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

- $k_e$ to $E$
- $k_d$ to $D$
- $m$ to $E$
- $r$ to $E$
- $c$ to $D$
- $m$ to $D$

ENS/CNRS/INRIA Cascade

David Pointcheval

1/68

0x0

2/68
**OW – CPA Security Game**

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

**IND – CPA Security Game**

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

---

**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**
**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

**Direct Reduction**

Outline

1. Basic Security Notions
2. Game-based Proofs
   - Provable Security
   - Game-based Approach
   - Transition Hops
3. Advanced Security for Encryption
4. Conclusion
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
0 / 1
Adversary

Game 1
Oracles
Simulator 1
Distribution 1
Adversary
Challenger
0 / 1

Game 2
Oracles
Simulator 2
Distribution 2
Adversary
Challenger
0 / 1
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
  - $E_v$ is non-negligible
  and independent of the output in Game_A
  $\rightarrow$ Simulator B terminates in case of event $E_v$
  - [Hop-S-Negl]
  - [Hop-S-Non-Negl]
Two Distributions

- Perfectly identical input distributions
- Different distributions
  - Statistically close
  - Computationally close

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: \( \Pr[\text{Game}_A] - \Pr[\text{Game}_B] \leq \Pr[\text{Ev}] \)

\[
\begin{align*}
|\Pr[\text{Game}_A] - \Pr[\text{Game}_B]| &= \Pr[\text{Game}_A|\text{Ev}] \Pr[\text{Ev}] + \Pr[\text{Game}_B|\neg\text{Ev}] \Pr[\neg\text{Ev}] \\
&- \Pr[\text{Game}_B|\text{Ev}] \Pr[\text{Ev}] - \Pr[\text{Game}_A|\neg\text{Ev}] \Pr[\neg\text{Ev}] \\
&= \left( \Pr[\text{Game}_A|\text{Ev}] - \Pr[\text{Game}_B|\text{Ev}] \right) \times \Pr[\text{Ev}] \\
&+ \left( \Pr[\text{Game}_A|\neg\text{Ev}] - \Pr[\text{Game}_B|\neg\text{Ev}] \right) \times \Pr[\neg\text{Ev}] \\
&\leq |1 \times \Pr[\text{Ev}] + 0 \times \Pr[\neg\text{Ev}]| \leq \Pr[\text{Ev}]
\end{align*}
\]

- \( \text{Ev} \) 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] = \text{Dist(Distrib}_A, \text{Distrib}_B) = \text{negl}()

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

Pr[Game_A] − Pr[Game_B] ≤ Adv_{Distr}(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**

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) \( \implies \) the adversary can obtain the plaintext of any ciphertext of its choice (excepted the challenge)
  - non-adaptive (CCA − 1)
  - adaptive (CCA − 2)
    - only before receiving the challenge
    - unlimited oracle access
IND − CCA Security Game

The adversary can ask any decryption of its choice:
Chosen-Ciphertext Attacks (oracle access)

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

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

Cramer-Shoup Encryption Scheme

Key Generation

- \(G = (\langle g \rangle, \times)\) group of order \(q\)
- \(sk = (x_1, x_2, y_1, y_2, z), \) where \(x_1, x_2, y_1, y_2, z \overset{R}{\leftarrow} \mathbb{Z}_q\)
- \(pk = (g_1, g_2, \mathcal{H}, c, d, h), \) 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

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

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

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

Decryption

- since \( h = g_1^t, \ h^t = u_1^t \), thus \( m = e/u_1^t \)
- 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

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

Real Attack Game

Theorem

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

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

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.

Proof: Invalid ciphertexts

- **Game_0**: use of the oracles \( \mathcal{K}, \mathcal{D} \)
- **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:

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

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

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

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

$$= 2 \times \Pr_{\text{Game}_1} [b' = b \| -F] \Pr_{\text{Game}_1} [-F]$$

$$+ 2 \times \Pr_{\text{Game}_0} [b' = b \| F] \Pr_{\text{Game}_0} [F] - 1$$

$$= 2 \times \Pr_{\text{Game}_0} [b' = b] \Pr_{\text{Game}_0} [-F] + \Pr_{\text{Game}_0} [F] - 1$$

$$= 2 \times \Pr_{\text{Game}_0} [b'] - 2 \times \Pr_{\text{Game}_0} [b' = b \| F] \Pr_{\text{Game}_0} [F]$$

$$+ \Pr_{\text{Game}_0} [F] - 1$$

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

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

Proof: Computable Adversary

Key Generation Simulation

- Game$_2$: we use the simulations

Key Generation Simulation

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

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 \( \alpha = \mathcal{H}(u_1, u_2, e) \): \( m = e / u_1^{z_1} u_2^{z_2} \)

