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**

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

**IND – CPA Security Game**

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

---

**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
EUF − NMA

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

Outline

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

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)

Direct Reduction

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)

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

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

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

- 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

**Output**

- perfectly identical behaviors
- different behaviors, only if event $Ev$ happens
  - $Ev$ is negligible
  - $Ev$ is non-negligible
  - and independent of the output in Game $A$
  - $\Rightarrow$ Simulator B terminates in case of event $Ev$
Two Simulations

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

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

  Simulator B terminates and flips a coin, in case of event $Ev$:

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

---

Event $Ev$

- Either $Ev$ is negligible, or the output is independent of $Ev$
- For being able to terminate simulation B in case of event $Ev$, this event must be efficiently detectable
- For evaluating $\Pr[Ev]$, one re-iterates the above process,
  with an initial game that outputs 1 when event $Ev$ 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 \(c^0\) such that the plaintext \(m^0\) is related to \(m\) in a meaningful way:

- ElGamal ciphertext: \(c_1 = g^r\) and \(c_2 = m \times y^r\)
- Malleability: \(c_1^0 = c_1 = g^r\) and \(c_2^0 = 2 \times c_2 = (2m) \times y^r\)

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**

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)
  - adaptive (CCA − 2)

**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)
  - 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)

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

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

### 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

### Key Generation

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

### Encryption

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

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

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**

- **Game$_0$**: use of the oracles $K, D$
- **Game$_1$**: 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$_1$ is: $\Pr_{\text{Game}_1}[b^0 = b | F] = 1/2$

\[
\frac{\Pr_{\text{Game}_0}[F]}{\Pr_{\text{Game}_1}[F]} = \frac{\Pr_{\text{Game}_0}[b^0 = b | F]}{\Pr_{\text{Game}_1}[b^0 = b | F]} = \frac{1}{2}
\]

$\implies$ Hop-S-Negl: $\text{Adv}_{\text{Game}_1} \geq \text{Adv}_{\text{Game}_0} - \Pr[F]$
Details: Shoup’s Lemma

\[ \text{Adv}_{\text{Game}_1} = 2 \times \Pr_{\text{Game}_1}[b^0 = b] - 1 \]
\[ = 2 \times \Pr_{\text{Game}_1}[b^0 = b \mid \neg F] \Pr_{\text{Game}_1}[-F] + 2 \times \Pr_{\text{Game}_1}[b^0 = b \mid F] \Pr_{\text{Game}_1}[F] - 1 \]
\[ = 2 \times \Pr_{\text{Game}_0}[b^0 = b \mid \neg F] \Pr_{\text{Game}_0}[-F] + \Pr_{\text{Game}_0}[F] - 1 \]
\[ = \text{Adv}_{\text{Game}_0} - \Pr_{\text{Game}_0}[F] (2 \times \Pr_{\text{Game}_0}[b^0 = b \mid F] - 1) \]
\[ \geq \text{Adv}_{\text{Game}_0} - \Pr_{\text{Game}_0}[F] \]

Proof: Computable Adversary

- **Game2**: we use the simulations

  **Key Generation Simulation**
  \[ x_1, x_2, y_1, y_2, z_1, z_2 \overset{R}{\leftarrow} \mathbb{Z}_q, \ g_1, g_2 \overset{R}{\leftarrow} \mathbb{G}: \ sk = (x_1, x_2, y_1, y_2, z_1, z_2) \]
  \[ c = g_1^{x_1} g_2^{x_2}, \ d = g_1^{y_1} g_2^{y_2}, \ \text{and} \ h = g_1^{x_2} g_2^{y_2}: \ pk = (g_1, g_2, H, c, d, h) \]
  \[ z = z_1 + sz_2 \]

  Distribution of the public key: Identical

  **Decryption Simulation**
  If \( v = u_1^{x_1 + \alpha y_1} u_2^{x_2 + \alpha y_2} \) where \( \alpha = H(u_1, u_2, e) \): \( m = e / u_1^{z_1} u_2^{z_2} \)

