Técnico Lisboa
University of Lisbon

Research Seminar in Information Security

(Prof. Paulo MATEUS)

Doctoral Program in Information Security

2023/2024 - 1st Semester

Quantum Protocols for Secure Multi-Party Computation:
From Theory to Practice

(Alex GRILO - November 23, 2023)

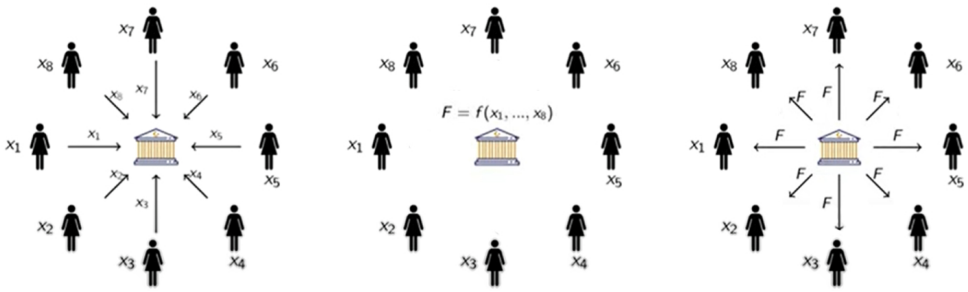
Report written by:

- Rúben BARREIRO:
  - *ruben.andre.letra.barreiro@tecnico.ulisboa.pt*

Last updated: May 4, 2024

1 Motivation

The task of interest in this seminar is to address how to use quantum properties to have Secure Multi-Party Computation (SMPC) protocols [1–6], from the theoretical to the practical side. First, we need to define the appropriate scenario for an SMPC protocol before entering into the deep details of this seminar. In these protocols, we have multiple N parties, and each one of them has some input data $\vec{x} = (x_1, x_2, ..., x_N)$. Then, they jointly want to compute an abstract function f on these mentioned input data. However, they do not want to reveal the respective information like that data itself. These SMPC protocols are one of the most general and fundamental cryptographic primitives if we do not know the number of rounds or how much communication we will need. So, ideally, in these protocols, we have a trusted party or a trusted node to which every other party can send their input data and information. This trusted node would compute the output data for the function f and send back this value to every other party or node.



(a) Each one of the N parties has some input data $\vec{x} = (x_1, x_2, ..., x_N)$ and send it to a "central" trusted party.

(b) The "central" trusted party computes an arbitrary function f on the input data $\vec{x} = (x_1, x_2, ..., x_N)$.

(c) The "central" trusted party send back the output result of the function f on the input data $\vec{x} = (x_1, x_2, ..., x_N)$.

Figure 1: High-level illustration of the usual steps involved in a Secure Multi-Party Computation (SMPC) protocol.

Therefore, we trust this "central" node would not reveal the information concerning that function or the input data. This function f can reveal some information about the input data, but the goal is that it does not

disclose more than the output data of that function f. Hence, this is the most that a party can ideally learn about the other parties' input data. However, the problem is we do not live in the ideal world but in the real world. Thus, we want to implement this functionality without the previously mentioned "central" trusted node. In that direction, we want a protocol where the parties interact and communicate with each other, and eventually, some of these parties are malicious. For example, they could want to deviate from the protocol to learn the input data of a single honest party. Then, the goal is to ensure that the function f is the only thing they are able to learn, even if some parties collude and behave dishonestly, trying to obtain as much information as they want from a single honest party.

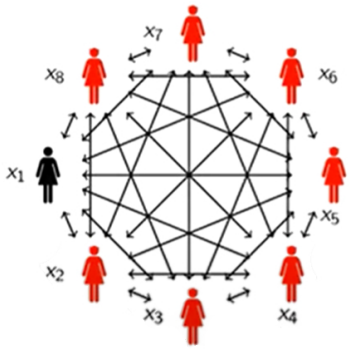


Figure 2: High-level illustration of a Secure Multi-Party Computation (SMPC) protocol, with malicious/dishonest parties.

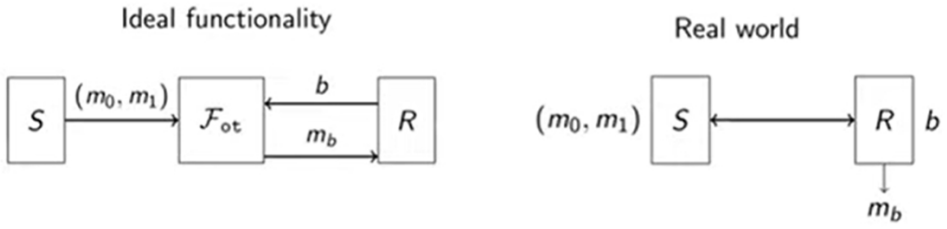
In Classical Cryptography, there are two principal families of SMPC protocols, and both of them rely on information-theoretic security (also called unconditional security), known as Goldreich-Micali-Wigderson (GMW) [7] and Ishai-Prabhakaran-Sahai (IPS) [8] families/paradigms. The former are SMPC protocols from honest SMPC based on ideal Zero-Knowledge Proofs (ZKPs) [9–11], and the latter are SMPC protocols from honest SMPC based on ideal Oblivious Transfers (OTs) [12–15]. Both families/paradigms rely on, for example, the property of the majority of the parties participating in the protocols are malicious and dishonest. Nevertheless, by the majority, we do not mean that there is necessarily only one honest party. Furthermore, these families can rely on much stronger cryptographic primitives, which we will address later. The research topic we will address in this seminar is how to

