{Pr}_{\text{Game}_0}[b' = b|\neg F] \text{Pr}_{\text{Game}_0}[\neg F] + \text{Pr}_{\text{Game}_0}[F] - 1 \]

\[ = \text{Adv}_{\text{Game}_0} - \text{Pr}_{\text{Game}_0}[F](2 \times \text{Pr}_{\text{Game}_0}[b' = b|F] - 1) \]

\[ \geq \text{Adv}_{\text{Game}_0} - \text{Pr}_{\text{Game}_0}[F] \]

Details: Bad Accept

In order to evaluate \( \text{Pr}[F] \), we study the probability that

- \( r_1 = \log g_1 u_1 \neq \log g_2 u_2 = r_2, \)
- whereas \( v = u_1^{x_1 + \alpha y_1} u_2^{x_2 + \alpha y_2} \)

The adversary just knows the public key:

\[ c = g_1^{x_1} g_2^{x_2}, d = g_1^{y_1} g_2^{y_2} \]

Let us move to the exponents, in basis \( g_1, \) with \( g_2 = g_1^s: \)

\[ \log c = x_1 + sx_2 \]
\[ \log d = y_1 + sy_2 \]
\[ \log v = r_1(x_1 + \alpha y_1) + sr_2(x_2 + \alpha y_2) \]

The system is under-defined: for any \( v \), there are \( (x_1, x_2, y_1, y_2) \) that satisfy the system \( \Rightarrow v \) is unpredictable

\[ \Rightarrow \text{Pr}[F] \leq q_D/q \]

\[ \Rightarrow \text{Adv}_{\text{Game}_1} \geq \text{Adv}_{\text{Game}_0} - q_D/q \]
Proof: DDH Assumption

**Game₄**: we modify the generation of the challenge ciphertext:

<table>
<thead>
<tr>
<th>Original Challenge</th>
</tr>
</thead>
</table>
| Random choice: \( b \overset{\text{R}}{\leftarrow} \{0,1\}, r \overset{\text{R}}{\leftarrow} \mathbb{Z}_q \)  
| \([\alpha = \mathcal{H}(u_1, u_2, e)]\)  
| \( u_1 = g_1^b, u_2 = g_2^r, e = m_b \times h', v = c' d'^{\alpha} \)  

<table>
<thead>
<tr>
<th>New Challenge 1</th>
</tr>
</thead>
</table>
| Given \( (U = g_1^b, V = g_2^r) \) from outside, and random choice \( b \overset{\text{R}}{\leftarrow} \{0,1\} \)  
| \( u_1 = U, u_2 = V, e = m_b \times U^{z_1} V^{z_2}, v = U^{x_1+\alpha y_1} V^{x_2+\alpha y_2} \)  
| With \( (U = g_1^b, V = g_2^r): U^{z_1} V^{z_2} = h' \) and \( U^{x_1+\alpha y_1} V^{x_2+\alpha y_2} = c' d'^{\alpha} \)  
| \( \implies \text{Hop-S-Perfect: } \text{Adv}_{\text{Game}_4} = \text{Adv}_{\text{Game}_3} \)  

Proof: DDH Assumption

The input from outside changes from \( (U = g_1^b, V = g_2^r) \) (a CDH tuple) to \( (U = g_1^{b'}, V = g_2^{c'}) \) (a random tuple):

\[
\text{Pr}_{\text{Game}_4}[b' = b] - \text{Pr}_{\text{Game}_5}[b' = b] \leq \text{Adv}_{\text{ddh}}^G(t)
\]

\( \implies \text{Hop-D-Comp: } \text{Adv}_{\text{Game}_5} \geq \text{Adv}_{\text{Game}_4} - 2 \times \text{Adv}_{\text{ddh}}^G(t) \)  
(Since \( \text{Adv} \geq 2 \times \text{Pr}[b' = b] - 1 \))

Proof: DDH Assumption

**Game₅**: we modify the generation of the challenge ciphertext:

<table>
<thead>
<tr>
<th>Previous Challenge 1</th>
</tr>
</thead>
</table>
| Given \( (U = g_1^b, V = g_2^r) \) from outside, and random choice \( b \overset{\text{R}}{\leftarrow} \{0,1\} \)  
| \( u_1 = U, u_2 = V, e = m_b \times U^{z_1} V^{z_2}, v = U^{x_1+\alpha y_1} V^{x_2+\alpha y_2} \)  

