The Quantum-Random-Oracle Model

Christian Schaffner

- Research Center for Quantum Software & Technology
- Theory of Computer Science (TCS) group
- Informatics Institute of University of Amsterdam

Goto
https://app.wooclap.com/QROM
for the quiz now!

Warsaw IACR summer school on Post-Quantum Cryptography 2024
Warsaw, Poland *Wednesday, 17 July 2024*

wants you!

CWI

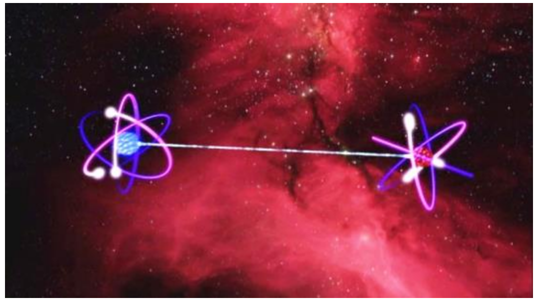
- Two faculty positions: to be announced shortly
- Talented Postdocs, PhD students
- University of Amsterdam: New Master program in

Quantum Computer Science

https://qusoft.org

What will you Learn from this Talk?

- Classical Random-Oracle Model
- Quantum Access
- Three Tools
- Extensions and Applications



Random Oracle (RO)

- A RO is a random function $f: \{0,1\}^n \to \{0,1\}^n$
- How many such functions are there? https://app.wooclap.com/QROM
  - *a)* $n!$
  - *b)* 2^n
  - *c)* $(2^n)!$
  - *d)* 2^{2^n}
  - *e)* $2^{n \cdot 2^n}$
- Truth table:
- Just specifying f requires exponentially many bits!

n columns

$f(0 \dots 00)$	0	1	1	…	0
$f(0 \dots 01)$	1	0	1	…	0
$f(0 \dots 10)$	0	0	1	…	1
$f(0 \dots 11)$	0	1	1	…	1
⋮				⋮	

2^n rows

Hash Functions

- A cryptographic hash function $\mathrm{H}: \{0,1\}^* \rightarrow \{0,1\}^n$
- Takes arbitrary-length input strings, outputs n bits.

Example SHA-3: $n = 256$ bits

H("The quick brown fox jumps over the lazy dog")=

0xf4202e3c5852f9182a0430fd8144f0a74b95e7417ecae17db0f8cfeed0e3e66e

H("The quick brown fox jumps over the lazy dof")=

0x853f4538be0db9621a6cea659a06c1107b1f83f02b13d18297bd39d7411cf10c

- An ideal hash function should behave as random oracle

Hash Functions As Random Oracles

- An ideal hash function should behave as random oracle

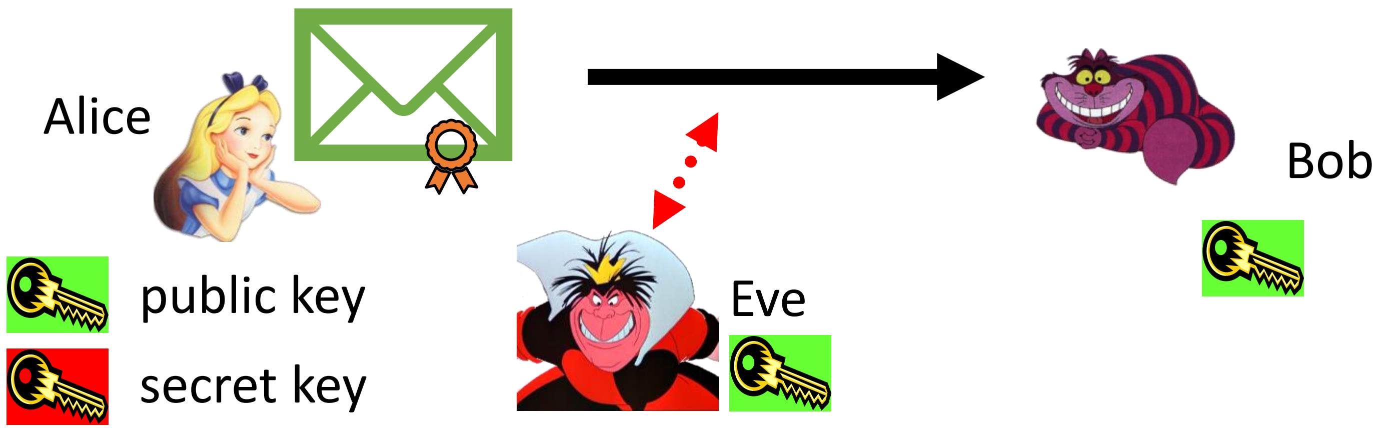
- Example: Collision-resistance
  Theorem: In a random function f, it is difficult to find two colliding inputs. Formally, for any adversary A making q queries to f, we have

$$\Pr[x \neq y \;\; and \;\; f(x) = f(y) \mid x, y \leftarrow A^f] \leq O\left(\frac{q^2}{2^n}\right)$$

- **Proof:** Let $\{x_1, x_2, \ldots, x_q\}$ be the list of A's distinct queries to f. For a random f, the outputs $f(x_i)$ are independent n-bit strings. The probability that two of them collide is $\frac{1}{2^n}$, and there are $\binom{q}{2} = O(q^2)$ pairs.

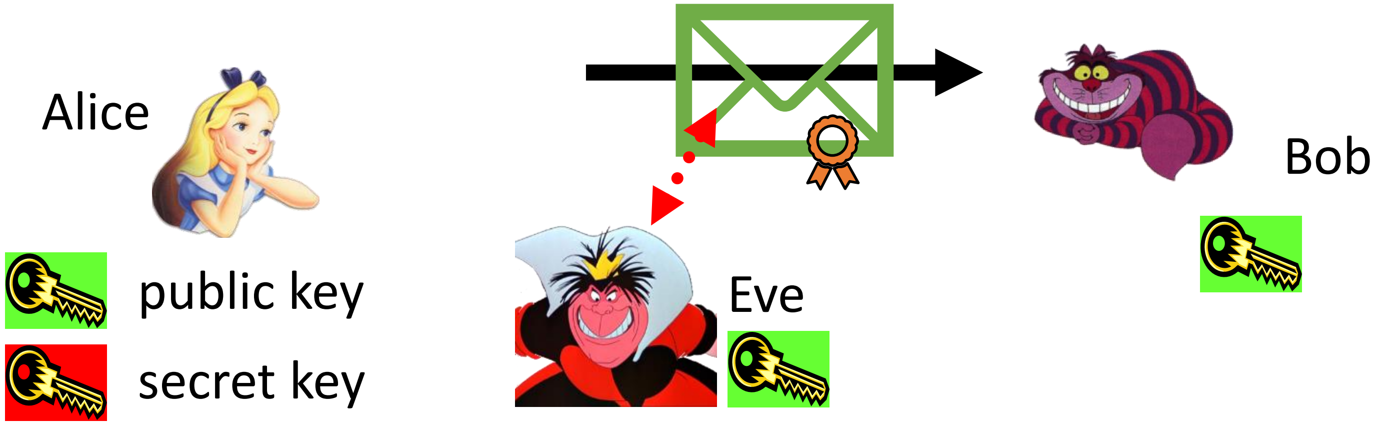
□

Digital Signatures



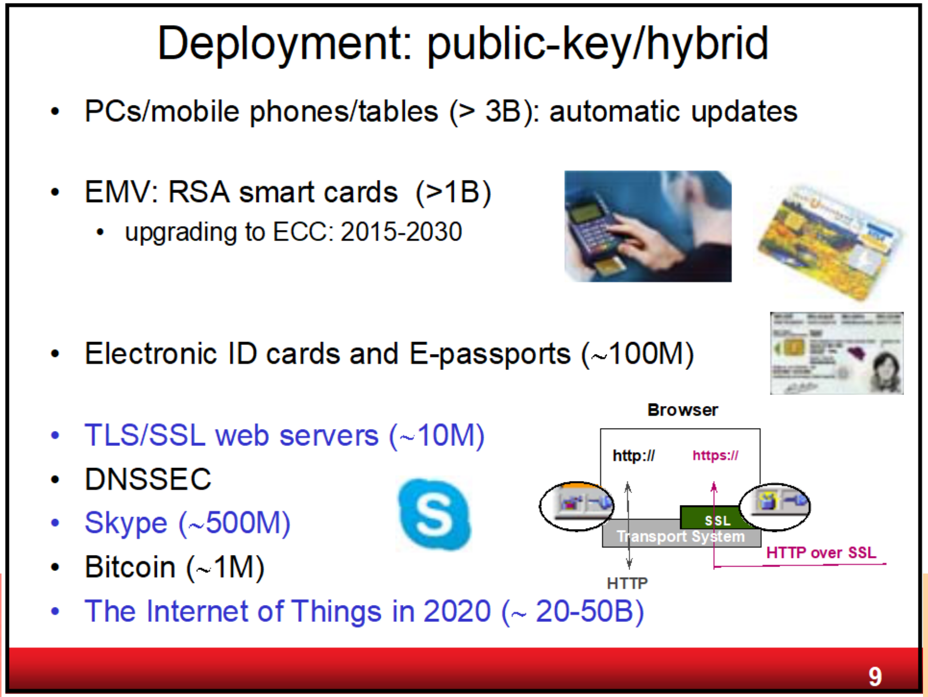
- Only secret-key holder can sign, but everyone can verify signatures using the public-key.

