6.857 Final Project: Security Analysis of Quantum Computing on the Ethereum Blockchain

Elisa Becker-Foss, Nishant Abhangi, Henry Jin

May 2022

1 Introduction

Blockchain is a relatively young technology that has grown into a significantly relevant blend of computer networks, security and economics in recent years. The first successful and widely adopted implementation of blockchain was the Bitcoin project spawned by the release of Satoshi Nakamoto's Bitcoin whitepaper in 2008. The market capitalisation of Bitcoin has since grown to 730 billion USD (as of May 1, 2022), with an estimated 120 million participants. Alongside this astronomic growth of value in Bitcoin, several other blockchains have also found traction, such as Ethereum, Solana, Cardano, and Tezos to name a few. In the aggregate, the entire cryptocurrency ecosystem built on various implementations of blockchain technology sums to a market value of roughly 2 trillion USD, rendering it an important and increasingly relevant field of study.

A particularly interesting blockchain is Ethereum, the largest blockchain with native smart contract functionalities. Unlike Bitcoin, Ethereum enables data to be stored on the distributed ledger, in addition to transactions of its native token Ether. This allows computer programs to be stored and executed on the blockchain in a distributed fashion. These programs are called smart contracts, and it is this functionality that has transformed Ethereum into an incredibly rich ecosystem with many different decentralised protocols and cryptocurrencies.

Alongside the rise of blockchain technology, quantum computing has also shown a lot of potential as a novel method of computing. With the imminent end of Moore's Law, physicists and computer scientists have turned to molecular and atomic scale representations of information. In November of 2021, IBM announced the completion of the currently largest quantum computer, the 127-Qubit Eagle. While quantum computing is certainly in its infancy, this technique, if scaled, promises to break current cryptographic security standards.

In this report, we analyse the implications of post-quantum security to the Ethereum blockchain. We first provide a more thorough illustration of the Ethereum blockchain, and it's importance to decentralised finance. We then discuss the relevant quantum computing techniques that pose threats to current security standards. The paper then offers an explanation of Ethereum's blockchain mechanics, and its cryptographic security measures. Lastly, we provide an exploration of the various attacks on and vulnerabilities of Ethereum in a post-quantum era, and their broader implications to decentralised finance and participants of the Ethereum ecosystem.

2 Background of Ethereum Blockchain

Ethereum inherits many of the features shared by all blockchains. The ultimate goal is to enable the exchange of value without a central authority. This is possible through cryptographic algorithms and consensus mechanisms, which allow for a decentralised ledger shared among all participants in a transparent, redundant, and accountable way. A consequence of this is that the ledger is immutable; any transactions logged onto the blockchain becomes permanently stored (unless a hard fork is triggered, which requires a consensus among network nodes). From a cryptographic point of view, hash functions and public key cryptography are the primary primitives used to ensure integrity, authenticity and privacy. The paper will dive deeper into the cryptography mechanics.

A differentiating feature of Ethereum from Bitcoin is that it uses the ”Account Model” (or ”Balance Model”) as opposed to the UTXO model used by Bitcoin. This is to say that Ethereum's ledger works by keeping track of balances in each wallet address, and the same address is used for all future transactions for that particular wallet unless explicitly forced to use a new address. In the UTXO model, on the other hand, addresses represent transactions, as opposed to wallets, so multiple addresses may correspond to an individual. This is a particularly relevant difference for quantum computing security, as all assets for person is grouped into one wallet, increasing the exposure under a security breach.

Consequently, funds in Ethereum can be categorised as ”safe funds” or ”quantum-exposed funds”. Safe funds are funds stored in accounts that have never committed an outbound transaction. Any outgoing transaction requires the emission of the origin's public key, which can be retroactively attacked by a quantum computer. Without an outgoing transaction then, an Ethereum wallet is secure from any known quantum computing attack. All other funds are quantum-exposed.

Figure 2 shows the amount of safe and quantum-exposed funds in each quarter of the last 7 years. It becomes apparent that the amount of quantum-exposed Ether has been growing significantly quarter over quarter, highlighting the importance of studying Ethereum security in a post-quantum world.