<table>
<thead>
<tr>
<th>New Challenge 2</th>
</tr>
</thead>
</table>
| Given \( (U = g_1^b, V = g_2^r) \) from outside, and random choice \( b \overset{\text{R}}{\leftarrow} \{0,1\} \)  
| \( u_1 = U, u_2 = V, e = m_b \times U^{z_1} V^{z_2}, v = U^{x_1+\alpha y_1} V^{x_2+\alpha y_2} \)  
| The input changes from \( (U = g_1^b, V = g_2^r) \) to \( (U = g_1^{b'}, V = g_2^{c'}) \):  
| \( \implies \text{Hop-D-Comp: } \text{Adv}_{\text{Game}_5} \geq \text{Adv}_{\text{Game}_4} - 2 \times \text{Adv}_{\text{ddh}}^G(t) \)  

Proof: Collision

**Game₆**: we abort (with a random output \( b' \)) in case of second pre-image with a decryption query

<table>
<thead>
<tr>
<th>Event ( F_H )</th>
</tr>
</thead>
</table>
| \( \mathcal{A} \) submits a ciphertext with the same \( \alpha \) as the challenge ciphertext, but a different initial triple: \( (u_1, u_2, e) \neq (u_1^*, u_2^*, e^*)\), but \( \alpha = \alpha^* \), were \( ^* \) are for all the elements related to the challenge ciphertext.

Second pre-image of \( \mathcal{H} \):  
\( \implies \text{Pr}[F_H] \leq \text{Succ}^\mathcal{H}(t) \)

The advantage in **Game₆** is:  
\[
\text{Pr}_{\text{Game}_6}[b' = b | F_H] = 1/2
\]

\[
\implies \text{Hop-S-Negl: } \text{Adv}_{\text{Game}_6} \geq \text{Adv}_{\text{Game}_5} - \text{Pr}[F_H] \]

\[
\text{Adv}_{\text{Game}_6} \geq \text{Adv}_{\text{Game}_5} - \text{Succ}^H(t)
\]
Details: Bad Accept

The adversary knows the public key, and the (invalid) challenge ciphertext:

\[ c = g_1^{x_1} g_2^{x_2} \quad d = g_1^{y_1} g_2^{y_2} \]

\[ v^* = U^{x_1 + \alpha y_1} V^{x_2 + \alpha y_2} = g_1^{r_1^* (x_1 + \alpha y_1)} g_2^{r_2^* (x_2 + \alpha y_2)} \]

Let us move to the exponents, in basis \( g_1 \), with \( g_2 = g_1^s \):

\[ \log c = x_1 + s x_2 \]
\[ \log d = y_1 + s y_2 \]
\[ \log v^* = r_1^* (x_1 + \alpha y_1) + s r_2^* (x_2 + \alpha y_2) \]
\[ \log v = r_1 (x_1 + \alpha y_1) + s r_2 (x_2 + \alpha y_2) \]

The determinant of the system is

\[ \Delta = \begin{vmatrix} 1 & s & 0 & 0 \\ 0 & 0 & 1 & s \\ r_1^* & s r_2^* & r_1^* \alpha^* & s r_2^* \alpha^* \\ r_1 & s r_2 & r_1 \alpha & s r_2 \alpha \end{vmatrix} \]

\[ = s^2 \times ((r_2 - r_1) \times (r_2^* - r_1^*) \times \alpha^* - (r_2^* - r_1^*) \times (r_2 - r_1) \times \alpha) \]
\[ = s^2 \times (r_2 - r_1) \times (r_2^* - r_1^*) \times (\alpha^* - \alpha) \]
\[ \neq 0 \]

The system is under-defined:

for any \( v \), there are \((x_1, x_2, y_1, y_2)\) that satisfy the system

\[ \implies v \text{ is unpredictable} \implies \Pr[F'_3] \leq q_D/q \]

\[ \implies \text{Adv}_{\text{Game}_7} \geq \text{Adv}_{\text{Game}_6} - q_D/q \]
Proof: Analysis of the Final Game

In the final Game7:

- only valid ciphertexts are decrypted
- the challenge ciphertext contains
  \[ e = m_b \times U^{z_1} V^{z_2} \]

- the public key contains
  \[ h = g_1^{z_1} g_2^{z_2} \]

Again, the system is under-defined:
for any \( m_b \), there are \( (z_1, z_2) \) that satisfy the system
\[ \implies m_b \text{ is unpredictable} \quad \implies b \text{ is unpredictable} \]
\[ \implies \text{Adv}_{\text{Game}_7} = 0 \]

