Técnico Lisboa
University of Lisbon


Research Seminar in Information Security

(Prof. Paulo MATEUS)

Doctoral Program in Information Security

2023/2024 - 1st Semester

---

# Robust Quantum Public-Key Encryption

**(with Applications to Quantum Key Distribution)**

**(Giulio MALAVOLTA - 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 phenomena to build a novel cryptographic protocol for public-key encryption, whose security, only relies on the use of authenticated classical channels [1].

The Modern Cryptography as we know it today started with the work of Whitfield Diffie and Martin Hellman in 1976, when they showed a Key Exchange (KE) protocol, nowadays known as the Diffie-Hellman (DH) protocol [2, 3]. This work proposal represents essentially the first time that people performed what we know today as Public Key Encryption (PKE). A couple of years later, namely in 1978, Ron Rivest, Adi Shamir, and Leonard Adleman published together a breakthrough PKE algorithm, today known as the Rivest-Shamir-Adleman (RSA) algorithm [4], based on the practical difficulty of factoring the product of two large prime numbers. These two research works greatly influenced the Modern Cryptography we use nowadays. Indeed, there is a difference between performing a KE protocol and a PKE algorithm, in the sense that a KE protocol is an interactive protocol and a PKE algorithm is just a two-message protocol, one from a sender party (e.g., Alice) that sends its public key to a receiver (e.g., Bob) and other from the receiver that can use it to encrypt a confidential message headed posteriorly to a sender party. Obviously, this KE protocol also works the other way around, where a receiver party can send its public key to a sender, and a sender can use it to encrypt a confidential message headed posteriorly to the receiver party. These work proposals won several prizes, including the Turing Awards, already in the XXI century. We cannot also deny the impact these work proposals had on our lives since it has several applications, including social networks (e.g., Facebook and Instagram), messaging services (e.g., WhatsApp), Internet and World Wide Web (WWW) and related protocols, such as Hypertext Transfer Protocol Secure (HTTPS) and Transport Layer Security (TLS) protocols, or even global economy, among many other. Actually, there is a famous quote from Silvio Micalli that ends up being true, and that says: "Cryptographers seldom sleep well". The reason for that is that essentially, both these two work proposals marked the birth of the Modern Cryptography field as we know it today when they just proposed a cryptographic scheme that allows two parties without previously meeting with each other, with the hope that it will be secure forever. In fact, each one of these cryptographic schemes comes with the respective security assumptions, namely the DH and the RSA assumptions, which work as

formal proofs that these cryptographic schemes are secure. However, some technological and scientific advancements happened in the last years, which led us to want to be sure to have Unconditional Security on the cryptographic schemes we develop today. In simple words, Unconditional Security implies we make no assumptions about the computing power and (technological) resources available to an adversary. At this stage, we also know we cannot achieve Unconditional Security for cryptographic tasks such as KE protocols and PKE algorithms unless we bring Quantum Information into the picture. In that direction, we have the famous Bennett-Brassard-1984 (BB84) protocol, proposed by Charles Bennett and Gilles Brassard in 1984 [5,6], that provides the well-desired Unconditional Security once we do not have to trust any Cryptography to have any proven assumption to have security. We can think of the BB84 protocol as a sort of KE protocol similar to the DH KE protocol, but we can also use Quantum Cryptography to perform the equivalent of the PKE task from the RSA algorithm. However, the BB84 protocol is not considered a standard PKE algorithm, no matter the sort of message we send through it. This quantum cryptographic protocol has at least three rounds of communication between Alice and Bob, for which we can design a "toy version" interactive protocol such as illustrated below:

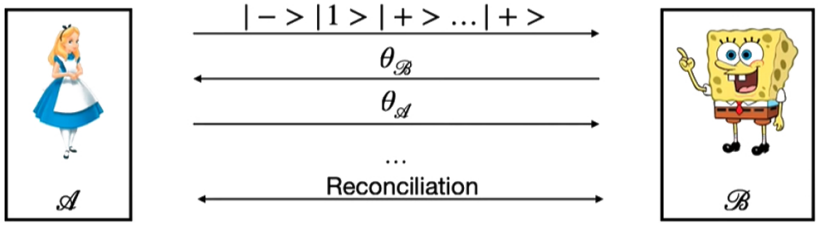


Figure 1: High-level illustration of a "toy version" of the Bennett-Brassard-1984 (BB84) protocol.