Ethereum also enables computer programs to be run in a decentralised and agreed-upon fashion in the form of smart contracts. Smart contracts have since become a significant aspect of the Ethereum ecosystem, enabling new cryptocurrencies to be built on top of the Ethereum ecosystem, as well as introducing a notion of digital assets through non-fungible tokens (NFTs).

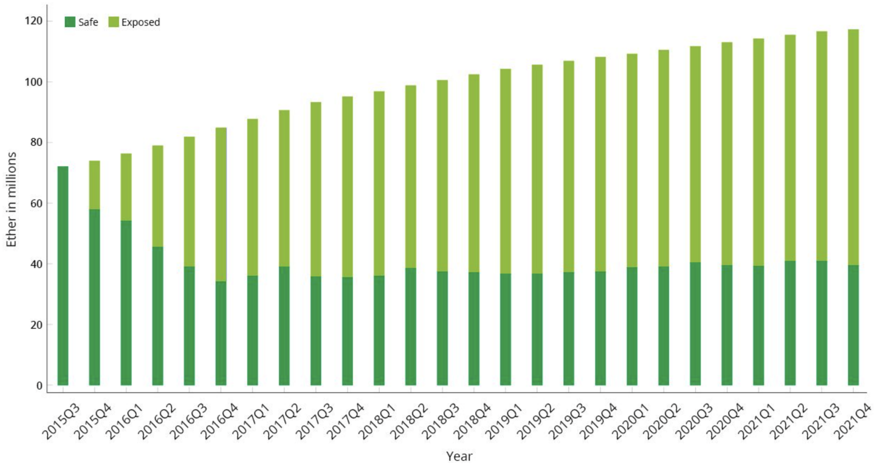


Figure 1: Safe vs Quantum-Exposed Funds on the Ethereum Blockchain

3 Quantum Computing Methods

Classical computers carry out logical operations using the definite position of a physical state. Its operations are based on either 1 (up) or 0 (down) and are thus binary. This single state is called a bit.

In quantum computing, operations instead use the so called quantum state of an object to produce what is called a qubit. These states are the undefined (instead of definite) properties of an object before they have been detected. A similar idea applies to the spin of an electron or the polarisation of a photon. Rather than having a clear position, unmeasured quantum states occur in superposition, similar a coin spinning through the air before it lands in your hand. These superpositions can be entangled with those of other objects, meaning their final outcomes will be mathematically related even if we don't know yet what they are. In other words, entangled means that if you observe something about one state, you automatically know this information about the other state. Once a qubit state is observed, its environment is disturbed and it collapses into a definite state of 0 or 1. Once a superposition interacts with materials that are part of a measured system, it loses its in-between state in what's known as decoherence and becomes a classical bit.

Therefore, devices need to be able to shield quantum states from decoherence, while still making them easy to read. Different processes are tackling this challenge from different angles, whether it's to use more robust quantum processes or to find better ways to check for errors.

If an eavesdropper, for example, were to secretly try and access a secret code, the resulting message would be distorted and the recipient would know that. However, we live in the real world where nothing is certain, so there are ways to access secure info even with a quantum computer. The question here is if the distorted message was hacked or just acted on by its environment. Researchers are actively working to reduce the uncertainty due to environmental factors and increase the fidelity of secure messaging.

This leads to the term of error-correction which is a characterization of uncertainty and how to account for it. For every piece of information stored in qubits, there is an error-correcting code to account for the environmental and computational loss of this information as it is stored as well as the method of its communication and realization.

