The Game-based Methodology for Computational Security Proofs

David Pointcheval

Ecole normale supérieure, CNRS & INRIA

Computational and Symbolic Proofs of Security
Atagawa Heights – Japan
April 6th, 2009

Outline

1 Cryptography
   • Introduction
   • Provable Security

2 Game-based Methodology
   • Game-based Approach
   • Transition Hops

3 Assumptions

4 Identity-Based Encryption
   • Definition
   • Description of BF
   • Security Proof

5 Conclusion

Cryptography
Game-based Proofs
Assumptions
BF IB-Encryption
Conclusion

Introduction

Public-Key Cryptography

Asymmetric cryptography

Encryption

Signature

Encryption guarantees privacy
Signature guarantees authentication, and even non-repudiation by the sender
**Introduction**

**Strong Security Notions**

**Signature**

**Existential Unforgeability under Chosen-Message Attacks**
An adversary, allowed to ask for signature on any message of its choice, cannot generate a new valid message-signature pair.

**Encryption**

**Semantic Security against Chosen-Ciphertext Attacks**
An adversary that chooses 2 messages, and receives the encryption of one of them, is not able to guess which message has been encrypted, even if it is able to ask for decryption of any ciphertext of its choice (except the challenge ciphertext).

**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)

**Game-based Methodology**

**Illustration: 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 showed the gap for IND-CCA2, under the OWP

Granted his new game-based methodology

FOPS proved the security for IND-CCA2, under the PD-OWP

Using the game-based methodology

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

[Bellare-Rogaway EC ‘94]

[Shoup - Crypto ‘01]

[Fujisaki-Okamoto-Pointcheval-Stern – Crypto ‘01]
Outline

1. Cryptography
   - Introduction
   - Provable Security

2. Game-based Methodology
   - Game-based Approach
   - Transition Hops

3. Assumptions

4. Identity-Based Encryption
   - Definition
   - Description of BF
   - Security Proof

5. Conclusion

Game-based Approach

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

Game 2

Oracles

Simulator 2

Distribution 2
**Game-based Approach**

**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

**Transition Hops**
Two Simulators
- perfectly identical behaviors
- different behaviors, only if event $Ev$ happens
  - $Ev$ is negligible
  - $Ev$ is non-negligible and independent of the output in $Game_A$
  → Simulator B terminates in case of event $Ev$

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

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

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 \textit{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

\[ \Pr[\text{Game}_A] - \Pr[\text{Game}_B] \leq \text{Adv}(D^{\text{oracles}}) \]
Two Distributions

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

- For identical/statistically close distributions, for any oracle:
  \[ \text{Pr}[\text{Game}_A] - \text{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:
  \[ \text{Pr}[\text{Game}_A] - \text{Pr}[\text{Game}_B] \leq \text{Adv}_{\text{Distrib}}(t) \]

where \( t \) is the computational time of the distinguisher.

Outline

1. Cryptography
   - Introduction
   - Provable Security
2. Game-based Methodology
   - Game-based Approach
   - Transition Hops
3. Assumptions
4. Identity-Based Encryption
   - Definition
   - Description of BF
   - Security Proof
5. Conclusion

Bilinear Maps

Gap Groups

Definition (Pairing Setting)
- Let \( G_1 \) and \( G_2 \) be two cyclic groups of prime order \( p \)
- Let \( g_1 \) and \( g_2 \) be generators of \( G_1 \) and \( G_2 \) respectively
- Let \( e : G_1 \times G_2 \rightarrow G^T \), be a bilinear map

Definition (Admissible Bilinear Map)
- Let \( (p, G_1, g_1, G_2, g_2, G^T, e) \) be a pairing setting, with \( e : G_1 \times G_2 \rightarrow G^T \) a non-degenerated bilinear map
  - Bilinear: for any \( g \in G_1, h \in G_2 \) and \( u, v \in \mathbb{Z} \),
    \[ e(g^u, h^v) = e(g, h)^{uv} \]
  - Non-degenerated: \( e(g_1, g_2) \neq 1 \)

Bilinear Diffie-Hellman Problems

We focus on the symmetric case: \( G_1 = G_2 = G \)

Diffie-Hellman Problems
- \( \text{CDH in } G : \text{Given } g, g^a, g^b \in G, \text{ compute } g^{ab} \)
- \( \text{DDH in } G : \text{Given } g, g^a, g^b, g^c \in G, \text{ decide whether } c = ab \text{ or not} \)