\[ \text{Adv}_{\text{Game}_7} = 0 \]
\[ \text{Adv}_{\text{Game}_7} \geq \text{Adv}_{\text{Game}_6} - \frac{q_D}{q} \]
\[ \text{Adv}_{\text{Game}_6} \geq \text{Adv}_{\text{Game}_5} - \text{Succ}_H(t) \]
\[ \text{Adv}_{\text{Game}_5} \geq \text{Adv}_{\text{Game}_4} - 2 \times \text{Adv}_{\text{ddh}}^G(t) \]
\[ \text{Adv}_{\text{Game}_4} = \text{Adv}_{\text{Game}_3} \]
\[ \text{Adv}_{\text{Game}_3} \geq \text{Adv}_{\text{Game}_2} - \frac{q_D}{q} \]
\[ \text{Adv}_{\text{Game}_2} = \text{Adv}_{\text{Game}_1} \]
\[ \text{Adv}_{\text{Game}_1} \geq \text{Adv}_{\text{Game}_0} - \frac{q_D}{q} \]
\[ \text{Adv}_{\text{Game}_0} = \text{Adv}_{\text{CS}}^\text{ind-cca}(A) \]
\[ \text{Adv}_{\text{CS}}^\text{ind-cca}(A) \leq 2 \times \text{Adv}_{\text{ddh}}^G(t) + \text{Succ}_H(t) + 3\frac{q_D}{q} \]

Conclusion

First Generic Conversion

For efficiency: random oracle model

Setup

- A trapdoor one-way permutation family \( \{(f, g)\} \) onto the set \( X \)
- Two hash functions, for the security parameter \( k_1 \),
  \[ \mathcal{G} : X \rightarrow \{0, 1\}^n \quad \text{and} \quad \mathcal{H} : \{0, 1\}^* \rightarrow \{0, 1\}^{k_1}, \]
  where \( n \) is the bit-length of the plaintexts.

Key Generation

One chooses a random element in the family

- \( f \) is the public key
- the inverse \( g \) is the private key

Outline

1. Basic Security Notions
2. Game-based Proofs
3. Advanced Security for Encryption
   - Advanced Security Notions
   - Cramer-Shoup Encryption Scheme
   - Generic Conversion
4. Conclusion
First Generic Conversion (Cont’ed)

Encryption
One chooses a random element \( r \in X \)
\[ a = f(r), \quad b = m \oplus G(r), \quad c = \mathcal{H}(m, r) \]

Decryption
Given \((a, b, c)\), and the private key \( g \),
- one first recovers \( r = g(a) \)
- one gets \( m = b \oplus G(r) \)
- one then checks whether \( c \overset{?}{=} \mathcal{H}(m, r) \)
If the equality holds, one returns \( m \), otherwise one rejects the ciphertext

Real Attack Game

Simulation of the Random Oracles

Game 0: use of the perfect oracles
Challenge Ciphertext
Random \( r \), random bit \( b \): \( a = f(r), \ b = m_b \oplus G(r) \), \( c = \mathcal{H}(m, r) \)
\[ \text{Adv}_{\text{Game}_0} = 2 \times \text{Pr}_{\text{Game}_0} [b' = b] - 1 = \varepsilon \]

Game 1: use of the simulation of the random oracles
Random Oracles
For any new query, a new random output: management of lists

Security of the Bellare-Rogaway Conversion

Theorem
The Bellare-Rogaway conversion achieves \( \text{IND} \rightarrow \text{CCA} \) security, under the one-wayness of the trapdoor permutation \( f \):
\[ \text{Adv}^\text{ind-cca}_{\text{BR}}(t) \leq 2 \times \text{Succ}^\text{ow}_{f}(T) + \frac{4q_D^2}{2^{k_1}}, \]
where \( T \leq t + (q_G + q_H) \cdot T_f \).

Let us prove this theorem, with a sequence of games, in which \( A \) is an \( \text{IND} \rightarrow \text{CCA} \) adversary against the Bellare-Rogaway conversion.
Simulation of the Challenge Ciphertext

- **Game**\(_2\): use of an independent random value \(h^+\)

**Challenge Ciphertext**

Random \(r\), random bit \(b\): \(a = f(r), b = m_b \oplus \mathcal{G}(r), c = h^+\)

This game is indistinguishable from the previous one, unless \((m_b, r)\) is queried to \(\mathcal{H}\): event \textbf{AskMR} (it can only be asked by the adversary, since such a query by the decryption oracle would be for the challenge ciphertext).

Note that in case of \textbf{AskMR}, we stop the simulation with a random output:

\[
\text{Adv}_{\text{Game}_2} = \text{Adv}_{\text{Game}_1} - 2 \times \Pr_{\text{Game}_1}[\text{AskMR}]
\]

Simulation of the Decryption Oracle

- **Game**\(_3\): reject if \((m, r)\) not queried to \(\mathcal{H}\)

**Decryption Oracle**

Look in the \(\mathcal{H}\)-list for \((m, r)\) such that \(c = \mathcal{H}(m, r)\).