implement these SMPC sub-protocols for the honest case. Namely, we know how to achieve information-theoretically security for which we do not need computation assumptions. However, these two paradigms based on ZKPs and OTs usually make computational assumptions to allow us to implement them in real life. We can consider classical SMPC protocols with two types of security: pre-quantum and post-quantum security. In the pre-quantum security setting, the malicious parties can only perform classical computations, and, in this case, (Classical Pre-Quantum) Public Key Encryption (PKE) is somewhat necessary and sufficient to implement these protocols. Therefore, we do not need to require too much strong assumptions. Alternatively, for the post-quantum security setting, we also want to ensure security against quantum adversaries. In this case, we need very strong assumptions, such as Learning With Errors (LWE) assumptions [16–20]. Additionally, we also know that OT protocols from the IPS family/paradigm [8] are quantum-secure if we assume an ideal Commitment schemes [21–24]. However, both these approaches involve purely classical cryptographic protocols. But here, the question is if we can have quantum SMPC protocols that implement ideal OTs from assumptions weaker than the ones involving PKE or the classical post-quantum setting, such as the mentioned LWE.

2 Quantum Oblivious Transfer (QOT)

First, let's introduce what OT protocols are. In the ideal scenario, we have two parties, the sender and the receiver. Here, the sender would choose two messages, m_0, and m_1, for which the receiver chooses only one bit, b. Ideally, if we have a "central" trusted node in the protocol for the ideal scenario that implements this functionality F_{OT}, the receiver would receive back the message m_b. Here, we can notice that the sender does not learn which message the receiver learned, and the receiver learned just one of the messages, not both. We call this protocol Oblivious Transfer (OT) due to this property since the sender obliviously sends one of these messages to the receiver. Some compilers show how to build Multi-Party Computation (MPC) protocols if we have this basic functionality F_{OT}, in both classical and quantum settings. We do not know what really are the functionality F_{OT} we need to implement in the quantum setting for the real-world scenario. Therefore, we need to design protocols between these two mentioned parties, the sender and receiver, that are as good as having this idea of functionality in the middle of them, ideally

implemented by a "central' trusted node.



(a) An Oblivious Transfer (OT) protocol in the ideal world.

(b) An Oblivious Transfer (OT) protocol in the real world.

Figure 3: High-level illustration of an Oblivious Transfer (OT) protocol in ideal and real worlds.

Now, let's dive into some quantum protocols to perform this mentioned OT primitive. At the beginning times of Quantum Cryptography, Claude Crépeau and Joe Kilian proposed the Crépeau-Kilian-1988 (CK88) protocol [25, 26] in 1988, as well as Charles Bennett, Gilles Brassard, Claude Crépeau and Marie-Hélène Skubiszewska together proposed a similar protocol in 1992, known as Bennett-Brassard-Crépeau-Skubiszewska-1992 (BBCS92) protocol [27], which are two QOT protocols based on Commitment schemes, initially without real security proofs, which only appeared some years later [22, 23]. Despite these Commitment schemes being simple cryptographic primitives, once we build them from weak assumptions, we need much stronger schemes to prove the security of QOT protocols, which we cannot construct from simple assumptions, such as One-Way Functions (OWFs). Namely, the first security proofs for these two quantum protocols only appeared in the XXI century, based on classical Commitment schemes, likely to lie outside the MiniCrypt paradigm [28], where lies every cryptographic primitive we can build from OWFs but where we do not expect MPC protocols to be [28–30]. More recently, the first QOT protocols for strong Commitment schemes from OWFs emerged, such as the BCKM21 protocol [31], proposed by James Bartusek, Andrea Coladangelo, Dakshita Khurana, and Fermi Ma, as well as the GLSV21 protocol [32], proposed together by Alex Grilo, Huijia Lin, Fang Song, and Vinod Vaikuntanathan, both dated from 2021. The main point

now is that we know how to build QOT protocols, as well as how to build OT protocols classically from strong assumptions. These recent research results showed us how to construct these Commitment schemes needed in the security proofs for quantum protocols from OWFs. We often consider OWFs the classical minimal assumption in these security proofs since we cannot perform computational cryptography with OWFs classically. These research results show that we can build these quantum MPC protocols on what we can call MiniQCrypt paradigm [32], a variant of the MiniCrypt paradigm that uses quantum phenomena and minimal classical primitives. Therefore, this ends up giving us some interesting results in terms of computational complexity, in a natural sense that it is not possible to build MPC protocols only from OWFs, and we need this strong separation between quantum and classical resources on top of OWFs. Furthermore, between the '80s and the '90s, the research community thought we could build Commitment schemes with unconditional security only with quantum resources [33–41]. But later on, a few security proofs appeared saying we could not achieve that, and we always need to consider some additional classical minimal assumptions for these types of quantum protocols.