Namely, the sender starts by sending a bunch of quantum states prepared on some randomly chosen bases to the receiver, for which the receiver measures them on some basis randomly chosen. Then, the receiver sends the bases used for the measurement of those quantum states to the sender, to which the sender replies with the bases used to prepare the quantum states sent. After that, both sender and receiver perform some general error reconciliation

steps, and we ultimately know that both parties will agree on a secret key. No matter how we try to simplify this illustrated "toy version" protocol, we cannot reduce its round complexity or at least reduce it beyond three rounds.

However, there is a recent theorem that the author and a colleague proved, saying that if a quantum-secure One-Way Function (OWF) exists, then there exists a two-message round Quantum Key Distribution (QKD) protocol, or in other words, a Quantum Public Key Encryption (QPKE) algorithm with Everlasting Security [1]. First of all, a two-message round protocol is clearly optimal once we know that the sender has to send something and the receiver also has to send something. If we choose the standard asynchronous model rather than the simultaneous message model, we conclude we cannot perform a KE protocol based on this QKD protocol in less than two message rounds. Second, although we do not achieve Unconditional Security, we achieve another form of security known as Everlasting Security. Achieving this form of security is also impossible for both KE protocols and PKE algorithms unless we use quantum information. This fact justifies we are manipulating quantum bits (qubits) instead of just sending (classical) bits in these QKD protocols, such as the BB84 protocol. The concept of Everlasting Security essentially says that we can have computational assumptions, but we only have to assume that they hold only during the execution of the protocol. Alternatively, the concept of Unconditional Security says that security must also hold after the execution of the protocol and for the whole eternity. Practically speaking, Everlasting Security implies we have to assume that a (computational) problem must be hard to solve for 30 seconds, for example, and not for 100 or 1000 years, which is probably what we usually care about for the security of these PKE algorithms and KE protocols. In some aspects, the Everlasting Security model seems to be a much better computational and security model, reason why we would like to achieve that. Even if we are willing to believe in the security of our protocol against any computational assumption, we know that using only classical information jointly with OWFs is insufficient to perform PKE algorithms and KE protocols. Namely, we do not have formal proof for this statement, but we do not have strong evidence against it. Another remarkable aspect of this approach is that we only use OWFs jointly with these two-message QKD protocols if we manipulate quantum information to achieve computational security. Of course, the most substantial part of this protocol is that it is a two-message round protocol if we do not care about rounds of interactions of a usual QKD protocol.
Now, we can highlight the differences between Everlasting Security and

Unconditional Security. For the Everlasting Security model, we admit some computational assumptions but only during the execution of the first stage of the protocol. Alternatively, we have no computational assumptions for the Unconditional Security model since the security must last indefinitely, independently of technological breakthroughs. We usually also need to use authenticated classical communication channels for the second phase of these quantum cryptographic protocols, which allows us to be sure about who we are talking with during the reconciliation step. However, the existence of an authenticated classical communication channel is an extra assumption for the Everlasting Security model, depending on computational assumptions, which we only consider during the execution of the unique stage of the protocol. This Everlasting Security model is very different from the Unconditional Security model from the BB84 protocol, where we have no computational

**Security Model**

| **Everlasting Security** | **Unconditional Security** |
|---|---|
| • Needs some Computational Assumptions, *only during* the protocol! | • Does not need Computational Assumptions! |
| • ~~Use Authenticated Classical Channels~~ | • Use Authenticated Classical Channels<br>– **Needs Computational Assumptions...**<br>– **... but *only during* the execution of the protocol!** |

Table 1: Overview of the main differences between the Everlasting Security and Unconditional Security.

assumptions but still need to authenticate classical information during the reconciliation step. Finally, we need an answer to the following question: "How do we get authenticated classical communication channels?". For that, we just need computational assumptions that only have to hold during the execution of the protocol. Additionally, the parties must have a pre-shared key *a priori* to achieve that authenticated classical communication channel.