If not found: reject,

if for one pair, \(a = f(r)\) and \(b = m \oplus \mathcal{G}(r)\), output \(m\)

This makes a difference if this value \(c\), without having been asked to \(\mathcal{H}\), is correct: for each attempt, the probability is bounded by \(1/2^k\):

\[
\text{Adv}_{\text{Game}_3} \geq \text{Adv}_{\text{Game}_2} - 2qD/2^{k_1}
\]

\[
\Pr_{\text{Game}_3}[\text{AskMR}] \geq \Pr_{\text{Game}_2}[\text{AskMR}] - qD/2^{k_1}
\]

Simulation of the Challenge Ciphertext

- **Game**\(_4\): use of an independent random value \(g^+\) (and \(h^+\))

**Challenge Ciphertext**

Random \(r\), random bit \(b\): \(a = f(r), b = m_b \oplus g^+, c = h^+\)

This game is indistinguishable from the previous one, unless \(r\) is queried to \(\mathcal{G}\) by the adversary or by the decryption oracle. We denote by \textbf{AskR} the event that \(r\) is asked to \(\mathcal{G}\) or \(\mathcal{H}\) by the adversary (which includes \textbf{AskMR}). But \(r\) cannot be asked to \(\mathcal{G}\) by the decryption oracle without \textbf{AskR}: only possible if \(r\) is in the \(\mathcal{H}\)-list, and thus asked by the adversary:

\[
\text{Adv}_{\text{Game}_4} = \text{Adv}_{\text{Game}_3} - 2 \times \Pr_{\text{Game}_3}[\text{AskR}|\neg\text{AskMR}]
\]

\[
\Pr_{\text{Game}_4}[\text{AskR}] \leq \Pr_{\text{Game}_3}[\text{AskMR}] + \Pr_{\text{Game}_3}[\text{AskR}|\neg\text{AskMR}]
\]

Simulation of the Challenge Ciphertext

- **Game**\(_5\): use of an independent random value \(a^+\) (and \(g^+, h^+\))

**Challenge Ciphertext**

Random bit \(b\): \(a = a^+, b = m_b \oplus g^+, c = h^+\)

This determines \(r\), the unique value such that \(a^+ = f(r)\), which allows to detect event \textbf{AskR}.

This game is perfectly indistinguishable from the previous one:

\[
\text{Adv}_{\text{Game}_5} = \text{Adv}_{\text{Game}_4}
\]

\[
\Pr_{\text{Game}_5}[\text{AskR}] = \Pr_{\text{Game}_4}[\text{AskR}]
\]
Inversion of the Permutation

Since we can assume that $a^+$ is a given challenge for inverting the permutation $f$, when one looks in the $G$-list or the $H$-list, one can find $r$, the pre-image of $a^+$:

$$\Pr_{\text{Game}_5}[\text{AskR}] \leq \text{Succ}^\text{OW}_f(t + (q_G + q_H) \cdot T_f)$$

But clearly, in the last game, because of $g^+$ that perfectly hides $m_b$:

$$\text{Adv}_{\text{Game}_5} = 0$$

As a consequence, $0 = \text{Adv}_{\text{Game}_5}$

$$= \text{Adv}_{\text{Game}_4} \geq \text{Adv}_{\text{Game}_3} - 2 \times \Pr_{\text{Game}_3}[\text{AskR}|\neg \text{AskMR}]$$

$$\geq \text{Adv}_{\text{Game}_2} - 2 \times \Pr_{\text{Game}_2}[\text{AskMR}] - 2 \times \Pr_{\text{Game}_3}[\text{AskR}|\neg \text{AskMR}] - 2q_D/2^{k_1}$$

$$\geq \text{Adv}_{\text{Game}_1} - 2 \times \Pr_{\text{Game}_1}[\text{AskMR}] - 2 \times \Pr_{\text{Game}_3}[\text{AskR}|\neg \text{AskMR}] - 4q_D/2^{k_1}$$

And then,

$$\text{Adv}_{\text{Game}_0} \leq 4q_D/2^{k_1} + 2 \times \text{Succ}^\text{OW}_f(T)$$

Conclusion

Game-based Methodology: the story of OAEP

- Reduction proven indistinguishable for an IND-CCA adversary (actually IND-CCA1, and not IND-CCA2) but widely believed for IND-CCA2, without any further analysis of the reduction

  The direct-reduction methodology

- [Shoup - Crypto ’01]
  - Shoup showed the gap for IND-CCA2, under the OWP
  - Granted his new game-based methodology

- [Fujisaki-Okamoto-Pointcheval-Stern – Crypto ’01]
  - FOPS proved the security for IND-CCA2, under the PD-OWP
  - Using the game-based methodology

Outline

1 Basic Security Notions
   - Public-Key Encryption
   - Signatures

2 Game-based Proofs
   - Provable Security
   - Game-based Approach
   - Transition Hops

3 Advanced Security for Encryption
   - Advanced Security Notions
   - Cramer-Shoup Encryption Scheme
   - Generic Conversion

4 Conclusion