Various Advantages of Lattice Based Cryptography

Mark Hollis
University of Nebraska-Lincoln
Software Engineering Undergraduate

Gregory Nail
University of Nebraska-Lincoln
Software Engineering Undergraduate

Josh Martin
University of Nebraska-Lincoln
Software Engineering Undergraduate

***Abstract*—Lattice cryptography is a field of cryptography that has many advantages over some of the more fundamental cryptographic algorithms. Some of such advantages are that it is impossibly hard algorithm to break, and performs well in the worst-case assumptions. We live in a period of rapid technological development, and as certain encryption schemes become less secure we will need to invest in cryptographic systems that can not so easily be exploited. In this paper we will discuss how lattice Cryptography works and some of its advantages.**

As each day passes, technology advances and so does the progress towards trivializing the decryption of our modern cryptographic systems. With the eventuality of quantum computing becoming commercialized and their use more spread it will be critical to rely on systems that are not vulnerable to quantum attacks. Lattice cryptography has conjectured security against quantum attacks, has simpler and more efficient algorithms, strong security guarantees from worst-case hardness, and versatile and powerful cryptographic objects.

I. Introduction

In a day and age where the value of information is regarded so highly, it is crucial that there exists schemes to reliably and efficiently encrypt and abstract this information. Today some of the most ubiquitously used encryption methods are public-key schemes such as RSA, Diffie-Hellman or elliptic-curve cryptosystems. These schemes all share one fatal weakness however. They are relatively easily attacked with the use of quantum computers.

In an age of post-quantum cryptography we must turn to new solutions that will secure our information and data. With the threat of quantum computing trivializing the foundation of our modern encryption schemes we must search for new solutions. One such solution is lattice encryption. Lattice-based cryptography appears to be resistant to attacks from both classical computers as well as quantum based computation. This security gained from using a lattice based encryption scheme comes from the understanding that certain well-studied computational lattice problems cannot be solved efficiently.

II. Background

A. Quantum Computing

Before we get into the background of lattices themselves, let's discuss the reasons why other cryptosystems are vulnerable. First, we will give a background on quantum computing. Quantum computing is still a field in its adolescence. Currently quantum computers are extremely expensive to manufacture, and require large volumes of physical space. This was similarly the case with standard computers less than 80 years ago. Companies like IBM have already made quantum computational power available to the public via the cloud. It is only a matter of time before these devices are made affordable and widely available.

The reason why quantum computing is scary for modern cryptographers is because of the computing power that it provides. Let's compare it to our current infrastructure of computing. Our computers use bits, which are 0's and 1's. Each bit is either set to 0 or 1, and off or on state respectively. Now we look at quantum computing, where these quantum bits, called qbits can be set to 0, 1, or both. So, a traditional computer will try to answer problems one by one, until it reaches a solution. However with a quantum computer, using this 'superposition' described previously, by considering many processes at once, sorting through many problems at a single instant that all lead to one answer. How much faster is this, you may ask? In 2015, Google and NASA reported that Quantum Computing tried to solve a standard optimization problem that took a standard computer 10,000 years, while the quantum computer was able to find the solution in just a few seconds, which is around 100 million times faster than a standardized computer.

B. Diffie-Hellman Protocol Weaknesses

The largest weakness of the Diffie-Hellman protocol is that it fails to establish an identity check with the other party which makes it easily susceptible to man in the middle attacks where a third party can exploit this aspect of the protocol to gain information otherwise not readily available and eventually break the encryption.

Another flaw in the system in its integration, is that the key system relies on persistent keys across parties, this means that the system becomes weaker as more information has been passed across the network. This makes it easier to collect a series of data points and be able to use them to break the encryption scheme. This is why Diffie-Hellman is often weak as a long term protocol between two separate parties. The algorithm is also not quantum resistant and is able to be broken easily using Shor's algorithm using a quantum computer.

C. Elliptical Curve Weaknesses

We will begin with an overview of the cryptosystem. Elliptic curve itself is a cubic curve over a field 'K' with at least one point, 'O'. The curve cannot cross over itself, or contain any 'sharp' points, or points that make an angle. The two main mathematical functions that deal with elliptic curve are point addition, and point doubling. These functions are used to calculate a point in the curve between two points, P and Q, and find a solution at point R. Below is an example of an elliptic curve.