# 2 Definitions

## 2.1 Quantum Public Key Encryption (QPKE)

First, we need to make clear what means a Quantum Public Key Encryption (QPKE) algorithm and define it. In a QPKE algorithm, the sender (i.e., Alice) sends a public key consisting of a classical component denoted as $pub\_key_S$ through a classical communication channel and a quantum state $\rho$ through a quantum communication channel. Then, the receiver (i.e., Bob) can use this information set to encrypt a message $msg$ using the quantum state $\rho$ and the public key $pub\_key_S$ that the sender sent. Finally, the sender should be able to recover the message $msg$ back from the encrypted data sent by the receiver. We show an illustrative sketch of this algorithm below:

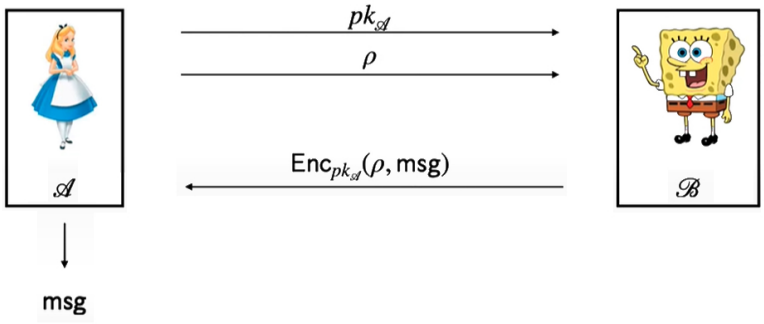


Figure 2: High-level illustration of
a simple Quantum Public Key Encryption (QPKE) protocol.

Note this particular quantum cryptographic primitive allows the receiver to encrypt any message and offers slightly better functionality than the standard approaches, which suffices for a whole implementation of a QKD protocol.

## 2.2 Security Definition

The security we want is that for all quantum polynomial time adversaries and all pairs of messages $(msg_0, msg_1)$, we are going to define a game where the sender sends the public key $pub\_key_S$ and the quantum state $\rho$ to an adversary (e.g., Eve). Here, we allow the adversary to perform the operations it wants on this (arbitrary) quantum state $\rho$ once we do not have any authentication on the transmission of the quantum state $\rho$. On the other hand, we assume that the public key $pub\_key_S$ of the sender is delivered honestly to the receiver in what we can call again an authenticated classical communication channel. On the other hand, we have no assumption on what the quantum state $\rho$ really is. Additionally, even with this strong condition, we want to hide the message $msg$ encrypted by the receiver from the eyes of the adversary. We can think of the quantum state $\rho$ as a quantum component of the output of the public key generation algorithm, along with a classical component, which is a classical secret key extracted from the private key generation algorithm. In this simple experiment, we considered the sample of the quantum state $\rho$ from flipping a coin $c \leftarrow \{0, 1\}$ which results in preparing $|0, \sigma_0\rangle\langle 0, \sigma_0|$ if $c = 0$ and $|1, \sigma_1\rangle\langle 1, \sigma_1|$ if $c = 1$, where $\sigma_0$ and $\sigma_1$ are two key pairs components generated from the private key $priv\_key_S$. If we have a perfectly random coin, we end up having the quantum state as a classical mixture given as follows:

$$\rho = \frac{|0, \sigma_0\rangle\langle 0, \sigma_0| + |1, \sigma_1\rangle\langle 1, \sigma_1|}{2}$$

The way we formalize these aspects of the security proof we are defining is to say the adversary can return a quantum state $\rho^*$ to the receiver, such that the encryption algorithm accepts (i.e., does not abort) with non-negligible probability. Therefore, this quantum state $\rho^*$ would just be delivered to the receiver, and the receiver uses it for encrypting the message $msg$. Additionally, the trace distance between the quantum state $\rho^*$ returned when $c = 0$ and the quantum state $\rho^*$ returned when $c = 1$ both along with the adversary's projector $\prod$, should be approximately equal to 0 and thus negligible in the security parameter $\lambda$ of this (honest) quantum cryptographic protocol. This security claim is given by the mathematical expression below:

$$Td(\tau_b) = Td(\tau_0, \tau_1) = Td\left(\frac{\prod \rho^* \prod}{Tr(\prod \rho^*)}, |c, \sigma_c\rangle\langle c, \sigma_c|\right) = negl(\lambda) \approx 0$$