Digital Signatures



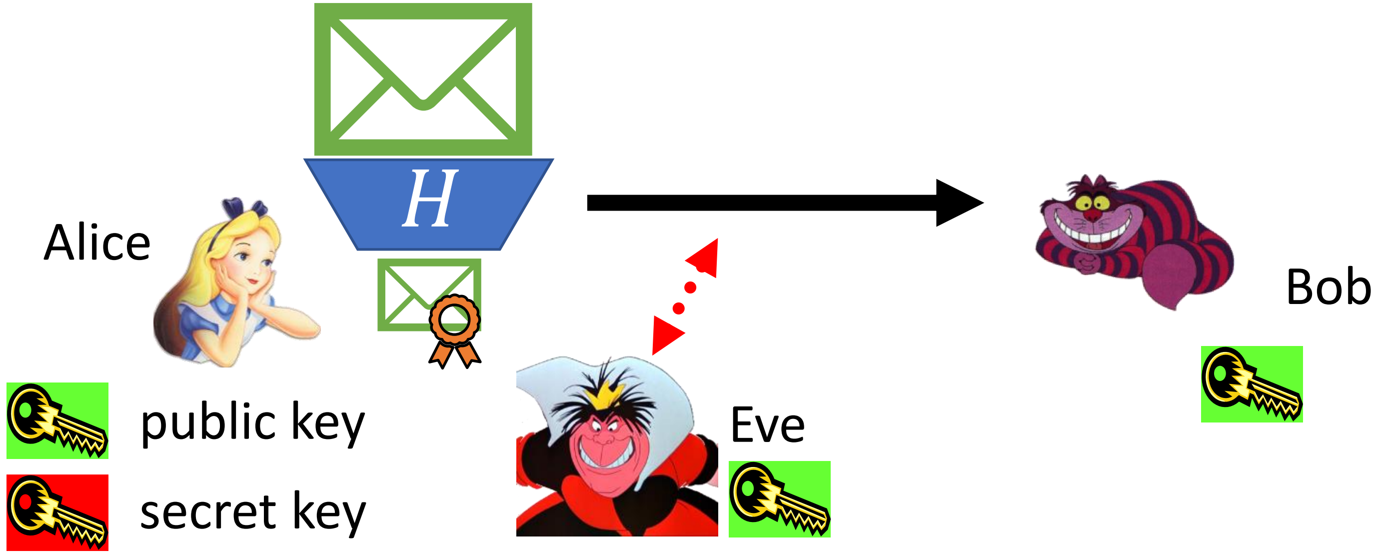
- Only secret-key holder can sign, but everyone can verify signatures using the public-key.

  - Very widely used:
  - Problems: expensive, insecure against quantum attacks



[Bart Preneel @QCrypt 2014]

Hash & Sign

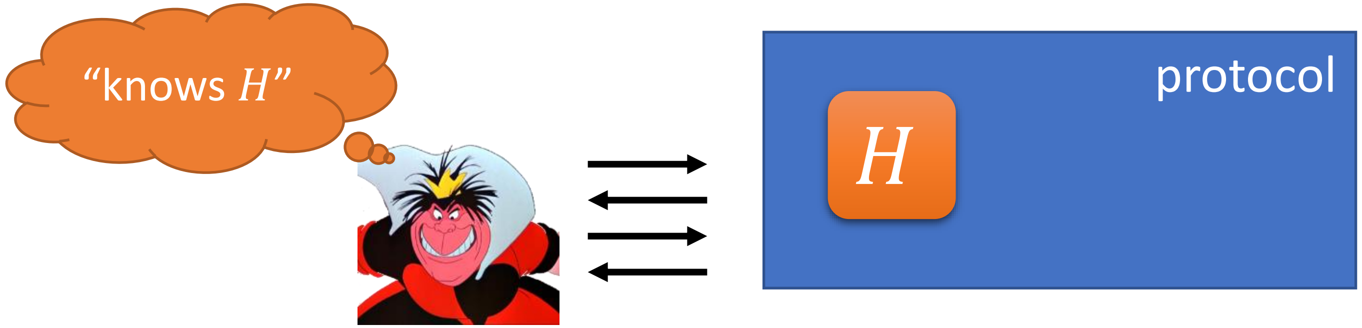


- Hash message to be signed, then digitally sign the hash
- **Theorem:** If H is a random oracle, then hash & sign is secure.
- **Proof sketch:** Let $\{x_1, x_2, \ldots, x_q\}$ be the list of Eve's queries to H. Either she finds a collision in H, or the security of sign applies. $\square$

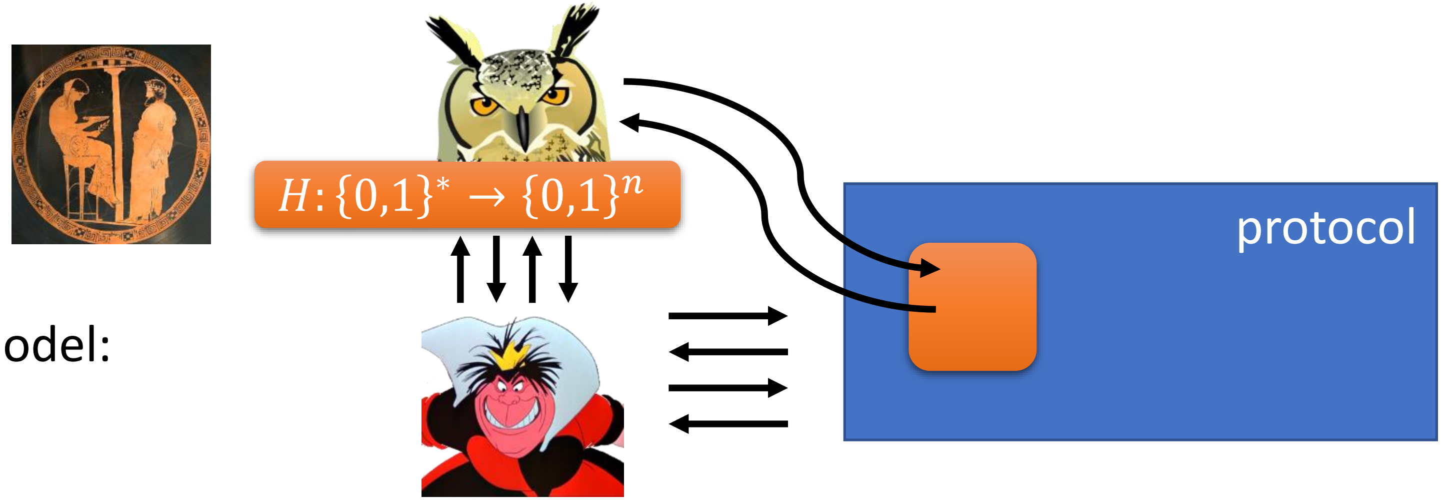
Random-Oracle Model (ROM)

- Heuristic to model hash functions in cryptographic proofs

Informal description:



Formal random-oracle model:

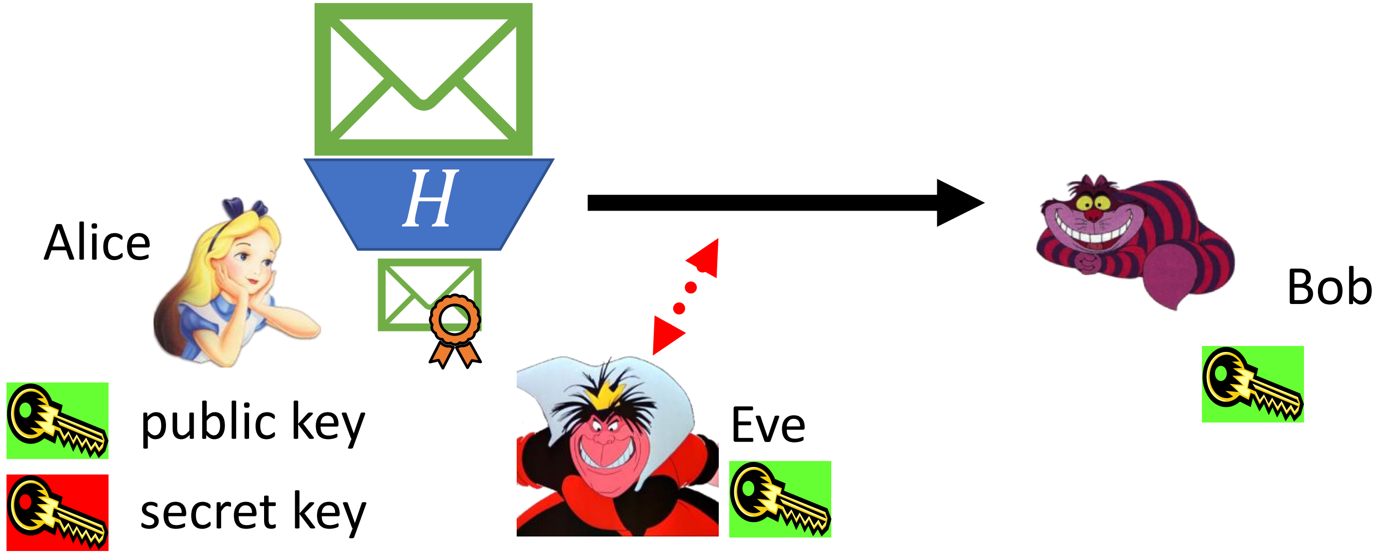


Every call to H is replaced with a query to RO.

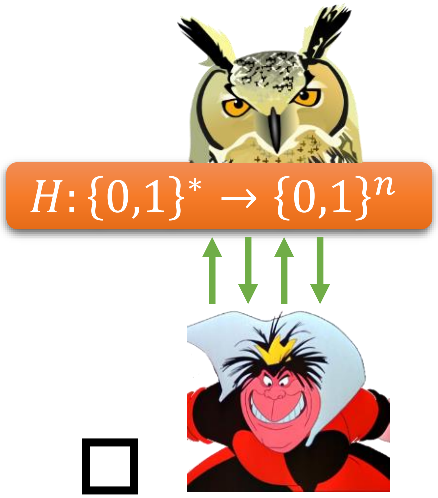
Adversarial queries are observed.