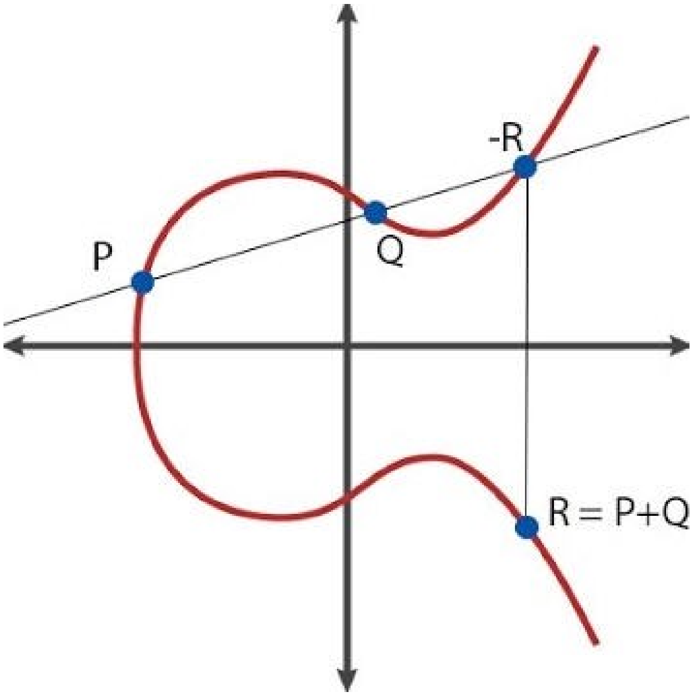

Fig. 1. Example of an elliptic-curve.

One of the most common attacks on elliptical curve cryptography are known as the side channel system attacks, and they typically occur during the implementation of the ECC, by taking measurements of the system. One of these side-channel attacks is known as a simple timing attack. Through this attack, the attacker will observe the different levels of power consumption in the system. Through the understanding of the time that it takes to encrypt certain pieces of data, the attacker will be able to deduce the secret key.

Another genre of attacks on Elliptical Curve Cryptography are known as twist-security attacks, where the attacker creates a public key that intentionally does not lie on the ellipical curve. The victim then computes a shared key, and the attacker is then able to decrypt the victim's private key.

D. Ring Learning With Errors (RLWE)

Ring learning with errors is one such computational problem that may serve as the foundation to quantum resistant cryptosystems. It can provide a basis for homomorphic encryption by utilizing a generalization of the learning with errors (LWE) problem mapped to a polynomial ring over a finite field. The presumed difficulty of solving the RWLE problem makes it a robust challenge to quantum computers and Turing machines alike. The future of Public Key encryption may rely on RLWE just as it has on integer factorization and discrete logarithm problems since the early 1980s.

The RLWE problem is built on the arithmetic of polynomials with coefficients from a finite field. A typical polynomial $a(x)$ is expressed as:
$a(x) = a_0 + a_1x + a_2x^2 + ... + a_{n-2}x^{n-2} + a_{n-1}x^{n-1}$

Polynomials may be added and multiplied as normal. In this context, coefficients of the polynomials and all operations involving the coefficients is done in a finite field. Typically the field $\mathbb{Z}$/q$\mathbb{Z}$ = F_q. When considering the set of polynomials that exist over a finite field with the operations of addition and multiplication forms an infinite polynomial ring ($F_q[x]$). The RWLE context then works within a finite quotient ring of this infinite ring.

E. RSA Weaknesses

The RSA algorithm is another one of the many algorithms that can be brute forced fairly easily through quantum computing. This attack is known as a cycle attack - taking the decrypted message, and cycling it through all possible encryptions before revealing the message eventually.

Another common attack on the RSA algorithm was shown when multiple keys were sent out, and attackers were able to simply find the common modulo. For example, if a message is encrypted and sent to more than person, simply find the common modulo between the two public keys. This is another fault in the RSA algorithm.

The last type of attack on the RSA algorithm that will be covered is a type of key manipulation, which is a use case of the 'common modulus' vulnerability that we talked about in the previous paragraph. Suppose you have an attacker who is able to see the communication between two people, Person 1 and Person 2. After one message is sent by Person 1, let's say the attacker is able to intercept the public key, and change a single bit. Person 2 encrypts a message with Person 1's public key, and sends the message back - but Person 1 will be unable to decrypt the message, given that the key is faulty. From here, Person 1 will re-send the public key, this time without attacker interference, and Person 2 sending a message back. Now the attacker is able to see the encrypted messages - one with a faulty encryption, and one properly encrypted. From here, the attacker can simply attack the Common Modulus attack to decipher the message.