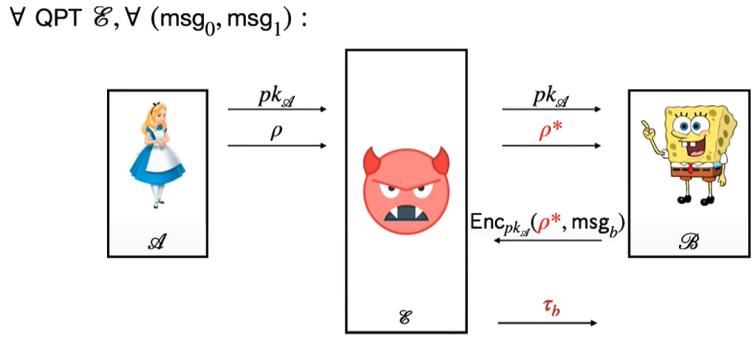


Figure 3: High-level illustration of the Security Definition sketch of a simple Quantum Public Key Encryption (QPKE) protocol.

## 2.3 One-Time Digital Signatures (OTSs)

In order to show how this quantum cryptographic protocol works, we need to recall the definition of some required cryptographic primitives. First, we need to recall what is a One-Time Digital Signature or simply One-Time Signature (OTS) [7]. An OTS is a classical cryptographic primitive consisting of three algorithms: a Key Generation, a Signing, and a Verification algorithm. Roughly speaking, this cryptographic primitive allows us to sign a message $msg$, which everyone can verify its authenticity and correctness. There is a (very weak) specific security notion called existential unforgeability of this cryptographic primitive that demands that nobody else besides the owner of the secret key can sign a message $msg$. Namely, this notion says that an adversary is only allowed to sign a single message of their choice, and even in that case, should not be able to produce a digital signature on a new message, or in other words, should be (computationally) hard for an adversary to forge a new valid digital signature. Furthermore, another significant point is that OTSs exist if and only if One-Way Functions (OWFs) [3, 7, 8] exist. This equivalence claim is a consequence of a research result from the '70$s$, and it is the only cryptographic material we will use here.

## 2.4 Key Generation (from Sender)

For the Key Generation algorithm of our quantum cryptographic protocol, we start with the sender party sampling an OTS as a classical component and computing a key pair $(sign\_key, ver\_key)$ for signing and verifying digital

signatures, respectively. After that, the sender party will also compute the following quantum state $|\psi\rangle$, prepared in a uniform quantum superposition:

$$|\psi\rangle = \frac{|0,0,\sigma_0\rangle + |1,1,\sigma_1\rangle}{\sqrt{2}}$$

Here, $\sigma_0$ is a digital signature on some arbitrary message $m_0$, and $\sigma_1$ is a digital signature on some arbitrary message $m_1$. Implicitly, the verification key sampled in this stage of the protocol defines a projective measurement $\left\{\prod_{ver\_key}, I - \prod_{ver\_key}\right\}$, which allows anyone to take an arbitrary quantum state and project it onto the span of valid digital signatures composed by the pair of a message and the respective digital signature. Namely, we define the projector $\prod_{ver\_key}$ for the verification key $ver\_key$ as presented below:

$$\prod_{ver\_key} = \sum_{\sigma:\ \mathrm{Verify}(ver\_key,0,\sigma)=1} |0,\sigma\rangle\langle 0,\sigma| + \sum_{\sigma:\ \mathrm{Verify}(ver\_key,1,\sigma)=1} |1,\sigma\rangle\langle 1,\sigma|$$

Note we can efficiently apply this projection operator $\prod_{ver\_key}$ if we know the verification key $ver\_key$ once we only need to execute the verification algorithm correctly. After that, the sender party will also measure the first register containing the quantum state $|\psi\rangle$ in the Hadamard basis to obtain a single (classical) bit $s \in \{0,1\}$ that will be a one-time secret key, but first that quantum state $|\psi\rangle$ should be kept in a quantum memory until applying the decryption algorithm on an incoming encrypted message. Thus, the secret key that the sender party ends up obtaining consists of this secret bit $s$ and the residual quantum state $\rho$ sent to the receiver party after measuring the first register containing the quantum state $|\psi\rangle$ in the Hadamard basis.