CDH can be hard to solve, but DDH is easy in gap-groups

Bilinear Diffie-Hellman Problems
- \( \text{CBDH in } G : \text{Given } g, g^a, g^b, g^c \in G, \text{ compute } e(g, g)^{abc} \)
- \( \text{DBDH in } G : \text{Given } g, g^a, g^b, g^c \in G \text{ and } h \in G^T, \text{ decide whether } h \stackrel{?}{=} e(g, g)^{abc} \)
Identity-Based Cryptography

**Public-Key Cryptography**

Each user $ID$ owns
- a public key $pk$
- a certificate that guarantees the link between $ID$ and $pk$
- a private key $sk$, related to $pk$

One has to access a dictionary in order to get $pk$, the public key of $ID$, together with the certificate, in order to encrypt a message to $ID$.

**Identity-Based Cryptography**

Each user $ID$ owns
- a private key $sk$, related to $ID$
- the public key $pk$ is indeed $ID$ itself

Identity-Based Encryption

**Setup**
The authority generates a master secret key $msk$, and publishes the public parameters, $PK$.

**Extraction**
Given an identity $ID$, the authority computes the private key $sk$ granted the master secret key $msk$.

**Encryption**
Any one can encrypt a message $m$ to a user $ID$ using only $m$, $ID$ and the public parameters $PK$.

**Decryption**
Given a ciphertext, user $ID$ can recover the plaintext, with $sk$.

Security Model: $IND-ID-CCA$

**Definition (IND – ID – CCA Security)**

- $A$ receives the global parameters
- $A$ asks any extraction-query, and any decryption-query
- $A$ outputs a target identity $ID^*$ and two messages $(m_0, m_1)$

The challenger flips a bit $b$, and encrypts $m_b$ for $ID^*$ into $c^*$

- $A$ asks any extraction-query, and any decryption-query
- $A$ outputs its guess $b'$ for $b$

Restriction: $ID^*$ never asked to the extraction oracle, and $(ID^*, c^*)$ never asked to the decryption oracle.

CPA: no decryption-oracle access

\[
Adv^{ind-id-cca} = 2 \times \Pr[b' = b] - 1
\]
Description of BF

Identity-Based Encryption

[Boneh-Franklin – Crypto ’01]

Setup

- The authority sets up a gap-group framework:
  - a group $G$ of prime order $p$, with a generator $g$,
  - equipped with an admissible bilinear map $e : G \times G \rightarrow G_T$

- It selects a master secret key $msk = s \in \mathbb{Z}_p$
- It publishes the public parameters: $PK = (p, G, e, g, P = g^s)$

Extraction

Given an identity $ID$, the authority computes

the private key $sk = H(ID)^s$

Note that $sk$ is a BLS signature of $ID$: $e(sk, g) = e(H(ID), P)$

BF IBE (Cont’d)

Encryption

In order to encrypt a message $m$ to a user $ID$

- one chooses a random $r \in \mathbb{Z}_p$
- computes $A = g^r$ and $K = e(P, H(ID)^r)$
- sends $(A, B = K \times m)$

Decryption

Upon reception of $(A, B)$, user $ID$

- computes $K = e(A, sk)$
- gets $m = B / K$

BF IBE Security Analysis

**Theorem**

The BF IBE is $\text{IND} - \text{ID} - \text{CPA}$ secure
under the $\text{DBDH}$ problem, in the random oracle model

By masking $m$ with $H(K)$: $B = m \oplus H(K)$,
the BF IBE is $\text{IND} - \text{ID} - \text{CPA}$ secure
under the $\text{CBDH}$ problem, in the random oracle model

**Theorem**

The BLS signature achieves $\text{EUF} - \text{CMA}$ security, under the $\text{CDH}$
assumption in $G$, in the Random Oracle Model
Simulations

- **Game_0**: use of the oracles Setup, Ext, and \( \mathcal{H} \)
- **Game_1**: use of the simulation of the Random Oracle

**Simulation of \( \mathcal{H} \)**

\( \mathcal{H}(\text{ID}): \mu \overset{R}{\leftarrow} \mathbb{Z}_p \), output \( M = g^\mu \)

\[ \Rightarrow \text{Hop-D-Perfect: } \Pr[\text{Game}_1] = \Pr[\text{Game}_0] \]