[Bellare Rogaway 93, slide by Dziembowski]

Hash & Sign in the Random Oracle Model



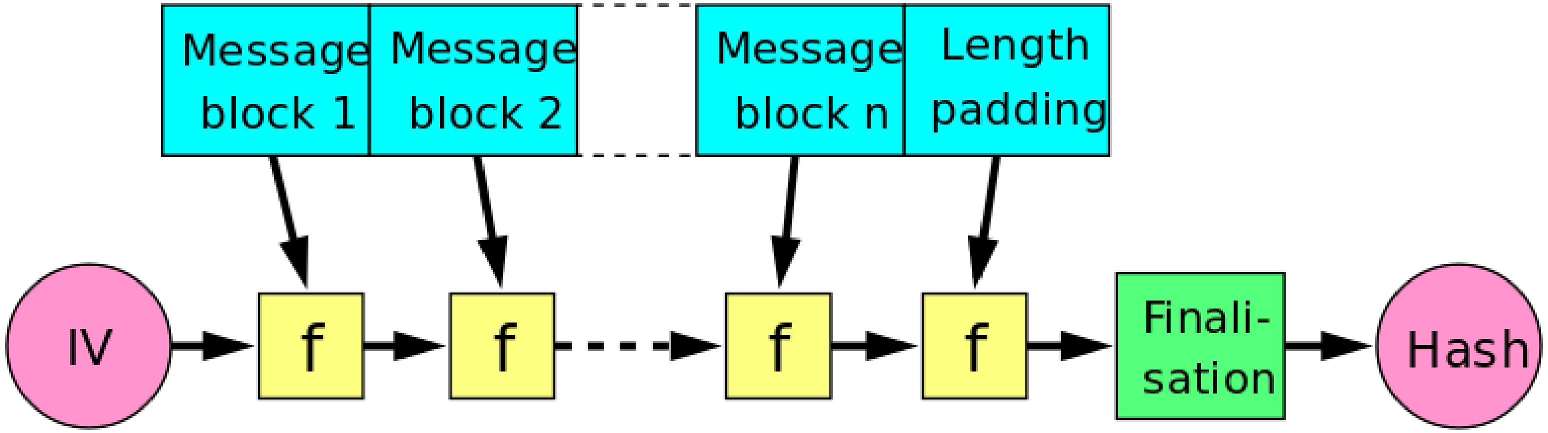
- Hash message to be signed, then digitally sign the hash
- **Theorem:** If H is a random oracle, then hash & sign is secure
- **Proof sketch:** Let $\{x_1, x_2, \ldots, x_q\}$ be the list of Eve's queries to H. Either she finds a collision in H, or the security of sign applies. □



Building Hash Functions

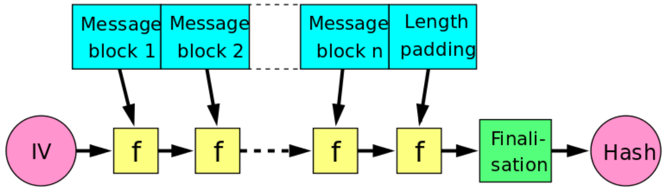
[Merkle Damgård 79]

- Secure building block $f: \{0,1\}^{2n} \rightarrow \{0,1\}^n$
- Construct a hash function $H: \{0,1\}^* \rightarrow \{0,1\}^n$ from it



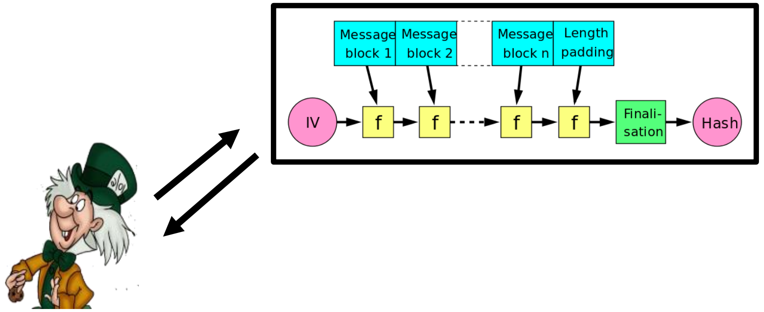
[Merkle Damgård 79, image: wikipedia]

Security Notions



- **Collision resistance**: If f is collision resistant, then so is H, obtained by the Merkle-Damgård construction.
  [5-line proof, exercise for crypto students 😀]
- **Indistinguishability**: If f is a random oracle, then H's input-output behavior is random, no efficient adv can distinguish interaction with H from interaction with RO.
  [follows from reasoning above 😀]

Real:



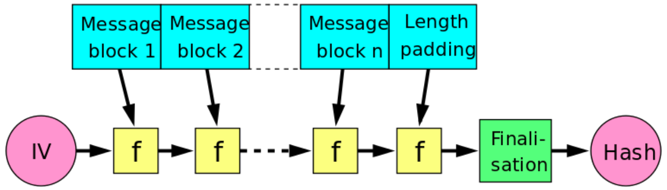
Dist

Ideal:

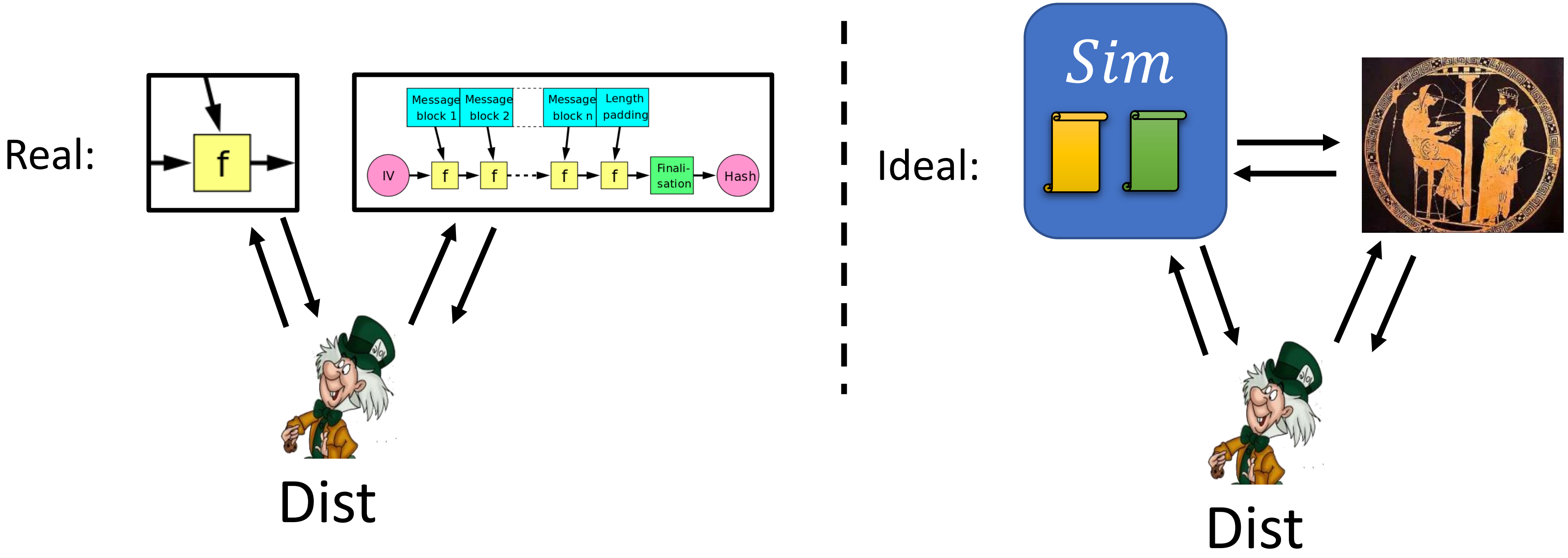
Dist

[Merkle Damgård 79, Bellare Canetti Krawczyk 05]

Indifferentiability

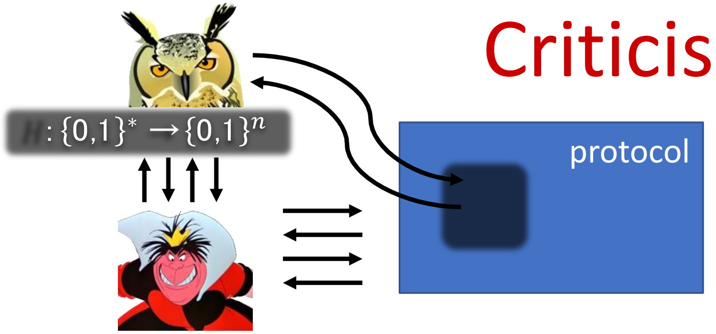


- **Indifferentiability**: If f is a random oracle, then there exists Sim such that no efficient adv can distinguish between interacting with (f, H) and (Sim^{RO}, RO)



[Maurer Renner Holenstein 04, Coron Dodis Malinaud Puniya 05]

Criticism of the Random-Oracle Model (ROM)



There exists a digital signature scheme that is

- **secure** in the ROM, but
- **not secure** if RO is instantiated with any real hash function.

- Very “artificial” example, no ”realistic” examples known
- Common view: ROM proof is better than no proof

[Canetti Goldreich Halevi 98, slide by Dziembowski]
[Matthew Green: https://blog.cryptographyengineering.com/2011/11/02/what-is-random-oracle-model-and-why/]

Classical Random-Oracle Model in Practice

- **Digital Signatures:** Fiat-Shamir Heuristic used by CRYSTALS-Dilithium, Hash-and-sign in FALCON
- **Public-Key Encryption:** KEMs are often built using the Fujisaki-Okamoto transform like in CRYSTALS-Kyber
- **Indifferentiability proofs**
- Etc.

https://app.wooclap.com/QROM

Example: Fiat-Shamir Transform

Schnorr in the lattice world [Lyu09,Lyu12]

$\boldsymbol{As} = \boldsymbol{u} \ (mod\ q)$ and $||\boldsymbol{s}|| \leq \beta$

$A, \boldsymbol{s}, \boldsymbol{u}$

$A, \boldsymbol{u}$

$\boldsymbol{y} \leftarrow \mathbb{Z}_q^m$

$\boldsymbol{w} = A\boldsymbol{y}$

$\boldsymbol{w}$ →

$c \leftarrow \boldsymbol{C} = \mathbb{Z}_q$

← c

$\boldsymbol{z} = \boldsymbol{y} + c\boldsymbol{s}$

$\boldsymbol{z}$ →

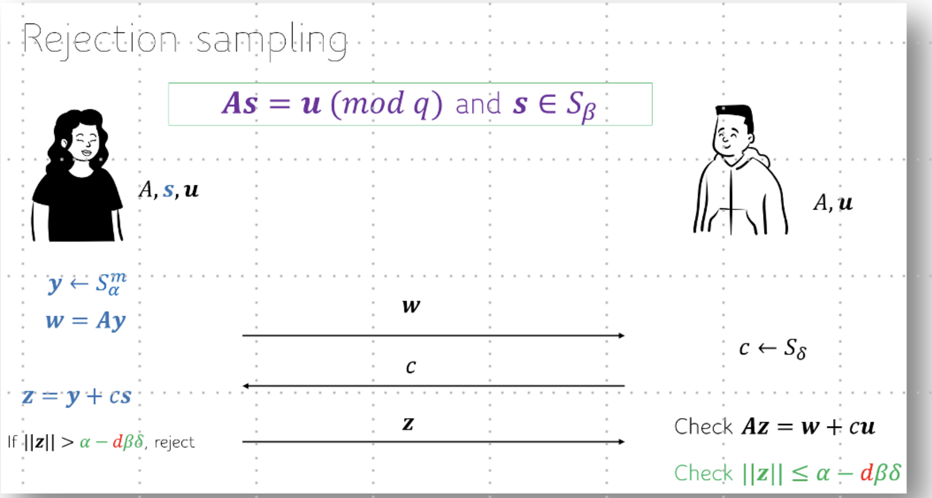
Check $\boldsymbol{Az} = \boldsymbol{w} + c\boldsymbol{u}$

[slide by Ngoc Khanh Nguyen]

Example: Fiat-Shamir Transform

Fiat-Shamir transformation

- Let $H: \{0,1\}^* \rightarrow S_\delta$ be a hash function.
- We obtain a *non-interactive proof* as follows.