F. Lattices

In the previous section it has been described that Lattice Cryptography is the only type of cryptosystem that is unbreakable by Quantum Computing. In this section we will explain why this is. A lattice is any space of equally spaced points that stretches out to infinity. Here is an example of a 2 dimensional lattice shown in figure 2.

Some of the common problems in Lattice cryptography are finding the closest two points in a lattice. This is known as the shortest vector problem (SVP), and the closest vector problem (CVP).

In SVP, we are given a basis in a vector space V, and a norm N. The norm N can typically be seen as L2, and are given a Lattice L.

The goal of the problem is to find the shortest non-zero vector inside of the Vector Space V, measured by the norm

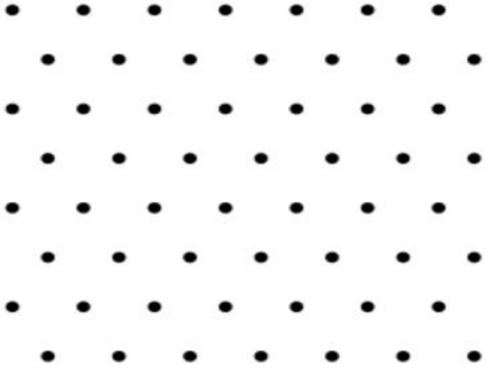

Fig. 2. Example of a two dimensional lattice.

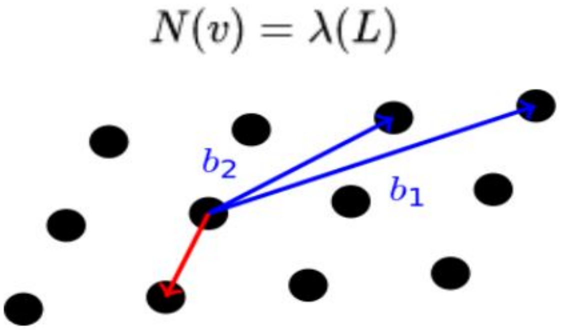

Fig. 3. Example of a Shortest Vector Problem.

N, inside of the lattice L. Solving this problem is very easy for a small basis, and does not require much time, or even a quantum computer. However when we create a very long basis, this becomes an np-hard problem at easiest. The time complexity for an SVP problem with a euclidean norm is a 2O(n)time problem. [needs explanation]

In CVP, a basis of a vector space V and a metric M (often L2) are given for a lattice L, as well as a vector v in V but not necessarily in L. It is desired to find the vector in L closest to v (as measured by M). In the gamma -approximation version CVP, one must find a lattice vector at distance at most gamma.

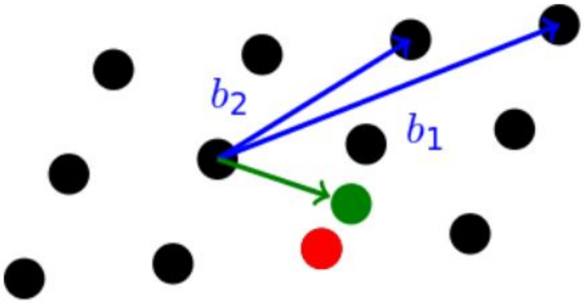

Fig. 4. Example of a Closest Vector Problem.

One popular cryptosystem that uses Lattice Based Cryptography is NTRU. Unlike the previous algorithms mentioned, NTRU is not known to be vulnerable to quantum computing attacks. In 2009, the National Institute of Standards and Technology wrote "[there] are viable alternatives for both public key encryption and signatures that are not vulnerable to Shor's Algorithm" and "[of] the various lattice based cryptographic schemes that have been developed, the NTRU family of cryptographic algorithms appears to be the most practical."

While one of the weaknesses of the lattice-based cryptography has been that it is slow to encrypt, when we look at equivalent cryptographic strength, the NTRU algorithm is able to create encryptions quadratically at the rate that RSA creates them cubically. This makes NTRU only about 20 times slower than the AES implementation, which is still extremely fast considering the time it will take to break these encryptions.

NTRU is implemented as two different algorithms: NTRU-Encrypt, which is used for encryption, and NTRUSign, which is used for digital signatures. We will only discuss NTRUEncrypt in this paper.

