Computational Security for Quantum Protocols: How Classical Parties can Obtain a Secure Access in the Quantum Internet

Petros Wallden

May 8th, 2019

Lattice Coding & Crypto Meetings, Royal Holloway, London

Outline

Based on joint works with:
A. Cojocaru, **L. Colisson** and **E. Kashefi** arXiv:1802.08759 (QCrypt2018) and arXiv:1904.06303

Outline

Based on joint works with:
A. Cojocaru, **L. Colisson** and **E. Kashefi** arXiv:1802.08759 (QCrypt2018) and arXiv:1904.06303

1. Enhancing Cryptography with Quantum Technologies
2. Basics of Quantum Computing

Outline

Based on joint works with:
A. Cojocaru, **L. Colisson** and **E. Kashefi** arXiv:1802.08759 (QCrypt2018) and arXiv:1904.06303

1. Enhancing Cryptography with Quantum Technologies
2. Basics of Quantum Computing
3. QFactory: Replacing the quantum communication

Outline

Based on joint works with:
A. Cojocaru, **L. Colisson** and **E. Kashefi** arXiv:1802.08759 (QCrypt2018) and arXiv:1904.06303

1. Enhancing Cryptography with Quantum Technologies
2. Basics of Quantum Computing
3. QFactory: Replacing the quantum communication
4. Major Applications

Outline

Based on joint works with:
A. Cojocaru, **L. Colisson** and **E. Kashefi** arXiv:1802.08759 (QCrypt2018) and arXiv:1904.06303

1. Enhancing Cryptography with Quantum Technologies
2. Basics of Quantum Computing
3. QFactory: Replacing the quantum communication
4. Major Applications
5. The honest-but-curious QFactory (8-states)
6. Towards the malicious case

The dawn of the Quantum Technologies Era

The dawn of the Quantum Technologies Era

The Second Quantum Revolution

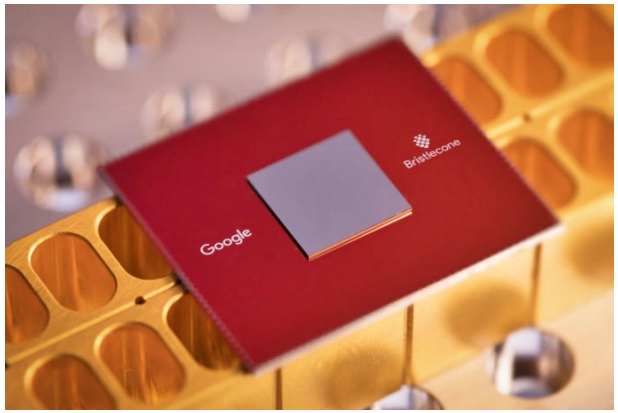
Progress in controlling quantum systems has lead to a new Era in which quantum theory is used to develop **Quantum Technologies**.

The dawn of the Quantum Technologies Era

The Second Quantum Revolution

Progress in controlling quantum systems has lead to a new Era in which quantum theory is used to develop **Quantum Technologies**.

- Google 72 qubit processor: *Bristlecone* (3/2018)
  Impossible to simulate with classical supercomputers

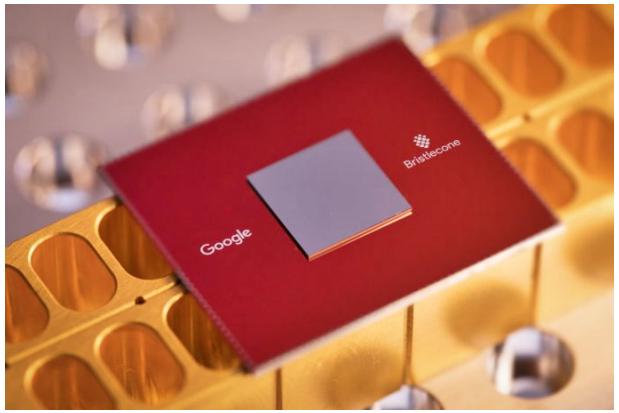


The dawn of the Quantum Technologies Era

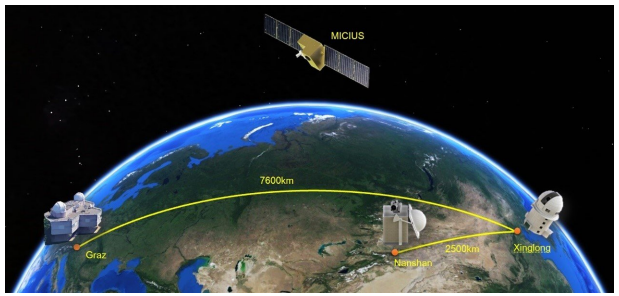
The Second Quantum Revolution

Progress in controlling quantum systems has lead to a new Era in which quantum theory is used to develop **Quantum Technologies**.

- Google 72 qubit processor: *Bristlecone* (3/2018)
  Impossible to simulate with classical supercomputers



- Intercontinental (7600km) Teleconference using QKD (1/2018)
  Austrian and Chinese Academies of Science



Quantum Technologies Initiatives

Map of major £ multimillion, national QTech programs
Major companies with QTech labs & Some major QTech start-ups

Tencent 腾讯

£270m
EPSRC

rigetti

Microsoft

intel

IBM

Google

D:wave
The Quantum Computing Company™

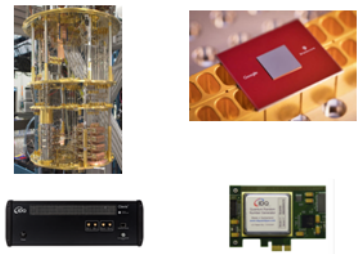
Baidu 百度

Atos

Alibaba Group
阿里巴巴集团

The Future Communication & Computation Networks

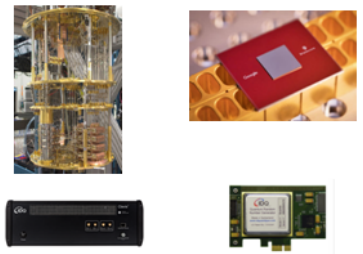
Classical Devices



Quantum Devices

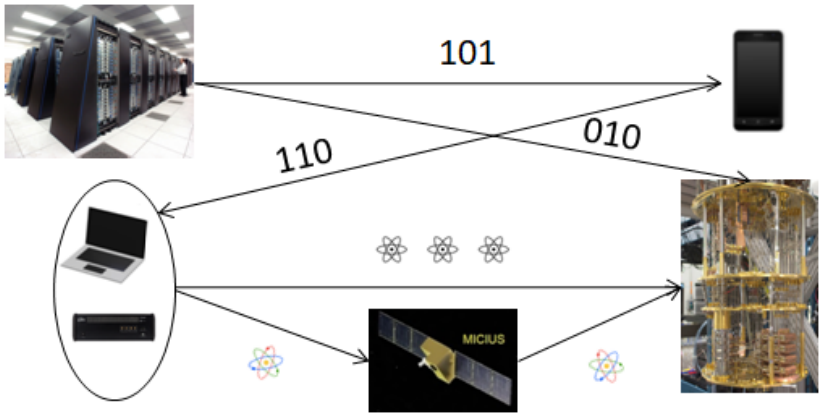
The Future Communication & Computation Networks

Classical Devices



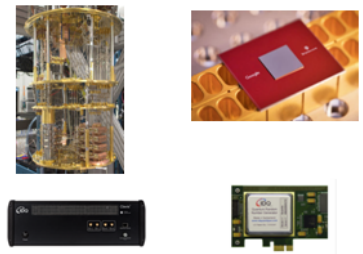
Quantum Devices

Classical & Quantum
Comms Network



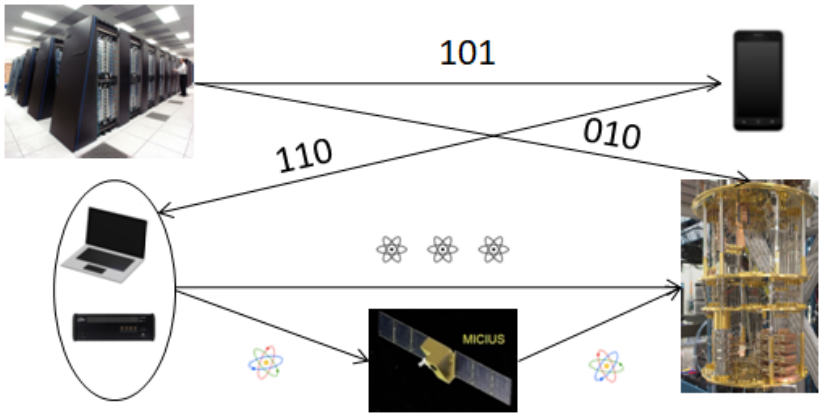
The Future Communication & Computation Networks