1. $\boldsymbol{y} \leftarrow S_\alpha^m$
2. $\boldsymbol{w} = A\boldsymbol{y}$
3. $c = H((A, \boldsymbol{u}), \boldsymbol{w})$
4. $\boldsymbol{z} = \boldsymbol{y} + c\boldsymbol{s}$
5. If $||\boldsymbol{z}|| > \alpha - d\beta\delta$, restart
6. Output $\pi = (\boldsymbol{w}, \boldsymbol{z})$.

Proof size: $n + m$ ring elements

To verify $\pi = (\boldsymbol{w}, \boldsymbol{z})$, check:

1. $||\boldsymbol{z}|| \leq \alpha - d\beta\delta$ and $A\boldsymbol{z} = \boldsymbol{w} + c\boldsymbol{u}$ where $c = H\big((A, \boldsymbol{u}), \boldsymbol{w}\big)$

[slide by Ngoc Khanh Nguyen]

Example: Fiat-Shamir Transform

Zero-knowledge in ROM

- No efficient adversary can distinguish between valid proofs and simulated proofs

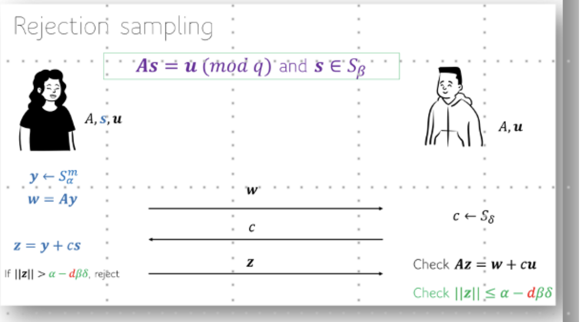
Simulate:

1. $\boldsymbol{z} \leftarrow [\alpha - d\beta\delta, \alpha + d\beta\delta]$
2. $c \leftarrow S_\delta$
3. $\boldsymbol{w} := \boldsymbol{Az} - c\boldsymbol{u}$.
4. Program $H((\boldsymbol{A}, \boldsymbol{u}), \boldsymbol{w}) := c$
5. Output $\pi := (c, \boldsymbol{z})$.

Simple entropy argument to show that we will never ``overwrite'' the random oracle

Fiat-Shamir transformation

- Let $H: \{0,1\}^* \rightarrow S_\delta$ be a hash function.
- Optimisation:



1. $\boldsymbol{y} \leftarrow S_\alpha^m$
2. $\boldsymbol{w} = \boldsymbol{Ay}$
3. $c = H((\boldsymbol{A}, \boldsymbol{u}), \boldsymbol{w})$
4. $\boldsymbol{z} = \boldsymbol{y} + c\boldsymbol{s}$
5. If $||\boldsymbol{z}|| > \alpha - d\beta\delta$, restart
6. Output $\pi = (c, \boldsymbol{z})$.

Proof size: $1 + m$ ring elements

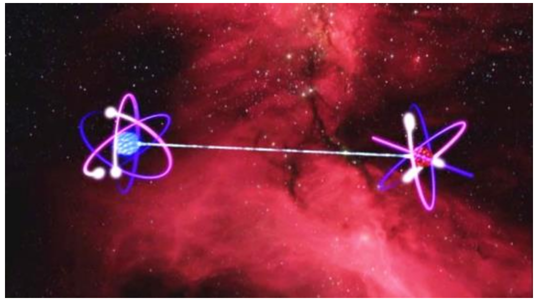
To verify $\pi = (c, \boldsymbol{z})$, check:

1. $||\boldsymbol{z}|| \leq \alpha - d\beta\delta$ and $c = H((\boldsymbol{A}, \boldsymbol{u}), \boldsymbol{Az} - c\boldsymbol{u})$.

[slide by Ngoc Khanh Nguyen]

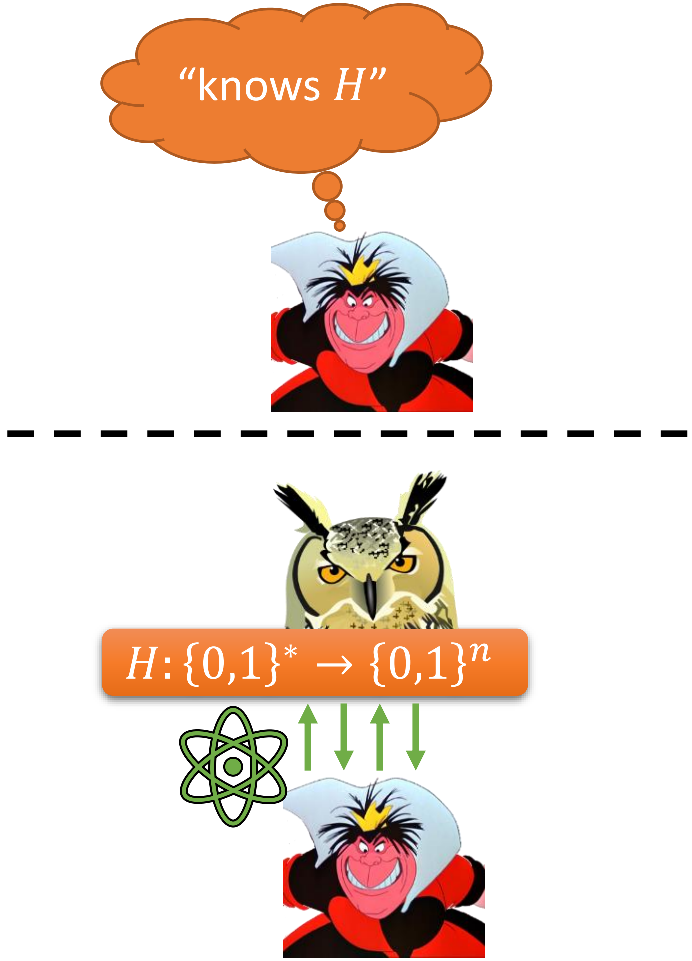
What will you Learn from this Talk?

- ✓ Classical Random-Oracle Model
- Quantum Access
- Three Tools
- Extensions and Applications



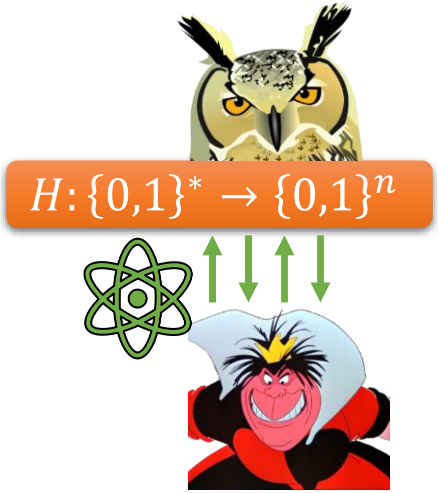
Quantum-Random-Oracle Model (QROM)

- Post-quantum cryptography or quantum-safe cryptography studies quantum attackers on classical crypto.
- Attacker can look up description of hash function on Wikipedia, then run it in superposition on her quantum computer
- We need to allow attacker quantum (superposition) access to the random oracle

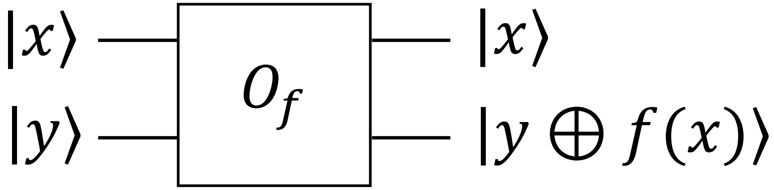


[Boneh Dagdelen Fischlin Lehmann Schaffner Zhandry 11]

Quantum Superposition Access



- Quantum attacker may query RO in superposition:
  - Standard oracle (StO): $|x\rangle_X |y\rangle_Y \mapsto |x\rangle_X |y \oplus f(x)\rangle_Y$

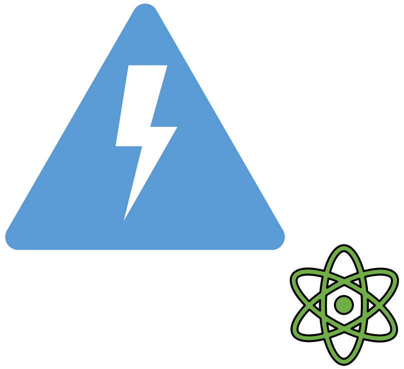


- Example: superposition over all inputs

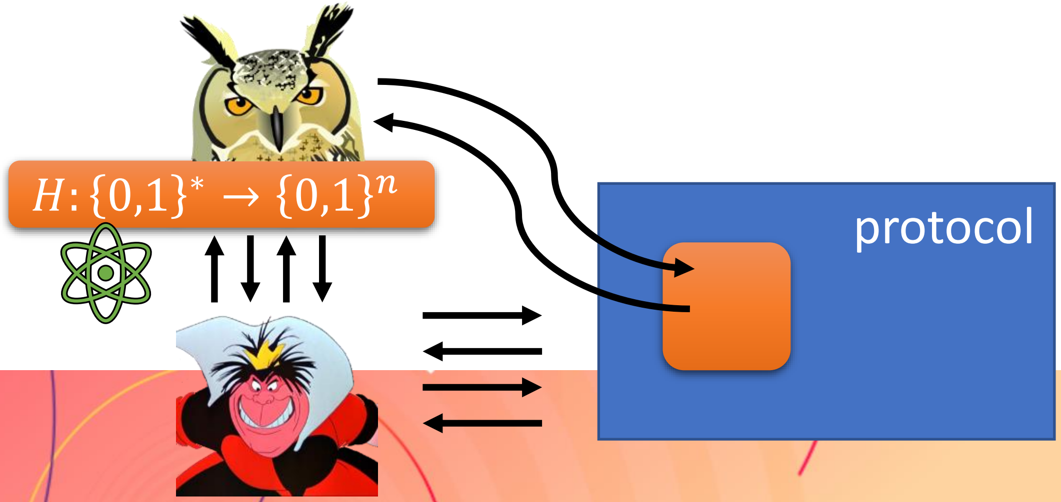
$$\frac{1}{2^{n/2}} \sum_{x \in \{0,1\}^n} |x\rangle\, |0\rangle \mapsto \frac{1}{2^{n/2}} \sum_{x \in \{0,1\}^n} |x\rangle\, |f(x)\rangle$$

- With a single quantum query, Eve can access all function values in superposition.