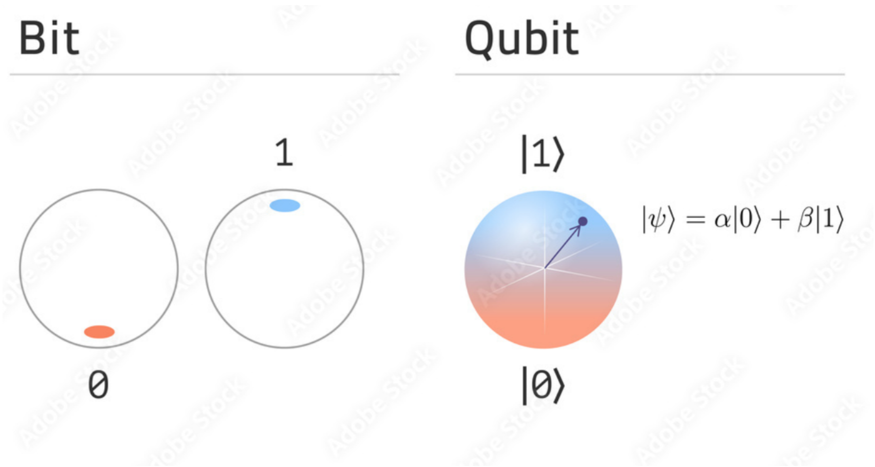


Figure 2: Bit vs Qubit. [1]

The complex mathematics behind these unsettled states can be plugged into special algorithms that drasitcally reduce the amount of time needed that would take a classical computer a long time to work out. Two are main algorithms could be feasible to run on a QC: Grover's algorithm, developed in 1996 by Lov Grover, is expected to reduce the security of hashing algorithms by 50% because of its O(sqrt(N)) time for brute force searching in N keys. The other alogorithm is called the Shor's algorithm for finding the discrete log in polynomial time, developed by Peter Shor in 1994. This algorithm is believed to completely break public-key cryptography which is used for generating digital signatures in practically all cryptocurrencies.

3.1 Shor's algorithm

We first describe the quantum fourier transform which is used to build an efficient algorith for phase estimation. Shor's algorithm can be seen

3.1.1 Quantum Fourier Transform

For an N-dimensional vector space, let $|0\rangle, \cdots, |N-1\rangle$ be its basis. Then its **Quantum Fourier Transform** (QFT) is an operator which transforms each basis state $|j\rangle$ as follows

$$|j\rangle \longrightarrow \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} e^{2\pi ijk/N}|k\rangle$$

If $N = 2^n$ for some $n \in \mathbb{N}$, we can write j in its binary representation $j = j_1j_2 \cdots j_n$. In other words, $j = j_1 2^{n-1} + j_2 2^{n-2} + \cdots + j_n 2^0$. Also, the notation $0.j_l j_{l+1} \cdots j_m$ denotes the fraction $j_l/2 + j_{l+1}/4 + \cdots + j_m/2^{m-l+1}$. Then, we can rewrite the QFT as

$$|j_1, \ldots, j_n\rangle \to \frac{\left(|0\rangle + e^{2\pi i 0 \cdot j_n}|1\rangle\right)\left(|0\rangle + e^{2\pi i 0 \cdot j_{n-1} j_n}|1\rangle\right) \cdots \left(|0\rangle + e^{2\pi i 0 \cdot j_1 j_2 \cdots j_n}|1\rangle\right)}{2^{n/2}}$$

It can be shown that this transform can be computed using $\theta(n^2)$ gates compared to the classical $\theta(n2^n)$ gates required for performing the classical fourier transform.

3.1.2 Phase estimation

Suppose a unitary operator U has an eigenvector $|u\rangle$ with eigenvalue $e^{2*\pi*i*\varphi}$. The phase estimation algorithm computes φ efficiently. This algorithm requires 2 sets of qubits (called registers):

1. The first register contains t qubits initially in the state $|0\rangle.t$ determines how many significant figures of φ we can estimate and the probability that the algorithm is successful.

2. The second register begins in the state $|u\rangle$ and contains as many qubits as is necessary to store $|u\rangle$.

The phase estimation algorithm is outlined below: The initial state of the qubits as this :

$$|0\rangle\,|u\rangle$$

. Then, we superpose these qubits so that we can perform computation using all of them simultaneously to get

$$\frac{1}{\sqrt{2^t}} \sum_{j=0}^{2^t-1} |j\rangle|u\rangle$$

Then we apply the quantum operator U on the second set but the number of times it is applied is controlled by the first set. We, get

$$\frac{1}{\sqrt{2^t}} \sum_{j=0}^{2^t-1} |j\rangle U^j|u\rangle = \frac{1}{\sqrt{2^t}} \sum_{j=0}^{2^t-1} e^{2\pi ij\varphi_u}|j\rangle|u\rangle$$

Then, we apply the inverse QFT. It is intuitive from the above expression that the state closest to ϕ would have the above QFT. Indeed, we get

$$|\widetilde{\varphi_u}\rangle\,|u\rangle$$

and upon measurement of the first register we recover $\widetilde{\varphi_u}$.

