Simple and efficient PRFs with tighter Security via All-Prefix Universal Hash Functions

Tibor Jager
Paderborn University

Rafael Kurek
Paderborn University

Jiaxin Pan
Karlsruhe Institute of Technology
This talk

• New notion for Hash Functions
  • All-Prefix Universality
  • Examples
This talk

• New notion for Hash Functions
  • All-Prefix Universality
  • Examples

• New framework for tightly secure Pseudorandom Functions
  • very simple, small keys, efficient
  • covering Matrix-DDH (MDDH) and learning with errors (LWE)
This talk

• New notion for Hash Functions
  • All-Prefix Universality
  • Examples

• New framework for tightly secure Pseudorandom Functions
  • very simple, small keys, efficient
  • covering Matrix-DDH (MDDH) and learning with errors (LWE)

• LWE-based PRF
  • Currently most efficient construction with weaker security assumption and super-poly modulus
Pseudorandom Functions (PRFs)

We call $F$ a pseudorandom Function if both worlds are computationally indistinguishable.
Cryptographic Reduction

Reduction B
(simulating PRF game)

Adversary A
(breaking PRF security)

"hard problem"

\(c\)

solution to problem

\(\phi(b)\)

\(\pi(c,x)\)

\(x\)

\(b\)
Tightness in reductions

We say that reduction B loses a factor $L$, if

$$\frac{t(B)}{e(B)} = L \frac{t(A)}{e(A)}$$

- $t$: running time
- $e$: advantage

We say the reduction is “tight“, if $L$ is small (i.e. constant or logarithmic).
Tightness in reductions

We say that reduction B loses a factor $L$, if

$$\frac{t(B)}{e(B)} = L \frac{t(A)}{e(A)}$$

- $t$: running time
- $e$: advantage

We say the reduction is “tight“, if $L$ is small (i.e. constant or logarithmic).

Loss might depend on input length!
PRFs with loss depending on input length

- **GGM PRF** [FOCS84]
- **Matrix-DDH-based PRFs** [Escala et al. CRYPTO13]
  - Naor-Reingold PRF [FOCS97]
  - Lewko-Waters PRF [CCS09]
- **LWE-based PRFs**
  - BPR PRF [Banerjee et al. EUROCRYPT12]
Naïve approach

1. Hash input $x$ with cryptographic Hash Function

\[
H : \{0,1\}^* \rightarrow \{0,1\}^n \\
x \mapsto h(x)
\]

2. Evaluate PRF on hash

\[
F_k(h(x))
\]
Naïve approach

1. Hash input $x$ with cryptographic Hash Function

   $$H: \{0,1\}^* \rightarrow \{0,1\}^n$$

   $$x \mapsto h(x)$$

2. Evaluate PRF on hash

   $$F_k(h(x))$$

   $n=2\lambda$, where $\lambda$ security parameter

   $\Rightarrow$ Security Loss $O(\lambda)$ and $|sk| = O(\lambda)$
On-the-fly adaption

[Döttling and Schröder CRYPTO15]
On-the-fly adaption

[Döttling and Schröder CRYPTO15]
On-the-fly adaption

[Döttling and Schröder CRYPTO15]
Döttling and Schröder [CRYPTO15]

• Works especially well for PRFs with loss in input length
• Tight security loss in framework
• Smaller keys
• $\lambda \cdot \omega(\log \lambda)$ invocations of underlying PRF (in the generic framework)
Döttling and Schröder [CRYPTO15]

- Works especially well for PRFs with loss in input length
- Tight security loss in framework
- Smaller keys
- $\lambda \cdot \omega(\log \lambda)$ invocations of underlying PRF (in the generic framework)

Can we do it with a single invocation?
Augmented cascade PRF
[Boneh at al. ACM CCS 2010]

Let $F: S \times K \times \{0,1\} \rightarrow K$ be a PRF.

Key space
Augmented cascade PRF
[Boneh et al. ACM CCS 2010]

Let \( F : S \times K \times \{0,1\} \rightarrow K \) be a PRF.

Augmented cascade PRF \( \hat{F}^m \)

\[
\begin{align*}
F & \quad x_1 \\
F & \quad x_2 \\
F & \quad x_3 \\
\cdots & \\
F & \quad x_m \\
\end{align*}
\]

Key space

\( \hat{F}^m(s_1, \ldots, s_m, k, x) \)
Augmented cascade PRF

[Boneh at al. ACM CCS 2010]

Let $F: S \times K \times \{0,1\} \rightarrow K$ be a PRF.

Key space

Augmented cascade PRF $\hat{F}^m$

Loss and $|sk|$ depend on input length!

=> shorter input => tighter proof and shorter keys
Universal Hash functions

Denote $H = \{ h \mid h: \{0,1\}^n \to \{0,1\}^m \}$.

$H$ is a family of universal hash functions, if

$$Pr_{h \leftarrow H}[h(x) = h(x')] \leq \frac{1}{2^m}$$

$\forall x \neq x'$
All-Prefix Universal Hash Functions

Denote $H = \{h \mid h: \{0,1\}^n \rightarrow \{0,1\}^m\}$.

$H$ is a family of all-prefix universal hash functions, if

$$Pr_{h \leftarrow H}[h(x)_i = h(x')_i] \leq \frac{1}{2^i}$$

$\forall x \neq x' \; \forall i \in [m]$
All-Prefix almost-Universal Hash Functions

Denote $H = \{h \mid h: \{0,1\}^n \to \{0,1\}^m\}$.