3 CK88/BBCS92 Protocols

In the CK88/BBCS92 protocol(s) [25–27], we have two parties as usually happens in a two-party quantum cryptographic protocol with λ rounds, namely the sender (e.g., Alice) and the receiver (e.g., Bob). The sender starts by picking a specific quantum state, such as the Bennett-Brassard-1984 (BB84) state [42, 43]. Then, the sender needs to pick a random bit string $\vec{x}$ (i.e., a string containing bits with the values 0s a 1s) of size λ in this case and a random bases list $\vec{\theta}$ (i.e., $\mathbb{X}$ and $\mathbb{Z}$ basis, for each round) of size λ to encode each one of those bits in a quantum state randomly prepared accordingly (i.e., $|0\rangle$, $|+\rangle$, $|1\rangle$, and $|-\rangle$, for each round) and send them to the receiver. On the other side, the receiver also picks a random bases list $\vec{\hat{\theta}}$ (i.e., $\mathbb{X}$ and $\mathbb{Z}$ basis, for each round) of size λ, measuring each one of the corresponding incoming quantum states on the respective randomly chosen basis, obtaining some resulting outcome list $\vec{\hat{x}}$, in the form of bits (i.e., 0s and 1s). Then, the sender sends the list $\vec{\theta}$ of bases randomly chosen to the receiver, who can split the outcomes information into two sets: the set where the bases randomly picked by both parties match between them and the set where those bases

do not match. Then, the receiver calls $I_b = \{i : \vec{\theta}_i = \vec{\hat{\theta}}_i\}$ to the set where the bases match where b is the classical bit they want to learn. This set I_b supposedly has the correct and completely correlated bits since the outcomes in these cases (i.e., the quantum states prepared and then measured on the same basis) are deterministic. On the other hand, each party has another set $I_{\bar{b}} = \{i : \vec{\theta}_i \neq \vec{\hat{\theta}}_i\}$ where the randomly chosen bases do not match, and the resulting obtained outcomes in these cases will be completely random in the form of incorrect and totally uncorrelated bits. For the set of matching bases, for example, the receiver has all the same information the sender has. The receiver sends the two sets to the sender only in the form of the corresponding indices. By doing that, we should notice that the sender does not know which set contains the matching bases because the receiver sends the I_0 and I_1, and this does not tell us what the I_b is for each case. Then, the sender prepares two messages m_0 and m_1, in the form of bit strings, and by using some sort of encryption, such as the Advanced Encryption Standard (AES) protocol [44], the sender encrypts the bits of those messages, using the bits corresponding to the indices of the initial randomly chosen bit string according to the sets I_b as the keys of the encryptor for the two messages m_0 and m_1, resulting in the two sets $a_0 = Enc_{\vec{x}_{I_0}}(m_0)$ and $a_1 = Enc_{\vec{x}_{I_1}}(m_1)$. Then, the sender sends these two encrypted messages a_0 and a_1 to the receiver. From these two encrypted messages, the receiver should be able to correctly learn the content of one of the messages m_b using the corresponding set of matching bases by performing the respective decryption $m_b = Dec_{\vec{x}_{I_b}}(a_b)$. However, there is a trivial security problem with these protocols, which is the receiver can decrypt both messages a_0 and a_1, if it has access to a bounded quantum memory, by keeping all the incoming quantum states in memory, waiting for the sender sending the randomly chosen bases used to prepare those quantum states, and then, being able to correctly measure all of them in the correct basis, obtaining the whole bit string randomly chosen by the sender, also obtaining both messages m_0 and m_1 at the end. This observation shows us that there are some security models where we assume the receiver has access to a bounded quantum memory, which makes these quantum protocols insecure. However, we want to be "paranoid" when building these security models and show that even malicious receivers with bounded quantum memory cannot cheat in this protocol. For this same reason, we need to add more cryptographic primitives to this quantum cryptographic protocol to force the receiver to measure always

the incoming quantum states without keeping them in a quantum memory.

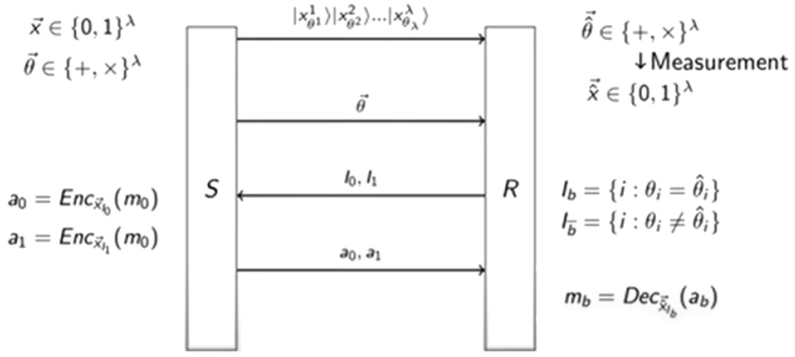


Figure 4: High-level illustration of both
the Crépeau-Kilian-1988 (CK88) and
Bennett-Brassard-Crépeau-Skubiszewska-1992 (BBCS92)
protocols.

4 Quantum Bit-Commitment Schemes with Simulation Model for Security

The cryptographic primitive we need as a solution for this last-mentioned security flaw is a Bit Commitment scheme. This cryptographic scheme is a two-phase protocol, where we still have two parties again: the sender, acting also as the committer of the protocol, and the receiver. In the first phase of the protocol, the committer has a message m, and it wants to commit to the value or content of this same message in a way that the sender and receiver exchange some message(s), and by the end, the receiver should obtain the commitment of this message m. In the second phase of the protocol, the committer will finally open this message m. Then, we want to ensure two properties for the correctness of this protocol. First, we want to ensure that the receiver should not be able to learn what the message really is during the first phase of the protocol, in a property usually called *hiding*. Second, we want to ensure that the committer cannot change its mind and choose to open a different message, and due to that, the commitment of m should fix this

message m, in a property we usually call *biding*. However, for Cryptography in general, we have several different types of security requirements, and these *hiding* and *biding* properties are known as the "vanilla" commitment requirements. But, there are other fencer ways of defining these commitment requirements and properties, such as by the well-known simulation model.