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

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

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

\[ \Rightarrow \Pr[F] \leq qD/q \quad \Rightarrow \text{Adv}_{\text{Game}_1} \geq \text{Adv}_{\text{Game}_0} - qD/q \]

Proof: Simulations

- Game$_2$: we use the simulations

Game$_2$: we use the simulations

Key Generation Simulation

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

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 \( \alpha = \mathcal{H}(u_1, u_2, e) \): \( m = e / u_1^{z_1} u_2^{z_2} \)

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

$$\Rightarrow \text{Hop-S-Perfect:} \ \text{Adv}_{\text{Game}_2} = \text{Adv}_{\text{Game}_1}$$
**Proof: DDH Assumption**

**Game\textsubscript{4}:** we modify the generation of the challenge ciphertext:

**Original Challenge**

Random choice: \( b \stackrel{R}{\leftarrow} \{0, 1\}, r \stackrel{R}{\leftarrow} \mathbb{Z}_q \) 

\[ u_1 = g_1^b, \quad u_2 = g_2^r, \quad e = m_b \times h', \quad v = c' d'^\alpha \]

**New Challenge 1**

Given \((U = g_1^b, V = g_2^r)\) from outside, and random choice \( b \stackrel{R}{\leftarrow} \{0, 1\} \)

\[ u_1 = U, \quad u_2 = V, \quad e = m_b \times U^{z_1} V^{z_2}, \quad v = U^{x_1+\alpha y_1} V^{x_2+\alpha y_2} \]

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

\(\implies\) **Hop-S-Perfect**: \(\text{Adv}_{\text{Game}\textsubscript{4}} = \text{Adv}_{\text{Game}\textsubscript{3}}\)

**Proof: DDH Assumption**

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

\[
\text{Pr}_{\text{Game}\textsubscript{4}} [b' = b] - \text{Pr}_{\text{Game}\textsubscript{5}} [b' = b] \leq \text{Adv}_{ddh}^G (t)
\]

\(\implies\) **Hop-D-Comp**: \(\text{Adv}_{\text{Game}\textsubscript{5}} \geq \text{Adv}_{\text{Game}\textsubscript{4}} - 2 \times \text{Adv}_{ddh}^G (t)\)

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

---

**Proof: DDH Assumption**

**Game\textsubscript{5}:** we modify the generation of the challenge ciphertext:

**Previous Challenge 1**

Given \((U = g_1^b, V = g_2^r)\) from outside, and random choice \( b \stackrel{R}{\leftarrow} \{0, 1\} \)

\[ u_1 = U, \quad u_2 = V, \quad e = m_b \times U^{z_1} V^{z_2}, \quad v = U^{x_1+\alpha y_1} V^{x_2+\alpha y_2} \]

**New Challenge 2**

Given \((U = g_1^b, V = g_2^r)\) from outside, and random choice \( b \stackrel{R}{\leftarrow} \{0, 1\} \)

\[ u_1 = U, \quad u_2 = V, \quad e = m_b \times U^{z_1} V^{z_2}, \quad v = U^{x_1+\alpha y_1} V^{x_2+\alpha y_2} \]

The input changes from \((U = g_1^b, V = g_2^r)\) to \((U = g_1^{b'}, V = g_2^r)\):

\(\implies\) **Hop-D-Comp**: \(\text{Adv}_{\text{Game}\textsubscript{5}} \geq \text{Adv}_{\text{Game}\textsubscript{4}} - 2 \times \text{Adv}_{ddh}^G (t)\)

---

**Proof: Collision**

**Game\textsubscript{6}:** we abort (with a random output \( b' \)) in case of second pre-image with a decryption query

**Event \( F_H \)**

\(A\) submits a ciphertext with the same \( \alpha \) as the challenge ciphertext, but a different initial tuple: \((u_1, u_2, e) \neq (u_1^*, u_2^*, e^*)\), but \( \alpha = \alpha^* \), were \( ^* \) are for all the elements related to the challenge ciphertext.

