On Quantum Secure Compressing Pseudorandom Functions

Ritam Bhaumi [1] Benoît Cogliati [2] Jordan Ethan [3] Ashwin Jha [3]

[1]EPFL, Switzerland

[2]Thales DIS France SAS, Meudon, France

[3]CISPA Helmholtz Center for Information Security, Saarbrücken, Germany

December 5, 2023

Table of Contents

Why study Compressing PRF's?

- Block ciphers are PRF's up to the BB (classic $q \ll 2^{n/2}$, quantum $q \ll 2^{n/3}$)

Why study Compressing PRF's?

- Block ciphers are PRF's up to the BB (classic $q \ll 2^{n/2}$, quantum $q \ll 2^{n/3}$)
- $2n$-bit Universal hash $+$ $2n$ to n-bit PRF $\rightarrow$ MAC, AEAD-SIV (classically).

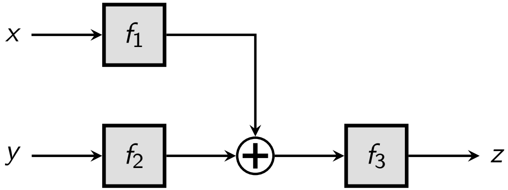
Why study Compressing PRF's?

- Block ciphers are PRF's up to the BB (classic $q \ll 2^{n/2}$, quantum $q \ll 2^{n/3}$)
- $2n$-bit Universal hash $+$ $2n$ to n-bit PRF $\rightarrow$ MAC, AEAD-SIV (classically).
- Are there Quantum secure PRF's?

LRWQ

- Hosoyamada and Iwata [HI19] in 2019 show the construction is QPRF as long as $q \ll 2^{n/4}$ and f_1, f_2, f_3 are assumed to be random.

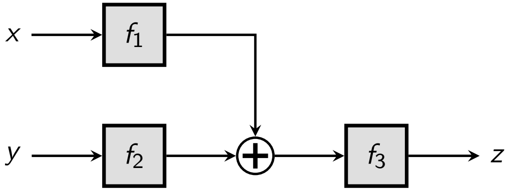
$$F(x, y) := f_3(f_1(x) \oplus f_2(y))$$



LRWQ

- Hosoyamada and Iwata [HI19] in 2019 show the construction is QPRF as long as $q \ll 2^{n/4}$ and f_1, f_2, f_3 are assumed to be random.
- A variant of Zhandry's compressed oracle [Zha18] is used to analyze the adversary's transcript.

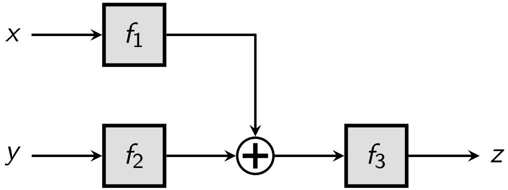
$$F(x, y) := f_3(f_1(x) \oplus f_2(y))$$



LRWQ

- Hosoyamada and Iwata [HI19] in 2019 show the construction is QPRF as long as $q \ll 2^{n/4}$ and f_1, f_2, f_3 are assumed to be random.
- A variant of Zhandry's compressed oracle [Zha18] is used to analyze the adversary's transcript.
- LRWQ uses 3 PRF calls:

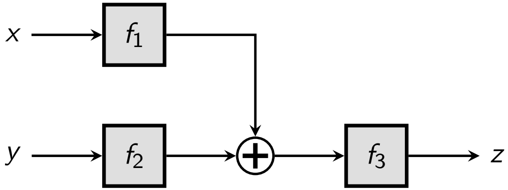
$$F(x, y) := f_3(f_1(x) \oplus f_2(y))$$



LRWQ

- Hosoyamada and Iwata [HI19] in 2019 show the construction is QPRF as long as $q \ll 2^{n/4}$ and f_1, f_2, f_3 are assumed to be random.
- A variant of Zhandry's compressed oracle [Zha18] is used to analyze the adversary's transcript.
- LRWQ uses 3 PRF calls:
  - Is there a QPRF secure construction with 2 PRF calls?