- **Game_2**: use of the simulation of the Extraction Oracle

**Simulation of Ext**

\( \text{Ext}(\text{ID}): \text{find } \mu \text{ such that } M = \mathcal{H}(\text{ID}) = g^\mu \), output \( sk = P^\mu \)

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

\( \mathcal{H}(\text{ID}): \mu \overset{R}{\leftarrow} \mathbb{Z}_p \), output \( M = g^\mu \)

**Game_3**: random index \( t \overset{R}{\leftarrow} \{1, \ldots, q_H\} \)

\( \Pr[\text{Game}_3] = \frac{1}{2} + \left( \Pr[\text{Game}_2] - \frac{1}{2} \right) \times \Pr[\neg \text{Ev}] \quad \Pr[\text{Ev}] = 1 - 1/q_H \)

\[ \Pr[\text{Game}_3] = \frac{1}{2} + \left( \Pr[\text{Game}_2] - \frac{1}{2} \right) \times \frac{1}{q_H} \]

\( \mathcal{H}(\text{ID}): \mu \overset{R}{\leftarrow} \mathbb{Z}_p \), output \( M = g^\mu \)

**Game_4**: True DBDH instance \((g, g^\alpha, g^\beta, g^\gamma)\) with \( h = e(g, g)^{\alpha\beta\gamma} \)

Use of the simulation of the Setup Oracle

**Simulation of Setup**

\( \text{Setup}(): \text{set } P \leftarrow g^\alpha \)

Modification of the simulation of the Random Oracle

**Simulation of \( \mathcal{H} \)**

If this is the \( t \)-th query, \( \mathcal{H}(\text{ID}): M \leftarrow g^\beta \), output \( M \)

Difference for the \( t \)-th simulation of the random oracle: we cannot extract the secret key. Since this is the challenge \( \text{ID} \), it cannot be queried to the extraction oracle:

\[ \Rightarrow \text{Hop-D-Perfect: } \Pr[\text{Game}_4] = \Pr[\text{Game}_3] \]

**Game_5**: True DBDH instance \((g, g^\alpha, g^\beta, g^\gamma)\) with \( h = e(g, g)^{\alpha\beta\gamma} \)

We have set \( P \leftarrow g^\alpha \), and for the \( t \)-th query to \( \mathcal{H}: M = g^\beta \)

**Ciphertext**

Set \( A \leftarrow g^\gamma \) and \( K \leftarrow h \) to generate the encryption of \( m_b \) under \( \text{ID} \)

\[ \Rightarrow \text{Hop-D-Perfect: } \Pr[\text{Game}_5] = \Pr[\text{Game}_4] \]

**Game_6**: Random DBDH instance \((g, g^\alpha, g^\beta, g^\gamma)\) with \( h \overset{R}{\leftarrow} \mathcal{G}^T \)

\[ \Rightarrow \text{Hop-D-Comp}: \quad |\Pr[\text{Game}_6] - \Pr[\text{Game}_5]| \leq \text{Adv}_{\text{dbdh}}(t + q_H T_e) \]
In this last Game₆, it is clear that \( \Pr[\text{Game}_6] = \frac{1}{2} \)

\[
|\Pr[\text{Game}_6] - \Pr[\text{Game}_5]| \leq \text{Adv}^{\text{dbdh}}(t + q_{H^τ_e})
\]

\[
\Pr[\text{Game}_5] = \Pr[\text{Game}_4] = \Pr[\text{Game}_3] = \frac{1}{2} + (\Pr[\text{Game}_2] - \frac{1}{2}) \times \frac{1}{q_H}
\]

\[
\Pr[\text{Game}_2] = \Pr[\text{Game}_1] = \Pr[\text{Game}_0] = \frac{1}{2} + \text{Adv}^{\text{ind}\text{-id}\text{-cpa}}(\mathcal{A})
\]

\[
\text{Adv}^{\text{ind}\text{-id}\text{-cpa}}(\mathcal{A}) \leq q_H \times \text{Adv}^{\text{dbdh}}(t + q_{H^τ_e})
\]

The game-based methodology uses a sequence of games.
The transition hops are simple and easy to check.

It leads to easy-to-read and easy-to-verify security proofs:

- Some mistakes have been found granted this methodology.

[Analysis of OAEP]

- Some security analyses became possible to handle.

[Analysis of EKE]

This approach can be automized.

[CryptoVerif]