Classical Devices



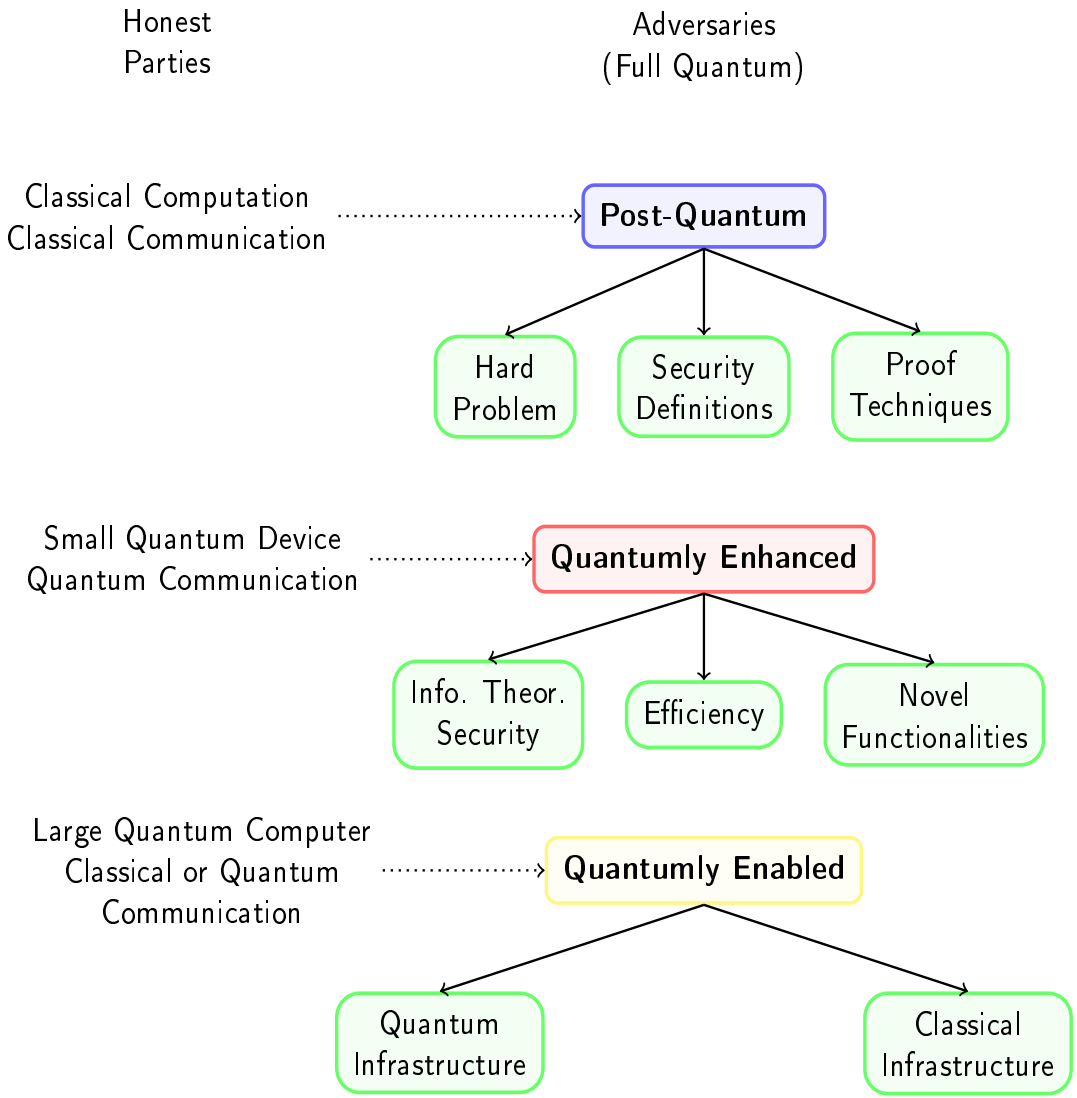
Quantum Devices

Classical & Quantum
Comms Network

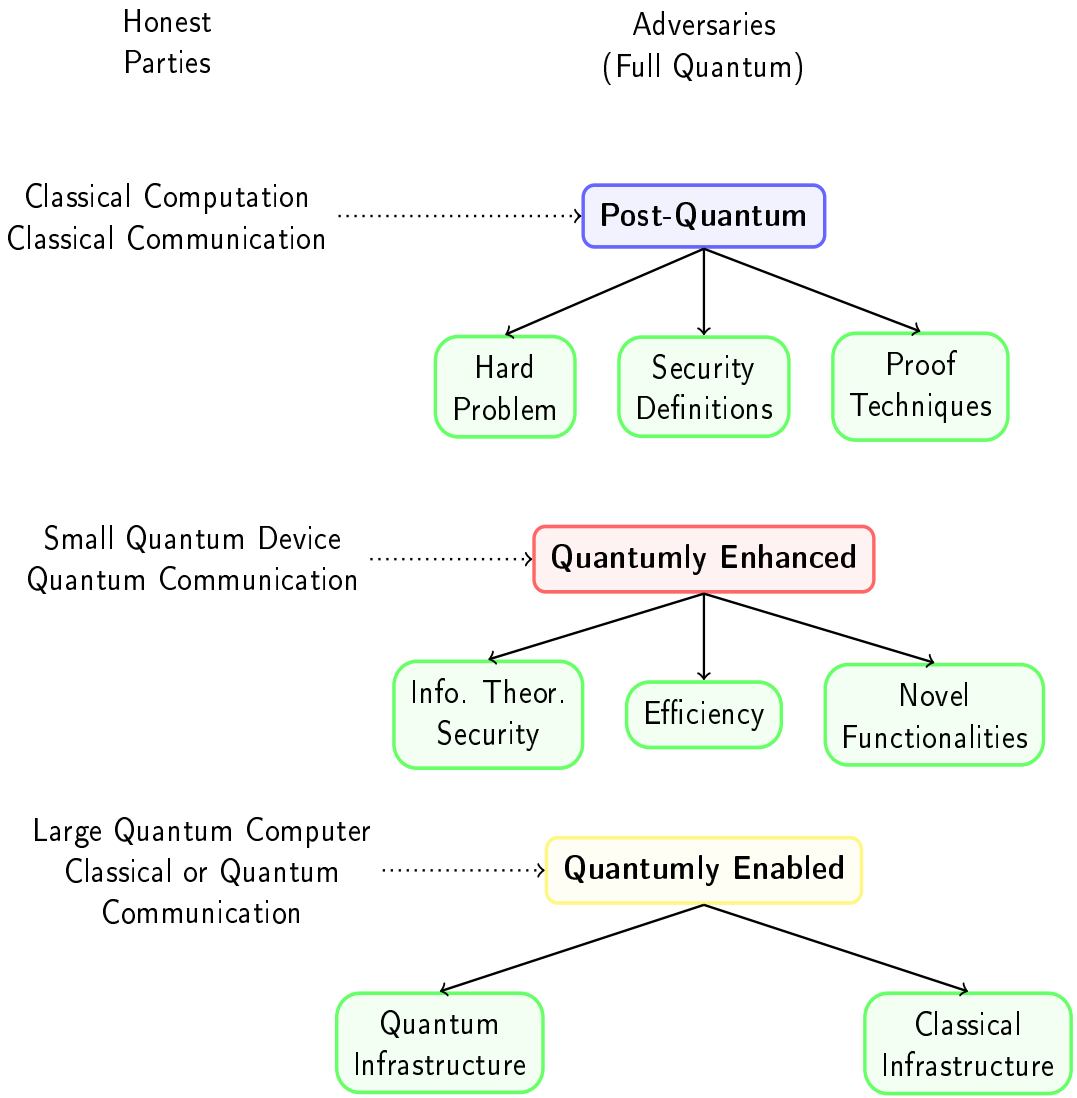


Security of Hybrid Comms & Computation Network

Quantum Cyber Security Landscape: Three Categories



Quantum Cyber Security Landscape: Three Categories



See "Cyber Security in the Quantum Era", April 2019, CACM

Quantum Enhancements

Quantum Internet vision: A network that can achieve many functionalities with q-improvements (blue doesn't require info theoretic security)

quantum key distribution (expansion), quantum randomness (expansion, amplification, certification), quantum fingerprinting, quantum digital "signatures" (secret, public key), quantum coin flipping, e-voting, Byzantine agreement, quantum secret sharing, quantum money (private and public key), quantum private information retrieval, secure multiparty (classical or quantum) computation (SMPC), position verification, fully homomorphic quantum encryption, etc

- Qubit-strings $|110\rangle$ are (unit) vectors with complex coefficients e.g.
  $|\psi\rangle = \frac{1}{\sqrt{2}}(|101\rangle + i\,|011\rangle)$

Basics of Quantum Computing

- Qubit-strings $|110\rangle$ are (unit) vectors with complex coefficients
  e.g.
  $|\psi\rangle = \frac{1}{\sqrt{2}}(|101\rangle + i\,|011\rangle)$
- Operations (Gates) are linear maps $H\,|x\rangle = \frac{1}{\sqrt{2}} \sum_y (-1)^{xy}\,|y\rangle$

- Qubit-strings $|110\rangle$ are (unit) vectors with complex coefficients e.g.
  $|\psi\rangle = \frac{1}{\sqrt{2}}(|101\rangle + i\,|011\rangle)$
- Operations (Gates) are linear maps $H\,|x\rangle = \frac{1}{\sqrt{2}} \sum_y (-1)^{xy}\,|y\rangle$
- Measurements are probabilistic: If $|\psi\rangle = \sum a_x\,|x\rangle$, then x occurs with probability $|a_x|^2$

- Qubit-strings $|110\rangle$ are (unit) vectors with complex coefficients e.g.
  $|\psi\rangle = \frac{1}{\sqrt{2}}(|101\rangle + i\,|011\rangle)$
- Operations (Gates) are linear maps $H\,|x\rangle = \frac{1}{\sqrt{2}} \sum_y (-1)^{xy}\,|y\rangle$
- Measurements are probabilistic: If $|\psi\rangle = \sum a_x\,|x\rangle$, then x occurs with probability $|a_x|^2$
- Multi-qubit operations can generate entanglement (CNOT)

Basics of Quantum Computing

- Qubit-strings $|110\rangle$ are (unit) vectors with complex coefficients e.g.
  $|\psi\rangle = \frac{1}{\sqrt{2}}(|101\rangle + i\,|011\rangle)$
- Operations (Gates) are linear maps $H\,|x\rangle = \frac{1}{\sqrt{2}} \sum_y (-1)^{xy}\,|y\rangle$
- Measurements are probabilistic: If $|\psi\rangle = \sum a_x\,|x\rangle$, then x occurs with probability $|a_x|^2$
- Multi-qubit operations can generate entanglement (CNOT)
- Why we have speed-up? (complex "probabilities")

Basics of Quantum Computing

- Qubit-strings $|110\rangle$ are (unit) vectors with complex coefficients e.g.
  $|\psi\rangle = \frac{1}{\sqrt{2}}(|101\rangle + i\,|011\rangle)$
- Operations (Gates) are linear maps $H\,|x\rangle = \frac{1}{\sqrt{2}} \sum_y (-1)^{xy}\,|y\rangle$
- Measurements are probabilistic: If $|\psi\rangle = \sum a_x\,|x\rangle$, then x occurs with probability $|a_x|^2$
- Multi-qubit operations can generate entanglement (CNOT)
- Why we have speed-up? (complex "probabilities")
- Gates need to be reversible (e.g. OR, AND)

