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**

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

**Outline**

1. Basic Security Notions
   - Public-Key Encryption
   - Signatures
2. Game-based Proofs
3. Advanced Security for Encryption
4. Conclusion

**Signature**

Goal: Authentication of the sender
Succ$^{\text{euf}}_{SG}(A) = \Pr[(sk, pk) \leftarrow \mathcal{K}(); (m, \sigma) \leftarrow A(pk) : V_{pk}(m, \sigma) = 1]$

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

Game 0
Oracles
Challenger
Adversary
0 / 1

Game 1
Distribution 1
Simulator 1
Game 1
0 / 1

Game 2
Distribution 2
Simulator 2
Game 2
0 / 1
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
- different behaviors, only if event $E_v$ happens
  - $E_v$ is negligible
  - $E_v$ is non-negligible and independent of the output in $Game_A$
  - $\rightarrow$ 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
  - $\text{Ev}$ is non-negligible and independent of the output in $\text{Game}_A$, Simulator $B$ terminates and outputs 0, in case of event $\text{Ev}$:

\[
\Pr[\text{Game}_B] = \Pr[\text{Game}_B|\text{Ev}] \Pr[\text{Ev}] + \Pr[\text{Game}_B|\neg\text{Ev}] \Pr[\neg\text{Ev}]
\]
\[
= 0 \times \Pr[\text{Ev}] + \Pr[\text{Game}_A|\neg\text{Ev}] \times \Pr[\neg\text{Ev}]
\]
\[
= \Pr[\text{Game}_A] \times \Pr[\neg\text{Ev}]
\]

- Simulator $B$ terminates and flips a coin, in case of event $\text{Ev}$:

\[
\Pr[\text{Game}_B] = \Pr[\text{Game}_B|\text{Ev}] \Pr[\text{Ev}] + \Pr[\text{Game}_B|\neg\text{Ev}] \Pr[\neg\text{Ev}]
\]
\[
= \frac{1}{2} \times \Pr[\text{Ev}] + \Pr[\text{Game}_A|\neg\text{Ev}] \times \Pr[\neg\text{Ev}]
\]
\[
= \frac{1}{2} + \left(\Pr[\text{Game}_A] - \frac{1}{2}\right) \times \Pr[\neg\text{Ev}]
\]

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

\[ \Pr[\text{Game}_A] - \Pr[\text{Game}_B] \leq \text{Adv}(D_{\text{oracles}}) \]

- For identical/statistically close distributions, for any oracle:

\[ \Pr[\text{Game}_A] - \Pr[\text{Game}_B] = \text{Dist}(\text{Distrib}_A, \text{Distrib}_B) = \text{negl}() \]

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

\[ \Pr[\text{Game}_A] - \Pr[\text{Game}_B] \leq \text{Adv}_{\text{Distrib}}(t) \]

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**

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**

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 $m = D_{sk}(c)$).
- 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): only before receiving the challenge
- adaptive (CCA − 2): unlimited oracle access

$Adv_{s}^{nm-cpa}(A) = \Pr[\mathcal{R}(m^\ast, m)] - \Pr[\mathcal{R}(m', m)]$

[Naor-Yung – STOC ’90]

[Rackoff-Simon – Crypto ’91]
The adversary can ask any decryption of its choice:

Chosen-Ciphertext Attacks (oracle access)

\[
( sk, pk ) \leftarrow \mathcal{K}(); ( m_0, m_1, \text{state} ) \leftarrow A_D(pk);
\]

\[
b \leftarrow \{0,1\}; c = \mathcal{E}_{pk}( m_b ); b' \leftarrow A_D( \text{state}, c )
\]

\[
\text{Adv}_{\text{ind-cca}}( A ) = \Pr[ b' = 1 | b = 1 ] - \Pr[ b' = 1 | b = 0 ] = 2 \times \Pr[ b' = b ] - 1
\]
Cramer-Shoup Encryption Scheme vs. ElGamal

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

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

**Decryption**

- if \( h = g_1^t, \ h^t = u_1^t \), thus \( m = e / u_1^t \)
- if \( c = g_1^{x_1} g_2^{x_2} \) and \( d = g_1^{y_1} g_2^{y_2} \)
  \[ c' = g_1^{rx_1} g_2^{rx_2} = u_1^{x_1} u_2^{x_2} \quad d' = u_1^{y_1} u_2^{y_2} \]

One thus first checks whether

\[ v = u_1^{x_1 + \alpha y_1} u_2^{x_2 + \alpha y_2} \] 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 \( H \):

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

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**

**Key Generation Oracle**

\[ x_1, x_2, y_1, y_2, z \overset{\$}{\leftarrow} \mathbb{Z}_q, \ g_1, g_2 \overset{\$}{\leftarrow} \mathbb{G}; \ sk = (x_1, x_2, y_1, y_2, z) \]