  Under the assumption of \( \neg F \), perfect simulation
  \[ \Rightarrow \text{Hop-S-Perfect}: \text{Adv}_{\text{Game}_2} = \text{Adv}_{\text{Game}_1} \]

Details: Bad Accept

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:

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

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

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

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

Original Challenge
Random choice: \( b \xleftarrow{\$} \{0, 1\}, r \xleftarrow{\$} \mathbb{Z}_q \) with \( [\alpha = H(u₁, u₂, e)] \)

\[ u₁ = g₁^b, \, u₂ = g₂^r, \, e = m_b \times h^r, \, v = c' d^\alpha \]

New Challenge 1
Given \((U = g₁^b, V = g₂^r)\) from outside, and random choice \( b \xleftarrow{\$} \{0, 1\} \)

\[ u₁ = U, \, u₂ = V, \, e = m_b \times U^{z₁} V^{z₂}, \, v = U^{x₁+yₙ₁} V^{x₂+y₂} \]

With \((U = g₁^b, V = g₂^r): U^{z₁} V^{z₂} = h^r \) and \( U^{x₁+yₙ₁} V^{x₂+y₂} = c' d^\alpha \)

\( \implies \) Hop-S-Perfect: \( \text{Adv}_{\text{Game₄}} = \text{Adv}_{\text{Game₃}} \)

Proof: DDH Assumption

The input from outside changes from \((U = g₁^b, V = g₂^r)\) (a CDH tuple) to \((U = g₁^{*₁}, V = g₂^{*₂})\) (a random tuple):

\[ \text{Pr}_{\text{Game₄}}[b^0 = b] - \text{Pr}_{\text{Game₅}}[b^0 = b] \leq \text{Adv}_{\text{ddh}}(t) \]

\( \implies \) Hop-D-Comp: \( \text{Adv}_{\text{Game₅}} \geq \text{Adv}_{\text{Game₄}} - 2 \times \text{Adv}_{\text{ddh}}(t) \)

(Since \( \text{Adv} = 2 \times \text{Pr}[b^0 = b] - 1 \))

Proof: DDH Assumption

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

Previous Challenge 1
Given \((U = g₁^b, V = g₂^r)\) from outside, and random choice \( b \xleftarrow{\$} \{0, 1\} \)

\[ u₁ = U, \, u₂ = V, \, e = m_b \times U^{z₁} V^{z₂}, \, v = U^{x₁+yₙ₁} V^{x₂+y₂} \]

New Challenge 2
Given \((U = g₁^{*₁}, V = g₂^{*₂})\) from outside, and random choice \( b \xleftarrow{\$} \{0, 1\} \)

\[ u₁ = U, \, u₂ = V, \, e = m_b \times U^{z₁} V^{z₂}, \, v = U^{x₁+yₙ₁} V^{x₂+y₂} \]

The input changes from \((U = g₁^b, V = g₂^r)\) to \((U = g₁^{*₁}, V = g₂^{*₂})\):

\( \implies \) Hop-D-Comp: \( \text{Adv}_{\text{Game₅}} \geq \text{Adv}_{\text{Game₄}} - 2 \times \text{Adv}_{\text{ddh}}(t) \)

Proof: Collision

■ Game₆: we abort (with a random output \( b^0 \)) in case of second pre-image with a decryption query

Event \( F_H \)

\( \mathcal{A} \) submits a ciphertext with the same \( \alpha \) as the challenge ciphertext, but a different initial triple: \((u₁, u₂, e) \neq (u₁^{*}, u₂^{*}, e^{*})\), but \( \alpha = \alpha^{*} \), were \(*\) are for all the elements related to the challenge ciphertext.

