Quantum Hardware Security and Near-term Applications

Yao Ma

THÈSE DE DOCTORAT
de Sorbonne Université

Spécialité : Informatique

École doctorale n°130: Informatique, Télécommunications et Electronique de Paris

réalisée

au Laboratoire d'informatique de Sorbonne Université

sous la direction de Prof. Elham KASHEFI, Dr. Myrto ARAPINIS et Dr. Marc KAPLAN

présentée par

Yao Ma

Sujet de la thèse :

Quantum Hardware Security and Near-term Applications

soutenue le 4ème Decembre 2023

devant le jury composé de :

Prof.	Romain ALLÉAUME,	Télécom Paris,	Rapporteur
Prof.	Jean-Pierre SEIFERT,	TU Berlin,	Rapporteur
Prof.	Damian MARKHAM,	Sorbonne Université,	Examinateur
Prof.	P.W.H. PINKSE,	University of Twente,	Examinateur
Prof.	Elham KASHEFI,	Sorbonne Université,	Directrice de thèse
Dr.	Myrto ARAPINIS,	University of Edinburgh,	Directrice de thèse
Dr.	Marc KAPLAN,	Veriqloud,	Directeur de thèse

Abstract

HARDWARE security primitives are hardware-based fundamental components and mechanisms used to enhance the security of modern computing systems in general. These primitives provide building blocks for implementing security features and safeguarding against threats to ensure integrity, confidentiality, and availability of information and resources. With the high-speed development of quantum computation and information processing, a huge potential is shown in constructing hardware security primitives with quantum mechanical systems. Meanwhile, addressing potential vulnerabilities from the hardware perspective is becoming increasingly important to ensure the security properties of quantum applications.

The thesis focuses on practical hardware security primitives in quantum analogue, which refer to designing and implementing hardware-based security features with quantum mechanical systems against various threats and attacks. Our research follows two questions: How can quantum mechanical systems enhance the security of existing hardware security primitives? And how can hardware security primitives protect quantum computing systems? We give the answers by studying two different types of hardware security primitives with quantum mechanical systems from constructions to applications: Physical Unclonable Function (PUF) and Trusted Execution Environments (TEE).

We first propose classical-quantum hybrid constructions of PUFs called HPUF and HLPUF. When PUFs exploit physical properties unique to each individual hardware device to generate device-specific keys or identifiers, our constructions incorporate quantum information processing technologies and implement quantum-secure authentication and secure communication protocols with reusable quantum keys. Secondly, inspired by TEEs that achieve isolation properties by hardware mechanisms, we propose the QEnclave construction with quantum mechanical systems. The idea is to provide an isolated and secure execution environment within a larger quantum computing system by utilising secure enclaves/processors to protect sensitive operations from unauthorised access or tampering with minimal trust assumptions. It results in an operationally simple enough QEnclave construction with performing rotations on single qubits. We show that QEnclave enables delegated blind quantum computation on the cloud server with a remote classical user under the security definitions.

Résumé

LES primitives de sécurité matérielle sont des composants et des mécanismes fondamentaux basés sur le matériel et utilisés pour améliorer la sécurité des systèmes informatiques modernes en général. Ces primitives fournissent des éléments de base pour la mise en œuvre des fonctions de sécurité et la protection contre les menaces afin de garantir l'intégrité, la confidentialité et la disponibilité des informations et des ressources. Avec le développement à grande vitesse de l'informatique quantique et du traitement de l'information, la construction de primitives de sécurité matérielle avec des systèmes mécaniques quantiques présente un énorme potentiel. Parallèlement, il devient de plus en plus important de traiter les vulnérabilités potentielles du point de vue du matériel pour garantir les propriétés de sécurité des applications quantiques.

La thèse se concentre sur les primitives de sécurité matérielles pratiques en analogie quantique, qui se réfèrent à la conception et à la mise en œuvre de fonctions de sécurité matérielles avec des systèmes mécaniques quantiques contre diverses menaces et attaques. Notre recherche s'articule autour de deux questions: Comment les systèmes mécaniques quantiques peuvent-ils améliorer la sécurité des primitives de sécurité matérielle existantes? Et comment les primitives de sécurité matérielle peuvent-elles protéger les systèmes d'informatique quantique? Nous apportons les réponses en étudiant deux types de primitives de sécurité matérielle avec des systèmes mécaniques quantiques, de la construction à l'application: Physical Unclonable Function (PUF) et Trusted Execution Environments (TEE).

Nous proposons tout d'abord des constructions hybrides classiques-quantiques de PUF appelées HPUF et HLPUF. Alors que les PUF exploitent les propriétés physiques propres à chaque dispositif matériel individuel pour générer des clés ou des identifiants spécifiques, nos constructions intègrent des technologies de traitement quantique de l'information et mettent en œuvre des protocoles d'authentification et de communication sécurisés avec des clés quantiques réutilisables. Deuxièmement, inspirés par les TEE qui obtiennent des propriétés d'isolation par un mécanisme matériel, nous proposons la construction de QEnclave avec des systèmes mécaniques quantiques. L'idée est de fournir des environnements d'exécution isolés et sécurisés au sein d'un système informatique quantique plus large en utilisant des enclaves/processeurs sécurisés pour protéger les opérations sensibles d'un accès non autorisé ou d'une altération avec des hypothèses de confiance minimales. Il en résulte une construction de QEnclave assez simple de manière opérationnelle, avec l'exécution de rotations sur des qubits uniques. Nous montrons que QEnclave permet un calcul quantique délégué privé sur le serveur sur le nuage avec un utilisateur classique distant dans le cadre des définitions de sécurité.

Acknowledgements

Every time I look back on my time for the last couple of years in France, from studying engineering, then working as an engineer, and eventually being a Ph.D. student in quantum, it still seems to me like a dream. Among these moments, having the chance to be a student to explore the domain of quantum is one of the most incredible things, and one of the biggest challenges in my life. For me, it is never a one-man work, but with the generous support of my mentors, family and friends. Here, it is so grateful to have a chance to express my deepest gratitude and appreciation to the following individuals and organisations for their invaluable support, guidance, and encouragement throughout the journey of completing my thesis. Meanwhile, I apologise in advance to those I do not mention.

First and foremost, I owe a debt of gratitude to my thesis advisor, Elham Kashefi, whose expertise, dedication, and insightful feedback were instrumental in shaping this research. Your guidance has been a cornerstone of my academic development. At the same time, your enthusiasm and wisdom in research and life always inspire me to overcome different difficulties and obstacles and will benefit me for the rest of my life. Furthermore, I extend my profound gratitude to my co-supervisors, Myrto Arapinis and Marc Kaplan. Your expertise constantly enlightens me when I feel I am missing directions in research. Meanwhile, your patience and careful tutoring support me all the time throughout the whole journey of study. Every discussion and question with you is always my source of inspiration.

I also want to extend my sincere appreciation to the rest of my thesis committee members. To my reviewers, Prof. Romain Alléaume and Prof. Jean-Pierre Seifert. Meanwhile my examiners, Prof. Damian Markham and Prof. P.W.H. Pinkse. Thank you all for accepting the invitation and your dedicated time for valuable comments and feedback.

I would like to thank every talented and fantastic colleague and friend I met in the QI team and the University of Edinburgh, for their intellectual discussions, collaborative efforts, and shared experiences that contributed to my personal and academic growth. Meanwhile, your friendships, enthusiasm and colourful personality spice up my life and form an energetic and entertaining working environment, especially when my stay mostly overlaps with the period of Covid. Thank you for all the coffee breaks, happy hours, and retreats that we have together. It is hard to imagine how I could continue my thesis without these. I wish you all a great continuation, no matter what you are doing and what you believe. In particular, there are a few people whom I want to mention to express my deepest gratitude: To Mina and Kaushik, who are my co-authors and dear friends in my personal life. Your generous sharing of knowledge and inspiring discussions always enlighten me

and support me during the journey of research. Meanwhile, your warm-hearted hosting in Edinburgh makes me feel at home. Thank you for all the fantastic memories that we shared in Edinburgh (except for the time when I fell asleep in bars). To Dominik, Matilde, Paul, Santiago and Uta, whom I have countless discussions with, both on research or personal life problems (and of course Hanabi games). Most importantly, your kindness helped me go through the most terrible period in my life.

En même temps, je suis profondément reconnaissant à Veriqloud, qui est le point de départ de ma carrière professionnelle, et qui m'a fait découvrir la fantaisie du monde quantique. J'ai beaucoup de plaisir à y travailler et j'ai la chance de rencontrer de nombreuses personnes talentueuses parmi mes collègues. Votre expertise et soutien généreux m'aident à surmonter les problèmes qui se succèdent au travail. J'ai beaucoup appris et je continue à apprendre de vous tous. Je vous souhaite à tous un avenir fantastique et une vie merveilleuse. Je remercie tout particulièrement Marc pour la confiance qu'il m'a témoignée au début, en me guidant dans le monde de la quantique et en acceptant d'être mon superviseur dans la recherche. Entre-temps, j'exprime ma gratitude à Georg (Je m'excuse que mon allemand ou mon russe ne me permettent pas d'écrire quoi que ce soit pour montrer ma gratitude) et à Anne, non seulement pour leur soutien généreux au travail, mais aussi pour leur gentillesse et leur compagnie tout au long de ces années. Nous avons vécu tant de moments extraordinaires ensemble, et nous souhaitons que notre amitié soit éternelle. Par ailleurs, je suis très heureux de vous voir former un couple et d'avoir notre cher Adrian.

在国外的这些年，我还有幸认识了很多善良、热心的来自国内的朋友。他们在我人生的各个阶段都起到了举足轻重的作用。正是因为你们的帮助和分享一直支持着我走到现在，也一直让我在这一路上得到成长。我很荣幸能有机会以这样的方式向你们表示感谢，感谢你们慷慨的友谊让我这些年留下许多开心的瞬间和美好的回忆。即使在见不到你们所有人的当下，我也真挚地希望你们的生活能如你们所愿般丰富多彩！在所有的这些朋友们中，我想重点感谢我的四位朋友兼不同时期的室友，感谢你们也能在分享你们的友谊之余忍受我的缺点；何南中，我们有太多共同的美好回忆了，从南特到巴黎，感谢有你一直以来的慷慨帮忙和无私分享；李轶平，从我心中的热心好师兄，到好朋友和室友，你的善良一直在我的心中熠熠生辉，同时也是我学习的榜样；潘家俊，感谢你用你对烹饪，桌游的热情和无私分享大大丰富了我的体验，也让我以更好的状态工作生活；吴凡，感谢你从本科这么多年以来这一路的扶持，这一路的起落，你的友谊是我人生中最珍贵的部分。

在这里还要特别感谢我多年来的好友黄鹂，简子云和谢嘉衡，感谢你们从高中以来十数年如一日的友谊。即使我们不能时刻相见，也慢慢各自有了生活的烦恼，但在有限的联系中我们总能第一时间拾起往日的默契，你们给予我的鼓励是我一直以来生活的动力。望我们的友谊长存！

最后，也是最重要的，我要将我最真挚的感谢给予到我的父母。感谢你们在这段时间，乃至三十年如一日般地慷慨支持，包容和理解。除此之外，作为我生活的榜样也时时刻刻让我学会方方面面做人的道理，有你们作为我的父母是我人生中最大的幸运！在疫情阴霾过去的当下，我也希望我能更加尽到孩子的责任，陪伴在你们的身边！

Table of Contents

1 Introduction **1**
- 1.1 Quantum Technologies . . . 1
- 1.2 Hardware Security . . . 2
- 1.3 Motivation . . . 4
- 1.4 Thesis Overview . . . 5

2 Preliminaries **7**
- 2.1 Quantum Information and Quantum Computation . . . 7
  - 2.1.1 Quantum States and Dirac Notation . . . 7
  - 2.1.2 Quantum Transformations and Measurements . . . 9
  - 2.1.3 Distance Measures and Distinguishability . . . 11
  - 2.1.4 Entropy and Uncertainty Relation . . . 13
  - 2.1.5 Quantum Computation Models . . . 16
- 2.2 Cryptographic Primitives . . . 17
  - 2.2.1 One-way Function . . . 18
  - 2.2.2 Pseudorandomness . . . 19
  - 2.2.3 Cryptographic Systems . . . 21
- 2.3 Different Models of Security . . . 25
  - 2.3.1 Game-based Security Model . . . 25
  - 2.3.2 Simulation-based Security Model . . . 26
  - 2.3.3 Security in General Composition: Universal Composition and Abstract Cryptography Frameworks . . . 27

3 Hardware Security Primitives **31**
- 3.1 Introduction . . . 31
  - 3.1.1 Structure of the Chapter . . . 32
- 3.2 Security in the Real World . . . 32
  - 3.2.1 Physical Root-of-Trust . . . 34
- 3.3 Physical Unclonable Functions . . . 37

3.3.1 Classical PUF Constructions . . . 39
3.3.2 Classical PUF Modelling . . . 41
3.3.3 Quantum PUF Constructions . . . 43
3.3.4 Quantum PUF Modelling . . . 45
3.4 Trusted Execution Environments . . . 46
3.4.1 Classical TEE Constructions . . . 48
3.4.2 Classical TEE Modelling . . . 49
3.4.3 Quantum TEE Construction . . . 51
3.4.4 Quantum TEE Modelling . . . 53

4 PUFs with Provable Quantum Advantages 55
4.1 Introduction . . . 55
4.1.1 Structure of the Chapter . . . 58
4.2 Hybrid (Locked) PUF Constructions . . . 58
4.2.1 Hybrid PUF Construction . . . 58
4.2.2 Hybrid Locked PUF Construction . . . 60
4.3 Adversarial Model . . . 61
4.3.1 Unforgeability with Game-based Security . . . 62
4.4 Security Analysis . . . 64
4.4.1 Security of HPUF against Weak Adversaries . . . 64
4.4.2 Security of HLPUF against General Adaptive Adversaries . . . 72
4.4.3 Limitation of Lockdown Mechanism in Quantum Analogue . . . 75
4.5 Numeral Simulations of H(L)PUF against ML Attacks . . . 77
4.5.1 BB84 encoding with Split Attack on HPUF/HLPUF . . . 78
4.5.2 Practical Solutions for Boosting the Security . . . 82
4.6 HLPUF-based Authentication Protocol . . . 85
4.6.1 Security Analysis . . . 86
4.6.2 Challenge Reusability . . . 87
4.7 Discussion . . . 92

5 TEE-based Quantum Cloud Computing Solution 97
5.1 Introduction . . . 97
5.1.1 Structure of the Chapter . . . 99
5.2 QEnclave Constructions . . . 99
5.2.1 Abstraction of QEnclave: Remote State Rotation . . . 100
5.2.2 Practical Design with Secure Enclave . . . 101
5.3 Adversarial Model . . . 101
5.3.1 Blindness with Simulation-based Composable Security . . . 102
5.4 Security Analysis . . . 105
5.4.1 Measurement-based Remote State Preparation . . . 105
5.4.2 Perfect Blindness with RSP_B . . . 106
5.4.3 Perfect Blindness with RSR . . . 108
5.5 QEnclave-based Secure Quantum Cloud Computing . . . 110
5.5.1 QEnclave-based Blind Quantum Cloud Computing . . . 111
5.5.2 QEnclave-based Verifiable Blind Quantum Cloud Computing: Limitations and Possibilities . . . 116
5.6 Discussion . . . 118

6 Conclusion **121**
6.1 Summary 121
6.2 Future Works 122
6.3 The Ending Word 124

Bibliography **125**

List of Figures

2.1 Visualisation of an arbitrary qubit state $|\psi\rangle$ in the Bloch sphere . . 8

3.1 Role of physical root-of-trust 37
3.2 Demonstration of the concept of PUFs 38
3.3 Structure of Arbiter PUF 40
3.4 TEE with co-existing execution environments 46

4.1 HPUF construction with conjugate coding 59
4.2 HLPUF construction 60
4.3 Evolution of p_{extract} 71
4.4 QLPUF construction 76
4.5 Illustration of the measure-then-forge attack 77
4.6 LR attack performance on PUFs with BB84 encoding (4-XORPUF) 79
4.7 LR attack performance on PUFs with BB84 encoding (5-XORPUF) 79
4.8 Attack with best-performing number of CRPs on PUFs 81
4.9 HLPUF success probability of guessing 82
4.10 Comparison of LR attack performance on HLPUFs 83
4.11 LR attack performance on HLPUF with MUB d=8 (5-XORPUF) . 83
4.12 HLPUF-based authentication protocol 86

5.1 Remote state rotation resource 100
5.2 Hardware architecture of QEnclave 101
5.3 DQC ideal resource with blindness 103
5.4 RSP ideal resources for blindness 105
5.5 Measured-based RSP for blindness 106
5.6 DQC ideal resource with verifiability 116

Abbreviations

NISQ	Noisy Intermediate-Scale Quantum
QKD	Quantum Key Distribution
RNG	Random Number Generator
(C/Q)PUF	Classical/Quantum Physical Unclonable Function
(C/Q)TEE	Classical/Quantum Trusted Execution Environment
CPTP	Completely Positive and Trace-preserving
POVM	Positive Operator-Valued Measure
MBQC	Measurement-based Quantum Computing
UBQC	Universal Blind Quantum Computation
PPT/QPT	Probabilistic Polynomial Time/Quantum Polynomial Time
(P)OWF	(Physical) One-way Function
PKE	Public Key Encryption
LWE	Learning with Errors
AC/UC	Abstract Cryptography/Universal Composition
HPUF	Hybrid Physical Unclonable Function
HLPUF	Hybrid Locked Physical Unclonable Function
CRPs	Challenge-Response Pairs
BB84	The set $\{\|0\rangle, \|1\rangle, \|+\rangle, \|-\rangle\}$ of quantum states
LR	Logistic Regression
MUB	Mutual Unbiased Bases
DQC	Delegated Quantum Computation
RSP	Remote State Preparation
MRSP	Measured-based Remote State Preparation
RSR	Remote State Rotation

Chapter1

Introduction

1.1 Quantum Technologies

THE development of quantum computation and quantum information technologies enables us to use quantum mechanical systems to process information. As a multi-disciplined subject, it comprises ideas and concepts from many fields, including quantum mechanics, computer science, information theory, cryptography, etc. In recent years, quantum computation and quantum information technologies have made significant strides with ongoing research, development, and practical applications in various domains.

When we trace back to the early twentieth century when the theory of quantum mechanics was created and built up, the description of the physical systems within the framework of quantum mechanics (superposition, entanglement, etc.), especially at the level of atoms and subatomic particles, revolutionized our understanding. However, people could not achieve complete control over single quantum systems, like a photon or an atom, until the 1970s. And it is indeed a fundamental requirement for quantum computation and quantum information technologies.

The idea of quantum computation was first proposed by Richard Feynman in 1982 [1]. It comes from the difficulties of simulating quantum mechanical systems efficiently with classical computers. Therefore, he suggested using quantum systems to simulate themselves would possibly avoid these difficulties, that is, to build a quantum computer ruled by the principles of quantum mechanics. Almost around the same time, inspired by the model of the universal Turing machine proposed by Alan Turing [2], Paul Benioff described a quantum mechanical model of Turing machine [3, 4], then formalised by David Deutsch in [5] to define a computational device that can simulate arbitrary physical system efficiently.

As Deutsch's model is recognised to be the model of quantum computers nowadays by many works and research, Shor's algorithm and Grover's algorithm are the two most representative ones. Shor's algorithm [6] showed that the mathematical problems of factoring and discrete logarithms, which are widely believed to have no efficient solution on classical computers, can be solved efficiently by a quantum computer. Grover's algorithm [7] showed a quadratic speedup of searching an unsorted database with quantum computers over classical search algorithms.

In practice, people do not stop putting effort into building a quantum computer. One main difficulty is that most useful quantum algorithms require a large-scale quantum computer with delicate control over the system. Still, there are construc-

tions of photonic quantum computers by USTC in China [8] and superconducting quantum computers by Google, IBM, etc. [9, 10], which achieve quantum advantages computationally. Also, regarding near-term practicality, the terminology NISQ (Noisy Intermediate-Scale Quantum) describes the devices that control the orders of 10s to 100 noisy qubits [11]. They can execute some limited quantum programs but allow researchers to explore potential quantum algorithms and applications for more powerful quantum computers in the future.

On the other hand, the development of quantum information theory was ongoing at almost the same time. Apart from the strong relevance of quantum computation and quantum information, quantum information theory also paves the way for the development of quantum cryptography. Recall that the information theory formalised by Claude Shannon [12] uses the terminology entropy, originally from thermodynamics, to measure in average the amount of information or uncertainty associated with the outcomes of a random process. High entropy sources, e.g., random number generators, ensure the unpredictability and security of cryptographic keys, which further secure information transmitted via communication channels involving two or more parties from different perspectives.

Unlike public key cryptosystems, an essential prerequisite for private key cryptosystems is to distribute the keys securely under the threat of eavesdropping in the regime of classical information. In this case, the idea of quantum key distribution (QKD) [13] is proposed to tackle this issue with the help of quantum information technology. It leverages the principles of quantum mechanics (no-cloning theorem, uncertainty principle, etc.) to enable the exchange of cryptographic keys between two parties in a way that is information-theoretic security against any form of eavesdropping, even by quantum computers. On the other hand, the widely-used public key cryptosystems (RSA, Diffie-Hellman, etc.), which rely on the hardness of either factoring or discrete logarithm problems solved by classical computers, are no longer secure in the presence of quantum computers in the future.

1.2 Hardware Security

In the 1940s, the design and construction of the first generation of general-purpose computers started the modern computing era. Since the wide usage of transistors, modern computing hardware devices have been through an incredibly high-speed pace of development over the last 80 years. A significant statement on this is Moore's law [14], which states that the number of transistors on microchips, as well as the computational power, doubles every two years. Nowadays, modern computing hardware devices facilitate people from every aspect of life.

However, with the increment in the complexity of devices and computing systems, security issues are becoming increasingly non-negligible. The threats can come from every perspective of the device's life cycle on both software and hardware levels. These threats can not only sabotage the proper running of the programs but also affect the security properties of data and applications. Among the threats, many of them are due to the flaws and vulnerabilities existing in hardware [15]. Therefore, more and more work from academia and industry focuses on developing hardware security from specification to application to investigate efficient

countermeasures and circumvent threats.

Understanding different hardware security properties is an important step in investigating efficient constructions in the first place. In brief, we obtain hardware security properties by the formal specifications on hardware behaviours related to security. Well-studied hardware security properties include confidentiality, integrity, availability, isolation, randomness, etc. Compared to the same terminologies in cryptographic primitives, hardware security properties provide similar security guarantees by establishing a practical secure foundation from different *hardware security primitives*. It allows the construction of functionalities on computing systems where hardware security primitives can cover the necessary security properties.

The construction and usage of hardware security primitives can be traced back to the birth of modern computers. Since the 1940s, the electronic noise generated by vacuum tubes and semiconductors has been exploited to generate random data [16], and people used this noise as the source of randomness in early computers. Furthermore, with the development of electronic engineering, a certain type of devices is designed to produce genuinely unpredictable and unbiased random numbers, making them valuable in various applications, particularly in cryptography and secure systems called true random number generators (TRNGs), also known as hardware random number generators (HRNGs). HRNGs exploit physical processes that are inherently random as sources of entropy to generate random numbers. Such physical processes include electronic noise, thermal noise, radioactive decay, etc. For cryptography systems in general, since a high entropy source plays a fundamental role in guaranteeing the security of cryptographic systems, HRNGs are one of the most well-known hardware security primitives in modern computing systems.

Also, other hardware security primitives with different security properties guaranteed by hardware are designed and widely exploited in modern computing systems, e.g., Physical unclonable functions (PUFs) and trusted execution environments (TEEs). These primitives can either provide specific properties from the behaviours of the hardware devices that people can exploit or secure mechanisms with hardware-based techniques to provide systems' security.

Meanwhile, security verification methodologies are necessary to verify the hardware security designs. Theoretically, we perform the verification process by providing frameworks for creating a mathematical model of the system, specifying the security properties, and formally proving whether the model complies with the properties or not by rigorous proofs. Alternatively, the security properties can be expressed as invariants within a system, and their validity is checked against all possible execution paths [17]. In practice, there exist different commercial tools that provide security verification, for example, Questa SecureCheck [18], JasperGold Security Path Verification [19], etc. Recently, there have been researches aiming to develop security-driven hardware design tools in order to evaluate the security along with the traditional design parameters on the hardware level [20, 21, 22, 23].

1.3 Motivation

In the thesis, we aim to study different hardware security primitives in the context of properties of quantum mechanical systems with near-term available quantum computation and quantum information techniques, as well as the corresponding applications in the real world. As we have known previously, the properties of quantum mechanical systems have significant implications for security properties by enabling cryptographic threats, as well as innovative tools and methods for securing data and communication. Intuitively, it is worth developing further the hardware security primitives with an extension of near-term quantum computation and quantum information technique to enable the unique properties of quantum mechanics in hardware security designs.

A well-known and practical quantum hardware security primitive is quantum random number generators (QRNGs), which exploit the inherent randomness of quantum phenomena, including photon polarization, vacuum fluctuations, entanglement, etc., to generate random numbers. With the advancement of the works of quantum information theory, QRNGs offer more substantial support over some other HRNGs with classical processes in theory. We will give a more detailed introduction later in the thesis. Commercially, companies like ID Quantique (IDQ), Toshiba, etc. [24, 25] offer sophisticated QRNG devices to meet the growing demand for secure and truly random numbers in various applications.

However, the growth of other hardware security primitives in quantum analogue is relatively slow compared to QRNGs, for example, the quantum physical unclonable functions (QPUFs). By reviewing the works later in our thesis, another obstacle to developing QPUFs, or other quantum hardware security primitives, is the issue of implementations with current hardware devices, mainly due to the limitations of quantum instruments' performance. As a result, another important criterion of our proposed designs, constructions and applications related to possibly different quantum hardware security primitives throughout the thesis is practicality.

We drive our ideas on quantum hardware security primitives introduced in the thesis by asking the following questions:

***Question* 1.** *"How can quantum mechanical systems enhance the security of existing hardware security primitives?"*

, and

***Question* 2.** *"How can hardware security primitives protect quantum computing systems?"*

To answer Question 1, we study the available hardware security primitives in modern computing systems with classical processes and input/output data. By analysing the constraints and possible security threats covered by the adversarial model we are concerned with, we aim to circumvent the issues and provide extra security guarantees in different use cases by possibly exploiting the properties of quantum mechanical systems. On the hardware level, the potential constructions should be implementable and practical with near-term classical/quantum hardware techniques.

Regarding Question 2, we consider that near-term quantum computing devices will be mostly available on a remote side by a cloud server due to the expense and use cases. In this sense, different classical hardware security primitives are utilised to protect the computation by satisfying different security properties dependently in classical scenarios. In the context of quantum cloud computing scenarios with different setups and adversarial behaviours, we expect that hardware security primitives can play the same role with a necessary extension of quantum mechanical systems while retaining reasonable designs and constructions to implement and utilise. It might seem too early to discuss security primitives based on hardware construction in quantum computing architecture, especially due to unsophisticated constructions and the limited computing power of quantum computation devices nowadays. We expect that our ideas can heuristically recall the attention on hardware security designs and constructions in building future quantum computers.

1.4 Thesis Overview

Here, we give a brief introduction to the structure of the thesis for a general overview of the content in each chapter:

- In Chapter 2, we introduce the significant preliminaries and necessary mathematical frameworks and toolkits that are tightly related to our work. This chapter is composed of two parts: The first part introduces the notions and techniques that we commonly use in the research of quantum information and quantum computation. The second part includes the basics of cryptographic primitives and different security models. They form together to derive our novel ideas for constructions, security analyses, and applications that we develop further in the thesis.

- In Chapter 3, we generally discuss the significance of security on the hardware level of modern computing systems, which results in the research of hardware security primitives in classical and quantum. Specifically, we instantiate three main different hardware security primitives: Hardware Random Number Generator (HRNG), Physical Unclonable Function (PUF) and Trusted Execution Environment (TEE), which provide specific cryptographic security properties, respectively. Recall Question 1 and 2 that we proposed previously, we focus on PUFs and TEEs classically and in quantum by introducing and comparing different state-of-the-art constructions with their advantages and drawbacks. We formalise with mathematical models to capture their cryptographic properties. A partial description of this chapter can be found in the following works: Quantum Lock: A Provable Quantum Communication Advantage [26], and QEnclave - A practical solution for secure quantum cloud computing [27], respectively.

- In Chapter 4, we answer to Question 1 by studying PUFs within our work: Quantum Lock: A Provable Quantum Communication Advantage [26]. We propose a generic design of provably secure PUFs, called hybrid locked PUFs (HLPUFs), providing a practical solution for securing classical PUFs (CPUFs).

In brief, an HLPUF uses an underlying CPUF and encodes the output into non-orthogonal quantum states, together with the lockdown technique, to prevent the adversary from accessing the structured information of the outcome of CPUF. Moreover, we show that by exploiting non-classical properties of quantum states, the HLPUF allows the server to reuse the challenge-response pairs for further client authentication. Later, we support our theoretical contributions by instantiating the HLPUF design using accessible real-world CPUFs via simulation in different scenarios. We use the optimal modelling attacks to forge both the CPUFs and HLPUFs, and we certify the security gap in our numerical simulation for construction, which is ready for implementation. This result provides an efficient solution for running PUF-based client authentication for an extended period while maintaining a small-sized challenge-response pairs database on the server side.

- In Chapter 5, we answer to Question 2 by TEEs within our work: QEnclave - A practical solution for secure quantum cloud computing [27]. Inspired by the TEE mechanism, we introduce a secure hardware design named a QEnclave that extends to quantum computing from the classical concept of a secure enclave that isolates a computation from its environment to provide privacy and tamper-resistance. To minimise the trust assumption for practicality, our QEnclave only performs single qubit rotations, ideally described by remote state rotation (RSR) functionality. We show it is sufficient to secure an arbitrary quantum cloud computation in terms of privacy with a classical client, even if an adversary controls the qubit source. An immediate consequence is the weakening of requirements for blind delegated computation with the same secure guarantees, while previous delegated protocols relied on a client that can either generate or measure quantum states.

- In Chapter 6, we conclude our works on developing hardware security primitives in the quantum era from two different directions. Meanwhile, we discuss the related work that can be done in the future.

Chapter2

Preliminaries

In this chapter, we aim to provide the necessary backgrounds, concepts, toolkits and notations used in our thesis. In general, it can be divided into two parts: In the first part, we introduce the preliminaries of quantum information and quantum computation technologies, including the necessary materials of quantum information theory we need to describe different quantum properties within our proposed quantum hardware security primitives. The second part covers the preliminaries of cryptography and different security models, which provides us with formal verification tools to define and analyse the cryptographic security properties of different quantum hardware security primitives that we propose in our thesis.

2.1 Quantum Information and Quantum Computation

We start by introducing the background of quantum information and quantum computation. We assume some familiarity with linear algebra, whereas familiarity with quantum mechanics beforehand is not necessarily required to understand the content of the thesis.

2.1.1 Quantum States and Dirac Notation

In quantum mechanics, a *quantum state* is a mathematical description that characterises the quantum properties of a physical system[1]. Quantum states are typically represented as vectors in a complex vector space, which is a *Hilbert space* $\mathcal{H}$, often called a *Hilbert space* $\mathcal{H}$, and thus a Hilbert space of dimension d by $\mathcal{H}^d$. For a d-dimensional Hilbert space, any vector can be expressed as a linear combination of d orthonormal *basis vectors*.

In quantum information and quantum computation technologies, a *qubit* is a quantum system analogous to a classical bit. It lives in a two-dimensional Hilbert space $\mathcal{H}$. In particular, the *computational basis* of a qubit in $\mathcal{H}^2$ is denoted as:

$$|0\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \qquad |1\rangle = \begin{bmatrix} 0 \\ 1 \end{bmatrix}. \tag{2.1}$$

[1]Indeed, a quantum state is supposed to encompass the properties of a physical system not only limited to quantum ones, but we care more about the quantum properties in the thesis.

By using the *ket* notation $|\cdot\rangle$, together with the *bra* notation $\langle\cdot|$ as the dual or conjugate, we obtain the *Dirac notations* that are widely used in quantum.

For a quantum system that can be represented by a vector in the Hilbert space, we describe its state deterministically, and therefore we call it a *pure state*. It can be written as a liner combination of the basis, e.g., $|\psi\rangle = \alpha\,|0\rangle + \beta\,|1\rangle$ with $\alpha, \beta \in \mathbb{C}$, where $|\alpha|^2 + |\beta|^2 = 1$ due to the normalisation. We call such a linear combination a *superposition* of the basis states (other quantum states). Particularly, we introduce the quantum states $|+\rangle$ and $|-\rangle$ with superposition of computational basis with equal coefficients:

$$|+\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) = \frac{1}{\sqrt{2}}\begin{bmatrix}1\\1\end{bmatrix}, \qquad |-\rangle = \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle) = \frac{1}{\sqrt{2}}\begin{bmatrix}1\\-1\end{bmatrix}. \tag{2.2}$$

They also form the basis of $\mathcal{H}^2$ due to their orthonormality, which we call X-basis or *Hadamard* basis. Meanwhile, we can also describe a pure quantum state by its *density matrix* $\rho = |\psi\rangle\langle\psi|$ with the *outer product* of the vector with itself.

Relatively, some quantum systems are represented by a probability distribution over different pure quantum states, so-called *mixed states* with the representation as follows:

$$\rho = \Sigma_s p_s |\psi_s\rangle\langle\psi_s|. \tag{2.3}$$

A density matrix is a Hermitian operator of trace one, i.e. $\mathrm{tr}(\rho) = 1$, acting on the Hilbert space of the system. While a pure state always satisfies $\mathrm{tr}(\rho^2) = 1$, $\mathrm{tr}(\rho^2) < 1$ for mixed states.

For two pure states $|\psi\rangle, |\phi\rangle$, the bra-ket product $\langle\psi|\,\phi\rangle$, also known as *inner product*, is a fundamental operation in the Dirac notation. It involves taking the complex conjugate of a bra and multiplying it by a ket, yielding a complex number. It is often used to calculate probabilities and overlaps between quantum states.

The *Bloch sphere*, on the other hand, is a geometric representation used to visualise possible pure and mixed states of a single qubit as shown in Figure 2.1 , where all possible pure states are on the surface of the sphere, and the mixed

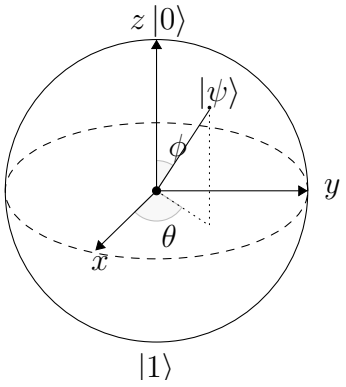


Figure 2.1: Visualisation of an arbitrary qubit state $|\psi\rangle$ in the Bloch sphere

states are inside the sphere. For $0 \le \theta \le 2\pi$ and $0 \le \phi \le \pi$ in the Bloch sphere, we describe an arbitrary pure state as:

$$|\psi\rangle = \cos\frac{\psi}{2}\,|0\rangle + e^{i\theta}\sin\frac{\psi}{2}\,|0\rangle \tag{2.4}$$

To describe a joint state $|\psi_A\psi_B\rangle$ in Hilbert spaces $\mathcal{H}_A$ and $\mathcal{H}_B$ of two quantum systems, we say the state is *separable* if the state can be written as a convex combination of product states $|\psi_A\psi_B\rangle = \sum_i p_i |\psi_A^{(i)}\rangle \otimes |\psi_B^{(i)}\rangle$ (or simply $|\psi_{AB}\rangle$) in $\mathcal{H}_A \otimes \mathcal{H}_B$. Meanwhile, there are other states that can not be written in the composition of tensor product form. These states are, therefore, non-separable and called *entangled* states, which are associated with the phenomenon of quantum entanglement.

Note that one of the most important bipartite entangled states is *Bell states*, which are:

$$|\phi^{\pm}\rangle = \frac{1}{\sqrt{2}}(|0\rangle_A \otimes |0\rangle_B \pm |1\rangle_A \otimes |1\rangle_B)$$
$$|\psi^{\pm}\rangle = \frac{1}{\sqrt{2}}(|0\rangle_A \otimes |1\rangle_B \pm |1\rangle_A \otimes |0\rangle_B). \tag{2.5}$$

Here, the subsystems A and B are correlated and cannot be described independently of each other. In this case, to describe the quantum state of a subsystem from an entangled state of a joint system, we use the *partial trace* on one of the other subsystem to get the quantum state of the targeted subsystem, with the density matrix representation of the quantum states as:

$$\rho_A = \mathrm{tr}_B(\rho_{AB}), \qquad \rho_B = \mathrm{tr}_A(\rho_{AB}). \tag{2.6}$$

Here, ρ_A and ρ_B are called *reduced density matrices*. For Bell states, the reduced density matrix of subsystems A and B results in the density matrix $\frac{I}{2}$, which are known as *maximally mixed states*, which are often associated with a lack of knowledge or structured information about the quantum system.

2.1.2 Quantum Transformations and Measurements

The transformation of a closed quantum system can be described by a *unitary* operation on a pure state $|\psi\rangle$ or mixed state ρ as:

$$|\psi'\rangle = U|\psi\rangle \qquad \rho' = U\rho U^{\dagger} \tag{2.7}$$

There are a few examples of unitary operators, which are important in quantum computation and quantum information. We use the letters $X/Y/Z$ to denote some particular unitary operators called *Pauli operators*. For single-qubit, the Pauli-X/Y/Z operators, as well as identity I, are:

$$\mathrm{I} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \mathrm{X} = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \mathrm{Y} = \begin{pmatrix} 0 & \mathrm{i} \\ -\mathrm{i} & 0 \end{pmatrix}, \mathrm{Z} = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}. \tag{2.8}$$

Another important operator is the *Hadamard* (H), which maps the computational basis to the Hadamard basis with the following matrix:

$$\mathrm{H} = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}. \tag{2.9}$$

Meanwhile, a *phase shift* operator is a family of single-qubit gates that map the computational basis states $|0\rangle$ to $|0\rangle$, and $|1\rangle$ to $e^{i\theta}|1\rangle$. It is equivalent to performing

a rotation about the z-axis on the Bloch sphere by θ radians. The following matrix represents the phase shift operator:

$$\mathrm{Z}(\theta) = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\theta} \end{pmatrix}. \tag{2.10}$$

For two-qubit systems, there are unitary operators called *controlled-U*, which use two qubits as input: a control qubit and a target qubit. It operates the unitary U on the target qubit when the control qubit is set to $|1\rangle$, e.g., the *controlled-X* (CNOT) and *controlled-Z* (CZ) operators are

$$\mathrm{CNOT} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}, \qquad \mathrm{CZ} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 \end{pmatrix}. \tag{2.11}$$

A general map between two quantum states is a quantum channel $\mathcal{E}$, which is a *completely positive and trace-preserving* (CPTP) map. It can always be described as a linear combination of *Kraus operators* $\{E_k\}$ as $\mathcal{E} = \Sigma_k E_k \rho E_k^\dagger$, where $\{E_k\}$ can be written as $E_k = \Sigma_i \alpha_i \sigma_i$. Here, α_i is a complex number, and σ_i is a Pauli operator. Alternatively, the quantum channels can be described by a unitary operation on an expanded Hilbert space that includes the system of environment $|E\rangle$ as:

$$\rho' = \mathcal{E}(\rho) = \mathrm{tr}_E[U(\rho \otimes |E\rangle\langle E|)U^\dagger]. \tag{2.12}$$

Quantum channels can model the effect of quantum noise. The latter description shows how any noise can be seen as an interaction of the quantum system with the environment. This helps us analyse the effect of quantum noise on the target system the same way we describe any other transformations. For example, the Pauli-X and Pauli-Z matrices are also referred to as the bit flip noise and phase flip noise. Meanwhile, there are other important and widely-studied quantum noise effects on the quantum system [28].

Quantum measurements are fundamental processes in quantum mechanics that allow us to extract information from quantum systems. The measurement of a quantum state is defined by a set of operators $\{M_i\}$ satisfying $\sum_i M_i^\dagger M_i = I$ with its conjugate transpose operators $M_i^\dagger$. The probability of getting measurement result i on quantum state $|\psi\rangle$ is

$$p(i) = \langle\psi| M_i^\dagger M_i |\psi\rangle, \tag{2.13}$$

where $\langle\psi| M_i^\dagger M_i |\psi\rangle = \langle\psi| M_i |\psi\rangle$ if and only if M are projective and hermitian. And the state of the system after the measurement is:

$$|\psi_i\rangle = \frac{M_i |\psi\rangle}{\sqrt{\langle\psi| M_i^\dagger M_i |\psi\rangle}}. \tag{2.14}$$

For a general mixed state ρ, the probability is given as:

$$p(i) = \mathrm{tr}(M_i \rho). \tag{2.15}$$

The above calculation of probabilities of measurement outcomes in quantum systems is described by Born's rules.

An important special class of measurement are *projective measurements*, where we measure a quantum state in a certain basis, e.g., performing the measurement on quantum state ρ in the computational basis, we project ρ onto the eigenstate $|0\rangle$ or $|1\rangle$ by projection operators $|0\rangle\langle 0|$ and $|1\rangle\langle 1|$, with the measurement outcome 1 or -1 as eigenvalues of the *observable* Pauli-Z. Here, we can easily calculate the *expectation value* of Z in the state ρ by

$$\langle Z\rangle_\rho = \mathrm{tr}(Z\rho) \tag{2.16}$$

A more general measurement is given by the mathematical tool known as the *Positive Operator-Valued Measure* (POVM) formalism. A POVM is described by a set of positive operators $\{E_i\}$ satisfying $\sum_i E_i^\dagger E_i = I$, and the probability of outcome i on quantum state $|\psi\rangle$ is given by

$$p(i) = \langle\psi| E_i |\psi\rangle = \mathrm{tr}(E_i|\psi\rangle\langle\psi|) \tag{2.17}$$

However, unlike projective measurements, the operators are not necessarily orthogonal. Thus, the cardinality of the set can be greater than the dimension of Hilbert space of the state to be measured. It allows capturing measurements with continuous outcomes, overlapping outcomes, and more.

2.1.3 Distance Measures and Distinguishability

Distance measures of classical and quantum information plays an important role in our thesis. In classical information theory, we use distance measures expressed by $\mathrm{Dist}(\cdot,\cdot)$ to quantify the dissimilarity or difference between two values, probability distributions, sets, data points, etc. For example, for two values x and y, we use $\mathrm{Dist}(x,y)$ to denote the distance between x and y according to some metrics, e.g., the *Hamming distance*. However, many of the methodologies for distance measures in classical information theory can not be exploited directly in quantum information. In cooperating with the nature of quantum mechanics, *trace distance* and *fidelity* are the two most common distance measures techniques utilised in the community of quantum computation and quantum information.

We start by introducing the distance measures for classical probability distributions, in contrast to the probability distribution over different quantum states in the Hilbert space of the quantum system. For two probability distributions $\{p_x\}$ and $\{q_x\}$, we compare them by the following distance measure called trace distance,

$$\mathrm{Dist}(p_x, q_x) := \frac{1}{2}\Sigma_x|p_x - q_x|\,. \tag{2.18}$$

In quantum analogue, it generalises the classical notions of trace distance. The formal definition is given as follows:

Definition 2.1 (Trace distance)**.** *For any two quantum states ρ and σ, the trace distance is as below*

$$D(\rho,\sigma) := \frac{1}{2}\|\rho-\sigma\|_1 = \frac{1}{2}tr\,|\rho-\sigma|\,, \tag{2.19}$$

where $|\rho-\sigma| := \sqrt{(\rho-\sigma)^\dagger(\rho-\sigma)}$ *is defined to be the positive square root of* $(\rho-\sigma)^\dagger(\rho-\sigma)$*. The two states are identical if and only if* $D(\rho,\sigma)=0$*. As the value increases, the states become more distinguishable.*

Another common distance measure for quantum states is the so-called fidelity, even if it is not a metric on density matrices. The fidelity between states ρ and σ is defined by the Uhlmann fidelity [28] as follows:

Definition 2.2 (Uhlmann fidelity [28])**.** *For any two quantum state* ρ *and* σ*, the Uhlmann fidelity is defined as:*

$$F(\rho,\sigma) = [tr(\sqrt{\sqrt{\rho}\sigma\sqrt{\rho}})]^2. \tag{2.20}$$

When $F(\rho,\sigma)=1$*, it means that the states are identical. As the fidelity decreases from 1 to 0, the states become less similar.*

For many cases, the trace distance and the Uhlmann fidelity are equivalent. In general, they are related by the following inequality:

$$1-F(\rho,\sigma) \leq D(\rho,\sigma) \leq \sqrt{1-F(\rho,\sigma)^2}. \tag{2.21}$$

Distance measures are essential in the *distinguishability* of quantum states. Recall that an important property of the quantum states is the impossibility of creating perfect copies of general unknown quantum states, known as the *no-cloning theorem* [29]. This is an important limitation imposed by quantum mechanics, which is particularly relevant for cryptography. A variation of the same feature makes it impossible to obtain the exact classical description of quantum states by having a single or very few copies. Therefore, there exists a bound on how much classical information can be extracted from quantum states, known as Holevo bound [30]. As a tightly co-related problem, distinguishing between two unknown quantum states is also a probabilistic procedure known in the quantum information literature as *quantum state discrimination*. The goal is to distinguish ρ by performing the optimal POVM and getting the minimum error probability, where it is bounded by their distance, known as *Holevo-Helstrom bound* as follows:

Definition 2.3 (Holevo-Helstrom bound)**.** *The best success probability to discriminate between two (mixed) states represented by* ρ *and* σ*, which are given uniformly at random with probability* $\frac{1}{2}$*, can be denoted by:*

$$Pr_{guess}^{opt} = \frac{1}{2}(1+D(\rho,\sigma)) = \frac{1}{2}(1+\frac{1}{2}\|\rho-\sigma\|_1). \tag{2.22}$$

Complementary to distinguishability, we consider two quantum states equal if their trace distance vanishes. Here, we use the expression $\texttt{Ver}(\cdot,\cdot)$ for checking the equality of any two quantum states by distance measures as a CPTP map $\texttt{Ver} : \mathcal{H}^d \otimes \mathcal{H}^d \to \{0,1\}$. For two quantum states ρ and σ the CPTP map $\texttt{Ver} : \mathcal{H}^d \otimes \mathcal{H}^d \to \{0,1\}$ measures the equality as

$$\texttt{Ver}(\rho,\sigma) := \begin{cases} 1 & \text{if } \|\rho-\sigma\|_1 \leq \epsilon, \\ 0 & \text{otherwise.} \end{cases} \tag{2.23}$$

This general verification also includes measurements of quantum states as verification algorithms since it has been defined as a general CPTP map. One example of such an algorithm that implements the verification map is the *SWAP test* [31]. The swap test's circuit uses the controlled version of a swap operator, where

$$\text{SWAP} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \tag{2.24}$$

that swaps the order of two quantum states if the control qubit is $|1\rangle$. For two pure states $|\psi\rangle$ and $|\phi\rangle$, one can calculate that the circuit outputs $|0\rangle$ with probability $\frac{1}{2} + \frac{1}{2}F(|\psi\rangle, |\phi\rangle)$ and it outputs $|1\rangle$ with probability $\frac{1}{2} - \frac{1}{2}F(|\psi\rangle, |\phi\rangle)$. As can be seen, the success probability of this test depends on the fidelity of the states.

Finally, the *diamond norm* is used to quantify the distance between quantum channels, with the following definition:

Definition 2.4. *For any two quantum channels $\mathcal{E}_1$ and $\mathcal{E}_2$, which are CPTP maps, the diamond norm is defined as:*

$$\|\mathcal{E}_1 - \mathcal{E}_2\|_\diamond := \max_\rho \|(\mathcal{E}_1 \otimes I)\rho - (\mathcal{E}_2 \otimes I)\rho\|_1, \tag{2.25}$$

with the maximum being taken over all the density matrices ρ.

2.1.4 Entropy and Uncertainty Relation

As a fundamental concept, *entropy* plays a central role in various aspects of information theory to measure the average amount of information or uncertainty of the variable's outcomes. For classical information, we also referred to the entropy as *Shannon entropy*, with the formal definition:

Definition 2.5 (Shannon entropy)**.** *Let X be a discrete random variable on a finite set $\mathcal{X} = \{x_1, \ldots, x_n\}$ with probability distribution function $p(x) = Pr(X = x)$. The entropy $H(X)$ of X is defined as:*

$$H(X) = -\Sigma_{x\in\mathcal{X}} p(x) \log p(x) = -E[\log p(x)]. \tag{2.26}$$

The logarithm is usually taken to base two, where the entropy is measured in bits.

In quantum information theory, we use the *Von Neumann entropy* to the amount of uncertainty or information contained in a quantum state ρ with the definition as follows:

Definition 2.6 (Von Neumann entropy)**.** *For a quantum mechanical system described by a density matrix ρ, the Von Neumann entropy is defined as:*

$$H(\rho) = -tr(\rho \log \rho) = -\Sigma_i \lambda_i \log(\lambda_i). \tag{2.27}$$

Here, λ_is are the eigenvalues of ρ.The logarithm is usually taken to base two. $H(\rho)$ is 0 for any pure quantum state, indicating no uncertainty or randomness.

According to the notion of entropy in classical and quantum information theory, we introduce an important and useful mathematical tool that we use in our thesis. It is called *entropic uncertainty relations*, which includes a mathematical framework with several inequalities. More recently, entropic uncertainty relations have emerged as the central ingredient in the security analysis of almost all quantum cryptographic protocols, such as QKD and two-party quantum cryptography [32].

Heisenberg's uncertainty principle is one of the most important fundamental properties of quantum mechanics, which, mathematically speaking, is due to the non-commuting property of some observables like Pauli X and Z measurements. Reformulating these relations in terms of entropic quantities has been very useful in the foundations of quantum information. It has also been widely used in the security proofs of quantum communication protocols, such as QKD. The most well-known uncertainty relation for these operators was given by Deutsch [33] and later improved [34] as follows:

$$H(X) + H(Z) \geq \log_2(\frac{1}{c}), \tag{2.28}$$

where c denotes the maximum overlap between any two eigenvectors of Pauli X and Z.

We consider a quantum system A where the state is described with the density matrix ρ_A on a finite-dimensional Hilbert space. If the measurement is performed in the Pauli X or Pauli Z basis, the measurements are projective operators that project the state into the subspace spanned by those bases. Here, the random variables that are associated with the entropy are defined by the measurement outcomes of observables X and Z. In the most general case, the measurements are a set of POVM operators $\{M^x\}_x$ and $\{N^z\}_z$ on system A. The probability of obtaining outcomes x and z is, by the Born rule,

$$P_X(x) = tr[\rho_A M^x] \quad , \quad P_Z(z) = tr[\rho_A N^z] \tag{2.29}$$

In this case, Equation (2.28) still gives the generalised uncertainty relation with the difference that c is defined as follows:

$$c = \max_{x,z} c_{zx}, \quad \text{and} \quad c_{xz} = \| \sqrt{M^x}\sqrt{N^z} \|^2, \tag{2.30}$$

where $\| \cdot \|$ denotes the operator norm (or infinity norm). The above uncertainty relation can be extended to conditional entropy as well in the context of guessing games [32]. Assume two parties, Alice and Bob, where Bob prepares a state ρ_A, and Alice randomly performs the X and Z measurements leading to a bit K. Then Bob wants to guess K given the basis choice $R = \{0, 1\}$. The conditional Shannon entropy is defined as follows:

$$H(K|R) := H(KR) - H(R) \tag{2.31}$$

Thus, one can get the same uncertainty relation with the conditional entropy as:

$$H(K|R = 0) + H(K|R = 1) \geq \log_2(\frac{1}{c}) \tag{2.32}$$

Similar to the classical case, for a bipartite system ρ_{AB} the conditional Von Neumann entropy is defined as follows:

$$H(A|B) := H(\rho_{AB}) - H(\rho_B) \tag{2.33}$$

Furthermore, this can be generalised to a tripartite quantum system with state ρ_{ABC}. An interesting property here is an inequality referred to as *data processing inequality* [32], which states that the uncertainty of A conditioned on some system B never goes down if B performs a quantum channel on the system. In other words, for any tripartite system ρ_{ABC} where system C will perform a quantum operation on the quantum state in order to extract some information, we have the following:

$$H(A|BC) \leq H(A|B) \tag{2.34}$$

The above inequality leads to the general uncertainty relations between any tripartite system, including two honest parties, Alice and Bob, and an eavesdropper, Eve. In this case, the following entropic inequality holds:

$$H(K|ER) + H(K|BR) \geq \log_2 \left(\frac{1}{c}\right), \tag{2.35}$$

where K is the measurement output and R is the basis bit. This imposes a fundamental bound on the uncertainty in terms of Von Neumann entropy, in other words, the amount of information that an eavesdropper can extract from the joint quantum systems shared between the three parties. These inequalities can also be extended to the case where n bits are encoded in n quantum states where R^n and K^n are bit-strings denoting the basis random choices for the qubits and measurement outputs respectively, and B^n denotes Bob's bit-string. Also, E denotes Eve's system, a general quantum system operating on n-qubit messages and any arbitrary local system. We have the following inequality:

$$H(K^n|ER^n) + H(K^n|B^nR^n) \geq n \log_2(\frac{1}{c}) \tag{2.36}$$

Meanwhile, the entropic uncertainty relations can also be exploited to a broader perspective on information content and uncertainty that is described by entropies. Here, we employ *Rényi entropy* [35] as a family of entropy measures in information theory that generalises different aspects of uncertainty and information content. With an adaptation to the quantum mechanical system, we introduce the definitions of min- and max-entropy from [36, 37] as follows:

Definition 2.7. *Let $\rho = \rho_{AB}$ be a bipartite density operator. The min-entropy of A conditioned on B is defined by:*

$$H_{\min}(A|B)_\rho := -\inf_{\sigma_B} D_\infty(\rho_{AB} \| I_A \otimes \sigma_B), \tag{2.37}$$

where the infimum ranges over all normalised density operators σ_B on subsystem B and where

$$D_\infty(\tau \| \tau') := inf\{\lambda \in \mathbb{R} : \tau \leq 2^\lambda \tau'\} \tag{2.38}$$

On the other hand, the max-entropy is defined as:

$$H_{\max}(A\|B)_\rho := -H_{\min}(A\|C)_\rho, \tag{2.39}$$

where the min-entropy is evaluated for a purification ρ_{ABC} of ρ_{AB}.

Furthermore, the *smooth min- and max-entropy* of a state ρ is given by min- and max-entropy for a state ρ', which is ε-closed to ρ with respect to distance measures. ε is also called the smoothness parameter.

Definition 2.8. *Let $\rho = \rho_{AB}$ be a bipartite density operator and let $\varepsilon \geq 0$. The ε-smooth min- and max-entropy of A conditioned on B are given by:*

$$\begin{aligned} H_{\min}^{\varepsilon}(A|B)_{\rho} &:= \sup_{\rho'} H_{\min}(A|B)'_{\rho} \\ H_{\max}^{\varepsilon}(A|B)_{\rho} &:= \inf_{\rho'} H_{\max}(A|B)'_{\rho}, \end{aligned} \tag{2.40}$$

2.1.5 Quantum Computation Models

Until now, we have introduced different notions of quantum information processing from different perspectives that are related to our works in the thesis. However, the same important question that people care about is how to gather these quantum information techniques to form a quantum computation device within the scope of the concept of modern computing systems. To answer this question, it is necessary to understand different models of quantum computation. They are also the essential guidelines for constructing quantum computers in practice.

Turing machine is an abstract mathematical model of computation proposed by Alan Turing [2], which is nowadays one of the foundational concepts in the theory of computation. As described in [28], a Turing machine mainly contains four elements: (a). A *program*, rather like an ordinary computer; (b). A *finite state control*, which acts like a stripped-down microprocessor, coordinates the other operations of the machine; (c). A *tape*, which acts like a computer memory; and (d). A *read-write tape-head* points to the position on the tape, which is currently readable or writable. A quantum Turing machine (QTM) has the same formalisation of abstraction for quantum computation, which deals with quantum states in a Hilbert space of the quantum mechanical system. It was first proposed in [3] and [4] by Paul Benioff, then further developed in [5] by David Deutsch.

In practice, there are different models for quantum computation. *Quantum circuit model* is the quantum analogue of the Boolean circuit model. In the classical Boolean circuit model with classical input/output, any computational task can be modelled by a Boolean function $f : \{0,1\}^n \to \{0,1\}^m$, which is computable by a circuit using just AND, OR and NOT gate, i.e., AND, OR, and NOT gates form the universal gate set. Whereas in the quantum circuit model, the inputs are qubits, or quantum states in general. The logic gates here are unitary transformations. For the basic quantum logic gates, we have already introduced and described their unitary matrices previously. Meanwhile, it also requires that the logic gate set in the quantum circuit model is supposed to be universal to approximate arbitrary unitary operations within a quantum circuit[2].