QFactory: Replacing the quantum channel

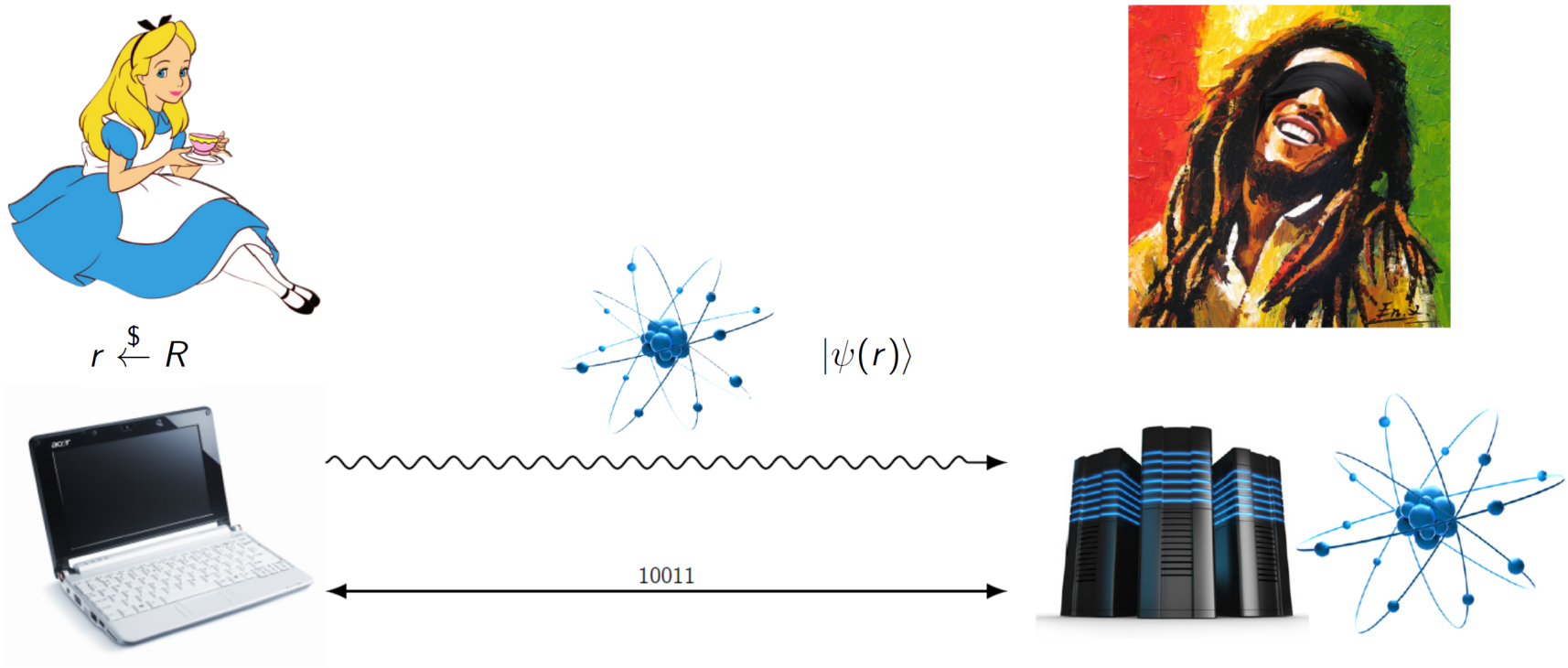


Figure: QFactory gadget: simulate quantum channel

QFactory: Replacing the quantum channel

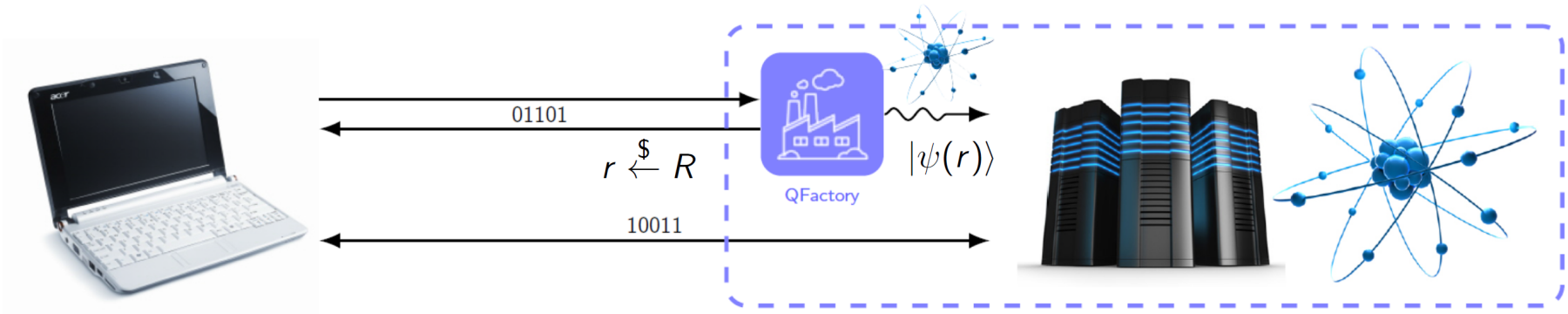
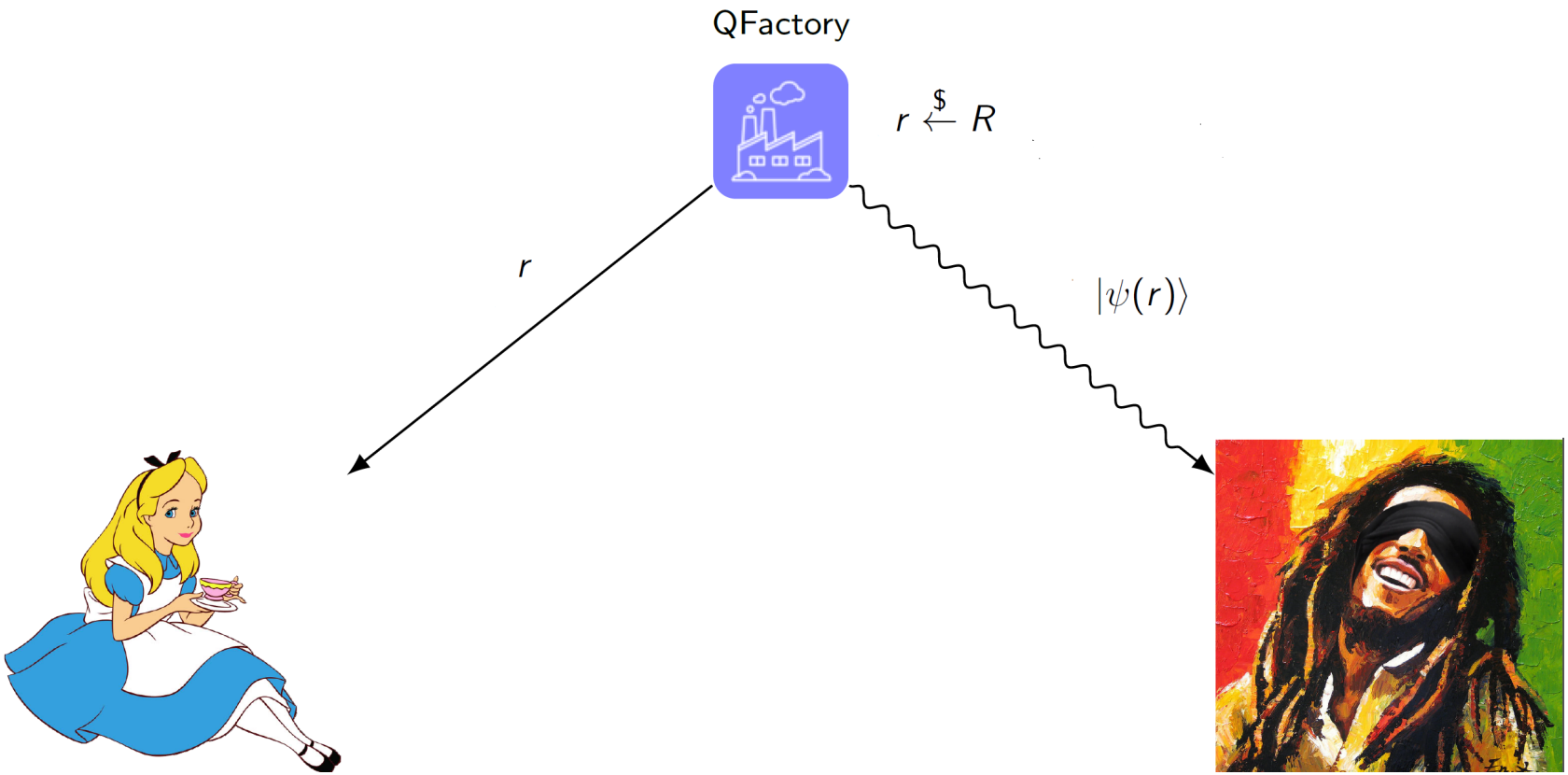


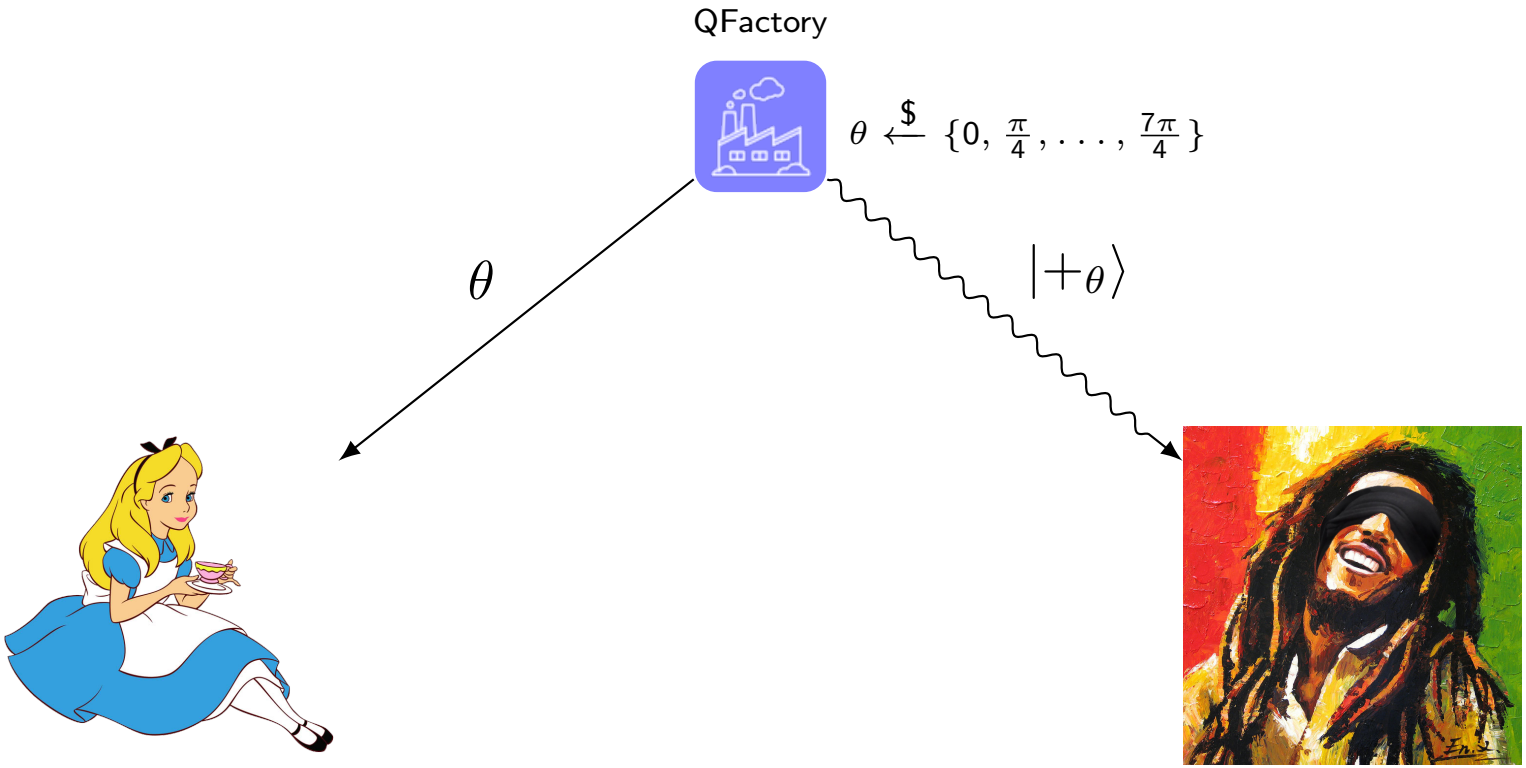
Figure: QFactory gadget: simulate quantum channel

QFactory: Ideal Functionality



QFactory: Ideal Functionality (General)

QFactory: Ideal Functionality



QFactory: Ideal Functionality

$$|+_\theta\rangle = \frac{1}{\sqrt{2}}(|0\rangle + e^{i\theta}|1\rangle)$$

QFactory: Ideal Functionality (Variations)

1. Weaker version, where Server could obtain some partial information about r, θ that is fixed by some leakage function (see Alex's talk)
   This weaker form **is sufficient for applications** and possible to achieve against **malicious adversaries**

QFactory: Ideal Functionality (Variations)

1. Weaker version, where Server could obtain some partial information about r, θ that is fixed by some leakage function (see Alex's talk)
   This weaker form **is sufficient for applications** and possible to achieve against **malicious adversaries**
2. Different sets of states:
   $\{|0\rangle, |1\rangle, |+\rangle, |-\rangle\}$ Vs $\{|+_\theta\rangle\} \mid \theta \in \{0, \pi/4, \cdots, 7\pi/8\}$

QFactory: Ideal Functionality (Variations)

1. Weaker version, where Server could obtain some partial information about r, θ that is fixed by some leakage function (see Alex's talk)
   This weaker form **is sufficient for applications** and possible to achieve against **malicious adversaries**
2. Different sets of states:
   $\{|0\rangle, |1\rangle, |+\rangle, |-\rangle\}$ Vs $\{|+_\theta\rangle\} \mid \theta \in \{0, \pi/4, \cdots, 7\pi/8\}$
3. Verifiable QFactory (see Alex's talk):
   Prove to Client that the desired state was really generated on Server's side (or abort)

QFactory Intuition

- **All quantum operations** (QFactory) are performed by Bob

QFactory Intuition

- **All quantum operations** (QFactory) are performed by Bob

  Question: How can Alice make Bob prepare some state of which he cannot guess the classical description?