The $\widetilde{\varphi_u}$ is an n bit approximation to φ_u. To obtain that, we need to set $t = n + \left\lceil \log\left(2 + \frac{1}{2\epsilon}\right)\right\rceil$, where $1-\epsilon$ is the success probability of the algorithm. the algorithm runs in time $O(t^2)$. Clearly, this shows that the expected runtime of the algorithm is polynomial in n even for very small ϵ (due to the logarithm).

3.1.3 Order Finding

Recall that the order of x is the smallest r, such that $x^r = 1(modN)$ where the group is Z_N^*. All operations are modulo N from now on. Order finding algorithm computes the order for given x and N.We would use phase estimation for this with $U\,|y\rangle = |xy\rangle$. Its eigenstates are

$$|u_s\rangle \equiv \frac{1}{\sqrt{r}}\sum_{k=0}^{r-1}\exp\left[\frac{-2\pi isk}{r}\right]\left|x^k \bmod N\right\rangle$$

for $0 \le s \le r-1$. The issue is that eigenstates depend on r, which is what we want to find. However, note that

$$\frac{1}{\sqrt{r}}\sum_{s=0}^{r-1}|u_s\rangle = |1\rangle$$

Hence, we use $t = 2L + 1 + \left\lceil \log\left(2 + \frac{1}{2\epsilon}\right)\right\rceil$, where L is the number of bits in N, and prepare the second register to be $|1\rangle$. This means that for each s, we will obtain an estimate of $\varphi \approx s/r$ with $2L+1$ bits accuracy and the probability of success is $(1-\epsilon)/r$. Then we can use continued fractions method to obtain r from s/r. The procedure is summarized below

1. $|0\rangle|1\rangle$ initial state
2. $\rightarrow \frac{1}{\sqrt{2^t}}\sum_{j=0}^{2^t-1}|j\rangle|1\rangle$ create superposition
3. $\rightarrow \frac{1}{\sqrt{2^t}}\sum_{j=0}^{2^t-1}|j\rangle\left|x^j \bmod N\right\rangle \approx \frac{1}{\sqrt{r2^t}}\sum_{s=0}^{r-1}\sum_{j=0}^{2^t-1}e^{2\pi isj/r}|j\rangle\,|u_s\rangle$ apply $U_{x,N}$
4. $\rightarrow \frac{1}{\sqrt{r}}\sum_{s=0}^{r-1}\widetilde{|s/r\rangle}\,|u_s\rangle$ apply inverse Fourier transform to first register
5. $\longrightarrow \widetilde{s/r}$ measure first register
6. $\longrightarrow r$ apply continued fractions algorithm

The runtime of this algorithm is $O(L^3)$ and it succeeds with probability O(1).

3.1.4 Factoring

Finally, we come to Shor's algorithm. We show that it can be reduced to order finding. Let N be a composite number. This algorithm outputs a non-trivial factor of N with probability O(1) in O($(\log N)^3$) time.

1. If N is even, return 2.
2. Determine whether $N = a^b$ for $a, b \geq 2$ and if so return a. This can be done efficiently classically.
3. Choose a random $x \in [0, N-1]$. If $gcd(x, N) > 1$, return $gcd(x, N)$.
4. Now, we have x is coprime to N. We use order finding to find the order r of x.
5. If r is even and $x^{r/2} \neq -1$, then compute $gcd(x^{r/2} - 1, N)$ and $gcd(x^{r/2} + 1, N)$. If either of them is a non-trivial factor return that factor. Otherwise, the algorithm fails.

Note that r is even and $x^{r/2} \neq -1$ with probability atleast 1/2 and then it can be shown that when these conditions are satisfied the algorithm always succeeds. Hence success probability is $O(1)$.