Second pre-image of \( H \):

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

The advantage in \( \text{Game}\textsubscript{6} \) is: \(\text{Pr}_{\text{Game}\textsubscript{6}} [b' = b | F_H] = 1/2\)

\[ \text{Pr}_{\text{Game}\textsubscript{6}} [b' = b | \neg F_H] = \text{Pr}_{\text{Game}\textsubscript{6}} [b' = b | \neg F_H]\]

\(\implies\) **Hop-S-Negl**: \(\text{Adv}_{\text{Game}\textsubscript{6}} \geq \text{Adv}_{\text{Game}\textsubscript{5}} - \text{Pr}[F_H]\)

\[ \text{Adv}_{\text{Game}\textsubscript{6}} \geq \text{Adv}_{\text{Game}\textsubscript{5}} - \text{Succ}^H (t)\]
Proof: Invalid ciphertexts

- **Game_7**: we abort (with a random output \(b'\)) in case of bad accepted ciphertext, we do as in **Game_1**

**Event F'**

\(\mathcal{A}\) submits a bad accepted ciphertext (note: this is not computationally detectable)

The advantage in **Game_7** is: \(\text{Pr}[F'] = 1/2\)

\[
\text{Pr}_{\text{Game_7}}[b' = b | F'] = \frac{1}{2}
\]

\[
\Rightarrow \text{Hop-S-Negl: } \text{Adv}_{\text{Game_7}} \geq \text{Adv}_{\text{Game_6}} - \text{Pr}[F']
\]

**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 \(g_2 = g_1^\delta\):

\[
\log c = x_1 + sx_2
\]

\[
\log d = y_1 + sy_2
\]

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

\[
\log \nu = r_1 (x_1 + \alpha y_1) + sr_2 (x_2 + \alpha y_2)
\]

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 \(\nu\), there are \((x_1, x_2, y_1, y_2)\) that satisfy the system

\[
\Rightarrow \nu \text{ is unpredictable } \Rightarrow \text{Pr}[F'_3] \leq q_D/q \\
\Rightarrow \text{Adv}_{\text{Game_7}} \geq \text{Adv}_{\text{Game_6}} - q_D/q
\]
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
  
  \[ h = g_z^1 g_z^2 \]

Again, the system is under-defined:

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

Conclusion

\[
\begin{align*}
\text{Adv}_{\text{Game}_7} & = 0 \\
\text{Adv}_{\text{Game}_7} & \geq \text{Adv}_{\text{Game}_6} - q_D/q \\
\text{Adv}_{\text{Game}_6} & \geq \text{Adv}_{\text{Game}_5} - \text{Succ}^H(t) \\
\text{Adv}_{\text{Game}_5} & \geq \text{Adv}_{\text{Game}_4} - 2 \times \text{Adv}^{\text{ddh}}_G(t) \\
\text{Adv}_{\text{Game}_4} & = \text{Adv}_{\text{Game}_3} \\
\text{Adv}_{\text{Game}_3} & \geq \text{Adv}_{\text{Game}_2} - q_D/q \\
\text{Adv}_{\text{Game}_2} & = \text{Adv}_{\text{Game}_1} \\
\text{Adv}_{\text{Game}_1} & \geq \text{Adv}_{\text{Game}_0} - q_D/q \\
\text{Adv}_{\text{Game}_0} & = \text{Adv}^{\text{ind-cca}}_{\text{CS}}(A)
\end{align*}
\]

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

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

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

Security of the Bellare-Rogaway Conversion

Theorem

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

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

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 \( \text{IND} - \text{CCA} \) 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 \( b \):

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

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

Game 1: use of the simulation of the random oracles

Random Oracles

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

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

- **Game$_2$:** use of an independent random value $h^+$

**Challenge Ciphertext**

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

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:

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

Simulation of the Decryption Oracle

- **Game$_3$:** reject if $(m, r)$ not queried to $H$

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

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

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 $G$ 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:

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

- **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 $r = f(a)$, which allows to detect event AskR.

This game is perfectly indistinguishable from the previous one:

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

$$\text{Adv}_{\text{Game}_5} = \text{Adv}_{\text{Game}_4}$$
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 $a^+$:

$$\Pr_{\text{Game}_5} [\text{AskR}] \leq \text{Succ}^{ow}_{f^*}(t + (q_G + q_H) \cdot T_f)$$

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

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

Conclusion

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

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

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

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

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

$$\geq \text{Adv}_{\text{Game}_0} - 2 \times \Pr_{\text{Game}_5} [\text{AskR}] - 4q_D / 2^{k_1}$$

And then,

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

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