## 2.5 Encryption (from Receiver)

For the encryption algorithm, the receiver obtains the residual quantum state $\rho$ from the sender party, checking if that quantum state $\rho$ is in the correct subspace by applying this projection defined by the verification key to that quantum state $\rho$ and see if it succeeds. Otherwise, if the projection fails, the receiver should abort the execution of the protocol. Again, this verification is efficiently implementable, and the essential guarantee we have this operation succeed is that the residual quantum state $\rho$ that consists of a quantum superposition of valid pairs of digital signatures of messages does not have to

be strictly equal to the one the sender prepared, but that will be sufficient for us. The receiver proceeds by measuring the projected residual state $\rho$ in the Hadamard basis to obtain a bit string of size $(n+1)$ we denote as a pair of information $(d_1, d_2) \in \{0,1\} \times \{0,1\}^n$, where $d_1$ is a single bit, which will work as a One-Time Pad (OTP) key that the receiver uses to mask its message $msg$ and $d_2$ is a form of auxiliary information for the encryption and decryption algorithms, which will be part of the ciphertext. Therefore, the encrypted message the receiver sends to the sender will be $ct = (msg \oplus d_1, d_2)$.

### 2.6 Decryption (from Sender)

For the decryption algorithm, we first pretend to delay the measurement of the sender party on the first register of the quantum state $|\psi\rangle$ mentioned before on the Hadamard basis, which does not affect the correctness of this quantum cryptographic protocol. Furthermore, we can recall the receiver party also measured the projected quantum state $\rho$ on the Hadamard basis. The rotated quantum state $|\psi\rangle$ in the Hadamard basis has the following form:

$$H \times |\psi\rangle = \sum_{d} (-1)^{[d\cdot(0,0,\sigma_0)]} \times |d\rangle + (-1)^{[d\cdot(1,1,\sigma_1)]} \times |d\rangle = \sum_{d:\ d\cdot(1,1,\sigma_0\oplus\sigma_1)=0} |d\rangle$$

Namely, this rotated quantum state consists of a quantum superposition of all bit strings $d$ such that this quantum state is orthogonal to them. Therefore, we can verify that this calculation is correct. Then, the measurement of the rotated quantum state gives us a bit string $d$ satisfying the following equation:

$$d = (s, d_0, d_1), \text{ such that } d_1 \oplus d_2 \cdot (\sigma_0, \sigma_1) = s$$

Note that if we rearrange the equation above, the sender can also compute the term $d_1$ by performing some rearrangements on the equation above. First, $d_2$ is part of the ciphertext $ct$ received by the sender. Second, the sender knows $\sigma_0$ and $\sigma_1$, as well as the one-time secret key $s$. For that, we must recall the receiver sent the encrypted message $ct = (msg \oplus d_1, d_2)$, which from computing $d_1$, the sender can recover the original message $msg$ as well.

## 3 Proof Sketch

For the proof sketch, we can simplify the proof a little bit but keep its main argument. In the first place, we observe that from the point of view of the

attacker who does not see the first measurement from the sender party, the residual quantum state $\rho$ is essentially a classical mixture given as follows:

$$|0, \sigma_0\rangle \text{ with probability of } \frac{1}{2}$$

$$|1, \sigma_1\rangle \text{ with probability of } \frac{1}{2}$$

Here, the quantum states $|0, \sigma_0\rangle$ and $|1, \sigma_1\rangle$ occur each one with a probability of 50% because we traced out the first register of the quantum state $\rho$.