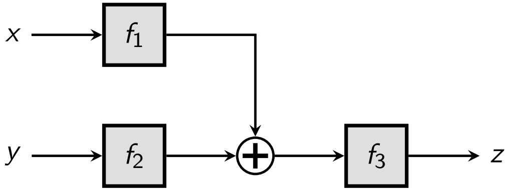
$$F(x, y) := f_3(f_1(x) \oplus f_2(y))$$



LRWQ

- Hosoyamada and Iwata [HI19] in 2019 show the construction is QPRF as long as $q \ll 2^{n/4}$ and f_1, f_2, f_3 are assumed to be random.
- A variant of Zhandry's compressed oracle [Zha18] is used to analyze the adversary's transcript.
- LRWQ uses 3 PRF calls:
  - Is there a QPRF secure construction with 2 PRF calls?
  - Are there other QPRF secure constructions with 3 PRF calls?

$$F(x, y) := f_3(f_1(x) \oplus f_2(y))$$



Our Contributions

- All constructions with 2 PRF calls are broken!

Our Contributions

- All constructions with 2 PRF calls are broken!
- We identify seven interesting QPRF candidates involving 3 PRF calls.

Our Contributions

- All constructions with 2 PRF calls are broken!
- We identify seven interesting QPRF candidates involving 3 PRF calls.
- We prove three of these constructions are secure in the quantum setting as long as $q \ll 2^{n/4}$ and the internal components are assumed to be random.

PRF Distinguishing Game

- **Real World:** a $2n$-bit-to-n-bit function F that internally calls several independent n-bit-to-n-bit uniform random functions $f_1, f_2, f_3, \ldots$.

PRF Distinguishing Game

- **Real World:** a $2n$-bit-to-n-bit function F that internally calls several independent n-bit-to-n-bit uniform random functions $f_1, f_2, f_3, \ldots$.
- **Ideal World:** a $2n$-bit-to-n-bit uniform random function F^*.

PRF Distinguishing Game

- **Real World:** a $2n$-bit-to-n-bit function F that internally calls several independent n-bit-to-n-bit uniform random functions $f_1, f_2, f_3, \ldots$.
- **Ideal World:** a $2n$-bit-to-n-bit uniform random function F^*.
- **Information-Theoretic Setting:** all uniform random functions are assumed to be unkeyed and have perfect randomness.

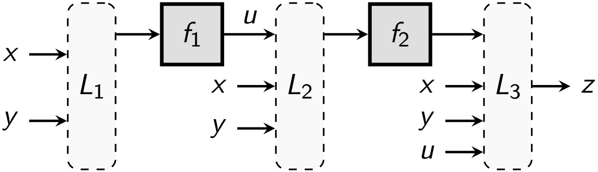
PRF Distinguishing Game

- **Real World:** a $2n$-bit-to-n-bit function F that internally calls several independent n-bit-to-n-bit uniform random functions $f_1, f_2, f_3, \ldots$.
- **Ideal World:** a $2n$-bit-to-n-bit uniform random function F^*.
- **Information-Theoretic Setting:** all uniform random functions are assumed to be unkeyed and have perfect randomness.
- The adversary makes q queries to to a secret oracle (either F or F^*) and has to guess (with good probability) which world it is.

Table of Contents

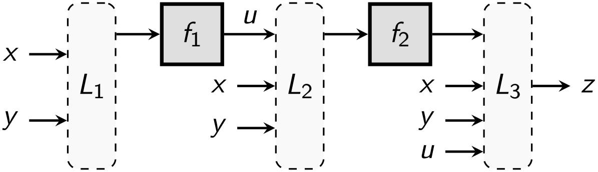
Classification of 2-call Candidate PRFs

- Generic construction with three linear layers L_1, L_2, and L_3:



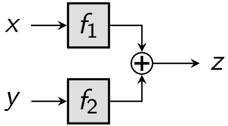
Classification of 2-call Candidate PRFs

- Generic construction with three linear layers L_1, L_2, and L_3:



- In this work, we do a full classification of all possible 2-call candidates, and show that none of them is quantum-secure.