G. NTRU Encryption Algorithm

The NTRU algorithm is parametrized by 3 integers. N, where N is a prime number, and p and q, where the $gcd(p, q) = 1$ and p is much less than q. Let R, R_p, and R_q be polynomial rings:

$$R = \frac{\mathbb{Z}[x]}{x^N - 1}, R_p = \frac{\mathbb{Z}/p\mathbb{Z}[x]}{x^N - 1}, R_q = \frac{\mathbb{Z}/q\mathbb{Z}[x]}{x^N - 1}. \tag{1}$$

Two of the polynomials $a(x)$, $b(x) \in R$ multiplied together will result in some $c(x)$.

$$c_k = \sum_{i+j=k \pmod N} a_i b_{k-i} \tag{2}$$

We then define positive integersd_1 and d_2. $\tau(d_1, d_2)$ denote the set of ternary polynomials given by

$$\left\{ a(x) \in R| \begin{array}{l} \text{a(x) has } d_1 \text{ coefficients equal to 1,} \\ \text{a(x) has } d_2 \text{ coefficients equal to -1,} \\ \text{a(x) has all other coefficients equal to 0} \end{array} \right\} \tag{3}$$

NTRUEncrypt starts off with the generation of a public and private key. The formula for the public key h is shown below, with polynomials f and g, both with a degree of $N-1$ and with coefficients {-1, 0, 1} . The polynomial f must also be able to be an inverse of mod(q) and mod(p), such that $f * f_q = 1$ mod(q), and $f * f_p = 1 mod(p)$.

The encryption of this public key h is as follows. The sender of the message puts their message into the form of a polynomial m with coefficients {-1, 0, 1} . After the message has been converted to a polynomial, the sender chooses a

$$\mathbf{h} = p\mathbf{f}_q \cdot \mathbf{g} \pmod{q}$$

Fig. 5. NTRU Public Key Formula.

random polynomial r (which can have coefficients other than {-1, 0, 1}). This polynomial r is what makes the message difficult to decrypt.

$$\mathbf{e} = \mathbf{r} \cdot \mathbf{h} + \mathbf{m} \pmod{q}$$

Fig. 6. NTRU Encryption Algorithm.

In order to decrypt the system, we can use the following equation, in conjunction with f, the public key of the person receiving the message, and e, the encrypted message.

$$\mathbf{a} = \mathbf{f} \cdot \mathbf{e} \pmod{q}$$

Fig. 7. NTRU Decryption Algorithm.

Although the NTRU cryptosystem is vulnerable to attacks of some forms, it will be able to withstand quantum computing attacks, which sets it far above the previously described algorithms for future considerations.

The NTRU cryptosystem in its current state does have some minor flaws that would require modification in an actual implementation use case. In order to understand these flaws the concepts of indistinguishability and malleability must initially be understood.

Indistinguishability is a statement that the hardness of finding any information of an underlying message is consistent across the cipher text. If it is weak to break one point of the message text that single point can be used to gather information to break the rest.

Malleability is the ability for an adversary to produce similar encryption messages that are meaningfully related to the underlying message of a given cipher text. This means that if someone trying to break the cipher text is able to develop a scheme to understand even small relations between various texts, they may be able to use those relations to break the cipher text itself.

This is a flaw for NTRUencrypt as it is neither indistinguishable nor non-malleable. This means that not only is it susceptible to both flaws, it does not guarantee the complex hardness of standard lattice based problems.

III. Discussion

As evidenced by the large number of cryptosystems that will be put to death by the advancement of quantum computing, it is clear that lattice cryptography has a realistic chance of becoming a staple in future cryptography. In this section, we will discuss the dangers of relying on traditional cryptosystems, some of the modern areas that Lattice Cryptography has been introduced, and the future of the cryptosystem.

A. The Trivialization of RSA, ECC, Diffie-Hellman

The Internet of Things (IoT) blurs the lines between what exist in the physical and virtual worlds. These devices have changed the way we live our lives and control much of what we consider critical to survival. It is estimated that IoT technologies will have an impact of potential several trillions into the global economy by the year 2020 (Rui Xu 2). The ominous secret regarding this fact however is that with the ever-looming breakthroughs of quantum computing, the standard cryptosystems that protect these devices may soon be trivially broken and decrypted. The quantum threats to cryptography apply equally, or even to a greater extent, to smart objects extensively used in smart IoT services since they involve platforms and systems which are difficult to update (Rui Xu 3). Experts are beginning to refer to the time we live in, as well as the future as a post-quantum cryptography (PQC) era. In this PQC era is it more important than ever to begin focusing efforts on cryptosystems that will be able to secure IoT devices.