Now, we can recall the first action from the receiver is to project the residual quantum state $\rho$ into the span of the pairs of correct digital signatures of messages $(b, \sigma_b)$. In other words, the receiver party needs to project the residual quantum state $\rho$ onto $\text{Span}(\{|b, \sigma_b\rangle : \text{Verify}(ver\_key, b, \sigma_b) = 1\})$. The attacker only has two options for passing the projection test and cheating this quantum cryptographic protocol. The attacker must have acted as the identity operator, on what is equivalent to just letting the quantum state pass in the respective quantum communication channel with no action, or it can put some non-trivial amplitude weights on another digital signature. However, this second option is not valid since it breaks unforgeability, which we defined before as a property of our OTS primitives. Namely, for a simple security reduction, given only one digital signature, it is (computationally) hard to compute any non-negligible amplitude weight on the other digital signature. We could measure that quantum state $\rho$ on the Hadamard basis, which results in a quantum superposition of a uniform bit string. Therefore, if we use any bit from the resulting (completely random) uniform bit string $d$ as an OTP primitive, we achieve the well-desired Unconditional Security:

$$d \sim \text{Uniform: } \{0,1\}^{(n+1)} \longrightarrow \begin{array}{c}\text{Unconditional Security} \\ \text{using One-Time Pad}\end{array}$$

# 4 Outlook

## 4.1 Possible Improvements

The author highlights some aspects we can think about possible improvement directions. First, the author and his colleague did not optimize the (initial) key rate of this initial work proposal for this quantum cryptographic protocol, and they need to improve this feature in the future. Second, the "naïve"

security model the author and his colleague developed currently only works for noiseless scenarios where we do not need to care about noise nuances, and the protocol should have a better noise tolerance for more realistic scenarios. Another aspect the author highlights is that this current quantum cryptographic protocol uses large coherent quantum states for the valid pairs of digital signatures of messages, which would imply better quantum hardware and larger quantum memories than the currently available ones, turning the implementation of this cryptographic primitive with a reasonable security parameter much more difficult at the moment. A possible way to overcome this issue should be to think about a (relatively) equivalent protocol with the same round complexity that works qubit-by-qubit, similarly to what happens with most Quantum Key Distribution (QKD) protocols [9–16], such as the previously mentioned and well-known BB84 protocol [5, 6]. Another aspect that the author highlights with a more foundational interest is the computational assumptions considered for this quantum cryptographic protocol once the author and his colleague considered OWFs sufficient for those assumptions. However, it turns out that we do not believe OWFs to be the minimal assumption in Quantum Cryptography. Namely, if we consider quantum states when designing these cryptographic protocols, some computational assumptions are believed to be even weaker than OWFs. However, the author and his colleague do not know how to achieve the same QPKE protocol from such assumptions because we currently do not have concrete candidates other than OWFs for this type of task. Finally, another significant point is to verify if this quantum cryptographic protocol can achieve Unconditional Security, and interestingly, we cannot achieve such a property. Actually, there is a very simple counter-example using an attack on the model used before for the security definition, even considering an authenticated classical communication channel. In this very simple attack, the adversary would essentially keep executing the key generation algorithm until it finds a public key $pub\_key_S$ that matches the one of the sender party and proceeds with the honest cryptographic protocol. For this reason, we cannot expect this protocol to achieve better than Everlasting Security.

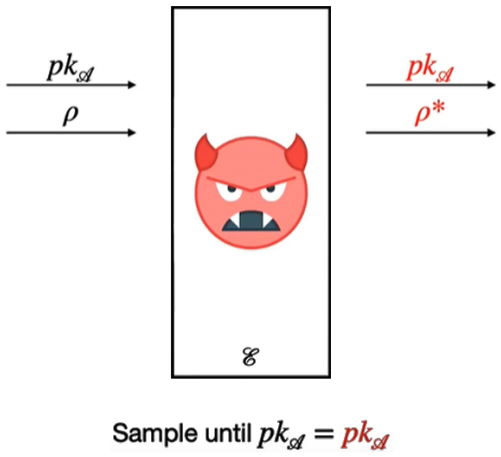


Figure 4: High-level illustration of the simple attack used to justify the reason for this Quantum Public Key Encryption (QPKE) protocol not achieving better than Everlasting Security.