Second pre-image of \( \mathcal{H} \):

\[ \text{Pr}[F_H] \leq \text{Succ}^H(t) \]

The advantage in Game₆ is: \( \text{Pr}_{\text{Game₆}}[b^0 = b | F_{H}] = 1/2 \)

\[ \text{Pr}_{\text{Game₅}}[F_{H} | b^0 = b] \]

\( \implies \) Hop-S-Negl: \( \text{Adv}_{\text{Game₆}} \geq \text{Adv}_{\text{Game₅}} - \text{Pr}[F_{H}] \)

\[ \text{Adv}_{\text{Game₆}} \geq \text{Adv}_{\text{Game₅}} - \text{Succ}^H(t) \]
**Details: Bad Accept (Case 3)**

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

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

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

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

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

The determinant of the system is

\[
\Delta = \begin{vmatrix}
1 & s & 0 & 0 \\
0 & 0 & 1 & s \\
r_1^* & sr_2^* & r_1^{\alpha^*} & sr_2^{\alpha^*} \\
r_1 & sr_2 & r_1^{\alpha} & sr_2^{\alpha} \\
\end{vmatrix}
\]

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

The system is under-defined:

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

\[ \Rightarrow v \text{ is unpredictable} \Rightarrow Pr[F_3^0] \leq q_D/q \]

\[ \Rightarrow \text{Adv}_{Game_7} \geq \text{Adv}_{Game_6} - q_D/q \]
Proof: Analysis of the Final Game

In the final Game₇:
- only valid ciphertexts are decrypted
- the challenge ciphertext contains
  \[ e = m_b \times U_z^2 V_z^2 \]
- 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
\[ \implies m_b \text{ is unpredictable} \quad \text{and} \quad \implies b \text{ is unpredictable} \]
\[ \implies \text{Adv}_{\text{Game}_7} = 0 \]

Conclusion

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

First Generic Conversion

[Bellare-Rogaway – Eurocrypt '93]

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

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 \( k_1 \),
  \[ G : X \rightarrow \{0, 1\}^n \text{ and } H : \{0, 1\}^* \rightarrow \{0, 1\}^{k_1}, \]
  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
Security of the Bellare-Rogaway Conversion

Theorem
The Bellare-Rogaway conversion achieves IND−CCA security, under the one-wayness of the trapdoor permutation $f$:

$$\text{Adv}_{\text{BR}}^{\text{ind-cca}}(t) \leq 2 \times \text{Succ}^{\text{OW}}_{f}(T) + \frac{4q_{D}}{2^{k_1}},$$

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

Simulation of the Random Oracles

- **Game$_0$**: use of the perfect oracles

  **Challenge Ciphertext**
  Random $r$, random bit $b$: $a = f(r)$, $b = m_{b} \oplus g(r)$, $c = \mathcal{H}(m, r)$

  $$\text{Adv}_{\text{Game}_0} = 2 \times \text{Pr}_{\text{Game}_0}[b^{0} = b] - 1 = \varepsilon$$

- **Game$_1$**: use of the simulation of the random oracles

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

  $$\text{Adv}_{\text{Game}_1} = \text{Adv}_{\text{Game}_0}$$
**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 AskR the event that $r$ is asked to $\mathcal{G}$ or $\mathcal{H}$ by the adversary (which includes AskMR). But $r$ cannot be asked to $\mathcal{G}$ by the decryption oracle without AskR: only possible if $r$ is in the $\mathcal{H}$-list, and thus asked by the adversary:

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

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

**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 $1/2^{k_1}$:

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

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

**Simulation of the Challenge Ciphertext**

- **Game$_5$:** 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 $a^+ = f(r)$, which allows to detect event AskR.

This game is perfectly indistinguishable from the previous one:

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

\[
\Pr_{\text{Game}_5}[\text{AskR}] = \Pr_{\text{Game}_4}[\text{AskR}]
\]
Conclusion

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

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

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

$$\geq \text{Adv}_{\text{Game}_1} - 2 \times \Pr_{\text{Game}_3}[\text{AskMR}] - \frac{q_D}{2^k_1}$$

$$\geq \text{Adv}_{\text{Game}_0} - 2 \times \Pr_{\text{Game}_3}[\text{AskMR}] - 4 \frac{q_D}{2^k_1}$$

And then,

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

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