  Answer: Alice instructs Bob to perform a hard computation:

  1. he cannot classically simulate (thus doesn't know the outcome)
  2. he cannot reproduce (measurements involved), and thus cannot do tomography or learn partial info

QFactory Intuition

- **All quantum operations** (QFactory) are performed by Bob

  Question: How can Alice make Bob prepare some state of which he cannot guess the classical description?

  Answer: Alice instructs Bob to perform a hard computation:

  1. he cannot classically simulate (thus doesn't know the outcome)
  2. he cannot reproduce (measurements involved), and thus cannot do tomography or learn partial info

- Alice uses crypto information (trapdoor) and can compute efficiently the classical description of the output q-state

Output q-state: random, unknown to Bob, known to Alice

A nice trick

How to produce non-reproducible quantum states:

Let U_f be a unitary corresponding to the function f such that: $U_f |x\rangle |a\rangle = |x\rangle |f(x) \oplus a\rangle$ (set $a = 0$)

A nice trick

How to produce non-reproducible quantum states:

Let U_f be a unitary corresponding to the function f such that: $U_f |x\rangle |a\rangle = |x\rangle |f(x) \oplus a\rangle$ (set $a = 0$)

By linearity: $U_f \sum_x c_x |x\rangle |0\rangle = \sum_x c_x |x\rangle |f(x)\rangle$, where c_x the amplitude of each term (can choose $c_n = \frac{1}{\sqrt{2^n}}$)

A nice trick

How to produce non-reproducible quantum states:

Let U_f be a unitary corresponding to the function f such that: $U_f |x\rangle |a\rangle = |x\rangle |f(x) \oplus a\rangle$ (set $a = 0$)

By linearity: $U_f \sum_x c_x |x\rangle |0\rangle = \sum_x c_x |x\rangle |f(x)\rangle$, where c_x the amplitude of each term (can choose $c_n = \frac{1}{\sqrt{2^n}}$)

Measuring the second register (range) we obtain some y: $\sum_{x|f^{-1}(y)=x} |x\rangle |y\rangle$ (where we need to normalise the state)

A nice trick

How to produce non-reproducible quantum states:

Let U_f be a unitary corresponding to the function f such that: $U_f |x\rangle |a\rangle = |x\rangle |f(x) \oplus a\rangle$ (set $a = 0$)

By linearity: $U_f \sum_x c_x |x\rangle |0\rangle = \sum_x c_x |x\rangle |f(x)\rangle$, where c_x the amplitude of each term (can choose $c_n = \frac{1}{\sqrt{2^n}}$)

Measuring the second register (range) we obtain some y: $\sum_{x|f^{-1}(y)=x} |x\rangle |y\rangle$ (where we need to normalise the state)

- Probability of each y is negligible $O(1/2^n)$ (cannot reproduce)
- If the function is one-way, cannot infer from y the preimages, i.e. the terms in the superposition of the first register

A nice trick

How to produce non-reproducible quantum states:

Let U_f be a unitary corresponding to the function f such that: $U_f |x\rangle |a\rangle = |x\rangle |f(x) \oplus a\rangle$ (set $a = 0$)

By linearity: $U_f \sum_x c_x |x\rangle |0\rangle = \sum_x c_x |x\rangle |f(x)\rangle$, where c_x the amplitude of each term (can choose $c_n = \frac{1}{\sqrt{2^n}}$)

Measuring the second register (range) we obtain some y: $\sum_{x|f^{-1}(y)=x} |x\rangle |y\rangle$ (where we need to normalise the state)

- Probability of each y is negligible $O(1/2^n)$ (cannot reproduce)
- If the function is one-way, cannot infer from y the preimages, i.e. the terms in the superposition of the first register
- This blindness (ignorance) can be used:
  1. If f is 2-regular the first register is: $|x_1\rangle + |x_2\rangle$, where $f(x_1) = f(x_2) = y$
  2. If f is trapdoor, the client can know the state

Cryptographic Assumptions on Functions Used

One-way

This function is hard to invert...

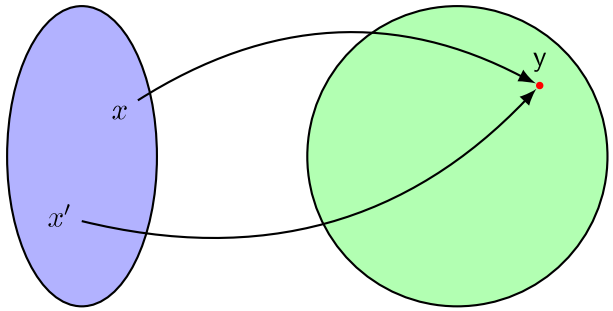
2-Regular

2 preimages for all elements in $Im(f_k)$

Functions $\{f_k\}$

Trapdoor

...except if you have the trapdoor t_k associated to the function index k.



Collision resistant

Without trapdoor t_k, hard to find $x \neq x'$ such that $f_k(x) = f_k(x')$

Why Classical Client?

Easier to perform:

- Light devices (e.g. mobile phones) can participate
- No extended quantum network infrastructure required
- Can test quantum device/computation with no special process, from any classical device

Why Classical Client?

Easier to perform:

- Light devices (e.g. mobile phones) can participate
- No extended quantum network infrastructure required
- Can test quantum device/computation with no special process, from any classical device

Fundamental:

- Incompatibility of current best QComm systems (Photons) and best Quantum Computing systems (Superconducting)
- Can test devices remotely with no trust required on any quantum device

Why Classical Client?

Easier to perform:

- Light devices (e.g. mobile phones) can participate
- No extended quantum network infrastructure required
- Can test quantum device/computation with no special process, from any classical device

Fundamental:

- Incompatibility of current best QComm systems (Photons) and best Quantum Computing systems (Superconducting)
- Can test devices remotely with no trust required on any quantum device

Replace Quantum Channel: QFactory

Primitive where **Classical Client** interacts with **Quantum Server**

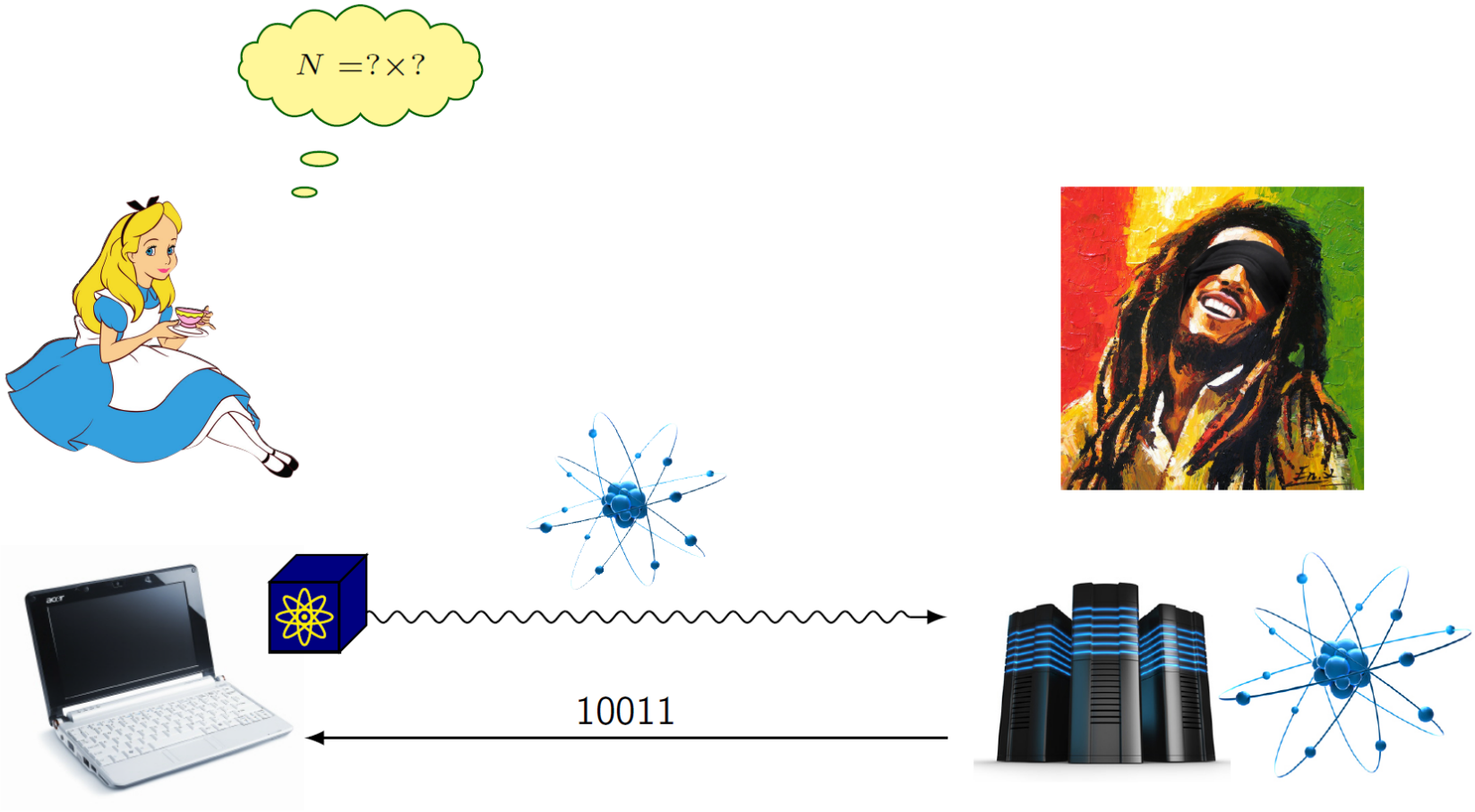
Major Application I: Blind Quantum Computation