Trouble in the QROM



- **Many classical ROM proofs break down in the QROM**
- Efficient simulation: how to emulate a RO towards an adversary
- Adaptive programmability: depending on the adversary's queries, plant a challenge in the answer
- Extractability: Simulator learns pre-images of adversary's queries
- Rewinding: replaying some hash values but changing some outputs



[Boneh Dagdelen Fischlin Lehmann Schaffner Zhandry 11]

Example: Fiat-Shamir Transform

Zero-knowledge in ROM

- No efficient adversary can distinguish between valid proofs and simulated proofs

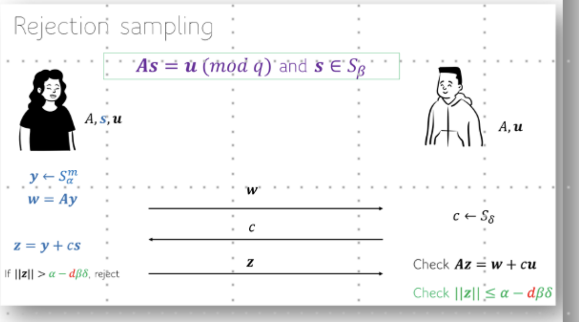
Simulate:

1. $\boldsymbol{z} \leftarrow [\alpha - d\beta\delta, \alpha + d\beta\delta]$
2. $c \leftarrow S_\delta$
3. $\boldsymbol{w} := \boldsymbol{Az} - c\boldsymbol{u}$.
4. Program $H((\boldsymbol{A}, \boldsymbol{u}), \boldsymbol{w}) := c$
5. Output $\pi := (c, \boldsymbol{z})$.

Simple entropy argument to show that we will never ``overwrite'' the random oracle

Fiat-Shamir transformation

- Let $H: \{0,1\}^* \rightarrow S_\delta$ be a hash function.
- Optimisation:



1. $\boldsymbol{y} \leftarrow S_\alpha^m$
2. $\boldsymbol{w} = \boldsymbol{Ay}$
3. $c = H((\boldsymbol{A}, \boldsymbol{u}), \boldsymbol{w})$
4. $\boldsymbol{z} = \boldsymbol{y} + c\boldsymbol{s}$
5. If $||\boldsymbol{z}|| > \alpha - d\beta\delta$, restart
6. Output $\pi = (c, \boldsymbol{z})$.

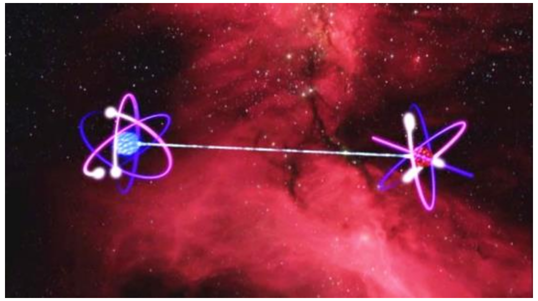
To verify $\pi = (c, \boldsymbol{z})$, check:
1. $||\boldsymbol{z}|| \leq \alpha - d\beta\delta$ and $c = H((\boldsymbol{A}, \boldsymbol{u}), \boldsymbol{Az} - c\boldsymbol{u})$.

Proof size: $1 + m$ ring elements

[slide by Ngoc Khanh Nguyen]

What will you Learn from this Talk?

- ✓ Classical Random-Oracle Model
- ✓ Quantum Access
- Three Tools
- Extensions and Applications



Tool 1: q-wise independent functions

- A function family $\mathcal{F} \subset \{f\colon \{0,1\}^n \to \{0,1\}^n\}$ is called **t-wise independent** if for t distinct inputs $\{x_1, x_2, \dots, x_t\}$, the values $\left(f(x_1), f(x_2), \dots, f(x_t)\right)$ for $f \leftarrow \mathcal{F}$ are independent and uniform.

- Example Construction: The family $\mathcal{F}_{\vec{a}} = \{f(x) = a_0 + a_1 x + \cdots + a_{t-2}x^{t-2} + a_{t-1}x^{t-1}\}$ where $\vec{a} = (a_0, a_1, \dots, a_{t-1}) \in GF(2^n)^t$ is t-wise independent.

- **Theorem:** Let $\mathcal{F}$ be a 2q-wise independent function family. For any q-query quantum algorithm A: $\Pr_{H \leftarrow RO}[1 \leftarrow A^H] = \Pr_{f \leftarrow \mathcal{F}}\left[1 \leftarrow A^f\right]$

Tool 1: Simulating RO to a quantum adversary

- **Theorem:** Let $\mathcal{F}$ be a 2q-wise independent function family. For any q-query quantum algorithm A: $\Pr_{H \leftarrow RO}[1 \leftarrow A^H] = \Pr_{f \leftarrow \mathcal{F}}[1 \leftarrow A^f]$

Proof (extension of the polynomial method):

- [Zhandry 19] The quantity $\Pr_{f \leftarrow \mathcal{F}}[1 \leftarrow A^f]$ is a linear combination of the quantities $\left\{ \Pr_{f \leftarrow \mathcal{F}}[f(x_i) = y_i \ \forall i \in \{1,2,\dots,2q\}] \right\}_{x_i, y_i}$
- These quantities are identical for f fully random and $f \leftarrow \mathcal{F}$.

$\square$

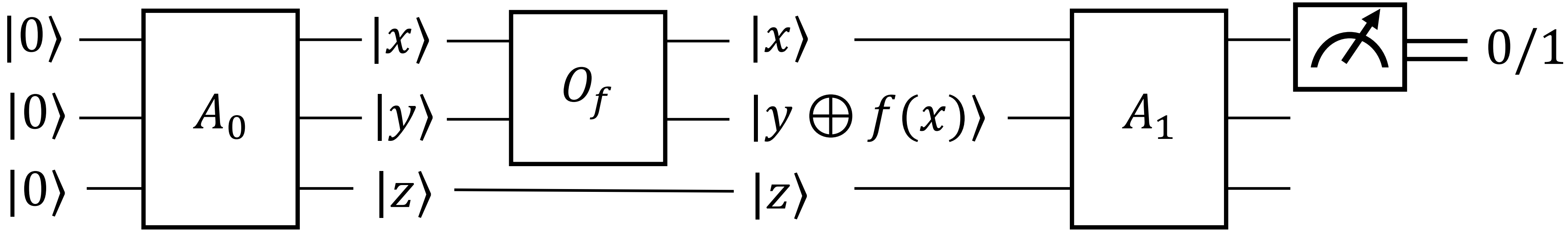
[Ronald de Wolf's lecture notes, Chapter 11, https://arxiv.org/abs/1907.09415]

[Zhandry 19: https://eprint.iacr.org/2012/076, Theorem 6.1]

Tool 1: Simulating RO to a quantum adversary

[Zhandry 19] The quantity $\Pr_{f \leftarrow \mathcal{F}}\left[1 \leftarrow A^f\right]$ is a linear combination of the quantities

$$\beta_{x_1 y_1 x_2 y_2} \coloneqq \Pr_{f \leftarrow \mathcal{F}}[f(x_1) = y_1, f(x_2) = y_2]$$



$$\sum \alpha_{xyz}|x\rangle|y\rangle|z\rangle \rightarrow \sum \alpha_{xyz}|x\rangle|y \oplus f(x)\rangle|z\rangle = \sum \alpha_{xyz} \sum_{y'} \Pr[f(x) = y']\,|x\rangle|y \oplus y'\rangle|z\rangle$$

$$= \sum \alpha_{xyz} \sum_{y'} \beta_{xy'}\,|x\rangle|y \oplus y'\rangle|z\rangle = \sum \alpha_{xy'z} \sum_{y} \beta_{xy' \oplus y}\,|x\rangle|y'\rangle|z\rangle$$

[Ronald de Wolf's lecture notes, Chapter 11, https://arxiv.org/abs/1907.09415]

[Zhandry 19: https://eprint.iacr.org/2012/076, Theorem 6.1]

Tool 1: Simulating RO to a quantum adversary

- **Theorem:** Let $\mathcal{F}$ be a 2q-wise independent function family. For any q-query quantum algorithm A: $\Pr_{H \leftarrow RO}[1 \leftarrow A^H] = \Pr_{f \leftarrow \mathcal{F}}\left[1 \leftarrow A^f\right]$

Proof (extension of the polynomial method):

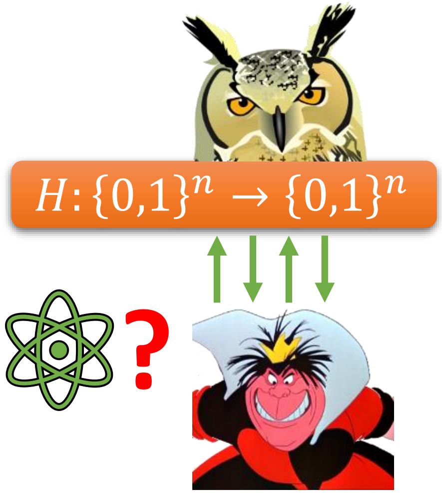
- [Zhandry 19] The quantity $\Pr_{f \leftarrow \mathcal{F}}\left[1 \leftarrow A^f\right]$ is a linear combination of the quantities $\left\{\Pr_{f \leftarrow \mathcal{F}}[f(x_i) = y_i \ \forall i \in \{1,2,\ldots,2q\}]\right\}_{x_i, y_i}$
- These quantities are identical for f fully random and $f \leftarrow \mathcal{F}$.