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

**Decryption Oracle**

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

**Proof: Invalid ciphertexts**

- **Game\(_0\)**: use of the oracles \( K, D \)
- **Game\(_i\)**: we abort (with a random output \( b' \)) in case of bad (invalid) accepted ciphertext, 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\(_i\)** is:

\[ \Pr_{\text{Game}_i}[b' = b \mid F] = 1/2 \]

\[ \Pr_{\text{Game}_i}[b' = b \mid \neg F] = \Pr_{\text{Game}_0}[b' = b \mid \neg F] \]

\( \Rightarrow \text{Hop-S-Negl}: \text{Adv}_{\text{Game}_i} \geq \text{Adv}_{\text{Game}_0} - \Pr[F] \)
In order to evaluate $\Pr[F]$, 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:

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

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

$$\begin{align*}
\log c &= x_1 + sx_2 \\
\log d &= y_1 + sy_2 \\
\log v &= r_1(x_1 + \alpha y_1) + sr_2(x_2 + \alpha y_2)
\end{align*}$$

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 $\Rightarrow \Pr[F] \leq q_D/q \quad \Rightarrow \text{Adv}_{\text{Game}_1} \geq \text{Adv}_{\text{Game}_0} - q_D/q$
**Proof: DDH Assumption**

**Game$_3$**: we modify the generation of the challenge ciphertext:

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

<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 $b \overset{R}{\leftarrow} \{0, 1\}$  
  $u_1 = U, u_2 = V, e = m_b \times U^{z_1} V^{z_2}$, $v = U^{x_1+\alpha y_1} V^{x_2+\alpha y_2}$ |

With $(U = g_1^b, V = g_2^r)$: $U^{z_1} V^{z_2} = h'$ and $U^{x_1+\alpha y_1} V^{x_2+\alpha y_2} = c' d'^\alpha$

$\Rightarrow$ **Hop-S-Perfect**: $\text{Adv}_{\text{Game}_3} = \text{Adv}_{\text{Game}_1}$

**Proof: DDH Assumption**

The input from outside changes from $(U = g_1^b, V = g_2^r)$ (a CDH tuple) to $(U = g_1^r, V = g_2^g)$ (a random tuple):

$$\text{Pr}[b' = b] - \text{Pr}[b' = b] \leq \text{Adv}_{\text{Game}_1}^{\text{ddh}}(t)$$

$\Rightarrow$ **Hop-D-Comp**: $\text{Adv}_{\text{Game}_5} \geq \text{Adv}_{\text{Game}_1} - 2 \times \text{Adv}_{\text{G}}^{\text{ddh}}(t)$

(Since $\text{Adv} = 2 \times \text{Pr}[b' = b] - 1$)

---

**Proof: DDH Assumption**

**Game$_5$**: 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 $b \overset{R}{\leftarrow} \{0, 1\}$  
  $u_1 = U, u_2 = V, e = m_b \times U^{z_1} V^{z_2}$, $v = U^{x_1+\alpha y_1} V^{x_2+\alpha y_2}$ |

<table>
<thead>
<tr>
<th>New Challenge 2</th>
</tr>
</thead>
</table>
| Given $(U = g_1^r, V = g_2^g)$ from outside, and random choice $b \overset{R}{\leftarrow} \{0, 1\}$  
  $u_1 = U, u_2 = V, e = m_b \times U^{z_1} V^{z_2}$, $v = U^{x_1+\alpha y_1} V^{x_2+\alpha y_2}$ |

The input changes from $(U = g_1^b, V = g_2^r)$ to $(U = g_1^r, V = g_2^g)$:

$\Rightarrow$ **Hop-D-Comp**: $\text{Adv}_{\text{Game}_5} \geq \text{Adv}_{\text{Game}_1} - 2 \times \text{Adv}_{\text{G}}^{\text{ddh}}(t)$

---

**Proof: Collision**

<table>
<thead>
<tr>
<th>Game$_6$</th>
</tr>
</thead>
</table>

(Since $\text{Adv} = 2 \times \text{Pr}[b' = b] - 1$)
Details: Bad Accept (Case 3)

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

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

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

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

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

Details: Bad Accept

The system is under-defined:
For any \( \nu \), there are \( (x_1, x_2, y_1, y_2) \) that satisfy the system

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

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

In the final Game_{7}:
- 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
\[ \implies m_b \text{ is unpredictable} \quad \implies b \text{ is unpredictable} \]
\[ \implies Adv_{Game_{7}} = 0 \]

Conclusion

\[
\begin{align*}
Adv_{Game_{7}} &= 0 \\
Adv_{Game_{7}} &\geq Adv_{Game_{6}} - \frac{qD}{q} \\
Adv_{Game_{6}} &\geq Adv_{Game_{5}} - \text{Succ}_H(t) \\
Adv_{Game_{5}} &\geq Adv_{Game_{4}} - 2 \times Adv_{ddh}(t) \\
Adv_{Game_{4}} &= Adv_{Game_{3}} \\
Adv_{Game_{3}} &\geq Adv_{Game_{2}} - \frac{qD}{q} \\
Adv_{Game_{2}} &= Adv_{Game_{1}} \\
Adv_{Game_{1}} &\geq Adv_{Game_{0}} - \frac{qD}{q} \\
Adv_{Game_{0}} &= Adv_{\text{ind-cca}}(A) \\
Adv_{\text{ind-cca}}(A) &\leq 2 \times Adv_{ddh}(t) + \text{Succ}_H(t) + 3\frac{qD}{q}
\end{align*}
\]