Another model of quantum computation is *Measurement-based Quantum Computing* (MBQC), originally proposed by Raussendorf and Briegel [39, 40, 41]. Unlike

[2]It turns out that the universal quantum gate sets are not unique. Nevertheless, according to Solovay—Kitaev theorem [38], it is possible to take $\Theta(\log^c(1/\epsilon))$ gates from a fixed finite set to ϵ-approximate any gate U on a single qubit, where c is a small constant approximately equal to 2.

the quantum circuit model, quantum information processing in MBQC is achieved through a sequence of measurements on an entangled quantum state as the resource state rather than by applying a series of quantum gates. MBQC is often implemented using a resource state called a *cluster state* in the family of *graph state*. A cluster state is formed by preparations of single qubits in the state $|+\rangle$, except the input qubits, then entanglements of two qubits with the CZ operator. Each operation is performed by applying a measurement on the qubits determined by a predefined measurement pattern with basis $\{|+_\theta\rangle, |-_\theta\rangle\}$, where $|\pm_\theta\rangle = \frac{1}{\sqrt{2}}(|0\rangle \pm e^{i\theta}|1\rangle)$ with measurement results 0 and 1 respectively, and do corrections on single-qubits with operators X, Z depending on measurement results [42]. This technique is also known as *gate teleportation* [43]. Also, MBQC is a universal model of quantum computation. Due to the limitation of quantum hardware devices, MBQC offers advantages in terms of fault tolerance and scalability compared to the quantum circuit model.

Meanwhile, MBQC model is also suited for certain quantum communication protocols. Universal blind quantum computation (UBQC), originally introduced in [44], is a MBQC-based quantum computation model for delegated quantum computation with a privacy-preserving guarantee with a quantum communication channel in between. At the start of a UBQC protocol, a client produces a sequence of single-qubit states of the form $|+_\theta\rangle$ with θ chosen uniformly at random from $\{0, \frac{\pi}{4}, \ldots \frac{7\pi}{4}\}$. After receiving N such qubits from the client through a quantum channel, a server entangles them to build a *brickwork state* in the family of *graph state*. The computational stage is interactive and uses only classical communication. During this stage, the client continuously sends the measurement angle for each qubit to the server, which returns the measurement result to the client. The client then updates the successive measurement angle. At the end of the computation, the server returns the quantum outputs to the client. We will elaborate on UBQC and its variants with different security properties in Chapter 5.

2.2 Cryptographic Primitives

From this section, we forward our steps to different concepts of cryptographic primitives that are related to our thesis. *Cryptography* is nowadays an essential tool for guaranteeing security in different dimensions throughout communications and computations in the presence of adversaries with different powers. For a cryptographic protocol, an *adversary* is assumed to deviate from the execution of the protocol arbitrarily, mostly to obtain underlying secret information. In the context of quantum information and quantum computation technologies, the power of the adversary in terms of its computational power can classify the adversary into *unbounded*, *probabilistic polynomial time* (PPT), or *quantum polynomial time* (QPT) adversary. A protocol is *information-theoretically secure* if it is proven to be secure against an unbounded adversary, where there are no additional assumptions made on the adversary. Otherwise, the protocol can be *computational secure* against an *efficient* adversary in PPT/QPT, where PPT adversary is limited to performing classical algorithms in polynomial time and possibly making random choices, and QPT adversary can make use of quantum oracle algorithms, and quantum com-

puters in polynomial time [3]. Meanwhile, we analyse the computational security by an asymptotic approach incorporating a *security parameter* to measure the level of security. In the rest of our thesis, since we consider the adversary to be with quantum power, **we refer to an efficient adversary/algorithm as a QPT adversary/algorithm.**

2.2.1 One-way Function

One-way function (OWF) is the most basic primitive for cryptographic functions. We say a function is one-way if it is easy to compute but hard to invert. On the one hand, by saying it is easy to compute, we mean that the function can be evaluated in polynomial time with arbitrary input. On the other hand, by saying it is hard to invert, the success probability of inverting the OWF is *negligible* by any efficient algorithm. A function ϵ is negligible if it decays faster than any inverse polynomial in the security parameter λ, and we denote it by $\mathsf{negl}(\lambda)$. A formal definition is given as follows:

Definition 2.9 (Negligible function)**.** *A positive function $\epsilon : \mathbb{N} \to \mathbb{R}$ is negligible if for all $c \in \mathbb{N}$, there exists $\Lambda \in \mathbb{N}$ such that $\epsilon(\lambda) \leq \frac{1}{\lambda^c}$ for all $\lambda \geq \Lambda$.*

Definition 2.9 also says that the function ϵ is *negliglible* if and only if $\epsilon(\lambda) \in O(1/p(\lambda))$ for all positive polynomials p. On the other hand, the function ϵ is *non-negligible* if there exists a polynomial p such that for a large enough λ, it holds that $\epsilon(\lambda) > 1/p(\lambda)$.

We then give a formal definition of *one-way function*:

Definition 2.10 (One-way Function (OWF))**.** *A function $f : \{0,1\}^n \to \{0,1\}^m$ is a one-way function if:*

- *(Easy to evaluate). The function f can be evaluated in polynomial time on every input.*
- *(Hard to invert). For an efficient adversary $\mathcal{A}$, there exists a negligible function ϵ with sufficiently large security parameter n such that:*

$$\Pr[f(\mathcal{A}(f(x))) = f(x) : x \leftarrow \{0,1\}^n] \leq \epsilon(n). \tag{2.41}$$

There are a few remarks for Definition 2.10: Firstly, we use $f(\mathcal{A}(f(x))) = f(x)$ instead of $\mathcal{A}(f(x)) = x$ by concerning the fact that the function f is not necessarily to be one-to-one but in the case that $f(x)$ might have more than one preimage. Secondly, by saying a single function $f : \{0,1\}^n \to \{0,1\}^m$ in the definition, one-wayness can also be defined over a family of functions.

It turns out that OWFs are so powerful that they are sufficient to construct other important cryptographic primitives, including *pseudorandom generator* and *pseudorandom function*. We will introduce these important primitives in the next subsection.

[3] For completeness, a formal definition is given in [45] states that an algorithm A is said to run in polynomial time if there exists a polynomial $p(\cdot)$ such that, for every input $x \in \{0,1\}^*$, the computation of $A(x)$ terminates within at most $p(||x||)$ steps, where $||x||$ denotes the length of the string x

2.2.2 Pseudorandomness

Randomness is one of the most important properties in modern cryptography. Theoretically, generating any cryptographic key requires using a truly random source. However, obtaining a truly random source with no bias or correlation between each time of generation is sometimes expensive. In this case, *pseudorandomness* has been introduced and is widely used nowadays in modern cryptography.

A crucial criterion for an object with pseudorandomness property is *computational indistinguishable* from the true randomness. To give an insight into the terminology computational indistinguishability, we give a formal definition as follows:

Definition 2.11 (Computational indistinguishability)**.** *Let $D_{1,\lambda}$ and $D_{2,\lambda}$ be two probability distributions with security parameter $\lambda \in \mathbb{N}$. We say D_1 and D_2 are computationally indistinguishable written as $D_1 \approx_c D_2$, if for any efficient adversary $\mathcal{A}$, we have*

$$\left|\Pr[\mathcal{A}(1^\lambda, x) = 1 : x \leftarrow D_{1,\lambda}] - \Pr[\mathcal{A}(1^\lambda, x) = 1 : x \leftarrow D_{2,\lambda}]\right| \leq \epsilon(\lambda). \tag{2.42}$$

Therefore, to formally define pseudorandomness, we have:

Definition 2.12 (Pseudorandomness)**.** *Let D_λ be a probability distribution with security parameter $\lambda \in \mathbb{N}$. We say D_λ is pseudorandom if for any efficient adversary $\mathcal{A}$, it can distinguish between D_λ and a uniformly random distribution $\mathcal{R}_\lambda$ with negligible probability $\epsilon(\lambda)$, i.e., D_λ is computationally indistinguishable from $\mathcal{R}_\lambda$:*

$$\left|\Pr[\mathcal{A}(1^\lambda, x) = 1 : x \leftarrow D_\lambda] - \Pr[\mathcal{A}(1^\lambda, x) = 1 : x \leftarrow \mathcal{R}_\lambda]\right| \leq \epsilon(\lambda). \tag{2.43}$$

In modern cryptography, we use *pseudorandom generator* (PRG) to obtain a pseudorandom string[4]. A pseudorandom generator is an efficient and deterministic algorithm that receives a short, truly random secret seed and stretches it into a long string that is pseudorandom [45]. Formally, we have:

Definition 2.13 (Pseudorandom Generator(PRG))**.** *Let ℓ be a polynomial and let G be a deterministic polynomial time algorithm such that upon any input seed $s \in \{0,1\}^n$ where $n \in \mathbb{N}$, the algorithm G outputs a string $\{0,1\}^{\ell(n)}$. We say that G is a pseudorandom generator if the following two conditions hold:*

- *(Expansion). For every n, it holds that $\ell(n) > n$.*
- *(Pseudorandomness). For any efficient adversary $\mathcal{A}$, there exists a negligible function ϵ in security parameter n such that:*

$$\left|\Pr[\mathcal{A}(r) = 1 : r \leftarrow \{0,1\}^{\ell(n)}] - \Pr[\mathcal{A}(G(s)) = 1 : s \leftarrow \{0,1\}^n]\right| \leq \epsilon(n). \tag{2.44}$$

[4]PRG is used as the formal concept in theoretical computer science. For the common meaning of this term, see pseudorandom number generator. (PRNG)

Note that we emphasize that the adversary $\mathcal{A}$ is bounded within quantum polynomial time. Otherwise, the output of pseudorandom generator G is indeed far from the true randomness. Imagine if $\mathcal{A}$ tries to distinguish between a random string of length $2n$ and a pseudorandom string with the pseudorandom generator $G : \{0,1\}^n \to \{0,1\}^{2n}$ within exponential time. The true random string follows the uniform distribution over $\{0,1\}^{2n}$, yet there are only 2^n different possible strings for the pseudorandom string since the input of G is with length n. Hence, given a string uniformly distributed in $\{0,1\}^{2n}$, the probability that there exists a seed of G for such string is $\frac{1}{2^n}$. As a result, the distinguishing probability is therefore:

$$\left|\Pr[D(r) = 1 : r \leftarrow \{0,1\}^{2n}] - \Pr[D(G(s)) = 1 : s \leftarrow \{0,1\}^n]\right| = 1 - \frac{1}{2^n} \tag{2.45}$$

Furthermore, we introduce the *pseudorandom function* (PRF) family, defined as a family of functions indexed by *key*, which is computationally indistinguishable from the set of truly random functions having the same domain and range. A *keyed function* takes two inputs: a key $k \in \mathcal{K}$, an input $x \in \mathcal{X}$, and output $y = F(k,x) \in \mathcal{Y}$. If the key k is chosen and fixed, we can rewrite the key function as $F_k(x) = F(k,x)$ with single input x. Formally, we have:

Definition 2.14 (Pseudorandom Function(PRF))**.** *Let $\mathcal{K}, \mathcal{X}, \mathcal{Y}$ be the key space, domain, and range parameterized by the security parameter n, a family of keyed functions $\{F : \mathcal{K} \times \mathcal{X} \to \mathcal{Y}\}$ is a pseudorandom function family if for any efficient adversary $\mathcal{A}$, F_k with a randomly chosen key $k \leftarrow \mathcal{K}$ is computationally indistinguishable from a function $f : \mathcal{X} \to \mathcal{Y}$ from the set of truly random functions $\mathcal{F}$:*

$$\left|\Pr[\mathcal{A}^{F_k(\cdot)}(1^n) = 1 : k \leftarrow \mathcal{K}] - \Pr[\mathcal{A}^{f(\cdot)}(1^n) = 1 : f \leftarrow \mathcal{F}]\right| \leq \epsilon(n), \tag{2.46}$$

with a negligible function $\epsilon(n)$.

Here, the key k must not be known by the adversary $\mathcal{A}$ since it is trivial to distinguish $F_k(\cdot)$ from $f(\cdot)$ if k is known: Then $\mathcal{A}$ queries the given function with 0^n and obtain the output y, it then compares the result with $F_k(0^n)$. If the given function is F_k, then the results are equal; Otherwise, the probability that two results are equal is $1/2^n$.

As mentioned by the end of Section 2.2.1, there are inherent relations among OWF, PRG, and PRF. Specifically, we have:

$$OWF \Leftrightarrow PRG \Leftrightarrow PRF, \tag{2.47}$$

where each arrow means we can construct one primitive from the other [46, 47]. Meanwhile, OWF and pseudorandomness are important concepts in cryptography and complexity. OWFs are indeed a minimal assumption for almost all cryptography, not only for obtaining PRGs and PRFs. Pseudorandomness, as a computational relaxation of true randomness, plays a fundamental role in cryptography in general as long as true randomness is inefficient in terms of usage and expense. Still, with the advancement of hardware random number generator techniques, true randomness is preferred in situations where security and unpredictability are critical.

In the quantum regime, the true randomness on the spaces of quantum states and unitary operators is referred to as *Haar randomness* according to the Haar measure. Furthermore, a family of pseudorandom quantum states (PRS's) is a set of random states $\{|\phi_k\rangle\}_{k\in\mathcal{K}}$ indistinguishable from Haar random quantum states, with the following definition in [48]:

Definition 2.15 (Pseudorandom quantum states (PRS's))**.** *Let κ be the security parameter. Let $\mathcal{H}$ be a Hilbert space and $\mathcal{K}$ the key space, both parameterized by κ. A keyed family of quantum states $\{|\phi_k\rangle \in \mathcal{H}\}_{k\in\mathcal{K}}$ is pseudorandom, if the following two conditions hold:*

- *(Efficient generation). There is a QPT algorithm that generates state $|\phi_k\rangle$ on input k. That is, for all $k \in \mathcal{K}, G(k) = |\phi_k\rangle$.*
- *(Pseudorandomness). Any polynomially many copies of $|\phi_k\rangle$ with the same random $k \in \mathcal{K}$ is computationally indistinguishable from the same number of copies of a Haar random state. More precisely, for any efficient adversary $\mathcal{A}$ and any $m \in poly(\kappa)$,*

$$\left|\Pr[\mathcal{A}(|\phi_k\rangle^{\otimes m}) = 1 : k \leftarrow \mathcal{K}] - \Pr[\mathcal{A}(|\psi\rangle^{\otimes m}) = 1 : |\psi\rangle \leftarrow \mu]\right| \leq \epsilon(\kappa), \quad (2.48)$$

with a negligible function $\epsilon(\kappa)$, where μ is the Haar measure on $\mathcal{H}$.

To construct pseudorandom states, it is shown in [48] that a family of *random phase states* satisfies Definition 2.15 of PRS's, and this family of states are proven to be constructed from PRF efficiently as:

$$|\phi_k\rangle = \frac{1}{\sqrt{N}}\Sigma_{x\in\mathcal{X}}\omega_N^{F_k(x)}|x\rangle\,, \quad (2.49)$$

with the family of pseudorandom functions $\{F_k\}_{k\in\mathcal{K}}$ in Definition 2.14. PRS's are shown to be very useful and fundamental in various cryptographic applications [48, 49, 50][5]. More importantly, we are highly interested in the construction of PRS's from PRF, such that it provides us with a guideline to implement it in the real world with currently available hardware devices. Since it is not the scope of our work in the thesis, we will further elaborate on the bridge and potential implementations between these two in practice.

2.2.3 Cryptographic Systems

Generally, a *cryptographic system* is a framework or set of rules and algorithms used to guarantee different security properties, e.g., confidentiality, integrity, authenticity, etc. of information during transmission and storage. A main distinction among different cryptographic systems is how to use the keys, leading to the basic concepts of private and public key cryptosystems. In this section, we briefly revisit the frameworks of private and public key cryptosystems by definition. For each, we discuss the real-world protocols/algorithms and their security against a QPT adversary, which are relevant to our work in the thesis.

[5] Another interesting question arose by [51] and get strong interest from the community is that we obtain PRS's even no post-quantum classical cryptographic primitive exists. That is, without assuming the world Minicrypt we live in, we can still obtain important cryptographic primitives, e.g., digital signature, quantum commitment, etc. Whereas it is even further to the hardware level.

Private Key Cryptosystems. Private key cryptosystems, or secret-key cryptography, is a class of cryptographic systems where the same key is used for both encryption and decryption. For a private key setup between two parties, they pre-shared some information in secret called *secret key*. The sender encrypts a message using the secret key and sends it to the receiver. When the receiver receives it, he uses the same secret key to decrypt and obtain the message. The message itself is called *plaintext*, and the encrypted message is called *ciphertext*. Since we use the same secret to encrypt and decrypt the message here, we also call the setup *symmetric key cryptosystems*. Here, we give a formal definition of a general private key cryptosystem:

Definition 2.16 (Private key cryptosystem)**.** *A private key cryptosystem includes three algorithms that are efficient to evaluate: (*Gen*,* Enc*,* Dec*). Let* $\mathcal{K}, \mathcal{M}, \mathcal{C}$ *be the space of key, plaintext and ciphertext; Each algorithm is specified as follows:*

- *The key generation* Gen *takes the security parameter* n *as input and outputs a secret key* $k \leftarrow \mathsf{Gen}(1^n)$*, where* $k \in \mathcal{K}$*.*
- *The encryption* Enc *takes the key* k *and a plaintext* $m \leftarrow \mathcal{M}$ *as input and outputs a ciphertext* $c := \mathsf{Enc}(k, m)$*, where* $c \in \mathcal{C}$*.*
- *The decryption* Dec *takes the key* k *and the ciphertext* c *as input and outputs the plaintext* $m := \mathsf{Dec}(k, c)$*.*

Meanwhile, it satisfies that $\mathsf{Dec}(k, \mathsf{Enc}(k, m)) = m$*, so-called the property of correctness.*

Classically, a well-known private key cryptosystem is the *one-time pad* (OTP). The idea is to use a truly random secret key (pad) that is at least as long as the plaintext to bitwise combine with the plaintext using XOR for encryption. And the recipient uses the same pad and bitwise combines it again with the ciphertext using XOR for decryption. With the conditions above satisfied, the OTP scheme provides information-theoretic security [45], which means an unbound adversary, without knowing the pad cannot determine any information about the plaintext. However, the main issue comes from the key management since generating truly random keys and securely distributing them to the sender and the receiver can be costly and complicated. Nevertheless, by possibly leveraging a combination of hardware security primitives with quantum information techniques as part of future work, we show that the key management issue can be much less overhead and more efficient.

Meanwhile, there exist practical private key cryptosystems to exploit with, e.g., *Advanced encryption standard* (AES) [52], proposed by Daemen and Rijmen. It has been established as a standard encryption algorithm by NIST since 2001. It is a Substitution-Permutation Network alternating between linear layers, non-linear layers and round key additions [53]. To perform encryption on blocks of 128 bits by AES, we first split it into 16 bytes. The round function of each round comprises four operations: *AddRoundKey* (ARK), which xors the round key with the current state; *SubBytes* (SB), which applies the AES S-Box to each byte; *ShiftRows* (SR), which shifts the i-th row by i bytes left; and *MixColumns* (MC), which multiplies

each column by the AES MDS matrix. To the best of our knowledge, AES seems a resistant primitive in the post-quantum world and the classical one, with a bigger security margin with respect to quantum generic attacks [53].

When AES is considered post-quantum secure on the level of the round function, the main threats can come from the key scheduling session, in which the original secret key is used to create a set of round keys [54]. First, if the length of the original secret key is not long enough, a brute-force attack might be achievable by trying out all possible secret keys. The security of AES depends on the secret key length in this case. Meanwhile, AES is vulnerable to certain weak key attacks, e.g., if all key bytes are set to the same value, the key schedule may become less secure. Last but not least, different attacks on the hardware level can either sabotage the proper running of the algorithm or intercept the side channel information of AES such that an adversary may gain information about the key schedule or even the original encryption key. Where the first and the second issues can be resolved trivially, we think a possible solution to tackle the third issue can be a combination of hardware security primitives and AES. On the one hand, an appropriate hardware security primitive can provide a secure vault for storing cryptographic keys and prevent them from unauthorized access. On the other hand, AES encryption and decryption operations can be performed within the protection of hardware-based mechanisms to ensure that sensitive data is processed securely. We will further discuss this possible solution in Chapter 5

Public Key Cryptosystems. Public key cryptosystems, also known as *asymmetric key cryptosystems*, are a class of cryptographic systems that use a pair of keys: a *public key* and a *private key*. These keys are mathematically related, but it is computationally infeasible to determine the private key based on the public key. Public key cryptosystems are widely used for encryption, digital signatures, key exchange, etc. Here, we give the formal definitions of a general public key encryption (PKE) system in Definition 2.17 and digital signature scheme in Definition 2.18 in the following[6]:

Definition 2.17 (Public key encryption (PKE) systems)**.** *Let $\mathcal{M}$ be the space of plaintext. A public key encryption system is a tuple of algorithms that are efficient to evaluate (*Gen*,* Enc*,* Dec*) satisfying the following:*

- *Algorithm* Gen *takes as input a security parameter* 1^n *and outputs a pair of key:* $(pk, sk) \leftarrow \mathsf{Gen}(1^n)$*. We refer to the first of these as the public key and the second as the private key. We assume for convenience that pk and sk each have a length at least n, and that n can be determined from* pk, sk*.*

- *Algorithm* Enc *takes as input a public key pk and a message* $m \leftarrow \mathcal{M}$*. It outputs a ciphertext c, and we write this as* $c := \mathsf{Enc}(m, pk)$*.*

[6]Note that digital signatures are often mistakenly viewed as the "inverse" of PKE systems, with the roles of the sender and receiver interchanged. Even though it has been suggested that digital signatures can be obtained by "reversing" PKE systems, the real-world construction in this way is completely unfounded. [45]

- *Algorithm* Dec *takes as input a private key* sk *and a ciphertext* c*, and outputs a message* m *or a special symbol* $\perp$ *denoting failure. We assume without loss of generality that* Dec *is deterministic and write this as* $m := \mathsf{Dec}(c, sk)$.

We require that for every n*, every* (pk, sk) *output by* $\mathsf{Gen}(1^n)$*, and every message* m *in* $\mathcal{M}$*, it holds that* $\mathsf{Dec}(\mathsf{Enc}(m, pk), sk) = m$.

Definition 2.18 (Digital signature schemes)**.** *Let* $\mathcal{M}$ *be the space of the message. A digital signature scheme is a tuple of algorithms that are efficient to evaluate (*Gen*,* Sign*,* Vrfy*) satisfying the following:*

- *The key generation algorithm* Gen *takes as input a security parameter* 1^n *and outputs a pair of key:* $(pk, sk) \leftarrow \mathsf{Gen}(1^n)$*. We refer to the first of these as the public key and the second as the private key. We assume for convenience that* pk *and* sk *each have a length at least* n*, and that* n *can be determined from* pk, sk*.*

- *The signing algorithm* Sign *takes as input a private key* sk *and a message* $m \leftarrow \mathcal{M}$*. It outputs a signature* σ*, and we write this as* $\sigma \leftarrow \mathsf{Sign}(m, sk)$*.*

- *The deterministic verification algorithm* Vrfy *takes as input a public key* pk*, a message* m*, and a signature* σ*. It outputs a bit* b*, with* $b = 1$ *meaning VALID and* $b = 0$ *meaning INVALID. We write this as* $b := \mathsf{Vrfy}(m, \sigma, pk)$

We require that for every n*, every* (pk, sk) *output by* $\mathsf{Gen}(1^n)$*, and every message* m *in* $\mathcal{M}$*, it holds that* $\mathsf{Vrfy}(m, \mathsf{Sign}(m, sk), pk) = 1$

The security of public key cryptosystems is based on mathematical hardness assumptions that are believed to be computationally hard, such as factoring large numbers, e.g., RSA, or solving the discrete logarithm problem, e.g., *Diffie-Hellman* (DH), *Elliptic curve cryptography* (ECC). However, the emergence of Shor's algorithm [6] menaces these cryptographic schemes by showing that the problems of factoring and discrete logarithms can be solved efficiently on a large, fault-tolerant and universal quantum computer.

According to these, *Lattice-based* cryptography is one of the most popular directions to derive post-quantum cryptographic systems. In [55], the problem of short integer solution (SIS) was firstly defined and related its average case complexity to the worst-case hardness of finding short vectors in every integer lattice, giving lattice-based one-way functions and lattice-based trapdoor functions. Meanwhile, [56] introduced the NTRU public-key encryption system and the related ring-based NTRU problem, which are believed to be related to the closest vector problem (CVP) in a Lattice. And CVP is known to be NP-hard. On the other hand, *learning with errors* (LWE) problem is also a mainstream Lattice-based cryptography defined by [57] as a mathematical hardness assumption to form the fundamental base for post-quantum secure public-key cryptosystems. It is asymptotically related to the problem of the worst-case hardness of finding short vectors in lattices, so-called *shortest vector problem* (SVP), known as NP-hard [58]. Furthermore, Miccancio [59] introduced a ring-based analogue of Ajtai' s SIS problem and a search variant of ring-based LWE (and an associated public-key encryption scheme, relying on the Goldreich-Levin hardcore function [60]) was introduced in [61]. Also,

an efficient reconciliation-based mechanism for constructing a simple and provably secure key exchange scheme from LWE was discovered in [62] as an analogue of the DH key exchange but with errors. Nowadays, cryptographic algorithms of PKE schemes like CRYSTALS—Kyber, and digital signature schemes like CRYSTALS-Dilithium and Falcon, which belong to lattice-based cryptography, have already been through multiple rounds of verification and acknowledged by NIST [63].

At the same time, other computational models with different mathematical hardness are also shown with quantum security. *Code-based* cryptography, e.g., BIKE, Classic McEliece, and HQC PKE or *Key encapsulation mechanism* (KEM)[7] schemes, and *Hash-based* cryptography, e.g., SPHINCS$^+$ PKE scheme [63] are also algorithms that have been verified and acknowledge by NIST. However, the overall drawbacks of quantum-secure cryptographic schemes are increased computational costs, longer processing times, and higher resource requirements, which may concern resource-constrained devices and applications, especially on the hardware level. Hence, even though optimisation is not in the scope of our thesis, it is still a very interesting and important question regarding implementations.

2.3 Different Models of Security

In modern cryptography, we prove the security of cryptographic protocols or algorithms in a formal way within different security models, which depend on how the target protocols are used in different scenarios. For example, one of the most well-known security models is *game-based security* model, which captures the security properties by defining a game and specifying the winning condition in the game setting. Another useful security model is *simulation-based security* model that instantiates the scenarios of the ideal and real world, respectively. Then, comparing the two worlds from different perspectives tells whether the characterisation of security in the ideal world can effectively reflect the actual run of the protocol in the real world. Meanwhile, it offers a stronger security guarantee from standalone security to a general composition scenario by *composability frameworks*. In this section, we introduce, in general, the definitions, setups and proof techniques in terms of different security models that are related to our works in the thesis.

2.3.1 Game-based Security Model

To obtain rigorous guarantees on the security of a cryptographic protocol in the game-based security model, we first define a specific security property that we expect the protocol to respect. Then, we capture the security property in a specified *attack game*, usually between an efficient party called *challenger* and another efficient party called *adversary* by definition. The definition of security is tied to some particular event with a *target probability*. That is, as long as the game ends up by definition, if the probability of such an event occurring in the game is negligibly close to this target probability, we say the protocol is secure in terms of the prop-

[7] Key encapsulation mechanisms are commonly used in hybrid encryption schemes, where PKE is used to securely exchange a shared symmetric key, which is then used for encrypting the actual message content. This combines the security benefits of asymmetric and symmetric cryptography.

erty that we specify in the first place. Otherwise, the adversary wins the game, and the protocol is not secure.

The sequence of games approach in the game-based security model is commonly used. Generally, one constructs a sequence of games $\mathcal{G}_1, \mathcal{G}_2, \ldots, \mathcal{G}_n$ with $\mathcal{G}_0$ capturing the original attack regarding the target cryptographic primitive and the malicious behaviours of the adversary. As long as $\mathcal{G}_1$ respects the security property with the target probability of the end-up event, meanwhile, the successive games follow the same with their respective end-up events related to the first one, and they show that each event respects the probability that is negligibly close to one another successively. We obtain the corresponding security properties of successive games in the game-based security model. This methodology can be figured as a way to perform security reduction. Or the other way around, such that a break in one game implies a break in the next game.

The game-based security model is widely used in cryptographic analysis to capture the security primitive of a specified classical/quantum protocol. In the case when we specify clearly the security property of the target protocol with the particular event, as well as the adversarial behaviours are well defined and captured in the game, the security proofs are simple to do and easy to understand. In Chapter 4 of our thesis, we leverage the game-based security model and proof techniques to capture the security definition regarding the unforgeability of Physical unclonable functions (PUFs) with different setups in combining quantum encoding schemes against network adversaries, and constructions while facing different types of adversaries. Meanwhile, we encourage the readers to have a further read on the work [64] for a complete understanding of the game-based security model and related proof techniques.

2.3.2 Simulation-based Security Model

The simulation-based security is yet another security model for the cryptographic analysis of classical/quantum protocols. The security here is captured by introducing two worlds, the *ideal world* and the *real world*, and comparing what happens in these two worlds, respectively. In the ideal world, the cryptographic primitive that we expect to achieve with the actual protocol in the real world, is trusted and *secure* by definition. On the other hand, the real world specifies the actual execution of the protocol among the participants. The comparison here is called *simulation* since its proof technique is formalised by means of constructing a *simulator* to show the cryptographic primitive is secure. For example, we say a private key encryption scheme is secure when nothing is learned about the plaintext from the ciphertext. In the simulation-based security model, the proof sketch is that an adversary, who is given the ciphertext in the real world and the public information, can not computationally distinguish from the ciphertext via the same encryption of "garbage" (e.g., a binary string with all 0), which is generated from the simulator by giving it only the public information without real ciphertext in the ideal world. If so, we say that the execution in the real world is at least as secure as the functionality in the ideal world. More details of the simulation paradigm can be found in [65]

Here, we are motivated to introduce the simulation-based security model be-

cause our proofs are mostly derived from the simulation paradigm in Chapter 5. Compared to Chapter 4, where our design, security analysis and protocol are in standalone settings, our works on proposed resources, real-world design and protocols in Chapter 5 should be applied mostly in the way of *general composition*, including *sequential* and *parallel* compositions. And the security result from the game-based security model might not be guaranteed in general composition. On the other hand, based on the simulation paradigm, there are *universal composition* (UC) framework and *abstract cryptography* (AC, also known as constructive cryptography) that provide us stronger guarantees.

2.3.3 Security in General Composition: Universal Composition and Abstract Cryptography Frameworks

Following the introduction of the simulation paradigm, we intend to briefly introduce UC and AC frameworks that both guarantee the security of cryptographic protocols by definition in general compositions. On the abstract level, they are quite similar, and AC framework is more like a generalisation of UC framework. Here, we do not want to clarify the difference in definitions in detail in terms of, e.g., computation models, between these two frameworks that might cause unnecessary confusion, but provide the general methodologies for using them.

Universal Composition (UC) Framework. The UC framework introduced in [66, 67] provides us with a *modular* security analysis of a protocol by dividing it into subroutines separately. For the target subroutine, we evaluate the security independently by means of realizing some ideal functionality $\mathcal{F}$. Briefly speaking, the main idea of the universal composability theorem states that if the expected security properties of a protocol that makes the subroutine call to $\mathcal{F}$ hold if $\mathcal{F}$ is replaced by the actual execution of the specified protocol π.

In this case, like the simulation paradigm, the ideal functionality $\mathcal{F}$ and actual specification of protocol π reside in the ideal and real worlds, respectively. We denote the security in the UC framework by showing that if information can be learned by some adversary $\mathcal{A}$ in an actual execution of π in the real world, it could also have been obtained by a simulator σ attacking an execution of functionality $\mathcal{F}$ in the ideal world. Here, the presence of arbitrary other protocols running alongside π is modelled by an environment $\mathcal{Z}$. As a result, the protocol π is said to UC-realise some ideal functionality $\mathcal{F}$, if for any real-world adversary $\mathcal{A}$, there exists a simulator σ, such that no efficient environment $\mathcal{Z}$ can distinguish an interaction with $\mathcal{A}$ and parties running π from an interaction with σ and $\mathcal{F}$. Formally, we have:

Definition 2.19 (UC-realisation [66])**.** *We say that protocol π UC-realises $\mathcal{F}$, if for any efficient adversary $\mathcal{A}$, there exists an efficient simulator σ, such that for any efficient environment $\mathcal{Z}$ with a negligible ε,*

$$IDEAL_{\mathcal{F},\sigma,\mathcal{Z}}(\lambda) \approx_{\varepsilon} EXEC_{\pi,\mathcal{A},\mathcal{Z}}(\lambda). \tag{2.50}$$

Here, the notations IDEAL and EXEC denote the views of the environment $\mathcal{Z}$ in the executions.

More generally, we can use UC to compare two arbitrary protocols, π and ω. That is, we say π UC-realises ω if for any efficient adversary $\mathcal{A}$ in an execution of π, there is an efficient simulator σ in the execution of ω that "disguise" the environment $\mathcal{Z}$' s view, in the sense that no efficient environment $\mathcal{Z}$ can distinguish an execution of π with $\mathcal{A}$ from an execution of ω with σ. To know more definitions and specifications of UC in detail, we encourage the readers to look into the references for a better understanding.

Abstract Cryptography (AC) Framework. The AC framework, on the other hand, was introduced in [68] by Maurer and Renner for getting composable security properties. Compared to UC framework, which is built in a bottom-up approach, AC framework is formalized with a top-down approach, where it considers the highest level of abstraction first, then the lower levels to instantiate particular objects of the protocol. UC can be realized by instantiating the abstraction of AC framework.

In AC framework, the functionality is called a *resource*. A resource has some *interfaces* I corresponding to the parties that the resource interacts with. Since we focus on two-party communication between the client and the server as our protocol in Chapter 5, our resources have two interfaces $I = \{\mathcal{A}, \mathcal{B}\}$ corresponding to the client and the server.

A protocol $\pi = \{\pi_i\}_{i \in I}$ is a set of *converters* indexed by I. A converter has two interfaces - an inside interface and an outside interface, where the inside interface is connected to the resource and the outside interface is connected to the outside world. Converters connected to resources build new resources with the same interface set. Meanwhile, a dishonest party in a protocol has more access to the functionalities of a resource than an honest one. We denote by $\perp$ a filter used to enforce the honest behaviour of a party. In this case, the functionalities accessed by the party are so-called the *filtered functionalities*.

An important concept of the AC framework is the *distinguisher* $\mathcal{D}$ (In UC, it is the role of the environment), which efficiently measures the distance between two resources by accessing all the interfaces of resources. For instance, consider a resource $\mathcal{R}$ and a protocol π_A, π_B, and denote $\pi_A \mathcal{R} \pi_B$ their composition. We say that two resources $\mathcal{R}, \mathcal{S}$ are ε-closed, or $\mathcal{R} \approx_\varepsilon \mathcal{S}$ if there is no distinguisher $\mathcal{D}$ that can distinguish between $\mathcal{R}$ and $\mathcal{S}$ with advantage greater that ε. If ε is negligible, we say that we can securely constructs $\mathcal{S}$ from $\mathcal{R}$ with the protocol π_A, π_B. Furthermore, if the resource $\mathcal{S}$ is secure, we say that the resource $\mathcal{R}$ AC-realises $\mathcal{S}$. The following definition formally defines this.

Definition 2.20 (AC-realisation [68])**.** *Given two resource $\mathcal{R}$ and $\mathcal{S}$, we say that a protocol $\pi = \{\pi_A, \pi_B\}$ realises $\mathcal{S}$ from $\mathcal{R}$ within ε if two following properties are satisfied:*

- *Correctness:*

$$\pi_A \mathcal{R} \pi_B \approx_\varepsilon \mathcal{S} \perp, \tag{2.51}$$

- *Security: if there exists a converter, where it is called a simulator σ such that*

$$\pi_A \mathcal{R} \approx_\varepsilon \mathcal{S}\sigma. \tag{2.52}$$

We denote the AC-realisation by:

$$\mathcal{R} \xrightarrow{\pi,\varepsilon} \mathcal{S} \tag{2.53}$$

Similarly to UC, AC can also be utilised in comparison between two arbitrary resources. To know more definitions and specifications of AC in detail, we encourage the readers to look into the references for a better understanding.

Chapter3

Hardware Security Primitives

3.1 Introduction

In this chapter, we review in detail hardware security primitives from different perspectives, including the backgrounds, concepts, definitions, properties, state-of-the-art constructions, applications, etc. Nowadays, with a huge increment in the use of digital devices for different use cases, there are more and more concerns about data security. From Section 2.2, we learn about the importance of cryptographic algorithms and operations in guaranteeing the different security requirements, e.g., confidentiality, integrity and authenticity of data and applications by mathematical hardness. However, these algorithms or operations can be compromised if the underlying hardware systems are compromised or not secure in the first place due to bad designs. As a result, hardware security primitives play a critical role and are widely studied. Here, hardware security primitives serve as fundamental hardware-based components, mechanisms, or techniques designed to provide security properties and enhance the security of computer systems, electronic devices and integrated circuits. As the physical root-of-trust, hardware security primitives with different cryptographic properties are implemented at the hardware level to protect against unauthorised access, tampering, information leakage, different types of side-channel attacks, etc. Studying different hardware security primitives can not only help us design more robust applications with security requirements in the real world but also allow us to leverage these primitives to reduce or remove computational assumptions in designing cryptographic algorithms with less overhead in complexity and making them more practical.

In the domain of quantum computation and quantum information, quantum cryptography is one of the most potential topics to study. By learning from the previous chapter, we know that there are properties in quantum mechanics, e.g., unclonablility, that are appropriate to construct robust cryptographic protocols. Meanwhile, adversaries with partial or full quantum power can threaten the existing or future classical/quantum cryptographic protocols/algorithms in design, and the computational hardness bounded by classical computing power is no longer suitable for quantum applications. That is also the main motivation for developing post-quantum secure protocols/algorithms. For the study of hardware security primitives in quantum analogue, the situation is the same, except that the quantum properties need to be taken into account in the hardware constructions and evaluations. Another important criterion that should be considered is the practi-

cality regarding the constructions of quantum hardware security primitives since we believe that studying hardware security primitives in the first place is to tackle the security of real-world implementation of applications. Throughout this chapter, we will study the existing state-of-the-art of different hardware security primitives in quantum analogue, including the constructions, applications and limitations.

Considering practical quantum information techniques nowadays, we believe that the near-term quantum hardware security primitives should be in the form of hybrid classical-quantum constructions. In this case, we especially study further in detail two hardware security primitives: *Physical Unclonable Function* (PUF) and *Trusted Execution Environment* (TEE), respectively. With the practicality of constructions in mind, we further introduce our contributions to the corresponding hardware security primitives in quantum analogue, including constructions, security analyses and applications.

3.1.1 Structure of the Chapter

In Section 3.2, we consider the concept of *black-box* constructions that is widely used in designing cryptographic protocols and algorithms from different primitives for security analyses. Then, we introduce different hardware security primitives that can achieve these perspectives with implementations in practice. In Section 3.3, we introduce PUF as a type of hardware security primitive. Furthermore, in Section 3.3.1 and 3.3.2, we go through from the physical constructions to theoretical abstraction by modelling for analysis in the classical setting, then in quantum analogue in Section 3.3.3 and 3.3.4. By understanding completely the properties of PUFs, we show the possible applications in the real world by leveraging PUFs to meet security requirements, with advantages and limitations. In Section 3.4, we introduce another hardware security primitive as TEE. In the classical setting, we discuss in Section 3.4.1 and 3.4.2 TEEs in different classical setups and their characterisations regarding the constructions and functionalities. Moreover, we capture the common abstraction of executing a function in TEEs generally, especially secure processors. Then, we attempt, for the first time, to give the properties that a TEE in quantum analogue should have from the perspectives of construction in Section 3.4.3 and modelling in Section 3.4.4.

3.2 Security in the Real World

For the implementation of cryptographic schemes in the real world, we consider these schemes should be efficient to evaluate while remaining secure in different perspectives by definition. By exploiting cryptographic primitives that we mentioned in 2.2 to construct cryptographic schemes, one of the most common techniques we use is *black-box* construction, i.e., the underlying primitive is only accessible via input-output behaviours by an adversary without knowing internal details. Meanwhile, the *non-black-box* construction permits the adversary to have more knowledge and information about the construction of the underlying primitive, e.g., the code that computes the functionality of the primitive.

The technique of black-box constructions simplifies the design of cryptographic

systems by focusing on the security properties on a high level of abstraction. Meanwhile, it is useful by allowing researchers to leverage the security properties of well-established schemes when analyzing new constructions. That is, it provides a modular way to build new cryptographic primitives or protocols on top of existing ones without needing to re-analyze the security of the underlying schemes from scratch. Last but not least, the security reduction via black-box constructions reduces the overall cryptographic systems to individual black-box components with possibly minimal trust assumptions.

Since the black-box constructions technique is powerful, the black-box constructions of cryptographic primitives should be under careful specification of the input-output behaviours with a clear abstraction of the components involved, while the inputs, outputs and functionalities of the targeted components are defined without exposing internal details.

In the case of using the black-box construction techniques to implement cryptographic schemes as real-world applications securely. Except for the theoretical feasibility, we also emphasise the importance of the practicality of implementing the cryptographic scheme, especially regarding efficiency. Relatively speaking, the non-black-box construction shows that the same cryptographic scheme can be feasible and with some security guarantees, but it always requires overheads on computational and communication complexities to achieve the security requirements.

Since these primitives always rely on some general cryptographic assumptions, using these primitives via black-box construction in the real world is always challenging. One immediate example is for the secure key generation to encrypt or sign data in general: We know the security of a generated key depends highly on the randomness of the source. As we said previously, most of the random number generators are implemented based on computational assumptions, i.e., pseudorandomness, as defined in Section 2.2. However, known attacks on real-world implementations of PRNGs based on cryptographic analysis lead to the immediate loss of security [69, 70, 71]. Meanwhile, instead of the black-box constructions of primitives in cryptoanalysis, the potential security issues of storage [72] and execution [73, 74] of the codes and data that achieve the functionality of cryptographic primitives by means of hardware devices in the real-world computing systems are also challenging. Sometimes, tiny hardware flaws can sabotage the security of the whole implementation of the targeted cryptography scheme.

As a result, people are paying more attention to hardware security, also in the cryptography domain. More and more cryptographic characteristics of hardware by artificial designs are generalised as *hardware security primitives*, then are studied in cryptoanalysis with formal security proofs in the framework of modern cryptography. By using hardware security primitives for the design of cryptographic algorithms/protocols, they are not only significant in achieving the security of cryptographic schemes in the real world but also effectively remove or reduce some complexities while designing cryptographic schemes.

In our thesis, we review and explore different hardware security primitives in cryptographic schemes mainly from three perspectives: Secure key generation, secure storage and execution. We instantiate secure key generation by saying a secure key can be generated every time it is called. For secure storage, the generated secure keys and related confidential data can be stored and assigned without

revealing them. Finally, the secure execution requires no leakage of keys and any related confidential information to an adversary.

3.2.1 Physical Root-of-Trust

Through the description of the physical root-of-trust, we aim to meet the cryptographic security properties based on different hardware security primitives. These primitives show different properties by hardware specifications and designs that provide certain objectives regarding security. Meanwhile, it is robust by saying that these properties are implementable by available hardware devices that we currently have and secure against certain attacks performed by software or hardware adversaries. Here, we briefly overview the well-studied hardware security primitives that can provide us with the different physical root-of-trust, and apply them in real-world cryptographic schemes from different perspectives for security purposes.

Hardware Random Number Generators (HRNGs). HRNGs, also known as *true random number generators* (TRNGs), are devices that generate random numbers from physical processes by hardware instead of mathematical algorithms only. Ideally, compared to pseudorandom numbers that are known to be vulnerable by cryptographic analysis, TRNGs based on physical processes of hardware devices could show good statistical characteristics of entropy and are unpredictable by most cryptographic means.

A traditional and trivial way to obtain true randomness is to flip a coin ideally, which ensures an equal probability of getting heads and tails. However, the main drawback is that the rate is too slow to use in practice. As examples of practical physical processes by hardware as sources of entropy, different physical noises generated from microelectronic devices show ideally true random properties that can be qualified as sources of entropy with near-term techniques, e.g., thermal noise from resistors can be amplified to act as a truly random source [75], avalanche noise from avalanche diodes [76], atmospheric noise that radio receivers can observe [77], etc. However, it should be noted that implementing these sources can be solely sensitive to external environmental influence, e.g., temperature fluctuations since these physical processes yield a low analogue signal. Furthermore, there are proposals for constructing TRNGs using field-programmable gate arrays (FPGAs). The physical entropic source is designed based on the jitter as the variations in the significant instants of a clock or data signal that can come from semiconductor noise, cross-talk, power supply variations, and electromagnetic fields in the operating environment [78]. Meanwhile, chaos-based TRNG practical realisations are proposed in [79, 80].

The realisation of TRNG is also one of the main research directions and applications in the domain of quantum information technology, so-called quantum TRNGs (QRNGs). Not only do we know that quantum mechanics is inherently non-deterministic, but we can also certify the quality of randomness efficiently according to the laws of quantum mechanics compared to the TRNGs above. The main idea is based on the postulate that by measuring a quantum system that is

prepared in a superposition of basis states, the measurement outcome is intrinsically random, as described in Born rule.

A QRNG for generating truly random bits can be realised within photonic systems, where the source of entropy includes preparing well-defined quantum superposition states with a single-photon source and a corresponding detection system. Theoretically, it trivially prepares each photon in superposition as $|+\rangle$ state and measures one by one in Z basis with eigenstates $|0\rangle$ and $|1\rangle$ to obtain one random bit classically. In practice, there are implementations such as polarisation-based QRNG and path-based QRNG [81, 82, 83]. For polarisation-based QRNG, $|0\rangle$ and $|1\rangle$ denote horizontal and vertical polarisations, and $|+\rangle$ denotes the polarisation at an angle of 45° to the horizontal polarisation. A polarising beam splitter (BS) then transmits the horizontal polarisation to a single-photon detector (SPD) and reflects the vertical polarisation to another SPD. As to the path-based QRNG, $|0\rangle$ and $|1\rangle$ denote transmitted and reflected paths, and $|+\rangle$ denotes the superposition of these two path. It utilises two SPDs to measure two corresponding paths split by a symmetric BS, which is similar to the set-up of polarisation-based QRNG.

However, the above implementations of QRNGs are limited by the capabilities of SPDs. For example, we normally evaluate the performance of SPDs by the parameter of dead time, which refers to the minimum time interval during which SPD is unable to register another incoming photon after detecting the first one. That is to say, during this dead time, SPD can not respond to subsequent photons. Typically, a single-photon avalanche diode (SPAD), as a type of SPD, can achieve relatively short dead times, like tens of nanoseconds to 10 microseconds [84]. As a result, the generation rate of random bits is bounded by tens of Mbps, which is unsuitable for high-speed requirement applications, e.g., high-speed QKD for key exchange. With this limitation in mind, on the one hand, there are state-of-the-art SPDs with even shorter dead time, e.g., superconducting nanowire single-photon detectors (SNSPD), whose dead time can achieve in the range of sub-nanosecond to a few nanosecond [85]. On the other hand, instead of using SPDs, high-performance macroscopic photodetectors can also be applied in various QRNG schemes for higher photon detection efficiencies (with generation rate up to Gbps), e.g., the QRNG scheme based on vacuum noise measurements [86, 87, 88] and the QRNG scheme based on measurements of laser phase noise [89, 90, 91, 92, 93]. Compared to setups of SPD, macroscopic photodetectors do not require specialized cooling or complex control electronics, making them more accessible and user-friendly. Meanwhile, macroscopic photodetectors are more cost-effective compared to some advanced SPD devices, such as high-performance SNSPDs or SPADs. Finally, there are QRNG schemes with certified output randomness with information-theoretical security, even though the hardware devices are potentially untrusted. For example, the self-testing QRNG schemes [94, 94, 95] based on the Bell inequality and semi-self-testing QRNG schemes, with either source device independence [96, 97, 98, 99] or measurement device independence [100, 101]. But still, these schemes are limited by practicality from different perspectives. We encourage the readers to look into the references we cite above to further understand the detailed specifications and state-of-the-art implementations on actual hardware devices.

To conclude, HRNGs are a critical component in ensuring that the generation of cryptographic keys has the foundation of true randomness and unpredictability

required for strong security. Meanwhile, as one of the most well-known and applicable hardware security primitives, HRNGs also lift up the attention and studies from academia and industries in other hardware security primitives, including PUFs and TEEs that we further study throughout the thesis.

Physical Unclonable Functions (PUFs). PUFs are another interesting type of hardware primitive, which can provide us with a specific root-of-trust. Generally speaking, a PUF is a unique physical entity that can produce a unique digital output when stimulated by a specific physical input. This function is typically implemented using a physical device or circuit that exhibits intrinsic randomness or variability, which makes it difficult to clone or replicate. In this case, it is more promising and lightweight to achieve secure key storage than using non-volatile memory without dedicated secure mechanisms. Meanwhile, PUFs are used in various applications such as secure key generation, device authentication, anti-counterfeiting and tamper detection. As one of the main research directions throughout the thesis for possible constructions with quantum information technology, we review and discuss in detail the characteristics of PUFs in Section 3.3.

Trusted Execution Environments (TEEs). In general, the main idea of a TEE is to enforce secure execution by leveraging *physical isolation*, i.e., it is typically implemented as a separate hardware module or a dedicated *secure enclave* within secure processors. As a result, the targeted hardware system is separated architecturally into two main areas: Trusted and untrusted areas, so-called *trusted execution environment* (TEE) with dedicated secure measures and *rich execution environment* (REE) with the remaining data and applications with less security requirement. Note that isolation does not intrinsically mean security where specific mechanisms and dedicated silicon designs based on isolation are required. One of the key benefits of a TEE is that it provides a secure platform for running trusted applications, such as mobile payment systems, digital wallets, and biometric authentication systems. Depending on different specifications and implementations, these applications can access sensitive data and resources within the TEE, which helps prevent certain software and hardware attacks, such as malware and side-channel attacks. As another main study throughout the thesis for possible specification in quantum analogue for securing quantum applications, we review and discuss in detail the characteristics of TEEs in Section 3.4.

Finally, in Figure 3.1, we summarise the relation between the physical root-of-trust, hardware security, cryptographic functions, and information security objectives. As we discussed above, the specific primitives based on hardware devices can not only achieve the same security objectives as mathematical hardness provided by protocols or algorithms but also be more relevant to the implementations and concentrate on actual utilisation in different applications. However, even with the security primitives of actual hardware constructions in hand, proper modellings for characterising targeted hardware devices' features are essential in correctly describing their behaviours and proving the security of hardware primitives-based

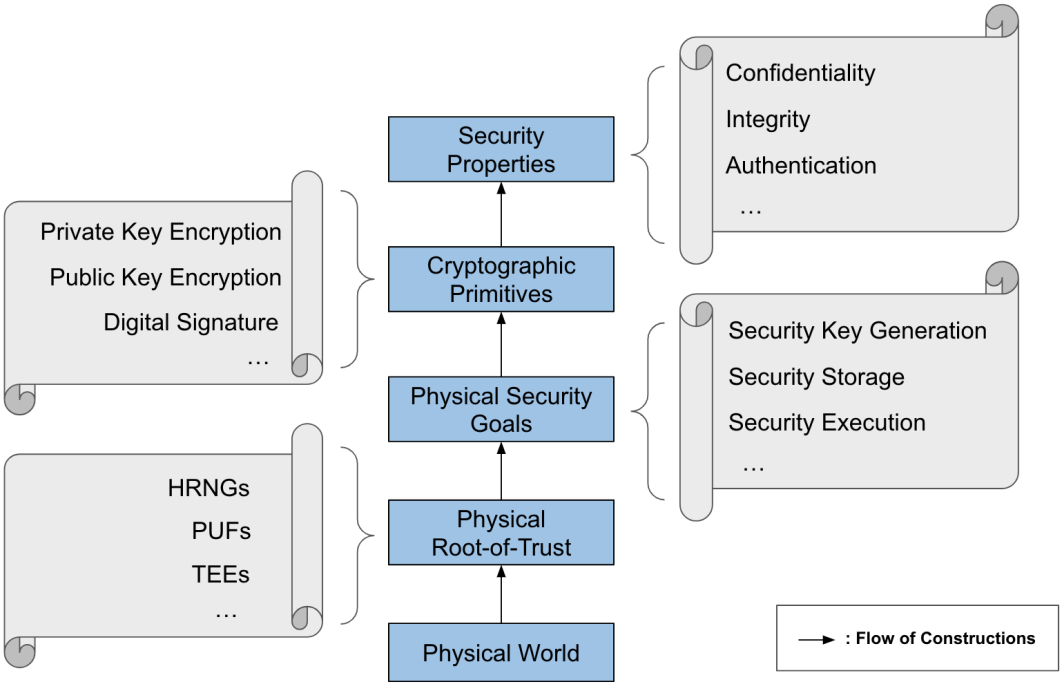


Figure 3.1: The relations among the physical root-of-trust, security goals, cryptographic primitives, and information security properties.

protocols/applications in the scope of modern cryptography. In the following, we elaborate on the PUFs and TEEs as two types of hardware security primitives, including the state-of-the-art constructions in classical/quantum analogues, and then the proper modellings that capture the fundamental and common features that we develop further in Chapter 4 and Chapter 5.

3.3 Physical Unclonable Functions

Physical unclonable functions (PUFs) are hardware devices that exploit inherent randomness to give each physical entity a unique "fingerprint". As introduced previously, the inherent randomness comes from the small, unavoidable variations during the manufacturing process when producing the devices. We show in Figure 3.2 the fundamental concept of PUFs. These variations can be caused by factors such as fluctuations in temperature or humidity, slight differences in the composition of materials used in the device, and imperfections in the manufacturing equipment. Note that these variations should not be systematic; Otherwise, they could be exploited to try to clone or predict the output of a PUF. These variations result in uncontrollable differences in the physical properties of each PUF device, such as the electrical properties of the transistors or the scattering pattern of crystals. These differences can lead to variations in the response of different PUFs to the same input stimulus as their "fingerprints". Ideally, the randomness of PUFs is thus not predictable or reproducible.

The behaviour of a PUF can be described by a set of *Challenge-Response Pairs* (CRPs) that it generates. Precisely speaking, a PUF takes a *challenge* as input.

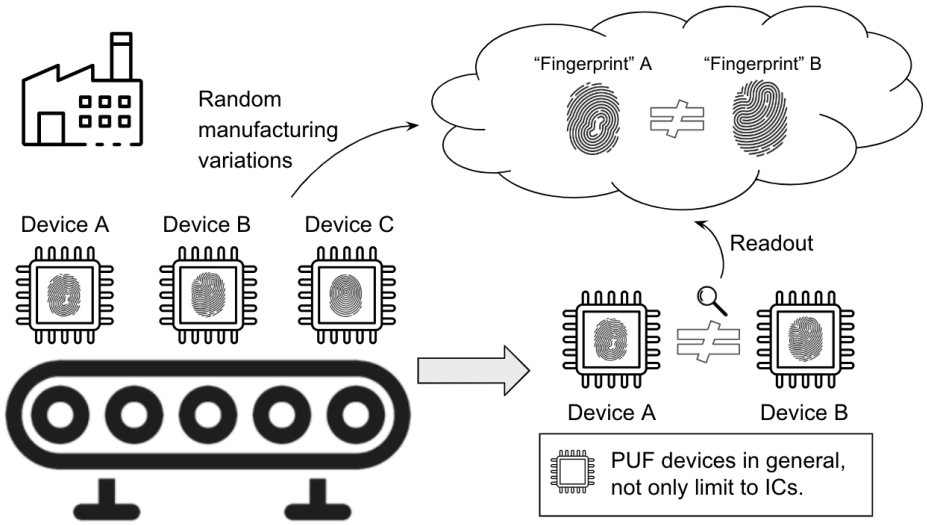


Figure 3.2: **Demonstration of the concept of PUFs.** The randomness appearing in the manufacturing process makes each PUF device a unique one by its "fingerprint".