$\square$

[Ronald de Wolf's lecture notes, Chapter 11, https://arxiv.org/abs/1907.09415]
[Zhandry 19: https://eprint.iacr.org/2012/076, Theorem 6.1]

Tool 2: One-way to Hiding Lemma (O2H)

- Illustrating example: Security of $Enc(m) := \big(f(r), m \oplus H(r)\big)$
- Security game: IND-CPA security: $\Pr[win\ G1] \approx 1/2$ where



Game 1:

1. $H \leftarrow RO,\ b \leftarrow \{0,1\},\ r \leftarrow \{0,1\}^n$
2. $m_0, m_1 \leftarrow A^H$
3. $b' \leftarrow A^H\big(f(r), m_b \oplus H(r)\big)$
4. $win := [b' = b]$

Game 2:

1. $H \leftarrow RO,\ b \leftarrow \{0,1\},\ r \leftarrow \{0,1\}^n,\ y \leftarrow \{0,1\}^n$
2. $m_0, m_1 \leftarrow A^H$
3. $b' \leftarrow A^H(f(r), m_b \oplus y)$
4. $win := [b' = b]$

- Note: $\Pr[win\ G2] = 1/2$ because y acts as one-time pad to hide m_b
- In classical ROM, we can argue that

$$|\Pr[win\ G1] - \Pr[win\ G2]| \le \Pr[H(r)\ is\ queried\ in\ G2] \approx 0 \quad \square$$

Example from [Unruh 22: https://www.youtube.com/watch?v=YAIFGB4ryGE]

Tool 2: One-way to Hiding Lemma (O2H)

To show: $\Pr[win\ G1] \approx 1/2$ Note: $\Pr[win\ G2] = 1/2$

Game 1:

1. $H \leftarrow RO,\ b \leftarrow \{0,1\},\ r \leftarrow \{0,1\}^n$
2. $m_0, m_1 \leftarrow A^H$
3. $b' \leftarrow A^H\big(f(r), m_b \oplus H(r)\big)$
4. $win := [b' = b]$

Game 2:

1. $H \leftarrow RO,\ b \leftarrow \{0,1\},$ $r \leftarrow \{0,1\}^n,\ y \leftarrow \{0,1\}^n$
2. $m_0, m_1 \leftarrow A^H$
3. $b' \leftarrow A^H(f(r), m_b \oplus y)$
4. $win := [b' = b]$

Game 3: run A^H *in*

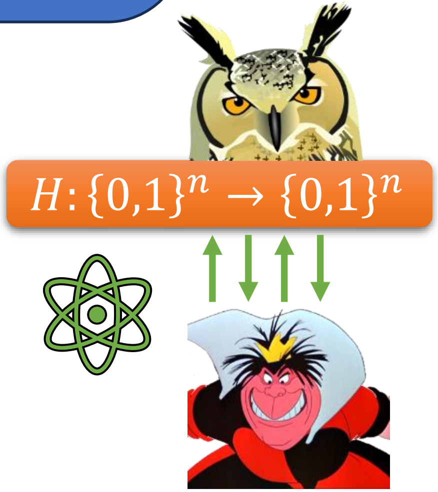
1. $H \leftarrow RO,\ b \leftarrow \{0,1\},\ r \leftarrow \{0,1\}^n,$ $y \leftarrow \{0,1\}^n, i \leftarrow \{1, 2, \dots, q\}$
2. $m_0, m_1 \leftarrow A^H$
3. $b' \leftarrow A^H(f(r), m_b \oplus y)$
4. $r' \leftarrow measure\ query\ i$
5. $win := [r' = r]$

- **Theorem:** [original O2H, Unruh 15]

  Fix a q-query adversary A^H. Let B^H run A^H until the i-th query for random $i \leftarrow \{1,2,\dots,q\}$, measure the query register. Then, for random x, y

$$\left|\Pr\big[A^H\big(x, H(x)\big) = 1\big] - \Pr[A^H(x,y) = 1]\right| \leq q\sqrt{\Pr[B^H(x,y) = x]}$$

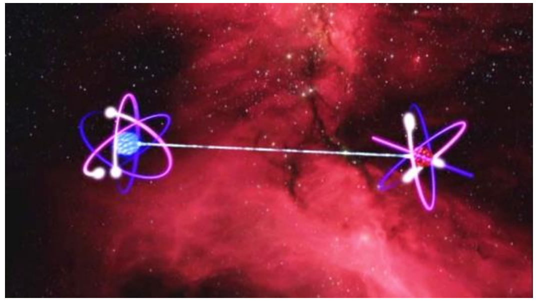
$$|\Pr[win\ G1] - \Pr[win\ G2]| \leq q\sqrt{\Pr[win\ G3]} \approx 0$$



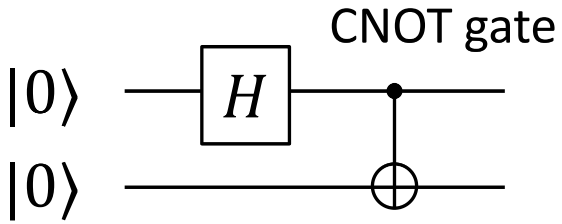
[Unruh 15: https://eprint.iacr.org/2013/606.pdf]

What will you Learn from this Talk?

- ✓ Classical Random-Oracle Model
- ✓ Quantum Access
- Three Tools:
  - ✓ t-wise independent functions
  - ✓ One-way to Hiding (O2H)
  - Compressed oracles
- Applications



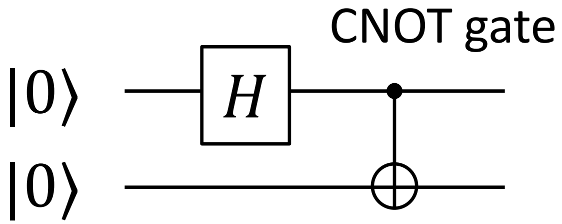
Quantum: Warm-Up



- What is the resulting quantum state?

https://app.wooclap.com/QROM

Quantum: Warm-Up



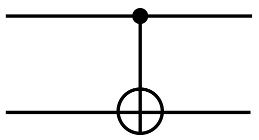
$$\frac{|00\rangle + |11\rangle}{\sqrt{2}}$$

- What is the resulting quantum state?

https://app.wooclap.com/QROM

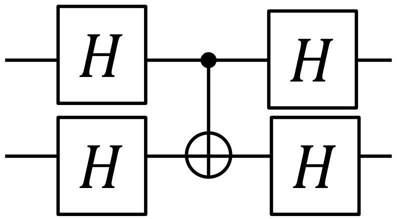
Quantum Circuit Identity

CNOT gate

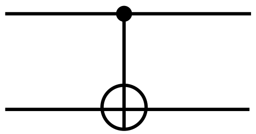


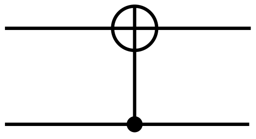
$$|00\rangle \mapsto |00\rangle$$
$$|01\rangle \mapsto |01\rangle$$
$$|10\rangle \mapsto |11\rangle$$
$$|11\rangle \mapsto |10\rangle$$

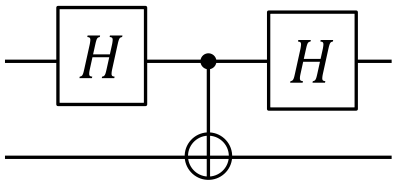
$$|++\rangle \mapsto |++\rangle$$
$$|+-\rangle \mapsto |--\rangle$$
$$|-+\rangle \mapsto |-+\rangle$$
$$|--\rangle \mapsto |+-\rangle$$

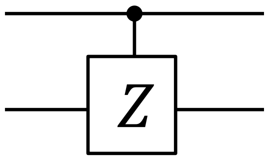


= ?





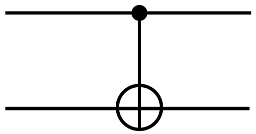




https://app.wooclap.com/QROM

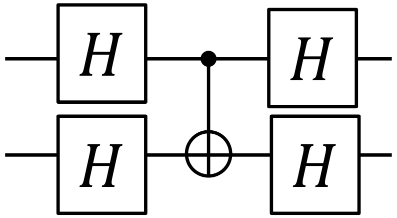
Quantum Circuit Identity

CNOT gate

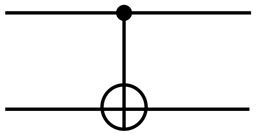


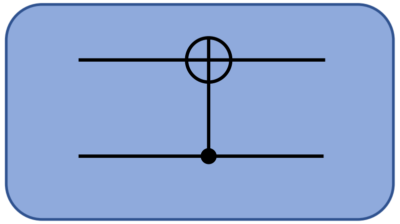
$$|00\rangle \mapsto |00\rangle \quad |01\rangle \mapsto |01\rangle \quad |10\rangle \mapsto |11\rangle \quad |11\rangle \mapsto |10\rangle$$

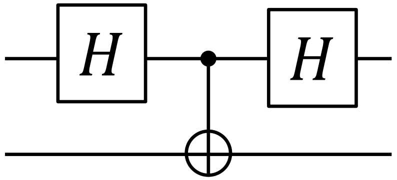
$$|+\,+\rangle \mapsto |+\,+\rangle \quad |+\,-\rangle \mapsto |-\,-\rangle \quad |-\,+\rangle \mapsto |-\,+\rangle \quad |-\,-\rangle \mapsto |+\,-\rangle$$