Behind the abstraction of security protocols, cryptography is used as a fundamental building block. The canonical implication of security is confidentiality, which requires that sensitive information can not be learned by unauthorized party. Modern cryptographic systems embedded within IoT devices rely on rigorous proofs for assuring security in extreme adversarial situations. The confidence that is found in well-established cryptosystems is the provable hardness of the mathematical problems that underlay the algorithms. The integer factorization problem and Elliptic Curve discrete logarithm problem are two famous problems of this kind. They are the bases for RSA, Diffie-Hellman and Elliptic Curve Cryptography (ECC), which are widely used in today's cryptography (Rui Xu). The best known classical algorithms, specifying the use of a Turing machine, for solving factorization and discrete logarithm problem work with sub-exponential time complexity. But Shor's quantum algorithm can solve both within polynomial time (Rui Xui 6). A direct consequence to this fact is that once large-scale quantum computers are readily available, our currently used public-key cryptography system such as RSA and ECC will become *completely trivialized.* Another mild yet universally influential impact of quantum computing techniques comes from Grover's algorithm which presents a quadratic speedup for search problems over classical algorithms on a Turing machine.

The current political climate within the United States, as well as around the world, is in a state of turbulence. This turbulence inevitably causes friction within social groups, and is a compelling reason for bad actors to attempt to invalidate ubiquitously used cryptosystems to secure private data and information or disrupt a groups access to certain technologies. Rui Xu et al discuss such an instance of an attack against Ukraine. On December 23, 2015 the Ukrainian Kyivoblenergo, a regional electricity distribution company reported service outages to customers. These outages were caused by a third party's illegal entry into the company's computer and SCADA systems (Robert Lee). Beginning around 3:00 p.m. local time, seven 110kV and twenty-three 35kV substations were disconnected for hours. It was found that the cyber attack impacted additional portions of the distribution grid and forced operators to switch to manual mode. The attackers demonstrated a variety of capabilities, including spear phishing emails, variants of the BlackEnergy 3 malware, and the manipulation of Microsoft Office documents that contained the malware to gain a foothold (Robert Lee).

It is clear the impact that the trivialization of the underlying problems that define our modern cryptosystems will cause on unprotected civilians. In this case the service of power to enormous amounts of an entire country was denied as a result of recovered passphrases and private data from an adversary. While this incident in global terms may seem insignificant, but it merely a window into the possibilities of what could easily occur at a much larger magnitude if a cryptosystem like RSA or ECC becomes quickly and widely invalidated before the transition to alternative cryptogrpahic systems, such as Lattice based methods, can occur.

B. Implementation Comparison

The NTRUencrypt algorithm currently stands as the most well known implementation of Lattice cryptography. There are however alternative implementations that also stand as feasible options for post-quantum security. One particularly strong candidate is the Ring learning with errors (RWLE) problem. There are various advantages and disadvantages to using both systems.

Since the initial proposal of the NTRU system several attacks have been devised. Most attacks on this system are focused on making a total break by finding the secret key instead of just recovering the message. If the key is known to have very few non-zero coefficients the attacker can successfully mount a brute force attack by trying all values for the key. It is also possible to mount a meet-in-the-middle attack against NTRU which can prove more devastating.

When looking at the RWLE problem, some advantages arise over NTRU. A major advantage of RWLE based cryprotsystems is the size of the public and private keys. For 128 bits of security an RLWE cryptographic algorithm would use public keys around 7000 bits in length. RWLE keys are significantly larger than the key sizes for currently used public key algorithms like RSA and Elliptic Curve as well. There are now several mathematical proofs that the only way to break an RWLE based cryptosystem, within some formal attack model on its random instances, is by being able to solve the underlying lattice problem in the worst case.

C. Hardware Implementations

As previously stated, the NTRUencrypt algorithm is considered the most well known implementation of Lattice cryptography. There have been a few hardware implementations of the algorithm.

Looking at the encryption the hardware implementation takes in the public key is loaded into the chip through a series of input and output or I/O pins. Similarly the plaintext is passed in with an ephermeral key on various clock cycles. One each cycle the required output is a portion of the finalized cipher text. During each cycle the chip performs the encryption algorithm and returns its finalized output. This cyclic based encryption is the bottleneck of the entire process and can be quite time consuming.

During a given cycle a series of shift, multiplication, and addition operations must be performed. These are then evaluated and the block is shifted out. The time using the hardware implementation can be quite long as encryption of a large number of plain text blocks scales immensely.