Example of Classical Distinguisher



$$F(x, y) := f_1(x) \oplus f_2(y)$$

- Pick $x \neq x'$, $y \neq y'$ such that $F(x, y) \oplus F(x', y) \oplus F(x', y') \oplus F(x, y') = 0$.
- For a random function F this property holds with negligible probability.

Simon's Algorithm

- $f : \{0,1\}^n \to \{0,1\}^n$ is a *periodic* function if for all $x \in \{0,1\}^n$, $f(x \oplus s) = f(x)$ for some constant s.

Simon's Algorithm

- $f : \{0,1\}^n \to \{0,1\}^n$ is a *periodic* function if for all $x \in \{0,1\}^n$, $f(x \oplus s) = f(x)$ for some constant s.
- **Simon's Algorithm:** recovers hidden s in $O(n)$ queries to f.

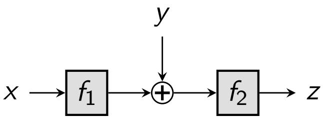
Simon's Algorithm

- $f : \{0,1\}^n \to \{0,1\}^n$ is a *periodic* function if for all $x \in \{0,1\}^n$, $f(x \oplus s) = f(x)$ for some constant s.
- **Simon's Algorithm:** recovers hidden s in $O(n)$ queries to f.
- Works also if f is almost periodic (expect some small subset of inputs) with high probability.

Simon's Algorithm

- $f : \{0,1\}^n \rightarrow \{0,1\}^n$ is a *periodic* function if for all $x \in \{0,1\}^n$, $f(x \oplus s) = f(x)$ for some constant s.
- **Simon's Algorithm:** recovers hidden s in $O(n)$ queries to f.
- Works also if f is almost periodic (expect some small subset of inputs) with high probability.
- Since a random function is far from periodic with high probability $\rightarrow$ Simon's Algorithm can be used to distinguish f from a random function.

Example of Quantum Distinguisher



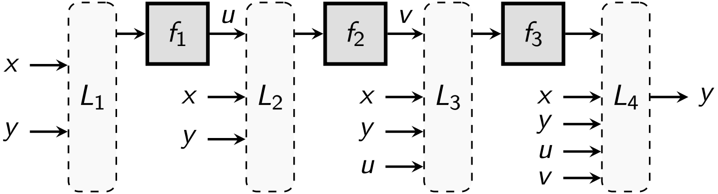
$$F(x, y) := f_2(f_1(x) \oplus y)$$

- Pick $x \neq x'$
- Define $g(y) := F(x, y) \oplus F(x', y)$
- g is periodic with period $s(x, x') = f_1(x) \oplus f_1(x')$.
- Use Simon's Algorithm to construct an efficient quantum distinguisher.

Table of Contents

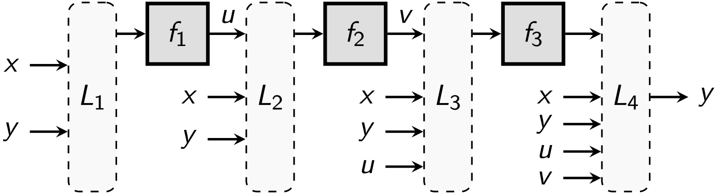
Classification of 3-call candidate PRFs

- Generic construction with four linear layers L_1, L_2, L_3, and L_4:



Classification of 3-call candidate PRFs

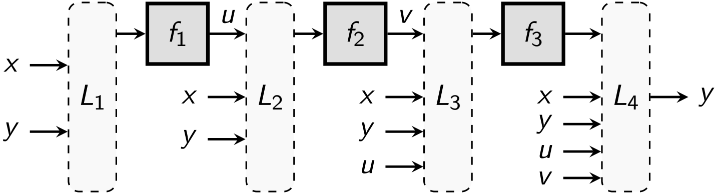
- Generic construction with four linear layers L_1, L_2, L_3, and L_4:



- We do a full classification as earlier.

Classification of 3-call candidate PRFs

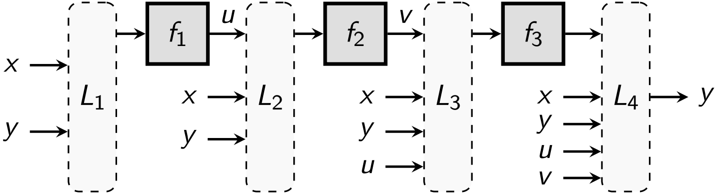
- Generic construction with four linear layers L_1, L_2, L_3, and L_4:



- We do a full classification as earlier.
- This time we are luckier, and can identify seven potentially quantum-secure candidates.