Hence, since the hardness of factoring and in general phase estimation is crucial for most public key cryptosystems, quantum computers break them. In particular ECDSA used in ethereum breaks down.

4 Ethereum Mechanics and Current State of Ethereum Security

Here, we provide an outline of the Ethereum blockchain's mechanics, as they are relevant for quantum-computing attacks.

There are two primary primitives used by the Ethereum blockchain: hash functions, and public-key cryptography. Ethereum uses keccak-256 as its hash function, and it is used in the proof-of-work protocol in order to achieve consensus.

Figure 3 illustrates how the keccak-256 hashing algorithm is used to chain blocks. A block contains the hash of the previous block. In order to generate a new hash for this block, a nonce is found by mining nodes (this is the proof-of-work protocol). The next block repeats this process.

The more relevant primitive for post-quantum security however, is public key cryptography, as it is more readily attacked by quantum computing algorithms such as Shor's Algorithm. Figure 4 illustrates what a transaction looks like on the blockchain. In this example, \$15 of assets is being sent from one address to another. In order to prevent anyone from announcing any transaction, every transaction must be signed. Below the transaction is the signature. This ensures authenticity, and prevents an adversary from arbitrarily sending themselves funds from any account.

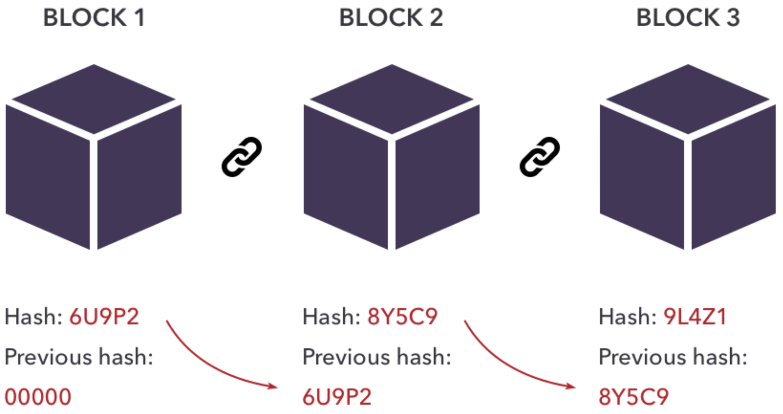
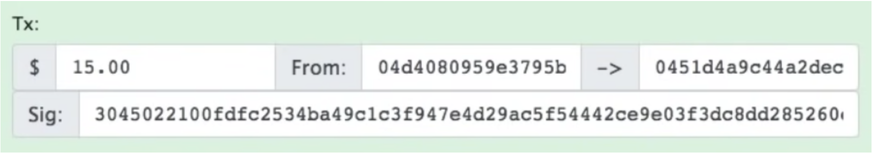


Figure 3: Hashing is used to chain blocks.

The exact algorithm used for digital signing on Ethereum is Elliptic Curve Digital Signing Algorithm. This algorithm works in a similar way to RSA. A public key is generated from a private key (on an elliptic curve finite field), and the private key is used to sign messages while the public key is used to verify.



Tx:

$ 15.00 From: 04d4080959e3795b -> 0451d4a9c44a2dec

Sig: 3045022100fdfc2534ba49c1c3f947e4d29ac5f54442ce9e03f3dc8dd285260

Figure 4: Every transaction on the blockchain is signed by the sender.

5 Attacks and Vulnerabilities in Post-Quantum Ethereum

The consequence of an efficient method of breaking Elliptic Curve Cryptography is existential forgery. With a sufficiently powerful quantum computer, Elliptic Curve Public Key Cryptography can be broken using a variation of Shor's algorithm designed specifically to solve the Elliptic Curve Discrete Log Problem. This enables efficient finding of the private key, given a public key and generator, both of which are public. Once an adversary has the corresponding private key for a particular Ethereum wallet address, the adversary can submit transactions to the memory pool, and sign them with the discovered private key. Then, all other nodes would verify the maliciously submitted transactions as authentic, and add them to the blockchain. This empowers the adversary to steal all assets held by the victim wallet.