## 4.2 Concurrent Work and Future Work

A concurrent work proposal from Fuyuki Kitagawa, Tomoyuki Morimae, Ryo Nishimaki, and Takashi Yamakawa, proposed recently in 2023 [17], has extremely similar constructions but results in a slightly different quantum cryptographic protocol, where the public keys are quantum, but the ciphertexts are classical. The main idea of their work proposal is the same as the one presented in this seminar, but considering only constructions of public encryption from OWFs (or weaker primitives, such as pseudo-random function-like states) [18, 19]. In particular, their cryptographic construction only achieves Computational Security as opposed to the Everlasting Security achieved with the QPKE protocol presented in this seminar. However, the quantum cryptographic protocol of this concurrent research work proposal guarantees the secrecy of the encrypted messages even if we assume only unauthenticated quantum communication channels during its execution.

One possible direction for future work that can be interesting would be to look at other cryptographic primitives beyond PKE algorithms to construct new quantum cryptographic protocols similar to the one presented in this seminar and achieve the same level of security as Everlasting Security. Some possible instances of these cryptographic primitives can be quantum variants of Identity-Based Encryption (IBE) [20–22], Attribute-Based Encryption (ABE) [20, 23–30], as well as Functional Encryption (FE) [23, 31–35], which are usually built on top of PKE primitives, at least in the classical setting.

# References


[1] Giulio Malavolta and Michael Walter. Robust Quantum Public-Key Encryption with Applications to Quantum Key Distribution, 2024.

[2] Ralph Merkle. Secure Communications Over Insecure Channels. *Commun. ACM*, 21(4):294–299, 1978.

[3] Whitfield Diffie and Martin Hellman. New Directions in Cryptography. *IEEE Transactions on Information Theory*, 22(6):644–654, 1976.

[4] Ron Rivest, Adi Shamir, and Leonard Adleman. A Method for Obtaining Digital Signatures and Public-Key Cryptosystems. *Commun. ACM*, 21(2):120–126, 1978.

[5] 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.

[6] 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.

[7] Leslie Lamport. Constructing Digital Signatures From a One Way Function. Technical Report CSL-98, Microsoft, 1979.

[8] Oded Goldreich. *Modern Cryptography, Probabilistic Proofs and Pseudorandomness*. Algorithms and Combinatorics. Springer Berlin Heidelberg, 2013.

[9] Artur Ekert. Quantum Cryptography Based on Bell's Theorem. *Phys. Rev. Lett.*, 67:661–663, 1991.

[10] Charles Bennett, Gilles Brassard, and David Mermin. Quantum Cryptography Without Bell's Theorem. *Phys. Rev. Lett.*, 68:557–559, 1992.

[11] Charles Bennett. Quantum Cryptography Using Any Two Nonorthogonal States. *Phys. Rev. Lett.*, 68:3121–3124, 1992.

[12] Yi Mu, Jennifer Seberry, and Yuliang Zheng. Shared Cryptographic Bits via Quantized Quadrature Phase Amplitudes of Light. *Optics Communications*, 123(1):344–352, 1996.

[13] Dagmar Bruß. Optimal Eavesdropping in Quantum Cryptography with Six States. *Phys. Rev. Lett.*, 81:3018–3021, 1998.

[14] Helle Bechmann-Pasquinucci and Nicolas Gisin. Incoherent and Coherent Eavesdropping in the Six-State Protocol of Quantum Cryptography. *Phys. Rev. A*, 59:4238–4248, 1999.

[15] Won-Young Hwang. Quantum Key Distribution with High Loss: Toward Global Secure Communication. *Phys. Rev. Lett.*, 91:057901, 2003.

[16] Cyril Branciard, Nicolas Gisin, Barbara Kraus, and Valerio Scarani. Security of Two Quantum Cryptography Protocols using the Same Four Qubit States. *Phys. Rev. A*, 72:032301, 2005.

[17] Fuyuki Kitagawa, Tomoyuki Morimae, and Takashi Nishimaki, Ryo and Yamakawa. Quantum Public-Key Encryption with Tamper-Resilient Public Keys from One-Way Functions, 2023.

[18] 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.

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

[20] Adi Shamir. Identity-Based Cryptosystems and Signature Schemes. In George Robert Blakley and David Chaum, editors, *Advances in Cryptology*, pages 47–53, Berlin, Heidelberg, 1985. Springer Berlin Heidelberg.