Classification of 3-call candidate PRFs

- Generic construction with four linear layers L_1, L_2, L_3, and L_4:



- We do a full classification as earlier.
- This time we are luckier, and can identify seven potentially quantum-secure candidates.
- We prove the quantum security of three of them.

Interesting Candidates

Candidate	Definition	Mem	XORs	Inv	Par
LRQ	$f_3(f_1(x) \oplus y) \oplus f_2(y)$	$2n$	2	✓	✓
CSUMQ	$f_2(f_1(x) \oplus y) \oplus f_3(f_1(x) \oplus x \oplus y)$	$2n$	3	×	✓
LMQ	$f_2(f_1(x \oplus y) \oplus x) \oplus f_3(f_1(x \oplus y) \oplus y)$	$2n$	4	×	✓
LRWQ†	$f_3(f_1(x) \oplus f_2(y))$	$2n$	1	✓	✓
EDMQ	$f_3(f_2(f_1(x) \oplus y) \oplus x)$	n	2	×	×
TNT†	$f_3(f_2(f_1(x) \oplus y) \oplus y)$	n	2	✓	×
EDMDQ	$f_3(f_1(x) \oplus f_2(f_1(x) \oplus y))$	n	2	×	×

- Note that LRQ, LRWQ and TNT can be seen as tweakable permutation (with y as a tweak) as long as f_1, f_2, f_3 are permutations.

†: studied in earlier works

Table of Contents

Historical Context

- Classical proof techniques rely heavily on transcripts.

Historical Context

- Classical proof techniques rely heavily on transcripts.
- Hard to generalize to quantum setting (No-cloning theorem).

Historical Context

- Classical proof techniques rely heavily on transcripts.
- Hard to generalize to quantum setting (No-cloning theorem).
- In 2018 Zhandry [Zha18] proposed the compressed oracle technique.

Historical Context

- Classical proof techniques rely heavily on transcripts.
- Hard to generalize to quantum setting (No-cloning theorem).
- In 2018 Zhandry [Zha18] proposed the compressed oracle technique.
- In 2019 Hosoyamada and Iwata [HI19] started using the compressed oracle in a good-bad database setting.

Historical Context

- Classical proof techniques rely heavily on transcripts.
- Hard to generalize to quantum setting (No-cloning theorem).
- In 2018 Zhandry [Zha18] proposed the compressed oracle technique.
- In 2019 Hosoyamada and Iwata [HI19] started using the compressed oracle in a good-bad database setting.
- Their work is done in the computational basis $\rightarrow$ long and tedious calculations.

Historical Context

- Classical proof techniques rely heavily on transcripts.
- Hard to generalize to quantum setting (No-cloning theorem).
- In 2018 Zhandry [Zha18] proposed the compressed oracle technique.
- In 2019 Hosoyamada and Iwata [HI19] started using the compressed oracle in a good-bad database setting.
- Their work is done in the computational basis $\rightarrow$ long and tedious calculations.
- In 2020 Chung et al. [Chu+20] introduced a framework for using the compressed oracle in classical-like arguments over the Fourier basis.

Historical Context

- Classical proof techniques rely heavily on transcripts.
- Hard to generalize to quantum setting (No-cloning theorem).
- In 2018 Zhandry [Zha18] proposed the compressed oracle technique.
- In 2019 Hosoyamada and Iwata [HI19] started using the compressed oracle in a good-bad database setting.
- Their work is done in the computational basis $\rightarrow$ long and tedious calculations.
- In 2020 Chung et al. [Chu+20] introduced a framework for using the compressed oracle in classical-like arguments over the Fourier basis.
- Our work extends Chung et al. framework to produce compact indistinguishability proofs that uses mostly classic counting reasoning.

Very High-Level Overview

- Query-response pairs are ‘stored’ in databases.