One particular type of attack that can occur is Storage Attacks. In a Storage Attack, the adversary would target a quantum-exposed wallet (one that has a record of an outgoing transaction, and thus a publicised public key). Given the public key and generator, the attacker uses a sufficiently powerful quantum computer to find the corresponding private key. Any wallet address that has ever sent outbound funds on the Ethereum blockchain is vulnerable to this type of attack.

Another attack is a Transit Attack. In a Transit Attack, the adversary targets a quantum safe wallet that has just submitted its first outbound transaction. While the authentic transaction awaits mining in the memory pool, the attacker uses the recently publicised public key to find the corresponding private key before the authentic transaction is mined by the Ethereum Network. Once found, the attacker can submit a competing and signed transaction that redirects the same amount of funds from the authentic transaction to another wallet of the attackers choosing. If the gas limit of the attacker's transaction is considerably higher than that of the authentic transaction, the attacker's competing transaction is likely to be mined first, and the funds will successfully be redirected out of the victim's wallet.

As of May 10 2022, the total market capitalisation of Ethereum is 285 billion USD. Of this, approximately two thirds are quantum-exposed, implying that roughly 190 billion USD of funds are vulnerable to the attacks outlined above.

5.1 Smart Contract Cryptocurrency Attacks

Both Storage Attacks and Transit Attacks occur on the native Ethereum currency Ether. They accomplish this by maliciously signing transactions that call for a value transfer from one wallet address to another in the Transaction field of an Ethereum block. However, Ethereum is a versatile blockchain that enables data transfers in addition to value (Ether) transfers in a separate Data field. Smart contracts are powered by these data transfers.

A particular format of smart contracts that has recently become widely adopted as a cryptocurrency standard is the ERC-20 (Ethereum Request for Comment). This format underlies several popular cryptocurrencies, including USDC, DAI, Tether, Chainlink and Shiba Inu. These cryptocurrencies track balances with a dictionary of wallet addresses as keys, known in the Solidity programming language as mappings. In the ERC-20 regime, transactions of value do not occur on the Transaction field of the Ethereum blockchain (as this field is reserved for the native Ether currency). Instead transactions are simply changes in the smart contract mapping of wallet addresses to balances.

In a post-quantum paradigm then, these smart contract based cryptocurrencies are also vulnerable to forgery attacks. An adversary with sufficient cubits can take the public key of a quantum-exposed wallet address, and use Shor's Algorithm to efficiently solve the Elliptic Curve Discrete Logarithm Problem and find the corresponding private key. With the private key, an attacker can forge any data-related transactions on the Ethereum blockchain, and interact with smart contracts on the victim's behalf. Consequently, a quantum-exposed user's assets in any smart contract is also vulnerable, along with their balance of Ether.

One difference that smart contract cryptocurrency attacks have from native Ether attacks is that smart contract calls are typically an order of magnitude more expensive in gas fees (the incentive fee for miners to mine a transaction and thus add it to the blockchain). Any transaction to be mined on the blockchain costs Ether, and it is solely a matter of how much Ether a transaction costs. For instance, as of May 10 2022, the fee for sending the ERC-20 token USDC from one wallet address to another required roughly 30 USD equivalent of Ether. Sending the native token Ether however from one wallet to another on the same day costs roughly 3 USD equivalent of Ether. This is because the transfer of USDC is data-based, and is more costly in computing power than a simple transaction of Ether. With insufficient fees to cover the gas fee of a transaction, that transaction will sit idly in the memory pool waiting to be mined until the gas price drops such that the fee offered is sufficient for miners to mine. As a result some quantum-exposed wallets may be inadvertently safe from smart contract attacks simply by having insufficient Ether for any adversarial data transactions to be executed. However, this is somewhat of a trivial protection scheme, since an attacker could simply send the victim's wallet enough Ether to cover the gas fees of interacting with smart contracts, thus enabling the attacker to siphon any smart-contract based assets from the victim's wallet to the attacker's wallet.