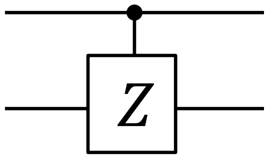


$= \; ?$





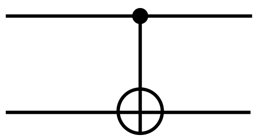




https://app.wooclap.com/QROM

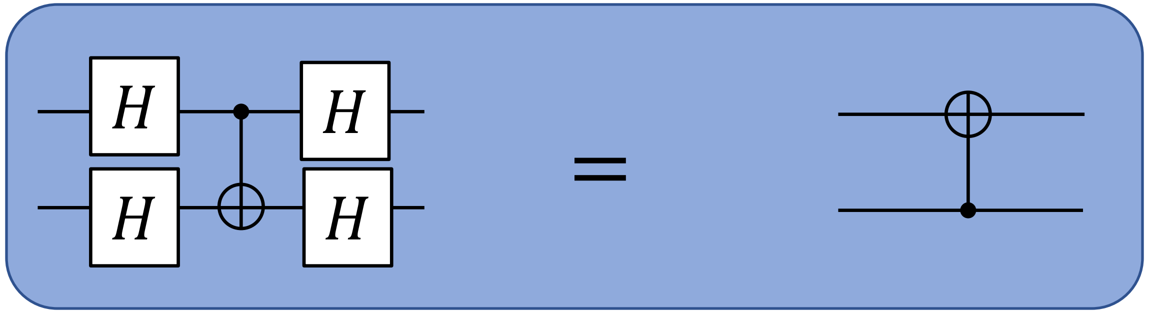
Quantum Circuit Identity

CNOT gate



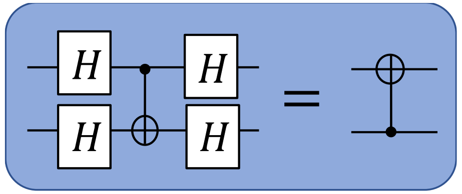
$$\begin{aligned} |00\rangle &\mapsto |00\rangle \\ |01\rangle &\mapsto |01\rangle \\ |10\rangle &\mapsto |11\rangle \\ |11\rangle &\mapsto |10\rangle \end{aligned}$$

$$\begin{aligned} |+\,+\rangle &\mapsto |+\,+\rangle \\ |+\,-\rangle &\mapsto |-\,-\rangle \\ |-\,+\rangle &\mapsto |-\,+\rangle \\ |-\,-\rangle &\mapsto |+\,-\rangle \end{aligned}$$

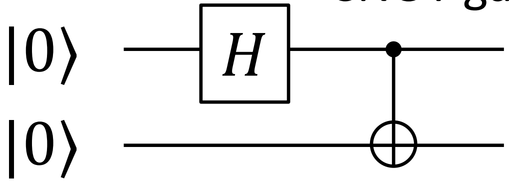


- When viewed "in the Hadamard basis", the control and target of the CNOT are swapped!

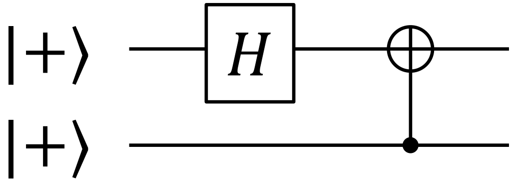
Quantum: Warm-Up Revisited



CNOT gate



$$\frac{|00\rangle + |11\rangle}{\sqrt{2}}$$

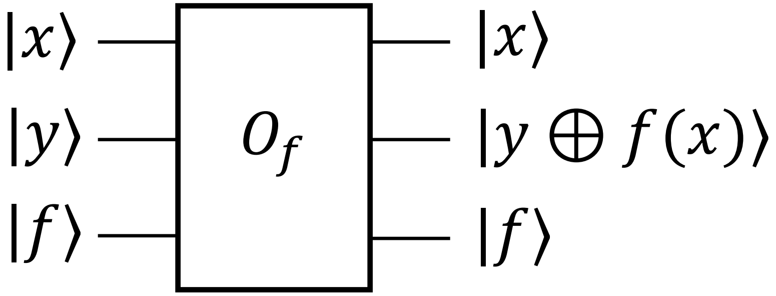


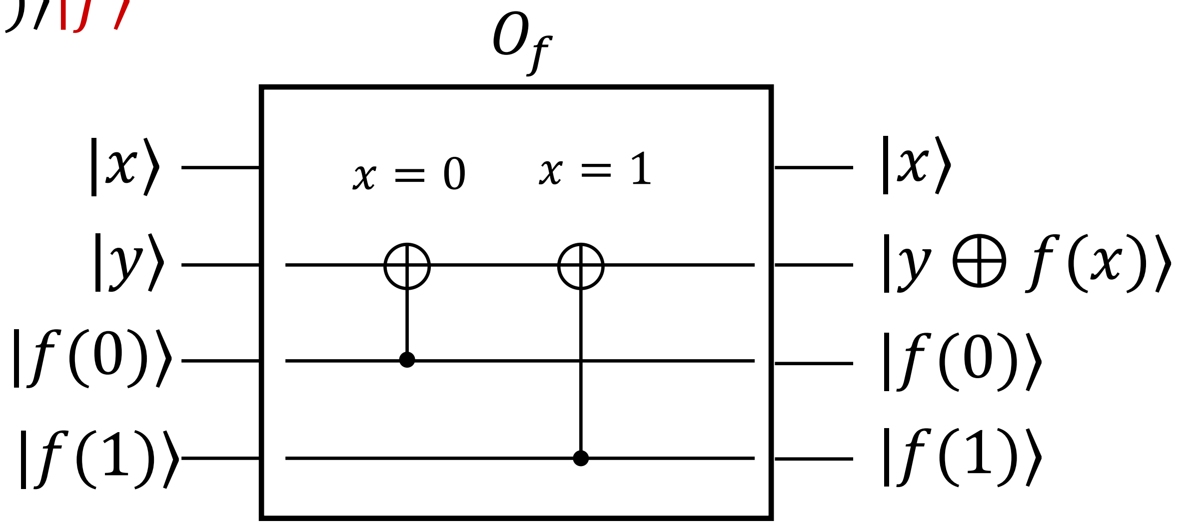
$$\frac{|{+}{+}\rangle + |{-}{-}\rangle}{\sqrt{2}} = \frac{|00\rangle + |11\rangle}{\sqrt{2}}$$

A Crucial Insight: Purification

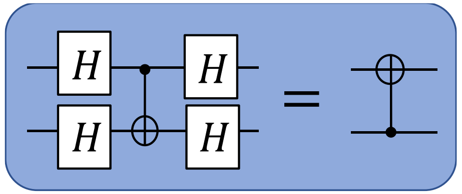
- Introduce purifying register for function truth table

$$\text{StO: } \sum_f |x\rangle|y\rangle|f\rangle \mapsto \sum_f |x\rangle|y \oplus f(x)\rangle|f\rangle$$



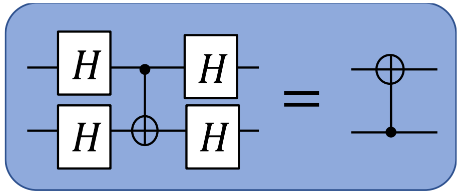


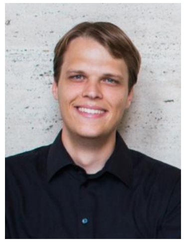
- For a random oracle f, we have a superposition over all truth tables
- From Eve's point of view, there is no difference!



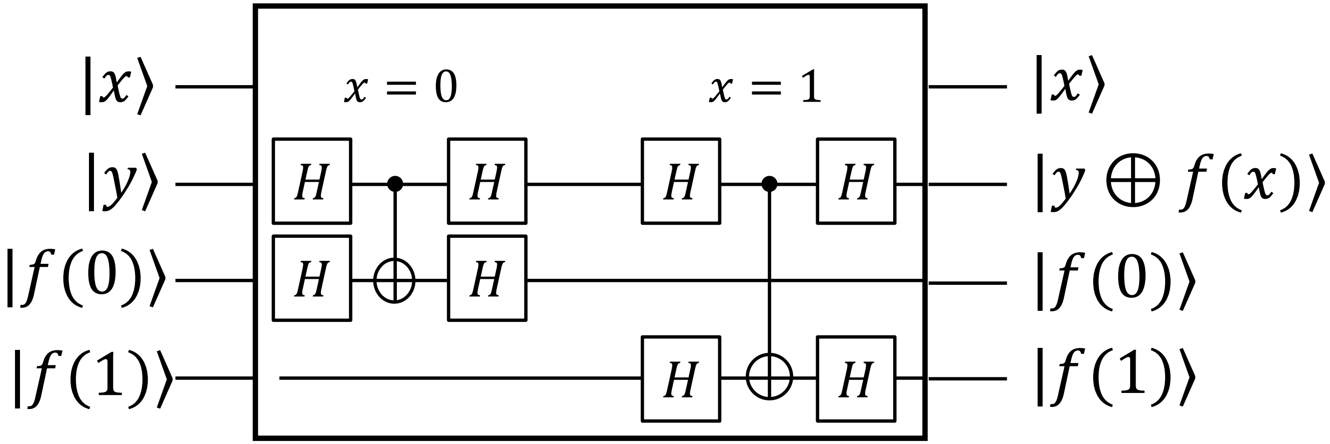
[Mark Zhandry 2018: How to Record Quantum Queries]