By processing the challenge based on its unique physical properties, it generates a *response* as output depending on both the challenge and the unique physical properties of the device. Note that a straightforward classification of PUFs is from the space of CRPs, into either *strong PUF*s or *weak PUF*s. Strong PUFs have a very large input-output space in general, which have also been referred to as *Physical one-way function* (POWF) in [102] that emphasises the similarity of POWF and OWF, but with a physical process specifically. With a large size of CRPs' and a limited read-out speed, we usually assume that in applications, there is no limitation on the interface of a strong PUF. On the other hand, weak PUFs have pretty few CRPs. Unlike strong PUFs, the interface of a weak PUF should not be accessed directly due to the small CRPs space. Otherwise, all CRPs of a weak PUF can be read out trivially. Hence, we usually use responses of weak PUFs as inputs of a cryptographic scheme to derive standard secret keys for protocols/algorithms.

To distinguish PUFs' properties from the others in terms of cryptographic functionalities, it is necessary to capture the most significant and common characteristics of PUFs' behaviours by rigorous language, especially when various PUF constructions are proposed and implemented. Furthermore, it plays an essential role in security analysis regarding applicability, composability, etc. Therefore, We will give a detailed introduction to the modelling in the thesis.

Meanwhile, the mathematical modelling is also significant to the security analysis regarding the main security property that people care about in most of the PUF-based applications, as *unforgeability*. In the case of PUFs, it evaluates the capability of a classical or quantum adversary to produce arbitrary input-output pairs with a previously learnt set of inputs and outputs of the targeted device. Regarding the unforgeability of PUFs and the adversarial model, we give a specific discussion in Chapter 4.

3.3.1 Classical PUF Constructions

In this section, we introduce classical PUFs (CPUFs) with classical CRPs that can be constructed using various physical systems that classically exhibit random or unique behaviours. A broad class of CPUFs, which are intensively studied and widely applied, is *silicon PUFs*. This class of PUFs is mainly derived from integrated circuits (ICs). In principle, the manufacturing of ICs produces enough variations even with identical masking such that these variations can be exploited to characterise each IC uniquely. Note that we argue that the optical PUFs [102], which are also introduced as classical PUFs in the first place, should be classified as quantum PUFs in our thesis. Indeed, the optical PUFs are shown to be queried naturally with quantum states. The attacks by adversaries are also no longer limited to classical methods. A detailed discussion is given in Section 3.3.3.

Delay-based PUFs For the constructions, Gassend et al. proposed the idea of silicon PUFs based on peculiar circuit design, as well as the techniques to identify and authenticate each PUF individually in 2002 [103]. Then, a concrete construction of COMS-compatible PUF called *Arbiter PUF* (APUF) was made in [104, 105] based on transistors. The idea is to construct a race condition within a digital circuit to generate a unique, unpredictable response. For APUF, the race condition occurs while two signals generated from a single source compete to arrive at a particular point called arbiter in a digital circuit via two paths. Each path has a series of logic gates introducing slight parameterized signal delays. The structure of these two paths is symmetric. At the same time, the manufacturing variations of transistors and interconnections cause differences in the physical parameters that determine the exact delay of each APUF's paths. In the end, the arbiter detects the arrival of the two signals: If the signal from the upper path arrives before the lower one, it outputs 0 as the response. Otherwise, it outputs 1.

To technically create the race condition within two paths, an APUF with n-bit size challenge consists of n blocks called *switches*. Each block has two inputs and two outputs, and a control bit, which is the role of each bit of the challenge. Each bit of challenge with value 0/1 controls the corresponding block either straightly or switched. In this case, with different challenges, the two paths are parameterized with a delay that feeds into the arbiter differently. Note that the number of challenges is exponential in the number of switches being used. Finally, we give an illustration of the APUF structure in Figure 3.3, and we will come back to it as an instance of our proposed construction in Chapter 4. Note that to extend from a 1-bit response to m-bit with APUFs, we can normally duplicate an APUF itself m times to obtain a multiple-bit response with a single evaluation.

On the other hand, a different approach to digitally measuring delay deviations in circuits in [106], so-called *ring oscillator PUFs* (ROPUFs). In principle, a ring oscillator consists of an odd number of inverter gates connected in a loop. The input signals propagate through the loop and result in an oscillating waveform. Each inverter gate has a slightly different propagation delay due to the manufacturing process variations, which causes the oscillation frequency of each ring oscillator to differ slightly but be distinctive. With multiple ring oscillators integrated on a single chip, it outputs a bit as a response by choosing two ring oscillators and

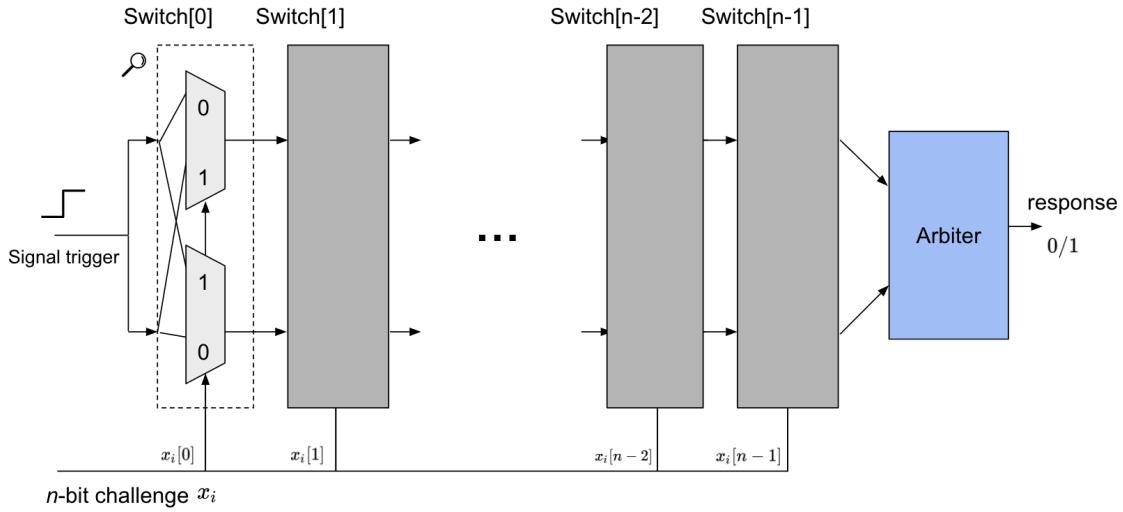


Figure 3.3: Structure of Arbiter PUF.

comparing their frequencies.

The two PUFs above are getting widely interested from industry and academia since the randomness is easily accessible. Meanwhile, the constructions are practical with compact, scalable structures and convenient usage. However, an adversarial strategy called *modelling attack* [107] is becoming a main threat to these two PUF constructions in practice. The idea is to obtain an algorithm that mimics the behaviour of the original PUF on almost all CRPs by giving a limited size of CRPs. For APUF, since the total delay of the paths is linear additive, the output of APUF can be predicted by an additive delay model, which is shown to be vulnerable to modelling attacks based on machine learning algorithms [108, 109, 110, 111, 107]. On the other hand, classic ROPUFs with k ring oscillators can only produce $O(k^2)$ CRPs, an adversary can efficiently perform a sorting-based algorithm by drawing a limited size of CRPs to almost perfectly predict all CRPs [107].

More and more sophisticated designs based on the existing constructions are proposed to exploit the randomness wisely. The constructions of XOR Arbiter PUF, Feed Forward Arbiter PUF, Lightweight Secure PUF, Permutation PUF, and Interpose PUF [112, 113, 114, 115] are proposed based on APUFs. Also the same for the ROPUFs [116]. However, modelling attacks are also becoming more and more powerful with the growth of computing power. Unless the adversary can no longer obtain CRPs without any cost, the mode of modelling attack is always a potential threat that should be taken into account in practice.

Memory-based PUFs Another main proposal for silicon PUFs is to leverage the states of digital devices' memory. The main idea is to read a destabilised memory cell's stable settling state (as 0/1): When the memory cells turn into unstable states, certain types of memory recover to their stable states with a certain bias. This phenomenon does not depend on the circuit's logic design but on the transistors' inherent randomness caused by the manufacturing variations. It shows that these types of memory can be implemented properly as good candidates for PUFs.

In 2007, the idea of *SRAM PUF* [117] was proposed as the first implementation of memory-based PUFs. It utilises the SRAM cells' memory address as a challenge and the read-out state while power-up as its response: When an SRAM cell powers up, it may settle into a random state (0/1) due to the difference in transistor characteristics from manufacturing process variations. With the necessary error-correcting design, it guarantees that the response is consistent and reliable. To use SRAM PUFs, a challenge is applied to select specific SRAM cells, and the corresponding response is extracted.

Since then, more and more memory-based PUFs designs have been proposed, e.g., *butterfly PUFs* [117], *latch PUFs* [118] and *flip-flop PUFs* [119]. One significant characteristic of these PUFs is that the space of CRPs is very limited by size, which indicates them as weak PUFs. Since the CRPs of a weak PUF should not be accessible directly, memory protection should be considered in practice. Otherwise, the security properties of the protocols/algorithms using the key derived from the value of CRPs no longer exist.

Other classical PUFs There are other candidates for PUF constructions, which measure the intrinsic randomness of specific electric or electronic components due to the manufacturing process. For example, the difference in threshold voltages of transistors [120], the difference in power distribution caused by resistance variations in the power grid of a chip [121], and the randomness of capacitance measurements in comb-shaped sensors in the top metal layer of an IC [122], etc. Since it is not the scope of our thesis, we encourage readers to look into the research papers we cite for more details.

In the following, we introduce the modelling of PUFs that capture the characteristics of PUFs' input-output behaviours in common. Note that in our thesis, our contribution mainly focuses on the development of novel PUF constructions with near-term available quantum information techniques based on the use cases of strong PUFs.

3.3.2 Classical PUF Modelling

As various proposals for CPUF constructions are shown in the previous section, we present a comprehensive description of strong CPUFs' behaviour with classical input-output by mathematical modelling in this section. Here, the importance of having a unified mathematical model is in manifolds: First of all, it helps capture precisely the properties of PUFs, e.g., *robustness*, *collision resistance*, and *uniqueness*, which are the most significant properties so-called PUFs in common. This aids in both theoretical analysis and practical design to achieve more robust and secure implementations of PUFs. Secondly, the mathematical model of PUF behaviour enables researchers to expressively evaluate the performance of potential PUF realisations.

Furthermore, a mathematical description can be very useful for security analysis in various PUF-based applications and for identifying potential vulnerabilities. Finally, it helps optimise the designs and constructions of PUFs to achieve desired trade-offs between security, practicality, and resource usage through mathematical models.

By referring to the work that gives a unified model of CPUFs in [123], we generally describe a CPUF as a probabilistic function due to their inherent physical randomness, with the formal description as follows:

Definition 3.1 (Probabilistic Function)**.** *A* probabilistic function *is a mapping* $f : \mathcal{R} \times \mathcal{X} \to \mathcal{Y}$ *with an input space* $\mathcal{X}$*, an* random coin *space* $\mathcal{R}$*, and an output space* $\mathcal{Y}$*.*

For a fixed input $x \in \mathcal{X}$, and a random coin (or key) $R \leftarrow \mathcal{R}$, we define the probability distribution of the output random variable $f(x) := f(R, x)$ over all $y \in \mathcal{Y}$ as,

$$p_x^f(y) := \Pr[f(x) = y|x] = \sum_{r:f(r,x)=y} \Pr[R = r]. \tag{3.1}$$

Therefore, a classical PUF can be modelled as a probabilistic function $f : \mathcal{R} \times \mathcal{X} \to \mathcal{Y}$ where $\mathcal{X}$ is the input space, $\mathcal{Y}$ is the output space of f and $\mathcal{R}$ is the identifier. The creation of a classical PUF is formally expressed by invoking a manufacturing process $f \leftarrow \mathcal{MP}_C(\lambda)$, where λ is the security parameter. Meanwhile, since the function is also referred to as a physical one-way function (POWF) [102], the function follows the definition of one-wayness similar to Definition 2.10 with manufacturing process $\mathcal{MP}_C$.

Furthermore, we give the specific requirements of CPUFs in general, which are parameterized by some threshold δ_i and a negligible function $\epsilon(\lambda) \leq \lambda^{-c}$, where $c > 0$ and λ is large enough.

Definition 3.2. *The classical PUF* $f : \mathcal{R} \times \mathcal{X} \to \mathcal{Y}$ *with* $(\mathcal{MP}_C, \delta_1, \delta_2, \delta_3, \epsilon, \lambda)$ *satisfies the requirements defined below:*

Requirement 3.1 (δ_1-Robustness)**.** *Whenever a single classical PUF:* $f \leftarrow \mathcal{MP}_C(\lambda)$ *is repeatedly evaluated with a fixed input, the maximum distance between any two outputs* $y_i \leftarrow f(x)$ *and* $y_j \leftarrow f(x)$ *is at most* δ_1*. That is for a created PUF* f *and* $x \in \mathcal{X}$*, it holds that:*

$$\Pr\left[\max(Dist(y_i, y_j)_{i \neq j}) \leq \delta_1\right] = 1 - \epsilon(\lambda). \tag{3.2}$$

Requirement 3.2 (δ_2-Collision Resistance)**.** *Whenever a single classical PUF:* $f \leftarrow \mathcal{MP}_C(\lambda)$ *is evaluated on different inputs, the minimum distance between any two outputs* $y_i \leftarrow f(x_i)$ *and* $y_j \leftarrow f(x_j)$ *is at least* δ_2*. That is for a created PUF* f *and* $x_i, x_j \in \mathcal{X}$*, it holds that:*

$$\Pr\left[\min(Dist(y_i, y_j)_{i \neq j}) \geq \delta_2\right] = 1 - \epsilon(\lambda). \tag{3.3}$$

Requirement 3.3 (δ_3-Uniqueness)**.** *Whenever any two classical PUFs:* $f_i \leftarrow \mathcal{MP}_C(\lambda)$ *and* $f_j \leftarrow \mathcal{MP}_C(\lambda)$ *are evaluated on a single, fixed input, the minimum distance between any two outputs* $y_i \leftarrow f_i(x)$ *and* $y_j \leftarrow f_j(x)$ *is at least* δ_3*. That is for a created PUF* f *and* $x \in \mathcal{X}$*, it holds that:*

$$\Pr\left[\min(Dist(y_i, y_j)_{i \neq j}) \geq \delta_3\right] = 1 - \epsilon(\lambda) \tag{3.4}$$

where $Dist(.,.)$ *is a general notion of distance between the responses. For correctly modelling CPUFs,* $\delta_1 < \delta_2$ *and* $\delta_1 < \delta_3$ *are necessary conditions to allow for a clear distinction between different inputs and different CPUFs.*

We also adapt the notion of *randomness* (also denoted as min-entropy) for the classical PUF f. It says the maximal probability of $p_x^f(y)$ with an input $x_j \in \mathcal{X}$ on PUF f_i where $i \in \mathcal{R}$, conditioned on the residual output space. A formal definition is as follows.

Definition 3.3 (p-Randomness)**.** *We define the p-Randomness of a classical PUF* $f : \mathcal{R} \times \mathcal{X} \to \mathcal{Y}$ *as*

$$p := \max_{\substack{x \in \mathcal{X} \\ y \in \mathcal{Y}}} p_x^f(y). \tag{3.5}$$

3.3.3 Quantum PUF Constructions

Compared to CPUFs, one significant characteristic of quantum PUFs (QPUFs) is that the evaluation of the inherent randomness of hardware devices can be described specifically via the quantum process. Similarly to CPUFs, we use the same term, challenge-response pairs (CRPs), while describing the input and output of QPUFs. However, unlike CPUFs with classical CRPs, the CRPs of QPUFs can be denoted and recorded with quantum states. Nevertheless, since the database for CRPs in quantum states will require a large size of quantum memory, we observe in the following that some proposals for QPUF constructions for implementations include the procedures of classical readout of quantum states to get rid of the overhead of quantum memory in practice. We argue that these proposals should also be classified as QPUFs in our thesis, as they can be naturally queried with quantum states. Therefore, we denote the CRPs as quantum states in the modelling of quantum PUFs in general.

The motivation for developing QPUFs in theory and practice comes from the fact that the quantum systems themselves exhibit inherent randomness and unpredictability due to the quantum phenomena, e.g., superposition and entanglement. Meanwhile, the no-cloning theorem guarantees that using quantum states as challenges and responses cannot be perfectly copied, while CRPs in bitstring can be intercepted and copied by adversaries in classical channels perfectly. However, we note that there exists the algorithm of quantum emulation called universal quantum emulator [124], which aims to mimic the behaviour of an unknown unitary transformation and give the corresponding output states upon unknown input quantum states based on some known input-output samples. In the case of QPUFs-based applications, it potentially allows an adversary to sabotage the security by means of emulating the unseen input-output behaviour of the target QPUF.

Optical PUFs and quantum readout PUFs. The idea of optical PUF constructions was originally introduced in [102], with the concept of POWF that we discussed previously. Here, the device of an optical PUF can be an inhomogeneous and transparent crystal/plastic token. It can accept the challenge as an illumination of laser from certain angles. When the laser passes through the token, the interference due to the scatters in the token is complicated and unpredictable. The beam out from the token is projected as a 2D speckle pattern, and then a hash function reduces the pattern to a bitstring as the response. In [125, 126], the entropy of optical PUFs is further deeply studied. In the original setup, the CRPs of optical PUFs are read out classically. Note that in this case, the optical PUF-based

protocols in a remote setting are still communicated via classical channels among parties.

Based on the optical PUFs, the idea of *quantum readout PUFs* (QR-PUFs) is introduced in the work [127] that we query optical PUFs using a quantum state as a challenge and obtain its response as quantum states as well. Generally, the whole interaction process can be treated as the unitary evolution of a physical system with the external and internal subsystems, which represent the input-output behaviours of QR-PUFs. The external state is either a reflected state or a transmitted state described by the motion of particles toward or away from the challenger. Therefore, the changed internal state described as spin/polarization of particles is the response in quantum analogue. Meanwhile, the authors analyse the security of QR-PUFs against intercept-resend attacks in a remote setup.

Furthermore, Goorden et al. [128] propose an experimental setup of QR-PUFs to achieve a quantum-secure device authentication protocol. Moreover, the idea of QR-PUFs authentication protocol exploiting continuous variables encoding scheme is proposed in [129, 130]. These proposals are secure against an adversary who attempts to clone an unknown quantum challenge. However, these proposals are more or less impractical due to the overhead of quantum resources.

Classical readout quantum PUFs. The idea of classical readout quantum PUFs (CR-QPUFs) is proposed firstly by Phalak et al. [131], which can possibly be implemented on near-term NISQ devices in practice to authenticate the quantum hardware remotely from the cloud server, e.g., IBM, Regatti and D-Wave. The main idea is to leverage the inherent errors caused by the imperfection of NISQ devices as devices' unique fingerprints. Here, the adversaries are considered as either malicious servers or third-party providers that fail to allocate the targeted quantum computer requested by the client's program, i.e. the quantum program delegated by the client does not run on the targeted hardware that it is supposed to run on.

The authors propose two possible architectures of QPUFs in the paper. Both of them have fixed structures of the quantum circuit. The *Hadamard gate-based* CR-QPUF initials each qubit as $|0\rangle$ state and performs elementary single qubit operations to obtain each qubit as a superposition state. The measurement result in 0/1 of each qubit with a computational basis is not with equal probability but biased since the readout errors of flipped measurement results are usually unique according to the devices. The *decoherence-based* CR-QPUF, on the other hand, exploits the difference of decoherence times of qubits among devices to obtain responses as devices' signatures.

Note that the responses of both constructions are measured using statistical query (SQ) models to obtain a robust QPUF in tolerating the extra noise, e.g., time-dependent noise. However, in a recent work [132], the authors propose an attack strategy where an adversary as an eavesdropper can fit a bounded degree polynomial function for each qubit via least square error regression and eventually obtain a model function for each qubit with a small size of collected CRPs.

Other quantum PUFs Another proposal for QPUF design is based on unitaries sampled from the Haar measure, furthermore using approximate t-designs, with

CRPs in quantum states [133]. Compared to Haar measure that the construction scales exponentially with the size of the input, the latter proposal is advantageous in using relatively simple quantum circuits with circuit depth scaling polynomially with the size of the input and the order t of the design. For the implementation in practice, it shows a possible way by using the simple random quantum circuit model [134]. Meanwhile, an alternative implementation can possibly be achieved by the measurement-based quantum computing (MBQC) model.

3.3.4 Quantum PUF Modelling

For the modelling of QPUFs, we refer to the work of [135] to give a formal description. Similarly to the modelling of CPUF in previous, we define a manufacturing process $\mathcal{MP}_Q$ with security parameter λ for the creation of a QPUF $\mathcal{E}_R \leftarrow \mathcal{MP}_Q(\lambda)$ with a random coin R, where $R \leftarrow \mathcal{R}$ and $\mathcal{R}$ is the space of identifier. The QPUF $\mathcal{E}_R$ takes any quantum state $\rho_{in} \in \mathcal{H}^{d_{\text{in}}}$ as input and maps to a quantum state $\rho_{out} \in \mathcal{H}^{d_{\text{out}}}$ as output as $\rho_{out} \leftarrow \mathcal{E}(\rho_{in})$. Finally, we introduce the requirements that QPUFs, in general, should satisfy with some threshold δ_i and a negligible function $\epsilon(\lambda) \leq \lambda^{-c}$ as follows:

Definition 3.4. *The quantum PUF $\mathcal{E} : \mathcal{H}^{d_{\text{in}}} \to \mathcal{H}^{d_{\text{out}}}$ with $(\mathcal{MP}_Q, \delta_r, \delta_u, \delta_c, \epsilon, \lambda)$ satisfies the requirements defined below:*

Requirement 3.4 (δ_r-Robustness)**.** *Whenever a single quantum PUF: $\mathcal{E} \leftarrow \mathcal{MP}_Q(\lambda)$ is evaluated with two input states ρ_{in} and σ_{in}, where the fidelity $\delta_r \leq F(\rho_{in}, \sigma_{in}) \leq 1$, the corresponding outcome quantum states ρ_{out} and σ_{out} are δ_r-indistinguishable, where:*

$$\Pr[\delta_r \leq F(\rho_{out}, \sigma_{out}) \leq 1] = 1 - \epsilon(\lambda) \tag{3.6}$$

Requirement 3.5 (δ_c-Collision Resistance)**.** *Whenever a single quantum PUF: $\mathcal{E} \leftarrow \mathcal{MP}_Q(\lambda)$ is evaluated with two input states ρ_{in} and σ_{in}, where the fidelity $0 \leq F(\rho_{in}, \sigma_{in}) \leq 1 - \delta_c$, the corresponding output states ρ_{out} and σ_{out} are δ_c-distinguishable, where:*

$$\Pr[0 \leq F(\rho_{out}, \sigma_{out}) \leq 1 - \delta_c] = 1 - \epsilon(\lambda) \tag{3.7}$$

A weaker variant of Collision-Resistance, with separate input/output bound can also be defined in a similar fashion where the responses on any two input state ρ_{in} and σ_{in} where $0 \leq F(\rho_{in}, \sigma_{in}) \leq 1 - \delta_c^i$, their corresponding output should satisfy $0 \leq F(\rho_{out}, \sigma_{out}) \leq 1 - \delta_c^o$. In fact, if $\delta_c^i = \delta_c^o = \delta_c$ we call the requirement a strong collision-resistance. Note that this equality holds up to a negligible value in the security parameter.

Requirement 3.6 (δ_u-Uniqueness)**.** *Whenever any two quantum PUFs: $\mathcal{E}_i \leftarrow \mathcal{MP}_Q(\lambda)$ and $\mathcal{E}_j \leftarrow \mathcal{MP}_Q(\lambda)$ are given by $\mathcal{MP}_Q(\lambda)$, the corresponding CPTP maps are δ_u-distinguishable:*

$$\Pr[\|(\mathcal{E}_i - \mathcal{E}_j)_{i \neq j}\|_\diamond \geq \delta_u] = 1 - \epsilon(\lambda). \tag{3.8}$$

, where $\|\cdot\|_\diamond$ is the diamond norm distance measure for the distinguishability between any two QPUFs.

For correctly modelling QPUFs, $\delta_r < \delta_c$ and $\delta_r < \delta_u$ are necessary conditions to allow for a clear distinction between different inputs and different QPUFs.

The notion of randomness for QPUFs modelling is characterised by the definition of *unknownness*. A formal definition is as follows:

Definition 3.5 ($\epsilon, \lambda, \delta$-Unknownness)**.** *We say that a QPUF $\mathcal{E}$ is $(\epsilon, \lambda, \delta)$-unknown, if for any quantum polynomial time (QPT) adversary $\mathcal{A}$, before making any query to $\mathcal{E}$, the probability of outputting a response $\mathcal{A}(\rho_{\text{in}})$ with fidelity at least $1-\delta$ with respect to the ideal response $\mathcal{E}(\rho_{\text{in}})$ on every state $\rho_{\text{in}} \in \mathcal{H}^{d_{\text{in}}}$ is bounded by:*

$$Pr[F(\mathcal{A}(\rho_{\text{in}}), \mathcal{E}(\rho_{\text{in}}) \geq 1-\delta) | \rho_{\text{in}} \in \mathcal{H}^{d_{\text{in}}}] \leq \epsilon(\lambda) \tag{3.9}$$

3.4 Trusted Execution Environments

Here, we define an execution environment that allows different applications to run on top of the hardware layer of a device. With the increasing requirements to address various security and privacy challenges against different means of sophisticated attacks by adversaries in the age of modern computing, the idea of *trusted execution environment* (TEE) is proposed as a role of a secure processor to provide an isolated execution environment with necessary components in a device that achieves a higher level of security for executing sensitive operations and preventing critical data from unauthorised access. The main feature of TEE is that it is a hardware-based security design that aims to separate different execution environments on the device and create a trusted and isolated space apart from a device's main but potentially untrusted execution environment, where only intercommunication with careful flow control is allowed (As shown in Figure 3.4 a general architecture). That is to say, TEE's physical root-of-trust comes from the hardware *isolation*.

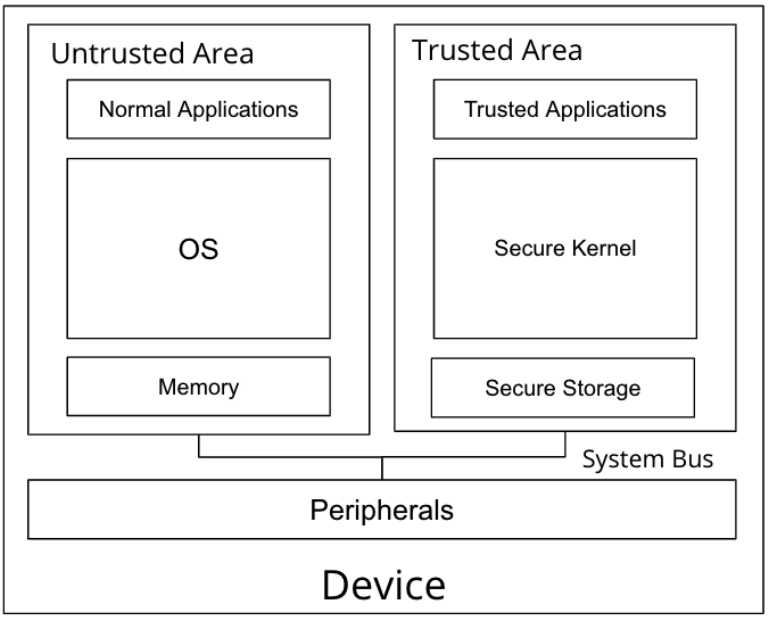


Figure 3.4: **TEE with co-existing execution environments.** The trusted area on the right-hand side includes all trusted components used for executing trusted functions. Everything outside the trusted area are treated to be untrusted.

Note that the term TEE is standardised specifically by the consortium Global Platform as a secure area of a main processor that helps code and data loaded inside it to be protected concerning confidentiality and integrity [136]. The need for standardisation comes from the increasing need for secure and isolated environments within computing systems. Nevertheless, the fundamental concept of isolation and building blocks of isolation on the hardware level with different techniques can be traced back for several decades, including smart cards, trusted platform modules (TPMs), ARM TrustedZone, Intel SGX, etc. We will review these different constructions for a more complete comprehension of the accessibility and necessity of TEEs nowadays.

Except for physical isolation, to guarantee the trust of TEEs in practice, there are multiple dedicated measures from the perspectives of hardware architectures and software management according to different manufacturers and security mechanisms. Here, we focus on the abstractions of functionalities that capture the characteristics of TEEs' execution. Surprisingly, most of the TEEs have converged on providing a common model as the *attested execution* functionality [137, 138, 139, 140, 141, 142], regardless of the different specifications. It forms together with the physical isolation to provide a high level of security for sensitive applications and data, protecting them from potential threats. For a more detailed description of the attested execution functionality, we introduce the modelling in Section 3.4.1.

As to the quantum TEEs, we are the first to introduce the idea of TEE in quantum analogue. The initial idea of quantum TEE is similar to the classical case: It provides a secure area that resides in a quantum computing system that executes quantum operations with confidentiality and integrity to secure the quantum computing tasks. Even though a sophisticated architecture of quantum computing systems is still under development, we believe that the role of quantum TEE is crucial in developing quantum computing. The reason is straightforward: If the quantum hardware devices themselves are compromised in the first place. It is meaningless to discuss the quantum advantages shown in most device-dependent quantum applications, although a robust universal quantum computer can be built up with enormous expense and huge efforts in the near-term future.

Meanwhile, we evaluate the role of quantum TEE from different aspects: Firstly, since the usage of quantum computing devices for applications in the near-term future will be in the form of cloud service, we develop the quantum TEE construction and the security analysis in the scenario of quantum cloud computing. Furthermore, since the quantum computing system is expected to be complicated regarding the operating mechanism and hardware architecture, minimizing a quantum TEE's complexity while the security is not compromised is another important criterion from the practicality perspective. Explicitly speaking, a less complex TEE reduces the potential attack surface for adversaries to bypass security mechanisms. In contrast, it is hard to verify the integrity of TEE from every possible attack vector if the overall mechanism is too complicated. With these perspectives in mind, we believe the specification of quantum TEE is realistic in consideration and practical in real-world implementation.

3.4.1 Classical TEE Constructions

In this section, we take a look at the different constructions and security mechanisms that provide a secure area utilizing hardware isolation, from isolated cryptographic modules to security extensions on CPUs based on isolation. All these generally define a trusted environment that is isolated from an untrusted environment, as the definition of TEE.

Smart cards A widely used hardware device that provides an isolated environment for executing applications and security storing data is a *smart card*. It is a hardware token integrated with an embedded microcontroller, dedicated storage areas and hardware-accelerated cryptographic modules. The definition of the trusted and untrusted areas is trivial: Every component on the smart card belongs to the trusted area, while the external environment of the smart card is supposed to be the untrusted area (including the host devices). We can consider the structure of a smart card as a complete physical separation between trusted and untrusted areas most of the time. In this case, the feature of the smart card functionalities should be as simple as possible to achieve practicality and a high level of tamper resistance.

Smart cards are commonly used in applications in daily life. For example, credit cards are used for secure electronic transactions, ID cards are used for secure digital identity, key management kits are used for digital signatures, etc.

Trusted platform modules A trusted platform module (TPM) is a hardware module designed to enhance the security of computing platforms by providing various secure cryptographic operations. The Trusted Computing Group has specified the most common standardisation since 2009 [143]. As a crucial role in ensuring the integrity, authenticity, and confidentiality of computing systems, TPMs are used in a wide range of devices, including embedded systems, laptops, desktops, servers, etc. [144]

Similar to the concept of TEE, TPMs aim to create isolated environments that are separated from the main operating system and other software components by leveraging hardware-based security mechanisms to protect against various threats, including unauthorised access and tampering. From the point of view of different environments, the trusted area includes engines for cryptographic operations (e.g., SHA-1 engine, RSA engine, HMAC engine, etc. and a random number generator), an isolated execution engine, platform configuration registers, and persistent memory for identification. The information flow between trusted and untrusted areas is managed by a gatekeeper mechanism, which enforces access control policies to the TPMs. However, TPMs cover a larger range of security functions related to platform integrity, secure boot, cryptographic services and attestation [145], while TEEs are specified more focusing on creating a secure execution environment for specific applications or tasks by isolating them from the rest of the system.

In practice, the implementations of TPMs are normally in the form of a separate hardware component or chip integrated as a peripheral of its corresponding host system by attaching it to the communications bus. In this case, the physical interface should not weaken the tamper-resistant of the TPM. Meanwhile, there are

proposals to implement a TPM in the first place with the processor with dedicated designs and mechanisms to circumvent possible attacks by attaching externally. Intel proposes *trusted execution technology* (TXT) as a hardware security extension to the Xeon processor family that cooperates with TPMs' security mechanisms [146]. Intel TXT enables the secure features of TPMs since the system's boot process, by the method of *Static root-of-trust for measurement* (SRTM). It measures the integrity of BIOS, bootloader, and operating system components to create a hardware-based root-of-trust.

Secure processors Without loss of generality, the concept of a *secure processor* is a general-purpose processor that provides security guarantees featured by TEE for crucial data and operations within it. In the thesis, we also refer to the TEE provided by secure processors as *secure enclave*. It offers a higher level of security than general-purpose processors by incorporating dedicated hardware mechanisms to safeguard against various attacks. Unlike TPMs that focus on establishing a secure platform by cryptographic functions and key management, secure processors offer a wider range of security features for the execution of sensitive operations in general within the isolated environment.

In the last twenty years, there have been multiple proposals of secure processors from the domain of research and industry [138, 147, 139, 148, 149, 142, 150, 151, 141, 152]. Meanwhile, more and more constructions are realised commercially, including ARM TrustZone [153, 154], Intel SGX [138, 155, 151], Apple secure enclave processor [156], etc. From the perspective of TEE, ARM TrustZone design specifies two execution environments: *Secure world* and *normal world*. The secure world plays the role of TEE and provides corresponding security mechanisms. Relatively, the normal world hosts the main operating system and applications that are assumed to be untrusted. Intel SGX leverages trusted hardware by isolation to create secure containers called enclaves, which provide isolated execution environments where sensitive code and data can be processed securely. It is designed especially to solve the problem of executing software on a remote computer of an untrusted party while guaranteeing integrity and confidentiality. Apple secure enclave processors focus on biometric data management and secure transactions within Apple devices by creating isolated execution environments for security-critical operations. Since the benchmarking of different secure processors is not the scope of our thesis, we encourage the readers to look into the reference for detailed information.

Nowadays, due to the increasing requirements for security and privacy on data of IoT and distributed computing scenarios [157, 158, 159, 160], more and more TEE-based hardware security techniques are exploited for securing IoT devices and edge computing nodes [161, 162, 163, 164]. At the same time, since the emergence of quantum computing, TEEs are expected to play a crucial role in post-quantum cryptography [165, 166], ensuring that sensitive data and operations remain secure against quantum attacks.

3.4.2 Classical TEE Modelling

Regardless of the different constructions of hardware-based isolation that can host TEEs, since a TEE provides a very strong guarantee of running sensitive operations

and protecting critical data within its secure and isolated environment, it essentially requires the mechanisms of *attestation* to establish trust and verify the integrity of the TEE's execution environment. Generally speaking, the attestation process allows external parties or services to assess the security properties of the TEE and the device it is running on. In other words, for external parties to trust the TEE, there needs to be a way to verify that the target TEE is indeed in a secure state and has not been compromised.

Furthermore, we note that attestation mechanisms enable TEEs to provide security guarantees in cloud computation scenarios. Recall that the problem of secure cloud computations requires a client to perform executions of software with some integrity and confidentiality guarantees on a remote computer owned by an untrusted server. While fully homomorphic encryption can solve this problem non-universally with huge overhead [167], TEEs with proper attestation mechanisms tackle this problem by leveraging trusted hardware to provide a secure and isolated environment for software executions. Meanwhile, the remote client can verify the integrity and authenticity of the TEE without needing direct access to its internal components.

As a result, in our thesis, we model the behaviours of TEEs by describing the functionality of *attested execution*. Since the versatility of secure processors, the model is adapted to the simulation-based security model. On the one hand, an ideal functionality of attested execution can effectively capture the core abstraction that a broad class of attested execution processors is supposed to provide. On the other hand, since the fact that the operations in secure processors should be with minimal complexity, we analyse the security of protocols by minimising security assumptions and have them executed in secure processors by the definitions of TEE. In this case, we can further analyse the security in composable frameworks, which ensures the reduction of security analysis from the whole protocol to the attested execution functionality works properly and rigorously with mathematical proofs. Meanwhile, the security still holds when other programs are executed in parallel, i.e., in general composition. We show more details of security analysis in Chapter 5.

Here, we introduce the ideal functionality G_{att} (See Functionality 1) of attested execution shown in [168] to formalize cryptographically the secure processors. G_{att} is parameterized by a signature scheme Σ and a register $\texttt{reg}$ that captures all parties P that equip with a secure enclave. For the activation points of G_{att}, the starred ones are reentrant activation points. Otherwise, it can only be executed once. In the registry stage, the secure processor enables a distribution of the manufacturer's trusted public key from key pair $(\texttt{mpk}, \texttt{msk})$ to P upon the query. For stateful enclave operations, the activation point $\texttt{install}$ denotes an installation of enclave application with a program $\texttt{prog}$ from P. It generates an identifier eid to P for identifying the enclave instance; the activation point $\texttt{resume}$ enables the execution of $\texttt{prog}$ upon the input $\texttt{inp}$ by G_{att}. G_{att} then signs the output $\texttt{outp}$ to be attested with $\texttt{msk}$ using Σ. The attestation σ is returned to P for verification.

Functionality 1 Classical TEE Modelling Attested Execution $G_{\text{att}}[\Sigma, \texttt{reg}]$

Registry:

// initialisation
On initialize: $(\texttt{mpk}, \texttt{msk}) := \Sigma.\texttt{KeyGen}(1^\lambda), T = \varnothing$.

// public query interface
On receive* $\texttt{getpk}()$ from some P : send $\texttt{mpk}$ to P.

Enclave Operations:

//install an enclave program
On receive* $\texttt{install}(\text{idx}, \texttt{prog})$ from some $P \in \texttt{reg}$:

- if P is honest, assert idx = sid.
- generate nonce eid $\in \{0,1\}^\lambda$, store $T[\text{eid}, P] := (\text{eid}, \texttt{prog}, 0)$, send eid to P.

//resume an enclave program
On receive* $\texttt{resume}(\text{eid}, \texttt{inp})$ from some $P \in \texttt{reg}$:

- let $(\text{idx}, \texttt{prog}, \texttt{mem}) := T[\text{eid}, P]$, abort if not found.
- let $(\texttt{outp}, \texttt{mem}) := \texttt{prog}(\texttt{inp}, \texttt{mem})$, update $T[\text{eid}, P] := (\text{idx}, \texttt{prog}, \texttt{mem})$.
- $\sigma := \Sigma.\texttt{Sig}_{\texttt{msk}}(\text{idx}, \text{eid}, \texttt{prog}, \texttt{outp})$, and send $(\texttt{outp}, \sigma)$ to P.

3.4.3 Quantum TEE Construction

The idea of TEE in quantum analogue intends to provide an isolated and secure execution environment for crucial quantum operations that may affect the security properties of quantum computing. In general, there are various security and privacy problems existing in different layers of quantum computing stacks from the hardware level to the application level [169]. Except for the unintentionally different types of noise caused by the limitation of technology on current quantum computing devices (NISQ devices), e.g., decoherence, gate operations errors, readout errors, crosstalks, etc., there are different threats indeed caused by untrusted quantum hardware services provided by the cloud servers. Since the infrastructures of quantum computing are most probably cloud-based access, it is especially important to have some integrity and confidentiality guarantees to the clients.

On the level of quantum hardware, more and more threats that can intentionally cause security and privacy issues are studied. One well-known threat from potentially untrusted servers is fault injection attacks, which intentionally introduce noises into the quantum operations, e.g., crosstalks and bit/phase flips. Unlike classical computations with sophisticated designs and architectures, quantum and classical controlling crosstalks are still a main challenge to scale up the size of quantum computing devices [170, 171, 172, 173]. In [174], the authors introduce an attack model based on crosstalk effects in NISQ devices by simulation and experimental results, assuming that the adversary can run his program on the same hardware as one or more programs from an honest client. In the case where the

adversary has reasonable capabilities, he can modulate the client's computation outcomes. For bit/phase flips fault injection, it can effectively influence the results of a certain class of delegate quantum computation protocols. We will further elaborate and demonstrate this kind of attack in Chapter 5 while developing our idea of quantum TEE and applications.

The idea of isolation is shown to tackle the issue of crosstalks effectively. In [174], the authors experimentally demonstrate that running parallel quantum programs on IBM quantum computers with buffer qubits in between can obtain a higher fidelity than the scenario of running parallel programs with adjacent qubits. However, the privacy issue still exists in this case since an untrusted server can still intercept the execution of quantum programs delegated by clients and steal crucial data that sabotages the security properties of computations. Meanwhile, leveraging the idea of TEE in quantum analogue to directly secure the whole quantum circuits is overhead and, therefore, impractical with near-term quantum hardware techniques.

While there are proposals to tackle the privacy issues, e.g., obfuscation of quantum circuits with dummy gates [175, 176] or split compilation by division of a quantum circuit into multiple parts [177], we develop the idea of isolation a step further by exploiting the idea of TEE. Here, we give a general description of our idea of a quantum TEE construction that can be applied in quantum computing applications in the following.

> *"A quantum TEE intends to provide a physically isolated environment to perform quantum operations with security guarantees against intentional adversarial behaviours, with minimal hardware extensions."*

According to the hardware-based isolation features of TEE, quantum computation and quantum information technologies, our point of view on the possible constructions of quantum TEE should satisfy the following guidelines:

> **Postulate 1:** The quantum operations within the quantum TEE should be simple with high quantum fidelity, e.g., a single-qubit gate, with classical controlling peripherals with security and privacy guarantees.

The advantages of this construction prototype are in many folds: First, it reduces the quantum crosstalks, where an operation on a qubit might modulate another qubit state. Secondly, the intentionally classical crosstalks from the incorrect controls can be constrained by the state-of-the-art classical constructions of TEEs and the other types of errors, e.g., bit or phase flips.

> **Postulate 2:** The operations within a quantum TEE should be compatible with scalable quantum computation scenarios. Meanwhile, the security properties can still be preserved.

Based upon the first argument, if the operations within a quantum TEE are limited to only a single-qubit gate operation, these operations should be reproducible under certain controls and meaningful in the quantum computation protocols with some privacy guarantees. By repeating the operations in the quantum TEE, the computation is scalable.

Postulate 3: With near-term quantum devices, the quantum memory components should be excluded in a quantum TEE construction.

Although recent works are showing the implementation of quantum memory is less susceptible to decoherence [178, 179, 180, 181, 182] with different quantum hardware platforms, we still think that a quantum TEE should be kept with minimal overhead on hardware complexity while considering the security properties. A quantum memory component, in this case, does satisfy this criterion.

3.4.4 Quantum TEE Modelling

Similar to the classical TEE modelling in Functionality 1, we model the quantum TEE with minimal extension of the protected quantum operations in G_{att}'s enclave operations. For the classical input/output, the cryptographic operations remain the same as classical TEE modelling. Meanwhile, the information can include the classical controlling of quantum operations. As for the quantum interactions between the trusted and untrusted environments, the quantum TEE construction only guarantees that the operations upon the incoming quantum states are with confidentiality and integrity.

Functionality 2 Quantum TEE Modelling Attested Execution $G_{\text{att}}[\Sigma, \texttt{reg}]$

Registry:

// initialisation
On initialize: $(\texttt{mpk}, \texttt{msk}) := \Sigma.\texttt{KeyGen}(1^\lambda), T = \varnothing$.

// public query interface
On receive* $\texttt{getpk}()$ from some P : send $\texttt{mpk}$ to P.

Enclave Operations:

//install an enclave program
On receive* $\texttt{install}(\text{idx}, \texttt{prog})$ from some $P \in \texttt{reg}$:

- if P is honest, assert idx = sid.
- generate nonce eid $\in \{0,1\}^\lambda$, store $T[\text{eid}, P] := (\text{eid}, \texttt{prog}, 0)$, send eid to P.

//resume an enclave program
On receive* $\texttt{resume}(\text{eid}, \texttt{inp})$ from some $P \in \texttt{reg}$:

- let $(\text{idx}, \texttt{prog}, \texttt{mem}) := T[\text{eid}, P]$, abort if not found.
- let $(\texttt{outp}, \texttt{mem}) := \texttt{prog}(\texttt{inp}, \texttt{mem})$, update $T[\text{eid}, P] := (\text{idx}, \texttt{prog}, \texttt{mem})$.
- $\sigma := \Sigma.\texttt{Sig}_{\texttt{msk}}(\text{idx}, \text{eid}, \texttt{prog}, \texttt{outp})$, and send $(\texttt{outp}, \sigma)$ to P.
- $\rho_{out} = U(\texttt{inp})\rho_{in}U^\dagger(\texttt{inp})$, send ρ_{out} to P or keep ρ_{out} locally for further computations depend on the specification of protocols.

Here, we give a description of the ideal functionality of quantum TEE enabled by G_{att}, with quantum state $\rho_{in} \in \mathcal{H}^{d_{\text{in}}}$ as input, and maps to a quantum state

$\rho_{out} \in \mathcal{H}^{d_{\text{out}}}$ as output via unitary transformation $U(\texttt{inp})$ controlled by classical input within quantum TEE.

Note that in quantum cloud computing scenarios with G_{att}, an untrusted server might maliciously prepare an initial quantum state ρ_{in} entangled with ancilla, forward a subsystem as the quantum input, and try to steal the information from the trusted environment by learning the entire states after the operation within a quantum TEE. Meanwhile, since the server has quantum computing capabilities, the implementations of cryptographic schemes within G_{att} should be considered with post-quantum security. With these potential threats in hand, we discuss in detail our proposed constructions with careful security analysis in Chapter 5.

Chapter4

Physical Unclonable Function

Practical Constructions with Quantum Communication Advantages

4.1 Introduction

IN the previous chapter, we review the state-of-the-art research and implementations of physical unclonable functions (PUFs) as a type of hardware security primitive in both classical and quantum analogues, meanwhile with the corresponding mathematical modellings of PUFs that capture the characteristics of PUFs' expected input-output behaviours. Generally speaking, a PUF derives unique volatile secret keys on the fly by exploiting the inherent random variations introduced by the manufacturing processes. Any slight (yet unavoidable and uncontrollable) variation in the manufacturing process produces a different PUF, rendering the fabrication of an identical physical "clone" of a PUF [183] infeasible. Hence, PUFs provide copy-proof, cost-efficient, unique hardware fingerprints.

To generate such fingerprints, in the classical setting, a PUF can be described as a classical function that a user can query with an input classical bitstring as a challenge, producing an output bitstring as a response. We refer to the query and response pairs as challenge-response pairs (CRPs). A classical database of CRPs is necessary and important in authenticating these fingerprints during usage.

Although CPUFs can provide unique and inexpensive hardware fingerprints. However, many CPUF constructions are vulnerable against modelling attacks, as shown in Section 3.3.1. In these types of attacks, the main idea is that the attacker first collects a sufficient number of CRPs by adaptively querying the PUF and then uses that data to derive a numerical model using the tools, e.g., different machine learning algorithms. Here, the goal of the model is to predict the response of the PUF to an arbitrary challenge. At the same time, the classical transmission of CRPs can be copied freely by an eavesdropper without detection in a wide range of CPUF-based applications.

The idea of quantum PUFs, on the other hand, is one of the most appropriate directions to look for solutions. By considering the evaluation of PUFs is now described as a quantum process, with quantum states as challenges and responses. As discussed, leveraging unclonability as a fundamental property of quantum systems can potentially benefit us in achieving a stronger notion of physical unclonability. Furthermore, the inherent random variations of PUFs are always observed at the atomic and subatomic levels, which follow the laws of quantum mechanics. There-

fore, learning these existing inherent randomnesses in terms of quantum analogue helps a deeper understanding of the physical unclonability phenomena on PUFs. Finally, the recent advances in quantum cryptography remind us of a potential enhancement of the security of classical PUFs combined with quantum information technologies.

However, there are other concerns about quantum PUFs, especially in practical usages. Firstly, quantum PUFs are limited by their scalability compared to classical PUFs due to their specialized structures and the high cost of the manufacturing process. Besides, recall that either the authentication of quantum PUFs' fingerprints or the secure usage of generated secret keys of quantum PUFs in other cryptographic applications requires a large-size quantum memory as the database to store CRPs as quantum states. Since current quantum memory is always with a high cost, meanwhile sensitive to the environment [184]. Recent research targets removing the requirement of quantum memory while using quantum PUFs. However, as shown in the last chapter, recording quantum challenges and responses via measurements can result in privileging adversaries to perform attacks trivially based on statistical query models, which violates the motivation of using quantum PUFs to enhance security.

Another main concern for both classical and quantum PUFs is that an adversary can query adaptively with arbitrary challenges, especially in network applications. On the one hand, it constrains that each CRP of CPUFs can be used only once in the vanilla setting. Otherwise, an eavesdropper can just reuse the previously used response. On the other hand, multiple copies of a quantum state obtained adaptively by an adversary permit the extraction of the classical description of the state in terms of the statistics of the measurement outcomes.

Considering both the advantages and disadvantages of classical/quantum PUFs and the problems raised above, we introduce in this chapter a new direction for using classical PUFs in combining quantum information technologies efficiently with several aspects: A new PUF construction and a novel quantum entity authentication protocol that exploits the combination of hardware security primitive of PUF and quantum information to achieve secure authentication with provable exponential security advantage compared to its classical counterparts. We also formally prove that the protocol fulfils a specific desired property, namely, *challenge reusability*, which is impossible unless using quantum communication, emphasizing the significance of quantum communication technology and quantum network for a new quantum security era. Moreover, we show that quantum communication makes our construction *cheat-sensitive*, i.e., our PUF-based authentication protocol can detect the adversarial attempts (both passive and active) on intercepting the responses of the PUF. We aim to keep our construction implementable using present-day quantum communication technologies while exploiting the desirable security promises that are provided due to the quantum nature of the challenges and responses. Our PUF construction utilises classical PUFs, which are too weak to be useful in a standalone manner, but present the advantage of being widely accessible and easy to use, and enhances their security using commercially available tools from quantum communication. Here, for the first time, we show that by encoding the output of classical PUFs into non-orthogonal qubits, one can enhance the security of PUFs against weak (non-adaptive) adversaries.

As such, the first building block of our design is a construction we refer to as *hybrid PUFs* (HPUFs), which encompasses a classical PUF and produces quantum responses for classical challenges. We prove that this construction provides security against the mentioned adversary. With this gadget in hand, we then introduce a construction that is secure against more powerful adaptive quantum adversaries (the general class of quantum polynomial-time (QPT) adversaries). To this end, we borrow the idea of the classical lockdown technique [185] and, by redefining it in the quantum setting, we present our final construction, namely *hybrid locked PUF* (HLPUF). We show that classical PUFs combined with quantum encoding and the new lockdown toolkit can considerably boost the security of classical PUFs without too much overhead. An important technological improvement compared to previous quantum-enhanced proposals where quantum memory was necessary is that for both HPUFs and HLPUFs, only a classical database of challenge-response pairs needs to be stored on the verifier's side. We formally prove adversarial bounds on the unforgeability of HPUF and HLPUF constructions in comparison with the underlying classical PUFs, using rigorous proof techniques from quantum information theory. Furthermore, we also formally prove the security of our HLPUF-based device authentication protocol under realistic hardware assumptions.

In addition to our theoretical contributions, to better demonstrate the applicability and strength of our results, we provide simulations for the design of HPUF constructions with underlying silicon CPUFs instantiated by the *pypuf* python-based library [186]. Furthermore, we simulate machine-learning-based modelling attacks on HLPUFs where an adversary acquires classical challenges and quantum-encoded responses from an HLPUF. Our simulation results assist in demonstrating our theoretical proofs by evidencing the security enhancement from CPUFs to HLPUFs. Another significance of our simulation results is that they certify the practicality and security of our construction, even beyond the scope of the proven theorems, in a real-world scenario, as the CPUFs used in our simulations are commercially available and not only theoretical models. We also bring forward practical proposals to improve the quality of such constructions further.

Finally, through studying this construction, we will also address a long-standing open problem in the field of PUF-based authentication, which is the reusability of challenge-response pairs stored in the verifier database. One significant drawback of PUF-based authentication protocols is that the server/verifier cannot use the same challenge multiple times to authenticate a client/prover due to man-in-the-middle attacks. Therefore, the server exhausts all the challenges from the database after running several rounds of the authentication protocol. This limitation is unavoidable in any such classical protocols.

However, we show that due to the entropy uncertainty principle in quantum information theory, with our proposed construction, the server can reuse a challenge as long as they can successfully authenticate the client using that challenge in the previous rounds. Our result overcomes this open problem as we prove the challenge reusability of PUF-based applications for the first time. The entropy uncertainty principle also allows the honest server/client to detect any adversarial attempts to extract information from the response of the HPUFs, providing the cheat sensitivity of our protocol.

4.1.1 Structure of the Chapter

In Section 4.2, we first describe the new hybrid construction called hybrid PUF (HPUF) with an underlying CPUF and quantum encoding in 4.2.1. The idea is to consider both the practicality of implementation and the security enhanced by quantum information technology. Furthermore, we show in 4.2.2 another construction of HPUF against a general adaptive adversary with the protection of lockdown technique (HLPUF). Section 4.3 introduces the adversarial models for security analysis related to our research. In detail, we give the unforgeability game schemes that capture the security requirements of PUFs and the regulation of adaptive and non-adaptive (weak) adversaries' behaviours that we consider in our work. In Section 4.4, on the one hand, we show the security of HPUF against a non-adaptive network adversary. On the other hand, we show the security of HLPUF against an adaptive adversary with more power. The idea is to reduce the power of network adversaries from adaptiveness to non-adaptiveness. To support our proofs, we give in Section 4.5 the numerical simulations with models of underlying CPUFs and alternative quantum encoding methods. In Section 4.6, we show our proposal for an HLPUF-based authentication protocol with security guarantees and reusability property. Finally, we conclude this chapter in Section 4.7.

4.2 Hybrid (Locked) PUF Constructions

In the previous chapter, we summarise the recent research works and state of the arts of both classical and quantum PUFs. On the one hand, CPUFs, especially most of the strong PUFs based on silicon design, are threatened by modelling attacks with different technologies nowadays. For weak PUFs, the limitation of indirect access means that the security of using a weak PUF depends on the cryptographic function that consumes the weak PUF's output. On the other hand, the proposed constructions of quantum PUFs nowadays are either impractical or vulnerable. Furthermore, a proper QPUF might require a database of CRPs in quantum states, i.e., the requirement of quantum memories, according to the description in [135]. These existing bottlenecks raise a question: Is there a possible construction of PUFs that can enhance security by quantum information technology while being practical to implement with near-term available quantum devices? In the rest of this chapter, we answer this question affirmatively by introducing the idea of hybrid PUF and its related constructions and applications.

4.2.1 Hybrid PUF Construction

Here, we propose a hybrid PUF (HPUF) design that aims to protect the output interface of the classical PUF by encoding the classical outcomes in non-orthogonal states. Thus, an HPUF can be treated as a device with classical bit-string as input and encoded quantum states as outputs.