Very High-Level Overview

- Query-response pairs are ‘stored’ in databases.
- **Bad Databases:** defined separately for each game as a predicate over the stored query-response pairs.

Very High-Level Overview

- Query-response pairs are ‘stored’ in databases.
- **Bad Databases:** defined separately for each game as a predicate over the stored query-response pairs.
- **Transition Capacity:** A measure of the probability of a database going bad after a single query.

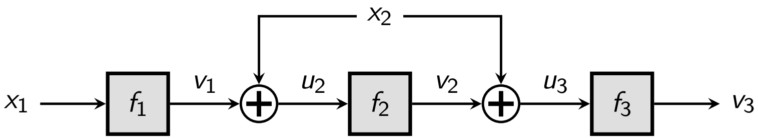
Very High-Level Overview

- Query-response pairs are 'stored' in databases.
- **Bad Databases:** defined separately for each game as a predicate over the stored query-response pairs.
- **Transition Capacity:** A measure of the probability of a database going bad after a single query.
- **Main Idea:** We show that the 'good' databases evolve identically in either game, and bound the distinguishing advantage by the cumulative transition capacity.

High-Level Proof of TNT

We examine the post-quantum security of the $2n$-bit-to-n-bit PRF TNT defined as

$$g_{\mathsf{re}}^{\mathsf{TNT}}(x_1, x_2) := f_3(f_2(f_1(x_1) \oplus x_2) \oplus x_2)$$



here f_1, f_2, f_3 are n-bit random functions, which we instantiate with compressed oracles.

Modified game

- Initial goal: bound the distinguishing advantage between $g_{\mathrm{re}}^{\mathrm{TNT}}$ (the real world) and a $2n$ to n bit random function g_{id} (ideal world).

Modified game

- Initial goal: bound the distinguishing advantage between $g_{\mathsf{re}}^{\mathsf{TNT}}$ (the real world) and a $2n$ to n bit random function g_{id} (ideal world).
- Chung et al. Framework can only handle a single database.

Modified game

- Initial goal: bound the distinguishing advantage between $g_{\mathrm{re}}^{\mathrm{TNT}}$ (the real world) and a $2n$ to n bit random function g_{id} (ideal world).
- Chung et al. Framework can only handle a single database.
- Define $f : \{0,1\}^{3n+2} \to \{0,1\}^n$ function such that:

$$f_1(x) = f(00\|x\|0^{2n}) \qquad f_2(x) = f(01\|x\|0^{2n})$$
$$f_3(x) = f(10\|x\|0^{2n}) \qquad g_{\mathrm{id}}(x, x') = f(11\|x\|x'\|0^n).$$

Modified game

- Initial goal: bound the distinguishing advantage between g_{re}^{TNT} (the real world) and a $2n$ to n bit random function g_{id} (ideal world).
- Chung et al. Framework can only handle a single database.
- Define $f : \{0,1\}^{3n+2} \to \{0,1\}^n$ function such that:

$$f_1(x) = f(00\|x\|0^{2n}) \qquad f_2(x) = f(01\|x\|0^{2n})$$

$$f_3(x) = f(10\|x\|0^{2n}) \qquad g_{id}(x,x') = f(11\|x\|x'\|0^n).$$

- Now f_1, f_2, f_3, g_{id} are independent.

Modified game

- Initial goal: bound the distinguishing advantage between $g_{\mathrm{re}}^{\mathrm{TNT}}$ (the real world) and a $2n$ to n bit random function g_{id} (ideal world).
- Chung et al. Framework can only handle a single database.
- Define $f : \{0,1\}^{3n+2} \to \{0,1\}^n$ function such that:

$$f_1(x) = f(00\|x\|0^{2n}) \qquad f_2(x) = f(01\|x\|0^{2n})$$

$$f_3(x) = f(10\|x\|0^{2n}) \qquad g_{\mathrm{id}}(x,x') = f(11\|x\|x'\|0^n).$$

- Now $f_1, f_2, f_3, g_{\mathrm{id}}$ are independent.
- Replace g_{id} with g_{id}^* defined as

$$g_{\mathrm{id}}^*(x_1,x_2) = f(11\|x_1\|x_2\|f_2(f_1(x_1)\oplus x_2)\oplus x_2)$$

.