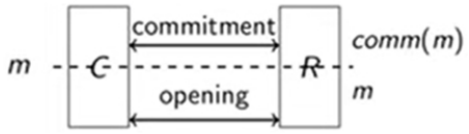
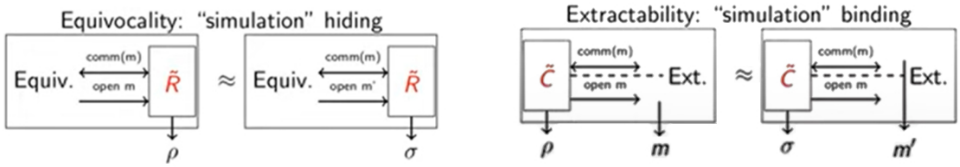


Figure 5: High-level illustration of
the Simulation Security Model of the Bit-Commitment protocols.

There is a simulation version of the *hiding* property called *equivocality* and a simulation version of the *biding* property called *extractability*. These two mentioned simulation variants of properties are the ones we morally need in this previous result that proves the security of the QOT protocol.



(a) Simulated version of *Hiding* property: *Equivocality*.

(b) Simulated version of *Biding* property: *Extractability*.

Figure 6: Simulated versions of the *Hiding* and *Biding*
security properties: *Equivocality* and *Extractability*.

Therefore, going back to the description of the QOT protocol, the idea is that we can add some additional steps to force the receiver to measure the incoming quantum states as soon as it receives them, making the protocol more secure. In order to achieve that, the receiver will commit to the pair with the randomly chosen basis and the resulting measurement outcome in each round $1 \leq i \leq \lambda$, sending this commitment pair $c_i = comm(\hat{\theta}_i, \hat{x}_i)$ to the sender. Then, the sender can pick just a random subset of indices T and ask the receiver to open the commitment pairs c_i for the rounds

corresponding to those indices $i \in T$. Namely, for the committed rounds where the randomly chosen bases match, the sender can verify if the outcomes are the same as expected. If we choose a big enough set T and perform enough testing rounds, we can have some guarantees that, even if the receiver does not measure all the corresponding quantum states, it had to have measured a lot of those quantum states to pass a pre-defined commitment test. After all the committed pairs are verified, the sender and the receiver can discard them. If the receiver has measured several quantum states corresponding to those committed pairs and passed the commitment test, we have guarantees that the receiver cannot decrypt both messages. And if we can ensure the receiver cannot decrypt both messages m_0 and m_1, we have what is necessary to prove the security of the QOT protocol in this case. Obviously, we need more formal security proof for this situation, but this is the intuition why we end up forcing the receiver to measure the incoming quantum states as soon as they arrive at its side by having this commitment verification. After performing this additional step of commitment verification, both parties continue the execution of the QOT protocol with the not discarded rounds.

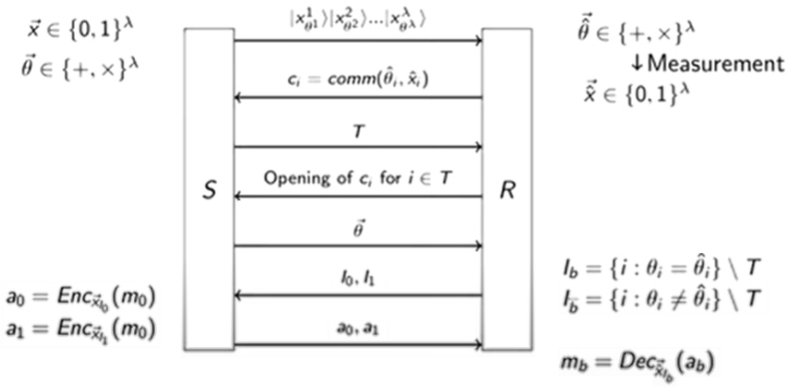


Figure 7: High-level illustration of
the Quantum Oblivious Transfer (QOT) CK88/BBCS92 protocol(s)
based on a Bit-Commitment scheme.

Obviously, we also need to consider the cases in which the simulator pretends to have committed to a measured quantum state but then changes its mind to prove security for the hiding property of the simulation model.

This scheme represents the security proof for the simulation model, using the *equivocality* property, allowing us to prove security against malicious senders:

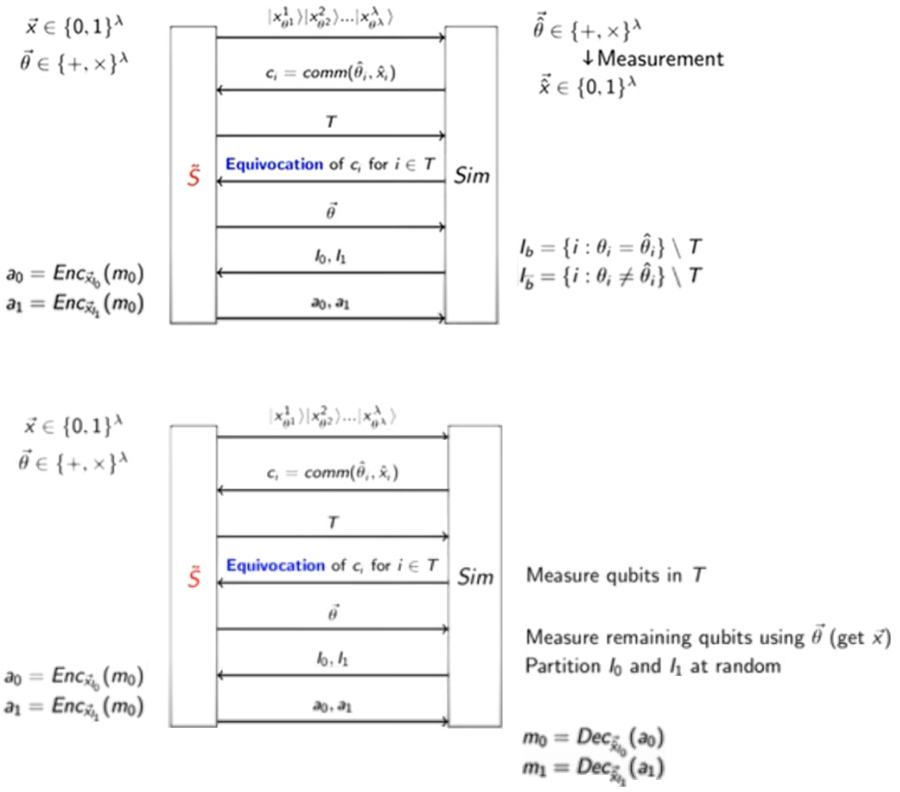


Figure 8: High-level illustration of
the simulated *Equivocality* security property for
the Quantum Oblivious Transfer (QOT) CK88/BBCS92 protocol(s)
based on a Bit-Commitment scheme.

On the other hand, this representative scheme represents the security proof for the simulation model, using the *extractability* property, which allows us to prove security against malicious receivers trying to learn both messages:

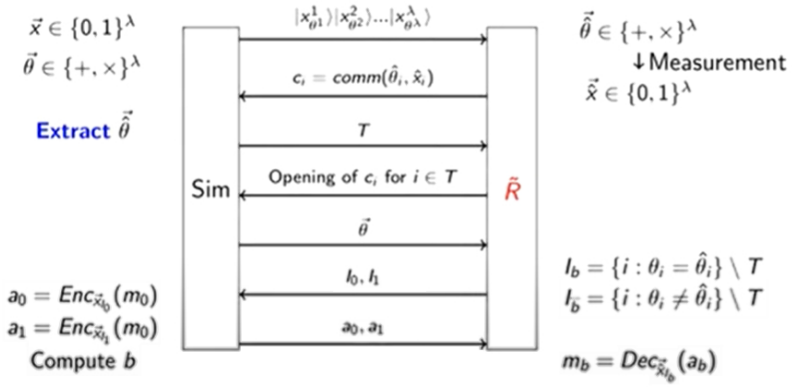


Figure 9: High-level illustration of
the simulated *Extractability* security property for
the Quantum Oblivious Transfer (QOT) CK88/BBCS92 protocol(s)
based on a Bit-Commitment scheme.

Then, we can use these two previously mentioned research results related to security proofs of BCKM21 [31] and GLSV21 [32] QOT protocols, for which the main goal is also to implement these Commitment schemes with the *hiding* and *biding* properties for the security model with Simulation. Namely, we can also use LWE problems to achieve these OT protocols based on Commitment schemes, for example. These two results showed us how to build these Commitment schemes using quantum protocols and having these hiding and biding properties. Actually, these two quantum protocols use OT as a bootstrap to achieve a Commitment scheme. Therefore, we take inspiration from both work proposals, and we can use the CK88 protocol [25, 26] with a weak commitment to obtain a Commitment scheme and then use this protocol again to achieve a full QOT protocol. The main differences

between BCKM21 [31] and GLSV21 [32] QOT protocols are given below:

<table>
<tr><th colspan="2">Protocols</th></tr>
<tr><th>BCKM21 [31]</th><th>GLSV21 [32]</th></tr>
<tr><th colspan="2">Specifications</th></tr>
<tr><td>1. (Black-Box)
Equivocality Compiler</td><td>1. Equivocal Commitment from
Naor's Commitment Scheme [45] and
Zero-Knowledge Proofs (ZKPs) [9–11]</td></tr>
<tr><td rowspan="2">2. Extractable
Commitment scheme from
Equivocal Bit-Commitment and
Quantum Communication</td><td>2. Unbounded-Simulator
Oblivious Transfer (OT) protocol from
Equivocal Commitment</td></tr>
<tr><td>3. Extractable and Equivocal
Bit-Commitment scheme from
Unbounded-Simulator
Oblivious Transfer (OT) protocol and
Quantum Communication</td></tr>
<tr><th colspan="2">Features</th></tr>
<tr><td>Black-Box use of
One-Way Functions (OWFs)</td><td>Constant-Round
Oblivious Transfer (OT) protocol in
the Common Reference String (CRS)
Model [46]</td></tr>
<tr><td>Statistical Security against
Malicious Receiver</td><td>Statistically Binding Extractable
Bit-Commitment Scheme</td></tr>
</table>

In an experimental "toy" scenario, let's consider we have two parties, the sender (Alex) and the receiver (Eleni), who start a conversation before the start of the protocol. The sender wants to tell the receiver it has a QOT protocol that only uses BB84 states. Thus, the only thing the sender has to send in this QOT protocol is BB84 states, and the receiver has to measure them, as happens in Quantum Key Distribution (QKD) protocols. Then, the parties agree on implementing this quantum protocol, and the conversation continues. The receiver asks the sender how much noise this protocol can tolerate, and the sender says there will be no noise in the quantum communication channel. Then, the receiver asks how many quantum states the sender will need to send, and the sender replies that it will need to send

$poly(\lambda)$ quantum states. After that, the receiver asks the sender what is this λ variable, to which the sender replies that it is the security parameter. Next, the receiver asks the sender how many bits of security will be achieved in this protocol since they are talking about a concrete implementation, to which the sender replies again that it will be $poly(\lambda)$. Finally, the receiver asks the sender about the requirements and complexity of classical post-processing, to which the sender replies that it has no idea how to implement it in practice.