One example of HPUF construction can generally be with a CPUF $f : \{0,1\}^n \to \{0,1\}^{4m}$ that has a certain amount of randomness, as min-entropy. To increase the min-entropy further, we encode the output of the CPUF into non-orthogonal $2m$

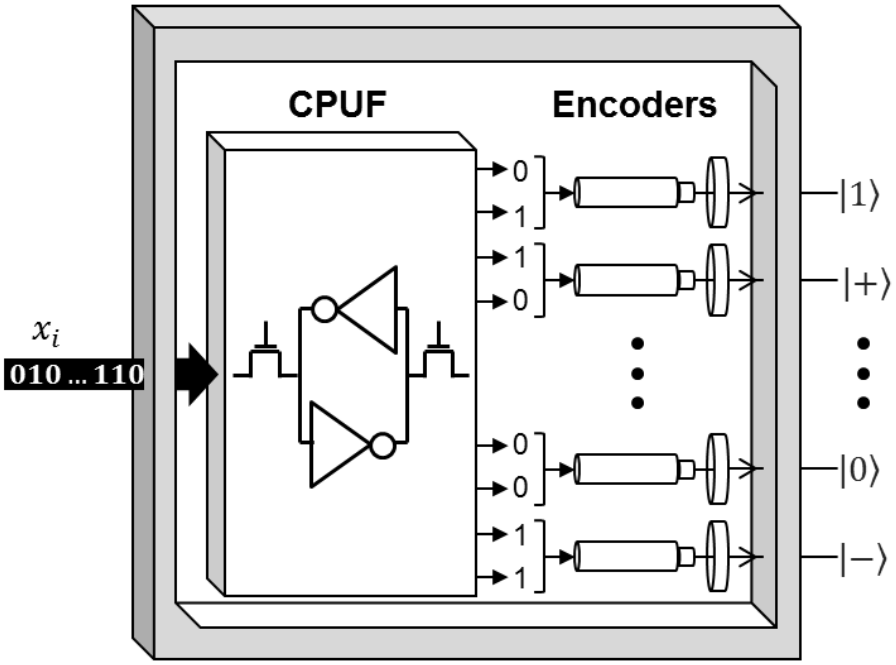


Figure 4.1: HPUF Construction with Conjugate Coding

qubits with $2m$ pairs of classical bits and send the qubits through the communication channel. Each pair takes the $(2j-1)$-th, and the $2j$-th (where $1 \leq j \leq 2m$) output bits. Next, we define a two-to-one mapping of the tuple (y_{2j-1}, y_{2j}) of f's outcome to a qubit $|\psi^j_{\text{out}}\rangle \in \{|0\rangle, |1\rangle, |+\rangle, |-\rangle\}$, as BB84 states. Here, the entire system, i.e., CPUF and the quantum encoding, is referred to as the hybrid PUF construction. We illustrate the HPUF construction in Figure 4.1 and give the formal description of our HPUF design as Construction 4.1, which is based on conjugate coding [187].

Construction 4.1 (Hybrid PUF)**.** *Suppose $f : \{0,1\}^n \to \{0,1\}^{4m}$ be a classical PUF, that maps an n-bit string $x_i \in \{0,1\}^n$ to an $4m$-bit string output $y_i \in \{0,1\}^{4m}$. We denote the j-th bit of y_i as $y_{i,j} \in \{0,1\}$. From the $4m$-bit string, we prepare the set of $2m$-tuples $\{(y_{i,(2j-1)}, y_{i,2j})\}_{1 \leq j \leq 2m}$. The hybrid PUF encodes each of the tuples $(y_{i,(2j-1)}, y_{i,2j})$ into a single qubit $|\psi^{i,j}\rangle$ (also known as* BB84 states*). The exact expression of the encoding is defined in the following way,*

$$|\psi^{i,j}_{\text{out}}\rangle\langle\psi^{i,j}_{\text{out}}| := \begin{cases} |0\rangle\langle 0| & (y_{i,(2j-1)}, y_{i,2j}) = (0,0) \\ |1\rangle\langle 1| & (y_{i,(2j-1)}, y_{i,2j}) = (1,0) \\ |+\rangle\langle +| & (y_{i,(2j-1)}, y_{i,2j}) = (0,1) \\ |-\rangle\langle -| & (y_{i,(2j-1)}, y_{i,2j}) = (1,1) \end{cases} \tag{4.1}$$

For any $x_i \in \{0,1\}^n$, the mapping of the HPUF $\mathcal{E}_f : \{0,1\}^n \to (\mathcal{H}^2)^{\otimes 2m}$ is defined as follows.

$$x_i \to |\psi^i_{\text{out}}\rangle\langle\psi^i_{\text{out}}| \quad (\textit{or } |\psi_{f(x_i)}\rangle\langle\psi_{f(x_i)}|) \tag{4.2}$$

where $|\psi^i_{\text{out}}\rangle\langle\psi^i_{\text{out}}| = \bigotimes_{j=1}^{2m} |\psi^{i,j}_{\text{out}}\rangle\langle\psi^{i,j}_{\text{out}}|$.

4.2.2 Hybrid Locked PUF Construction

Furthermore, we complete our proposed construction by equipping HPUFs with a mechanism called *quantum lock*. In brief, the quantum lock is a mechanism of the embedded small verification resources. This construction aims to restrict the adversary from adaptively querying the device and reduces a QPT adversary to a weak adversary. Here, An adaptive quantum adversary is free to build their database with any arbitrary query and in an adaptive manner, potentially depending on the previous queries[1]. Particularly, such adversaries can query HPUF multiple times with the same challenge x, obtaining several copies of $|\psi_{\text{out}}\rangle$ and can easily extract the outcome $f(x)$ from multiple copies. Hence, the construction of HPUFs alone is insufficient to achieve the most compelling desired notion of quantum security. Regarding this issue, we give a detailed security analysis in the next section.

We start by subdividing the output of the HPUF $\mathcal{E}_f : \{0,1\}^n \to (\mathcal{H}^2)^{\otimes 2m}$ corresponding to a classical PUF $f : \{0,1\}^n \to \{0,1\}^{4m}$ into two different parts, where $\mathcal{H}^d$ denotes a d-dimensional Hilbert space of quantum states. The first part contains the first m qubits, and the second half contains the last m qubits of the outcome of the HPUF $\mathcal{E}_f$. Note that the first m qubits of the HPUF's outcome come from the first $2m$ bits outcome of the underlying classical PUF f. For any challenge $x \in \mathcal{D}^{in}$, we can write the outcome of the classical PUF as $f(x) = f_1(x)||f_2(x)$, where the mapping $f_1 : \{0,1\}^n \to \{0,1\}^{2m}$ denotes the first $2m$ bits of f and $f_2 : \{0,1\}^n \to \{0,1\}^{2m}$ denotes the last $2m$ bits of f. Similarly, we can rewrite the HPUF $\mathcal{E}_f$ as a tensor product of two mappings $\mathcal{E}_{f_1} : \{0,1\}^n \to (\mathcal{H}^2)^{\otimes m}$, and $\mathcal{E}_{f_2} : \{0,1\}^n \to (\mathcal{H}^2)^{\otimes m}$, where for any challenge $x \in \{0,1\}^n$, $\mathcal{E}_{f_1}(x)$ denotes the first m qubits of $\mathcal{E}_f(x)$, and $\mathcal{E}_{f_2}(x)$ denotes the last m qubits of $\mathcal{E}_f(x)$.

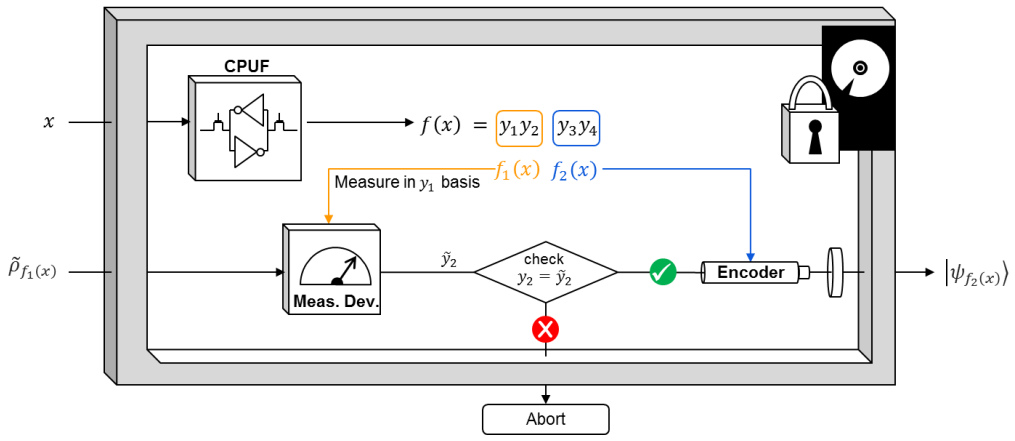


Figure 4.2: HLPUF construction: HLPUF uses an HPUF, a single-qubit quantum encoder device and a single-qubit measurement device, all inside a tamper-proof environment which prevents any quantum adversary from adaptively querying the HPUF.

The HLPUF takes the classical input x_i and a quantum state $\tilde{\rho}_1$ and produces the second half of the response of the hybrid PUF, $|\psi_{f_2(x_i)}\rangle \langle\psi_{f_2(x_i)}|$, as an output if

[1]Note that here we don't allow superposition queries to the underlying CPUF inside the HLPUF. However, we allow the adversaries to run quantum algorithms on the challenge-response pair database.

$\tilde{\rho}_1$ is equal to the first half of the output of the hybrid PUF $|\psi_{f_1(x_i)}\rangle\langle\psi_{f_1(x_i)}|$. Figure 4.2 illustrates the construction of HLPUF, and we formalise it as Construction 4.2 in the following:

Construction 4.2 (HLPUF)**.** *Suppose we have a hybrid PUF $\mathcal{E}_f$ where $f : \{0,1\}^n \to \{0,1\}^{4m}$ is a CPUF. The mapping of the HLPUF $\mathcal{E}_f^L : \mathcal{D}^{in} \times \mathcal{H}^{d_{\text{out}_1}} \to \mathcal{H}^{d_{\text{out}_2}} \otimes \mathcal{H}^{\perp}$ corresponding to a hybrid PUF $\mathcal{E}$ is defined as follows:*

$$(x_i, \tilde{\rho}_1) \to \begin{cases} |\psi_{f_2(x_i)}\rangle\langle\psi_{f_2(x_i)}| & \text{if } \texttt{Ver}(|\psi_{f_1(x_i)}\rangle\langle\psi_{f_1(x_i)}|, \tilde{\rho}_1) = 1 \\ \perp & \text{otherwise.} \end{cases} \tag{4.3}$$

where $\texttt{Ver}(.,.)$ is a verification algorithm that checks the equality of the first half of the response based on the classical response y_i^1. To be precise, $\texttt{Ver}(.,.)$ is specified by measuring each qubit of the incoming quantum state with the corresponding basis according to $\{y_{i,2j}\}_{1\leq j\leq 2m}$ of response y_i and check the equality $\texttt{Equal}(y_{i,2j}, \tilde{y}_{i,2j})_{1\leq j\leq 2m}$ in our construction.

4.3 Adversarial Model

Evaluating PUFs under different but precise adversarial models helps us accurately capture the security. Normally, we consider two main types of adversaries: The first type of adversary does not have physical access to the PUFs. Therefore, the adversary mainly accesses PUFs remotely via the network, obtains CRPs, and performs attacks based on software, e.g., machine learning-based modelling attacks. In this case, we refer to this type of adversary as *network adversary*.

On the other hand, we refer to the other type of adversary as *physical adversary*, who is permitted to have physical access to PUFs. It grants the adversary more power to try forging the PUFs. For example, such an adversary can perform reverse engineering of hardware and various hardware side-channel attacks like micro-probing and differential power analysis [188, 189]. Meanwhile, the physical access of an adversary makes the manufacturers potentially malicious as well. The security requirements are required to be more restricted, as well as the architectural sophistication of hardware design, or possibly leverage quantum properties as we discussed previously.

In our work, we consider mainly the network adversarial model regarding most of the use cases of PUFs. That is, the adversary has only access to the communication channel. Moreover, we assume that **the manufacturer of the PUF is honest**. The network adversaries can get the challenge-response pairs just by intercepting the message that is exchanged between the server and the clients. They can pretend to be the server and make queries to the PUF on the client side with a challenge and get the response.

Any network adversary that tries to predict the response of a PUF, namely $\mathcal{E} : \mathcal{D}^{in} \to \mathcal{D}^{out}$, can be modelled as an interactive algorithm. Here, we consider QPT adversaries that have q-query classical access to the evaluation of the PUF, where q is polynomial in the security parameter. An adaptive adversary can choose and issue any arbitrary query (up to q-query), which could also depend on the previous responses received from the PUF. On the other hand, a weak non-adaptive

adversary cannot choose the queries and instead receives q CRPs of $\mathcal{E}$. In this case, the queries are randomly picked from a uniform distribution by an honest party and sent to the adversary.

4.3.1 Unforgeability with Game-based Security

Unforgeability is the main security property of PUFs. Unforgeability means that given a subset of challenge-response pairs of the target PUF, the probability of correct estimation of an unseen challenge-response pair is negligible regarding the security parameter. The unforgeability for classical PUFs has been defined in [123], and for quantum PUFs in [135] as a game-based definition. Moreover, a general game-based framework for quantum unforgeability has been defined in [190] for both quantum and classical primitives in an abstract way. Following the previous works, we present a game-based unforgeability definition for PUFs, emphasizing the adversary's capabilities in the learning phase and capturing both adaptive and weak adversaries as defined previously.

We define the unforgeability of PUFs as a formal game between two parties: a *challenger* ($\mathcal{C}$) and an *adversary* ($\mathcal{A}$). The game is divided into 4 phases: *Setup, Learning, Challenge* and *Guess*. A formal description is given in Game 4.1.

Game 4.1 (Universal Unforgeability of PUF[2])**.** *Let $\mathcal{MP}$ be the manufacturing process, `Ver(.)` be a verification algorithm for checking the responses, and λ the security parameter. We define the following game $\mathcal{G}^{PUF}(\mathcal{A}, \lambda)$ running between an adversary $\mathcal{A}$ and a challenger $\mathcal{C}$:*

- ***Setup phase.***
  - *$\mathcal{C}$ selects a manufacturing process $\mathcal{MP}$ and security parameter λ. Then $\mathcal{C}$ creates a PUF by $\mathcal{E} \leftarrow \mathcal{MP}(\lambda)$, which is described by a CPTP map. The challenge and response domain $\mathcal{D}^{in}$ and $\mathcal{D}^{out}$ are shared between $\mathcal{C}$ and $\mathcal{A}$.*
- ***Learning phase.***
  - *If the adversary is adaptive, $\mathcal{A} = \mathcal{A}_{ad}$:*
    - *$\mathcal{A}_{ad}$ selects any desired challenge $c_i \in \mathcal{D}^{in}$, and issues to $\mathcal{C}$ (up to q queries).*
    - *$\mathcal{C}$ queries the PUF with each challenge c_i and sends the response $r_i = \mathcal{E}(c_i) \in \mathcal{D}^{out}$ back to $\mathcal{A}_{ad}$.*
  - *If the adversary is weak (non-adaptive), $\mathcal{A} = \mathcal{A}_{weak}$:*
    - *$\mathcal{C}$ selects a challenge $c_i \in \mathcal{D}^{in}$ uniformly at random from $\mathcal{D}^{in}$ and independent of i (up to q queries).*
    - *$\mathcal{C}$ queries the PUF with c_i and produces the response $r_i = \mathcal{E}(c_i)$.*
    - *$\mathcal{C}$ issues to $\mathcal{A}_{weak}$ the set of random challenges and their respective responses $\{(c_i, r_i)\}_{i=1}^{q}$.*

[2]We use the term *Universal Unforgeability* as defined in [190], to avoid confusion with a stronger security model. Nevertheless, in the PUF literature, this level of security is also called *Selective Unforgeability* as also was used in [135].

- ***Challenge phase.***
  - *$\mathcal{C}$ chooses a challenge $\tilde{c}$ uniformly at random from challenge domain $\mathcal{D}^{in}$.*
  - *$\mathcal{C}$ issues $\tilde{c}$ to $\mathcal{A}$.*
- ***Guess phase.***
  - *For the challenge $\tilde{c}$, $\mathcal{A}$ produces his forgery $\sigma^r \leftarrow \mathcal{A}(1^\lambda, \tilde{c}, \{(c_i, r_i)\}_{i=1}^q)$ and sends to $\mathcal{C}$.*
  - *$\mathcal{C}$ runs a verification algorithm $b \leftarrow \texttt{Ver}(\sigma^r, \tilde{r})$, where $\tilde{r} = \mathcal{E}(\tilde{c})$ is the correct output and $b \in \{0, 1\}$, to check the fidelity or equality of the responses.*
  - *$\mathcal{C}$ outputs b. $\mathcal{A}$ wins if $b = 1$.*

The above game is the abstract version of the unforgeability game that can be used for different classical or quantum PUFs and with different challenge types. For instance, the learning phase challenge c_i can be classical bitstring or quantum states, and in that case, the domain $\mathcal{D}^{in}$ will be a finite-dimensional Hilbert space.

Note that the adversary could not arbitrarily choose the challenges in the challenge phase of the game. Therefore, it is so-called *universal unforgeability*. Relatively, there are different notions of unforgeability, e.g., *unconditional unforgeability* and *existential unforgeability* [135]. Unconditional unforgeability models the PUF against an unbounded adversary with unlimited queries during the learning phase, which is the strongest notion of unforgeability. The difference between existential unforgeability and universal unforgeability is that the adversary could choose the challenges during the challenge phase with existential unforgeability instead of choosing the challenges by the challenger $\mathcal{C}$. Even though the universal unforgeability is weaker than the other two, it is sufficient for most PUF-based applications.

Finally, we define game-based security in terms of universal unforgeability in this setting:

Definition 4.1 (Universal Unforgeability against Adaptive Adversary)**.** *A PUF with manufacturing process $\mathcal{MP}$ and verification algorithm `Ver(.)` provides (ϵ, λ)-universal unforgeability against adaptive adversary if the success probability of any adaptive QPT adversary $\mathcal{A}_{ad}$ in winning the game $\mathcal{G}^{PUF}(\mathcal{A}_{ad}, \lambda)$ is at most $\epsilon(\lambda)$.*

$$Pr[1 \leftarrow \mathcal{G}^{PUF}(\mathcal{A}_{ad}, \lambda)] \leq \epsilon(\lambda) \tag{4.4}$$

Definition 4.2 (Universal Unforgeability against Weak Adversary)**.** *A PUF with manufacturing process $\mathcal{MP}$ and verification algorithm `Ver(.)` provides (ϵ, λ)-universal unforgeability against weak (non-adaptive) adversary if the success probability of any weak QPT adversary $\mathcal{A}_{weak}$ in winning the game $\mathcal{G}^{PUF}(\mathcal{A}_{weak}, \lambda)$ is at most $\epsilon(\lambda)$.*

$$Pr[1 \leftarrow \mathcal{G}^{PUF}(\mathcal{A}_{weak}, \lambda)] \leq \epsilon(\lambda) \tag{4.5}$$

4.4 Security Analysis

In this section, we give a comprehensive security analysis of the previously proposed construction by following steps: First, we show that using hybrid construction will exponentially improve the security of classical PUFs. More precisely, it will exponentially decrease the success probability of a weak quantum adversary in the universal unforgeability game, compared to a classical PUF with the same number of learning queries. Further, we show how much quantum communication can improve the security of a weaker classical PUF and, as a result, propose an efficient and secure construction that can be built using existing classical PUFs.

For the security analysis of our construction, by leveraging the security model we give in section 3.3.2, we consider in addition the following assumptions of the CPUFs $f : \{0,1\}^n \to \{0,1\}^{4m}$.

1. For any input $x \in \{0,1\}^n$ the probability distributions of the $4m$ output bits $f(x)_1, \ldots, f(x)_{4m}$ are independent and identically distributed (i.i.d).
2. The output distributions $\{p_x^f(y)\}_{y\in\{0,1\}^{4m}}$ for all the inputs x are independent and identically distributed (i.i.d).

4.4.1 Security of HPUF against Weak Adversaries

Intuitively, to forge the HPUF, the adversary needs to extract the classical outcome of each challenge from a series of quantum states produced by the HPUF. Distinguishing an unknown non-orthogonal quantum state from a pre-determined set of states is a well-known problem in quantum information theory. Here, the security of our HPUF comes from the indistinguishability property of the non-orthogonal quantum states. In Theorem 4.1, we first show that the HPUFs based on Construction 4.1 are at least as secure as the underlying CPUFs.

Theorem 4.1. *Let $f : \{0,1\}^n \to \{0,1\}^{2m}$ be a classical PUF. If there is no QPT weak adversary that can win the universal unforgeability game for CPUF with more than a negligible probability in the security parameter, then the HPUF constructed from f is also universally unforgeable.*

Proof. We show the contrapositive statement that if you can break HPUF you can also break underlying CPUF. Here, we give the proof for $m = 1$, and it can easily be generalized for any arbitrary integer $m > 0$. Suppose for the HPUF, a q-query weak-adversary wins the unforgeability game with a non-negligible probability $P(m = 1, p, q)$. This implies, given a database of q random challenge response from the HPUF, the adversary can produce $|\psi_{f(x^*)}\rangle$ corresponding to a random challenge $x^* \in \{0,1\}^n$ with a non-negligible probability $P(m = 1, p, q)$. Note that, for the deterministic adversarial strategy, the adversary can produce multiple copies of the forged state $|\psi_{\tilde{f}(x^*)}\rangle$ for a random challenge x^*. For the random adversaries, we can produce multiple copies of the same forged state $|\psi_{\tilde{f}(x^*)}\rangle$ just by fixing the internal randomness parameter of the adversarial strategy. Hence, both the random and deterministic adversary can produce multiple copies of the forged state $|\psi_{\tilde{f}(x^*)}\rangle$ for a random challenge x^*. From the multiple (say K) such copies of $|\psi_{\tilde{f}(x^*)}\rangle$, the adversary will extract $\tilde{f}(x^*)$ using the following strategy.

Algorithm 1 Algorithm to Forge CPUF from HPUF

Require: $K \geq 2$-copies of the forged state $|\psi_{\tilde{f}(x^*)}\rangle$
  Measure the 1-st copy of the state $|\psi_{\tilde{f}(x^*)}\rangle$ in $\{|0\rangle, |1\rangle\}$-basis.
  Let $z_1 \in \{0, 1\}$ be the measurement outcome.
  for $i = 2; i \leq (K-1); i++$ **do**
    Measure the i-th copy of the state $|\psi_{\tilde{f}(x^*)}\rangle$ in $\{|0\rangle, |1\rangle\}$-basis.
    Let $z_i \in \{0, 1\}$ be the measurement outcome.
    if $z_i \neq z_{i-1}$ **then**
      break ▷ Implies $|\psi_{\tilde{f}(x^*)}\rangle \in \{|+\rangle, |-\rangle\}$.
    end if
  end for
  if $i = K$ **then**
    return $\tilde{f}(x^*) = (0, z_i)$
  else
    Measure the $i+1$-th copy in $\{|+\rangle, |-\rangle\}$-basis.
    Let z_{i+1} be the measurement outcome.
    return $\tilde{f}(x^*) = (1, z_{i+1})$.
  end if

If $|\psi_{f(x^*)}\rangle = |\psi_{\tilde{f}(x^*)}\rangle \in \{|0\rangle, |1\rangle\}$ then in Algorithm 1 all the measurement outcomes z_i (for $1 \leq i \leq K$) would be the same, and $\tilde{f}(x^*) = f(x^*)$. However, if $|\psi_{f(x^*)}\rangle = |\psi_{\tilde{f}(x^*)}\rangle \in \{|+\rangle, |-\rangle\}$ then we $\tilde{f}(x^*) \neq f(x^*)$ if and only if all the measurement outcomes z_i are equal ($1 \leq i \leq K$). This happens with probability $\frac{1}{2^K}$. Therefore, we get

$$\Pr_{x^*}[\tilde{f}(x^*) = f(x^*) || \psi_{f(x^*)}\rangle = |\psi_{\tilde{f}(x^*)}\rangle] \geq (1 - \frac{1}{2^K}). \tag{4.6}$$

If the adversary successfully forges the HPUF with a non-negligible probability $P(m = 1, p, q)$ then from Equation (4.6) we get that the adversary manages the CPUF with probability at least $P(m = 1, p, q)(1 - \frac{1}{2^K})$, which is also non-negligible. Therefore, if an adversary manages to win the unforgeability game for the HPUF with a non-negligible probability, then using the same forging strategy, it can also win the unforgeability game for the corresponding CPUF with a non-negligible probability. This implies if no QPT weak adversary can win the universal unforgeability game with a non-negligible probability for the CPUF, then no QPT adversary can win the universal unforgeability game with a non-negligible probability for the corresponding HPUF. This concludes the proof. □

The above theorem is an intuitive result showing that HPUF is stronger or at least as strong as the underlying CPUF. However, we want to prove a more powerful and explicit statement regarding HPUFs by quantifying how much the hybrid construction will boost security. In fact, we want to show that one can construct a secure, unforgeable HPUF against a quantum adversary even if the underlying CPUF is breakable (with a certain probability) against the classical forger.

To this end, we compare the success probability of a QPT adversary in breaking the HPUF in the universal unforgeability game with the success probability of the

adversary who breaks the CPUF with a certain non-negligible probability in a fixed query setting. This will allow us to show that some weak and considerably broken CPUFs can still be used to construct an asymptotically secure HPUF against stronger quantum adversaries since the quantum encoding drastically decreases the success probability.

By giving the formal theorem as described above, we start by proving three lemmas: In Lemma 4.1, we give an upper bound on the adversary's guessing probability of the response $f(x_i)$ corresponding to a challenge x_i and a single copy of the quantum response state $|\psi_{f(x_i)}\rangle$.

Lemma 4.1. *Suppose $f : \{0,1\}^n \to \{0,1\}^{4m}$ be a CPUF with the following property,*

$$\forall \;\; x_i \in \{0,1\}^n, \forall \;\; 1 \leq j \leq 4m, \;\; p^f_{x_i}(y_{i,j} = 0) = \frac{1}{2} + \delta_r, \tag{4.7}$$

with a biased distribution $p = \frac{1}{2} + \delta_r$ where $0 \leq \delta_r \leq \frac{1}{2}$, and $\mathcal{E}_f$ be a HPUF corresponding to f that we construct using Construction 4.1. Let a quantum adversary $\mathcal{A}^{i,j}_{guess}$ extract the value $y_{i,(2j-1)}$ out of $(y_{i,(2j-1)}, y_{i,2j})$ from quantum state $|\psi^{i,j}_{\text{out}}\rangle\langle\psi^{i,j}_{\text{out}}|$ corresponding to a random challenge x_i. If all the output bits of the CPUF are independent and identically distributed, then for any quantum adversary $\mathcal{A}^{i,j}_{guess}$, and $\forall\; x_i \in \{0,1\}^n$,

$$\begin{aligned} p_{guess} &:= \Pr[\mathcal{A}^{i,j}_{guess}(x_i, |\psi^{i,j}_{\text{out}}\rangle\langle\psi^{i,j}_{\text{out}}|) = y_{i,(2j-1)}] \\ &\leq p(1 + \sqrt{p^2 + (1-p)^2}) \\ &\leq p(1 + \sqrt{2}p) \end{aligned} \tag{4.8}$$

Proof. According to Construction 4.1, for a given x_i, we use the $2j$-th bit $y_{i,2j} \in \{0,1\}$ of the outcome of the CPUF to choose the basis (either $\{|0\rangle, |1\rangle\}$-basis or $\{|+\rangle, |-\rangle\}$-basis) of the j-th qubit output of the HPUF. Further, we use the $y_{i,(2j-1)} \in \{0,1\}$ to choose a state from the chosen basis. Here, if $y_{i,(2j-1)} = 0$ then from an adversarial point of view, the output state is $\rho_0 = (\frac{1}{2} + \delta_r)|0\rangle\langle 0| + (\frac{1}{2} - \delta_r)|+\rangle\langle +|$. Similarly, if $y_{i,(2j-1)} = 1$ then from an adversarial point of view, the output state is $\rho_1 = (\frac{1}{2} + \delta_r)|1\rangle\langle 1| + (\frac{1}{2} - \delta_r)|-\rangle\langle -|$. For the adversary, the probability of correctly guessing $y_{i,(2j-1)}$ is the same as distinguishing the two states ρ_0, ρ_1. Here $\Pr[\mathcal{A}^{i,j}_{guess}(x_i, |\psi^{i,j}_{\text{out}}\rangle\langle\psi^{i,j}_{\text{out}}|) = y_{i,(2j-1)}]$ denotes the optimal probability of guessing the bit correctly. From the Helstorm-Holevo bound [191, 30] we get,

$$\begin{aligned} &\Pr[\mathcal{A}^{i,j}_{guess}(x_i, |\psi^{i,j}_{\text{out}}\rangle\langle\psi^{i,j}_{\text{out}}|) = y_{i,(2j-1)}] \\ &\leq p[1 + \max_E \text{tr}[E(\rho_0 - \rho_1)]] \\ &= p[1 + \frac{1}{2}\|\rho_0 - \rho_1\|_1]. \\ &= p(1 + \sqrt{p^2 + (1-p)^2}) \\ &\leq p(1 + \sqrt{2}p) \end{aligned} \tag{4.9}$$

This concludes the proof. □

In Lemma 4.2 , we show that the adversary needs to extract the classical information $f(x)$ that is encoded in the quantum state $|\psi_{f(x)}\rangle$ for the forgery of the HPUFs. With this lemma, we give bounds on the maximum amount of information that the adversary can extract from the overall response state using quantum information tools as described in the preliminaries, Section 2.1.3.

Lemma 4.2. *Suppose* $|D_q\rangle = \bigotimes_{i=1}^{q} \left(|x_i\rangle_C \otimes |\psi_{f(x_i)}\rangle_R\right)$ *denotes the adversary's database of* q *random CRPs that are generated from a HPUF* $\mathcal{E}_f : \{0,1\}^n \to (\mathcal{H}^2)^{\otimes m}$. *If* $E(D_q)$ *denotes the measurement strategy for forging the HPUF with probability* p_{forge} *using the database* D_q, *then using the following measure-then-forge strategy that can forge the HPUF with the same probability* p_{forge}.

- *Adversary extracts the classical encoding* $\{f(x_i)\}_{1\leq i\leq q}$ *from* $|D_q\rangle$. *Let* $\{\tilde{f}(x_i)\}_{1\leq i\leq q}$ *denotes the extracted classical string.*
- *The QPT adversary applies a forging strategy using the extracted data set* $\{\tilde{f}(x_i)\}_{1\leq i\leq q}$.

Proof. For a successful forgery, the adversary needs to win the universal unforgeability defined in Game 4.1. This implies, using the measurement strategy $E(D_q)$ the adversary needs to produce a quantum state $|\psi_{f(x^*)}\rangle$ corresponding to a challenge $x^* \in_R \{0,1\}^n$ that is chosen uniformly at random. Without loss of generality, we can write the measurement strategy as a POVM with two outcomes $E(D_q) = \{E_{\text{forge}}(D_q, x^*), E_{\text{fail}}(D_q, x^*)\}$, where $E_{\text{forge}}(D_q, x^*), E_{\text{fail}}(D_q, x^*)$ denote the measurement operators corresponding to the successful forgery and the failure forgery respectively. Therefore, we can write the successful forging probability p_{forge} as follows.

$$p_{\text{forge}} = \text{tr}[E_{\text{forge}}(D_q, x^*)\rho_{D_q}^{x^*}], \tag{4.10}$$

where $\rho_{D_q}^{x^*} := |D_q\rangle\langle D_q| \otimes |x^*\rangle\langle x^*| \otimes |0^m\rangle_{out}\langle 0^m|$. Here, the *out* register would contain the forged state. If we write $E_{\text{forge}}(D_q, x^*) = M_{\text{forge}}^{\dagger}(D_q, x^*)M_{\text{forge}}(D_q, x^*)$, then we can rewrite the post-measurement state corresponding to the successful forgery as follows:

$$\begin{aligned} &\frac{M_{\text{forge}}(D_q, x^*)\,|D_q\rangle \otimes |x^*\rangle \otimes |0^{m'}\rangle_{out}}{\sqrt{p_{\text{forge}}}} \\ &= \frac{|\tilde{D}_q\rangle_R \otimes |x^*\rangle \otimes |\psi_{f(x^*)}\rangle_{out} \otimes |\tilde{a}\rangle_{out}}{\sqrt{p_{\text{forge}}}}, \end{aligned} \tag{4.11}$$

where $|\tilde{D}_q\rangle_R$ denotes the post-measurement database state, and $|\tilde{a}\rangle_{out}$ is the post-measurement state of the ancillary system which is a $(m'-m)$ dimensional state while as $|\psi_{f(x^*)}\rangle_{out}$ is m dimensional. As $\bigotimes_{i=1}^{q}|x_i\rangle_C$ is a classical state, we don't write them in the expressions in the rest of the proof.

Using the *Naimark's theorem*, we can replace the POVM measurement strategy $E(D_q)$ with the combination of a unitary acting on an extended system including an ancilla $|anc\rangle_A$, followed by a projective measurement. Let us denote the unitary as $U_{D_q}^{x^*}$ which couples the input state $|D_q\rangle \otimes |0^{m'}\rangle_{out}$ with the ancillary system

$|anc\rangle_A$, and let $\{|v\rangle\}$ be the basis on which the projective measurement is applied to the ancilla. We first rewrite the impact of the unitary $U_{D_q}^{x^*}$ on the input state:

$$\begin{aligned} & U_{D_q}^{x^*}\left(\bigotimes_{i=1}^{q}|\psi_{f(x_i)}\rangle_R\otimes|0\rangle_{out}\otimes|anc\rangle_A\right) \\ &= U_{D_q}^{x^*}\left(|\Psi_f^q\rangle_R\otimes|0\rangle_{out}\otimes|anc\rangle_A\right) \\ &= \sum_v \sqrt{p_v}\,|\Psi_v^q\rangle_R\otimes|\tilde{\psi}_v\rangle_{out}\otimes|v\rangle_A\,. \end{aligned} \tag{4.12}$$

In the second line, we have rewritten everything after applying the unitary in the $\{|v\rangle\}$-basis. Now, the adversary performs a projective measurement on the state (4.12) in this basis. Suppose for the correct forgery, the ancilla is projected into the $|v_{\text{forge}}\rangle_A$ state. Therefore we can rewrite the expression of p_{forge} as follows:

$$p_{\text{forge}} = \sum_{v:v=v_{\text{forge}}} p_v|\langle v_{\text{forge}}|\,v\rangle|^2. \tag{4.13}$$

Overall, following this strategy, the purification of the adversary's post-measurement state with an optimal POVM measurement can be written as the following:

$$\frac{|\tilde{D}_q\rangle_R\otimes|x^*\rangle\otimes|\psi_{f(x^*)}\rangle_{out}\otimes|v_{\text{forge}}\rangle_A}{\sqrt{p_{\text{forge}}}}, \tag{4.14}$$

where $|\tilde{D}_q\rangle$ denotes the post-measurement database state. Note that, due to Naimark's theorem the post-measurement database states in Equation (4.11), and (4.14) are the same if the same ancillary system has been assumed after the purification and POVM, *i.e.* if $|v_{\text{forge}}\rangle_A = |\tilde{a}\rangle_{out}$.

Now, let us use the unitary $U_{D_q}^{x^*}$ and the measurement basis $\{|v\rangle\}$ to construct a *measure-then-forge* strategy. As the unitary $U_{D_q}^{x^*}$ only depends on the input x^* and D_q, we can rewrite it in the basis that is diagonalised with respect to the states $\{|\Psi_v^q, v\rangle\}_v$.

For the post-measurement state $|v_{\text{forge}}\rangle$, of the ancilla, the adversary applies $U_{D_q,\Psi_{\text{forge}}^q,v_{\text{forge}}}^{x,x^*}$ on the $|0\rangle_{out}$ register. Note that the adversary doesn't have any information about the $\{f(x_i)\}_{1\leq i\leq q}$ before measuring the ancillary sub-system in the $\{|v\rangle\}$-basis. Hence, the measurement basis $\{|v\rangle\}$ choice only depends on the classical challenges x_i's and x^*. Therefore, the adversary can use the same information to find the $\{|v\rangle\}$-basis, the adversary first performs the measurement on the RA register in $\{|\Psi_v^q, v\rangle\}$-basis, and obtains the state $|\Psi_{\text{forge}}^q, v_{\text{forge}}\rangle$ with the same probability p_{forge}. After the measurement, the adversary applies the unitary $U_{D_q,\Psi_{\text{forge}}^q,v_{\text{forge}}}^{x^*}$ on $|0\rangle_{out}$, and get the forged state $|\psi_{f(x^*)}\rangle$. Therefore, with this strategy, the adversary also wins the unforgeability game with the probability p_{forge}.

Note that, there always exists a unitary U such that $U(\bigotimes_{i=1}^q|\tilde{f}(x_i)\rangle)\otimes|anc\rangle = |\Psi_{\text{forge}}^q, v_{\text{forge}}\rangle$, where $\tilde{f}(x_i)$ denotes the extracted information about $f(x_i)$'s from the encoded database $|D_q\rangle$. Therefore, from any generalized measurement strategy $E(D_q)$ we can construct a strategy for the measure-then-forge protocol that can win the universal unforgeability game with the same probability p_{forge}. This concludes the proof. □

Lemma 4.2 suggests that the optimal adversary (including POVM strategies) is to extract the classical information from the database state $|D_q\rangle$, and then perform the modelling attack to guess $|\psi_{f(x^*)}\rangle$. In general, if the extracted classical information $\{\tilde{f}(x_i)\}_{1\leq i\leq q}$ from the database state $|D_q\rangle$ is very far from the original encoded string $\{f(x_i)\}_{1\leq i\leq q}$ then it would be difficult for the adversary to forge the HPUF, based on that noisy data set. Here, we define the distance between $\tilde{D}_q^x = \{\tilde{f}(x_i)\}_{1\leq i\leq q}$, and $D_q^x = \{f(x_i)\}_{1\leq i\leq q}$ as follows.

$$\text{Dist}(\tilde{D}_q^x, D_q^x) := \frac{\sum_{i=1}^{q} \text{Mis-match}(\tilde{f}(x_i), f(x_i))}{q}, \tag{4.15}$$

where we define Mis-match$(\tilde{f}(x_i), f(x_i))$ as follows.

$$\text{Mis-match}(\tilde{f}(x_i), f(x_i)) := \begin{cases} 1 & \text{If } (\tilde{f}(x_i) \neq f(x_i)) \\ 0 & \text{Otherwise.} \end{cases} \tag{4.16}$$

It is reasonable to assume that no forging strategy can forge the HPUF with a non-negligible probability that runs on the noisy database set $\tilde{D}_q^x$ such that $\text{Dist}(\tilde{D}_q^x, D_q^x) > \varepsilon$, where $0 \leq \varepsilon \leq 1$ is a parameter that quantifies the error threshold. In Lemma 4.3, we give an upper bound on extracting $\tilde{D}_q^x$ from $|D_q\rangle$ such that $\text{Dist}(\tilde{D}_q^x, D_q^x) \leq \varepsilon$. Intuitively, a robust HPUF is with low ε such that an adversary can not forge it with a noisy data set which is very far away from the original data set. Otherwise, the ε should be high with a bad HPUF.

Lemma 4.3. *Suppose* $|D_q\rangle = \bigotimes_{i=1}^{q} \left(|x_i\rangle_C \otimes |\psi_{f(x_i)}\rangle_R\right)$ *denotes the adversary's database of* q *random CRPs that are generated from a HPUF* $\mathcal{E}_f : \{0,1\}^n \to (\mathcal{H}^2)^{\otimes m}$. *If* $\tilde{D}_q$ *denotes the noisy classical response set that is extracted from* $|D_q\rangle$ *such that Dist*$(D_q, \tilde{D}_q) \leq \varepsilon$ *with probability* $p_{extract}$, *then*

$$p_{extract} \leq \sum_{k=(1-\varepsilon)q}^{q} \binom{q}{k} (p_{guess})^{2mk} (1 - (p_{guess})^{2m})^{q-k}, \tag{4.17}$$

where $p_{guess} \leq p(1 + \sqrt{2}p)$, *defined in Lemma 4.1.*

Proof. In this lemma, we give an upper bound on the probability of extracting the CPUF outcomes from the $(1-\varepsilon)q$ out of q responses of the HPUF. Let $\mathcal{A}_h$ be a quantum adversary who plays the unforgeability game against the HPUF. $\mathcal{A}_h$ has access to q queries of the HPUF as q pairs of $\{(X_i, |\psi_{f(X_i)}\rangle)\}_{i=1}^{q}$. Note that, according to the construction 4.1, $|\psi_{f(X_i)}\rangle\langle\psi_{f(X_i)}| = \bigotimes_{j=1}^{2m} |\psi_{f(X_i)}^{i,j}\rangle\langle\psi_{f(X_i)}^{i,j}|$, where $|\psi_{f(X_i)}^{i,j}\rangle \in \{|0\rangle, |1\rangle, |+\rangle, |-\rangle\}$. As the state in the adversary's possession depends fully on a classical string, we can describe this situation using a classical-quantum state, where the C register contains the classical string $f(X_i)$, and the S register contains the quantum state $|\psi_{f(X_i)}\rangle\langle\psi_{f(X_i)}|$. We assume the j-th bit of the string $f(X_i)$ as $Y_{i,j}$. The classical-quantum state for the j-th qubit is of the following form.

$$(\rho_{CS})_j = \sum_{\substack{Y_{2j-1,2j}, \\ Y_{2j}\in\{0,1\}}} \frac{1}{4} |Y_{2j-1,2j}\rangle_{C_{i,j}} \langle Y_{2j-1,2j}| \otimes |\psi_{f(X_i)}^{i,j}\rangle\langle\psi_{f(X_i)}^{i,j}|. \tag{4.18}$$

In Lemma 4.1, we prove that the probability of guessing Y_j is p_{guess}, and it has the following upper bound.

$$p_{\text{guess}} \leq p(1 + \sqrt{2}p). \tag{4.19}$$

Since we assume that all the output bits of the CPUF are i.i.d. Therefore, the entire classical-quantum state for the i-th challenge X_i is ρ_{CS} of the following form.

$$\rho_{CS} = \bigotimes_{j=1}^{m} (\rho_{CS})_j. \tag{4.20}$$

Therefore, the probability of guessing $f(X_i)$ from the S subsystem is upper bounded by

$$(p_{\text{guess}})^{2m}. \tag{4.21}$$

Let $\rho_{C^q S^q}$ denote the joint state shared between the server and the q-query weak adversary. Due to the i.i.d assumption on all the outputs of the underlying classical PUF of the HPUF, $\rho_{C^q S^q}$ has the following form.

$$\rho_{C^q S^q} = \left(\bigotimes_{j=1}^{m} (\rho_{CS})_j \right)^{\otimes q}. \tag{4.22}$$

Here, we would like to find an upper bound on the probability of successfully guessing $f(X_i)$'s for at least $(1-\varepsilon)q$ responses out of q responses. We denote this guessing probability as p_{extract}. Note that, due to the i.i.d assumption on the different outcomes of the CPUF, the adversary's success probability of guessing exactly k responses out of q responses is upper bounded by $\binom{q}{k}(p_{\text{guess}})^{2mk}(1 - (p_{\text{guess}})^{2m})^{q-k}$. Therefore, we can re-write the expression of p_{extract} as follows,

$$p_{\text{extract}} \leq \sum_{k=(1-\varepsilon)q}^{q} \binom{q}{k} (p_{\text{guess}})^{2mk} (1 - (p_{\text{guess}})^{2m})^{q-k}. \tag{4.23}$$

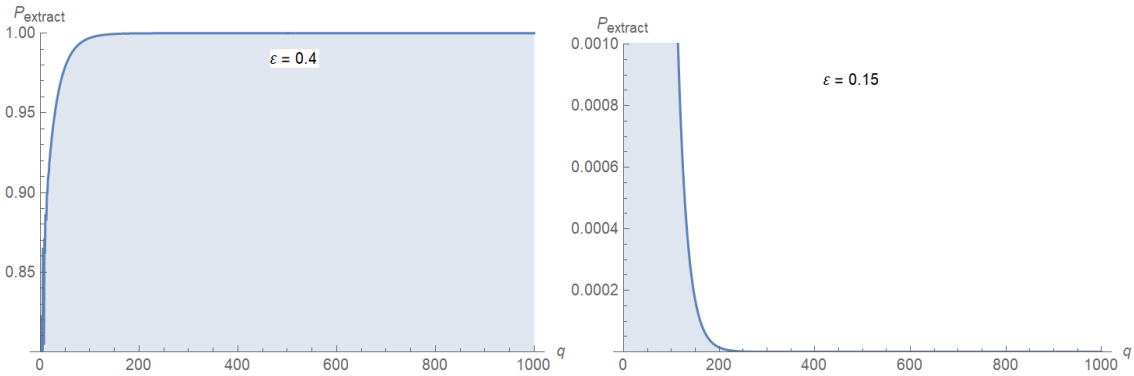
This concludes the proof. □

To provide a better intuition of the expression of p_{extract} to show the exponential gap, we give in Figure 4.3 the evolution of p_{extract} for different values of ε. It means that with a bad HPUF with high ε, the p_{extract} converges to $1 - negl(\lambda)$ as the number of queries of the QPT weak adversary increases. Otherwise, for a smaller error threshold, corresponding to a better HPUF, it decreases exponentially with q. Later, we show in Section 4.5 the ε of HPUF depends on its underlying CPUFs, and the machine-learning algorithm we use to forge the HPUF.

Now, we formally give the theorem that the upper bounds of the success probability of forging a HPUF by a QPT weak adversary.

Theorem 4.2. *Let $f : \{0,1\}^n \to \{0,1\}^{4m}$ be a classical PUF with p-Randomness, where $p = (\frac{1}{2} + \delta_r)$ with the following two properties.*

1. *Let any q-query weak adversary win the universal unforgeability game for the CPUF f with probability at most $p_{forge}^{classical}(m, p, q) \geq non - negl(\lambda)$.*



(a) (b)

Figure 4.3: Evolution of p_{extract} with different values of ε

2. *There is no QPT adversary that can win the universal unforgeability game for the CPUF using a noisy database $\tilde{D}_q$ such that $Dist(D_q, \tilde{D}_q) > \varepsilon$.*

If we construct a HPUF $\mathcal{E}_f$ from such a CPUF f, then the q-query weak quantum adversary can win the universal unforgeability game for the HPUF $\mathcal{E}_f$ with probability $p_{forge}^{quantum}(x^*, p, |Q_q\rangle)$, such that,

$$p_{forge}^{quantum}(x^*, p, |Q_q\rangle) \leq p_{extract} \times p_{forge}^{classical}(m, p, q), \tag{4.24}$$

where

$$p_{extract} \leq \sum_{k=(1-\varepsilon)q}^{q} \binom{q}{k} (p_{guess})^{2mk} (1 - (p_{guess})^{2m})^{q-k}.$$

Proof. From Lemma 4.2, we get that the optimal adversary's strategy is measure-then-forge. Let $\tilde{D}_q$ denotes the set of extracted database response. From the 2nd property, we get that the adversary can forge the HPUF with a non-negligible probability if and only if $\text{Dist}(\tilde{D}_q, D_q) \leq \varepsilon$. Suppose p_{extract} denotes the optimal success probability of extracting $\tilde{D}_q$ from $|D_q\rangle$ such that $\text{Dist}(D_q, \tilde{D}_q) \leq \varepsilon$. If $p_{\text{forge}}^{\text{classical}}(\tilde{D}_q, X^*, p)$ denotes the optimal forging probability using the database $\tilde{D}_q$, then the total forging probability is given by the following equation.

$$p_{\text{forge}}^{\text{quantum}}(X^*, p, |D_q\rangle) = p_{\text{extract}} \times p_{\text{forge}}^{\text{classical}}(\tilde{D}_q, X^*, p). \tag{4.25}$$

Note that the adversary's optimal forging probability with database D_q is always higher than the optimal forging probability with the database $\tilde{D}_q$, i.e.,

$$p_{\text{forge}}^{\text{classical}}(m, p, q) \geq p_{\text{forge}}^{\text{classical}}(\tilde{D}_q, X^*, p). \tag{4.26}$$

Substituting the relation in Equation (4.26) in Equation (4.25) we get the following expression of $p_{\text{forge}}^{\text{quantum}}(X^*, p, |D_q\rangle)$.

$$p_{\text{forge}}^{\text{quantum}}(X^*, p, |D_q\rangle) \leq p_{\text{extract}} \times p_{\text{forge}}^{\text{classical}}(m, p, q). \tag{4.27}$$

From Lemma 4.3 we get that $p_{\text{extract}} \leq \sum_{k=(1-\varepsilon)q}^{q} \binom{q}{k} (p_{\text{guess}})^{2mk} (1 - (p_{\text{guess}})^{2m})^{q-k}$. By substituting the expression of p_{success} in Equation (4.27), we get the desired upper bound on the $p_{\text{forge}}^{\text{quantum}}(X^*, p, |D_q\rangle)$. This concludes the proof. □

The above result is a general statement for any fixed number of queries and compares the success probability of a weak adversary in breaking the unforgeability of CPUF and HPUF. Given this theorem, we can also easily state the following corollary that ensures the universal unforgeability of an HPUF constructed from a CPUF that does not provide suitable security, yet is not totally broken with overwhelming probability.

Corollary 4.1. *Let the success probability of any QPT weak-adversary in the universal unforgeability game with a CPUF $f : \{0,1\}^n \to \{0,1\}^{4m}$ with p-Randomness, be at most $p^{classic}_{forge}$, where $0 \leq p^{classic}_{forge} \leq 1 - non\text{-}negl(2m)$. Then, there always exists an error threshold $0 < \varepsilon \leq 1$ for which the success probability of any QPT adversary in the universal unforgeability game for the HPUF $\mathcal{E}_f$, is at most $\epsilon(2m)$, which is a negligible function in the security parameter. Hence, such HPUFs are universally unforgeable.*

Proof. This directly follows from Theorem 4.2 where $p^{\text{classic}}_{\text{forge}} = p^{\text{classical}}_{\text{forge}}(m, p, q)$ for any $q = poly(m)$ is a value between 0 and 1, and not negligibly close to 1. As shown in the proof of Theorem 4.2 in the Appendix, for a large family of ε the first part of the probability, namely p_{extract} becomes negligibly small (in $2m$). Hence, the overall probability becomes a negligible function $\epsilon(2m)$. □

Note that here, we only consider the adversarial model assuming that the adversary gets access to a random set of these classical challenges and quantum responses, where there exists only one copy of each pair in the adversary's database. This model is usually referred to as a weak adversary. We then upgrade this adversary into a more powerful one, which is our target most powerful quantum adversary of interest, when introducing the locking mechanism of the construction in the next section.

4.4.2 Security of HLPUF against General Adaptive Adversaries

In this section, we need to uplift the previously considered weak adversary into any general *adaptive* quantum adversary. Recall that an adaptive adversary is free to build their database with any arbitrary query and in an adaptive manner, potentially depending on the previous queries. Particularly such adversaries can query HPUF multiple times with the same challenge x, obtaining several copies of $|\psi_{\text{out}}\rangle$ and can easily extract the outcome $f(x)$ from multiple copies. Consequently, a probability $p_{\text{guess}} \approx 1$ can be achieved in theory, and a strong adversary can forge the HPUF efficiently. Hence the construction of HPUFs on its own is not sufficient to achieve the most compelling desired notion of quantum security.

In Theorem 4.3, we show that if the HPUFs are secure against weak adversaries, then we can make the HLPUFs secure against adaptive adversaries with the lockdown technique.

Theorem 4.3. *Let $\mathcal{E}_f : \{0,1\}^n \to (\mathcal{H}^2)^{\otimes m} \otimes (\mathcal{H}^2)^{\otimes m}$ be a hybrid PUF that we construct from a classical PUF $f : \{0,1\}^n \to \{0,1\}^{2m} \times \{0,1\}^{2m}$ and let $\mathcal{E}^L_f : \{0,1\}^n \times (\mathcal{H}^2)^{\otimes m} \to (\mathcal{H}^2)^{\otimes m}$ denotes the HLPUF that we construct from $\mathcal{E}_f$ using the Construction 4.2. If $\mathcal{E}_f = \mathcal{E}_{f_1} \otimes \mathcal{E}_{f_2}$ and if each of the mappings $\mathcal{E}_{f_1}, \mathcal{E}_{f_2}$*

has (ϵ, m)-universal unforgeability against the q-query weak adversaries, then the corresponding HLPUF $\mathcal{E}_f^L$ is (ϵ, m)-secure against the q-query adaptive adversaries.

Proof. At the i-th round, the HLPUF $\mathcal{E}_f^L$ receives the queries of the form $(x_i, \tilde{\rho}_1)$, where the classical string $x_i \in \{0,1\}^n$, and $\tilde{\rho}_1 \in (\mathcal{H}^2)^{\otimes m}$. The HLPUF returns $\mathcal{E}_{f_2}(x_i)$ if $\texttt{Ver}(\tilde{\rho}_1, \mathcal{E}_{f_1}(x_i)) = 1$, otherwise it returns an abort state $|\perp\rangle\langle\perp|$ corresponding to $\perp$. Hence, to get any non-abort state $|\perp\rangle$ from the HLPUF, the adaptive adversaries $\mathcal{A}_{ad}$ need to produce a query of the form $(x_i, \mathcal{E}_{f_1}(x_i))$. As the adversary doesn't have any direct access to the mapping $\mathcal{E}_{f_1}$, the only way it can get any information about $\mathcal{E}_{f_1}(x_i)$ is by intercepting the challenges that are sent by the server to the client. Suppose that the adaptive adversary has access to a set of q queries $X_{[q]} := \{X_i\}_{1\leq i\leq q}$ and the corresponding responses $\Psi_{[q]} := \{\mathcal{E}_{f_1}(x_i)\}_{1\leq i\leq q}$. Here, each X_i follows a uniform distribution over the challenge set $\{0,1\}^n$. Hence, for the mapping $\mathcal{E}_{f_1}$, the power of the adaptive adversary reduces to the power of a weak adversary. As $\mathcal{E}_{f_1}$ has the universal unforgeability property against any q-query weak adversary, hence we get, for any random challenge $X \notin X_{[q]}$,

$$\begin{aligned}&\Pr_{X,X_{[q]}}[1 \leftarrow \mathcal{G}^{\mathcal{E}_{f_1}}(\mathcal{A}_{ad}, m, X, X_{[q]})]\\ &= \Pr_{X,X_{[q]}}[1 \leftarrow \mathcal{G}^{\mathcal{E}_{f_1}}(\mathcal{A}_{weak}, m, X, X_{[q]})] \leq \epsilon(m).\end{aligned} \tag{4.28}$$

This implies, using the set of challenges $X_{[q]}$ and responses $\Psi_{[q]}$, the adversary cannot produce the response corresponding to a random challenge $X \notin X_{[q]}$. Suppose from the query set $X_{[q]}$ and the responses, the adaptive adversary successfully generates a set $X'_{[q']}$ of q' adaptive queries, and corresponding responses $\Psi_{[q']}$ for the HLPUF $\mathcal{E}_f^L$. Without any loss of generality, we assume that for all of the queries, $X'_i \in X'_{[q']}$, the HLPUF returns a non-abort state.