Modified game

- Initial goal: bound the distinguishing advantage between $g_{\mathrm{re}}^{\mathrm{TNT}}$ (the real world) and a $2n$ to n bit random function g_{id} (ideal world).
- Chung et al. Framework can only handle a single database.
- Define $f : \{0,1\}^{3n+2} \to \{0,1\}^n$ function such that:

$$f_1(x) = f(00\|x\|0^{2n}) \qquad f_2(x) = f(01\|x\|0^{2n})$$

$$f_3(x) = f(10\|x\|0^{2n}) \qquad g_{\mathrm{id}}(x, x') = f(11\|x\|x'\|0^n).$$

- Now $f_1, f_2, f_3, g_{\mathrm{id}}$ are independent.
- Replace g_{id} with g_{id}^* defined as

$$g_{\mathrm{id}}^*(x_1, x_2) = f(11\|x_1\|x_2\|f_2(f_1(x_1) \oplus x_2) \oplus x_2)$$

.

- $g_{\mathrm{id}}^*(x_1, x_2)$ is random in $x_1||x_2$.

Singe-Database Setup

- Now d_f acts as a single database $\rightarrow$ can track f_1, f_2, f_3, and g^*_{id}.

Singe-Database Setup

- Now d_f acts as a single database $\rightarrow$ can track f_1, f_2, f_3, and g^*_{id}.
- In the real world d_{re} tracks f_1, f_2, f_3 (resp. in the ideal world d_{id} tracks f_1, f_2, g^*_{id}).

Singe-Database Setup

- Now d_f acts as a single database $\rightarrow$ can track f_1, f_2, f_3, and g^*_{id}.
- In the real world d_{re} tracks f_1, f_2, f_3 (resp. in the ideal world d_{id} tracks f_1, f_2, g^*_{id}).
- $[x]_1 = 00\|x\|0^{2n}$, $[x]_2 = 01\|x\|0^{2n}$, $[x]_3 = 10\|x\|0^{2n}$.

Singe-Database Setup

- Now d_f acts as a single database $\rightarrow$ can track f_1, f_2, f_3, and g^*_{id}.
- In the real world d_{re} tracks f_1, f_2, f_3 (resp. in the ideal world d_{id} tracks f_1, f_2, g^*_{id}).
- $[x]_1 = 00\|x\|0^{2n}$, $[x]_2 = 01\|x\|0^{2n}$, $[x]_3 = 10\|x\|0^{2n}$.
- $\tilde{\mathcal{X}}_{re} = \{\{[x]_1, [x]_2, [x]_3\}$ and $\tilde{\mathcal{X}}_{id} = \{[x]_1, [x]_2, 11\|x\|x'\|y\}$ are the sets of inputs for d_{re} and d_{id} respectively.

Singe-Database Setup

- Now d_f acts as a single database $\rightarrow$ can track f_1, f_2, f_3, and g^*_{id}.
- In the real world d_{re} tracks f_1, f_2, f_3 (resp. in the ideal world d_{id} tracks f_1, f_2, g^*_{id}).
- $[x]_1 = 00\|x\|0^{2n}$, $[x]_2 = 01\|x\|0^{2n}$, $[x]_3 = 10\|x\|0^{2n}$.
- $\tilde{\mathcal{X}}_{re} = \{\{[x]_1, [x]_2, [x]_3\}$ and $\tilde{\mathcal{X}}_{id} = \{[x]_1, [x]_2, 11\|x\|x'\|y\}$ are the sets of inputs for d_{re} and d_{id} respectively.
- $\mathcal{D}_{re} = \mathcal{D}|_{\tilde{\mathcal{X}}_{re}}$, $\mathcal{D}_{id} = \mathcal{D}|_{\tilde{\mathcal{X}}_{id}}$.