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

[Bellare-Rogaway – Eurocrypt ‘93]

For efficiency: random oracle model

Setup

- A trapdoor one-way permutation family \( \{f\} \) onto the set \( X \)
- Two hash functions, for the security parameter \( k_1 \),
  \[ g : X \rightarrow \{0, 1\}^n \text{ and } H : \{0, 1\}^* \rightarrow \{0, 1\}^{k_1} \],
  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
First Generic Conversion (Cont'ed)

Encryption

One chooses a random element \( r \in X \)

\[ a = f(r), \quad b = m \oplus G(r), \quad c = H(m, r) \]

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 \( c \? = H(m, r) \)

If the equality holds, one returns \( m \), otherwise one rejects the ciphertext

Security of the Bellare-Rogaway Conversion

Theorem

The Bellare-Rogaway conversion achieves IND – CCA security, under the one-wayness of the trapdoor permutation \( f \):

\[
\text{Adv}_{BR}^{\text{ind-cca}}(t) \leq 2 \times \text{Succ}_f^{\text{ow}}(T) + \frac{4q_D}{2^{|k_1|}},
\]

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 IND – CCA adversary against the Bellare-Rogaway conversion.

Real Attack Game

Game 0

<table>
<thead>
<tr>
<th>Oracles</th>
</tr>
</thead>
<tbody>
<tr>
<td>Setup</td>
</tr>
<tr>
<td>D</td>
</tr>
<tr>
<td>H</td>
</tr>
</tbody>
</table>

Challenge Ciphertext

Random \( r \), random bit \( b \):

\[ a = f(r), \quad b = m_b \oplus G(r), \quad c = H(m, r) \]

\[ \text{Adv}_{\text{Game}_0} = 2 \times \text{Pr}\left[ b' = b \right] - 1 = \varepsilon \]

Game 1: use of the simulation of the random oracles

Random Oracles

For any new query, a new random output: management of lists

\[ \text{Adv}_{\text{Game}_1} = \text{Adv}_{\text{Game}_0} \]
Simulated of the Challenge Ciphertext

- **Game₂**: use of an independent random value $h^+$

**Challenge Ciphertext**

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

This game is indistinguishable from the previous one, unless $(m_b, r)$ is queried to $H$: event **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 **AskMR**, we stop the simulation with a random output:

$$
\text{Adv}_\text{Game₂} \geq \text{Adv}_\text{Game₁} - 2 \times \Pr_{\text{Game₁}}[\text{AskMR}]
$$

Simulated of the Decryption Oracle

- **Game₃**: reject if $(m, r)$ not queried to $H$

**Decryption Oracle**

Look in the $H$-list for $(m, r)$ such that $c = H(m, r)$. If not found: reject, if for one pair, $a = f(r)$ and $b = m \oplus G(r)$, output $m$

This makes a difference if this value $c$, without having been asked to $H$, is correct: for each attempt, the probability is bounded by $1/2^{k_1}$:

$$
\text{Adv}_\text{Game₃} \geq \text{Adv}_\text{Game₂} - 2q_D/2^{k_1}
$$

Simulated of the Challenge Ciphertext

- **Game₄**: 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 $G$ by the adversary or by the decryption oracle. We denote by **AskR** the event that $r$ is asked to $G$ or $H$ by the adversary (which includes **AskMR**). But $r$ cannot be asked to $G$ by the decryption oracle without **AskR**: only possible if $r$ is in the $H$-list, and thus asked by the adversary:

$$
\text{Adv}_\text{Game₄} \geq \text{Adv}_\text{Game₃} - 2 \times \Pr_{\text{Game₃}}[\text{AskR}]
$$

- **Game₅**: use of an independent random value $a^+$ (and $g^+$, $h^+$)

**Challenge Ciphertext**

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

This determines $r$, the unique value such that $r = f(a)$, which allows to detect event **AskR**.

This game is perfectly indistinguishable from the previous one:

$$
\text{Adv}_\text{Game₅} = \text{Adv}_\text{Game₄}
$$
$$
\Pr_{\text{Game₅}}[\text{AskR}] = \Pr_{\text{Game₄}}[\text{AskR}]
$$
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 \( m_b \):

\[
\text{Adv}_{\text{Game}_5} = 0
\]

Conclusion

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

\[
= \text{Adv}_{\text{Game}_4} \geq 2 \times \Pr_{\text{Game}_3}[\text{AskR} \mid \neg \text{AskMR}] - 2q_D/2^{k_1}
\]

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

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

And then,

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

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

Game-based Methodology: the story of OAEP

[Bellare-Rogaway EC '94]

- 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