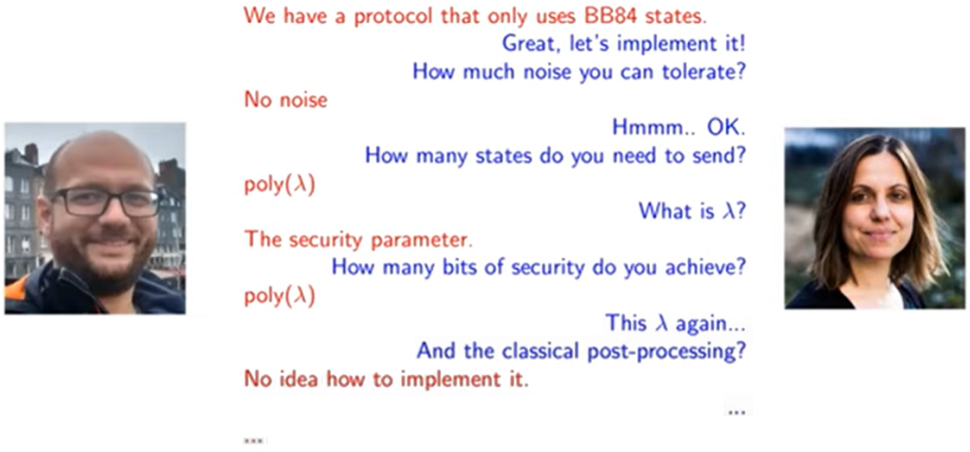


Figure 10: Experimental "toy" scenario for the setup of a QOT protocol based on a Bit-Commitment scheme.

5 Practical Implementation of QOT Protocols from OWFs

In order to implement this QOT protocol from OWF, we need to send these BB84 quantum states, which seems to imply the use of basic technology for QKD protocols. However, there are several reasons to assume a perfect creation/transmission/measurement of the quantum states in these quantum cryptographic protocols. Some of the physical apparatus we use for these quantum protocols are easy to fix if they fail, but some of them do not. These quantum cryptographic protocols rely on complicated classical primitives. For example, the author highlights a quantum protocol developed with his

team and based on ZKPs with great theoretical results, on which we can obtain ZKPs for all Nondeterministic Polynomial-time (NP) statements. But when we have a concrete language we want to prove using that protocol, we might be required to perform and implement a complicated reduction. In that case, the author's team usually avoids it and implements a different quantum protocol since translating asymptotic security to concrete cases is hard, especially in the quantum setting. From these previous theoretical results, we also know that, in theory, we need to repeat poly rounds on the protocol. However, we also want to know how many bits of security we achieve at the end, for which we need to be precise on the respective calculations that are hard. Therefore, the goal of the author and his team, is to push these quantum cryptographic protocols from theory to practice. Several interesting theory questions come out from these types of experiments. The goal of the author and his team is not to design new quantum cryptographic protocols from scratch but to consider some parts of the protocols and redesign them to be more memory efficient, for example. In that direction, they are focusing more on experimental and practical problems. One of the focuses of these tasks is to redesign the quantum cryptographic protocols to account for noise and to save classical memory since one of the problems of quantum cryptographic protocols for the latter case is that they require a lot of classical memory for classical post-processing. After redesigning these quantum protocols, the author and his team do not want to prove the security only from an abstract point of view, but also want to figure out how many quantum states are sending and how many bits of security they obtain at the end, by reproving the security of the protocols asymptotically and with concrete parameters. Finally, the author and his team want to physically implement and test these quantum cryptographic protocols in the lab, as in a real scenario where we have two real parties, impersonating Alice and Bob.

6 Open Questions

In this seminar, the author addressed some aspects of MPC protocols, but we should also think about some open questions and future work. For example, we can think about what else we can achieve on top of MPC or OT protocols with quantum resources and OWFs, usually as happens in the MiniQCrypt paradigm. We should also think about what (black-box) separations exist for cryptographic primitives in the quantum setting and MiniQCrypt paradigm.

Another direction we can follow is to show what we cannot achieve by building cryptographic protocols with quantum resources and OWFs. There are other interesting partial research results proposed recently on building new quantum cryptographic primitives from weaker assumptions than OWFs, such as Pseudo-Random Quantum States (PRQS) [47, 48] or even Efficient-Far-Indistinguishable (EFI) pairs of quantum states [49–51]. On the other hand, we can think about which assumptions we can consider from the quantum world that could be weaker than OWFs, maybe unveilling a new paradigm. Finally, we can also think about other more practical new quantum cryptographic protocols or what we should do to transform the existing quantum cryptographic primitives on more practical variants.

References

[1] Adi Shamir, Ronald Rivest, and Leonard Adleman. *Mental Poker*, pages 37–43. Springer US, 1981.