Bad Databases

Let $\mathcal{B}_{re}$ be the set of databases d_{re} satisfying the following: we can find $x_1, v_1, x_1', v_1', x_2, v_2, x_2', v_2', v_3$ such that

- $([x_1]_1, v_1), ([x_1']_1, v_1'), ([v_1 \oplus x_2]_2, v_2), ([v_1' \oplus x_2']_2, v_2') \in d_{re}$
- $v_2 \oplus x_2 = v_2' \oplus x_2'$
- $([v_2 \oplus x_2]_3, v_3) \in d_{re}$

Let $\mathcal{B}_{id}$ be the set of databases d_{id} satisfying the following: we can find $x_1, v_1, x_1', v_1', x_2, v_2, x_2', v_2', v_3$ such that

- $([x_1]_1, v_1), ([x_1']_1, v_1'), ([v_1 \oplus x_2]_2, v_2), ([v_1' \oplus x_2']_2, v_2') \in d_{id}$
- $v_2 \oplus x_2 = v_2' \oplus x_2'$
- One of $(11\|x_1\|x_2\|(v_2 \oplus x_2), v_3)$ and $(11\|x_1'\|x_2'\|(v_2 \oplus x_2), v_3) \in d_{id}$

Bijection between Good Databases

- $\mathcal{G}_{re} = \mathcal{D}_{re} \setminus \mathcal{B}_{re}, \mathcal{G}_{id} = \mathcal{D}_{id} \setminus \mathcal{B}_{id}$.

Bijection between Good Databases

- $\mathcal{G}_{re} = \mathcal{D}_{re} \setminus \mathcal{B}_{re}, \mathcal{G}_{id} = \mathcal{D}_{id} \setminus \mathcal{B}_{id}$.
- In $\mathcal{G}_{re}, \mathcal{G}_{id}$ each $u_3 = v_2 \oplus x_2$ is associated with a unique (x_1, x_2).

Bijection between Good Databases

- $\mathcal{G}_{re} = \mathcal{D}_{re} \setminus \mathcal{B}_{re}, \mathcal{G}_{id} = \mathcal{D}_{id} \setminus \mathcal{B}_{id}$.
- In $\mathcal{G}_{re}, \mathcal{G}_{id}$ each $u_3 = v_2 \oplus x_2$ is associated with a unique (x_1, x_2).
- We can define the bijection $h : \mathcal{G}_{re} \rightarrow \mathcal{G}_{id}$ as follows:

Bijection between Good Databases

- $\mathcal{G}_{re} = \mathcal{D}_{re} \setminus \mathcal{B}_{re}, \mathcal{G}_{id} = \mathcal{D}_{id} \setminus \mathcal{B}_{id}$.
- In $\mathcal{G}_{re}, \mathcal{G}_{id}$ each $u_3 = v_2 \oplus x_2$ is associated with a unique (x_1, x_2).
- We can define the bijection $h : \mathcal{G}_{re} \to \mathcal{G}_{id}$ as follows:
  - for each x_1, $d_{id}([x_1]_1) = d_{re}([x_1]_1)$

Bijection between Good Databases

- $\mathcal{G}_{re} = \mathcal{D}_{re} \setminus \mathcal{B}_{re}, \mathcal{G}_{id} = \mathcal{D}_{id} \setminus \mathcal{B}_{id}$.
- In $\mathcal{G}_{re}, \mathcal{G}_{id}$ each $u_3 = v_2 \oplus x_2$ is associated with a unique (x_1, x_2).
- We can define the bijection $h : \mathcal{G}_{re} \to \mathcal{G}_{id}$ as follows:
    - for each x_1, $d_{id}([x_1]_1) = d_{re}([x_1]_1)$
    - for each x_2, $d_{id}([x_2]_2) = d_{re}([x_2]_2)$

Bijection between Good Databases

- $\mathcal{G}_{re} = \mathcal{D}_{re} \setminus \mathcal{B}_{re}, \mathcal{G}_{id} = \mathcal{D}_{id} \setminus \mathcal{B}_{id}$.
- In $\mathcal{G}_{re}, \mathcal{G}_{id}$ each $u_3 = v_2 \oplus x_2$ is associated with a unique (x_1, x_2).
- We can define the bijection $h : \mathcal{G}_{re} \to \mathcal{G}_{id}$ as follows:
  - for each x_1, $d_{id}([x_1]_1) = d_{re}([x_1]_1)$
  - for each x_2, $d_{id}([x_2]_2) = d_{re}([x_2]_2)$
  - for each x_1, x_2 and the associated u_3, $d_{id}(11\|x_1\|x_2\|u_3) = d_{re}([u_3]_3)$