[21] Clifford Cocks. An Identity Based Encryption Scheme Based on Quadratic Residues. In Bahram Honary, editor, *Cryptography and Coding*, pages 360–363, Berlin, Heidelberg, 2001. Springer Berlin Heidelberg.

[22] Dan Boneh and Matthew Franklin. Identity-Based Encryption From the Weil Pairing. *SIAM Journal on Computing*, 32(3):586–615, 2003.

[23] Amit Sahai and Brent Waters. Fuzzy Identity Based Encryption. Cryptology ePrint Archive, Paper 2004/086, 2004.

[24] Vipul Goyal, Omkant Pandey, Amit Sahai, and Brent Waters. Attribute-Based Encryption for Fine-Grained Access Control of Encrypted Data. In *Proceedings of the 13th ACM Conference on Computer and Communications Security*, CCS '06, page 89–98, New York, NY, USA, 2006. Association for Computing Machinery.

[25] Melissa Chase. Multi-Authority Attribute Based Encryption. In Salil Vadhan, editor, *Theory of Cryptography*, pages 515–534, Berlin, Heidelberg, 2007. Springer Berlin Heidelberg.

[26] Melissa Chase and Sherman Chow. Improving Privacy and Security in Multi-Authority Attribute-Based Encryption. In *Proceedings of the 16th ACM Conference on Computer and Communications Security*, CCS '09, page 121–130, New York, NY, USA, 2009. Association for Computing Machinery.

[27] Sascha Müller, Stefan Katzenbeisser, and Claudia Eckert. On Multi-Authority Ciphertext-Policy Attribute-Based Encryption. *Bulletin of the Korean Mathematical Society*, 46(4):803–819, 2009.

[28] Allison Lewko and Brent Waters. Decentralizing Attribute-Based Encryption. In Kenneth Paterson, editor, *Advances in Cryptology – EUROCRYPT 2011*, pages 568–588, Berlin, Heidelberg, 2011. Springer Berlin Heidelberg.

[29] Taeho Jung, Xiang-Yang Li, Zhiguo Wan, and Meng Wan. Privacy Preserving Cloud Data Access with Multi-Authorities. In *2013 Proceedings IEEE INFOCOM*, pages 2625–2633, 2013.

[30] Taeho Jung, Xiang-Yang Li, Zhiguo Wan, and Meng Wan. Control Cloud Data Access Privilege and Anonymity with Fully Anonymous Attribute-Based Encryption. *IEEE Transactions on Information Forensics and Security*, 10(1):190–199, 2015.

[31] Dan Boneh, Amit Sahai, and Brent Waters. Functional Encryption: Definitions and Challenges. In Yuval Ishai, editor, *Theory of Cryptography*, pages 253–273, Berlin, Heidelberg, 2011. Springer Berlin Heidelberg.

[32] Shafi Goldwasser, Yael Kalai, Raluca Popa, Vinod Vaikuntanathan, and Nickolai Zeldovich. Reusable Garbled Circuits and Succinct Functional Encryption. Cryptology ePrint Archive, Paper 2012/733, 2012.

[33] Sergey Gorbunov, Vinod Vaikuntanathan, and Hoeteck Wee. Attribute-Based Encryption for Circuits. Cryptology ePrint Archive, Paper 2013/337, 2013.

[34] Sanjam Garg, Craig Gentry, Shai Halevi, Amit Sahai, and Brent Waters. Attribute-Based Encryption for Circuits from Multilinear Maps. In Ran Canetti and Juan Garay, editors, *Advances in Cryptology – CRYPTO 2013*, pages 479–499, Berlin, Heidelberg, 2013. Springer Berlin Heidelberg.

[35] Shafi Goldwasser, Yael Tauman Kalai, Raluca Ada Popa, Vinod Vaikuntanathan, and Nickolai Zeldovich. How to Run Turing Machines on Encrypted Data. In Ran Canetti and Juan Garay, editors, *Advances in Cryptology – CRYPTO 2013*, pages 536–553, Berlin, Heidelberg, 2013. Springer Berlin Heidelberg.