[2] Andrew Yao. Protocols for Secure Computations. In *23rd Annual Symposium on Foundations of Computer Science (SFCS - 1982)*, pages 160–164, 1982.

[3] Andrew Yao. How to Generate and Exchange Secrets. In *27th Annual Symposium on Foundations of Computer Science (SFCS - 1986)*, pages 162–167, 1986.

[4] Oded Goldreich, Silvio Micali, and Avi Wigderson. How to Play Any Mental Game or A Completeness Theorem for Protocols with Honest Majority. In Alfred Aho, editor, *Proceedings of the Nineteenth Annual ACM Symposium on Theory of Computing*, STOC '87, pages 218–229. ACM, 1987.

[5] Michael Ben-Or, Shafi Goldwasser, and Avi Wigderson. Completeness Theorems for Non-Cryptographic Fault-Tolerant Distributed Computation. In *Proceedings of the Twentieth Annual ACM Symposium on Theory of Computing*, STOC '88, page 1–10. Association for Computing Machinery, 1988.

[6] David Chaum, Claude Crépeau, and Ivan Damgård. Multiparty Unconditionally Secure Protocols. In *Proceedings of the Twentieth Annual ACM Symposium on Theory of Computing*, STOC '88, page 11–19. Association for Computing Machinery, 1988.

[7] Oded Goldreich, Silvio Micali, and Avi Wigderson. How to Play ANY Mental Game. In *Proceedings of the Nineteenth Annual ACM Symposium on Theory of Computing*, STOC '87, page 218–229. Association for Computing Machinery, 1987.

[8] Yuval Ishai, Manoj Prabhakaran, and Amit Sahai. Founding Cryptography on Oblivious Transfer – Efficiently. In David Wagner, editor, *Advances in Cryptology – CRYPTO 2008*, pages 572–591. Springer Berlin Heidelberg, 2008.

[9] Shafi Goldwasser, Silvio Micali, and Charles Rackoff.
The Knowledge Complexity of Interactive Proof-Systems. In *Proceedings of the Seventeenth Annual ACM Symposium on Theory of Computing*, STOC '85, page 291–304. Association for Computing Machinery, 1985.

[10] Amos Fiat and Adi Shamir. How To Prove Yourself:
Practical Solutions to Identification and Signature Problems. In Andrew Odlyzko, editor, *Advances in Cryptology – CRYPTO' 86*, pages 186–194. Springer Berlin Heidelberg, 1987.

[11] Manuel Blum, Paul Feldman, and Silvio Micali.
Non-Interactive Zero-Knowledge and its Applications. In *Proceedings of the Twentieth Annual ACM Symposium on Theory of Computing*, STOC '88, page 103–112. Association for Computing Machinery, 1988.

[12] Michael Rabin. How to Exchange Secrets by
Oblivious Transfer. Technical Report TR-81,
Harvard Aiken Computation Laboratory, 1981.

[13] Shimon Even, Oded Goldreich, and Abraham Lempel. A Randomized Protocol for Signing Contracts. *Commun. ACM*, 28(6):637–647, 1985.

[14] Claude Crépeau. Equivalence Between Two Flavours of
Oblivious Transfers. In Carl Pomerance, editor, *Advances in Cryptology – CRYPTO '87*, pages 350–354.
Springer Berlin Heidelberg, 1988.

[15] Joe Kilian. Founding Cryptography on
Oblivious Transfer. In *Proceedings of the Twentieth Annual ACM Symposium on Theory of Computing*,
STOC '88, page 20–31. Association for Computing Machinery, 1988.

[16] Oded Regev. On Lattices, Learning With Errors, Random Linear Codes, and Cryptography. *J. ACM*, 56(6), 2009.

[17] Rikke Bendlin, Ivan Damgård, Claudio Orlandi, and
Sarah Zakarias. Semi-Homomorphic Encryption and
Multiparty Computation. In Kenneth Paterson, editor,
Advances in Cryptology – EUROCRYPT 2011, pages 169–188.
Springer Berlin Heidelberg, 2011.

[18] Gilad Asharov, Abhishek Jain, Adriana López-Alt, Eran Tromer, Vinod Vaikuntanathan, and Daniel Wichs. Multiparty Computation with Low Communication, Computation and Interaction via Threshold FHE. In David Pointcheval and Thomas Johansson, editors, *Advances in Cryptology – EUROCRYPT 2012*, pages 483–501. Springer Berlin Heidelberg, 2012.

[19] Johannes Buchmann, Niklas Büscher, Florian Göpfert, Stefan Katzenbeisser, Juliane Krämer, Daniele Micciancio, Sander Siim, Christine van Vredendaal, and Michael Walter. Creating Cryptographic Challenges Using Multi-Party Computation: The LWE Challenge. In *Proceedings of the 3rd ACM International Workshop on ASIA Public-Key Cryptography*, AsiaPKC '16, page 11–20. Association for Computing Machinery, 2016.

[20] Pratyay Mukherjee and Daniel Wichs. Two Round Multiparty Computation via Multi-key FHE. In Marc Fischlin and Jean-Sébastien Coron, editors, *Advances in Cryptology – EUROCRYPT 2016*, pages 735–763. Springer Berlin Heidelberg, 2016.

[21] Serge Fehr and Christian Schaffner. Composing Quantum Protocols in a Classical Environment, 2008.

[22] Ivan Damgård, Serge Fehr, Carolin Lunemann, Louis Salvail, and Christian Schaffner. Improving the Security of Quantum Protocols via Commit-and-Open, 2009.