D. Modern Applications

End-to-End Encryption Many forms of modern communication are using a variety of cryptographic schemes to encrypt data being transferred among two members. This is something that can be greatly improved upon by using lattice based encryption. Email currently is one of the weaker encrypted platforms as most emails do not contain sensitive enough information to require high security. This is flawed though as some applications such as military and government require increased security among its email platforms to secure data being passed through. Using lattice cryptography we are able to securely encrypt these platforms and ensure that these private messages are left unbroken. A small leak from one of these secure platforms can lead to problems for an entire nation if important enough. Lattice cryptography opens up a whole new level of security as it is unable to be broken even via quantum computing should that become a reality.

Disk Encryption This is a method of encrypting the contents of an entire disk drive in order to allow users to not worry about leaving traces of unencrypted data on the disk for others to find. Typically this is done using a password encrypted system that algorithmically encrypts a disk allowing only one user to have access to the disk. This can be improved as typically now it is not very difficult on most systems to access other users data on the same disk if sufficient compute power exists. This is where lattice cryptography can improve this system. It is unlikely that there will be a point where a user without sufficient information will have strong enough computational resources to break the encryption.

E. Versatility

Lattice-based Cryptography is special not only because of it's security, but also because it offers us an incredibly versatile range of cryptographic schemes that we can build with it. Currently, we haven't begun to discover all of the ways that we can use this cryptosystem. Lattice Cryptography will introduce an entirely new class of cryptosystems and problems to solve, many of which we haven't thought of yet, and all of these problems will be immune to the power of quantum computing due to their complexity, much more complicated than the problems discussed earlier, SVP and CVP.

F. Post Quantum Computing

As the computational world of technology grows and the reality of quantum computing becomes more and more possible as the days pass, the need for a quantum safe cryptography solutions is growing more and more prevalent. Lattice cryptography currently stands to fill this need, but at a bit of a cost. The algorithm is quite complex and is difficult to efficiently perform even knowing the implementation, in its current state.

This introduces a unique problem though. While in our current tech standards lattice cryptography may not provide enough benefit to outweigh its cost, when quantum computing becomes a reality in a feasible manner, it will be quite difficult to develop cryptography schemes that are safe, but are not complex such as lattice cryptographer, just purely due to the power of a quantum computer. This introduces a unique problem. As technology grows more and more the complexity of these systems will also continue to grow, and the more complex the system, the more computationally expensive the implementation.

Lattice cryptography as it currently stands is incredibly strong, and in some cases is optimal for security, such as government and corporate applications, but for standard day to day device encryption, Lattice cryptography is too expensive, and complex to really see a need.

For most use cases the traditional crypto schemes seem to be good enough, most of the data users are protecting is not sensitive enough nor valuable enough to require such costly encryption and decryption. While optimizations ultimately will be developed, in its current state Lattice cryptography stands to be more of theory than a reality in our daily lives.

G. Future Considerations

The future of Lattice-based Cryptography will cover a wide variety of fields. Most notably, we will see it embedded into our devices, such as our phones and laptop hard drives, in which there are already being developments. Safe fields of communication have become a necessity when sending over sensitive data. And, as of now, Lattice-based cryptography accounts for 40% of the submissions to the NIST post-quantum standardization effort. This means that nearly half of the encryptions developed in order to withstand quantum computing derive from the field of Lattice-based cryptography, and a main reason for this is because Lattice-based cryptography is fairly simple for how difficult the problems are to solve, which goes in line with why the field is seen as so versatile.

IV. Conclusion

Lattice-based Cryptography seems to be where the field of cryptography is headed, due to it's worst-case hardness, versatility, and ability to withstand quantum computing realization. If quantum computing becomes a reality, many of the current cryptosystems that are used, if not all, will be easily brute-forced by the immense computing power of a quantum computer, and this will leave the field of cryptography looking for a secure way to encrypt data. Lattice-based Cryptography is very versatile, and it is this versatility that leaves many cryptographers believing that the field of Lattice-based Cryptography is going to create encryption problems so difficult that they will never be solved, thus keeping our data secure for many years to come.

V. References

- Xu, Rui, et al. “Lighting the Way to a Smart World: Lattice-Based Cryptography for Internet of Things.” Cornell University, Https://Arxiv.org/, 2019, pp. 1–7.
- Lee, Robert M, et al. “Analysis of the Cyber Attack on the Ukrainian Power Grid.” Electricity Information Sharing and Analysis Center, 18 Mar. 2016.
- Abdel Alim Kamal; Amr M. Youssef, et al. “An FPGA implementation of the NTRUEncrypt cryptosystem” IEEE, 2009.