We assume that the adaptive adversary wins the universal unforgeability game using the query set $X_{\text{ad}} = X_{[q]} \cap X'_{[q']}$. This implies,

$$\Pr_{X,X^{\mathcal{E}_L}_{[q]_{\text{ad}}}}[1 \leftarrow \mathcal{G}^{\mathcal{E}_f^L}(\mathcal{A}_{ad}, m, X, X_{\text{ad}})] \geq \text{non-negl}(m). \tag{4.29}$$

From the construction of our HLPUF in Construction 4.2, we get that winning the universal unforgeability game with the HLPUF $\mathcal{E}_f^L$ implies winning the universal unforgeability with $\mathcal{E}_{f_2}$. Hence, we can rewrite Equation (4.29) in the following way,

$$\Pr_{X,X_{\text{ad}}}[1 \leftarrow \mathcal{G}^{\mathcal{E}_{f_2}}(\mathcal{A}_{ad}, m, X, X_{\text{ad}})] \geq \text{non-negl}(m). \tag{4.30}$$

Note that if the adaptive adversary manages to get non-abort outcomes from the HLPUF corresponding to all $X'_i \in X_{\text{ad}}$ then from the Construction 4.2 we get, $1 \leftarrow \mathcal{G}^{\mathcal{E}_{f_1}}(\mathcal{A}_{ad}, m, X'_i, X_{\text{ad}})$. Due to the unforgeability assumption of Equation (4.28) we get,

$$\begin{aligned}&\Pr_{X,X_{[q]}}[1 \leftarrow \mathcal{G}^{\mathcal{E}_{f_1}}(\mathcal{A}_{weak}, m, X, X_{[q]})]\\ &= \Pr_{X,X_{\text{ad}}}[1 \leftarrow \mathcal{G}^{\mathcal{E}_{f_1}}(\mathcal{A}_{ad}, m, X, X_{\text{ad}})] \leq \epsilon(m).\end{aligned} \tag{4.31}$$

Note that the main difference between adaptive and weak adversaries lies in the choice of the query set. If we fix the query set X_{ad}, then the both adaptive $\mathcal{A}_{ad}$ and the weak adversary can extract the same amount of information from the responses corresponding to the query set X_{ad}. Therefore, their winning probability of the universal unforgeability game becomes equivalent. This implies, we can rewrite Equation (4.31) in the following way,

$$\begin{aligned}&\Pr_{X,X_{\text{ad}}}[1 \leftarrow \mathcal{G}^{\mathcal{E}_{f_1}}(\mathcal{A}_{ad}, m, X, X_{\text{ad}})] \\ &= \Pr_{X,X_{\text{ad}}}[1 \leftarrow \mathcal{G}^{\mathcal{E}_{f_1}}(\mathcal{A}_{weak}, m, X, X_{\text{ad}})] \leq \epsilon(m). \end{aligned} \tag{4.32}$$

By combining Equation (4.31) and Equation (4.32) we get, both the random variables $X_{[q]}$ and X_{ad} are equivalent. From the universal unforgeability property of the PUF $\mathcal{E}_{f_2}$ against any q-query weak adversary, we get

$$\Pr_{X,X_{[q]}}[1 \leftarrow \mathcal{G}^{\mathcal{E}_{f_2}}(\mathcal{A}_{weak}, m, X, X_{[q]})] \leq \epsilon(m). \tag{4.33}$$

As both of the random variables $X_{[q]}$ and X_{ad} are equivalent, so we get,

$$\begin{aligned}&\Pr_{X,X_{[q]}}[1 \leftarrow \mathcal{G}^{\mathcal{E}_{f_2}}(\mathcal{A}_{weak}, m, X, X_{[q]})] \\ &= \Pr_{X,X_{\text{ad}}}[1 \leftarrow \mathcal{G}^{\mathcal{E}_{f_2}}(\mathcal{A}_{weak}, m, X, X_{\text{ad}})] \\ &= \Pr_{X,X_{\text{ad}}}[1 \leftarrow \mathcal{G}^{\mathcal{E}_{f_2}}(\mathcal{A}_{ad}, m, X, X_{\text{ad}})] \leq \epsilon(m). \end{aligned} \tag{4.34}$$

The second equality follows from the fact that for a fixed query set X_{ad}, the adaptive adversary $\mathcal{A}_{ad}$ and weak adversary $\mathcal{A}_{weak}$ become equivalent. Note that only one of Equation (4.30) and Equation (4.34) is true. The Equation (4.34) is true because of the unforgeability of $\mathcal{E}_{f_2}$. Hence, our assumption of Equation (4.30) is wrong. Therefore, Equation (4.29) is also not true. Hence, with the proof by contradiction, we get

$$\Pr_{X,X_{\text{ad}}}[1 \leftarrow \mathcal{G}^{\mathcal{E}_f^L}(\mathcal{A}_{ad}, m, X, X_{\text{ad}})] \leq \epsilon(m). \tag{4.35}$$

This concludes the proof. □

As a result, if an adversary tries to query HLPUF with any arbitrary challenge x, they need to produce a correct quantum state $|\psi_{f_1(x)}\rangle$. Otherwise, the verification procedure inside the HLPUF fails, and the HLPUF replies with a garbage output $\perp$. The inability of the adversary to produce the outcome $|\psi_{f_1(x)}\rangle$ has itself insured via the unforgeability of the HPUF construction and the *no-cloning* principle of the quantum states.

The only remaining option for the adaptive adversary would be to intercept the challenges sent by the server in the previous rounds and use them to query the HLPUF. Therefore, practically, with the same challenge x, they can query the HLPUF only once. Given that the server chooses the challenges uniformly at

random from its database, the adversary querying the HLPUF with those challenges will reduce their power to a weak adversary. As we showed the security of $\mathcal{E}_{f_1}$, and $\mathcal{E}_{f_2}$ against the q-query weak adversaries, with the proposed construction, the HLPUF remains secure against any q-query adaptive adversaries.

4.4.3 Limitation of Lockdown Mechanism in Quantum Analogue

In the last section, we show the power of a quantum lock mechanism by enabling the unforgeability of HPUF from against a weak adversary to any general adaptive quantum adversary. Therefore, it is intuitive to think about whether such a mechanism is extendable to arbitrary encoding strategies on CPUFs, where a quantum process can be used to describe the overall evolution of HPUF and characterised by the modelling of quantum PUFs (QPUFs) in Section 3.3.4. For simplicity, we study here the possibility of exploiting the quantum lock mechanism on generic QPUFs to discuss the possibility of arbitrary encodings of HLPUF.

Meanwhile, showing the possibility of exploiting the quantum lock mechanism on generic QPUFs is also very meaningful. Recall that one of the main problems in the case of QPUFs is that if an adaptive adversary manages to query a QPUF with the same input multiple times, then such an adversary can get multiple copies of the same output quantum state. This allows the adversary to use the tools from the quantum state tomography [192], and the quantum emulation algorithm to emulate the input-output behaviour [124] of the target QPUF. One possible way to protect it from such sophisticated attacks is to use the lockdown technique. The main goal of such a lockdown technique is to prevent the adversary from querying in an adaptive manner with arbitrary challenges.

Similarly to the hybrid PUF setting, an important feature of the lockdown technique on QPUFs is the equality test of unknown quantum states for verification. As introduced previously, the verification algorithm can be efficiently implemented by SWAP test [31] if two states ρ_1, ρ_2 are two pure states. With this constraint in mind, we prove that only very restricted QPUFs can be efficiently constructed as a quantum-locked PUF (QLPUF) with a verification algorithm.

Theorem 4.4. *The construction of QLPUF with verification algorithm can be achieved if and only if the input/output mapping of the targeted quantum PUF* $\mathcal{E} : \mathcal{H}^{d_{\text{in}}} :\to \mathcal{H}^{d_{\text{out}_1}} \otimes \mathcal{H}^{d_{\text{out}_2}}$ *is of the form* $|\psi_{\text{in}}\rangle\langle\psi_{\text{in}}| \mapsto |\psi_{\text{out}}\rangle_{S^1}\langle\psi_{\text{out}}| \otimes |\psi_{\text{out}}\rangle_{S^2}\langle\psi_{\text{out}}|$. *Otherwise, such a lockdown technique is incompatible with the targeted quantum PUFs.*

Proof. The proof is twofold. For a quantum PUF $\mathcal{E} : \mathcal{H}^{d_{\text{in}}} :\to \mathcal{H}^{d_{\text{out}_1}} \otimes \mathcal{H}^{d_{\text{out}_2}}$ that maps an input state $|\psi^i_{\text{in}}\rangle_{S_i}\langle\psi^i_{\text{in}}| \in \mathcal{H}^{d_{\text{in}}}$ to an output state $|\psi^i_{\text{out}}\rangle_{S^1_i S^2_i}\langle\psi^i_{\text{out}}| \in \mathcal{H}^{d_{\text{out}_1}} \otimes \mathcal{H}^{d_{\text{out}_2}}$ with subsystem S^1 and S^2. The mapping of the QLPUF $\mathcal{E}_L : \mathcal{H}^{d_{\text{in}}} \otimes \mathcal{H}^{d_{\text{out}_1}} \to \mathcal{H}^{d_{\text{out}_2}} \otimes \mathcal{H}^{\perp}$ corresponding to a quantum PUF $\mathcal{E}$ is defined as follows:

$$|\psi^i_{\text{in}}\rangle_{S_i}\langle\psi^i_{\text{in}}| \otimes \tilde{\rho}_{S^1_i} \to \begin{cases} \rho_{S^2_i} & \text{if } \texttt{Ver}(\rho_{S^1_i}, \tilde{\rho}_{S^1_i}) = 1 \\ \perp & \text{otherwise.} \end{cases} \tag{4.36}$$

where $\rho_{S^1_i} = \text{tr}_{S^2_i}\left[|\psi^i_{\text{out}}\rangle_{S^1_i S^2_i}\langle\psi^i_{\text{out}}|\right]$ and $\rho_{S^2_i} = \text{tr}_{S^1_i}[|\psi^i_{\text{out}}\rangle_{S^1_i S^2_i}\langle\psi^i_{\text{out}}|]$.

According to such construction, the QLPUF takes the input $|\psi^i_{\text{in}}\rangle_{S_i}\langle\psi^i_{\text{in}}| \otimes \tilde{\rho}_{S_i^1}$. Among the two input states, the QLPUF uses $|\psi^i_{\text{in}}\rangle_{S_i}\langle\psi^i_{\text{in}}|$ to get an output state $|\psi^i_{\text{out}}\rangle_{S_i^1 S_i^2}\langle\psi^i_{\text{out}}|$. The QLPUF outputs a state $\rho_{S_i^2}$ if $\rho_{S_i^1}$ is same as the state $\tilde{\rho}_{S_i^1}$. Otherwise, it outputs an abort state $\perp$. We refer to Figure 4.4 for the circuit of the QLPUF. Note that the QLPUF needs to internally check whether $\rho_{S_i^1} = \tilde{\rho}_{S_i^1}$. If $\rho_{S_i^1}$ is a pure state, then we can use the SWAP test to check the equality of two pure states. The circuit of the SWAP test makes the circuit of the entire QLPUF efficient.

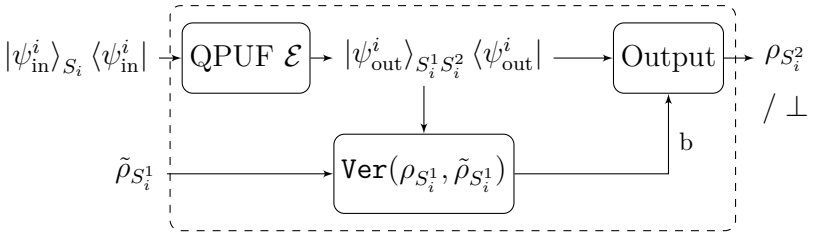


Figure 4.4: Construction of QLPUF $\mathcal{E}_L$ with quantum PUF $\mathcal{E} : \mathcal{H}^{d_{\text{in}}} :\to \mathcal{H}^{d_{\text{out}_1}} \otimes \mathcal{H}^{d_{\text{out}_2}}$

On the other hand, however, in the case when the quantum channel $\mathcal{E}$ of the quantum PUF can have entangling power and hence the subsystems S^1 and S^2 that represent the different parts of the response, may be entangled. Let's start from the simple situation with 2-qubit entangled state as $|\psi^i_{\text{out}}\rangle\langle\psi^i_{\text{out}}|$. i.e., for a quantum PUF $\mathcal{E}$ that maps an input state $|\psi^i_{\text{in}}\rangle\langle\psi^i_{\text{in}}|$ to an entangled output state $|\psi^i_{\text{out}}\rangle\langle\psi^i_{\text{out}}| := (\alpha\,|a_1^i\rangle\,|b_1^i\rangle + \beta\,|a_2^i\rangle\,|b_2^i\rangle)(\alpha^*\,\langle a_1^i|\,\langle b_1^i| + \beta^*\,\langle a_2^i|\,\langle b_2^i|)$ where $|\alpha|^2 + |\beta|^2 = 1$, $|a_1\rangle$ and $|a_2\rangle$ are any two vectors in the space of subsystem S^1, and $|b_1\rangle$ and $|b_2\rangle$ are any two vectors in the space of subsystem S^2. Consider a POVM measurement on the subsystem S^1 with m elements $\{E_m\}$ where $\Sigma_m E_m = I$, the reduced density operator of S^2 after tracing out S^1 is:

$$\begin{aligned}\rho_{S_i^2} &= \Sigma_m \text{tr}_{S_i^1}[\text{tr}(|\psi^i_{\text{out}}\rangle_{S_i^1 S_i^2}\langle\psi^i_{\text{out}}|E_m)] \\ &= \Sigma_m \text{tr}_{S_i^1}[\langle\psi^i_{\text{out}}|E_m|\psi^i_{\text{out}}\rangle_{S_i^1 S_i^2}] \\ &= |\alpha|^2\,|b_1^i\rangle\,\langle b_1^i| + |\beta|^2\,|b_2^i\rangle\,\langle b_2^i| \end{aligned} \tag{4.37}$$

The state of subsystem S^2 is clearly a mixed state. However, checking the equality between two mixed states is difficult and sometimes not possible. For example, we have two different mixed states:

$$|\psi_1^i\rangle = \begin{cases} |b_1^i\rangle & \text{with probability } |\alpha|^2 \\ |b_2^i\rangle & \text{with probability } |\beta|^2 \end{cases} \tag{4.38}$$

and

$$|\psi_2^i\rangle = \begin{cases} \alpha\,|b_1^i\rangle + \beta\,|b_2^i\rangle & \text{with probability } \frac{1}{2} \\ \alpha\,|b_1^i\rangle - \beta\,|b_2^i\rangle & \text{with probability } \frac{1}{2} \end{cases} \tag{4.39}$$

The density operators of both mixed states are represented as Equation (4.37). That is to say, these two mixed states are unequal but totally indistinguishable. This can be trivially extended to the n-qubit situation. So the lockdown technique is not implementable with generic quantum PUFs. □

In the case of quantum PUFs, our study shows that some quantum mechanical properties, such as entanglement generation, make it challenging to use the straightforward quantum analogue of the classical lockdown technique. That is to say, such entanglement generation should be avoided while designing the encoding of CPUFs' output within the HLPUF constructions. Fortunately, to the best of our knowledge, it is neither sufficient nor necessary to encode the output of classical PUF to construct an HPUF with the lockdown technique.

4.5 Numeral Simulations of H(L)PUF against ML Attacks

In this section, we validate and showcase the practicality of our theoretical results of hybrid (locked) PUF constructions using numerical results and simulations. While introducing our security in the previous sections, we give a theoretical upper bound on the forging probability. Our theoretical security analysis shows that exponential security can be achieved for HPUF construction, relying on certain reasonable assumptions, including the existence of a classical PUF that is not broken with probability 1, nonetheless is breakable with non-negligible probability given enough queries.

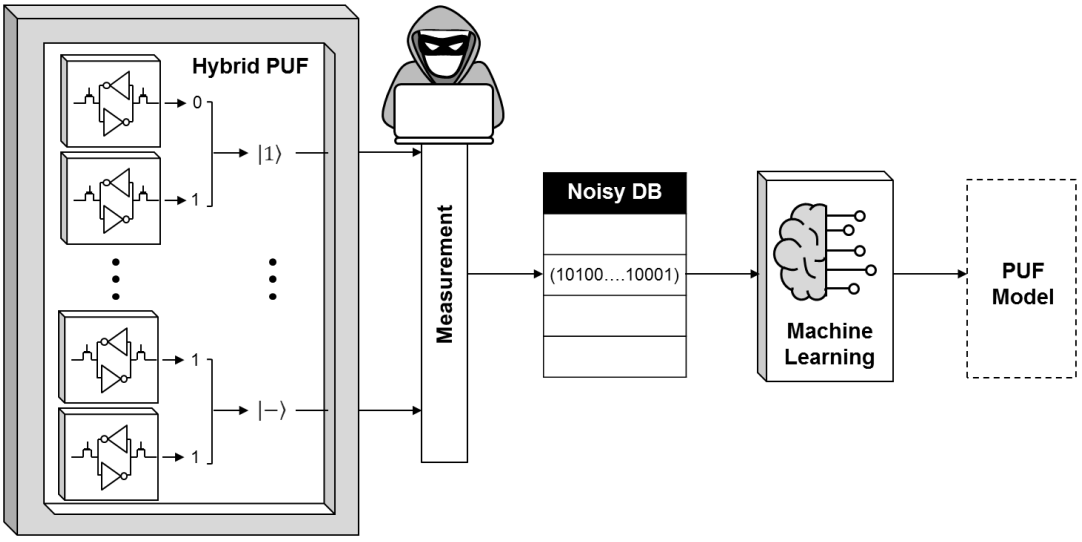


Figure 4.5: Illustration of the measure-then-forge attack. The quantum adversary receives a sequence of (BB84) quantum states as the output of the HPUF and measures them with the optimal measurement strategy to obtain the underlying classical information of the responses of CPUF. Due to the quantum nature of the HPUF responses, even the best measurement strategy is still probabilistic, which leaves the adversary with a noisy version of the classical database. Then the adversary can run a machine-learning attack on the noisy database (in the optimal attack, this classical machine-learning algorithm is assumed to be optimal as well) to extract the mathematical model of the PUF.

Although such mid-level classical PUFs can be theoretically found, especially in optical-based constructions, we focus on putting our construction on top of the

cheapest and most widely available CPUFs. We choose silicon CPUFs such as arbiter PUFs for this purpose, which are known to be weak in security and breakable using machine-learning attacks. We compare the performance of these CPUFs with an HPUF that is constructed with the same underlying CPUF, performing measure-then-forge attacks using classical machine-learning algorithms (see Figure 4.5 for the illustration of the attack). The numerical simulation results assist in demonstrating our theoretical proofs by exhibiting an exponential advantage of success probability of HPUF forgery compared to its underlying CPUF with a limited q-query.

We instantiate the underlying CPUFs by *pypuf* [186]. *pypuf* is a python-based emulator that features different existing CPUFs. From its library, we utilise the XOR Arbiter PUFs, [106] which are CMOS-based silicon PUFs of the form $f : \{0,1\}^n \to \{0,1\}$, as we introduced in Section 3.3. For constructing the HPUFs, we need an underlying CPUF with at least two bits outcome. Therefore, we use multiple such XOR arbiter PUFs (say f_1, and f_2, etc...) in parallel composition for instantiating the underlying CPUFs [106] of HPUF/HLPUF constructions.

For the forgery, we use the measure-then-forge strategy defined in Section 4.4.1. As the best measurement strategy for the measure-then-forge attack, we use the upper bound we derive on the adversary's guessing probability of extracting a single-bit outcome of the CPUF from the outcome of the HPUF (see Lemma 4.1). After the measurement phase in the measure-then-forge strategy, the adversary ends up with a classical database.

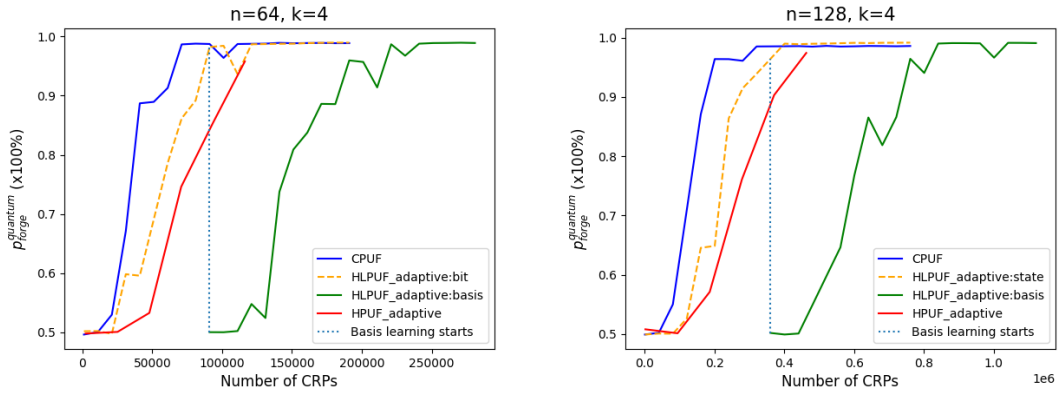
The security of k-XOR PUFs is studied widely by Ulrich Rührmair et al. [107]. In that paper, the performance of different machine learning attacks like *Logistic Regression* (LR), *Support Vector Machines* (SVMs), and *Evolution Strategies* (ES) is evaluated in terms of the prediction accuracy of responses with unseen challenges. It turns out that the LR has the best performance. Moreover, it shows that the LR attacks can handle the situation well while the training data is erroneous with noise up to 40%. In practice, this noise comes from the PUF implementation with the integrated circuit. Meanwhile, quantum encoding of HPUF can be treated as another source of noise to prevent the adversary from modelling CPUFs.

4.5.1 BB84 encoding with Split Attack on HPUF/HLPUF

We start with the simulation that HPUF encodes BB84 states, where every two-bit tuple of response $(y_{i,(2j-1)}, y_{i,2j})_{1\leq j\leq 2m}$ into a qubit with $y_{i,2j}$ the basis value and $y_{i,(2j-1)}$ the bit value. Here, we assume that each bit of response is generated independently uniformly at random by an XOR Arbiter PUF ($p = \frac{1}{2}$).

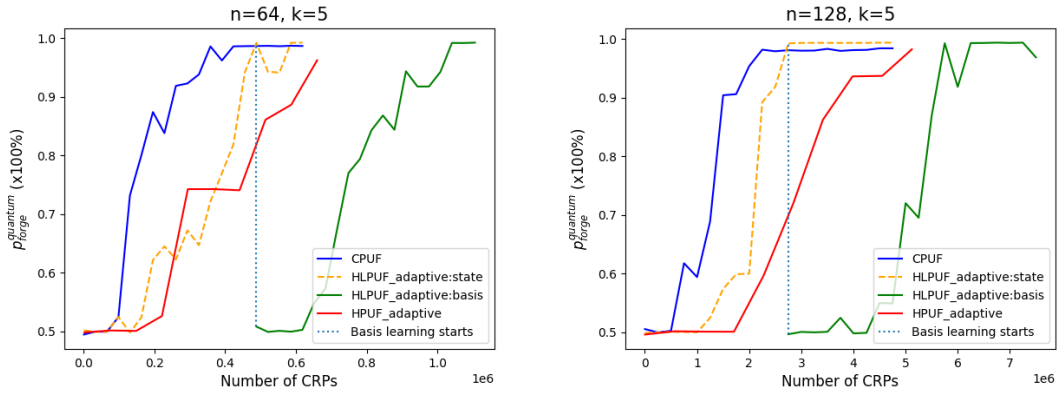
We simulate an adaptive adversary firstly on HPUF, where the adversary queries with the same classical challenge multiple times until he extracts the classical information from multiple copies of quantum response with high accuracy. The simulation results for modelling underlying CPUF are shown in red of Figure 4.6 and 4.7, where the X-axis denotes the number of CRPs we use for the forgery, and the Y-axis denotes the accuracy of the forgery.

Furthermore, while we consider HLPUF against an adaptive adversary, the lockdown technique reduces the capability of an adversary from adaptive to weak queries on HPUF. With a single copy of each quantum response uniformly at ran-



(a) (b)

Figure 4.6: LR attack performance on CPUF(in blue), HPUF(BB84, in red for modelling a qubit), and HLPUF(BB84, in green for modelling a qubit) with different CRPs as training set while the challenge size is 64 (4.6a)/128 (4.6b) bits with k=4 XORPUFs



(a) (b)

Figure 4.7: LR attack performance on CPUF(in blue), HPUF(BB84, in red for modelling a qubit), and HLPUF(BB84, in green for modelling a qubit) with different CRPs as training set while the challenge size is 64 (4.7a)/128 (4.7b) bits with k=5 XORPUFs

dom, we intuitively think that the adversary has a 50% probability of guessing the basis value correctly for each qubit of HPUF. If he guesses the basis value correctly, he can then measure the qubit correctly to obtain the exact $(y_{i,(2j-1)}, y_{i,2j})$. Otherwise, the classical tuple $(y'_{i,(2j-1)}, y'_{i,2j})$ of each qubit obtained by the adversary is always incorrect. Hence, the success probability of recovering each tuple $\{(y_{i,(2j-1)}, y_{i,2j})\}$ from corresponding qubit $|\psi^{i,j}_{\text{out}}\rangle\langle\psi^{i,j}_{\text{out}}|$ by such an adversary is not greater than guessing a tossing coin.

In practice, we discover a specific way to attack HPUFs throughout the simulation, so-called *split attack*. To the best of our knowledge, it is the optimal strategy that a weak adversary can perform on HPUF with underlying XORPUFs. We elaborate the attack as follows: Instead of predicting the tuple $(y_{i,(2j-1)}, y_{i,2j})$ simultaneously, the adversary first predicts the bit value $y_{i,(2j-1)}$ of each qubit. For

the HPUF with BB84 states encoding, the problem of distinguishing a state from uniformly distributed BB84 states then reduces to the problem of distinguishing two mixed states $\rho_1^{i,j} = \frac{1}{2}|0\rangle\langle 0| + \frac{1}{2}|+\rangle\langle +|$ and $\rho_2^{i,j} = \frac{1}{2}|1\rangle\langle 1| + \frac{1}{2}|-\rangle\langle -|$ with equal probability.

From Lemma 4.1, we get the optimal success probability as,

$$\begin{aligned}
&\Pr[\mathcal{A}^{i,j}_{\text{guess}}(x_i, \rho_1^{i,j}, \rho_2^{i,j}) = y_{i,(2j-1)}] \\
&\leq \frac{1}{2} + \frac{1}{2}(\frac{1}{2}\left\|\rho_1^{i,j} - \rho_2^{i,j}\right\|_1) \\
&= \frac{1}{2} + \frac{1}{2\sqrt{2}} \\
&\approx 0.85.
\end{aligned} \tag{4.40}$$

As it is to say, the adversary $\mathcal{A}$ can perform LR attacks on bit value with a 15% error afflicted CRPs training set. We do the simulation of HPUF, with BB84 encoding and an underlying of 4-XOR Arbiter PUF and 5-XOR Arbiter PUF and a challenge size of 64 bits and 128 bits. Here, $k = 4/5$ of XOR Arbiter PUF is the parameter related to its hardware structure. With a higher value of k of XORPUF, it takes more CRPs to model accurately with LR attacks. The evolution of accuracy in predicting the bit value of each qubit with different underlying XORPUFs are shown in orange of Figure 4.6 and 4.7.

After the bit value of each qubit can be predicted accurately with a given challenge, the problem of predicting the basis value $y_{i,2j}$ of the following qubits is equivalent to the adversary discriminates either a quantum state $|0\rangle$ from $|+\rangle$ if $y_{i,(2j-1)} = 0$ or a quantum state $|1\rangle$ from $|-\rangle$ if $y_{i,(2j-1)} = 1$. We denote the success probability of guessing the basis value correctly conditioned on an accurate prediction on bit value $y'_{i,(2j-1)} = y_{i,(2j-1)}$ by $Pr[\mathcal{A}^{i,j}_{\text{guess}}(x_i, |\psi^{i,j}_{\text{out}}\rangle\langle\psi^{i,j}_{\text{out}}|) = y_{i,2j}|y'_{i,(2j-1)} = y_{i,(2j-1)}]$ from a quantum state $|\psi^{i,j}\rangle\langle\psi^{i,j}|$, we have:

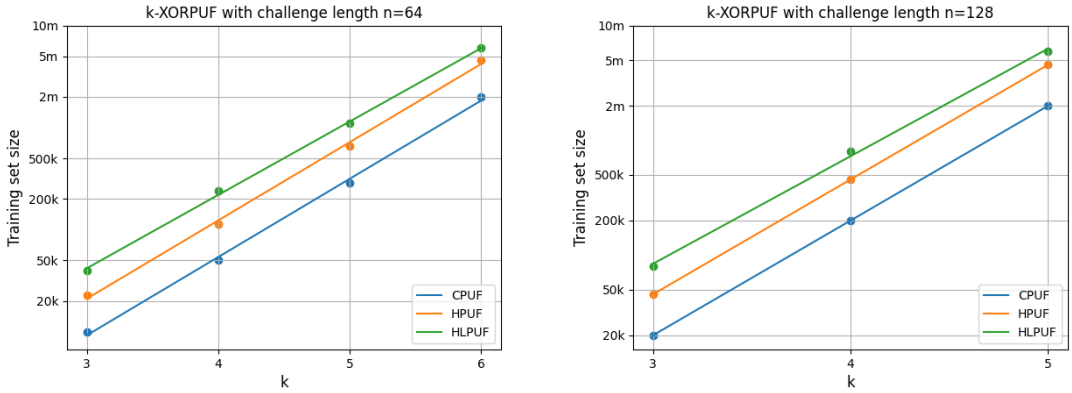
$$\begin{aligned}
&\Pr[\mathcal{A}^{i,j}_{\text{guess}}(x_i, |\psi^{i,j}_{\text{out}}\rangle\langle\psi^{i,j}_{\text{out}}|) = y_{i,2j}|y'_{i,(2j-1)} = y_{i,(2j-1)}] \\
&= \frac{1}{2} + \frac{1}{2}\sin 45^\circ \approx 0.85.
\end{aligned} \tag{4.41}$$

With the same level of noise introduced by HPUF on guessing the basis value and bit value, the similar performance of LR attack is expected to predict the basis value as long as the prediction accuracy of the bit value is high enough. We have the success probability of guessing both bit and basis values of tuple $(y_{i,(2j-1)}, y_{i,2j})$ as:

$$\begin{aligned}
&\Pr[\mathcal{A}^{i,j}_{guess}(x_i, |\psi^{i,j}_{\text{out}}\rangle\langle\psi^{i,j}_{\text{out}}|) = (y_{i,(2j-1)}, y_{i,2j})] = \\
&\Pr[\mathcal{A}^{i,j}_{\text{guess}}(x_i, |\psi^{i,j}_{\text{out}}\rangle\langle\psi^{i,j}_{\text{out}}|) = y_{i,2j}|y'_{i,(2j-1)} = y_{i,(2j-1)}].
\end{aligned} \tag{4.42}$$

In the end, we get the evolution of accuracy on predicting a tuple $(y_{i,(2j-1)}, y_{i,2j})$ with different CRPs for training as the green curves in Figure 4.6 and 4.7. The gap between blue and green curves denotes the enhancement in terms of security by HPUF construction. See [193] for details of the simulation. We also simulate in Figure 4.8 the best-performing training set sizes of CRPs for obtaining accurate

enough models from machine learning attacks with different k-XORPUFs in the cases of CPUFs, HPUFs, and HLPUFs constructions. See [193] for details of the simulation.

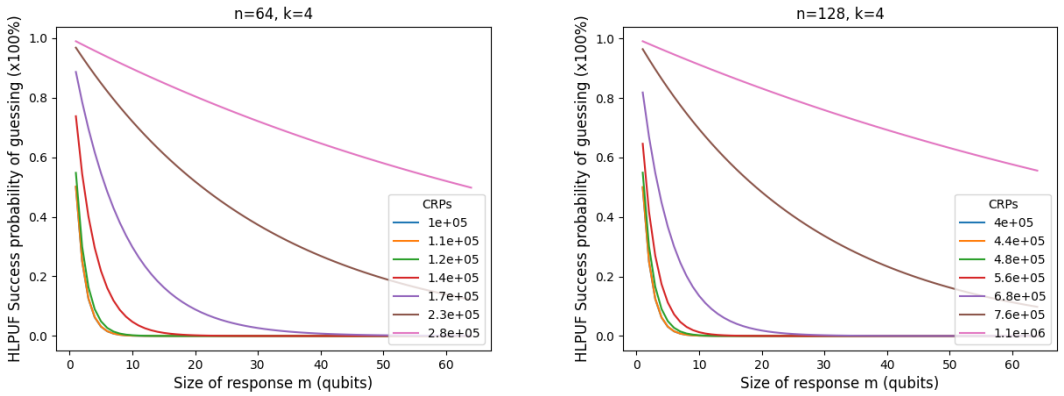


(a) (b)

Figure 4.8: Attack with best-performing number of CRPs for k-XORPUFs (CPUF, HPUF and HLPUF constructions) with challenge length $n = 64/128$ and BB84 encoding

In our simulations, the constructions of HPUF and HLPUF with underlying Arbiter-based PUFs generate a 1-bit response per query, thus although one can observe the exponential gap for a fixed number of queries between CPUF and HPUF, the inverse exponential scaling with m cannot be witnessed. While for a general m-qubit response construction this inverse-exponential scaling can be seen from the theoretical results. In Figure 4.9, we also attempt to simulate this behaviour for a m-qubit response constructed by several Arbiter-based PUFs. The construction is a relatively trivial one via parallelism, i.e., we simply duplicate the single structure m times and query them by the same challenge [106]. We note that this construction is far from optimal in terms of security, as it does not provide the independent m-qubit outcome required in the theoretical result. Hence, it allows the adversary to perform more effective parallel attacks. However, we can still see that the guessing probability of an eavesdropper decreases inverse exponentially on m until the averaged learning models are all accurate enough (See Figure 4.9 with 4-XORPUFs and different lengths of challenges). Moreover, the quantum encoding can, in any case, help with the detection of a network adversary trying to perform ML attacks, as such adversaries will perturb the quantum state in the quantum channels due to measurement, enabling the honest parties to detect their existence with high probability, and preventing the adversary from learning m-qubit states simultaneously during the protocol, as we show later in Section 4.6.2.

Corresponding to our proofs in Lemma 4.3 and Theorem 4.2, our simulation shows an exponential advantage of HPUF compared to the same CPUF with a limited q-query in terms of the modelling success probability against an adversary by LR attacks. As to a larger q-query, the advantage shown in the simulation limits by the fact that k-XORPUFs is a vulnerable CPUF with a large ε, which allows a modelling attack with a noisy data set. That is to say, the probability p_{extract} can be high with $\text{Dist}(\tilde{D}_q^x, D_q^x) = 0.15$. As long as $Pr(1, \frac{1}{2}, q) = 1 - negl(\lambda)$, the



(a) (b)

Figure 4.9: HLPUF (BB84) success probability of guessing with 4-XORPUFs and challenge length $n = 64/128$

success probability of modelling with hybrid construction converges to $1 - negl(\lambda)$ with an increasing q. Therefore, to decrease the forging probability in practice, there are mainly two directions: Firstly, we choose more robust underlying CPUFs to construct HPUF with lower ε and $Pr(1, \frac{1}{2}, q) = 1 - negl(\lambda)$ with a greater q. Second, we can consider other sophisticated encodings of HPUF, e.g., MUB encoding of quantum states with higher dimensions. In the following, we show the construction of HPUF with MUB encoding in 8-dimension and the simulation result.

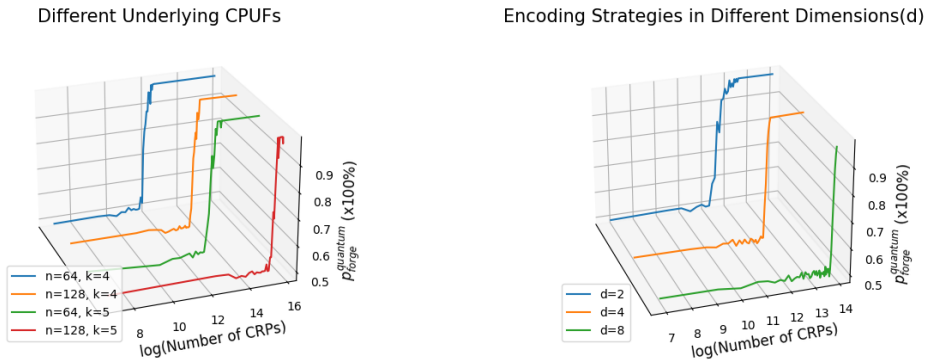
4.5.2 Practical Solutions for Boosting the Security

In the simulation, we observe that if we increase the value of k in the underlying k-XOR PUFs, the adversary requires more challenge-response pairs for a successful forgery. This observation suggests that one possible way to enhance the security of the HLPUFs is to use more secure classical PUFs. Hence, we elaborate on the effect of different k-values on the HLPUF forgery in Figure 4.10a. Moreover, the red plot in this figure also suggests that one can improve the security of HLPUFs significantly just by increasing the input size of the HLPUFs.

We also explore another possible way to improve the security of the HLPUFs. The idea is to use a more sophisticated encoding than encoding two classical bits into a quantum state $|\psi\rangle$ such that $|\psi\rangle \in \{|0\rangle, |1\rangle, |+\rangle, |-\rangle\}$. Here we use the concept of Mutually Unbiased Bases (MUBs) [194] of dimension $d = 4$ or $d = 8$ for the encoding. For the dimension $d = 4$ ($d = 8$), we encode four (six) classical bits to a two (three) qubits quantum state. Intuitively, the higher dimensional encoding leads to more noise introduced to the database that an adversary emulates CPUFs with, and thus helps to reduce substantially the value of p_{guess} in the measure-then-forge strategy significantly. For example, by encoding an 8-dimensional quantum state with 9 MUB [195], we denote the encoding quantum state as:

$$|x^{\theta}\rangle, x = x_0x_1x_2 \text{ and } \theta \in \{0, 1, ...8\}, \tag{4.43}$$

where θ represents the basis and x represents the state. We denote the set of basis



(a) (b)

Figure 4.10: Comparison of LR attack performance on HLPUFs with different underlying CPUFs (4.10a) and different encodings (4.10b) strategies

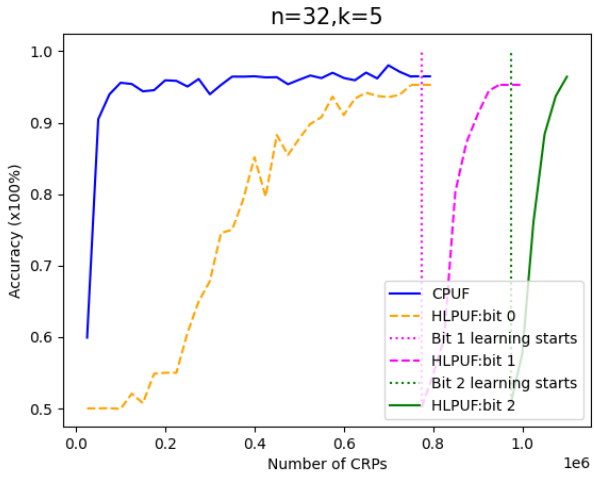


Figure 4.11: Evolution of LR attack performance on classical (in black) and hybrid (MUB in 8-dimension encoding, in red for modelling 3-qubit) constructions with different CRPs as the training set while the challenge size is 32 bits with k=5 XORPUFs

vectors for each basis using the matrices $B^\theta, \theta \in \{0, 1, ...8\}$. The column B^θ_j denotes the j^{th} basis vector for the basis set θ. The MUB set is given as:

$$\begin{aligned} B = \{&\mathbb{I}_8, \mathbf{O}\otimes\mathbf{O}\otimes\mathbf{O}, \mathbf{U}(\mathbf{O}\otimes\mathbf{O}\otimes\mathbf{I}), \mathbf{V}(\mathbf{O}\otimes\mathbf{I}\otimes\mathbf{O}), \\ &\mathbf{W}(\mathbf{O}\otimes\mathbf{I}\otimes\mathbf{I}), \mathbf{W}(\mathbf{I}\otimes\mathbf{O}\otimes\mathbf{O}), \mathbf{V}(\mathbf{I}\otimes\mathbf{O}\otimes\mathbf{I}), \\ &\mathbf{U}(\mathbf{I}\otimes\mathbf{I}\otimes\mathbf{O}), \mathbf{I}\otimes\mathbf{I}\otimes\mathbf{I}\} \end{aligned} \tag{4.44}$$

where $\mathbf{O} = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}, \mathbf{I} = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ i & -i \end{bmatrix}$,
$\mathbf{U} = \text{diag}\{1, 1, 1, 1, 1, -1, -1, 1\}$,
$\mathbf{V} = \text{diag}\{1, 1, 1, -1, 1, -1, 1, 1\}$,
$\mathbf{W} = \text{diag}\{1, 1, 1, -1, 1, 1, -1, 1\}$

Here, the adversary attempts to obtain the accurate models of $x_0x_1x_2$ from 3 CPUFs associated with the state value. Similarly to the strategy shown in BB84 encoding, the optimal strategy is to perform a split attack as shown previously on modelling bit by bit of $x = x_0x_1x_2$. In general, we have:

$$p_{dist}(\rho_0, \rho_1) = \max_E(\frac{1}{2} + \frac{1}{2}Tr(E(\rho_0 - \rho_1))) \tag{4.45}$$

, which denotes the optimal probability of distinguishing two mixed states with POVM element E. For each mixed state $\rho_x = \frac{1}{9}\sum_{\theta=0}^{8}|x^\theta\rangle\langle x^\theta|$, the optimal success probabilities of guessing x_0, x_1 and x_2 are given as (See [193] for more details):

$$\begin{aligned}
p_0 &= p_{guess}(x_0) = p_{dist}(\frac{1}{4}\sum_{i=0}^{3}\rho_i, \frac{1}{4}\sum_{i=4}^{7}\rho_i) \approx 0.62 \\
p_1 &= p_{guess}(x_1|x_0) \\
&= \frac{1}{2}(p_{guess}(x_1|x_0 = 0) + p_{guess}(x_1|x_0 = 1)) \\
&\leq p_{dist}(\frac{\rho_0 + \rho_1}{2}, \frac{\rho_2 + \rho_3}{2}) + p_{dist}(\frac{\rho_4 + \rho_5}{2}, \frac{\rho_6 + \rho_7}{2}) \\
&\approx 0.69 \\
p_2 &= p_{guess}(x_2|x_0, x_1) \\
&= \frac{1}{4}\sum_{i,j\in\{0,1\}} p_{guess}(x_2|x_0 = i, x_1 = j) \\
&\leq \frac{p_{dist}(\rho_0, \rho_1) + p_{dist}(\rho_2, \rho_3) + p_{dist}(\rho_4, \rho_5) + p_{dist}(\rho_6, \rho_7)}{4} \\
&\approx 0.77
\end{aligned} \tag{4.46}$$

The result gives us an upper bound on the probabilities, allowing us to fit this attack into our existing simulation framework easily while only giving more power to the adversary, i.e., in an actual scenario, the number of CRPs required to obtain an accurate model would be the same or more than in our simulations.

In Figure 4.10b, we show the impact of this encoding on the forging probability. Specifically, we show an interesting simulation result in Figure 4.11, where we only use 32-bits input 5-XOR PUF as an underlying CPUF. For such CPUFs, the total number of possible challenges is $2^{32} \approx 10^9$. In Figure 4.11, we observe that the underlying CPUF can be forged using only 5000 CRPs. On the other hand, for the forgery of the HLPUFs, the adversary requires almost 10^6 queries. For the forgery of the HLPUF, the adversary needs to use almost all the CRPs. Therefore, we can enhance the security of the HLPUFs by using higher-dimensional MUBs. However, the MUB in an 8-dimension encoding setting (or high dimensions) requires multi-qubit gates on both the server and client sides. Hence, there is a trade-off between the complexity of encoding and implementation effort. Furthermore, we should consider the imperfect quantum channels and measurements with the HPUF setting. We leave these as one of our benchmarking works in the future.

4.6 HLPUF-based Authentication Protocol

From the introduction of the HPUF/HLPUF constructions, as well as the security proofs and numerical simulation results are given, we believe that most of the applications based on PUFs in the classical world are compatible with H(L)PUFs, and obtain effective enhancements in terms of security with practical quantum information technology, and no overhead on quantum resources. Moreover, we observe that the quantum responses and the existence of uncertainty relations in quantum information provide us with another property, so-called *reusability*. In this section, we propose an HLPUF-based authentication protocol security guarantees and the reusability property that we can implement practically.

We first give the formal description in Protocol 1 and Figure 4.12 an illustration of the protocol.

Protocol 1 Hybrid PUF-based Authentication Protocol with Lockdown Technique

1. **Setup:**
   (a) The Prover $\mathcal{P}$ equips a Hybrid Locked PUF: $\mathcal{E}_f^L$ with HPUF $\mathcal{E}_f : \{0,1\}^n \to (\mathcal{H}^2)^{\otimes 2m}$ constructed upon a classical PUF $f : \mathcal{X} \to \mathcal{Y}$. Here, the classical PUF f maps an n-bit string $x_i \in \{0,1\}^n$ to an $4m$-bit string output $y_i \in \{0,1\}^{4m}$.
   (b) The Verifier $\mathcal{V}$ has a classical database $D := \{(x_i, y_i)\}_{i=1}^d$ with all d CRPs of f, as well as the necessary quantum devices for preparing and measuring quantum states.
2. **Authentication:**
   (a) $\mathcal{V}$ randomly chooses a CRP (x_i, y_i) and splits the response equally into two partitions $y_i = f_1(x_i)||f_2(x_i) = y_i^1||y_i^2$ with length $2m$.
   (b) $\mathcal{V}$ then encodes the first partition of response into $|\psi_{f_1(x_i)}\rangle\langle\psi_{f_1(x_i)}| := \bigotimes_{j=1}^m |\psi^{i,j}_{f_1(x_i)}\rangle\langle\psi^{i,j}_{f_1(x_i)}|$ and issues the joint state $(x_i, |\psi_{f_1(x_i)}\rangle\langle\psi_{f_1(x_i)}|)$ to the client.
   (c) $\mathcal{P}$ receives the joint state $(x_i, \tilde{\rho}_1)$ and queries Hybrid Locked PUF $\mathcal{E}_f^L$. If the verification algorithm $\texttt{Ver}(|\psi_{f_1(x_i)}\rangle\langle\psi_{f_1(x_i)}|, \tilde{\rho}_1) \geq 1 - \epsilon(\lambda)$ with negligible $\epsilon(\lambda)$, $\mathcal{P}$ obtains $|\psi_{f_2(x_i)}\rangle\langle\psi_{f_2(x_i)}| := \bigotimes_{j=1}^m |\psi^{i,j}_{f_2(x_i)}\rangle\langle\psi^{i,j}_{f_2(x_i)}|$ from $\mathcal{E}_f^L$ and sends back to $\mathcal{V}$. Otherwise, the authentication aborts.
   (d) $\mathcal{V}$ receives the quantum state $\tilde{\rho}_2$ and performs the the verification algorithm $\texttt{Ver}(.,.)$ as described in Construction 4.2. If $\texttt{Ver}(|\psi_{f_2(x_i)}\rangle\langle\psi_{f_2(x_i)}|, \tilde{\rho}_2) \geq 1 - \epsilon(\lambda)$ with negligible $\epsilon(\lambda)$, the authentication passes. Otherwise, it aborts.

In each authentication round of the protocol, the verifier (server) uses a classical database and a quantum encoder to create the required form of challenge for HLPUF, which consists of two parts: the classical challenge x, and the quantum state $|\psi_{f_1(x)}\rangle$, constructed based of the first half of the classical response, stored in

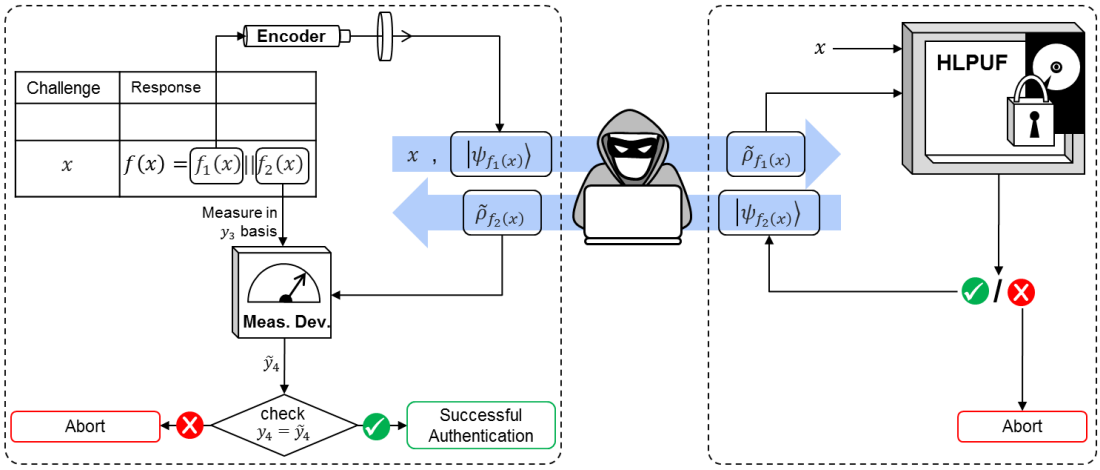


Figure 4.12: HLPUF-based authentication protocol

the database. Then, the verifier sends them through a public channel fully controlled by a quantum adversary, as illustrated in the figure. The prover (client) then inputs this two-part challenge into the HLPUF and either receives the state $|\psi_{f_2(x)}\rangle$ or gets a reject outcome and aborts the protocol, meaning the message did not come from the authentic verifier. The prover then sends back the quantum state to the verifier through the same public quantum channel, which will verify the client's response by measuring in y_3 according to the classical database. Recall that here, $f_2(x) = y_3 y_4$. Also, $\tilde{\rho}_{f_1(x)}$ and $\tilde{\rho}_{f_2(x)}$ denote the real quantum state received by the prover and verifier, respectively, after the adversary's interaction with the original states.

4.6.1 Security Analysis

Following the description of the HLPUF-based authentication protocol, we first give the completeness definition of our HLPUF-based authentication protocol.

Definition 4.3 (Completeness of HLPUF-based Authentication Protocol 1)**.** *We say the HLPUF-based authentication protocol 1 satisfies completeness if in the absence of any adversary, an honest client and server generating* $|\psi_{f_1(x_i)}\rangle\langle\psi_{f_1(x_i)}|$ *and* $|\psi_{f_2(x_i)}\rangle\langle\psi_{f_2(x_i)}|$ *with a valid HLPUF* $\mathcal{E}_f^L$ *for any selected challenge* x_i*, can pass the verification algorithms with overwhelming probability:*

$$\begin{aligned}\Pr[&\mathit{Ver}(|\psi_{f_1(x_i)}\rangle\langle\psi_{f_1(x_i)}|, \tilde{\rho}_1) \\ &= \mathit{Ver}(|\psi_{f_2(x_i)}\rangle\langle\psi_{f_2(x_i)}|, \tilde{\rho}_2) = 1] \geq 1 - \epsilon(\lambda)\end{aligned} \tag{4.47}$$

Furthermore, we also define the security of our HLPUF-based authentication protocol in relation to the universal unforgeability game as follows:

Definition 4.4 (Security of the HLPUF-based Authentication Protocol 1)**.** *We say the HLPUF-based authentication protocol 1 is secure if the success probability of any QPT adaptive adversary* $\mathcal{A}_{ad}$ *(a q-query adaptive adversary for any polynomial q) in winning the universal unforgeability game to forge an output of HLPUF* $\mathcal{E}_f^L$

according to Construction 4.2, for any randomly selected challenge of the form $\tilde{c} = (x, |\psi_{f_1(x)}\rangle\langle\psi_{f_1(x)}|)$ *is at most negligible in the security parameter:*

$$\Pr[1 \leftarrow \mathcal{G}^{\mathcal{E}_f^L}(\mathcal{A}_{ad}, \lambda)] \leq \epsilon(\lambda) \tag{4.48}$$

Now, we formally prove the completeness and security of Protocol 1 as follows:

Theorem 4.5. *If the HLPUF* $\mathcal{E}_f^L$ *is constructed from a hybrid PUF* $\mathcal{E}_f$ *using the Construction 4.2, then the locked PUF-based authentication Protocol 1 satisfies both the completeness and security conditions.*

Proof. In Protocol 1 with hybrid PUF $\mathcal{E}_f = \mathcal{E}_{f_1} \otimes \mathcal{E}_{f_2}$, the server chooses the classical input $x_i \in \mathcal{X}$, encodes the quantum state corresponding to $2m$ bits of $f_1(x_i)$ and issues the joint state to the client. If there is no adversary, the client receives the joint state and queries $\mathcal{E}_f^L$ with x_i and $\tilde{\rho}_1$, where $\tilde{\rho}_1 = \mathcal{E}_{f_1}(x_i) = |\psi_{f_1(x_i)}\rangle\langle\psi_{f_1(x_i)}|$ for the first m qubits of $\mathcal{E}_f(x_i)$. Hence, we have the following:

$$\Pr\left[\texttt{Ver}(|\psi_{f_1(x_i)}\rangle\langle\psi_{f_1(x_i)}|, \tilde{\rho}_1) = 1\right] = 1 \tag{4.49}$$

On the client side, since the verification algorithm of HLPUF $\mathcal{E}_f^L$ always passes with $\texttt{Ver}(|\psi_{f_1(x_i)}\rangle\langle\psi_{f_1(x_i)}|, \tilde{\rho}_1) = 1$, he returns the quantum state $\mathcal{E}_{f_2}(x_i)$ corresponding to $2m$ bits of $f_2(x_i)$ to the server. Without the presence of an adversary, the server always receives the state with $\tilde{\rho}_2 = |\psi_{f_2(x_i)}\rangle\langle\psi_{f_2(x_i)}|$, and we obtain the equation similarly to Equation (4.49). Therefore, we can say the locked PUF-based authentication protocol satisfies the completeness condition with

$$\begin{aligned}\Pr[&\texttt{Ver}(|\psi_{f_1(x_i)}\rangle\langle\psi_{f_1(x_i)}|, \tilde{\rho}_1) \\ &= \texttt{Ver}(|\psi_{f_2(x_i)}\rangle\langle\psi_{f_2(x_i)}|, \tilde{\rho}_2) = 1] = 1\end{aligned} \tag{4.50}$$

On the other hand for security, we rely on Theorem 4.3 that the HLPUF $\mathcal{E}_f^L$ is (ϵ, m)-secure against any q-query adaptive adversaries. In the theorem, we show the fact that the adaptive adversary cannot boost from the weak-learning phase of HPUF $\mathcal{E}_{f_2}$, producing a forgery σ_2 for $\mathcal{E}_f^L$ that passes the verification $\texttt{Ver}(|\psi_{f_2(x_i)}\rangle\langle\psi_{f_2(x_i)}|, \sigma_2)$, reduces to forging the HPUF $\mathcal{E}_{f_2}$ by q-query weak adversary. Since $\mathcal{E}_{f_2}$ has the universal unforgeability against the weak adversary by assumption, we have:

$$\begin{aligned}\Pr[1 \leftarrow \mathcal{G}^{\mathcal{E}_f^L}(\mathcal{A}_{ad}, m)] &= Pr[1 \leftarrow \mathcal{G}^{\mathcal{E}_{f_2}}(\mathcal{A}_{weak}, m)] \\ &\leq \epsilon(m)\end{aligned} \tag{4.51}$$

This concludes the proof. □

As a result, we show that Protocol 1 is completeness and security against QPT adversaries with certain rational assumptions of underlying classical PUFs.

4.6.2 Challenge Reusability

In any PUF-based authentication protocols with challenge and response issued via classical channels, each challenge can be used only in a single authentication round

due to man-in-the-middle attacks. The problem arises because the adversaries can simply copy and record the challenges and responses by intercepting the classical communication channels and have a perfect copy of the challenger's database. Later, they can use it to identify themselves fakely. Therefore, the server needs to store an enormous database to run the authentication protocol for a long period. This is a fundamental limitation of classical PUFs [106, 196].

However, HLPUF provides an efficient and unique solution to this issue by exploiting the unclonability of the quantum states and the existence of uncertainty relations in quantum mechanics and quantum information. It allows the use of the same challenge several times for authentication without any security compromise. More precisely, each challenge-response pair can be reused under the circumstance of previous successful authentication rounds. This solution will resolve the important practical limitation of the challenger storing a big database or frequently renewing the database of challenge-response pairs.

We first clarify the condition under which the challenge can be reused. Recall that in hybrid construction, the challenges are still being sent as classical bitstrings over the public channel, and hence, the adversary, after polynomial rounds of communication, can have the same challenge set as the server's database. Due to this fact, we should emphasize that the adversary does not get any physical access to the internal classical PUF in the HLPUF construction during the authentication, and no query can be directly issued to the CPUF by the adversary. This condition is satisfied using the lockdown technique. Thus, the adversary has access to the following information: a set of classical challenges used during the protocol and the set of quantum states that encode the first/second half of the response in the BB84 states, as a proof of concept.

It is a straightforward observation that the challenges for which the verification test has failed should never be used again. In this case, a trivial attack would be that the adversary intercepts the communication and stores the response state, and later, when the same challenge has to be queried again, will re-send the stored correct response state to pass the verification. As a result, all the challenges in the failed rounds should be discarded.

Nonetheless, we show that the challenges can be reused in the event of successful authentication. Here, by successful identification, we mean that the received response state passes the verification on the client and server sides, and both of them are identified as honest parties. Even though the events of false identification of an adversary, are still possible (for example, if the challenge is the same as one of the challenges that previously existed in the adversary's local database).

For Protocol 1, we are interested in the eavesdropping attacks by the adversary on the first and second half of the response states that are of the form $|\psi_{f_1(x_i)}\rangle\langle\psi_{f_1(x_i)}| = \bigotimes_{j=1}^{m} |\psi^{i,j}_{f_1(x_i)}\rangle\langle\psi^{i,j}_{f_1(x_i)}|$ and $|\psi_{f_2(x_i)}\rangle\langle\psi_{f_2(x_i)}| = \bigotimes_{j=1}^{m} |\psi^{i,j}_{f_2(x_i)}\rangle\langle\psi^{i,j}_{f_2(x_i)}|$. Note that eavesdropping on the states that encode the first part of the response will lead to breaking the locking mechanism while eavesdropping on the second half will lead to an attack on the identification. Without loss of generality, we only consider one of the cases where the adversary wants to eavesdrop on the first (or second) half to break the protocol in the upcoming rounds where the challenge is reused. The arguments will hold equivalently for both cases since the states and verification are symmetric.