[23] Niek Bouman and Serge Fehr. Sampling in a Quantum Population, and Applications, 2012.

[24] Dominique Unruh. Universally Composable Quantum Multi-Party Computation. In Henri Gilbert, editor, *Advances in Cryptology – EUROCRYPT 2010*, pages 486–505. Springer Berlin Heidelberg, 2010.

[25] Claude Crepéau and Joe Kilian. Achieving Oblivious Transfer Using Weakened Security Assumptions. In *29th Annual Symposium on Foundations of Computer Science*, pages 42–52, 1988.

[26] Claude Crepéau and Joe Kilian. Weakening Security Assumptions and Oblivious Transfer. In *Advances in Cryptology - CRYPTO '88 - 8th Annual International Cryptology Conference*, volume 403 of *Lecture Notes in Computer Science*, pages 2–7. Springer, 1988.

[27] Charles Bennett, Gilles Brassard, Claude Crépeau, and Marie-Hélène Skubiszewska. Practical Quantum Oblivious Transfer. In Joan Feigenbaum, editor, *Advances in Cryptology – CRYPTO '91*, pages 351–366. Springer Berlin Heidelberg, 1992.

[28] Russell Impagliazzo. A Personal View of Average-Case Complexity. *Proceedings of Structure in Complexity Theory. Tenth Annual IEEE Conference*, pages 134–147, 1995.

[29] Russell Impagliazzo and Steven Rudich. Limits on the Provable Consequences of One-Way Permutations. In Shafi Goldwasser, editor, *Advances in Cryptology – CRYPTO' 88*, pages 8–26. Springer New York, 1990.

[30] Steven Rudich. The Use of Interaction in Public Cryptosystems. In Joan Feigenbaum, editor, *Advances in Cryptology – CRYPTO '91*, pages 242–251. Springer Berlin Heidelberg, 1992.

[31] James Bartusek, Andrea Coladangelo, Dakshita Khurana, and Fermi Ma. One-Way Functions Imply Secure Computation in a Quantum World, 2020.

[32] Alex Grilo, Huijia Lin, Fang Song, and Vinod Vaikuntanathan. Oblivious Transfer is in MiniQCrypt, 2020.

[33] Gilles Brassard, Claude Crépeau, Richard Jozsa, and Denis Langlois. A Quantum Bit Commitment Scheme Provably Unbreakable by Both Parties. In *Proceedings of 1993 IEEE 34th Annual Foundations of Computer Science*, pages 362–371, 1993.

[34] Gilles Brassard and Claude Crépeau. 25 Years of Quantum Cryptography. *ACM Sigact News*, 27(3):13–24, 1996.

[35] Claude Crépeau. What is Going On With Quantum Bit Commitment? In *Proceedings of Pragocrypt*, volume 96, pages 193–203, 1996.

[36] Dominic Mayers. Unconditionally Secure Quantum Bit Commitment is Impossible. *Physical Review Letters*, 78(17):3414–3417, 1997.

[37] Gilles Brassard, Claude Crépeau, Dominic Mayers, and Louis Salvail. A Brief Review on the Impossibility of Quantum Bit Commitment, 1997.

[38] Hoi-Kwong Lo and Hoi Chau. Why Quantum Bit Commitment and Ideal Quantum Coin Tossing are Impossible? *Physica D: Nonlinear Phenomena*, 120(1):177–187, 1998. Proceedings of the Fourth Workshop on Physics and Consumption.

[39] Gilles Brassard, Claude Crépeau, Dominic Mayers, and Louis Salvail. Defeating Classical Bit Commitments with a Quantum Computer, 1998.

[40] Adrian Kent. Quantum Bit Commitment From a Computation Bound, 1999.

[41] Adrian Kent. Permanently Secure Quantum Bit Commitment From a Temporary Computation Bound, 1999.

[42] Charles Bennett and Gilles Brassard. Quantum Cryptography: Public Key Distribution and Coin Tossing. In *Proceedings of IEEE International Conference on Computers, Systems, and Signal Processing*, 1984.

[43] Charles Bennett and Gilles Brassard. Quantum Cryptography: Public Key Distribution and Coin Tossing. *Theoretical Computer Science*, 560:7–11, 2014. Theoretical Aspects of Quantum Cryptography – celebrating 30 years of BB84.

[44] Vincent Rijmen and Joan Daemen. Advanced Encryption Standard. *Proceedings of Federal Information Processing Standards Publications, National Institute of Standards and Technology*, 19:22, 2001.

[45] Moni Naor. Bit Commitment Using Pseudorandomness. *Journal of Cryptology*, 4(2):151–158, 1991.

[46] Ran Canetti and Marc Fischlin. Universally Composable Commitments, 2001.

[47] Zhengfeng Ji, Yi-Kai Liu, and Fang Song. Pseudorandom Quantum States. In Hovav Shacham and Alexandra Boldyreva, editors, *Advances in Cryptology – CRYPTO 2018*, pages 126–152. Springer International Publishing, 2018.

[48] Prabhanjan Ananth, Luowen Qian, and Henry Yuen. Cryptography From Pseudorandom Quantum States, 2022.

[49] Jun Yan. General Properties of Quantum Bit Commitments, 2020.

[50] Zvika Brakerski, Ran Canetti, and Luowen Qian. On the Computational Hardness Needed for Quantum Cryptography, 2022.

[51] Tomoyuki Morimae and Takashi Yamakawa. One-Wayness in Quantum Cryptography, 2022.