Change of Viewpoint: Fourier Oracle

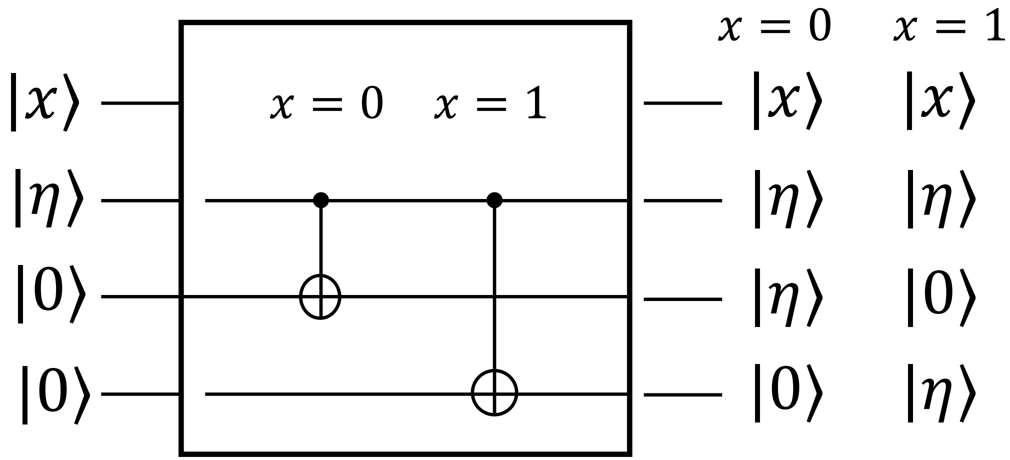




Standard Oracle



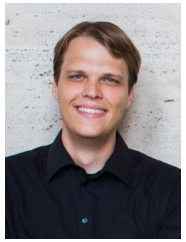
Fourier Oracle



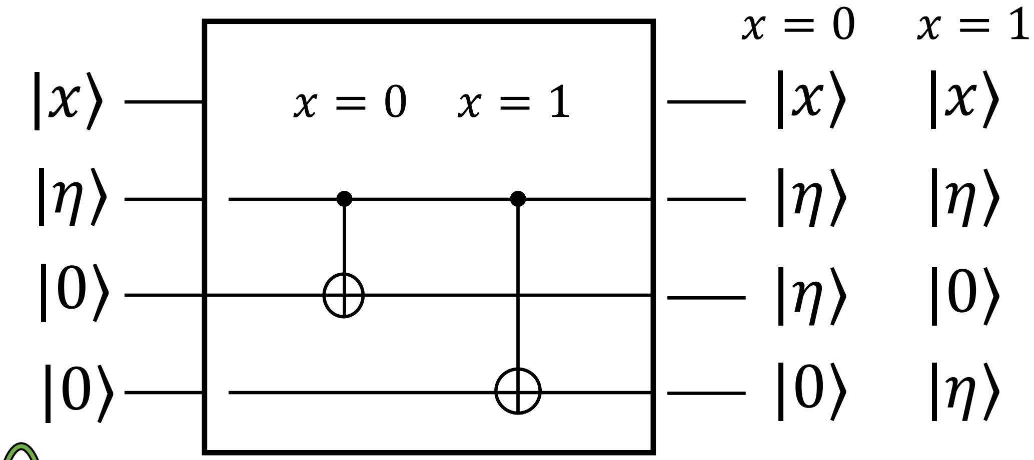
- By making a query, Eve entangles herself with the truth table in a very clean way, when observed in the Fourier basis!

[Mark Zhandry 2018: How to Record Quantum Queries]

Compressing the Database



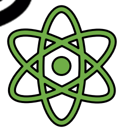
Fourier Oracle



- Fourier Oracle (FO):

$$|x\rangle|\eta\rangle|0^n \cdots 0^n\rangle \xrightarrow{FO} |x\rangle|\eta\rangle|0^n \cdots 0^n\ \eta\ 0^n \cdots 0^n\rangle$$

- Multiple FO queries “populate” the database



- Compression: only keep track of the non-zero entries

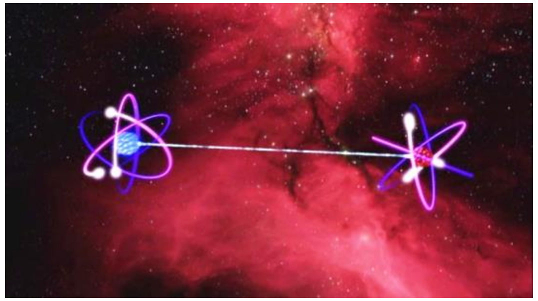
$$|x\rangle|\eta\rangle|D\rangle \xrightarrow{FO} |x\rangle|\eta\rangle|D \cup (x, \eta)\rangle$$

- Allows efficient simulation of the random oracle to the adversary

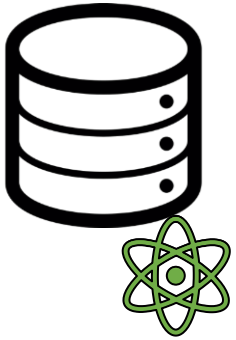
[Mark Zhandry 2018: How to Record Quantum Queries]

What will you Learn from this Talk?

- ✓ Classical Random-Oracle Model
- ✓ Quantum Access
- ✓ Three Tools:
  - ✓ t-wise independent functions
  - ✓ One-way to Hiding (O2H)
  - ✓ Compressed oracles
- Extensions and Applications



Query Lower Bounds



- Intuition: The quantum queries are recorded in the database, an adversary can only learn about the function what is recorded there
- **Theorem:** For any quantum player making q queries, if the database D is measured after the q queries, the probability that it contains a pair $(x, 0^n)$ is at most $O\left(\frac{q^2}{2^n}\right)$.
- **Idea:** Track the norm of the state projected onto D containing a zero. It starts at 0, and every query increases it by at most $\frac{1}{2^{n/2}}$. After q queries, its norm is at most $\frac{q}{2^{n/2}}$. $\square$
- Using newer tools from [Chung Fehr Huang Liao 21], such reasoning is almost classical.

[Mark Zhandry 2018: How to Record Quantum Queries, Chung, Fehr, Huang, Liao 2021]

Extensions and More Advanced Tools

- Tool 1: t-wise independent function families:
  semi-constant distributions, small-range distributions, …
- Tool 2: one-way to hiding
  semi-classical O2H, many variants
- Measure and reprogram tools for Fiat-Shamir [PhD thesis by Jelle Don 24]
- Tool 3: compressed oracles
  online extraction, (tight) adaptive reprogramming
  compressed permutation oracles? Ideal-cipher model?

[Mark Zhandry 2018: How to Record Quantum Queries, Chung, Fehr, Huang, Liao 2021]

[Ambainis Hamburg Unruh 18]

Numerous Applications

- Query lower bounds for searching and (multi-)collisions in random functions
- Fujisaki-Okamoto transformation (to build public-key encryption)
- Fiat-Shamir transform (for digital signatures)
- Indifferentiability
- 4-round Luby-Rackoff/Feistel construction
- Succinct arguments
- Separations between ROM and QROM
- Adaptive reprogramming
- …

Major Open Question: Compressed Permutation Oracles

Paper 2024/1140

Permutation Superposition Oracles for Quantum Query Lower Bounds

Christian Majenz, Technical University of Denmark
Giulio Malavolta, Bocconi University, Max Planck Institute for Security and Privacy
Michael Walter, Ruhr University Bochum

Abstract

We propose a generalization of Zhandry's compressed oracle method to random permutations, where an algorithm can query both the permutation and its inverse. We show how to use the resulting oracle simulation to bound the success probability of an algorithm for any predicate on input-output pairs, a key feature of Zhandry's technique that had hitherto resisted attempts at generalization to random permutations. One key technical ingredient is to use strictly monotone factorizations to represent the permutation in the oracle's database. As an application of our framework, we show that the one-round sponge construction is unconditionally preimage resistant in the random permutation model. This proves a conjecture by Unruh.

Metadata

Available format(s)

PDF

Category

Foundations

Publication info

Preprint.

Keywords

Quantum Cryptography; Quantum Random Oracle

Contact author(s)

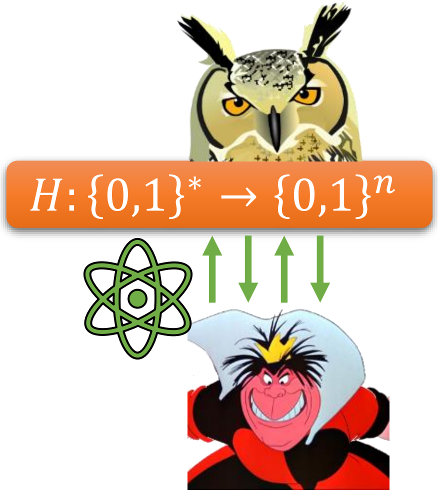
chmaj @ dtu dk
giulio malavolta @ hotmail it
michael walter @ rub de

History

2024-07-15: approved

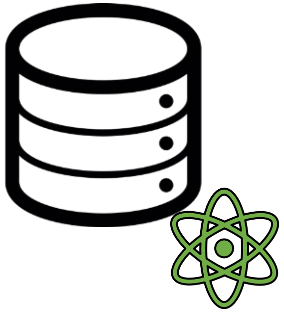
[Czajkowski Majenz Schaffner Zur 19, Unruh 21, Unruh 24, Majenz Malavolta Walter 24]

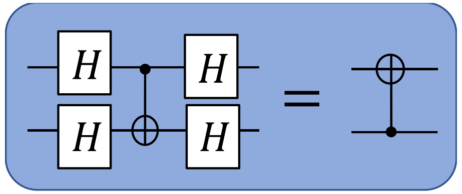
Summary



- Classical Random-Oracle Model
- Quantum Access
- Three Tools:
  - t-wise independence
  - One-way to Hiding (O2H)
  - Compressed oracles
- Extensions & Applications

$$|x\rangle|\eta\rangle|0^n \cdots 0^n\rangle_F \xrightarrow{FO} |x\rangle|\eta\rangle|0^n \cdots 0^n\ \eta\ 0^n \cdots 0^n\rangle_F$$





Thank you for your attention!

Questions

wants you!

QUANTUM
COMPUTER
SCIENCE

More Resources

- detailed links and references at bottom of slides
- 2024 QSI Spring school on PQC: https://pqc-spring-school.nl/
- 2022 IPAM summer school: https://www.ipam.ucla.edu/programs/summer-schools/graduate-summer-school-on-post-quantum-and-quantum-cryptography
  Dominique Unruh on quantum tools
- 2021: Quantum Techniques for Provable Security: https://quiques.huelsing.net/
  Kai-Min Chung on compressed oracles, Kathrin Hövelmanns on O2H lemmas, …
- 2021: 11th BIU Winter School on `Cryptography in a Quantum World’:
  https://www.youtube.com/playlist?list=PL8Vt-7cSFnw2JZsskO0bzeO7FswokQC7-
  Mark Zhandry on compressed oracles
- 2020: Simons institute: https://www.youtube.com/watch?v=LOtxqBJ6Qqk
  Christian Majenz on attacking hash functions