Given all these considerations, the challenge reusability problem will reduce to the optimal probability of the eavesdropping attack on $|\psi_{f_1(x_i)}\rangle\langle\psi_{f_1(x_i)}|$ which is, in fact, m qubit states encoded in conjugate basis is the same as BB84 states. In the most general case, the adversary can perform any arbitrary quantum operation on the state $\bigotimes_{j=1}^{m}|\psi^{i,j}_{f_1(x_i)}\rangle\langle\psi^{i,j}_{f_1(x_i)}|$ or separately on each qubit state $|\psi^{i,j}_{f_1(x_i)}\rangle$, together with a local ancillary system and sends a partial state of this larger state to the verifier to pass the verification test and keep the local state to extract the encoded response bits.

Let ρ_{SEC} be the joint state of the server, the eavesdropper, and the client. Since the states used in the protocol are from Mutually Unbiased Basis (MUB) states i.e. from either $Z = \{|0\rangle, |1\rangle\}$ or $X = \{|+\rangle, |-\rangle\}$, in order to show the optimal attack, we can rely on the entropy uncertainty relations that have been used for the security of QKD. The measurements for verification are also performed in the $\{Z, X\}$ bases accordingly. We use the entropy uncertainty relations from [32], where the security criteria for QKD have been given in terms of the conditional entropy for MUB measurements. Using these results, we show that the entropy of Eve in guessing the correct classical bits for the response is very high if the state sent to the verification algorithm passes the verification with a high probability. Intuitively, this is due to the uncertainty that exists related to the commutation relation between X and Z operators in quantum mechanics. Hence, we conclude that Eve's success probability in extracting information from the encoded halves of the response is relatively low. Also, we know that this uncertainty increases linearly with m, similar to the number of rounds for QKD. This argument results in the following theorem. In proving this theorem, we have used the entropic uncertainty relation, which is introduced in Section 2.1.4

Theorem 4.6. *In Protocol 1, let x be a challenge and $(y_1, \ldots, y_{2m})$ be the response of a classical PUF used inside the HPUF construction, with randomness bias $p = (\frac{1}{2}+\delta_r)^{2m}$ in generating the random classical responses. If the verification algorithm for a state $\tilde{\rho}$ passes with probability $1-\epsilon(m)$, then Eve's conditional min-entropy H_{min}^{Eve} in terms of von Neumann entropy over the server's (or client's) classical response, satisfies the following inequality:*

$$H_{min}^{Eve} = H_{min}(S^m|ER^m) \geq m - \epsilon(m) \tag{4.52}$$

Proof. We prove this theorem based on the first half of the state used in Protocol 1, i.e. the state $|\psi_{f_1(x_i)}\rangle\langle\psi_{f_1(x_i)}| = \bigotimes_{j=1}^{m}|\psi^{i,j}_{f_1(x_i)}\rangle\langle\psi^{i,j}_{f_1(x_i)}|$ that is being sent by the Server (S) and received and measured by the Client (C). Nevertheless, the same proof applies to the second state due to the symmetry of the states and the protocol.

Let $R^m = (R_1, \ldots, R_m)$ be the randomness bitstring showing the choice of the basis encoding of the response, $S^m = (S_1, \ldots, S_m)$ be the server's bit encoded in the R^m bases. Note that both R^m and S^m are produced according to the bitstring $(y_1, \ldots, y_{2m})$ which is the first half of the response of CPUF to a given challenge x. Also, let $C^m = (C_1, \ldots, C_m)$ be the client's correct bit string. We denote the arbitrary joint state of three systems by $\rho_{S^mEC^m}$ where E denotes any arbitrary quantum system held by the eavesdropper. Now, let the Client's measurement outcomes, after the verification be $\tilde{Y}^m = (\tilde{Y_1}, \ldots, \tilde{Y_m})$ which shows the estimated bits by the Client. Now we can write the tripartite uncertainty principle, in terms

of the von Neumann entropy, for MUB measurements and MUB states as follows:

$$\begin{aligned} &H(X_1X_2Z_3X_4\dots X_{m-1}Z_m|E)+ \\ &H(Z_1Z_2X_3Z_4\dots Z_{m-1}X_m|C) \geq \log_2\left(\frac{1}{c}\right)^m \end{aligned} \tag{4.53}$$

where $c = \max_{x,z} c_{xz}$ and $c_{xz} = \| \sqrt{M^x}\sqrt{N^z} \|^2$ for an arbitrary POVM sets $M = \{M^x\}_x$ and $N = \{N^z\}_z$. We note that if the CPUF creates a perfect random bitstring for R^m then states are perfect MUB states and $c = \frac{1}{2}$. Nonetheless, we consider a weaker CPUF with a biased distribution of $p = (\frac{1}{2} + \delta_r)^{2m}$ in creating 0s and 1s in the response. Hence, we can translate this imperfectness into a disturbance in the measurement bases. Let $M^0 = |0\rangle\langle 0|$ and $M^1 = |1\rangle\langle 1|$ be the usual measurement in the computational basis, but let the N measurements be a slightly shifted version of the measurements in the X basis. Consider the following states:

$$\begin{aligned} |\psi_N\rangle &= \sqrt{\frac{1}{2} + \delta_r}\,|0\rangle + \sqrt{\frac{1}{2} - \delta_r}\,|1\rangle \\ |\psi_N^\perp\rangle &= \sqrt{\frac{1}{2} - \delta_r}\,|0\rangle - \sqrt{\frac{1}{2} + \delta_r}\,|1\rangle \end{aligned} \tag{4.54}$$

We define the new N projective operators according to the following states as $N^0 = |\psi_N\rangle\langle\psi_N|$ and $N^1 = |\psi_N^\perp\rangle\langle\psi_N^\perp|$. Now we calculate the operator norm for all the pairs of measurements, and we have:

$$\begin{aligned} \| \sqrt{M^0}\sqrt{N^0} \|^2 &= \frac{1}{2} + \delta_r, \quad \| \sqrt{M^0}\sqrt{N^1} \|^2 = \frac{1}{2} - \delta_r \\ \| \sqrt{M^1}\sqrt{N^0} \|^2 &= \frac{1}{2} - \delta_r, \quad \| \sqrt{M^1}\sqrt{N^1} \|^2 = \frac{1}{2} + \delta_r \end{aligned} \tag{4.55}$$

Thus we conclude that $c = \frac{1}{2} + \delta_r$ and the Equation (4.53) can be re-written as follows:

$$\begin{aligned} &H(X_1X_2Z_3X_4\dots X_{m-1}Z_m|E)+ \\ &H(Z_1Z_2X_3Z_4\dots Z_{m-1}X_m|C) \geq m - m\log_2(1+2\delta_r) \end{aligned} \tag{4.56}$$

Now, as mentioned at the beginning of the section, using the data processing inequality [32], we have got the following security criteria that show Eve's uncertainty (in terms of the von Neumann entropy) of the actual response bits S^m:

$$H(S^m|ER^m) + H(S^m|\tilde{Y}^m) \geq m - m\log_2(1+2\delta_r). \tag{4.57}$$

We can get the same inequality in terms of smooth min and max entropy [32, 197], which is more appropriate for ensuring the security in the finite size, for min and max entropy we equivalently have:

$$H^\epsilon_{min}(S^m|ER^m) \geq m - H^\epsilon_{max}(S^m|\tilde{Y}^m) - m\log_2(1+2\delta_r) \tag{4.58}$$

In order to calculate the above bound, we need to find the bound on the $H^\epsilon_{max}(S^m|\tilde{Y}^m)$. Here we use another result from [197] where it states that for any bitstring X of

n bit and the respective measurement outcome X', which at most a fraction ζ of them disagree according to the performed statistical test, then the smooth max entropy is bounded as follows:

$$H^{\epsilon}_{max}(X|X') \leq nh(\zeta) \tag{4.59}$$

where $h(.)$ denotes the classical binary Shannon entropy. Now, we can use this result and our assumption of successful verification together. Given the assumption that the verification is passed with a probability $1 - \epsilon(m)$, and the verification algorithm consists of measuring the states in the Z and X bases, we can conclude that the final bits differ in at most a fraction $\zeta = \epsilon(m)$ where $\epsilon(m)$ is a negligible function. As a result, we have:

$$H^{\epsilon}_{max}(S^m|\tilde{Y}^m) \leq mh(\zeta) \approx m\epsilon(m) \tag{4.60}$$

Putting Equations (4.58) and (4.60) together, we have:

$$H^{\epsilon}_{min}(S^m|ER^m) \geq m - m\epsilon(m) - m\log_2(1 + 2\delta_r) \tag{4.61}$$

On the right-hand side of the above inequality, the second term is still a negligible function, and the third term depends on the CPUF bias probability distribution. We assume the CPUF satisfies p-Randomness, as defined in the Definition 3.3. Thus, the δ_r is a small value, and hence the term $(1 + 2\delta_r)$ is negligibly close to 1, which means that the third term is negligibly close to 0 in the security parameter, which is m. Finally, we conclude that:

$$H^{Eve}_{min} = H^{\epsilon}_{min}(S^m|ER^m) \geq m - \epsilon'(m) \tag{4.62}$$

where $\epsilon'(m)$ is a negligible function, and the proof is complete. □

By giving the above information-theoretic bound, we can formally use it to prove the challenge reusability of Protocol 1. We first define the reusability in relation to the unforgeability game in a formal way. Then, using Theorem 4.6, we prove the challenge reusability of the protocol.

Definition 4.5 (Challenge (k-)reusability in the universal unforgeability game)**.** *Let $\mathcal{G}_{re}(\lambda, \mathcal{A}, x_{k+1})$ be a special instance of the universal unforgeability game, where a challenge x, picked uniformly at random by the challenger, has been previously used k times. We are interested in the events where the same challenge is used in the $(k+1)$-th round, which we denote by x_{k+1}. We say the challenge x is (k-)reusable if the success probability of any QPT adversary in winning $\mathcal{G}_{re}(\lambda, \mathcal{A}, x_{k+1})$, i.e, in forging message x_{k+1}, is negligible in the security parameter:*

$$Pr_{forge}(\mathcal{A}, x_{k+1}) = Pr[1 \leftarrow \mathcal{G}_{re}(\lambda, \mathcal{A}, x_{k+1})] \leq \epsilon(\lambda) \tag{4.63}$$

Theorem 4.7 (Challenge reusability of HLPUF-based Authentication Protocol 1)**.** *A challenge x can be reused k times during the Protocol 1 as long as the received respective response σ for each round passes the (client's or server's) verification with overwhelming probability. In other words, under the successful verification, the success probability of the adversary in passing the $(k+1)$-th round with the same challenge x is bounded as follows:*

$$Pr_{forge}(\mathcal{A}, x_{k+1}) \leq k2^{-m} \approx \epsilon(m). \tag{4.64}$$

Proof. To prove this theorem, we directly use the Theorem 4.6. First, we assume that x has been used one time before in a previous round. Given the assumption that the verification is passed with probability $1 - \epsilon(m)$, and this theorem, we conclude that the uncertainty of the adversary in guessing the encoded response of the HLPUF is larger than $m - \epsilon(m)$. In our case, the joint quantum state between the server and the adversary is a classical-quantum state (server has the classical description of $f(x)$, and the adversary has the quantum state $|\psi_{f(x)}\rangle$). For such states, Eve's uncertainty, H_{min}^{Eve} is the same as $-\log P_{guess}^{Eve}$, where P_{guess}^{Eve} is Eve's guessing probability of the classical information encoded in the quantum state [37]. Therefore,

$$\begin{aligned} Pr_{guess}^{Eve} &= 2^{-H_{min}^{Eve}} \\ &\leq 2^{-m+\epsilon(m)} \end{aligned} \tag{4.65}$$

This probability is negligible in the security parameter, which means that after performing any arbitrary quantum operations, the adversary's local state includes, at most, a negligible amount of information on the response of x, each round that the state x is reused. Now, we can use the union bound to show that this success probability only linearly scales with k:

$$Pr_{guess}^{Eve,k} = P(\bigcup_{i=1}^{k} E_{guess}^{i}) \leq \sum_{i=1}^{k} P(E_{guess}^{i}) \approx k2^{-m}, \tag{4.66}$$

where E_{guess}^{i} are the events where Eve correctly guesses the response and $P(E_{guess}^{i}) = (P_{guess}^{Eve})^{i}$ is the success probability of Eve in guessing in the i-th round. Finally, let the success probability of an adversary in the universal unforgeability game for the HLPUF be upper-bounded by $\epsilon_1(m)$, which is a negligible function in the security parameter since we assume that the HLPUF satisfies the universal unforgeability. This is the same as the success probability of the adversary in passing the verification for a new challenge, chosen at random from the database.

Now, in the $(k+1)$-th round, where the same x is reused, the success probability is at most boosted by the guessing probability over the previous k-th rounds. Hence, we will have the following:

$$Pr_{forge}(\mathcal{A}, x_{k+1}) \leq \epsilon_1(m) + k2^{-m} = \epsilon(m) \tag{4.67}$$

As long as k is polynomial in the security parameter, the second term is also a negligible function, and since the sum of two negligible probabilities will also be negligible. This concludes the proof. □

4.7 Discussion

In this chapter, we proposed a new, practical way to enhance the security and usability of the hardware security primitive – PUF, using quantum communication technology and showed a new use case for quantum communication, which benefits from both provability and practicality. We classify the adversaries into adaptive and weak adversaries based on their querying capabilities. This classification is not only

useful in proof reductions but also provides a step-by-step path towards a provably secure PUF against the strongest possible quantum adversaries. By harnessing the power of quantum information theory, here we propose a construction for a hybrid PUF with classical challenge and quantum response. The main idea is to encode the output of classical PUF into non-orthogonal quantum states. We show that for the forgery of the HPUF, any q-query weak adversary first needs to extract the classical string $f(x)$ from the outcome of the HPUF. The adversary tries to forge the CPUF using that extracted data. Due to the indistinguishability of the non-orthogonal quantum states, the adversary introduces extra randomness at the outcome of the CPUF, which in turn complicates the forging task for any QPT adversary. We have established the result under the assumption that for a q random outcomes of the HPUF if the distance between the outcomes of CPUF and the extracted outcomes from the HPUF is above a threshold ε, then no QPT adversary can forge the HPUF. Under this assumption, we show that the probability of forging the HPUF is exponentially smaller than forging the CPUF. This is an exponential provable gap that is only achievable via quantum communication.

We also instantiated our HPUF design using real-world CPUF, called XOR-PUFs. In Figure 4.6 and Figure 4.7, we show the gap in the number of queries the adversary needs to forge the HPUF compared to the underlying CPUF. As displayed in those figures, the probability of the HPUFs being fully broken is considerably small compared to their underlying CPUF. However, using an enormous number of samples, the adversary eventually forges the HPUF, certifying the assumption in our theoretical result. A more sophisticated encoding can enhance this gap. Later in Figure 4.11, we show that the MUB of dimension 8 encoding of the outcome of the CPUFs can enhance this gap substantially.

In PUF-based authentication protocols, one important issue (both for classical and quantum PUFs) is that an adaptive adversary can query the PUF with arbitrary input challenges. It permits such an adversary to learn efficiently and emulate the input/output behaviour of the targeted PUF. We solve this problem with our quantum locking mechanism, leading to our HLPUF construction as discussed. In our proposed authentication protocol, we prove the security against adaptive adversaries. The advantage is twofold: On one hand, the probability of knowing information about a quantum state is upper-bounded compared to a classical PUF due to the quantum information theory. On the other hand, the implementation of hybrid PUFs is practical nowadays with the existing quantum communication technology.

Another advantage of the hybrid locked construction is the reusability of the challenge-response pairs, which was impossible prior to this work for similar protocols. Therefore, with our solution, a server can perform secure client authentication for an extended period without exhausting its CRPs database. This result overcomes the fundamental drawbacks of the existing classical PUF-based authentication protocols while putting forward a novel and practical use case for our HLPUF construction as well as a unique feature enabled solely by quantum communication.

The no-cloning property of quantum states also prevents passive adversaries from intercepting and storing the qubits for forgery without getting detected by the server/client. Unlike the classical setting, quantum communication forces all adversaries to behave like active ones. In general, it is impossible for adversaries to

extract information about the outcome of the underlying classical PUFs from the outcome of the HLPUFs without getting detected. This makes our HLPUF protocol cheat-sensitive, providing another advantage over CPUF-based authentication protocols.

The quantum communication part of our HLPUF construction relies on the conjugate coding, which is used in the quantum key distribution (QKD) protocols. QKD technology is one of the most mature quantum technologies. Long-distance QKD networks are already implemented and used in several countries like the USA, UK, China, EU, Japan, [198, 199, 200, 201, 202] etc. Many commercially available QKD infrastructures provide almost 300kb/s secret key rate over optical fibre links of length 120km [203]. Moreover, the availability of the mature QKD on-chip technology [204, 205, 206] makes all the proposed constructions in this paper implementable using existing quantum technology. Our results show that picking off-the-shelf classical PUF technology and QKD technology can partially solve significant shortcomings of the device authentication problem in a quantum network.

In the thesis, we show that our HLPUF construction makes the current-day insecure classical PUFs, secure with the help of quantum conjugate coding and lockdown techniques, and against present and future powerful quantum adversaries. However, all of our results are based on ideal implementations of the protocol. The next research direction will be to explore the performance of our HLPUF-based authentication protocol under channel noise and imperfect single-photon sources. Yet another intriguing research direction will be the design of robust variants of our protocol. Like some QKD protocols, our HLPUF becomes vulnerable to photon number splitting attacks if the source suffers from a multi-photon emission problem. Therefore, a further study of the feasibility and practicality of hybrid PUF constructions is an important future direction for bringing this technology from theory to practice.

Another interesting question arises in terms of the engineering design of the HLPUF, where a lockdown technique is exploited to prevent adaptive queries by network adversaries during usage. Explicitly, as a stand-alone construction, HLPUF construction implies a tamper-proof box where the underlying CPUF, as well as the quantum measurement and preparation apparatus, are under protection, except for the locked interface. A relevant question here is how a server can obtain a classical database of HLPUF given such tamper-proof environments. We argue that this is not an issue in the context of our proposed protocol and under the formal assumptions under which the protocol provides security guarantees. Firstly, we note that in the proposed protocols, the manufacturer, the server, and the client are all honest parties, and the construction of the HLPUF can be seen as a recipe for an honest manufacturer/server to construct such mechanisms given a CPUF which is potentially insecure, while followed by our adversarial model, the CPUF should not be queried directly at any point during the protocol. One can reasonably assume that the server first obtains the classical database of underlying CPUF prior to assembling HLPUF construction, then after assembling and sealing the box, transfers it to the client. We emphasise that such considerations will not affect the security guarantees of the protocol as they have been taken into account in our network adversarial model.

Nonetheless, we also propose an alternative solution that can be implemented at the hardware engineering level to ensure our assumptions are being met while enabling the HLPUF to operate as a stand-alone hardware token and not just within our given protocol. This can be achieved by integrating a *programmable read-only memory* (PROM) based device inside HLPUF while assembling by the manufacturer. A PROM is a type of non-volatile classical memory chip that permits data to be written in only once after the device's manufacture [207, 208]. Once PROM is programmed, its content cannot be changed, which means the data is permanent. In practice, a small piece of PROM is needed, with at least 2 registers, to enable the HLPUF device to switch between *setup* and *handover* modes. The mode-switch procedure can be performed as follows: When the manufacturer produces an HLPUF device within a tamper-proof box, the registers of PROM are set to value 11 as *setup* mode, and it can be queried from outside. Once the mode has been set differently, it can never go back to 11, which means that HLPUF has been used before in the setup mode. In setup mode, the server can query the box with classical queries. On the first classical query, the register updates the mode to 01 internally and will output classical responses as long as it stays so. After the setup is done, the server can set the value of registers to 00, in which case the encoding part of the device is activated, and the HLPUF will output the quantumly encoded queries, i.e., $|\psi_{f(x)}\rangle$. Of course, an adversary can do the same by querying HLPUF classically by setting registers from 11 to 01. However, this behaviour can be easily detected, and when an honest party (server) receives the box, they will not use the HLPUF box if it has ever been on a setup mode before. Furthermore, another engineering aspect to be taken is by harnessing device wear-out property to create limited access to the underlying CPUF [209]. Finally, we note that the most efficient and practical design for such boxes although an interesting engineering problem, is not in the scope of this paper and is a completely distinct direction for future works.

Chapter5

Trusted Execution Environment

A Practical Quantum Cloud Computing Solution with QEnclave

5.1 Introduction

In Chapter 3, we review another type of hardware security primitive as trusted execution environments (TEEs) with state-of-the-art research and implementations in Section 3.4. Recall that the core concept of TEEs is to provide a secure area in processors and isolate crucial applications and data from the main targeted computation (and the rest of the system) to guarantee the security properties of the targeted computation by means of dedicated hardware separations. Here, we focus especially on TEE-featured implementations within secure processors (also referred to as a *secure enclave*).

We study the abstraction of the functionality enabled by enclaves in common as attested execution, regardless of the numerous designs proposed for trusted hardware as secure enclaves in terms of different implementation details and cryptographic techniques. In the classical setting, the computing system equipped with a secure enclave on its hardware can forward a program, as well as the inputs, to the enclave. By evaluating the program over inputs, it computes the output, signs it with a secret key, and obtains a digital signature. Here, the signature is treated as an attestation that the program is executed internally inside the enclave.

The attested execution inside the secure enclaves can be viewed as a sandbox environment. In this sense, an adversary should not be able to tamper with the execution or try to intercept the data inside the enclave from outside. Besides securing local computations, a secure enclave can also be exploited in cloud computing scenarios. Considering a cloud computing scenario, a client with a public key of a secure enclave can establish a secure channel in between while the enclave is on the server's side. This enables the client to send encrypted and authenticated data or programs to the enclave. Even though the encrypted message passes through the server, the server should not be able to tamper with the communication channel to intercept the content of the message.

On the other hand, quantum computing is an emerging field of computation technology that promises to produce faster algorithms for solving computational problems [6, 210]. Many government agencies and large companies like Google, IBM and Amazon are putting efforts into building a programmable quantum device that can outperform existing classical computers [9, 10]. Some of them have already

managed to develop small-scale quantum computers and provide cloud services, allowing users to delegate their quantum computations [211, 212, 213, 214].

Although this form of delegated quantum computation (DQC) services is very useful in practice for education and research, running algorithms on untrusted quantum hardware raises, however, important privacy issues. A major challenge of DQC is to ensure the privacy of the client's computation, who does not have any quantum computation capability.

The first efficient universal protocol for secure delegated quantum computation in terms of privacy was introduced in [44], also seen from recent reviews for other similar protocols [215, 216]. However, these protocols all assume a quantum channel between the client and the server, which has proven impractical for some quantum hardware platforms, such as superconducting or cold atom qubits, at least in the near-term future. For this reason, constructing an efficient, private and secure DQC protocol using only classical communication will be extremely important. Given the impossibility of achieving information-secure delegated computing using only classical communication [217] other assumptions must be considered.

Recent breakthroughs based on post-quantum secure trapdoor one-way functions paved the way for developing entirely new approaches toward fully classical client protocols for emerging quantum servers [218, 219, 220]. Nevertheless, the challenge for these protocols is the huge server overhead. This is due to the fact that one has to ensure the quantum circuit implementing the required masking protocol based on the learning with errors (LWE) encryption [57] remains unhackable both classically and in quantum. That leads to current proposals that require an order of 1000 server qubits for masking a single gate of the target client computation.

Our work explores a different approach based on the hardware security assumption to derive a practical secure DQC protocol with a fully classical client setting. We explore the modular approach introduced in [221] that defines the remote state preparation (RSP) as the main building block for DQC protocol. It is worth noting that in [222], an RSP protocol was also proposed using a classical channel between client and server but assuming a resource called measurement buffer, which externalizes a quantum state measurement from the server's side. However, such a resource can not be realized classically, as it was proven in [223]. Indeed, it is known that it is impossible to construct a composably secure RSP protocol using only a classical channel between the client and the server without any hardware assumption, which confirms our approach to be the only way forward to construct an efficient DQC protocol with a classical client from the RSP module. In fact, the measurement buffer resource can indeed be implemented by trusted hardware. However, as discussed later, securing the measuring device creates an unnecessarily complicated architecture.

With these constraints in mind, we introduce the construction of QEnclave based on a classical secure enclave, as a practical way to perform equivalently RSP and achieve DQC with a purely classical client. Our QEnclave only transforms single-qubit states via rotations, as remote state rotation (RSR) functionality, without generating or measuring them. Nevertheless, it can be composed with the universal blind quantum computing protocol of [44] to achieve secure DQC with perfect blindness (assuming minimal hardware assumption) while using only

classical communication between the client and the server with optimal server overhead. Remarkably only one call to our simple hardware module is enough to create one remote blind qubit. Meanwhile, the blindness of the protocol holds even if the potentially malicious server controls the quantum source. We formally prove the security of QEnclave under the simulation-based security model and composable frameworks. By describing the abstraction of QEnclave as RSR functionality, we show that RSR resource constructs DQC with perfect blindness with a classical channel with composable security. Then, we give the real-world specification of QEnclave functionality inspired by the attested execution of secure enclaves with composable security. Finally, we explore the leverage of QEnclave in the DQC scenario with verifiability.

5.1.1 Structure of the Chapter

In Section 5.2, we first describe the functional specification of QEnclave as the resource of remote state rotation, then we further propose the design of QEnclave according to the construction of TEE in classical usage while combining limited quantum apparatus to achieve RSR functionality in practice. In Section 5.3, we formalise the security definitions of RSR in the context of remote state preparation functionality under the composability framework, where we refer our adversarial model of RSR to the adversarial model of the functionality of delegated quantum computation. Furthermore, in Section 5.4, we give the security analysis of RSR with two steps: In 5.4.1, we first introduce the idea of measurement-based remote state preparation (MRSP_B) resource. On the one hand, this resource constructs the functionality of blind delegated quantum computation by combining universal blind quantum computing protocol while replacing the quantum channel with simulation-based security and general composition. On the other hand, we show in the thesis that it also established the relation between the generic resource of remote state preparation and our proposed remote state rotation resource. Secondly, we show in subsections 5.4.2 and 5.4.3 how the remote state preparation resource and remote state rotation can construct the measurement-based remote state preparation, respectively. This allows, by accessing RSR for remote state preparation functionality, it is sufficient to build up a perfect blind delegated quantum computation with a real-world protocol. In Section 5.5, by showing the necessity of hardware assumption as quantum TEE for achieving efficient and secure remote state preparation-based functionalities, we specify a practical QEnclave-based outsourcing protocol for remote state rotation with blindness, with composable security. Meanwhile, we explore RSR by QEnclave in a stronger security notion as verification. Finally, we conclude this chapter in Section 5.6.

5.2 QEnclave Constructions

In Chapter 3, we describe the concept of constructing a trusted execution environment in quantum analogue (i.e., a quantum TEE). From the functionality point of view, a quantum TEE should be capable of performing meaningful quantum operations with integrity and confidentiality. Meanwhile, the operation within the

quantum TEE should be as simple as possible to reduce the surface of the interface to the external untrusted execution environment. In this section, we attempt, for the first time, to introduce the idea of constructing quantum TEE based on a classical secure enclave, so-called *QEnclave*. We start by introducing the abstraction of the functionality of QEnclave, as remote state rotation (RSR), and study RSR in the context of a general problem: *remote state preparation* (RSP). Furthermore, we propose an implementable architecture of QEnclave with hybrid classical-quantum construction, which integrates a secure enclave for encoding and decoding classical secrets and necessary quantum apparatus.

5.2.1 Abstraction of QEnclave: Remote State Rotation

As the abstraction of QEnclave functionality, RSR has two interfaces, A and B, for two different parties, Alice and Bob. The functionality of RSR is trivial: For each time of operation, it simply rotates an arbitrary input quantum state with dimension two from Bob, with an angle θ on X-Y plane chosen uniformly at random from the set $\{0, \frac{\pi}{4}, \ldots \frac{7\pi}{4}\}$. It then returns the state after rotation to Bob and the classical description of the rotation angle to Alice. A formal definition of RSR is given in the following:

Definition 5.1. *(See Figure 5.1) The ideal resource named remote state rotation (RSR) for blindness has two interfaces, A and B. After receiving a quantum state ρ_{in} with dimension 2 from interface B, it performs a single-qubit rotation $Z(\theta)$ with θ chosen uniformly at random from the set $\{0, \frac{\pi}{4}, \ldots \frac{7\pi}{4}\}$. It then outputs ($\rho_{\text{out}}$) at Bob's interface and the angle θ at Alice's interface.*

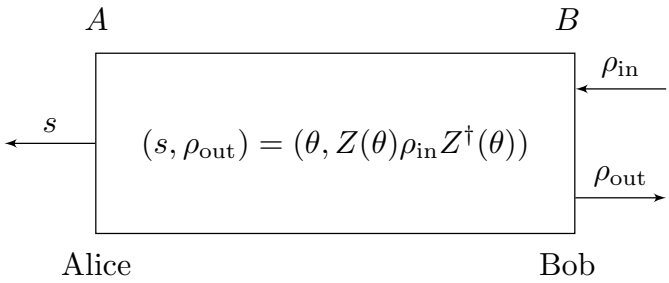


Figure 5.1: **Remote State Rotation Resource.** Remote State Rotation (RSR) performs arbitrary single-qubit rotation with angel θ on income quantum state, outputs the angle at interface A and the post-rotation quantum state at interface B.

Ideally, RSR can be treated as the functionality for preparing quantum states jointly by Alice and Bob. In this case, we further define a two-party protocol $\pi = (\pi_A, \pi_B)$ to prepare quantum states with RSR in which π_A only receives the angle θ from the interface A of RSR, and π_B takes as input a quantum state from Bob, and returns the state after rotation from RSR to Bob.

5.2.2 Practical Design with Secure Enclave

By introducing RSR as an abstraction of the functionality of QEnclave, we furthermore propose a hybrid classical-quantum construction of QEnclave heuristically by possibly using a TEE-enabled classical secure enclave, with the protection of quantum devices that implement the single-qubit rotations, as well as the flow in between. Recall that the guidelines of quantum TEE construction with near-term quantum hardware devices in Section 3.4.3, the only quantum operation for QEnclave design is the rotations on single-qubit states, which can be achieved by a single-qubit gate with classical controls. Meanwhile, there is no requirement for any quantum memory components with QEnclave.

While the QEnclave is placed on the side of Bob, it allows an implementation of the specification of RSR functionality. The security of RSR functionality is guaranteed by the classical enclave and its hardware assumption. Meanwhile, it allows communications classically only with Alice, performs single-qubit rotation on the quantum states from Bob inside QEnclave and returns the rotated states back to Bob, as shown in Figure 5.2.

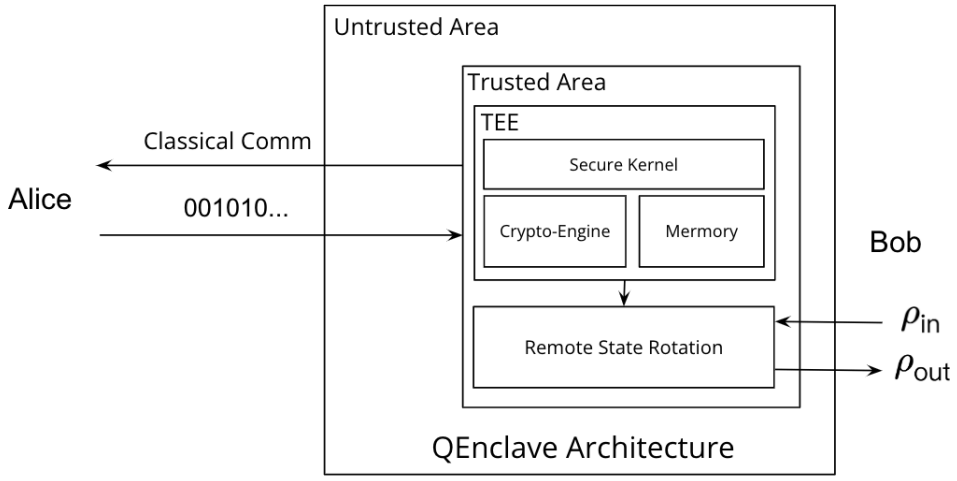


Figure 5.2: **Hardware Architecture of QEnclave.** The architecture of QEnclave is divided into two areas: The trusted and untrusted areas. The rotation angles from Alice can not be revealed when the rotation is performed inside the trusted area. The quantum source is external to the QEnclave.

5.3 Adversarial Model

We evaluate the security of RSR implemented by QEnclave by considering the scenario that Alice is always honest, and Bob is malicious with the capability to perform deviations. We phase the security analysis mainly from two perspectives: On the one hand is *blindness*, which means that a malicious Bob can not learn any information about the rotation angles of states. On the other hand, the second aspect is *verifiability*, which allows Alice to verify the operations of Bob and accept

results with deviations with negligible probability. In this work, we give a positive result on the security of blindness for RSR functionality. Nevertheless, we also discuss the possibility and limitations of achieving verifiability.

5.3.1 Blindness with Simulation-based Composable Security

In this section, we formalise adversarial behaviour and definitions of security. Note that, unlike Chapter 4, where we capture the security in the game-based model for PUF. Here, we exploit the approach of the simulation-based security model with composition theorems in *universal composability* (UC) or *abstract cryptography* (AC) to analyze the security of RSR by QEnclave construction. On the one hand, since RSR resource is aimed at preparing quantum states from a specific set, it is very important that the resource is capable of forming larger systems as a part of the modular composition while still preserving the security requirements. On the other hand, by means of showing the security of a large system in composition, we can obtain the security properties with its modular resources. Finally, it allows us to combine other resources that are possibly implementable in the real world to construct the ideal but not trivial functionalities we expect to achieve.

In the scope of the simulation-based security model with composability, we clarify the different behaviours while Bob is honest or malicious by additionally instantiating an indicator c in Section 5.2.1, as an input of π_B of $\pi = (\pi_A, \pi_B)$ of RSR resource from Bob. If $c = 0$, Bob is honest, and π_B accepts $|+\rangle\langle+|$ as input from Bob. If $c = 1$, Bob is malicious and prepares an arbitrary quantum state $\rho = \Omega(|+\rangle\langle+| \otimes \rho_{aux})\Omega^\dagger$. Here, Ω is an arbitrary unitary representing Bob's deviation in general, and ρ_{aux} denotes an auxiliary state of Bob. After tracing out the auxiliary state, we get ρ_{in} as the input state to RSR. In particular, this state can be entangled with Bob's auxiliary system.

Our adversary model of RSR and related security evaluations are associated with the models defined in delegated quantum computation (DQC). Here, we refer to [224] and adopt our definitions of DQC from there in the ideal world to capture different security requirements concerning blindness and verifiability. For blindness, the formal definition of the ideal resource of DQC with blindness $\mathcal{S}^{\text{blind}}$ is as follows:

Definition 5.2. (See [209]) *For a given unitary U, the ideal resource for DQC $\mathcal{S}^{blind}$ (see Figure 5.3) provides both correctness and blindness. It takes an input ψ_A at Alice's interface, and at Bob's interface, a filtered control bit c (set by default to 0) and a pair that consists in a state ψ_B and a description of a CPTP map $\mathcal{E}$. It outputs the allowed leak ℓ^{ψ_A} at Bob's interface. If $c = 0$, it outputs the correct result $U(\psi_A)$ at Alice's interface; otherwise it outputs Bob's choice, $\mathcal{E}(\psi_{AB})$*

Intuitively, the blindness states that at most ℓ^{ψ_A} of information leaked to Bob during the interactions. These permitted leaks are classical bitstrings that allow revealing information to Bob without compromising the privacy of the computation. For example, an upper bound on the size of the computation and the input/output of the computation should be classical or quantum.

For a real-world protocol that realises ideal resource with blindness, Broadbent, Fitzsimons, and Kashefi [44] propose a protocol called Universal Blind Quantum

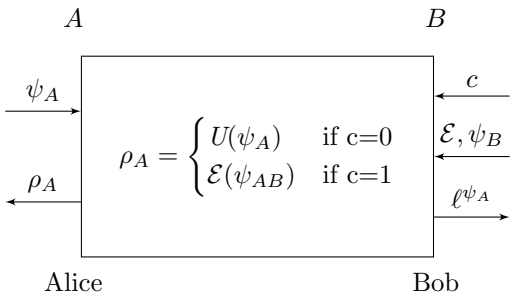


Figure 5.3: **DQC ideal resource with blindness.** The filtered control bit $c = 0/1$ denotes the honest/malicious behaviour of Bob. The filtered functionalities with the input ψ_B, $\mathcal{E}$ and output ℓ^{ψ_A} of resource are accessible only to a malicious Bob with $c = 1$.

Computing (UBQC) as a quantum computation model whose operations can easily be described in the MBQC model with an implementable resource which is a two-way quantum communication channel. At the start of a UBQC protocol, Alice produces a sequence of single-qubit states of the form $|+_\theta\rangle$ with θ chosen uniformly at random from $\{0, \frac{\pi}{4}, \ldots \frac{7\pi}{4}\}$. That is to say, each state prepared by Alice can be treated as a maximally mixed state, and no information about the classical description of each state is leaked to Bob. After receiving N such qubits from Alice through a quantum channel, Bob entangles them to build a universal family of graph states called the *brickwork* states with the definition in [44] as well. The computational stage is interactive and uses only classical communication. During this stage, Alice sequentially sends the measurement angle for each qubit to Bob, which returns the measurement result to Alice. Alice then computes the following measurement angle. At the end of the computation, Bob returns the outputs to Alice. As for the security of blindness, Dunjko, Fitzsimons, Portmann, et al., [224] show that the UBQC protocol provides perfect blindness compared to the ideal functionality in the AC framework. Note that in our work, we focus on the UBQC scenario with classical input-output. A formal theorem is given in their paper:

Theorem 5.1. *(See [224]) The UBQC protocol $\pi = (\pi_A, \pi_B)$ construct ideal resource $\mathcal{S}^{blind}$ with ε-blindness from both quantum and classical communication channels: $\mathcal{R}_{channels} = \mathcal{R}_{c\_channel} || \mathcal{R}_{q\_channel}$ provides perfect blindness. It satisfies:*

$$\mathcal{R}_{channels} \xrightarrow{\pi, \varepsilon=0} \mathcal{S}^{blind} \tag{5.1}$$

From the description above, we mention the quantum states that qualify as resource states in UBQC protocol for achieving blindness property. Furthermore, Dunjko and Kashefi [221] have introduced the concept of weak correlations, which is a necessary and sufficient condition on the set of states sent by Alice to obtain the blindness of the protocol. The following theorem formally introduces this notion.

Theorem 5.2. (See [221]) *The UBQC protocol with classical input and computation of size N, where Alice's preparation stage is replaced by the preparation of N states*

of the form σ^i_{AB}

$$\sigma^i_{AB} = \frac{1}{|\Theta|}\Sigma_{\theta_i\in\Theta}|\theta_i\rangle\langle\theta_i| \otimes \rho_i^{\theta_i}, \tag{5.2}$$

is blind if and only if the following conditions hold:

1. ρ^θ *is a normalized quantum state, for all* θ*,*
2. $\rho^\theta + \rho^{\theta+\pi} = \rho^{\theta'} + \rho^{\theta'+\pi}$ *for all* θ*,* $\theta' \in \Theta$ $(\Theta \in \{0, \frac{\pi}{4}, \dots \frac{7\pi}{4}\})$*,*
3. $|\Theta|$ *is the size of the set* Θ*.*

In other words, for all θ*,* $\rho^\theta + \rho^{\theta+\pi} = 2\eta$ *where* η_i *is some fixed quantum state. The correlations of the type given in Eq.(5.2) are called correlations for blindness, or weak correlations.*

Since we consider the UBQC protocol in the scenario of classical input-output, we naturally raise the question of whether a purely classical resource can replace the two-way quantum communication channel between Alice and Bob and maintain the security of blindness. To understand this question step by step, the resources related to remote state preparations are defined to formalise the security definition of the functionality.

Remote State Preparation (RSP) is an ideal resource introduced in [221]. In their work, RSP guaranteed that Alice acquires θ of each state. Meanwhile, Bob learns no information about the classical description of angle θ but obtains a prepared quantum state encoded by θ each time. From now on, we denote such an ideal resource RSP for blindness as (RSP_B).

Precisely, RSP_B is specified as follows: If Bob is honest, the resource outputs $|+_\theta\rangle\langle+_\theta|$ to Bob. If not, it takes as input from Bob the classical description of a quantum state $[\rho^\theta]$ and outputs the corresponding quantum state ρ^θ to Bob. In both cases, Alice receives the classical angle θ. A formal definition is given in the following:

Definition 5.3. *(See [221]) The ideal resource remote state preparation for blindness that is denoted* RSP_B*, has two interfaces, A to Alice and B to Bob (See Figure 5.4). The resource chooses an angle of rotation* θ *uniformly at random from the set* $\{0, \frac{\pi}{4}, \dots \frac{7\pi}{4}\}$*. There is a filtered functionality at interface B with a classical bit c. If* $c = 0$*, Bob is honest and the resource outputs a state* $|+_\theta\rangle\langle+_\theta|$ *on B. If* $c = 1$*, the ideal resource takes as input the set* $\{(\theta, [\rho^\theta])\}_\theta$ *from Bob, If the states provided by Bob do not satisfy the conditions from Theorem 5.2,* RSP_B *ignores the input and waits for a new valid set. Once the set is received, the resource outputs* ρ^θ *at Bob's interface. In both cases,* RSP_B *outputs the angle* θ *at Alice's interface.*

It is trivial to observe that the output of RSP_B establish exactly the weak correlation stated in Theorem 5.2. However, it is not sufficient to claim the security by saying this. Recall in Theorem 5.1, we know that the UBQC protocol constructs an ideal resource $\mathcal{S}^{\text{blind}}$ with perfect blindness from classical and quantum communication channels, as well as the quantum states preparation procedures in π_A of Alice. By replacing the quantum channel with the resource RSP_B, we give a formal statement of security of RSP_B by definition:

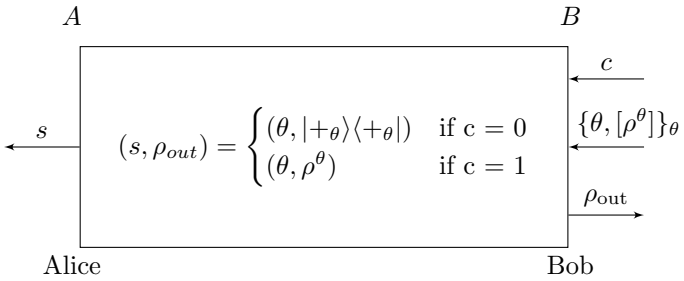


Figure 5.4: **RSP ideal resources for blindness.** The filtered control bit $c = 0/1$ denotes the honest/malicious behaviour of Bob. The filtered functionality with the input $\{(\theta, [\rho^\theta])\}_\theta$ of resource is accessible only to a malicious Bob with $c = 1$.

Definition 5.4 (Blindness of RSP_B against malicious Bob)**.** *The resource RSP_B provides ε-blindness by replacing the resource of a quantum channel if there exists a protocol $\pi = (\pi_A, \pi_B)$ between two parties interacting classically such that:*

$$RSP_B, \mathcal{R}_{c\_channel} \xrightarrow{\pi,\varepsilon} \mathcal{S}^{blind} \tag{5.3}$$

Similarly to RSP_B, a formal statement of security of RSR is as follows:

Definition 5.5 (Blindness of RSR against malicious Bob)**.** *The resource RSR provides ε-blindness by replacing the resource of a quantum channel if there exists a protocol $\pi = (\pi_A, \pi_B)$ between two parties interacting classically such that*

$$RSR, \mathcal{R}_{c\_channel} \xrightarrow{\pi,\varepsilon} \mathcal{S}^{blind} \tag{5.4}$$

5.4 Security Analysis

In this section, we give the security analysis of our proposed construction of QEnclave. First, we demonstrate the roadmap for proving the blindness of the resource RSP_B under the simulation-based security model with composability. Furthermore, we exploit the same proof techniques to show the blindness of RSR. As a result, with only access to the resource RSR and classical communication channels, we show that Alice can prepare quantum states remotely for further cloud quantum computation while guaranteeing confidentiality.

5.4.1 Measurement-based Remote State Preparation

First, we introduce another resource from [221] better suited for our purpose. It is a variant of RSP_B that allows for a larger set of options for a dishonest player. We show this later in the proof that the resource MRSP_B can be constructed from RSP_B.

Definition 5.6. *(See Figure 5.5) The ideal resource measurement-based remote state preparation for blindness ($MRSP_B$) has two interfaces, A and B. The resource chooses an angle of rotation θ uniformly at random from the set $\{0, \frac{\pi}{4}, \ldots \frac{7\pi}{4}\}$. There is a filtered functionality at the interface B and a classical bit c. If $c = 0$, Bob is honest and the resource outputs a state $|+_\theta\rangle\langle+_\theta|$ on B. If $c = 1$, the ideal resource takes as input the descriptions of eight positive operators $\{\Pi^\theta\}$, such that for all θ in $\{0, \frac{\pi}{4}, \ldots \frac{7\pi}{4}\}$, $\Pi^\theta + \Pi^{\theta+\pi} = I$. In addition, it accepts an arbitrary quantum state ρ of the same dimension as the operator Π^θ. If Bob's input does not satisfy the properties of Theorem 5.2, $MRSP_B$ ignores it and waits for a new valid set. Once a valid input is received, $MRSP_B$ applies the measurement $\Pi^\theta, \Pi^{\theta+\pi}$ corresponding to the chosen angle θ to ρ. Finally, $MRSP_B$ outputs the measurement result θ', whose value is either θ or $\theta + \pi$, at Alice's interface and the post-measurement state $\rho^{\theta'}$ at Bob's interface.*

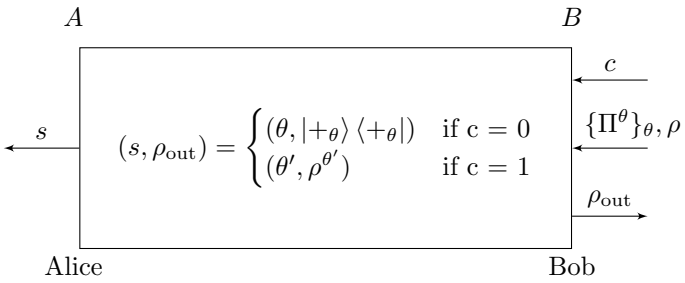


Figure 5.5: **Measured-based RSP for blindness.** The filtered control bit $c = 0/1$ denotes the honest/malicious behaviour of Bob. The filtered functionality with the input $\{\Pi^\theta\}$, ρ of resource is accessible only to a malicious Bob with $c = 1$.

With the resource MRSP_B, Dunjko and Kashefi [221] show that:

$$\text{MRSP}_B, \mathcal{R}_{\text{c\_channel}} \xrightarrow{\pi, \epsilon=0} \mathcal{S}^{\text{blind}} \tag{5.5}$$

The formal description of this theorem is given as follows:

Theorem 5.3. *(See [221]) The UBQC protocol in which Alice has access to the ideal functionality $MRSP_B$ rather than to a quantum channel and a random generator of the $|+_\theta\rangle$ states, exactly constructs DQC with perfect blindness.*

5.4.2 Perfect Blindness with RSP_B

In this section, we prove the perfect blindness of RSP_B. The core idea of the proof consists of two steps. We first show that the resource RSP_B can be used to construct a resource MRSP_B perfectly while following both correctness and security conditions with an existing simulator σ_B under the simulation-based security model. By the composition theorem, we show affirmatively that the UBQC protocol with MRSP_B accessing by Alice guarantees perfect blindness, as described in Definition 5.4.

By constructing MRSP_B from RSP_B, the protocol $\pi = (\pi_A, \pi_B)$ is trivial: it forwards the inputs from one interface to another. Then, we formally show that this protocol is sufficient to construct MRSP_B from RSP_B. Here, we complete the proof formally.

Theorem 5.4. *The protocol $\pi = (\pi_A, \pi_B)$ introduced above with ideal resource RSP_B constructs the ideal resource $MRSP_B$.*

Proof. We show that both the correctness and the security conditions are satisfied. More precisely, proving the security amounts to showing that a distinguisher cannot distinguish MRSP_B from the protocol. This translates into the following equations, for a simulator σ_B and the protocol $\pi = (\pi_A, \pi_B)$ with RSP_B.

$$\pi_A \mathrm{RSP}_B \pi_B \approx_\varepsilon \mathrm{MRSP}_B \perp, \tag{5.6}$$

and

$$\pi_A \mathrm{RSP}_B \approx_\varepsilon \mathrm{MRSP}_B \sigma_B. \tag{5.7}$$

The correctness follows from the definition of the resources in Definition 5.3 and 5.6, where the outputs are the same in the honest case. For security, the simulator σ_B can be defined as follows: It takes c, as well as the set $\{(\theta, [\rho^\theta])\}_\theta$ from Bob. σ_B then checks if the states provided by Bob satisfy the conditions from Theorem 5.2. If not, it ignores the input and waits for a new valid set. If so, since:

$$\rho^\theta + \rho^{\theta+\pi} = 2\eta. \tag{5.8}$$

By rewriting η into

$$\eta = \Sigma_{k=1}^{M} \lambda_k |\psi_k\rangle\langle\psi_k|. \tag{5.9}$$

The simulator then prepares the state $\eta_{AB} = \Sigma_{k,k'} \sqrt{\lambda_k \lambda_{k'}} |\psi_k\rangle_A \langle\psi_{k'}| \otimes |\psi_k\rangle_B \langle\psi_{k'}|$. Let:

$$\begin{aligned} \Pi^\theta &= \frac{1}{2} \eta^{-1/2} \rho^\theta \eta^{-1/2} \\ \Pi^{\theta+\pi} &= \frac{1}{2} \eta^{-1/2} \rho^{\theta+\pi} \eta^{-1/2}, \end{aligned} \tag{5.10}$$

which satisfies the completeness condition:

$$\Pi^\theta + \Pi^{\theta+\pi} = \frac{1}{2} \eta^{-1/2} \rho^\theta \eta^{-1/2} + \frac{1}{2} \eta^{-1/2} \rho^{\theta+\pi} \eta^{-1/2} = I \tag{5.11}$$

The simulator forwards θ, operators $\{\Pi^\theta, \Pi^{\theta+\pi}\}$, as well as the first subsystem of η_{AB} to MRSP_B. MRSP_B performs the measurements by definition. It forwards the measurement result to Alice, and the simulator returns the second subsystem of η_{AB} to Bob. In the case of obtaining the classical description θ by Alice associated with Π^θ, the state η'_B forwarded to Bob is denoted as:

$$\begin{aligned} \eta'_B &= \Sigma_{k,k'} \sqrt{\lambda_k \lambda_{k'}} \langle\psi_{k'}|\Pi^\theta|\psi_k\rangle |\psi_k\rangle_B \langle\psi_{k'}| \\ &= \frac{1}{2} \Sigma_{k,k'} \sqrt{\lambda_k \lambda_{k'}} \langle\psi_{k'}| \Sigma_i \frac{1}{\sqrt{\lambda_i}} |\psi_i\rangle\langle\psi_i| \rho^\theta \Sigma_j \frac{1}{\sqrt{\lambda_j}} |\psi_j\rangle\langle\psi_j||\psi_k\rangle |\psi_k\rangle_B \langle\psi_{k'}| \\ &= \frac{1}{2} \Sigma_{k,k'} \langle\psi_{k'}|\rho^\theta|\psi_k\rangle |\psi_k\rangle_B \langle\psi_{k'}| \\ &= \frac{1}{2} \rho^\theta \end{aligned} \tag{5.12}$$

with probability $\frac{1}{2}$. Similarly, it outputs $\theta+\pi$ that associates with $\Pi^{\theta+\pi}$ on Alice' side, and $\rho^{\theta+\pi}$ on Bob's side with the same probability. As a result, the general joint outcome state of Alice and Bob of MRSP_B is exactly the same as RSP_B with $\varepsilon=0$. □

Since RSP_B can implement MRSP_B perfectly, and the UBQC protocol while Alice has access to MRSP_B also satisfies the security of perfect blindness. As described in Definition 5.4, we have the following statement:

Corollary 5.1. *See [221] The UBQC protocol where Alice, instead of access to a quantum channel and a random generator of $|+_\theta\rangle$ states, has access to the ideal resource RSP_B exactly constructs the ideal resource $\mathcal{S}^{blind}$ with perfect blindness.*

5.4.3 Perfect Blindness with RSR

In this section, we utilise similar proof techniques to demonstrate the security of the RSR resource and its composition with UBQC protocol for perfect blindness in two steps. First, in Lemma 5.1, we prove that the outcome of RSR satisfies the conditions for the blindness of Theorem 5.2. Then, in Theorem 5.5 and Corollary 5.2, we show the security of DQC with blindness obtained from RSR.

Lemma 5.1. *For any quantum states ρ_{in} that are used as input of RSR, the outcome system of Alice and Bob σ_{AB} satisfies the conditions of weak correlation of UBQC.*

Proof. On the one hand, we first assume that ρ_{in} is not entangled with Bob's auxiliary system. Without loss of generality, we get $\rho_{\mathrm{in}}=|\alpha|^2|0\rangle\langle 0|+\alpha\beta^*|0\rangle\langle 1|+\alpha^*\beta|1\rangle\langle 0|+|\beta|^2|1\rangle\langle 1|$. In this case, the output of RSR ρ^θ is

$$\rho^\theta=|\alpha|^2|0\rangle\langle 0|+\mathrm{e}^{-\mathrm{i}\theta}\alpha\beta^*|0\rangle\langle 1|+\mathrm{e}^{\mathrm{i}\theta}\alpha^*\beta|1\rangle\langle 0|+|\beta|^2|1\rangle\langle 1|. \tag{5.13}$$

For any θ in the set $\{0,\frac{\pi}{4},\ldots\frac{7\pi}{4}\}$, we thus have

$$\rho^\theta+\rho^{\theta+\pi}=2|\alpha|^2|0\rangle\langle 0|+2|\beta|^2|1\rangle\langle 1|. \tag{5.14}$$

Since this is independent of θ, the state satisfies the weak correlation conditions.

On the other hand, when ρ_{in} can be entangled with Bob's auxiliary system, we can thus write $\rho'_{\mathrm{in}}=|\alpha|^2|0\rangle\langle 0|\otimes|\psi_0\rangle\langle\psi_0|+\alpha\beta^*|0\rangle\langle 1|\otimes|\psi_0\rangle\langle\psi_1|+\alpha^*\beta|1\rangle\langle 0|\otimes|\psi_1\rangle\langle\psi_0|+|\beta|^2|1\rangle\langle 1|\otimes|\psi_1\rangle\langle\psi_1|$, where we decompose the qubit to be rotated into the orthogonal bases of $|0\rangle$ and $|1\rangle$, and $|\psi_0\rangle$ and $|\psi_1\rangle$ denote Bob's auxiliary system. Note that the coefficients α and β do not necessarily follow $|\alpha|^2+|\beta|^2=1$, since there are coefficients remaining in $|\psi_0\rangle$ and $|\psi_1\rangle$.

After the rotation of RSR on the first subsystem, we get the following entangled state:

$$\begin{aligned}\rho^\theta=|\alpha|^2|0\rangle\langle 0|\otimes|\psi_0\rangle\langle\psi_0|+\mathrm{e}^{-\mathrm{i}\theta}\alpha\beta^*|0\rangle\langle 1|\otimes|\psi_0\rangle\langle\psi_1|\\+\mathrm{e}^{\mathrm{i}\theta}\alpha^*\beta|1\rangle\langle 0|\otimes|\psi_1\rangle\langle\psi_0|+|\beta|^2|1\rangle\langle 1|\otimes|\psi_1\rangle\langle\psi_1|.\end{aligned} \tag{5.15}$$

For any θ in the set $\{0,\frac{\pi}{4},\ldots\frac{7\pi}{4}\}$, we have

$$\rho^\theta+\rho^{\theta+\pi}=2|\alpha|^2|0\rangle\langle 0|\otimes|\psi_0\rangle\langle\psi_0|+2|\beta|^2|1\rangle\langle 1|\otimes|\psi_1\rangle\langle\psi_1|. \tag{5.16}$$

Since the result $\rho^{\theta} + \rho^{\theta+\pi}$ is again independent of θ, the joint state of Alice and Bob also satisfies the weak correlation conditions for any state σ_{AB}. Such results also hold with arbitrary mixed states. □

We now prove the security of RSR with the UBQC protocol. We prove it by showing that the resource MRSP_B introduced in Definition 5.6 can be constructed from RSR. Since MRSP_B can be composed with a UBQC protocol to get DQC with perfect blindness, so does RSR

Theorem 5.5. *The protocol $\pi = (\pi_A, \pi_B)$ introduced above with ideal resource RSR constructs the ideal resource $MRSP_B$.*

Proof. We show that both the correctness and the security conditions are satisfied. More precisely, proving the security amounts to showing that a distinguisher cannot distinguish MRSP_B from the protocol. This translates into the following equations, for a simulator σ_B and the protocol $\pi = (\pi_A, \pi_B)$ with RSR.

$$\pi_A \text{RSR} \pi_B \approx_{\varepsilon} \text{MRSP}_B \perp, \tag{5.17}$$

and

$$\pi_A \text{RSR} \approx_{\varepsilon} \text{MRSP}_B \sigma_B. \tag{5.18}$$

For the correctness, when Bob is honest, the ideal resources RSR and MRSP_B both output an angle θ at interface A and its corresponding quantum state $|+_\theta\rangle\langle+_\theta|$ at interface B. Equation 5.17 is thus immediately satisfied.