11010

10011

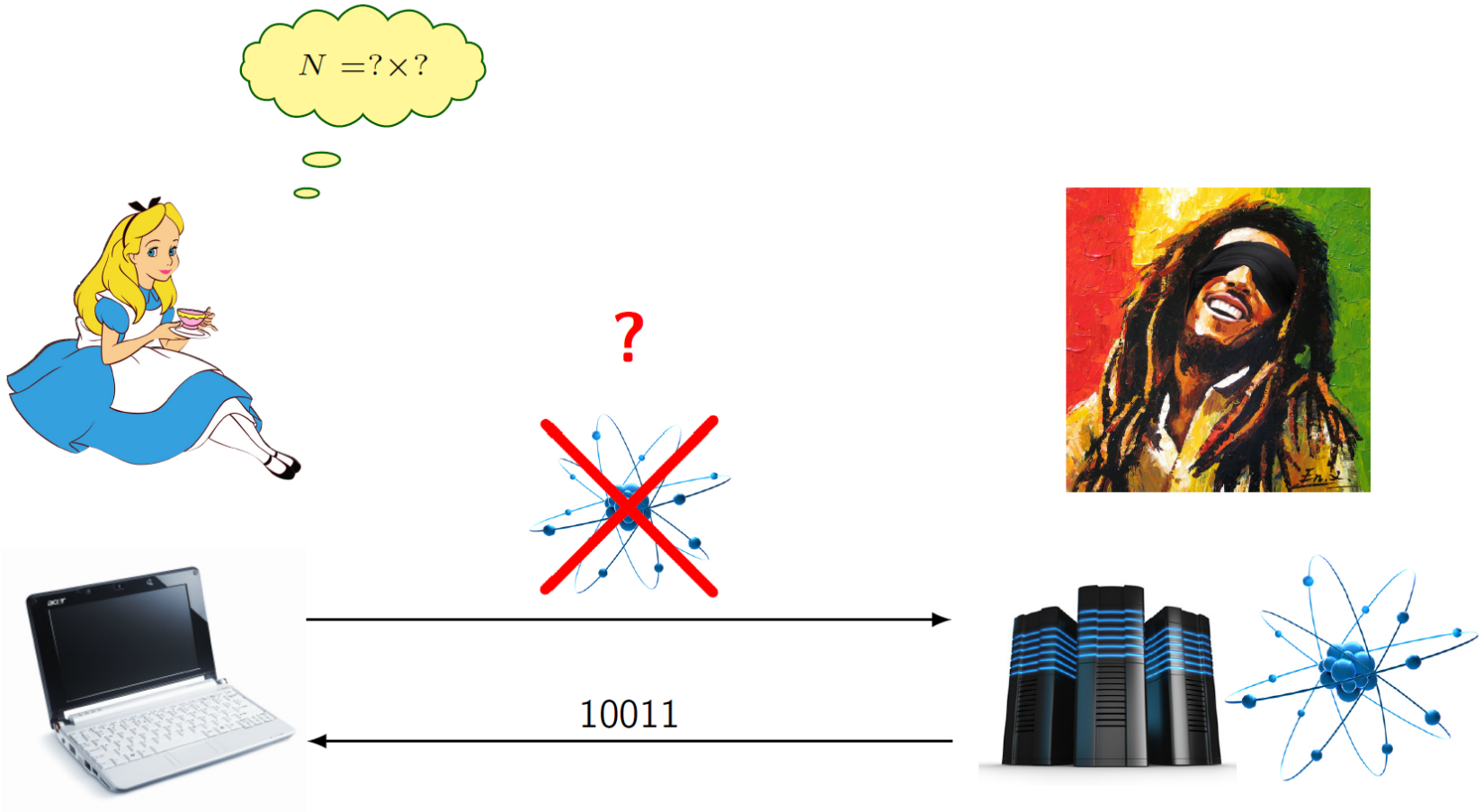
Delegated Quantum Computation

Major Application I: Blind Quantum Computation



Delegated **Blind** Quantum Computation

Major Application I: Blind Quantum Computation



Delegated **Classical Client** Blind Quantum Computation

Major Application I: Blind Quantum Computation

Modern Cyber Security:

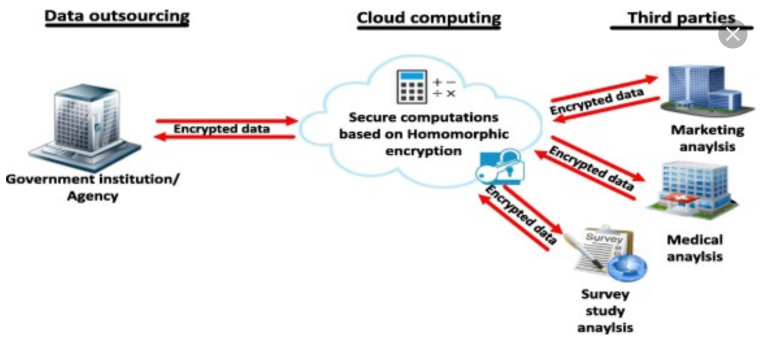
Use **Efficiency** of Cloud Computing with Security Guarantees

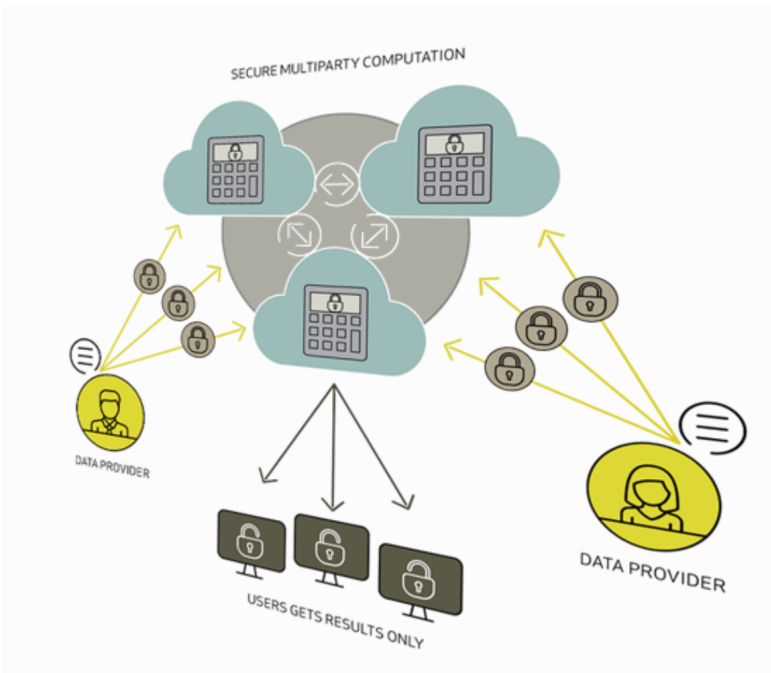
Major Application I: Blind Quantum Computation

Modern Cyber Security:

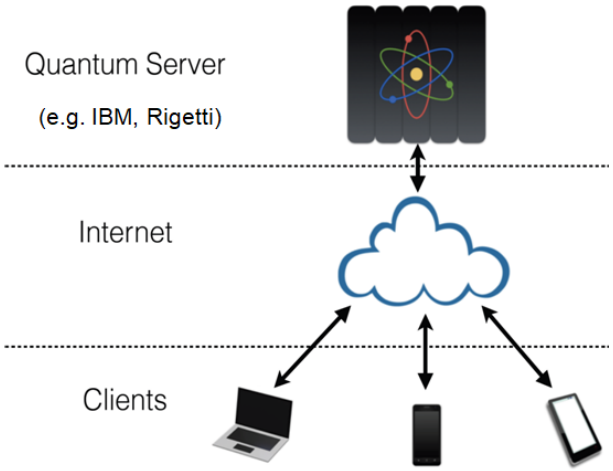
Use **Efficiency** of Cloud Computing with Security Guarantees

Examples: Privacy-preserving data mining, medical records, e-voting, auctions

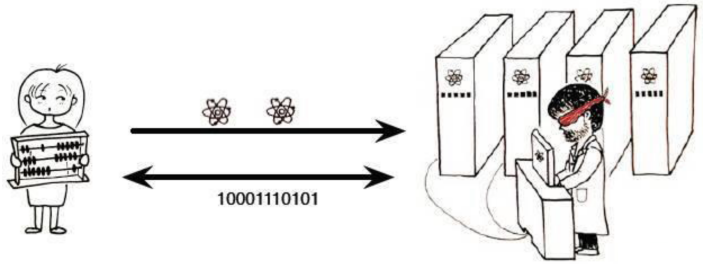




Major Application I: Blind Quantum Computation



- Clients wants to maintain **privacy**, **accuracy** and **reliability**
- Clients wants to use the **extra power of quantum computing**



- Universal Blind Quantum Computation (Broadbent, Fitzsimons, Kashefi 2009)
- Basis for numerous extra functionalities
- Client sends **random single qubits** to Server

Major Application II: Verification of Quantum Computation

Quantum Verification Question

How can we test a quantum computer without having one?

Major Application II: Verification of Quantum Computation

Quantum Verification Question

How can we test a quantum computer without having one?

- Theoretical interest
- Practical interest (commercial applications)

Major Application II: Verification of Quantum Computation

Quantum Verification Question

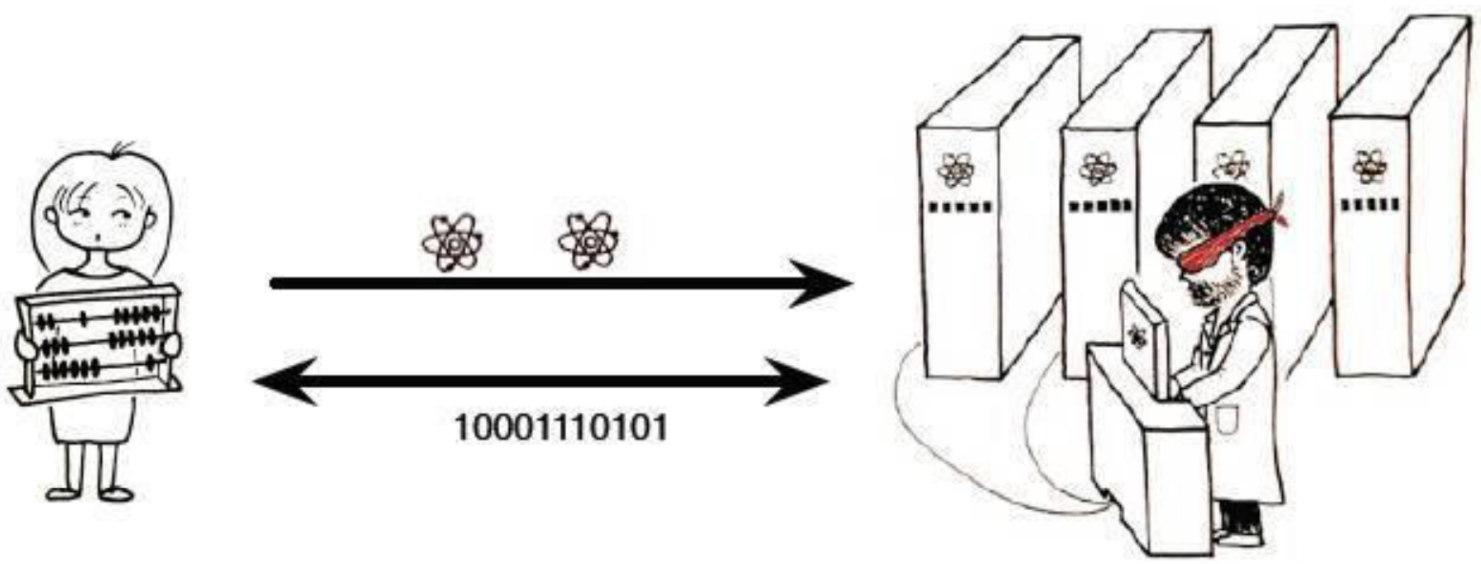
How can we test a quantum computer without having one?

