Nesting Hybrids

Georg Fuchsbauer (IST Austria)
Momchil Konstantinov (Oxford)
Krzysztof Pietrzak (IST Austria)
Vanishree Rao (UCLA)

Visions of Cryptography
Weizmann Institute, December 11th 2013
Adaptive vs. Selective Security

**adaptive attack by A on \( \{\Pi_n\}_{n \in \mathbb{N}} \)**

- A queries \( \Pi_n(.) \)
- A chooses challenge \( x^* \in \{0,1\}^n \)
- A must break \( \Pi_n(.) \) on input \( x^* \)
selective attack by $A$ on $\{\Pi_n\}_{n \in \mathbb{N}}$

- $A$ chooses challenge $x^* \in \{0, 1\}^n$
- $A$ queries $\Pi_n(.)$
- $A$ must break $\Pi_n(.)$ on input $x^*$
Lemma (Security Leveraging)

If A breaks adaptive security with advantage \( \epsilon \) ⇒ can use A to break selective security with advantage \( \epsilon / 2^n \).

selective attack using adaptive A

- guess random challenge \( x' \in \{0, 1\}^n \)
- A queries \( \Pi_n(.) \)
- A chooses challenge \( x^* \)
- if \( x' \neq x^* \) give up
- A must break \( \Pi_n(.) \) on input \( x^* \)
proving adaptive security via leveraging

1. adaptive $\Pi_n \rightarrow$ selective $\Pi_n$ (losing factor $2^n$).
2. selective $\Pi_n \rightarrow \Phi$ (hybrid argument loses poly$(n)$).
proving adaptive security via leveraging

1. adaptive $\Pi_n \rightarrow$ selective $\Pi_n$ (losing factor $2^n$).
2. selective $\Pi_n \rightarrow \Phi$ (hybrid argument looses $\text{poly}(n)$).

nesting hybrids

1. adaptive $\Pi_n \rightarrow$ adaptive* $\Pi_n$ (losing small factor $\alpha$)
2. adaptive* $\Pi_n \rightarrow$ adaptive $\Pi_{n/2}$ (hybrid losing factor $\beta$)
3. iterate 1 and 2 $\log(n)$ times:
   adaptive $\Pi_n \rightarrow$ adaptive $\Pi_1$ losing $(\alpha \beta)^{\log(n)}$
4. adaptive $\Pi_1 \rightarrow \Phi$ lossless.
## Applications

<table>
<thead>
<tr>
<th></th>
<th>old</th>
<th>new</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>GGM Constrained PRF</strong>&lt;sup&gt;1&lt;/sup&gt;</td>
<td></td>
<td></td>
</tr>
<tr>
<td>loss in reduction to PRG</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$n = \text{input length}$, $q = # \text{queries}$</td>
<td>$2^n$</td>
<td>$q^{\log n}$</td>
</tr>
<tr>
<td><strong>Generalized Selective Decryption</strong>&lt;sup&gt;2&lt;/sup&gt;</td>
<td></td>
<td></td>
</tr>
<tr>
<td>loss in reduction to ENC</td>
<td></td>
<td></td>
</tr>
<tr>
<td>caveat: on trees</td>
<td>$2^n$</td>
<td>$2^{\log^2 n}$</td>
</tr>
<tr>
<td>$n = # \text{keys}$</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<sup>1</sup> *Functional Signatures and Pseudorandom Functions.* Elette Boyle, Shafi Goldwasser, Ioana Ivan [eprint.iacr.org/2013/401](http://eprint.iacr.org/2013/401)

<sup>2</sup> *Constrained Pseudorandom Functions and Their Applications.* Dan Boneh and Brent Waters *Asiacrypt 2013*

*Delegatable Pseudorandom Functions and Applications.* A.Kiayias, S.Papadopoulos, N.Triandopoulos, T.Zacharias. *CCS 2013*

*Tackling Adaptive Corruptions in Multicast Encryption Protocols.* Saurabh Panjwani *TCC 2007*
GGM PRF $F_K(x) = K_x$

- PRG $G : \{0, 1\}^n \rightarrow \{0, 1\}^{2n}$
- $K = K_\emptyset \leftarrow \{0, 1\}^n$
- $K_x\|0\|K_x\|1 = G(K_x)$
GGM Hybrid Argument

Adv($H_0$, $H_{qn}$) = $\epsilon$

$q = \#\text{queries}, \ n=\text{input length}$.