For security, we introduce the simulator σ_B, defined as follows: It accepts and sends $c = 1$ to MRSP_B, as well as a set of operators $\{\Pi^{\theta}\}$, where $\Pi^{\theta} = |+_{-\theta}\rangle\langle+_{-\theta}|$. After receiving a quantum system ρ from Bob, the simulator takes the input ρ_{in} of the same dimension as Π^{θ} and generates a qubit $|0\rangle$. A CNOT gate is applied to these two qubits, where ρ_{in} is used as the control qubit ($|\phi_1\rangle$) and $|0\rangle$ the target bit ($|\phi_2\rangle$). This gives the simulator state ($\rho_{\sigma_B} = |\phi_{12}\rangle\langle\phi_{12}|$). Finally, σ_B sends the first qubit $|\phi_1\rangle$ back as the outcome state to Bob, whereas the second qubit, $|\phi_2\rangle$, is sent to the resource MRSP_B.

We show that the outcome is similar to the expression obtained in Lemma 5.1. Again, we start by considering the case that ρ_{in} is not entangled with Bob's auxiliary system. We then obtain the following expression for $|\phi'_{12}\rangle$ after the operation of MRSP_B:

$$\begin{aligned}
|\phi'_{12}\rangle &= \frac{\Pi_2^{\theta}}{\sqrt{\langle\phi_{12}|\,\Pi_2^{\theta}\,|\phi_{12}\rangle}}\,|\phi_{12}\rangle \\
&= \frac{1}{\sqrt{2}}(|0\rangle + \mathrm{e}^{-\mathrm{i}\theta}\,|1\rangle)(\langle 0| + \mathrm{e}^{\mathrm{i}\theta}\,\langle 1|)(\alpha\,|00\rangle + \beta\,|11\rangle) \\
&= \frac{1}{\sqrt{2}}(\alpha\,|00\rangle + \mathrm{e}^{-\mathrm{i}\theta}\alpha\,|01\rangle + \mathrm{e}^{\mathrm{i}\theta}\beta\,|10\rangle + \beta\,|11\rangle)
\end{aligned} \tag{5.19}$$

We obtain the outcome of the simulator by tracing out the second quantum subsystem.

$$\begin{aligned}
\rho_1 &= \mathrm{tr}_2(|\phi'_{12}\rangle\langle\phi'_{12}|) \\
&= |\alpha|^2\,|0\rangle\langle 0| + \mathrm{e}^{-\mathrm{i}\theta}\alpha\beta^*\,|0\rangle\langle 1| + \mathrm{e}^{\mathrm{i}\theta}\alpha^*\beta\,|1\rangle\langle 0| + |\beta|^2\,|1\rangle\langle 1|
\end{aligned} \tag{5.20}$$

The outcome quantum state is exactly the same result as the outcome of RSR in Eq.(5.13), with the angle θ that associates with Π^θ on Alice's side with probability $\frac{1}{2}$. Since a similar calculation holds for the operator $\Pi^{\theta+\pi}$, the general joint outcome state of Alice and Bob of MRSP_B is the same as RSR.

Similarly, we now consider the RSR with a qubit to be rotated, entangled with Bob's arbitrary auxiliary system. We denote it by decomposing the qubit to be rotated into the orthogonal bases of $|0\rangle$ and $|1\rangle$ as $\alpha |0\rangle |\psi_0\rangle + \beta |1\rangle |\psi_1\rangle$, where $|\psi_0\rangle$ and $|\psi_1\rangle$ denote Bob's auxiliary system. The simulator σ_B takes the first single-qubit subsystem as the control qubit and performs the same operation as in the previous case. After the operation of MRSP_B, we have:

$$\begin{aligned}
|\phi'_{12}\rangle &= \frac{\Pi_2^\theta}{\sqrt{\langle\phi_{12}| \Pi_2^\theta |\phi_{12}\rangle}} |\phi_{12}\rangle \\
&= \frac{1}{\sqrt{2}}(|0\rangle + \mathrm{e}^{-\mathrm{i}\theta} |1\rangle)(\langle 0| + \mathrm{e}^{\mathrm{i}\theta} \langle 1|)(\alpha |0\rangle |\psi_0\rangle |0\rangle + \beta |1\rangle |\psi_1\rangle |1\rangle) \\
&= \frac{1}{\sqrt{2}}(\alpha |0\rangle |\psi_0\rangle |0\rangle + \mathrm{e}^{-\mathrm{i}\theta}\alpha |0\rangle |\psi_0\rangle |1\rangle + \mathrm{e}^{\mathrm{i}\theta}\beta |1\rangle |\psi_1\rangle |0\rangle + \beta |1\rangle |\psi_1\rangle |1\rangle)
\end{aligned} \tag{5.21}$$

Then, after tracing out the second qubit, we obtain:

$$\begin{aligned}
\rho_1 =& \mathrm{tr}_2(|\phi'_{12}\rangle \langle\phi'_{12}|) \\
=& |\alpha|^2 |0\rangle \langle 0| \otimes |\psi_0\rangle \langle\psi_0| + \mathrm{e}^{-\mathrm{i}\theta}\alpha\beta^* |0\rangle \langle 1| \otimes |\psi_0\rangle \langle\psi_1| \\
&+ \mathrm{e}^{\mathrm{i}\theta}\alpha^*\beta |1\rangle \langle 0| \otimes |\psi_1\rangle \langle\psi_0| + |\beta|^2 |1\rangle \langle 1| \otimes |\psi_1\rangle \langle\psi_1| .
\end{aligned} \tag{5.22}$$

Again, The output quantum state is exactly equal to ρ^θ specified in Eq.(5.15). In consequence, the resource RSR is perfectly indistinguishable from the resource MRSP_B, that is, Equations 5.17 and 5.18 are satisfied with $\varepsilon = 0$. Such results also hold with arbitrary mixed states.

□

Finally, combining the fact that we can perfectly construct MRSP_B from RSR with Theorem 5.3, we obtain the following result, as described in Definition 5.5, with $\varepsilon = 0$.

Corollary 5.2. *The UBQC protocol with the client accessing the RSR constructs the ideal functionality of DQC with perfect blindness.*

5.5 QEnclave-based Secure Quantum Cloud Computing

From the previous proofs, we know that it is sufficient for a remote Alice as the role of **Client** to perform DQC securely with perfect blindness with a potentially malicious Bob as the role of **Server** while accessing the resource RSR instead of the resource of a quantum channel. However, to make the client purely classical, it is only true if RSR is on the server's side. Here, we do not consider the scenario that a trusted third party performs RSR. Meanwhile, the operations of RSR should

be guaranteed with integrity and confidentiality to the server. However, we haven't yet clarified how to implement RSR with existing near-term resources in practice. Theoretically, if the resource can be constructed only classically with no hardware assumption, it should follow the definition below:

Definition 5.7. *(See [223]) An ideal resource $\mathcal{S}$ is said to be ε-classical-realizable if it is realizable from a resource of classical channel $\mathcal{R}_{c\_channel}$, i.e. if there exists a protocol $\pi = (\pi_A, \pi_B)$ between two parties interacting classically such that:*

$$\mathcal{R}_{c\_channel} \xrightarrow{\pi,\varepsilon} \mathcal{S} \tag{5.23}$$

In order to prove the composable security of ε-classical-realizable RSP, we need to show that no unbounded adversary can learn information on θ by accessing only the right interface B. Unfortunately, the authors of [223] show that there is no *describable* remote state preparation protocol with composable security. In the context, describable means extracting a classical approximate description of a quantum state $[\rho]$ by accessing the state ρ on the interface B. Since a protocol using only classical communication is describable, there is no classical-realizable RSP with composable security. It further implies that UBQC protocol with classical-realizable RSP cannot be composably secure. As a result, it is necessary to make additional assumptions to remove the quantum interaction between the client and the server. While [223] considers additional computational assumptions to bound the adversary's power, we take a different approach, introducing additional hardware assumptions such as tamper-proof quantum operations to get a secure DQC protocol with blindness using only classical communication.

As a result, we propose the design of QEnclave (See Figure 5.2), which is inspired by the hardware assumption of TEE in the classical world, with limited protection of RSR quantum devices and the flows in between. With the security guaranteed by construction, we further specify an enclave-based program in quantum analogue with the characteristics of attested execution functionality G_{att}. It performs RSR functionality within QEnclave, where the classical secret remains blind to a QPT malicious server. In the following sections, we evaluate the security of QEnclave-based blind quantum cloud in the real world and discuss further the security of verifiable quantum cloud computing with QEnclave-based RSR.

5.5.1 QEnclave-based Blind Quantum Cloud Computing

We start by recalling a simple 2-party outsourcing computation $\mathcal{F}_{\text{outsrc}}[C, S]$ with a target function $y = f(x)$, where the client C outsources f and x with encoding and finally obtains the output y while the server S or any other adversary only knows the size of inputs and outputs $(|f| + |x|, |y|)$ during the computation process.

In the classical case, Pass et al. [168] show that G_{att} setup assumption can be used to compute a public function $\mathcal{F}_{\text{outsrc}}$ jointly with an honest client and malicious server with composable security, while the server owns a secure enclave as a trusted hardware platform and initializes an enclave program running $\mathcal{F}_{\text{outsrc}}$. They also give the protocol between a client C and a server S to realise $\mathcal{F}_{\text{outsrc}}$ with rigorous proofs.

As to the $\mathcal{F}_{\text{outsrc}}$ of QEnclave, we specify an efficient classical function $f^{\theta}(\cdot)$ from the client with input x, and together generate the classical description of rotation angles $\theta_1 \dots \theta_N$ uniformly at random from the set Θ, and N is the public information according to the size of targeted quantum computation. For each rotation angle θ_i, QEnclave takes an initial state from the server, performs rotation, and returns it to the server.

Functionality 3 The ideal functionality $\mathcal{F}_{\text{outsrc}}$ of QEnclave

Input of Client:

- The client C sends the target function $f^{\theta}(\cdot)$ with input x.

Input of Server:

For $i = 1, \dots, N$:

- The server S sends an initial state $|+\rangle_i\langle+|$, if he is honest. Otherwise, he deviates by preparing an arbitrary state ρ_{in}^{i} as input.

Execution of $\mathcal{F}_{\text{outsrc}}$:

- On receiving $f^{\theta}(\cdot)$ with input x from C, it computes

$$\theta_1 \dots \theta_N = f^{\theta}(x).$$

- For $i = 1, \dots, N$, it performs a single-qubit rotation $Z(\theta_i)$ upon the initial state from S, and keeps it with S.
- It sends $(|f| + |x|, |\theta_i|_{i \in N})$ to S.

It is not hard to observe that the description of $\mathcal{F}_{\text{outsrc}}$ is almost the same as the RSR resource, except that the client can produce rotation angles instead of those generated uniformly at random by the resource individually. Note that the transformation does not change the security property in this case since the client is expected to be honest, and QEnclave should follow its instructions correctly by specification.

In this case, we give in Protocol 2 the specification of G_{att}-hybrid protocol $\texttt{Prot}_{\text{outsrc}}$ that enables outsourcing of RSR functionality on the server's side by QEnclave based on the Functionality 2. Compared to the proof in [168] that the PPT indistinguishability of ideal-world and real-world executions is reduced to the decisional Diffie-Hellman (DDH) assumption for secure key exchange [225] and authenticated encryption. Here, to obtain the composable security result against a QPT adversarial server, Most of the proofs follow similarly with the assumption of a post-quantum secure key exchange between C and G_{att}. In practice, as long as the key exchange scheme is post-quantum secure, G_{att}-based RSR is feasible in terms of security.

Furthermore, a quantum-safe digital signature scheme Σ [226, 227] is necessary for the remote attestation scheme since we assume the server is potentially

Protocol 2 QEnclave-based RSP Protocol for Blindness with $\texttt{prog}_{\text{rsr}}$

G_{att}-enabled QEnclave Program $\texttt{prog}_{\text{rsr}}$:

On input ("keyex", p_k) :

- let $(k, c_k) = \text{KEM.}\texttt{Enc}(p_k)$, seal k, and return c_k.

On input* ("compute", ct):

- let $(f^\theta, x) := \text{AES.}\texttt{Dec}(k, ct)$.
- assert decryption success, ct not seen before.
- let $\theta_1 \dots \theta_N := f^\theta(x)$, and $\theta_1 \dots \theta_N$ are classical descriptions of rotation angles that are applied to quantum states from the external source of S. For each instance $i = 1, \dots, N$, it outputs:

$$\rho^i_{\text{out}} = \begin{cases} |+_{\theta_i}\rangle\langle+_{\theta_i}| & \text{If } S \text{ is honest} \\ Z(\theta_i)\rho^i_{\text{in}}Z^\dagger(\theta_i) & \text{Otherwise} \end{cases}$$

Server S:

On receive ("keyex", p_k) from C:

- let eid := $G_{\text{att}}.\texttt{install}(\text{sid}, \texttt{prog}_{\text{rsr}})$.
- let $(c_k, \sigma) := G_{\text{att}}.\texttt{resume}(\text{eid}, (\text{"keyex"}, p_k))$, and send $(\text{eid}, c_k, \sigma)$ to C.

On receive* ("compute", ct) from C:

- let $\rho_{\text{rsr}} = G_{\text{att}}.\texttt{resume}(\text{eid}, (\text{"compute"}, ct))$, and keep them for further computation.

Client C:

On initialize:

- let $(p_k, s_k) \leftarrow \text{KEM.}\texttt{KeyGen}(1^\lambda)$, $\texttt{mpk} := G_{\text{att}}.\texttt{getpk}()$.
- send ("keyex", p_k) to S, await $(\text{eid}, c_k, \sigma)$ from S.
- assert $\Sigma.\text{Vf}_{\texttt{mpk}}((\text{sid}, \text{eid}, \texttt{prog}_{\text{rsr}}, c_k), \sigma)$.
- let $k = \text{KEM.}\texttt{Dec}(s_k, c_k)$.

On receive* ("compute", f^θ, x):

- let $ct := \text{AES.}\texttt{Enc}(k, f^\theta, x)$, and send ("compute", ct) to S.

malicious. Meanwhile, more practical remote attestation schemes provide post-quantum security [228].

The confidentiality consists of hiding the rotation angles chosen by the client. The requirement of using quantum-safe encryption makes symmetric schemes more appropriate than asymmetric ones for this task. Instead of a key exchange protocol based on DDH, there are other key encapsulation mechanism schemes (KEM) [229, 230, 231] available to share a secret key between the client and the QEnclave and proven to be secure against a QPT adversary for now.

Once the secure channel is established between the client and QEnclave, the client can send the encrypted rotation angles to the QEnclave. QEnclave decrypts them and encodes the initial quantum state from the external source using the classical angles chosen by the client. At this stage, we assume that the trusted area includes the enclave and minimal quantum devices. It leads to a remote state preparation protocol for delegated quantum computation with blindness using the QEnclave and classical communication between the client and the server. We summarise all the steps in Protocol 2 with a post-quantum secure digital signature scheme Σ with signature σ, and a post-quantum secure key encapsulation mechanism KEM scheme with the key derivation function for generating a symmetric key to establish a secure channel between the client and QEnclave. Here, we instantiate it as AES for encryption and decryption[1]. Finally, we are ready to give a formal statement of the security of G_{att}-hybrid protocol $\texttt{Prot}_{\text{outsrc}}$ that UC-realises the ideal functionality $\mathcal{F}_{\text{outsrc}}$ of QEnclave with the target function $f^{\theta}(x)$:

Theorem 5.6. *Assume that the signature scheme Σ is existentially unforgeable under chosen message attacks, and the security of the KEM scheme holds against a QPT adversary. The G_{att}-hybrid protocol $\texttt{Prot}_{outsrc}$ with QEnclave UC-realises $\mathcal{F}_{outsrc}$, when the client C is honest, and the server S is malicious by trying to extract the classical description of rotation angles θs.*

Proof. With an honest client and corrupted server, a simulator `Sim` in the ideal world that makes no QPT indistinguishability of ideal-world and real-world executions can be described as follows:

- Unless we note specifically, the simulator `Sim` always forwards any communication between G_{att} and adversary (the corrupted S) or between C and S.
- `Sim` starts by emulating the setup of a secure channel between C and G_{att}. `Sim` sends ("keyex", p_k) to S.
- When `Sim` receives a tuple (eid, c_k, σ), `Sim` aborts outputting `sig-failure` if the digital signature σ would be validated by a honest C, while `Sim` has not recorded the following communication between G_{att} and S:
  - eid $:= G_{\text{att}}.\texttt{install}(\text{sid}, \texttt{prog}_{\text{rsr}})$;
  - $(c_k, \sigma) := G_{\text{att}}.\texttt{resume}(\text{eid}, (\text{"keyex"}, p_k))$

  Else, `Sim` computes $k = \text{KEM}.Dec(s_k, c_k)$.

[1]For example, AES-GCM provides authenticated encryption with INT-CTXT

- When `Sim` receives a message $(|f| + |x|, |\theta_i|_{i \in N})$ from $\mathcal{F}_{\text{outsrc}}$, it proceeds as follows: `Sim` sends ("compute", ct) = AES.$\texttt{Enc}(k, f_0^\theta, x_0)$) to S where f_0 and x_0 are some canonical function and input with the same fixed size as f and x.

Recall that for UC realisation, the simulator should follow

$$\text{IDEAL}_{\mathcal{F},\texttt{Sim},\mathcal{Z}}(\lambda) \approx_\varepsilon \text{EXEC}_{\pi,\mathcal{A},\mathcal{Z}}(\lambda) \tag{5.24}$$

with negligible ε by environment $\mathcal{Z}$. The indistinguishability of the ideal-world and real-world execution can be proven within multiple steps of hybrids:
Claim: Assume that the signature scheme Σ is secure, except with negligible probability, the simulated execution does not abort outputting `sig-failure`.

Proof. Straightforward reduction to the security of the digital signature scheme Σ. □

Hybrid 1. *Identical to the simulated execution, but the secret key $k = KEM.Dec(s_k, c_k)$ shared between C and G_{att} is replaced with a random key from the appropriate domain.*

Claim: Assume that the security of the KEM scheme holds, then Hybrid 1 is computationally indistinguishable from the simulated execution.

Proof. Straightforward by the reduction to the security of the KEM scheme. □

Hybrid 2. *Instead of sending $ct = AES.\texttt{Enc}(k, f_0, x_0)$ to S, the simulator now sends $ct = AES.\texttt{Enc}(k, f, x)$ where f and x are the honest client's true inputs.*

Claim: Hybrid 2 is computationally indistinguishable from Hybrid 1.

Proof. Straightforward reduction to the weak correlations and blindness proof of RSR resource in previous, and security of AES-GCM with INT-CTXT. □

Hybrid 3. *Now, using real key k instead of using a random key between C and G_{att}.*

Claim: Assume that the security of KEM holds, then Hybrid 3 is computationally indistinguishable from Hybrid 2.

Proof. Straightforward by the reduction to the security of the KEM scheme. □

Finally, observe that conditioned on the simulator not aborting and AES encryption/decryption being perfectly correct, Hybrid 3 is identically distributed as the real execution. □

For the assumption that the communication between the enclave and the quantum device is protected against a tampering server in terms of confidentiality. Although it may seem strong, the idea of sealing hardware components into a tamper-proof box is already widespread in the world of hardware security. In particular, the FIPS-140 certification for hardware security modules (HSM) includes criteria for physical tamper-evidence (level 2 certification), physical tamper-resistance (level 3), or even robustness against environmental attacks (level 4).

With Protocol 2 based on QEnclave and UBQC protocol, we show that a client can delegate a perfect blind quantum computation on an adversarial quantum server with only a classical channel in between. Meanwhile, we note that it is sufficient if the adversarial server is honest but curious in the protocol. However, recalling the security in terms of verifiability means that the client can verify the operations of the server and accept incorrect results with negligible probability. It is not trivial to obtain from QEnclave. In the next section, we show the related results regarding this perspective.

5.5.2 QEnclave-based Verifiable Blind Quantum Cloud Computing: Limitations and Possibilities

For verification, a resource for remote state preparation with verification (RSP_V) is proposed in [222]. In their work, the authors show that when the client has access to the resource RSP_V, and a verifiable UBQC protocol so-called FK protocol [232] without a quantum channel constructs an ideal resource $\mathcal{S}_{\text{verif}}^{\text{blind}}$ as the blind and verifiable DQC as:

$$\mathrm{RSP}_V, \mathcal{R}_{\text{c\_channel}} \xrightarrow{\pi,\epsilon} \mathcal{S}_{\text{verif}}^{\text{blind}}, \tag{5.25}$$

with a formal definition of $\mathcal{S}_{\text{verif}}^{\text{blind}}$ in the following:

Definition 5.8. (See Figure 5.6) *For a given unitary U, the ideal resource DQC resource $\mathcal{S}_{verif}^{blind}$ provides correctness, blindness and verifiability. It takes an input ψ_A at Alice's interface, and a filtered control bits c (set by default to 0) at Bob's interface. It outputs the allowed leak ℓ^{ψ_A} at Bob's interface. If $c = 0$, it simply outputs $U(\psi_A)$ at Alice's interface. If $c = 1$, it outputs an error message at Alice's interface.*

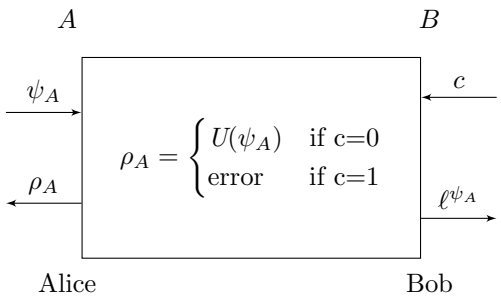


Figure 5.6: **DQC ideal resource with verifiability.** The filtered control bit $c = 0/1$ denotes the honest/dishonest behaviour of Bob. The filtered functionalities with the input ψ_B, and the output ℓ^{ψ_A} of resource are accessible only to the dishonest Bob with $c = 1$.

The verifiability of [232] comes from the specification that Alice could insert in the target computation a set of *trap qubits* that are isolated from the computation, while Bob has no way to distinguish these trap qubits from the rest. As long as the deviations of Bob are independent of the hidden nature of qubits, this construction ensures that the measurement results of trap qubits are always deterministic and known by Alice in advance, while Bob is honest and can be used as a test of the

correctness of the entire computation while Bob is malicious as the probability that Bob deviates the computation without deviating trap qubits is negligible in the security parameter when the positions of trap qubits are only known by Alice.

As for the resource of remote state preparation that constructs $\mathcal{S}_{\text{verif}}^{\text{blind}}$ from the resource $\mathcal{R}_{\text{c\_channel}}$, it requires the resource to prepare correctly single-qubit states in the set of $|+_\theta\rangle$ for $\theta \in \{0, \frac{\pi}{4}, \ldots \frac{7\pi}{4}\}$, and $|b\rangle$ for $b \in \{0, 1\}$. Here, the trap qubits are initialised as a $|+_\theta\rangle$ state. While the qubits are entangled in a graph state by Bob for computations, all neighbours of the trap qubit in the underlying graph state are initialized in a random $|b\rangle$ state, called the *dummy qubits*. In FK protocol, the dummy qubits isolate the trap qubits from the computation, and Alice's measurement of trap qubits can be verified independently of the computations. That is what RSP_V is specified in [222]. Furthermore, they propose a protocol called the so-called *buffered remote state preparation* (BRSP) protocol in the real world to construct RSP_V, and furthermore implement FK protocol with composable security in replacing the resource of a quantum channel.

On the one hand, as we discussed previously, the resource RSP_V is incompatible with classical communications with no additional assumption. That is to say, the resource RSP_V can not be constructed by only the resource of a classical channel. In brief, BRSP protocol specifies a classical channel between Alice and Bob but assumes a resource called *measurement buffer*, which externalizes a quantum state measurement from Bob's side. Here, the buffer interacts classically with Alice and Bob in quantum and behaves honestly throughout the protocol while assuming Bob is the only malicious party. This, in fact, can be treated as a different approach to defining a quantum TEE that securely implements the measurement buffer (on Bob's side) and then uses BRSP protocol. However, we believe it is desired that the hardware assumption should be as simple as possible and securing the measuring device leads to an unnecessarily complicated architecture.

By considering to construct $\mathcal{S}_{\text{verif}}^{\text{blind}}$ by RSR-like functionality and the FK protocol, to maintain the construction of QEnclave as simple as possible in practice, we consider an extension of the functionality of RSR by enabling the resource preparing single-qubit states $|b\rangle$ for $b \in \{0, 1\}$. In the scope of QEnclave, the hardware primitive guarantees that the operations for the preparation of either dummy qubits or computational/trap qubits are indistinguishable from Bob's operation since it is obvious that each prepared quantum state is a maximally mixed state from the perspective to Bob. However, adapting the same approach for RSR is not trivial, as a malicious Bob controlling the source is now enabled to perform correlated attacks before and after the call to RSR. Hence, the proof technique from [232] does not directly apply. For example, Bob maliciously deviates by performing Pauli Z rotations before and after RSR each instance without knowing what type of qubit he is preparing. For computational and trap qubits, these deviations cancel each other due to the commutation of $Z(\theta)$ and Z operators. For the dummy qubits, it causes a flip of dummy qubits from $|0\rangle$ to $|1\rangle$ and from $|1\rangle$ to $|0\rangle$. Such deviations require no secret computation information by Bob, but these deviations do affect differently depending on the types of qubits. This leads to the fact that for trap qubits with an even number of deviated neighbouring dummies qubits, the measurement results of trap qubits remain deterministic with no deviations, while the computational qubits with an odd number of deviated neighbouring dummies

qubits are maliciously changed without notice. For example, [233] introduces a verification protocol as a variant of the FK scenario. Such deviations above can leave trap qubits unchanged by measurement but affect the computational qubits via flipping adjacent dummy qubits with odd numbers. It changes the execution of the protocol apparently but remains unnoticed by Alice at the same time so that the protocol is no longer verifiable in this case.

On the other hand, a recent work of [234] proposes a blind and verifiable protocol requiring the preparation of only $|+_\theta\rangle$ states. The main idea is to find a generating set of stabilisers of the graph state for the delegated computation on Bob's side that can be written with Pauli I, X and Y only. In this case, the authors show that the deviations by Bob as the server in the protocol can either be detected, or these deviations won't have any effect on the outcome of the computation. In this case, verification can be achieved without preparing dummy qubits but only X-Y plane states, which is compatible perfectly with QEnclave-based remote state rotation functionality. We leave the implementation details and further problem exploration as future work.

5.6 Discussion

In this chapter, we explore another hardware security primitive – TEE, which is widely used in classical outsourcing cloud computing scenarios to securely isolate partially sensitive data and applications in practice in the quantum analogue. Recall that in the adversarial model, we consider that the server can be potentially malicious. We introduce the notion of security against an adversarial server with respect to blindness and verifiability. To guarantee the security of the functionality minimally in delegated quantum computations by means of hardware security, we study an abstraction of the functionality called remote state rotation, performed by QEnclave. From the functionality perspective, it only performs single-qubit rotations with external input and outputs quantum states for computation after rotations. We show that it is sufficient to achieve perfect blindness of computations by RSR in composable security jointly by an honest classical client and a malicious quantum server. Our proposed functionality with simple rotations lowers the minimal requirement on the client's operations while keeping minimal overhead on the server's side. Furthermore, we specify QEnclave construction with an enclave and isolated quantum apparatus for achieving RSR functionality in practice. We then present a real-world specification of QEnclave-based $\texttt{prog}_{\text{rsr}}$ protocol for blind quantum computation and show the composable security in the context of attested execution functionality, which captures the characteristics of programs executed by an enclave. As a realistic hardware assumption of trustworthy quantum operations with classical secrets, QEnclave circumvents the impossibility results of [217, 223] of implementing a composable RSR with a classical channel only. Besides privacy, we explore another security requirement for verifiability. On the one hand, we show that a malicious server with the capability of controlling the source and performing correlated attacks before and after the call to QEnclave is very powerful, and the proof technique from [232] does not directly apply. On the other hand, inserting traps without dummy qubits but a set of stabilisers of graph state, which

requires quantum states with only X-Y plane rotations, can again be composed with QEnclave to achieve verifiability potentially.

As to the integration of the QEnclave in current quantum computing technology, our current QEnclave only implements a single qubit rotation, and it interacts with the server's quantum computer while residing at the server's computing facility. In this case, QEnclave always requires a quantum communication channel to interact with the source and the server's quantum computer. The linear optics-based photonics platforms are efficient for both quantum communication and single qubit rotation [235]. Hence, we predict that the photonics-based platform would be ideal for implementing the QEnclave. Such QEnclaves would fit perfectly with photonics-based quantum computing techniques.

However, for the other kinds of quantum computing techniques, like ion traps-based processors or superconducting-based qubits, we need to use an external interface for the interaction between the server and the QEnclave. Note that a promising approach to scaling ion-trap quantum computers to arbitrarily large numbers of qubits is to use many similar ion-trap processors (nodes) connected together in a modular network. Such a quantum network can produce ion-photon entanglement [236]. A potential solution for designing the interface between QEnclave and ion trap-based quantum computers would be to use such an ion-photon entanglement to teleport the outcome of the QEnclave to the ion qubits. The detailed description of such an interface is beyond the scope of this paper. We leave this interesting study for our future work.

Meanwhile, we believe that other quantum communication and computation applications can exploit QEnclave rather than UBQC-like protocols. In general, the concept of QEnclave can be used for any client-server-based protocol [237] with a quantum communication channel. First of all, QEnclave can be exploited in the prepare-and-send universal blind quantum computation with multiple clients to replace the multiple quantum communication channels from a server to the clients [238, 239, 240] for scalability. In this protocol, the security of blindness that the server does not learn the delegated computation and its input/output is guaranteed against either a dishonest server or a coalition of dishonest clients. In the case of the dishonest server, the protocol is equivalent to thinking of all honest clients as one with multiple input qubits. The blindness of DQC with QEnclave can be obtained intuitively.

In the case of the coalition of dishonest clients, the quantum channels among clients are replaced by RSP by QEnclave on the server's side. Meanwhile, since the clients are assumed to have secure access to verifiable secret sharing (VSS) in the protocol as a classical multiparty computation protocol, by committing classically every round of rotation angles during the RSP stage via VSS, the correctness of committed values can be verified by the honest server and the rest of honest clients. Note that the restriction to performing multiparty quantum computation for blindness with this protocol is that the collusion of a dishonest server and clients is impossible.

Secondly, in terms of quantum homomorphic encryption (QHE), which is formalized by [241], it permits an evaluation of quantum circuits on encrypted quantum data in DQC setting. Furthermore, a protocol of prepare-and-send QHE is proposed in [242] with quantum communication between client and server. Unlike

UBQC, the quantum circuit is not hidden from the server, but the client can verify the computation by the decryption of the output. However, the composition of QEnclave and QHE is tricky since QHE requires the encryption and decryption of quantum data with confidentiality, integrity, and a trusted quantum source. Alternatively, one can put the encryption, evaluation, and decryption circuit fully inside the QEnclave. With such a powerful assumption, any classical client can run a secure QHE protocol using just classical communication. However, our primary goal here is to reduce the assumptions on the QEnclave functionality, i.e., we try to make the quantum circuit inside the QEnclave as simple as possible. For example, the QEnclave contains only a single qubit rotation gate in our current setup. Making the QEnclave circuit simple for the QHE, without loosing the security is challenging and beyond the scope of this paper. However, this is an interesting direction for our future research.

Finally, we think QEnclave can be relevant to be used in quantum money schemes [243], especially the protocol [244] considers that the bank mints the quantum states used as banknotes on the user's side and verifies their validity using only classical interactions. It matches our definition of remote state preparation once the problem of verifiability is also addressed. Then, by using a QEnclave, a bank might be able to authenticate the banknote by remotely performing quantum operations but using only classical communication.

Chapter6

Conclusion

6.1 Summary

WITH the importance of hardware security in classical modern computing systems in mind, we start the thesis by asking two general questions regarding the use cases of hardware security while considering the coming applicable quantum computation and quantum information technologies. Recall that these two questions are:

> *"How can quantum mechanical systems enhance the security of existing hardware security primitives?"*
>
> , and
>
> *"How can hardware security primitives protect quantum computing systems?"*

To answer these two questions, we investigate different types of hardware secure primitives in Chapter 3, which are figured as hardware-based fundamental components and mechanisms used to provide building blocks for implementing security features, safeguarding against threats, and ensuring confidentiality, integrity, and availability of information and resources. A type of hardware security primitives that are shown to be versatile and sophisticated for practical implementation is the Quantum Random Number Generators (QRNGs). Meanwhile, QRNGs can fit in both these two questions we asked. On the one hand, quantum mechanics and quantum information theory provide unique and solid theoretical support to guarantee the high entropic quantum sources for true randomness. On the other hand, quantum computing systems that require remote access can exploit QRNGs to generate secret keys for authentication or other cryptographic protocols. Therefore, it is also worth investigating other types of hardware security primitives within quantum technologies while the overhead of implementation in practice remains rational, especially the overhead of quantum resources.

In Chapter 4, we move a step forward in learning the hardware security primitive, Physical Unclonable Function (PUF), in the context of quantum information techniques and try to answer the first question. Recall that the scenarios of using PUFs with classical input/output are limited due to the copy-free of classical information. Meanwhile, constructing PUFs with purely quantum resources (QPUFs)

is overhead with current quantum information techniques. It is natural to think about the constructions of PUFs with a hybrid classical-quantum setting.

While considering the classical readout of evolution by quantum circuits is vulnerable and totally losing quantum advantages, we propose the construction of PUFs with underlying classical PUFs and quantum information encoding techniques that have already been widely implemented in experiments and industries. They form together as the construction of Hybrid Locked PUFs (HLPUFs). In the game-based security model framework, we rigorously prove the unforgeability of HLPUFs against any adaptive adversaries in the authentication protocols, with the support of simulation results. Meanwhile, another amazing property obtained from quantum information techniques is reusability, which inherits from the unclonability property of quantum information and privileges a lot of the service life of PUFs. Finally, our proposal is not limited to some concrete construction but is more like a general guideline due to structural compatibility.

In Chapter 5, we explore the second question by looking into another hardware security primitive, Trusted Execution Environment (TEE), and its role in quantum computation scenarios. Our motivation comes from the fact that future quantum computing devices will most probably be hosted by cloud servers and accessed remotely. Nowadays, accessible quantum cloud computations use cases have no security guarantees, e.g., privacy, verifiability, etc., especially in cases when the server might be malicious by definition. There are theoretical algorithms or protocols for achieving delegated quantum computation with satisfying different security properties. However, these protocols either require quantum channels between clients, or a huge overhead on quantum resources to achieve security guarantees.

Inspired by a broad class of secure enclaves, which instantiate TEEs in computing systems, being used in classical cloud computation for privacy-preserving, we proposed a quantum analogue of TEEs called QEnclave to secure the quantum cloud computations. With the minimal trust assumption by TEEs in mind, we show that a hybrid classical-quantum setting is sufficient and efficient for achieving privacy on cloud quantum computations, where QEnclave creates a secure environment based on the hardware of a secure enclave and its control of only single-qubit rotations as the quantum extension. With the help of simulation-based proofs and composition frameworks, we show that QEnclave, together with UBQC protocol, guarantees the privacy of cloud quantum computation with reasonable overhead regarding operational complexity. Furthermore, with an adaption of the protocol, QEnclave can potentially achieve another security property, so-called verifiability, but rigorous proofs should still be done in the future. As a result, these results show a rational and interesting research perspective that exploits TEE mechanism in future quantum computing devices.

6.2 Future Works

By the end of each chapter, we explore possible future directions for each hardware security primitive based on our work. Furthermore, another interesting yet challenging perspective to explore is the possible composition of different hardware security primitives with their security features to achieve applications/protocols

that are more efficient or robust. Here, we give some prototypes that we will explore in the future:

In the proposal of HPUF, and further HLPUF constructions. The main motivation for the quantum lockdown mechanism is to reduce the power of a network adversary from querying adaptively to non-adaptively. And it permits 2-way authentication for both server and client in the authentication protocol. However, the overhead of quantum resources is in two folds: On the one hand, both the server and the client in the protocol must be capable of preparing and measuring quantum states (including qubits or higher dimension quantum states). On the other hand, the quantum communication in between is necessarily a two-way quantum channel. Both of these limitations might, to a large extent, limit the usage scenarios of HLPUFs.

Nevertheless, there are also other methods to secure HPUF against an adaptive adversary with less overhead in applications. Recall in Section 2.2.3, we discuss the information-theoretic security of OTP scheme against an unbounded adversary, yet the issues of generating truly random keys and securely distributing them to both the sender and the receiver make it costly and complicated. We propose the possibility of combining two hardware security primitives, HRNG and PUF, and show that the composition can be quite appropriate and efficient. In this case, as the receiver with the PUF receives a classical challenge, the corresponding response is split into two binary strings equally. For the first part of the response, a true random binary string from HRNG with the same length can pad it via XOR bitwise. Meanwhile, the pad is encoded into BB84 states as the HPUF construction. Then, the receiver takes every tuple of two elements (a, b) successively from the second part of the response and computes each single-qubit mixed state with X^a and Z^b operators controlled by a and b. After the operations above are done, the receiver can return the joint classical-quantum states to the sender. The authentic sender can verify the equality of the random binary string by taking the corresponding response from its database to decode the classical and quantum ciphertext, respectively.

Here, we argue that the security guarantees of the new setup come from two folds: On the one hand, the classical OTP scheme is information-theoretic secure against an unbound network adversary. On the other hand, the quantum encoding and padding methodology can be figured as a quantum OTP scheme [245], where each state results in a maximally mixed state and leaks no information to an unbounded adversary. Note that the properties of challenge reusability may no longer be applied here due to the security requirements. Since this, we think the new setup is very likely to be used to derive other cryptographic protocols, e.g., key distribution. From the quantum resource perspective, it is obvious that the current setup no longer requires the measurement devices on the receiver's side, and the sender does not need to prepare quantum states either. And the quantum channel is also a one-way from the receiver to the sender. Even though security analysis and proof should be further done rigorously, we conjecture that this direction can have many possibilities, especially regarding simulation and implementation.

Furthermore, our development of QEnclave is still in an early stage. In our work, our discussion is limited to as simple operations within QEnclave as possible due to the potential increase of complexity on the hardware level, especially quantum

devices. As TEE is a hardware-based mechanism to provide an isolated execution environment for crucial data and applications in the classical case, it is rational to think of integrating different hardware security primitives together in QEnclave. These hardware security primitives can be in classical or quantum. For example, a HRNG can be integrated within QEnclave to generate random angles for rotation. However, one of the main difficulties of utilising QEnclave with more functionalities and border applications is that they require the design engineer to identify the vulnerability on the hardware level manually. A very recent work [246] proposes a solution to construct a superconducting quantum computer trusted execution environment by decoy pulse approach. The main idea is to obfuscate analogue control pulses to quantum computers, together with tamper-resistant and trusted hardware components that attenuate the decoy pulses inside the superconducting quantum computer's dilution refrigerator before arriving at the qubits. In this case, the possibly malicious quantum computer cloud providers cannot learn the controlling signal of quantum operations on qubits. From the engineering point of view, it effectively tackles one main vulnerability on the hardware level yet raises another potential issue about the necessary range of protection, no matter the perspectives of functionality or actual hardware. Meanwhile, a sophisticated quantum computing system's general hardware architecture is still unclear. These pose a significant challenge in larger designs and coordinate with other hardware security primitives. Nevertheless, considering the performance and limitations of near-term quantum devices, we think that a quantum analogue of TEE, in general, should still follow the postulates we introduced previously and maintain its functionality with a minimal trust assumption.

6.3 The Ending Word

"不积跬步，无以至千里；不积小流，无以成江海"

Bibliography

[1] J. Preskill; "Quantum computing 40 years later;" arXiv preprint arXiv:2106.10522 (2023); URL http://dx.doi.org/10.48550/arXiv.2106.10522. (page 1)

[2] A. M. Turing; "On Computable Numbers, with an Application to the Entscheidungsproblem;" Proceedings of the London Mathematical Society **s2-42**, pp. 230–265 (1937); URL http://dx.doi.org/10.1112/plms/s2-42.1.230. (pages 1 and 16)

[3] P. Benioff; "The computer as a physical system: A microscopic quantum mechanical Hamiltonian model of computers as represented by Turing machines;" Journal of Statistical Physics **22**, pp. 563–591 (1980); URL http://dx.doi.org/10.1007/BF01011339. (pages 1 and 16)

[4] P. Benioff; "Quantum mechanical Hamiltonian models of Turing machines;" Journal of Statistical Physics **29**, pp. 515–546 (1982); URL http://dx.doi.org/10.1007/BF01342185. (pages 1 and 16)

[5] D. Deutsch; "Quantum theory, the Church—Turing principle and the universal quantum computer;" Proceedings of the Royal Society of London. A. Mathematical and Physical Sciences **400**, pp. 97–117 (1985); URL http://dx.doi.org/10.1098/rspa.1985.0070. (pages 1 and 16)

[6] P. Shor; "Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer;" SIAM Journal on Computing **26**, pp. 1484–1509 (1997); URL http://dx.doi.org/10.1137/S0097539795293172. (pages 1, 24, and 97)

[7] L. K. Grover; "A fast quantum mechanical algorithm for database search;" arXiv preprint arXiv:quant-ph/9605043 (1996); URL http://dx.doi.org/10.48550/arXiv.quant-ph/9605043. (page 1)

[8] H.-S. Zhong, H. Wang, Y.-H. Deng, M.-C. Chen, L.-C. Peng, Y.-H. Luo, J. Qin, D. Wu, X. Ding, Y. Hu, P. Hu, X.-Y. Yang, W.-J. Zhang, H. Li, Y. Li, X. Jiang, L. Gan, G. Yang, L. You, Z. Wang, L. Li, N.-L. Liu, C.-Y. Lu & J.-W. Pan; "Quantum computational advantage using photons;" Science **370**, pp. 1460–1463 (2020); URL http://dx.doi.org/10.1126/science.abe8770. (page 2)

[9] F. Arute *et al.*; "Quantum supremacy using a programmable superconducting processor;" Nature **574**, pp. 505–510 (2019); ISSN 0028-0836, 1476-4687; URL http://dx.doi.org/10.1038/s41586-019-1666-5. (pages 2 and 97)

[10] "IBM | Quantum Computing;" (2019); URL https://www.ibm.com/quantum-computing. (pages 2 and 97)

[11] J. Preskill; "Quantum Computing in the NISQ era and beyond;" Quantum **2**, p. 79 (2018); URL http://dx.doi.org/10.22331/q-2018-08-06-79. (page 2)

[12] C. E. Shannon; "A mathematical theory of communication;" The Bell System Technical Journal **27**, pp. 379–423 (1948); URL http://dx.doi.org/10.1002/j.1538-7305.1948.tb01338.x. (page 2)

[13] C. H. Bennett & G. Brassard; "Quantum cryptography: Public key distribution and coin tossing;" Theoretical Computer Science **560**, p. 7—11 (2014); URL http://dx.doi.org/10.1016/j.tcs.2014.05.025. (page 2)

[14] R. Schaller; "Moore's law: past, present and future;" IEEE Spectrum **34**, pp. 52–59 (1997); URL http://dx.doi.org/10.1109/6.591665. (page 2)

[15] W. Hu, C.-H. Chang, A. Sengupta, S. Bhunia, R. Kastner & H. Li; "An Overview of Hardware Security and Trust: Threats, Countermeasures, and Design Tools;" IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems **40**, pp. 1010–1038 (2021); URL http://dx.doi.org/10.1109/TCAD.2020.3047976. (page 2)

[16] J. Cobine & J. Curry; "Electrical Noise Generators;" Proceedings of the IRE **35**, pp. 875–879 (1947); URL http://dx.doi.org/10.1109/JRPROC.1947.229646. (page 3)

[17] F. Erata, S. Deng, F. Zaghloul, W. Xiong, O. Demir & J. Szefer; "Survey of Approaches and Techniques for Security Verification of Computer Systems;" J. Emerg. Technol. Comput. Syst. **19** (2023); ISSN 1550-4832; URL http://dx.doi.org/10.1145/3564785. (page 3)

[18] SIEMENS; "Questa Secure Check;" (2023); URL https://eda.sw.siemens.com/en-US/ic/questa/formal-verification/secure-check/. (page 3)

[19] J. R. M. Ziyad Hanna; "Formal Analysis of Security Data Paths in RTL Design;" (2012); URL https://research.ibm.com/haifa/conferences/hvc2012/papers/HVC2012_jamil_mazzawi.pdf. (page 3)

[20] J. Urdahl, S. Udupi, T. Ludwig, D. Stoffel & W. Kunz; "Properties first? A new design methodology for hardware, and its perspectives in safety analysis;" in "2016 IEEE/ACM International Conference on Computer-Aided Design (ICCAD)," pp. 1–8 (2016); URL http://dx.doi.org/10.1145/2966986.2980086. (page 3)

[21] S. Takarabt, K. Chibani, A. Facon, S. Guilley, Y. Mathieu, L. Sauvage & Y. Souissi; "Pre-silicon Embedded System Evaluation as New EDA Tool for Security Verification;" in "2018 IEEE 3rd International Verification and Security Workshop (IVSW)," pp. 74–79 (2018); URL http://dx.doi.org/10.1109/IVSW.2018.8494881. (page 3)

[22] Y. Hu, V. V. Menon, A. Schmidt, J. Monson, M. French & P. Nuzzo; "Security-Driven Metrics and Models for Efficient Evaluation of Logic Encryption Schemes;" in "Proceedings of the 17th ACM-IEEE International Conference on Formal Methods and Models for System Design," MEMOCODE '19 (Association for Computing Machinery, New York, NY, USA) (2019); ISBN 9781450369978; URL http://dx.doi.org/10.1145/3359986.3361207. (page 3)

[23] S. Patnaik, M. Ashraf, O. Sinanoglu & J. Knechtel; "Best of Both Worlds: Integration of Split Manufacturing and Camouflaging into a Security-Driven CAD Flow for 3D ICs;" in "2018 IEEE/ACM International Conference on Computer-Aided Design (ICCAD)," pp. 1–8 (2018); URL http://dx.doi.org/10.1145/3240765.3240784. (page 3)

[24] IDQuantique; "Delivering true Quantum randomness and unbreakable keys for greater trust;" (2023); URL https://www.idquantique.com/random-number-generation/overview/. (page 4)

[25] Toshiba; "Quantum Random Number Generators;" (2023); URL https://www.toshiba.eu/pages/eu/Cambridge-Research-Laboratory/quantum-random-number-generators. (page 4)

[26] K. Chakraborty, M. Doosti, Y. Ma, C. Wadhwa, M. Arapinis & E. Kashefi; "Quantum Lock: A Provable Quantum Communication Advantage;" Quantum **7**, p. 1014 (2023); ISSN 2521-327X; URL http://dx.doi.org/10.22331/q-2023-05-23-1014. (page 5)

[27] Y. Ma, E. Kashefi, M. Arapinis, K. Chakraborty & M. Kaplan; "QEnclave - A practical solution for secure quantum cloud computing;" npj Quantum Information **8** (2022); URL http://dx.doi.org/10.1038/s41534-022-00612-5. (pages 5 and 6)

[28] M. A. Nielsen & I. L. Chuang; *Quantum computation and quantum information*; 10th edition (Cambridge University Press) (2010); URL http://dx.doi.org/10.1017/CBO9780511976667. (pages 10, 12, and 16)

[29] W. K. Wootters & W. H. Zurek; "A single quantum cannot be cloned;" Nature **299**, pp. 802–803 (1982); URL http://dx.doi.org/10.1038/299802a0. (page 12)

[30] A. S. Holevo; "Statistical decision theory for quantum systems;" Journal of Multivariate Analysis **3**, pp. 337–394 (1973); URL http://dx.doi.org/10.1016/0047-259X(73)90028-6. (pages 12 and 66)

[31] H. Buhrman, R. Cleve, J. Watrous & R. de Wolf; "Quantum fingerprinting;" Physical Review Letters **87**, p. 167902 (2001); URL http://dx.doi.org/10.1103/PhysRevLett.87.167902. (pages 13 and 75)

[32] P. J. Coles, M. Berta, M. Tomamichel & S. Wehner; "Entropic uncertainty relations and their applications;" Reviews of Modern Physics **89**, p. 015002 (2017); URL http://dx.doi.org/10.1103/RevModPhys.89.015002. (pages 14, 15, 89, and 90)

[33] D. Deutsch; "Uncertainty in quantum measurements;" Physical Review Letters **50**, p. 631 (1983); URL http://dx.doi.org/10.1103/PhysRevLett.50.631. (page 14)

[34] H. Maassen & J. B. Uffink; "Generalized entropic uncertainty relations;" Physical review letters **60**, p. 1103 (1988); URL http://dx.doi.org/10.1103/PhysRevLett.60.1103. (page 14)

[35] A. Rényi; "On Measures of Entropy and Information;" (1961); URL https://api.semanticscholar.org/CorpusID:123056571. (page 15)

[36] R. Renner; "Security of quantum key distribution;" International Journal of Quantum Information **6**, pp. 1–127 (2008); URL https://arxiv.org/abs/quant-ph/0512258. (page 15)

[37] R. Konig, R. Renner & C. Schaffner; "The operational meaning of min-and max-entropy;" IEEE Transactions on Information theory **55**, pp. 4337–4347 (2009); URL http://dx.doi.org/10.1109/TIT.2009.2025545. (pages 15 and 92)

[38] A. Y. Kitaev; "Quantum computations: algorithms and error correction;" Russian Mathematical Surveys **52**, p. 1191 (1997); URL http://dx.doi.org/10.1070/RM1997v052n06ABEH002155. (page 16)

[39] R. Raussendorf & H. J. Briegel; "A One-Way Quantum Computer;" Phys. Rev. Lett. **86**, pp. 5188–5191 (2001); ISSN 0031-9007, 1079-7114; URL http://dx.doi.org/10.1103/PhysRevLett.86.5188. (page 16)

[40] R. Raussendorf, D. E. Browne & H. J. Briegel; "Measurement-based quantum computation on cluster states;" Phys. Rev. A **68**, p. 022312 (2003); URL http://dx.doi.org/10.1103/PhysRevA.68.022312. (page 16)

[41] H. J. Briegel, D. E. Browne, W. Dür, R. Raussendorf & M. V. d. Nest; "Measurement-based quantum computation;" Nature Physics **5**, pp. 19–26 (2009); ISSN 1745-2473, 1745-2481; URL http://dx.doi.org/10.1038/nphys1157. (page 16)

[42] V. Danos, E. Kashefi & P. Panangaden; "The measurement calculus;" Journal of the ACM **54**, p. 8 (2007); ISSN 0004-5411, 1557-735X; URL http://dx.doi.org/10.1145/1219092.1219096. (page 17)

[43] D. Gottesman & I. L. Chuang; "Demonstrating the viability of universal quantum computation using teleportation and single-qubit operations;" Nature **402**, pp. 390–393 (1999); URL http://dx.doi.org/10.1038/46503. (page 17)

[44] A. Broadbent, J. Fitzsimons & E. Kashefi; "Universal Blind Quantum Computation;" in "2009 50th Annual IEEE Symposium on Foundations of Computer Science," pp. 517–526 (2009); URL http://dx.doi.org/10.1109/FOCS.2009.36. (pages 17, 98, 102, and 103)

[45] J. Katz & Y. Lindell; *Introduction to Modern Cryptography* (Chapman and Hall/CRC Press) (2007); ISBN 978-1-58488-551-1; URL https://www.bibsonomy.org/bibtex/22aaba26235ec3b771dd49c89fc66395f/dblp. (pages 18, 19, 22, and 23)

[46] J. Hastad, R. Impagliazzo, L. Levin & M. Luby; "A Pseudorandom Generator from any One-way Function;" SIAM Journal on Computing **28** (1999); URL http://dx.doi.org/10.1137/S0097539793244708. (page 20)

[47] O. Goldreich, S. Goldwasser & S. Micali; "How to construct random functions;" in "JACM," (1986); URL https://api.semanticscholar.org/CorpusID:17064126. (page 20)

[48] Z. Ji, Y.-K. Liu & F. Song; "Pseudorandom Quantum States;" in "IACR Cryptology ePrint Archive," (2018); URL https://api.semanticscholar.org/CorpusID:51603717. (page 21)

[49] P. Ananth, L. Qian & H. Yuen; "Cryptography from Pseudorandom Quantum States;" arXiv preprint arXiv:2112.10020 (2022); URL http://dx.doi.org/10.48550/arXiv.2112.10020. (page 21)

[50] T. Morimae & T. Yamakawa; "Quantum Commitments and Signatures Without One-Way Functions;" in "Advances in Cryptology – CRYPTO 2022," pp. 269–295 (Springer Nature Switzerland) (2022); URL http://dx.doi.org/10.1007/978-3-031-15802-5_10. (page 21)

[51] W. Kretschmer; "Quantum Pseudorandomness and Classical Complexity;" (Schloss Dagstuhl - Leibniz-Zentrum für Informatik) (2021); URL http://dx.doi.org/10.4230/LIPICS.TQC.2021.2. (page 21)

[52] J. Daemen; "AES Proposal : Rijndael;" (1998); URL https://api.semanticscholar.org/CorpusID:17885291. (page 22)

[53] X. Bonnetain, M. Naya-Plasencia & A. Schrottenloher; "Quantum Security Analysis of AES;" Cryptology ePrint Archive, Paper 2019/272 (2019); URL https://eprint.iacr.org/2019/272. (pages 22 and 23)

[54] O. Dunkelman, N. Keller & A. Shamir; "Improved Single-Key Attacks on 8-round AES;" Cryptology ePrint Archive, Paper 2010/322 (2010); URL https://eprint.iacr.org/2010/322. (page 23)

[55] M. Ajtai; "Generating Hard Instances of Lattice Problems (Extended Abstract);" in "Proceedings of the Twenty-Eighth Annual ACM Symposium on Theory of Computing," STOC '96; p. 99—108 (Association for Computing Machinery, New York, NY, USA) (1996); ISBN 0897917855; URL http://dx.doi.org/10.1145/237814.237838. (page 24)

[56] J. Hoffstein, J. Pipher & J. H. Silverman; "NTRU: A Ring-Based Public Key Cryptosystem;" in "International Workshop on Ant Colony Optimization and Swarm Intelligence," (1998); URL https://api.semanticscholar.org/CorpusID:15330263. (page 24)

[57] O. Regev; "On Lattices, Learning with Errors, Random Linear Codes, and Cryptography;" in "Proceedings of the Thirty-seventh Annual ACM Symposium on Theory of Computing," pp. 84–93 (ACM) (2009); ISBN 1-58113-960-8; URL http://dx.doi.org/10.1145/1060590.1060603. (pages 24 and 98)

[58] C. Peikert; "A Decade of Lattice Cryptography;" Cryptology ePrint Archive, Paper 2015/939 (2015); URL https://eprint.iacr.org/2015/939. (page 24)

[59] D. Micciancio; "Generalized compact knapsacks, cyclic lattices, and efficient one-way functions from worst-case complexity assumptions;" in "The 43rd Annual IEEE Symposium on Foundations of Computer Science, 2002. Proceedings.", pp. 356–365 (2002); URL http://dx.doi.org/10.1109/SFCS.2002.1181960. (page 24)

[60] O. Goldreich & L. A. Levin; "A Hard-Core Predicate for All One-Way Functions;" in "Proceedings of the Twenty-First Annual ACM Symposium on Theory of Computing," STOC '89; p. 25—32 (Association for Computing Machinery, New York, NY, USA) (1989); ISBN 0897913078; URL http://dx.doi.org/10.1145/73007.73010. (page 24)

[61] V. Lyubashevsky, C. Peikert & O. Regev; "On Ideal Lattices and Learning with Errors Over Rings;" Cryptology ePrint Archive, Paper 2012/230 (2012); URL https://eprint.iacr.org/2012/230. (page 24)