This attack extends beyond cryptocurrencies built on Ethereum. It broadly applies to any smart-contract based digital asset class. This also includes semi-fungible tokens and non-fungible tokens (NFTs). These classes of assets have their own standards (ERC-1155 and ERC-721 respectively), but the attack is the same, since the mechanism by which these smart contracts track balances is the same. There exists a mapping from a wallet address to their balance, which is stored in a decentralised way. Thus an adversary could use the discovered private key to send any smart-contract based digital asset from the victim's wallet to their own. Certain NFT assets are extremely valuable, and could pose as more targeted attack opportunities, as the owner's wallet address is easily findable on platforms like OpenSea and Etherscan. Bored Apes, for instance, is a series of non-fungible tokens with a floor price of 234,000 USD. So each single attack on this particular non-fungible token costs at least 234,000 USD.

5.2 Governance Attacks

Governance attacks can also occur in a post-quantum world. In the spirit of decentralisation, many blockchain protocols are community-governed through a distribution of governance tokens. Ownership of such tokens grants the owner voting power in the protocol's or community's future directions. Typical voting mechanisms are simple majorities. A threat then is that a quantum-computing adversary can find the private keys of the governance token holders of a protocol, and transfer over 50% of that token's supply to their own wallet, thus having a simple majority to direct the protocol's direction. This could have serious ramifications for decentralised finance protocols such as Uniswap, the largest decentralised exchange for cryptocurrencies. By controlling a majority voting share of governance tokens for Uniswap, an adversary can manipulate the exchange rates and even sabotage the entire exchange.

6 Conclusion

In this project we showed how the security of a Blockchain can be disturbed by the evolution of Quantum Computing. We decomposed the security of a Blockchain into different components, the Hash algorithms on the one side and the public key cryptography on the other side. We looked at several different attacks that might be feasible running on a Blockchain given the fast increasing computing power of a Quantum computer. We discuss potential attacks on the native Ethereum token Ether, as well as on smart contracts and governance tokens. Moving forward, there are a few research areas in the field of post-quantum security, such as supersingular elliptic curve isogeny, lattice-based cryptography, and hash-based cryptography to name a few. While there hasn't been a serious effort from the Ethereum foundation to migrate to these post-quantum security techniques yet, we believe the quantum-computing era is a few decades away, and a long-term plan to shift to stronger security measures should be initiated.

References

Alberto Torres, W. A., Steinfeld, R., Sakzad, A., Liu, J. K., Kuchta, V., Bhattacharjee, N., … Cheng, J. (2018). Post-quantum one-time linkable ring signature and application to ring confidential transactions in blockchain (lattice ringct v1.0). In W. Susilo & G. Yang (Eds.), *Information security and privacy* (pp. 558–576). Cham: Springer International Publishing.

Childs, A. (2008). *Lecture 3: Quantum attacks on elliptic curve cryptography.* University Lecture.

Fernández-Caramès, T. M., & Fraga-Lamas, P. (2020). Towards post-quantum blockchain: A review on blockchain cryptography resistant to quantum computing attacks. *IEEE Access*, *8*, 21091-21116. doi: 10.1109/ACCESS.2020.2968985

Gao, Y.-L., Chen, X.-B., Chen, Y.-L., Sun, Y., Niu, X.-X., & Yang, Y.-X. (2018). A secure cryptocurrency scheme based on post-quantum blockchain. *IEEE Access*, *6*, 27205-27213. doi: 10.1109/ACCESS.2018.2827203

Jenkins, D. (2021, Mar). *Qubits: Ten things you should know about quantum computing.* Science Next. Retrieved from `https://lsuscienceblog.squarespace.com/blog/2021/3/1/qubits-ten-things-you-should-know-about-quantum-computing`

Mina, M., Mihai-Zicu, & Simion, E. (2021). Information security in the quantum era. threats to modern cryptography: Grover's algorithm. Cryptology ePrint Archive, Report 2021/1662. (`https://ia.cr/2021/1662`)

Quantum risk to the ethereum blockchain - a bump in the road or a brick wall? (2022, Feb). Retrieved from `https://www2.deloitte.com/nl/nl/pages/risk/articles/quantum-risk-to-the-ethereum-blockchain.html`