- Theoretical interest
- Practical interest (commercial applications)

Previously Necessary

Either quantum communication and **quantum verifiers** or untrusted **multiple non-communicating quantum devices**

Solution based on Blind QC (Fitzsimons & Kashefi 2012)



Verifiable Quantum Computation

Our Protocol

Our Protocol

t_k, k

Our Protocol

t_k, k

$(\alpha_i \xleftarrow{\$} \{0, \frac{\pi}{4} \dots \frac{7\pi}{4}\})_{i=1}^{n-1}$

Our Protocol

t_k, k

$(\alpha_i \xleftarrow{\$} \{0, \frac{\pi}{4} \dots \frac{7\pi}{4}\})_{i=1}^{n-1}$

$k, (\alpha_i)$ $\longrightarrow$

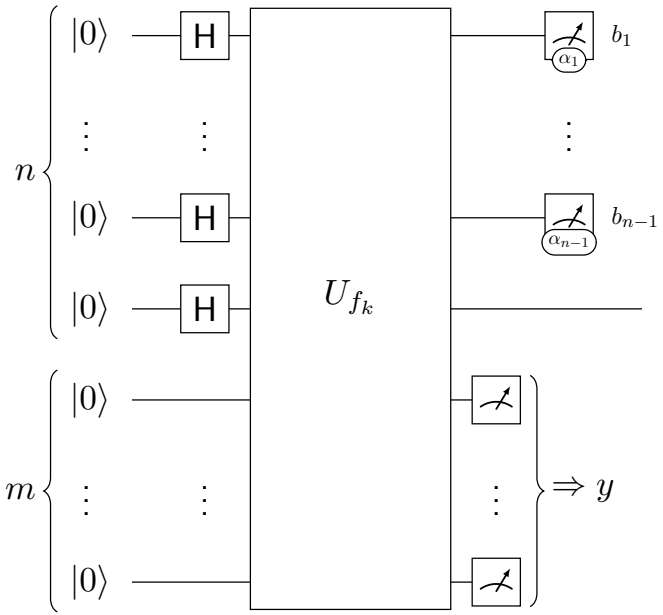
Our Protocol

t_k, k

$(\alpha_i \xleftarrow{\$} \{0, \frac{\pi}{4} \dots \frac{7\pi}{4}\})_{i=1}^{n-1}$

$k, (\alpha_i)$

Compute circuit



Our Protocol

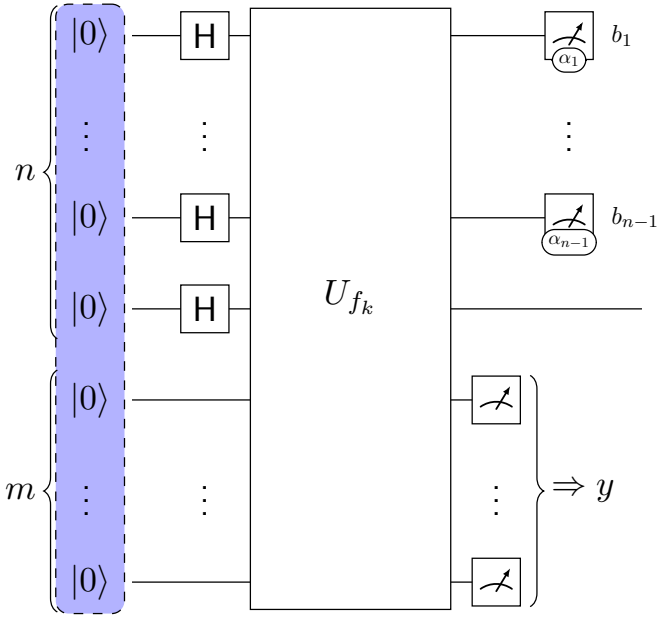
$|0\rangle^{\otimes n}|0\rangle^{\otimes m}$

t_k, k

$(\alpha_i \xleftarrow{\$} \{0, \frac{\pi}{4} \dots \frac{7\pi}{4}\})_{i=1}^{n-1}$

$k, (\alpha_i)$

Compute circuit



Our Protocol

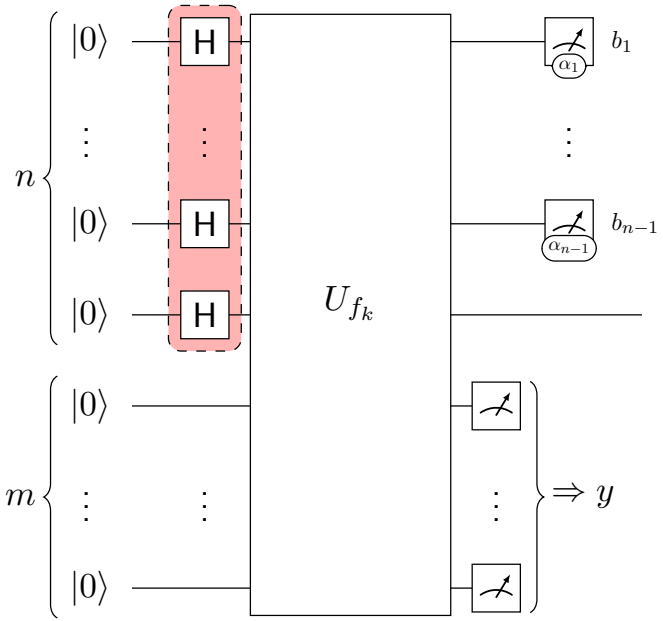
$|0\rangle^{\otimes n}|0\rangle^{\otimes m} \Rightarrow \sum_x |x\rangle|0\rangle^{\otimes m}$

t_k, k

$(\alpha_i \xleftarrow{\$} \{0, \frac{\pi}{4} \dots \frac{7\pi}{4}\})_{i=1}^{n-1}$

$k, (\alpha_i)$

Compute circuit



Our Protocol

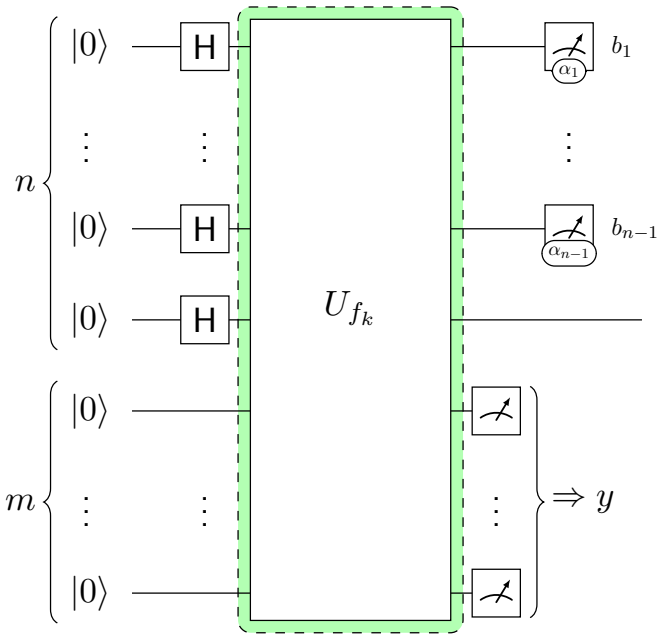
$|0\rangle^{\otimes n}|0\rangle^{\otimes m} \Rightarrow \sum_x |x\rangle|0\rangle^{\otimes m} \Rightarrow \sum_x |x\rangle|f_k(x)\rangle$

t_k, k

$(\alpha_i \xleftarrow{\$} \{0, \frac{\pi}{4} \dots \frac{7\pi}{4}\})_{i=1}^{n-1}$

$k, (\alpha_i)$

Compute circuit



Our Protocol

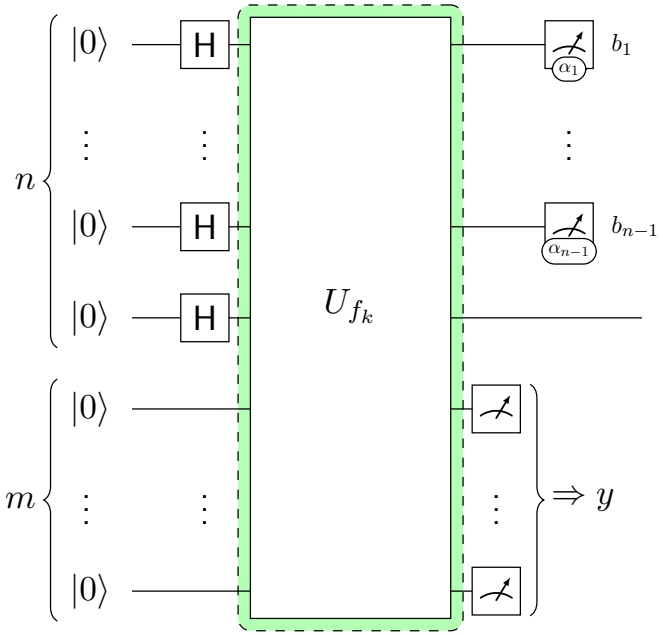
$|0\rangle^{\otimes n}|0\rangle^{\otimes m} \Rightarrow \sum_x |x\rangle|0\rangle^{\otimes m} \Rightarrow \sum_x |x\rangle|f_k(x)\rangle = \sum_y (|x\rangle + |x'\rangle) \otimes |y\rangle$

t_k, k

$(\alpha_i \xleftarrow{\$} \{0, \frac{\pi}{4} \dots \frac{7\pi}{4}\})_{i=1}^{n-1}$

$k, (\alpha_i)$

Compute circuit



Our Protocol

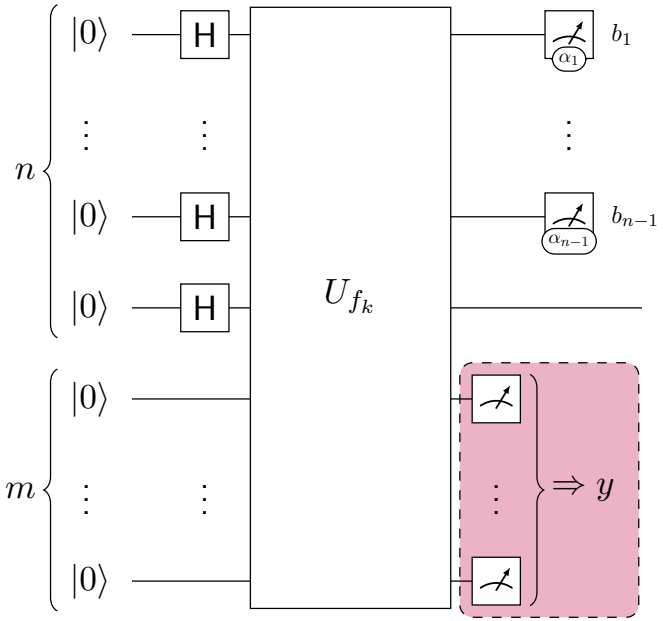
$|0\rangle^{\otimes n}|0\rangle^{\otimes m} \Rightarrow \sum_x |x\rangle|0\rangle^{\otimes m} \Rightarrow \sum_x |x\rangle|f_k(x)\rangle = \sum_y (|x\rangle + |x'\rangle) \otimes |y\rangle \Rightarrow (|x\rangle + |x'\rangle) \otimes |y\rangle$

t_k, k

$(\alpha_i \xleftarrow{\$} \{0, \frac{\pi}{4} \dots \frac{7\pi}{4}\})_{i=1}^{n-1}$