$H$ is a family of all-prefix almost-universal hash functions, if

$$P_{r_h \leftarrow H}[h(x)_i = h(x')_i] \leq \frac{2}{2^i}$$

$\forall x \neq x', \forall i \in [m]$
The Augmented Cascade with Encoded Input

1. Hash input $x$ with All-Prefix Universal Hash Function with output length $m = \omega(\log \lambda)$
The Augmented Cascade with Encoded Input

1. Hash input $x$ with All-Prefix Universal Hash Function with output length $m = \omega(\log \lambda)$
2. Evaluate Augmented Cascade PRF on $h(x)$
Proof sketch

• B breaks AC-PRF with length $j$, where $j$ depends on adversary $A$
• Simulates security game for $A$, breaking AC-PRF with encoded input

$h(x)_1 \ldots h(x)_j \xrightarrow{F} s_{j+1}$

$h(x)_{j+1} \xrightarrow{F} \ldots$

$h(x)_m \xrightarrow[\hat{F}^m(s_1, \ldots, s_m, k, h(x))]{F} j=O(\log k), m=\omega(\log \lambda)$
Proof sketch

- B breaks AC-PRF with length \( j \), where \( j \) depends on adversary \( A \)
- Simulates security game for \( A \), breaking AC-PRF with encoded input

\[
h(x)_1 \ldots h(x)_j \quad h(x)_{j+1} \quad \ldots \quad h(x)_m
\]

\[
R \quad F \quad \ldots \quad F
\]

\[
s_{j+1} \quad s_m
\]

\[
\tilde{F}^m(s_1, \ldots, s_m, k, h(x))
\]

\[
j=O(\log k), \; m=\omega(\log \lambda)
\]
Proof sketch

**APUHF:**

- $R(h(\cdot)_j)$ uniformly random $\Rightarrow$ A gains no information about $h$
  $\Rightarrow$ information-theoretically hard to find collision
- no collision on $h(\cdot)_j$ $\Rightarrow$ $R(h(\cdot)_j)$ uniformly random for all queries

$$h(x)_1 \ldots h(x)_j \downarrow \quad h(x)_{j+1} \downarrow \quad \ldots \quad h(x)_m \downarrow \quad \hat{F}^m(s_1, \ldots, s_m, k, h(x))$$

$j=O(\log k), m=\omega(\log \lambda)$
One additional property required!

**Perfect one-time security**

\[
\Pr_{k\leftarrow K} [\hat{F}(s, k, x) = k'] = \frac{1}{|K|}
\]

for all \((s, k', x) \in S \times K \times \{0, 1\}\)
## Comparison MDDH-PRFs

<table>
<thead>
<tr>
<th></th>
<th>Key Size</th>
<th>Loss</th>
<th>Invocations</th>
<th>Domain</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>MDDH PRFs</strong></td>
<td>$n$</td>
<td>$n$</td>
<td>1</td>
<td>$n &gt;&gt; m$</td>
</tr>
<tr>
<td><strong>DötSch15 PRF</strong></td>
<td>$m = \omega(\log \lambda)$</td>
<td>$\mathcal{O}(\log \lambda)$</td>
<td>1</td>
<td>domain $\mathbb{Z}_q$</td>
</tr>
<tr>
<td><strong>Our PRF</strong></td>
<td>$m = \omega(\log \lambda)$</td>
<td>$\mathcal{O}(\log \lambda)$</td>
<td>1</td>
<td></td>
</tr>
</tbody>
</table>
## Comparison LWE

<table>
<thead>
<tr>
<th></th>
<th>Key Size</th>
<th>Loss</th>
<th>Invocations</th>
<th>Modulus</th>
</tr>
</thead>
<tbody>
<tr>
<td>BPR PRFs</td>
<td>$n$</td>
<td>$Q \cdot N \cdot n$</td>
<td>1</td>
<td>exp in $\lambda$</td>
</tr>
<tr>
<td>DötSch15 PRF</td>
<td>$m = \omega(\log \lambda)$</td>
<td>$Q \cdot N \cdot \mathcal{O}(\log \lambda)$</td>
<td>$\lambda \cdot \omega(\log \lambda)$</td>
<td>super-poly in $\lambda$</td>
</tr>
<tr>
<td>Our PRF</td>
<td>$m = \omega(\log \lambda)$</td>
<td>$Q \cdot N \cdot \mathcal{O}(\log \lambda)$</td>
<td>1</td>
<td>super-poly in $\lambda$</td>
</tr>
</tbody>
</table>
Example: All-Prefix Universal HF

• Pairwise-independent hash functions mapping to bits

\[ h_{a,b} : GF(2^n) \rightarrow GF(2^n) \]

\[ x \mapsto ax + b \]

\[ H = \{ h_{a,b} : a, b \in \{0, 1\}^n \} \]
Example: All-Prefix almost-Universal HF

• Dietzfelbinger et al. [DHKP, J ALG97]

\[ h_a : \{0, 1\}^m \rightarrow \{0, 1\}^n \]

\[ x \mapsto (ax \mod 2^n) \text{div}^{n-m} \]

\[ H_{n,m} = \{ h_a : a \in [2^n - 1] \text{ and } a \text{ is odd} \} \]
Comparison to Truncation Collision Resistance

Both
• Similar technical properties
• Chosen prefix length depends on adversary

APUHF
• Security based on secret key
• Known Construction

Tru-CR HF
• Security not based on secret key
• Additional complexity assumption for standard HF
Conclusion

• New notion for Hash Functions
  • All-Prefix Universality
  • Examples

• New framework for tightly secure Pseudorandom Functions
  • very simple, small keys, efficient
  • covering Matrix-DDH (MDDH) and learning with errors (LWE)

• LWE-based PRF
  • Currently most efficient construction with weak security assumption

Thank you for your attention!

This talk: iacr.org/2018/826  Tuesday: iacr.org/2017/061