[62] J. Ding, X. Xie & X. Lin; "A Simple Provably Secure Key Exchange Scheme Based on the Learning with Errors Problem;" Cryptology ePrint Archive, Paper 2012/688 (2012); URL https://eprint.iacr.org/2012/688. (page 25)

[63] G. Alagic, D. Cooper, Q. Dang, T. Dang, J. M. Kelsey, J. Lichtinger, Y.-K. Liu, C. A. Miller, D. Moody, R. Peralta, R. Perlner, A. Robinson, D. Smith-Tone & D. Apon; "Status Report on the Third Round of the NIST Post-Quantum Cryptography Standardization Process;" (2022); URL http://dx.doi.org/10.6028/NIST.IR.8413. (page 25)

[64] V. Shoup; "Sequences of Games: A Tool for Taming Complexity in Security Proofs;" IACR Cryptology ePrint Archive **2004**, p. 332 (2004); URL https://eprint.iacr.org/2004/332. (page 26)

[65] Y. Lindell; "How To Simulate It - A Tutorial on the Simulation Proof Technique;" Cryptology ePrint Archive, Paper 2016/046 (2016); URL https://eprint.iacr.org/2016/046. (page 26)

[66] R. Canetti; "Universally composable security: a new paradigm for cryptographic protocols;" in "Proceedings 42nd IEEE Symposium on Foundations of Computer Science," pp. 136–145 (2001); URL http://dx.doi.org/10.1109/SFCS.2001.959888. (page 27)

[67] R. Canetti, Y. Dodis, R. Pass & S. Walfish; "Universally Composable Security with Global Setup;" in "Theory of Cryptography," , edited by S. P. Vadhan; pp. 61–85 (Springer) (2007); ISBN 978-3-540-70936-7; URL http://dx.doi.org/10.1007/978-3-540-70936-7_4. (page 27)

[68] U. Maurer & R. Renner; "Abstract Cryptography;" in "ICS," (2011); URL http://conference.iiis.tsinghua.edu.cn/ICS2011/content/papers/14.html. (page 28)

[69] J. Kelsey, B. Schneier, D. Wagner & C. Hall; "Cryptanalytic Attacks on Pseudorandom Number Generators;" in "Fast Software Encryption," , edited by S. Vaudenay; pp. 168–188 (Springer Berlin Heidelberg, Berlin, Heidelberg) (1998); ISBN 978-3-540-69710-7; URL https://doi.org/10.1007/3-540-69710-1_12. (page 33)

[70] S. Knellwolf & W. Meier; "Cryptanalysis of the Knapsack Generator;" in "Fast Software Encryption," , edited by A. Joux; pp. 188–198 (Springer Berlin Heidelberg, Berlin, Heidelberg) (2011); ISBN 978-3-642-21702-9; URL https://doi.org/10.1007/978-3-642-21702-9_11. (page 33)

[71] F. Martinez; "Attacks on Pseudo Random Number Generators Hiding a Linear Structure;" Cryptology ePrint Archive, Paper 2021/1204 (2021); URL https://eprint.iacr.org/2021/1204. (page 33)

[72] R. Torrance & D. James; "The State-of-the-Art in IC Reverse Engineering;" in "Cryptographic Hardware and Embedded Systems - CHES 2009," , edited by C. Clavier & K. Gaj; pp. 363–381 (Springer Berlin Heidelberg, Berlin, Heidelberg) (2009); ISBN 978-3-642-04138-9; URL https://doi.org/10.1007/978-3-642-04138-9_26. (page 33)

[73] M. Lipp, M. Schwarz, D. Gruss, T. Prescher, W. Haas, A. Fogh, J. Horn, S. Mangard, P. C. Kocher, D. Genkin, Y. Yarom & M. Hamburg; "Meltdown: Reading Kernel Memory from User Space;" in "USENIX Security Symposium," (2018); URL https://dl.acm.org/doi/10.5555/3277203.3277276. (page 33)

[74] P. Kocher, J. Horn, A. Fogh, D. Genkin, D. Gruss, W. Haas, M. Hamburg, M. Lipp, S. Mangard, T. Prescher, M. Schwarz & Y. Yarom; "Spectre Attacks: Exploiting Speculative Execution;" in "2019 IEEE Symposium on Security and Privacy (SP)," pp. 1–19 (2019); URL http://dx.doi.org/10.1109/SP.2019.00002. (page 33)

[75] H. Zhun & C. Hongyi; "A truly random number generator based on thermal noise;" in "ASICON 2001. 2001 4th International Conference on ASIC Proceedings (Cat. No.01TH8549)," pp. 862–864 (2001); URL http://dx.doi.org/10.1109/ICASIC.2001.982700. (page 34)

[76] M. Ewert; "A Random Number Generator Based on Electronic Noise and the Xorshift Algorithm;" in "Proceedings of the 2018 VII International Conference on Network, Communication and Computing," ICNCC '18; p. 357—362 (Association for Computing Machinery, New York, NY, USA) (2018); ISBN 9781450365536; URL http://dx.doi.org/10.1145/3301326.3301359. (page 34)

[77] M. Haahr; "Random.org;" URL https://www.random.org/. (page 34)

[78] P. Kohlbrenner & K. Gaj; "An Embedded True Random Number Generator for FPGAs;" in "Proceedings of the 2004 ACM/SIGDA 12th International Symposium on Field Programmable Gate Arrays," FPGA '04; p. 71—78 (Association for Computing Machinery, New York, NY, USA) (2004); ISBN 1581138296; URL http://dx.doi.org/10.1145/968280.968292. (page 34)

[79] T. Stojanovski, J. Pihl & L. Kocarev; "Chaos-based random number generators. Part II: practical realization;" IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications **48**, pp. 382–385 (2001); URL http://dx.doi.org/10.1109/81.915396. (page 34)

[80] L. Bonilla, M. Alvaro & M. Carretero; "Chaos-based true random number generators;" Journal of Mathematics in Industry (2016); URL http://dx.doi.org/10.1186/s13362-016-0026-4. (page 34)

[81] T. Jennewein, U. Achleitner, G. Weihs, H. Weinfurter & A. Zeilinger; "A fast and compact quantum random number generator;" Review of Scientific Instruments **71**, pp. 1675–1680 (2000); URL http://dx.doi.org/10.1063/1.1150518. (page 35)

[82] A. Stefanov, N. Gisin, O. Guinnard, L. Guinnard & H. Zbinden; "Optical quantum random number generator;" Journal of Modern Optics **47**, pp. 595–598 (2000); URL http://dx.doi.org/10.1080/09500340008233380. (page 35)

[83] J. Rarity, P. Owens & P. Tapster; "Quantum Random-number Generation and Key Sharing;" Journal of Modern Optics **41**, pp. 2435–2444 (1994); URL http://dx.doi.org/10.1080/09500349414552281. (page 35)

[84] M. D. Eisaman, J. Fan, A. Migdall & S. V. Polyakov; "Invited Review Article: Single-photon sources and detectors;" Review of Scientific Instruments **82**, p. 071101 (2011); ISSN 0034-6748; URL http://dx.doi.org/10.1063/1.3610677. (page 35)

[85] C. M. Natarajan, M. G. Tanner & R. H. Hadfield; "Superconducting nanowire single-photon detectors: physics and applications;" Superconductor

Science and Technology **25**, p. 063001 (2012); URL http://dx.doi.org/10.1088/0953-2048/25/6/063001. (page 35)

[86] C. Gabriel, C. Wittmann, D. Sych, R. Dong, W. Mauerer, U. Andersen, C. Marquardt & G. Leuchs; "A generator for unique quantum random numbers based on vacuum states;" Nature Photonics **4**, pp. 711–715 (2010); URL http://dx.doi.org/10.1038/nphoton.2010.197. (page 35)

[87] Y. Shen, L. Tian & H. Zou; "Practical quantum random number generator based on measuring the shot noise of vacuum states;" Phys. Rev. A **81**, p. 063814 (2010); URL http://dx.doi.org/10.1103/PhysRevA.81.063814. (page 35)

[88] T. Symul, S. M. Assad & P. K. Lam; "Real time demonstration of high bitrate quantum random number generation with coherent laser light;" Applied Physics Letters **98** (2011); URL http://dx.doi.org/10.1063/1.3597793. (page 35)

[89] B. Qi, Y.-M. Chi, H.-K. Lo & L. Qian; "High-speed quantum random number generation by measuring phase noise of a single-mode laser;" Opt. Lett. **35**, pp. 312–314 (2010); URL http://dx.doi.org/10.1364/OL.35.000312. (page 35)

[90] F. Xu, B. Qi, X. Ma, H. Xu, H. Zheng & H.-K. Lo; "Ultrafast quantum random number generation based on quantum phase fluctuations;" Opt. Express **20**, pp. 12366–12377 (2012); URL http://dx.doi.org/10.1364/OE.20.012366. (page 35)

[91] Z. L. Yuan, M. Lucamarini, J. F. Dynes, B. Fröhlich, A. Plews & A. J. Shields; "Robust random number generation using steady-state emission of gain-switched laser diodes;" Applied Physics Letters **104** (2014); URL http://dx.doi.org/10.1063/1.4886761. (page 35)

[92] C. Abellá n, W. Amaya, M. Jofre, M. Curty, A. Acín, J. Capmany, V. Pruneri & M. W. Mitchell; "Ultra-fast quantum randomness generation by accelerated phase diffusion in a pulsed laser diode;" Optics Express **22**, p. 1645 (2014); URL http://dx.doi.org/10.1364/oe.22.001645. (page 35)

[93] Y.-Q. Nie, L. Huang, Y. Liu, F. Payne, J. Zhang & J.-W. Pan; "The generation of 68 Gbps quantum random number by measuring laser phase fluctuations;" Review of Scientific Instruments **86** (2015); URL http://dx.doi.org/10.1063/1.4922417. (page 35)

[94] S. Fehr, R. Gelles & C. Schaffner; "Security and composability of randomness expansion from Bell inequalities;" Physical Review A **87** (2013); URL http://dx.doi.org/10.1103/physreva.87.012335. (page 35)

[95] S. Pironio & S. Massar; "Security of practical private randomness generation;" Physical Review A **87** (2013); URL http://dx.doi.org/10.1103/physreva.87.012336. (page 35)

[96] G. Vallone, D. G. Marangon, M. Tomasin & P. Villoresi; "Quantum randomness certified by the uncertainty principle;" Physical Review A **90** (2014); URL http://dx.doi.org/10.1103/physreva.90.052327. (page 35)

[97] M. Avesani, D. G. Marangon, G. Vallone & P. Villoresi; "Source-device-independent heterodyne-based quantum random number generator at 17 Gbps;" Nature Communications **9** (2018); URL http://dx.doi.org/10.1038/s41467-018-07585-0. (page 35)

[98] D. Drahi, N. Walk, M. J. Hoban, A. K. Fedorov, R. Shakhovoy, A. Feimov, Y. Kurochkin, W. S. Kolthammer, J. Nunn, J. Barrett & I. A. Walmsley; "Certified Quantum Random Numbers from Untrusted Light;" Physical Review X **10** (2020); URL http://dx.doi.org/10.1103/physrevx.10.041048. (page 35)

[99] Y.-H. Li, X. Han, Y. Cao, X. Yuan, Z.-P. Li, J.-Y. Guan, J. Yin, Q. Zhang, X. Ma, C.-Z. Peng & J.-W. Pan; "Quantum random number generation with uncharacterized laser and sunlight;" npj Quantum Information **5** (2019); URL http://dx.doi.org/10.1038/s41534-019-0208-1. (page 35)

[100] Z. Cao, H. Zhou & X. Ma; "Loss-tolerant measurement-device-independent quantum random number generation;" New Journal of Physics **17**, p. 125011 (2015); URL http://dx.doi.org/10.1088/1367-2630/17/12/125011. (page 35)

[101] Y.-Q. Nie, J.-Y. Guan, H. Zhou, Q. Zhang, X. Ma, J. Zhang & J.-W. Pan; "Experimental measurement-device-independent quantum random-number generation;" Physical Review A **94** (2016); URL http://dx.doi.org/10.1103/physreva.94.060301. (page 35)

[102] R. Pappu, B. Recht, J. Taylor & N. Gershenfeld; "Physical One-Way Functions;" Science **297**, pp. 2026–2030 (2002); URL http://dx.doi.org/10.1126/science.1074376. (pages 38, 39, 42, and 43)

[103] B. Gassend, D. Clarke, M. Van Dijk & S. Devadas; "Silicon physical random functions;" in "Proceedings of the 9th ACM Conference on Computer and Communications Security," pp. 148–160 (2002); URL http://dx.doi.org/10.1145/586110.586132. (page 39)

[104] J. W. Lee, D. Lim, B. Gassend, G. E. Suh, M. Van Dijk & S. Devadas; "A technique to build a secret key in integrated circuits for identification and authentication applications;" in "2004 Symposium on VLSI Circuits. Digest of Technical Papers (IEEE Cat. No. 04CH37525)," pp. 176–179 (IEEE) (2004); URL http://dx.doi.org/10.1109/VLSIC.2004.1346548. (page 39)

[105] Daihyun Lim, J. Lee, B. Gassend, G. Suh, M. van Dijk & S. Devadas; "Extracting secret keys from integrated circuits;" IEEE Transactions

on Very Large Scale Integration (VLSI) Systems **13**, pp. 1200–1205 (2005); URL http://dx.doi.org/10.1109/TVLSI.2005.859470. (page 39)

[106] G. E. Suh & S. Devadas; "Physical Unclonable Functions for Device Authentication and Secret Key Generation;" in "2007 44th ACM/IEEE Design Automation Conference," pp. 9–14 (2007); URL http://dx.doi.org/10.1145/1278480.1278484. (pages 39, 78, 81, and 88)

[107] U. Rührmair, F. Sehnke, J. Sölter, G. Dror, S. Devadas & J. Schmidhuber; "Modeling Attacks on Physical Unclonable Functions;" CCS '10; p. 237—249 (Association for Computing Machinery) (2010); URL http://dx.doi.org/10.1145/1866307.1866335. (pages 40 and 78)

[108] G. T. Becker; "The Gap Between Promise and Reality: On the Insecurity of XOR Arbiter PUFs;" in "Cryptographic Hardware and Embedded Systems – CHES 2015," pp. 535–555 (2015); URL http://dx.doi.org/10.1007/978-3-662-48324-4_27. (page 40)

[109] G. T. Becker; "On the Pitfalls of using Arbiter-PUFs as Building Blocks;" 532 (2014); URL http://dx.doi.org/10.1109/TCAD.2015.2427259. (page 40)

[110] J. Delvaux; "Machine-learning attacks on polypufs, ob-pufs, rpufs, lhs-pufs, and puf–fsms;" IEEE Transactions on Information Forensics and Security **14**, pp. 2043–2058 (2019); URL http://dx.doi.org/10.1109/TIFS.2019.2891223. (page 40)

[111] U. Rührmair, J. Sölter, F. Sehnke, X. Xu, A. Mahmoud, V. Stoyanova, G. Dror, J. Schmidhuber, W. Burleson & S. Devadas; "PUF modeling attacks on simulated and silicon data;" IEEE transactions on information forensics and security **8**, pp. 1876–1891 (2013); URL http://dx.doi.org/10.1109/TIFS.2013.2279798. (page 40)

[112] B. Gassend, D. Lim, D. Clarke, M. van Dijk & S. Devadas; "Identification and Authentication of Integrated Circuits: Research Articles;" Concurr. Comput.: Pract. Exper. **16**, p. 1077—1098 (2004); URL http://dx.doi.org/10.1002/cpe.805. (page 40)

[113] M. Majzoobi, F. Koushanfar & M. Potkonjak; "Lightweight secure PUFs;" in "2008 IEEE/ACM International Conference on Computer-Aided Design," pp. 670–673 (2008); URL http://dx.doi.org/10.1109/ICCAD.2008.4681648. (page 40)

[114] N. Wisiol, G. T. Becker, M. Margraf, T. A. A. Soroceanu, J. Tobisch & B. Zengin; "Breaking the Lightweight Secure PUF: Understanding the Relation of Input Transformations and Machine Learning Resistance;" IACR Cryptol. ePrint Arch. **2019**, p. 799 (2019); URL https://ia.cr/2019/799. (page 40)

[115] P. H. Nguyen, D. P. Sahoo, C. Jin, K. Mahmood, U. Rührmair & M. van Dijk; "The Interpose PUF: Secure PUF Design against State-of-the-art Machine Learning Attacks;" IACR Transactions on Cryptographic Hardware and Embedded Systems **2019**, p. 243—290 (2019); URL http://dx.doi.org/10.13154/tches.v2019.i4.243-290. (page 40)

[116] M. Delavar, S. Mirzakuchaki & J. Mohajeri; "A Ring Oscillator-Based PUF With Enhanced Challenge-Response Pairs;" Canadian Journal of Electrical and Computer Engineering **39**, pp. 174–180 (2016); URL http://dx.doi.org/10.1109/CJECE.2016.2521877. (page 40)

[117] J. Guajardo, S. S. Kumar, G.-J. Schrijen & P. Tuyls; "FPGA intrinsic PUFs and their use for IP protection;" in "International workshop on cryptographic hardware and embedded systems," pp. 63–80 (Springer) (2007); URL http://dx.doi.org/10.1007/978-3-540-74735-2_5. (page 41)

[118] Y. Su, J. Holleman & B. P. Otis; "A Digital 1.6 pJ/bit Chip Identification Circuit Using Process Variations;" IEEE Journal of Solid-State Circuits **43**, pp. 69–77 (2008); URL http://dx.doi.org/10.1109/JSSC.2007.910961. (page 41)

[119] R. Maes, P. Tuyls & I. M. R. Verbauwhede; "Intrinsic PUFs from Flip-flops on Reconfigurable Devices;" (2008); URL https://api.semanticscholar.org/CorpusID:45613285. (page 41)

[120] K. Lofstrom, W. Daasch & D. Taylor; "IC identification circuit using device mismatch;" in "2000 IEEE International Solid-State Circuits Conference. Digest of Technical Papers (Cat. No.00CH37056)," pp. 372–373 (2000); URL http://dx.doi.org/10.1109/ISSCC.2000.839821. (page 41)

[121] R. Helinski, D. Acharyya & J. Plusquellic; "A physical unclonable function defined using power distribution system equivalent resistance variations;" in "2009 46th ACM/IEEE Design Automation Conference," pp. 676–681 (2009); URL http://dx.doi.org/10.1145/1629911.1630089. (page 41)

[122] P. Tuyls, G. Schrijen, B. Skoric, J. Geloven, N. Verhaegh & R. Wolters; "Read-Proof Hardware from Protective Coatings;" pp. 369–383 (2006); ISBN 978-3-540-46559-1; URL http://dx.doi.org/10.1007/11894063_29. (page 41)

[123] F. Armknecht, D. Moriyama, A.-R. Sadeghi & M. Yung; "Towards a Unified Security Model for Physically Unclonable Functions;" in "Topics in Cryptology - CT-RSA 2016," , volume 9610pp. 271–287 (2016); URL http://dx.doi.org/10.1007/978-3-319-29485-8-16. (pages 42 and 62)

[124] I. Marvian & S. Lloyd; "Universal quantum emulator;" arXiv preprint arXiv:1606.02734 (2016); URL http://dx.doi.org/10.48550/arXiv.1606.02734. (pages 43 and 75)

[125] P. Tuyls, B. Škorić, S. Stallinga, A. H. M. Akkermans & W. Ophey; "Information-Theoretic Security Analysis of Physical Uncloneable Functions;" in "Financial Cryptography and Data Security," , edited by A. S. Patrick & M. Yung; pp. 141–155 (Springer Berlin Heidelberg, Berlin, Heidelberg) (2005); ISBN 978-3-540-31680-0; URL https://doi.org/10.1007/11507840_15. (page 43)

[126] B. Skoric; "The entropy of keys derived from laser speckle;" arXiv preprint arXiv:0710.5002 (2007); URL http://dx.doi.org/10.48550/arXiv.0710.5002. (page 43)

[127] B. Skoric; "Quantum readout of Physical Unclonable Functions: Remote authentication without trusted readers and authenticated Quantum Key Exchange without initial shared secrets;" Cryptology ePrint Archive, Paper 2009/369 (2009); URL https://eprint.iacr.org/2009/369; https://eprint.iacr.org/2009/369. (page 44)

[128] S. A. Goorden, M. Horstmann, A. P. Mosk, B. Škorić & P. W. H. Pinkse; "Quantum-secure authentication of a physical unclonable key;" Optica **1**, pp. 421–424 (2014); URL http://dx.doi.org/10.1364/OPTICA.1.000421. (page 44)

[129] L. Fladung, G. M. Nikolopoulos, G. Alber & M. Fischlin; "Intercept-Resend Emulation Attacks against a Continuous-Variable Quantum Authentication Protocol with Physical Unclonable Keys;" Cryptography **3** (2019); ISSN 2410-387X; URL http://dx.doi.org/10.3390/cryptography3040025. (page 44)

[130] G. M. Nikolopoulos & E. Diamanti; "Continuous-variable quantum authentication of physical unclonable keys;" Scientific Reports **7** (2017); URL http://dx.doi.org/10.1038/srep46047. (page 44)

[131] K. Phalak, A. Ash-Saki, M. Alam, R. O. Topaloglu & S. Ghosh; "Quantum PUF for Security and Trust in Quantum Computing;" IEEE Journal on Emerging and Selected Topics in Circuits and Systems **11**, pp. 333–342 (2021); URL http://dx.doi.org/10.1109/jetcas.2021.3077024. (page 44)

[132] N. Pirnay, A. Pappa & J.-P. Seifert; "Learning classical readout quantum PUFs based on single-qubit gates;" Quantum Machine Intelligence **4** (2022); URL http://dx.doi.org/10.1007/s42484-022-00073-1. (page 44)

[133] N. Kumar, R. Mezher & E. Kashefi; "Efficient Construction of Quantum Physical Unclonable Functions with Unitary t-designs;" arXiv preprint arXiv:2101.05692 (2021); URL http://dx.doi.org/10.48550/arXiv.2101.05692. (page 45)

[134] F. G. S. L. Brandão, A. W. Harrow & M. Horodecki; "Local Random Quantum Circuits are Approximate Polynomial-Designs;" Communications in Mathematical Physics **346**, pp. 397–434 (2016); URL http://dx.doi.org/10.1007/s00220-016-2706-8. (page 45)

[135] M. Arapinis, M. Delavar, M. Doosti & E. Kashefi; "Quantum Physical Unclonable Functions: Possibilities and Impossibilities;" Quantum **5**, p. 475 (2021); URL http://dx.doi.org/10.22331/q-2021-06-15-475. (pages 45, 58, 62, and 63)

[136] GlobalPlatform; "TEE System Architecture v1.2;" (2018); URL https://globalplatform.org/wp-content/uploads/2017/01/GPD_TEE_SystemArch_v1.2_PublicRelease.pdf. (page 47)

[137] S. Sprague; "Attestation and TEE: Cybersecurity Controls with Privacy for Cloud Access;" URL https://globalplatform.org/attestation-and-tee-cybersecurity-controls-with-privacy-for-cloud-access/. (page 47)

[138] I. Anati, S. Gueron, S. Johnson & V. Scarlata; "Innovative Technology for CPU Based Attestation and Sealing;" (2013); URL https://api.semanticscholar.org/CorpusID:14218854. (pages 47 and 49)

[139] V. Costan, I. Lebedev & S. Devadas; "Sanctum: Minimal Hardware Extensions for Strong Software Isolation;" in "25th USENIX Security Symposium (USENIX Security 16)," pp. 857–874 (USENIX Association, Austin, TX) (2016); ISBN 978-1-931971-32-4; URL https://www.usenix.org/conference/usenixsecurity16/technical-sessions/presentation/costan. (pages 47 and 49)

[140] F. McKeen, I. Alexandrovich, A. Berenzon, C. V. Rozas, H. Shafi, V. Shanbhogue & U. R. Savagaonkar; "Innovative Instructions and Software Model for Isolated Execution;" in "Proceedings of the 2nd International Workshop on Hardware and Architectural Support for Security and Privacy," HASP '13 (Association for Computing Machinery, New York, NY, USA) (2013); ISBN 9781450321181; URL http://dx.doi.org/10.1145/2487726.2488368. (page 47)

[141] G. E. Suh, D. Clarke, B. Gassend, M. van Dijk & S. Devadas; "AEGIS: Architecture for Tamper-Evident and Tamper-Resistant Processing;" in "Proceedings of the 17th Annual International Conference on Supercomputing," ICS '03; p. 160—171 (Association for Computing Machinery, New York, NY, USA) (2003); ISBN 1581137338; URL http://dx.doi.org/10.1145/782814.782838. (pages 47 and 49)

[142] D. L. C. Thekkath, M. Mitchell, P. Lincoln, D. Boneh, J. Mitchell & M. Horowitz; "Architectural Support for Copy and Tamper Resistant Software;" in "Proceedings of the Ninth International Conference on Architectural Support for Programming Languages and Operating Systems," ASPLOS IX; p. 168—177 (Association for Computing Machinery, New York, NY, USA) (2000); ISBN 1581133170; URL http://dx.doi.org/10.1145/378993.379237. (pages 47 and 49)

[143] TrustedComputingGroup; "Trusted computing group;" URL http://www.trustedcomputinggroup.org. (page 48)

[144] M. Achemlal, S. Gharout & C. Gaber; "Trusted Platform Module as an Enabler for Security in Cloud Computing;" in "2011 Conference on Network and Information Systems Security," pp. 1–6 (2011); URL http://dx.doi.org/10.1109/SAR-SSI.2011.5931361. (page 48)

[145] W. Arbaugh, D. Farber & J. Smith; "A secure and reliable bootstrap architecture;" in "Proceedings. 1997 IEEE Symposium on Security and Privacy (Cat. No.97CB36097)," pp. 65–71 (1997); URL http://dx.doi.org/10.1109/SECPRI.1997.601317. (page 48)

[146] J. González; *Operating System Support for Run-Time Security with a Trusted Execution Environment*; Ph.D. thesis (2015); URL http://dx.doi.org/10.13140/RG.2.1.4827.8161. (page 49)

[147] D. Champagne & R. B. Lee; "Scalable architectural support for trusted software;" in "HPCA - 16 2010 The Sixteenth International Symposium on High-Performance Computer Architecture," pp. 1–12 (2010); URL http://dx.doi.org/10.1109/HPCA.2010.5416657. (page 49)

[148] A. Ferraiuolo, Y. Wang, R. Xu, D. Zhang, A. Myers & E. Suh; "Full-processor timing channel protection with applications to secure hardware compartments;" (2017); URL https://ecommons.cornell.edu/items/ca16b1e8-0f68-4204-9ea8-b0078d942981. (page 49)

[149] C. W. Fletcher, M. v. Dijk & S. Devadas; "A Secure Processor Architecture for Encrypted Computation on Untrusted Programs;" in "Proceedings of the Seventh ACM Workshop on Scalable Trusted Computing," STC '12; p. 3—8 (Association for Computing Machinery, New York, NY, USA) (2012); ISBN 9781450316620; URL http://dx.doi.org/10.1145/2382536.2382540. (page 49)

[150] M. Maas, E. Love, E. Stefanov, M. Tiwari, E. Shi, K. Asanovic, J. Kubiatowicz & D. Song; "PHANTOM: Practical Oblivious Computation in a Secure Processor;" in "Proceedings of the 2013 ACM SIGSAC Conference on Computer & Communications Security," CCS '13; p. 311—324 (Association for Computing Machinery, New York, NY, USA) (2013); ISBN 9781450324779; URL http://dx.doi.org/10.1145/2508859.2516692. (page 49)

[151] F. McKeen, I. Alexandrovich, A. Berenzon, C. V. Rozas, H. Shafi, V. Shanbhogue & U. R. Savagaonkar; "Innovative Instructions and Software Model for Isolated Execution;" in "Proceedings of the 2nd International Workshop on Hardware and Architectural Support for Security and Privacy," HASP '13 (Association for Computing Machinery, New York, NY, USA) (2013); ISBN 9781450321181; URL http://dx.doi.org/10.1145/2487726.2488368. (page 49)

[152] D. Zhang, Y. Wang, G. E. Suh & A. C. Myers; "A Hardware Design Language for Timing-Sensitive Information-Flow Security;" **43**, p. 503—

516 (2015); ISSN 0163-5964; URL http://dx.doi.org/10.1145/2786763.2694372. (page 49)

[153] T. Alves & D. Felton; "Trustzone: Integrated Hardware and Software Security;" (2004); URL https://www.techonline.com/tech-papers/trustzone-integrated-hardware-and-software-security/. (page 49)

[154] ARM; "ARM Security Technology Building a Secure System using TrustZone Technology;" (2009); URL https://developer.arm.com/documentation/PRD29-GENC-009492/c. (page 49)

[155] V. Costan & S. Devadas; "Intel SGX Explained;" Cryptology ePrint Archive, Paper 2016/086 (2016); URL https://eprint.iacr.org/2016/086. (page 49)

[156] T. K. Mandt, M. Solnik & D. Wang; "Demystifying the Secure Enclave Processor;" (2016); URL https://api.semanticscholar.org/CorpusID:150384840. (page 49)

[157] A.-R. Sadeghi, C. Wachsmann & M. Waidner; "Security and privacy challenges in industrial Internet of Things;" in "2015 52nd ACM/EDAC/IEEE Design Automation Conference (DAC)," pp. 1–6 (2015); URL http://dx.doi.org/10.1145/2744769.2747942. (page 49)

[158] D. Oliveira, M. Costa, S. Pinto & T. Gomes; "The Future of Low-End Motes in the Internet of Things: A Prospective Paper;" Electronics **9** (2020); ISSN 2079-9292; URL http://dx.doi.org/10.3390/electronics9010111. (page 49)

[159] X. Wu, R. Dunne, Q. Zhang & W. Shi; "Edge Computing Enabled Smart Firefighting: Opportunities and Challenges;" in "Proceedings of the Fifth ACM/IEEE Workshop on Hot Topics in Web Systems and Technologies," HotWeb '17 (Association for Computing Machinery, New York, NY, USA) (2017); ISBN 9781450355278; URL http://dx.doi.org/10.1145/3132465.3132475. (page 49)

[160] Y. Wu, J. Liao, P. Nguyen, W. Shi & Y. Yesha; "Bring Trust to Edge: Secure and Decentralized IoT Framework with BFT and Permissioned Blockchain;" in "2022 IEEE International Conference on Edge Computing and Communications (EDGE)," pp. 104–113 (2022); URL http://dx.doi.org/10.1109/EDGE55608.2022.00025. (page 49)

[161] L. Luo, Y. Zhang, C. Zou, X. Shao, Z. Ling & X. Fu; "On Runtime Software Security of TrustZone-M Based IoT Devices;" in "GLOBECOM 2020 - 2020 IEEE Global Communications Conference," pp. 1–7 (2020); URL http://dx.doi.org/10.1109/GLOBECOM42002.2020.9322370. (page 49)

[162] J. Noorman, F. Freiling, J. Van Bulck, J. Mühlberg, F. Piessens, P. Maene, B. Preneel, I. Verbauwhede, J. Götzfried & T. Müller; "Sancus 2.0: A Low-Cost Security Architecture for IoT Devices;" ACM

Transactions on Privacy and Security **20**, pp. 1–33 (2017); URL http://dx.doi.org/10.1145/3079763. (page 49)

[163] D. Oliveira, T. Gomes & S. Pinto; "uTango: an open-source TEE for IoT devices;" arXiv preprint arXiv:2102.03625 (2022); URL http://dx.doi.org/10.48550/arXiv.2102.03625. (page 49)

[164] Z. Ning, J. Liao, F. Zhang & W. Shi; "Preliminary Study of Trusted Execution Environments on Heterogeneous Edge Platforms;" (2018); URL http://dx.doi.org/10.1109/SEC.2018.00057. (page 49)

[165] K. Basu, D. Soni, M. Nabeel & R. Karri; "NIST Post-Quantum Cryptography- A Hardware Evaluation Study;" Cryptology ePrint Archive, Paper 2019/047 (2019); URL https://eprint.iacr.org/2019/047. (page 49)

[166] M. R. Albrecht, C. Hanser, A. Hoeller, T. Pöppelmann, F. Virdia & A. Wallner; "Implementing RLWE-based Schemes Using an RSA Co-Processor;" IACR Transactions on Cryptographic Hardware and Embedded Systems **2019**, p. 169—208 (2018); URL http://dx.doi.org/10.13154/tches.v2019.i1.169-208. (page 49)

[167] C. Gentry; *A Fully Homomorphic Encryption Scheme*; Ph.D. thesis; Stanford, CA, USA (2009); URL https://dl.acm.org/doi/10.5555/1834954. (page 50)

[168] R. Pass, E. Shi & F. Tramèr; "Formal Abstractions for Attested Execution Secure Processors;" in "Advances in Cryptology —EUROCRYPT 2017," , volume 10210, edited by J.-S. Coron & J. B. Nielsen; pp. 260–289 (Springer) (2017); ISBN 978-3-319-56619-1 978-3-319-56620-7; URL http://dx.doi.org/10.1007/978-3-319-56620-7_10. (pages 50, 111, and 112)

[169] S. Ghosh, S. Upadhyay & A. A. Saki; "A Primer on Security of Quantum Computing;" arXiv preprint arXiv:2305.02505 (2023); URL http://dx.doi.org/10.48550/arXiv.2305.02505. (page 51)

[170] E. Magesan & J. M. Gambetta; "Effective Hamiltonian models of the cross-resonance gate;" Physical Review A **101** (2020); URL http://dx.doi.org/10.1103/physreva.101.052308. (page 51)

[171] S. Sheldon, E. Magesan, J. M. Chow & J. M. Gambetta; "Procedure for systematically tuning up cross-talk in the cross-resonance gate;" Phys. Rev. A **93**, p. 060302 (2016); URL http://dx.doi.org/10.1103/PhysRevA.93.060302. (page 51)

[172] K. Rudinger, T. Proctor, D. Langharst, M. Sarovar, K. Young & R. Blume-Kohout; "Probing Context-Dependent Errors in Quantum Processors;" Physical Review X **9** (2019); URL http://dx.doi.org/10.1103/physrevx.9.021045. (page 51)

[173] M. Sarovar, T. Proctor, K. Rudinger, K. Young, E. Nielsen & R. Blume-Kohout; "Detecting crosstalk errors in quantum information processors;" Quantum **4**, p. 321 (2020); URL http://dx.doi.org/10.22331/q-2020-09-11-321. (page 51)

[174] A. Ash-Saki, M. Alam & S. Ghosh; "Analysis of Crosstalk in NISQ Devices and Security Implications in Multi-Programming Regime;" in "Proceedings of the ACM/IEEE International Symposium on Low Power Electronics and Design," ISLPED '20; p. 25—30 (Association for Computing Machinery, New York, NY, USA) (2020); ISBN 9781450370530; URL http://dx.doi.org/10.1145/3370748.3406570. (pages 51 and 52)

[175] A. Suresh, A. A. Saki, M. Alam, R. o Topalaglu & D. S. Ghosh; "A Quantum Circuit Obfuscation Methodology for Security and Privacy;" (2021); URL http://dx.doi.org/10.48550/arXiv.2104.05943. (page 52)

[176] S. Das & S. Ghosh; "Randomized Reversible Gate-Based Obfuscation for Secured Compilation of Quantum Circuit;" (2023); URL http://dx.doi.org/10.48550/arXiv.2305.01133. (page 52)

[177] A. A. Saki, A. Suresh, R. O. Topaloglu & S. Ghosh; "Split Compilation for Security of Quantum Circuits;" in "2021 IEEE/ACM International Conference On Computer Aided Design (ICCAD)," p. 1—7 (IEEE Press) (2021); URL http://dx.doi.org/10.1109/ICCAD51958.2021.9643478. (page 52)

[178] C. Langer, R. Ozeri, J. D. Jost, J. Chiaverini, B. DeMarco, A. Ben-Kish, R. B. Blakestad, J. Britton, D. B. Hume, W. M. Itano, D. Leibfried, R. Reichle, T. Rosenband, T. Schaetz, P. O. Schmidt & D. J. Wineland; "Long-Lived Qubit Memory Using Atomic Ions;" Phys. Rev. Lett. **95**, p. 060502 (2005); URL http://dx.doi.org/10.1103/PhysRevLett.95.060502. (page 53)

[179] P. C. Maurer, G. Kucsko, C. Latta, L. Jiang, N. Y. Yao, S. D. Bennett, F. Pastawski, D. Hunger, N. Chisholm, M. Markham, D. J. Twitchen, J. I. Cirac & M. D. Lukin; "Room-Temperature Quantum Bit Memory Exceeding One Second;" Science **336**, pp. 1283–1286 (2012); URL http://dx.doi.org/10.1126/science.1220513. (page 53)

[180] M. Steger, K. Saeedi, M. L. W. Thewalt, J. J. L. Morton, H. Riemann, N. V. Abrosimov, P. Becker & H.-J. Pohl; "Quantum Information Storage for over 180 s Using Donor Spins in a^{28} Si "Semiconductor Vacuum";" Science **336**, pp. 1280–1283 (2012); URL http://dx.doi.org/10.1126/science.1217635. (page 53)

[181] N. Bar-Gill, L. Pham, A. Jarmola, D. Budker & R. Walsworth; "Solid-state electronic spin coherence time approaching one second;" Nature Communications **4** (2013); URL http://dx.doi.org/10.1038/ncomms2771. (page 53)

[182] J. Yang, X. He, R. Guo, P. Xu, K. Wang, C. Sheng, M. Liu, J. Wang, A. Derevianko & M. Zhan; "Coherence Preservation of a Single Neutral Atom Qubit Transferred between Magic-Intensity Optical Traps;" Phys. Rev. Lett. **117**, p. 123201 (2016); URL http://dx.doi.org/10.1103/PhysRevLett.117.123201. (page 53)

[183] U. Rührmair, S. Devadas & F. Koushanfar; "Security based on physical unclonability and disorder;" in "Introduction to Hardware Security and Trust," pp. 65–102 (2012); URL http://dx.doi.org/10.1007/978-1-4419-8080-9_4. (page 55)

[184] A. I. Lvovsky, B. C. Sanders & W. Tittel; "Optical quantum memory;" Nature Photonics **3**, pp. 706–714 (2009); URL http://dx.doi.org/10.1038/nphoton.2009.231. (page 56)

[185] M.-D. Yu, M. Hiller, J. Delvaux, R. Sowell, S. Devadas & I. Verbauwhede; "A Lockdown Technique to Prevent Machine Learning on PUFs for Lightweight Authentication;" IEEE Transactions on Multi-Scale Computing Systems **2**, pp. 146–159 (2016); URL http://dx.doi.org/10.1109/TMSCS.2016.2553027. (page 57)

[186] N. Wisiol, C. Gräbnitz, C. Mühl, B. Zengin, T. Soroceanu, N. Pirnay, K. T. Mursi & A. Baliuka; "pypuf: Cryptanalysis of Physically Unclonable Functions;" (2021); URL https://zenodo.org/records/5222515. (pages 57 and 78)

[187] S. Wiesner; "Conjugate Coding;" SIGACT News **15**, p. 78—88 (1983); URL http://dx.doi.org/10.1145/1008908.1008920. (page 59)

[188] F.-X. Standaert; *Introduction to Side-Channel Attacks*; pp. 27–42 (2010); ISBN 978-0-387-71827-9; URL http://dx.doi.org/10.1007/978-0-387-71829-3_2. (page 61)

[189] P. Kocher, J. Jaffe & B. Jun; "Differential Power Analysis;" in "Advances in Cryptology — CRYPTO' 99," , edited by M. Wiener; pp. 388–397 (Springer Berlin Heidelberg, Berlin, Heidelberg) (1999); ISBN 978-3-540-48405-9; URL https://doi.org/10.1007/3-540-48405-1_25. (page 61)

[190] M. Doosti, M. Delavar, E. Kashefi & M. Arapinis; "A Unified Framework For Quantum Unforgeability;" arXiv preprint arXiv:2103.13994 (2021); URL http://dx.doi.org/10.48550/arXiv.2103.13994. (page 62)

[191] C. W. Helstrom; "Quantum detection and estimation theory;" Journal of Statistical Physics **1**, pp. 231–252 (1969); URL http://dx.doi.org/10.1007/BF01007479. (page 66)

[192] G. D'Ariano & P. Lo Presti; "Quantum Tomography for Measuring Experimentally the Matrix Elements of an Arbitrary Quantum Operation;" Physical review letters **86**, pp. 4195–8 (2001); URL http://dx.doi.org/10.1103/PhysRevLett.86.4195. (page 75)

[193] Y. Ma, C. Wadhwa, K. Chakraborty & M. Doosti; "Hybrid Locked PUF Simulation;" (2022); URL https://github.com/mayaobobby/hybridpuf_simulation/tree/main/Simulation_pypuf. (pages 80, 81, and 84)

[194] S. Bandyopadhyay, P. O. Boykin, V. Roychowdhury & F. Vatan; "A new proof for the existence of mutually unbiased bases;" Algorithmica **34**, pp. 512–528 (2002); URL http://dx.doi.org/10.1007/s00453-002-0980-7. (page 82)

[195] I. Tselniker, M. Nazarathy & M. Orenstein; "Mutually Unbiased Bases in 4, 8, and 16 Dimensions Generated by Means of Controlled-Phase Gates With Application to Entangled-Photon QKD Protocols;" IEEE Journal of Selected Topics in Quantum Electronics **15**, pp. 1713–1723 (2009); URL http://dx.doi.org/10.1109/JSTQE.2009.2021146. (page 82)

[196] C. Herder, M.-D. Yu, F. Koushanfar & S. Devadas; "Physical Unclonable Functions and Applications: A Tutorial;" Proceedings of the IEEE **102**, pp. 1126–1141 (2014); URL http://dx.doi.org/10.1109/JPROC.2014.2320516. (page 88)

[197] M. Tomamichel & R. Renner; "Uncertainty relation for smooth entropies;" Physical review letters **106**, p. 110506 (2011); URL http://dx.doi.org/10.1103/PhysRevLett.106.110506. (page 90)

[198] M. Sasaki, M. Fujiwara, H. Ishizuka, W. Klaus, K. Wakui, M. Takeoka, S. Miki, T. Yamashita, Z. Wang, A. Tanaka *et al.*; "Field test of quantum key distribution in the Tokyo QKD Network;" Optics express **19**, pp. 10387–10409 (2011); URL http://dx.doi.org/10.1364/OE.19.010387. (page 94)

[199] D. Stucki, M. Legre, F. Buntschu, B. Clausen, N. Felber *et al.*; "Long-term performance of the SwissQuantum quantum key distribution network in a field environment;" New Journal of Physics **13**, p. 123001 (2011); URL http://dx.doi.org/10.1088/1367-2630/13/12/123001. (page 94)

[200] A. Poppe, M. Peev & O. Maurhart; "Outline of the SECOQC quantum-key-distribution network in Vienna;" International Journal of Quantum Information **6**, pp. 209–218 (2008); URL http://dx.doi.org/10.1142/S0219749908003529. (page 94)

[201] S. Wang, W. Chen, Z.-Q. Yin, H.-W. Li, D.-Y. He *et al.*; "Field and long-term demonstration of a wide area quantum key distribution network;" Optics express **22**, pp. 21739–21756 (2014); URL http://dx.doi.org/10.1364/OE.22.021739. (page 94)

[202] R. Courtland; "China's 2,000-km quantum link is almost complete [News];" IEEE Spectrum **53**, pp. 11–12 (2016); URL http://dx.doi.org/10.1109/MSPEC.2016.7607012. (page 94)

[203] B. Fröhlich, M. Lucamarini, J. F. Dynes, L. C. Comandar, W. W.-S. Tam, A. Plews, A. W. Sharpe, Z. Yuan & A. J. Shields; "Long-distance quantum key distribution secure against coherent attacks;" Optica **4**, pp. 163–167 (2017); URL http://dx.doi.org/10.1364/OPTICA.4.000163. (page 94)

[204] P. Sibson, C. Erven, M. Godfrey, S. Miki, T. Yamashita *et al.*; "Chip-based quantum key distribution;" Nature communications **8**, pp. 1–6 (2017); URL http://dx.doi.org/10.1038/ncomms13984. (page 94)

[205] H. Semenenko, P. Sibson, A. Hart, M. G. Thompson, J. G. Rarity & C. Erven; "Chip-based measurement-device-independent quantum key distribution;" Optica **7**, pp. 238–242 (2020); URL http://dx.doi.org/10.1364/OPTICA.379679. (page 94)

[206] D. Bunandar, A. Lentine, C. Lee, H. Cai, C. M. Long, N. Boynton, N. Martinez, C. DeRose, C. Chen, M. Grein *et al.*; "Metropolitan quantum key distribution with silicon photonics;" Physical Review X **8**, p. 021009 (2018); URL http://dx.doi.org/10.1103/PhysRevX.8.021009. (page 94)

[207] D. Harris & S. Harris; *Digital design and computer architecture* (2010); URL http://dx.doi.org/10.1016/C2013-0-14352-8. (page 95)

[208] J. Arthur; "Microelectronics: Digital and Analog Circuits and Systems;" Electronics and Power **25**, pp. 729– (1979); URL http://dx.doi.org/10.1049/ep.1979.0409. (page 95)

[209] Z. Deng, A. Feldman, S. A. Kurtz & F. T. Chong; "Lemonade from Lemons: Harnessing Device Wearout to Create Limited-Use Security Architectures;" SIGARCH Comput. Archit. News **45**, p. 361—374 (2017); URL http://dx.doi.org/10.1145/3079856.3080226. (pages 95 and 102)

[210] D. Aharonov, V. Jones & Z. Landau; "A Polynomial Quantum Algorithm for Approximating the Jones Polynomial;" in "Proceedings of the Thirty-Eighth Annual ACM Symposium on Theory of Computing," p. 427—436 (Association for Computing Machinery) (2006); ISBN 1595931341; URL http://dx.doi.org/10.1145/1132516.1132579. (page 97)

[211] D. Alsina & J. I. Latorre; "Experimental test of Mermin inequalities on a five-qubit quantum computer;" Phys. Rev. A **94**, p. 012314 (2016); URL http://dx.doi.org/10.1103/PhysRevA.94.012314. (page 98)

[212] S. J. Devitt; "Performing quantum computing experiments in the cloud;" Phys. Rev. A **94**, p. 032329 (2016); URL http://dx.doi.org/10.1103/PhysRevA.94.032329. (page 98)

[213] M. Hebenstreit, D. Alsina, J. I. Latorre & B. Kraus; "Compressed quantum computation using a remote five-qubit quantum computer;" Phys. Rev. A **95**, p. 052339 (2017); URL http://dx.doi.org/10.1103/PhysRevA.95.052339. (page 98)

[214] Y. Wang, Y. Li, Z.-q. Yin & B. Zeng; "16-qubit IBM universal quantum computer can be fully entangled;" npj Quantum Inf. **4**, p. 46 (2018); ISSN 2056-6387; URL http://dx.doi.org/10.1038/s41534-018-0095-x. (page 98)

[215] J. F. Fitzsimons; "Private quantum computation: an introduction to blind quantum computing and related protocols;" npj Quantum Inf. **3**, p. 23 (2017); ISSN 2056-6387; URL http://dx.doi.org/10.1038/s41534-017-0025-3. (page 98)

[216] A. Gheorghiu, T. Kapourniotis & E. Kashefi; "Verification of Quantum Computation: An Overview of Existing Approaches;" Theory of Computing Systems **63**, pp. 715–808 (2019); URL http://dx.doi.org/10.1007/s00224-018-9872-3. (page 98)

[217] S. Aaronson, A. Cojocaru, A. Gheorghiu & E. Kashefi; "Complexity-Theoretic Limitations on Blind Delegated Quantum Computation;" in "46th ICALP 2019," , *Leibniz International Proceedings in Informatics (LIPIcs)*, volume 132pp. 6:1–6:13 (Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik) (2019); ISBN 978-3-95977-109-2; ISSN 1868-8969; URL http://dx.doi.org/10.4230/LIPIcs.ICALP.2019.6. (pages 98 and 118)

[218] U. Mahadev; "Classical Homomorphic Encryption for Quantum Circuits;" in "IEEE 59th Annual Symposium on Foundations of Computer Science (FOCS)," pp. 332–338 (IEEE Computer Society) (2018); URL http://dx.doi.org/10.1137/18M1231055. (page 98)

[219] U. Mahadev; "Classical Verification of Quantum Computations;" in "IEEE 59th Annual Symposium on Foundations of Computer Science (FOCS)," pp. 259–267 (IEEE Computer Society) (2018); URL http://dx.doi.org/10.1109/FOCS.2018.00033. (page 98)

[220] A. Cojocaru, L. Colisson, E. Kashefi & P. Wallden; "QFactory: Classically-Instructed Remote Secret Qubits Preparation;" in "Advances in Cryptology – ASIACRYPT 2019," , edited by S. D. Galbraith & S. Moriai; pp. 615–645 (Springer) (2019); ISBN 978-3-030-34578-5; URL http://dx.doi.org/10.1007/978-3-030-34578-5_22. (page 98)

[221] V. Dunjko & E. Kashefi; "Blind quantum computing with two almost identical states;" arXiv preprint arXiv:1604.01586 (2016); URL http://dx.doi.org/10.48550/arXiv.1604.01586. (pages 98, 103, 104, 105, 106, and 108)

[222] A. Gheorghiu & T. Vidick; "Computationally-Secure and Composable Remote State Preparation;" in "2019 IEEE 60th Annual Symposium on Foundations of Computer Science (FOCS)," pp. 1024–1033 (2019); URL http://dx.doi.org/10.1109/FOCS.2019.00066. (pages 98, 116, and 117)

[223] C. Badertscher *et al.*; "Security Limitations of Classical-Client Delegated Quantum Computing;" in "Advances in Cryptology – ASIACRYPT 2020," ,

edited by S. Moriai & H. Wang; pp. 667–696 (Springer) (2020); ISBN 978-3-030-64834-3; URL http://dx.doi.org/10.1007/978-3-030-64834-3_23. (pages 98, 111, and 118)

[224] V. Dunjko, J. F. Fitzsimons, C. Portmann & R. Renner; "Composable Security of Delegated Quantum Computation;" in "Advances in Cryptology – ASIACRYPT 2014," , edited by P. Sarkar & T. Iwata; pp. 406–425 (Springer) (2014); ISBN 978-3-662-45608-8; URL http://dx.doi.org/10.1007/978-3-662-45608-8_22. (pages 102 and 103)

[225] U. Maurer, B. Tackmann & S. Coretti; "Key Exchange with Unilateral Authentication: Composable Security Definition and Modular Protocol Design;" IACR Cryptology ePrint Archive **2013**, p. 555 (2013); URL https://ia.cr/2013/555. (page 112)

[226] S. Akleylek *et al.*; "An Efficient Lattice-Based Signature Scheme with Provably Secure Instantiation;" in "Progress in Cryptology – AFRICACRYPT 2016," , edited by D. Pointcheval, A. Nitaj & T. Rachidi; pp. 44–60 (Springer) (2016); ISBN 978-3-319-31517-1; URL http://dx.doi.org/10.1007/978-3-319-31517-1_3. (page 112)

[227] J. Buchmann, E. Dahmen & A. Hülsing; "XMSS - A Practical Forward Secure Signature Scheme based on Minimal Security Assumptions;" Cryptology ePrint Archive, Paper 2011/484 (2011); URL https://eprint.iacr.org/2011/484. (page 112)

[228] X. Liu, R. Misoczki & M. R. Sastry; "Remote Attestation for Low-End Prover Devices with Post-Quantum Capabilities;" in "Proceedings of the Eighth ACM Conference on Data and Application Security and Privacy," pp. 84–94 (ACM) (2018); ISBN 978-1-4503-5632-9; URL http://dx.doi.org/10.1145/3176258.3176324. (page 114)

[229] M. Baldi *et al.*; "LEDAkem: A Post-quantum Key Encapsulation Mechanism Based on QC-LDPC Codes;" in "Post-Quantum Cryptography," , edited by T. Lange & R. Steinwandt; pp. 3–24 (Springer) (2018); ISBN 978-3-319-79063-3; URL http://dx.doi.org/10.1007/978-3-319-79063-3_1. (page 114)

[230] N. Bindel *et al.*; "Hybrid Key Encapsulation Mechanisms and Authenticated Key Exchange;" in "Post-Quantum Cryptography," , edited by J. Ding & R. Steinwandt; pp. 206–226 (Springer) (2019); ISBN 978-3-030-25510-7; URL http://dx.doi.org/10.1007/978-3-030-25510-7_12. (page 114)

[231] W. Wang & M. Stöttinger; "Post-Quantum Secure Architectures for Automotive Hardware Secure Modules;" Cryptology ePrint Archive, Paper 2020/026 (2020); URL https://eprint.iacr.org/2020/026. (page 114)

[232] J. F. Fitzsimons & E. Kashefi; "Unconditionally verifiable blind quantum computation;" Phys. Rev. A **96**, p. 012303 (2017); URL http://dx.doi.org/10.1103/PhysRevA.96.012303. (pages 116, 117, and 118)

[233] E. Kashefi & P. Wallden; "Optimised resource construction for verifiable quantum computation;" Journal of Physics A: Mathematical and Theoretical **50**, p. 145306 (2017); ISSN 1751-8121; URL http://dx.doi.org/10.1088/1751-8121/aa5dac. (page 118)

[234] T. Kapourniotis, E. Kashefi, D. Leichtle, L. Music & H. Ollivier; "Asymmetric Quantum Secure Multi-Party Computation With Weak Clients Against Dishonest Majority;" Cryptology ePrint Archive, Paper 2023/379 (2023); URL https://eprint.iacr.org/2023/379. (page 118)

[235] J. Carolan *et al.*; "Universal linear optics;" Science **349**, pp. 711–716 (2015); URL https://www.science.org/doi/10.1126/science.aab3642. (page 119)

[236] A. Stute *et al.*; "Tunable ion–photon entanglement in an optical cavity;" Nature **485**, pp. 482–485 (2012); URL https://www.nature.com/articles/nature11120. (page 119)

[237] VeriQloud; "Quantum Protocol Zoo;" (2019); URL https://wiki.veriqloud.fr/index.php?title=Main_Page. (page 119)

[238] E. Kashefi & A. Pappa; "Multiparty Delegated Quantum Computing;" Cryptography **1**, p. 12 (2017); URL http://dx.doi.org/10.3390/cryptography1020012. (page 119)

[239] M. Doosti, L. Hanouz, A. Marin, E. Kashefi & M. Kaplan; "Establishing shared secret keys on quantum line networks: protocol and security;" (2023); URL https://doi.org/10.48550/arXiv.2304.01881. (page 119)

[240] B. Polacchi, D. Leichtle, L. Limongi, G. Carvacho, G. Milani, N. Spagnolo, M. Kaplan, F. Sciarrino & E. Kashefi; "Multi-client distributed blind quantum computation with the Qline architecture;" (2023); URL https://doi.org/10.48550/arXiv.2306.05195. (page 119)

[241] A. Broadbent & S. Jeffery; "Quantum Homomorphic Encryption for Circuits of Low T-gate Complexity;" in "Advances in Cryptology – CRYPTO 2015," , edited by R. Gennaro & M. Robshaw; pp. 609–629 (Springer) (2015); ISBN 978-3-662-48000-7; URL https://doi.org/10.1007/978-3-662-48000-7_30. (page 119)

[242] Y. Dulek, C. Schaffner & F. Speelman; "Quantum Homomorphic Encryption for Polynomial-Size Circuits;" Theory of Computing **14**, pp. 1–45 (2018); URL http://dx.doi.org/10.4086/toc.2018.v014a007. (page 119)

[243] S. Wiesner; "Conjugate coding;" ACM SIGACT News **15**, pp. 78–88 (1983); ISSN 0163-5700; URL http://dx.doi.org/10.1145/1008908.1008920. (page 120)

[244] R. Radian & Sattath; "Semi-Quantum Money;" in "Proceedings of the 1st ACM Conference on Advances in Financial Technologies," p. 132—146 (Association for Computing Machinery) (2019); ISBN 9781450367325; URL http://dx.doi.org/10.1145/3318041.3355462. (page 120)

[245] M. Mosca, A. Tapp & R. Wolf; "Private Quantum Channels and the Cost of Randomizing Quantum Information;" (2000); URL https://doi.org/10.48550/arXiv.quant-ph/0003101. (page 123)

[246] T. Trochatos, C. Xu, S. Deshpande, Y. Lu, Y. Ding & J. Szefer; "A Quantum Computer Trusted Execution Environment;" IEEE Computer Architecture Letters pp. 1–4 (5555); ISSN 1556-6064; URL http://dx.doi.org/10.1109/LCA.2023.3325852. (page 124)