$k, (\alpha_i)$

Compute circuit



Our Protocol

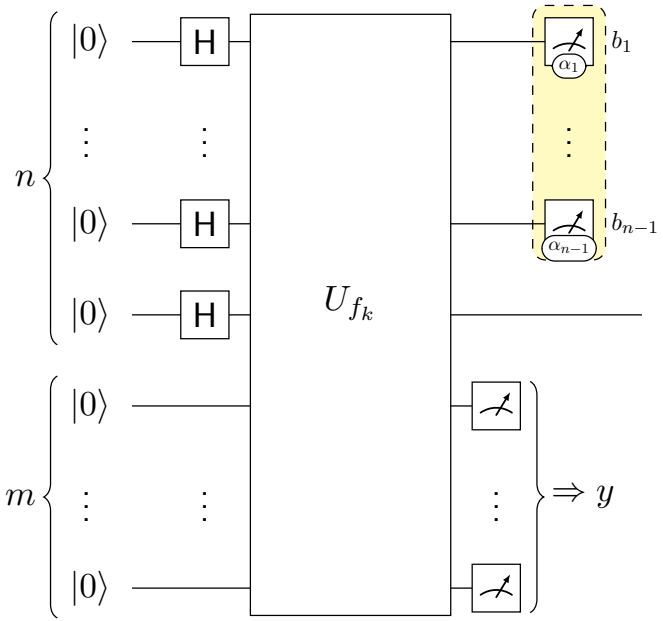
$|0\rangle^{\otimes n}|0\rangle^{\otimes m} \Rightarrow \sum_x |x\rangle|0\rangle^{\otimes m} \Rightarrow \sum_x |x\rangle|f_k(x)\rangle = \sum_y (|x\rangle + |x'\rangle) \otimes |y\rangle \Rightarrow (|x\rangle + |x'\rangle) \otimes |y\rangle \Rightarrow (\bigotimes_i |b_i\rangle) \otimes |+_\theta\rangle \otimes |y\rangle$

t_k, k

$(\alpha_i \xleftarrow{\$} \{0, \frac{\pi}{4} \dots \frac{7\pi}{4}\})_{i=1}^{n-1}$

$k, (\alpha_i)$

Compute circuit



Our Protocol

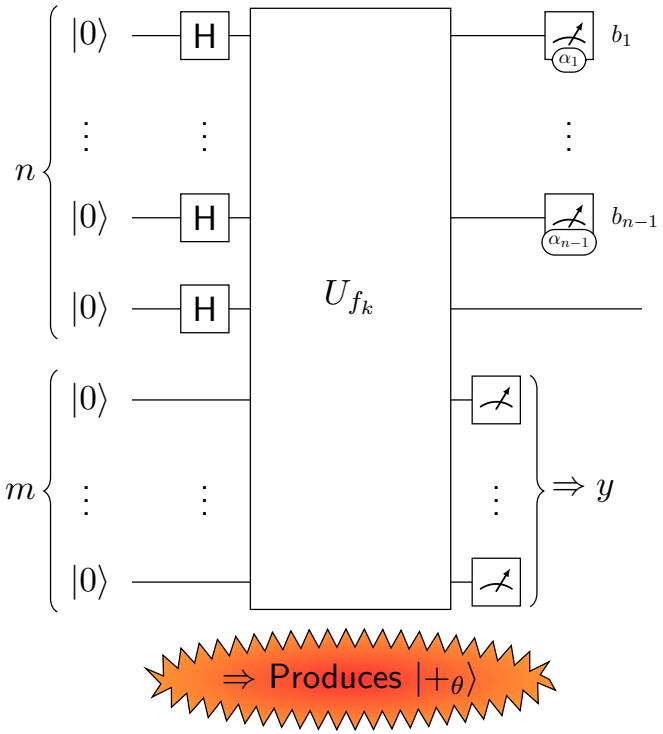
$|0\rangle^{\otimes n}|0\rangle^{\otimes m} \Rightarrow \sum_x |x\rangle|0\rangle^{\otimes m} \Rightarrow \sum_x |x\rangle|f_k(x)\rangle = \sum_y (|x\rangle + |x'\rangle) \otimes |y\rangle \Rightarrow (|x\rangle + |x'\rangle) \otimes |y\rangle \Rightarrow (\bigotimes_i |b_i\rangle) \otimes |+_\theta\rangle \otimes |y\rangle$

t_k, k

$(\alpha_i \xleftarrow{\$} \{0, \frac{\pi}{4} \dots \frac{7\pi}{4}\})_{i=1}^{n-1}$

$k, (\alpha_i)$

Compute circuit



$\Rightarrow$ Produces $|+_\theta\rangle$

Our Protocol

$|0\rangle^{\otimes n}|0\rangle^{\otimes m} \Rightarrow \sum_x |x\rangle|0\rangle^{\otimes m} \Rightarrow \sum_x |x\rangle|f_k(x)\rangle = \sum_y (|x\rangle + |x'\rangle) \otimes |y\rangle \Rightarrow (|x\rangle + |x'\rangle) \otimes |y\rangle \Rightarrow (\bigotimes_i |b_i\rangle) \otimes |+_\theta\rangle \otimes |y\rangle$

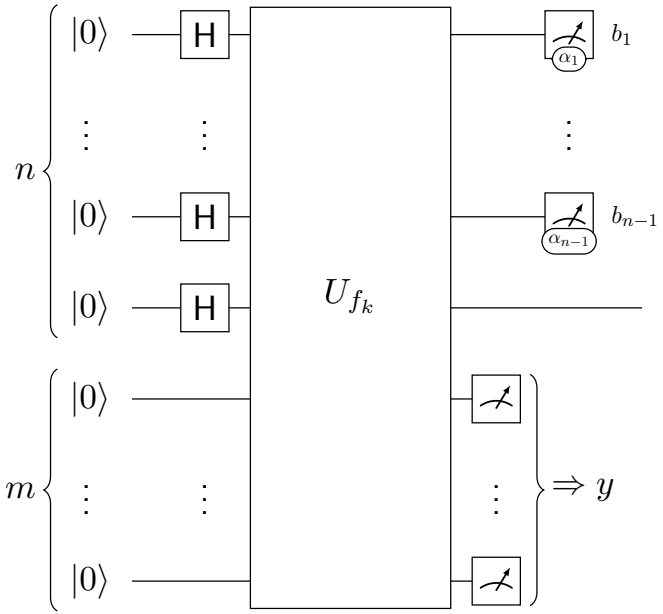
t_k, k

$(\alpha_i \xleftarrow{\$} \{0, \frac{\pi}{4} \dots \frac{7\pi}{4}\})_{i=1}^{n-1}$

$k, (\alpha_i)$ →

Compute circuit

← $y, (b_i)$



$\Rightarrow$ Produces $|+_\theta\rangle$

Our Protocol

$|0\rangle^{\otimes n}|0\rangle^{\otimes m} \Rightarrow \sum_x |x\rangle|0\rangle^{\otimes m} \Rightarrow \sum_x |x\rangle|f_k(x)\rangle = \sum_y (|x\rangle + |x'\rangle) \otimes |y\rangle \Rightarrow (|x\rangle + |x'\rangle) \otimes |y\rangle \Rightarrow (\bigotimes_i |b_i\rangle) \otimes |+_\theta\rangle \otimes |y\rangle$

t_k, k

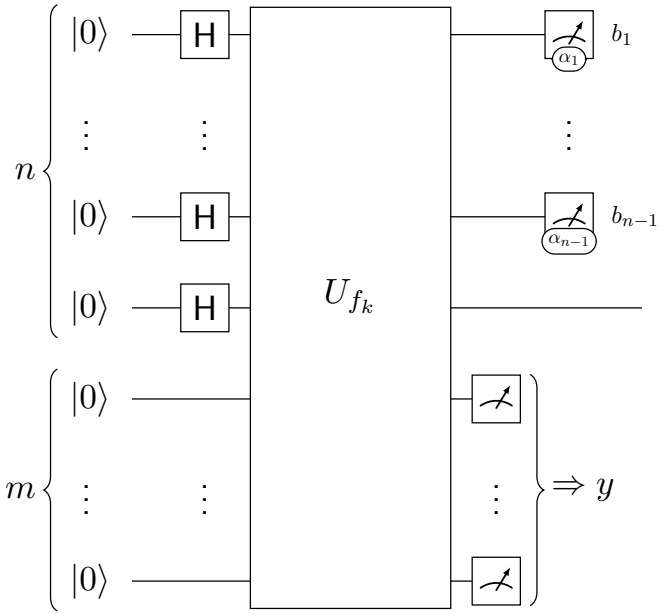
$(\alpha_i \xleftarrow{\$} \{0, \frac{\pi}{4} \dots \frac{7\pi}{4}\})_{i=1}^{n-1}$

$k, (\alpha_i)$ →

Compute circuit

← $y, (b_i)$

$(x, x') := \texttt{Inv}(t_k, y)$



⇒ Produces $|+_\theta\rangle$

Our Protocol

$|0\rangle^{\otimes n}|0\rangle^{\otimes m} \Rightarrow \sum_x |x\rangle|0\rangle^{\otimes m} \Rightarrow \sum_x |x\rangle|f_k(x)\rangle = \sum_y (|x\rangle + |x'\rangle) \otimes |y\rangle \Rightarrow (|x\rangle + |x'\rangle) \otimes |y\rangle \Rightarrow (\bigotimes_i |b_i\rangle) \otimes |+_\theta\rangle \otimes |y\rangle$

t_k, k

$(\alpha_i \xleftarrow{\$} \{0, \frac{\pi}{4} \dots \frac{7\pi}{4}\})_{i=1}^{n-1}$

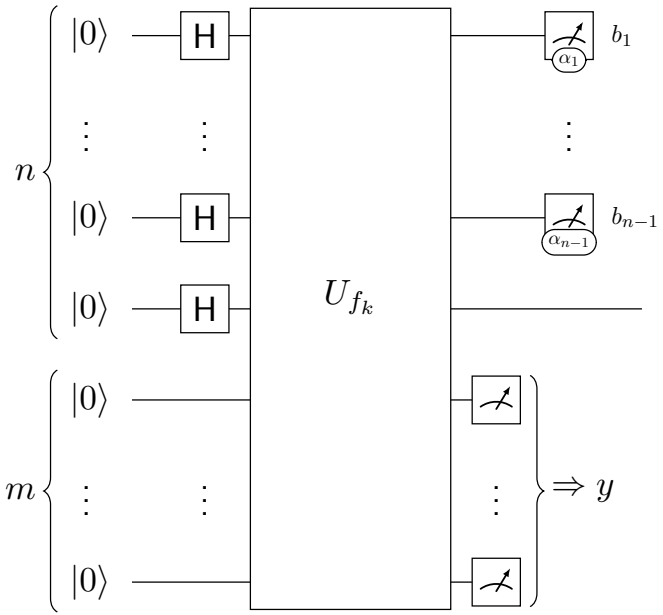
$k, (\alpha_i)$ →

Compute circuit

← $y, (b_i)$

$(x, x') := \texttt{Inv}(t_k, y)$

$\theta := (-1)^{x_n} \sum_{i=1}^{n-1} (x_i - x'_i)(b_i \pi + \alpha_i)$



$\Rightarrow$ Produces $|+_\theta\rangle$

Our Protocol

$|0\rangle^{\otimes n}|0\rangle^{\otimes m} \Rightarrow \sum_x |x\rangle|0\rangle^{\otimes m} \Rightarrow \sum_x |x\rangle|f_k(x)\rangle = \sum_y (|x\rangle + |x'\rangle) \otimes |y\rangle \Rightarrow (|x\rangle + |x'\rangle) \otimes |y\rangle \Rightarrow (\bigotimes_i |b_i\rangle) \otimes |+_\theta\rangle \otimes |y\rangle$

t_k, k

$(\alpha_i \xleftarrow{\$} \{0, \frac{\pi}{4} \dots \frac{7\pi}{4}\})_{i=1}^{n-1}$