Finalizing The Proof

- The main point is to show that:

$$\left(\perp \overset{3q}{\rightsquigarrow} \mathcal{B}_{re}\right) + \left(\perp \overset{3q}{\rightsquigarrow} \mathcal{B}_{id}\right) \leq 4\sqrt{\frac{10q^4}{2^n}},$$

this is done by analyzing the effect of each action $\{f_1, f_2, f_3\}$ on the transition capacity at each query i.

Finalizing The Proof

- The main point is to show that:

$$\left(\perp \overset{3q}{\rightsquigarrow} \mathcal{B}_{re}\right) + \left(\perp \overset{3q}{\rightsquigarrow} \mathcal{B}_{id}\right) \leq 4\sqrt{\frac{10q^4}{2^n}},$$

  this is done by analyzing the effect of each action $\{f_1, f_2, f_3\}$ on the transition capacity at each query i.
- From our framework we can deduce:

$$\mathbf{Adv}_{\mathrm{TNT}}^{\mathrm{qprf}} \leq 4\sqrt{\frac{10q^4}{2^n}}.$$

Future Work

- Our proof framework has a potential of developing into a go-to technique for doing quantum proofs for symmetric constructions.

Future Work

- Our proof framework has a potential of developing into a go-to technique for doing quantum proofs for symmetric constructions.
- Limitation: compressed oracle can only replace PRFs, not SPRPs (where inverse calls are required as part of the mode's functionality)

Future Work

- Our proof framework has a potential of developing into a go-to technique for doing quantum proofs for symmetric constructions.
- Limitation: compressed oracle can only replace PRFs, not SPRPs (where inverse calls are required as part of the mode's functionality)
- A concurrent publication has proposed a compressed permutation oracle to resolve this issue.

Future Work

- Our proof framework has a potential of developing into a go-to technique for doing quantum proofs for symmetric constructions.
- Limitation: compressed oracle can only replace PRFs, not SPRPs (where inverse calls are required as part of the mode's functionality)
- A concurrent publication has proposed a compressed permutation oracle to resolve this issue.
- We are presently working on integrating this permutation oracle into our proof framework.

Future Work

- Our proof framework has a potential of developing into a go-to technique for doing quantum proofs for symmetric constructions.
- Limitation: compressed oracle can only replace PRFs, not SPRPs (where inverse calls are required as part of the mode's functionality)
- A concurrent publication has proposed a compressed permutation oracle to resolve this issue.
- We are presently working on integrating this permutation oracle into our proof framework.
- Another direction: getting tighter security proofs $\rightarrow$ seems difficult.

Conclusions

- We showed constructions with 2 PRF calls are not secure (either classical or quantum).

Conclusions

- We showed constructions with 2 PRF calls are not secure (either classical or quantum).
- We identified seven interesting QPRF candidates that involve 3 PRF calls.

Conclusions

- We showed constructions with 2 PRF calls are not secure (either classical or quantum).
- We identified seven interesting QPRF candidates that involve 3 PRF calls.
- We proved the quantum security of LRQ, LRWQ and TNT as long as $q \ll 2^{n/4}$ using our new framework.

Thank You!

References

[Chu+20] Kai-Min Chung et al. *On the Compressed-Oracle Technique, and Post-Quantum Security of Proofs of Sequential Work*. Cryptology ePrint Archive, Paper 2020/1305. https://eprint.iacr.org/2020/1305. 2020. URL: https://eprint.iacr.org/2020/1305.

[HI19] Akinori Hosoyamada and Tetsu Iwata. *4-Round Luby-Rackoff Construction is a qPRP: Tight Quantum Security Bound*. Cryptology ePrint Archive, Report 2019/243. https://eprint.iacr.org/2019/243. 2019.

[Zha18] Mark Zhandry. *How to Record Quantum Queries, and Applications to Quantum Indifferentiability*. Cryptology ePrint Archive, Report 2018/276. https://eprint.iacr.org/2018/276. 2018.