$\Rightarrow$ Adv($H_i$, $H_{i+1}$) $\geq \epsilon/qn$

$\Rightarrow$ Adv($G(U_\lambda), U_{2\lambda}$) $\geq \epsilon/qn$
GGM Hybrid Argument

- \( \text{Adv}(H_0, H_{qn}) = \epsilon \) for \( q = \#\text{queries} \), \( n = \text{input length} \).
- \( \Rightarrow \text{Adv}(H_i, H_{i+1}) \geq \epsilon/qn \)
- \( \Rightarrow \text{Adv}(G(U_\lambda), U_{2\lambda}) \geq \epsilon/qn \)
GGM Hybrid Argument

\[ \text{Adv}(H_0, H_{qn}) = \epsilon \]
\[ q = \# \text{queries} , \; n = \text{input length.} \]
\[ \Rightarrow \text{Adv}(H_i, H_{i+1}) \geq \epsilon / qn \]
\[ \Rightarrow \text{Adv}(G(U_\lambda), U_{2\lambda}) \geq \epsilon / qn \]
GGM Hybrid Argument

Hybrid $H_0$ (the real game)

$\text{Adv}(H_0, H_{qn}) = \epsilon \quad q = \#\text{queries}, \ n = \text{input length}$.

$\Rightarrow \text{Adv}(H_i, H_{i+1}) \geq \frac{\epsilon}{qn}$

$\Rightarrow \text{Adv}(G(U^{\lambda}), U^{2\lambda}) \geq \frac{\epsilon}{qn}$
GGM Hybrid Argument

Hybrid $H_1$

- $\text{Adv}(H_0, H_{qn}) = \epsilon$
- $q = \#\text{queries}$, $n = \text{input length}$.
- $\Rightarrow \text{Adv}(H_i, H_{i+1}) \geq \epsilon/qn$
- $\Rightarrow \text{Adv}(G(U_\lambda), U_{2\lambda}) \geq \epsilon/qn$
Adv($H_0, H_{qn}) = \epsilon$

$q = \#\text{queries}, \ n=\text{input length}.$

$\Rightarrow$ Adv($H_i, H_{i+1}$) $\geq \epsilon/qn$

$\Rightarrow$ Adv($G(U_\lambda), U_{2\lambda}$) $\geq \epsilon/qn$
Adv($H_0, H_{qn}$) = $\epsilon$  
$q = \#\text{queries}$, $n=$input length.

$\Rightarrow$ Adv($H_i, H_{i+1}$) $\geq \epsilon/qn$

$\Rightarrow$ Adv($G(U_\lambda), U_{2\lambda}$) $\geq \epsilon/qn$
Adv($H_0$, $H_{qn}$) = $\epsilon$

$q = \# \text{queries}$, $n = \text{input length}$.

$\Rightarrow$ Adv($H_i$, $H_{i+1}$) $\geq$ $\epsilon/qn$

$\Rightarrow$ Adv($G(U_\lambda)$, $U_{2\lambda}$) $\geq$ $\epsilon/qn$
GGM Hybrid Argument

Hybrid $H_5$ (the random game)

- $\text{Adv}(H_0, H_{qn}) = \epsilon$
- $q = \#\text{queries}$, $n=\text{input length}$.
- $\Rightarrow \text{Adv}(H_i, H_{i+1}) \geq \epsilon/qn$
- $\Rightarrow \text{Adv}(G(U_\lambda), U_{2\lambda}) \geq \epsilon/qn$
Constrained/Delegatable/Functional GGM PRF

Functional Signatures and Pseudorandom Functions. Elette Boyle, Shafi Goldwasser, Ioana Ivan
eprint.iacr.org/2013/401

Constrained Pseudorandom Functions and Their Applications. Dan Boneh and Brent Waters Asiacrypt 2013

CCS 2013
Security game for constrained PRFs

choose $x^*$ where no prefix of $x^*$ was queried.
distinguish $K_x^*$ from random.
Constrained/Delegatable/Functional GGM PRF

Security game for constrained PRFs

choose $x^*$ where no prefix of $x^*$ was queried.

distinguish $K \cdot x^*$ from random.
$K_x \parallel y$ trivially distinguishable from random given $K_x$. 
$K_{x\|y}$ trivially distinguishable from random given $K_x$.