$k, (\alpha_i)$ →

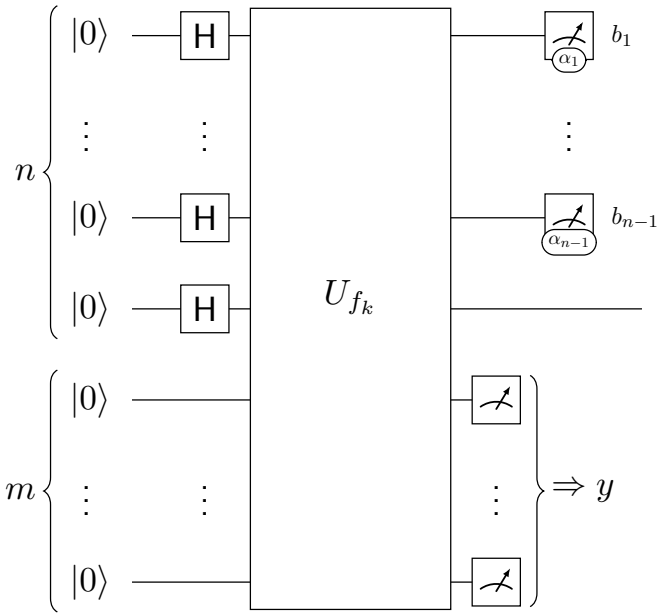
Compute circuit

← $y, (b_i)$

$(x, x') := \texttt{Inv}(t_k, y)$

$\theta := (-1)^{x_n} \sum_{i=1}^{n-1} (x_i - x'_i)(b_i \pi + \alpha_i)$

⇒ Gets θ



⇒ Produces $|+_\theta\rangle$

Level of Security

Information Theoretic: Secure against unbounded adversaries

Computational: Secure against Quantum Adversaries with Polynomially bounded computational resources (QPT).

Security Setting

Level of Security

Information Theoretic: Secure against unbounded adversaries

Computational: Secure against Quantum Adversaries with Polynomially bounded computational resources (QPT).

Types of Adversaries

Honest-but-curious: Follows protocol but keeps records and tries to learn from these

Malicious: Can deviate in any step in any way

Security Setting

Level of Security

Information Theoretic: Secure against unbounded adversaries

Computational: Secure against Quantum Adversaries with Polynomially bounded computational resources (QPT).

Types of Adversaries

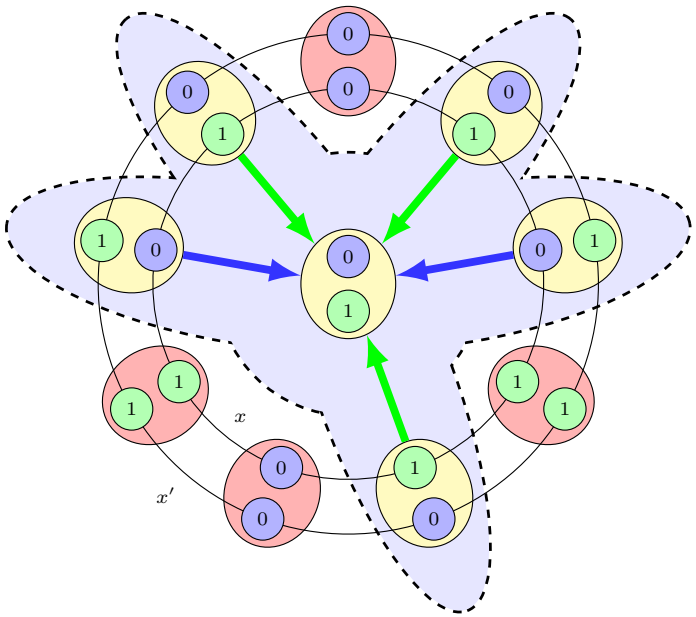
Honest-but-curious: Follows protocol but keeps records and tries to learn from these

Malicious: Can deviate in any step in any way

- Current protocol against honest-but-curious
- Alex will give protocol against malicious adversaries
- Conjectured verifiable QFactory (against malicious)

Security Intuition

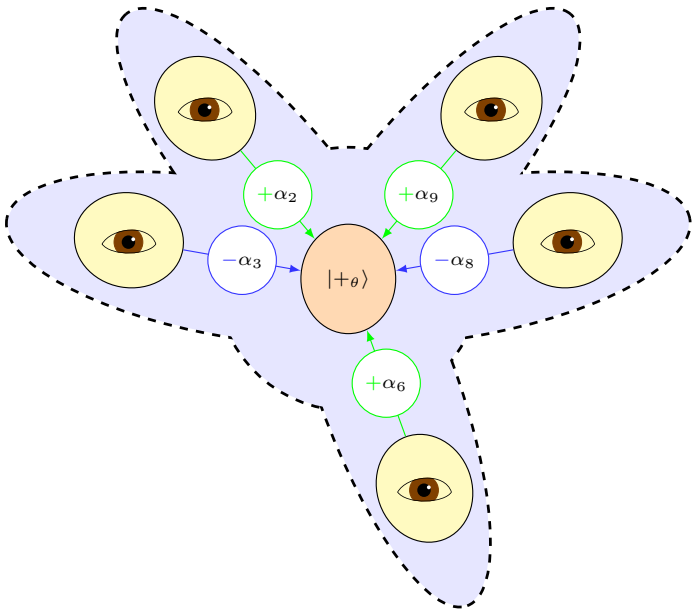
- State $|x\rangle + |x'\rangle$ is a product of GHZ and single qubits
- Connectivity depends on $x \oplus x'$. unknown to server



Security Intuition

Measure all but output qubit

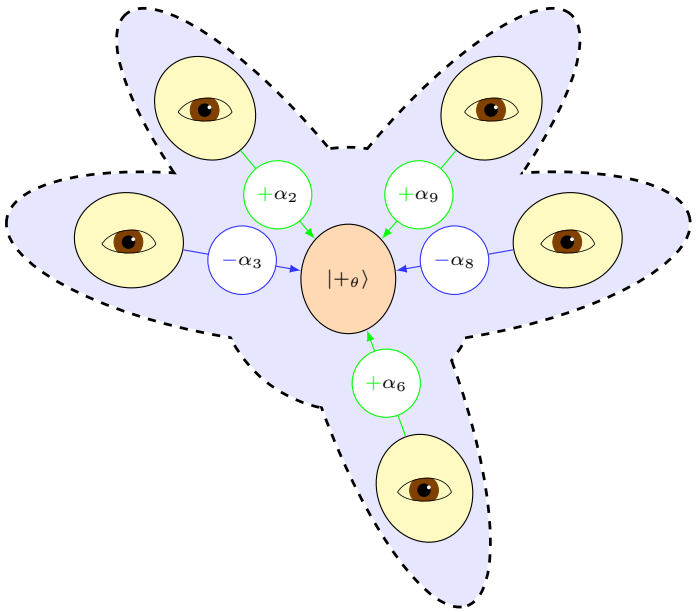
- Entangled qubits **rotate** output
- Non-entangled have **no effect** on output



Security Intuition

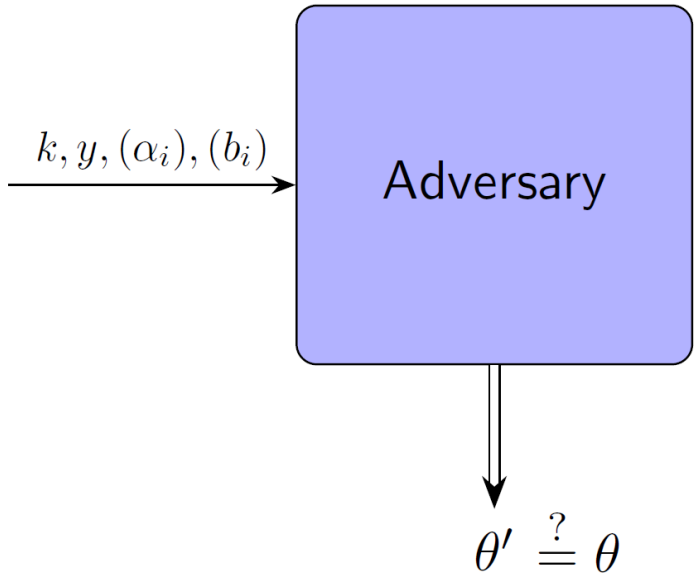
Measure all but output qubit

- Entangled qubits **rotate** output
- Non-entangled have **no effect** on output



Honest-but-curious Setting

- Adversary follows protocol, but uses classical registers



Cannot be better than random guess: θ **hard-core** function.

Intuition of proof

θ is a hardcore function: proof based on Goldreich-Levin Theorem:

Theorem

If f is a one-way function, then the predicate $hc(x, r) = \sum x_i r_i \bmod 2$ cannot be distinguished from a random bit, given r and $f(x)$.

Recall, in our case: $f(x) \approx y$ and

$$\theta \approx \sum \underbrace{(x_i - x_i')}_{\text{Unknown to server}} \underbrace{(4b_i + \alpha_i)}_{\text{Known to server}} \mod 8$$

Towards the malicious case

- **Can ensure:** classical info of an honest run leaks nothing about θ not available through the legitimate (ideal) quantum output

Towards the malicious case

- **Can ensure:** classical info of an honest run leaks nothing about θ not available through the legitimate (ideal) quantum output
- The classical connectivity of the GHZ state $|x_1\rangle + |x_2\rangle$ is **not fully unknown**

Towards the malicious case

- **Can ensure:** classical info of an honest run leaks nothing about θ not available through the legitimate (ideal) quantum output
- The classical connectivity of the GHZ state $|x_1\rangle + |x_2\rangle$ is **not fully unknown**
- Amplify this ignorance (**privacy amplification**) to ensure that even malicious acts cannot learn anything (see Alex's talk)

Towards the malicious case

- **Can ensure:** classical info of an honest run leaks nothing about θ not available through the legitimate (ideal) quantum output
- The classical connectivity of the GHZ state $|x_1\rangle + |x_2\rangle$ is **not fully unknown**
- Amplify this ignorance (**privacy amplification**) to ensure that even malicious acts cannot learn anything (see Alex's talk)
- Need to identify what part of θ is essential (and possible) to keep secret