Security game for constrained PRFs
- choose $x^*$ where no prefix of $x^*$ was queried.
- distinguish $K_{x^*}$ from random.
Proving Selective Security

Adv($H_0, H_6$) = $\epsilon$  
$\Rightarrow$ Adv($H_i, H_{i+1}$) $\geq$ $\epsilon/6$  
$\Rightarrow$ Adv($G(U_\lambda), U_{2\lambda}$) $\geq$ $\epsilon/6$
Proving Selective Security

Choose challenge

\[
\begin{align*}
\text{Adv}(H_0, H_6) &= \epsilon \\
\Rightarrow \quad \text{Adv}(H_i, H_{i+1}) &\geq \epsilon/6 \\
\Rightarrow \quad \text{Adv}(G(U^\lambda), U^{2\lambda}) &\geq \epsilon/6
\end{align*}
\]
Proving Selective Security

Hybrid $H_0$ (real game)

- $\text{Adv}(H_0, H_6) = \epsilon$
- $\Rightarrow \text{Adv}(H_i, H_{i+1}) \geq \epsilon/6$
- $\Rightarrow \text{Adv}(G(U_\lambda), U_{2\lambda}) \geq \epsilon/6$
Proving Selective Security

Hybrid $H_1$

$Adv(H_0, H_6) = \epsilon \Rightarrow Adv(H_i, H_{i+1}) \geq \epsilon/6 \Rightarrow Adv(G(U_\lambda), U_{2\lambda}) \geq \epsilon/6$
Proving Selective Security

Hybrid $H_2$

$\text{Adv}(H_0, H_6) = \epsilon \Rightarrow \text{Adv}(H_i, H_{i+1}) \geq \epsilon / 6 \Rightarrow \text{Adv}(G(U_\lambda), U_{2\lambda}) \geq \epsilon / 6$
Proving Selective Security

Hybrid $H_3$

\[
\begin{align*}
\text{Adv}(H_0, H_6) &= \epsilon \\
\Rightarrow \quad \text{Adv}(H_i, H_{i+1}) &\geq \epsilon/6 \\
\Rightarrow \quad \text{Adv}(G(U_{\lambda}), U_{2\lambda}) &\geq \epsilon/6
\end{align*}
\]
Proving Selective Security

Hybrid $H_4$

$$\text{Adv}(H_0, H_6) = \epsilon$$

$$\Rightarrow \text{Adv}(H_i, H_{i+1}) \geq \epsilon/6$$

$$\Rightarrow \text{Adv}(G(U_\lambda), U_{2\lambda}) \geq \epsilon/6$$
Adv(H₀, H₆) = \epsilon \\
⇒ Adv(Hᵢ, Hᵢ₊₁) ≥ \epsilon/6 \\
⇒ Adv(G(U_λ), U₂λ) ≥ \epsilon/6
Proving Selective Security

Hybrid $H_6$ (random game)

- $\text{Adv}(H_0, H_6) = \epsilon$
- $\Rightarrow \text{Adv}(H_i, H_{i+1}) \geq \epsilon/6$
- $\Rightarrow \text{Adv}(G(U_\lambda), U_{2\lambda}) \geq \epsilon/6$

Krzysztof Pietrzak  Nesting Hybrids
Proof

- Leveraging: Guess Challenge

\[ \epsilon \rightarrow \frac{\epsilon}{2^n} \]
Proving Adaptive Security using Leveraging

Proof

- Leveraging: Guess Challenge
- Hybrid Argument

\[ \epsilon \xrightarrow{2^n} \frac{\epsilon}{2^n} \xrightarrow{2^n} \epsilon \]
Proof

1. Guess first query that agrees with $x^*$ on 4-prefix.

\[ \epsilon \xrightarrow{\epsilon} q \]
Proof

1. Guess first query that agrees with $x^*$ on 4-prefix.
2. Hybrid argument.

$$
\epsilon \rightarrow \frac{\epsilon}{q} \rightarrow \frac{\epsilon}{3q}
$$
Proving Adaptive Security by Nesting

Proof

1. Guess first query that agrees with $x^*$ on 6-prefix.
2. Hybrid argument.

\[ \epsilon \rightarrow \frac{\epsilon}{q} \rightarrow \frac{\epsilon}{3q} \rightarrow \ldots \rightarrow \]
Proof

1. Guess first query that agrees with $x^*$ on 6-prefix.
2. Hybrid argument.

\[ \epsilon \rightarrow ^q \epsilon \rightarrow ^{3q} \epsilon \rightarrow \ldots \rightarrow \]
Proof

1. Guess first query that agrees with $x^*$ on 5-prefix.
2. Hybrid argument.

\[
\epsilon \rightarrow \frac{\epsilon}{q} \rightarrow \frac{\epsilon}{3q} \rightarrow \ldots \rightarrow
\]
Proof

1. Guess first query that agrees with $x^*$ on 5-prefix.
2. Hybrid argument.

$$\epsilon \rightarrow \frac{\epsilon}{q} \rightarrow \frac{\epsilon}{3q} \rightarrow \ldots \rightarrow \frac{\epsilon}{(3q)^{\log n}}$$