Graph Theory-Envisioned New Directions in Multivariate Based Cryptography and Its Application to IoT Technology

A dissertation Submitted

in partial fulfillment of the requirements for the award of the degree of

Master of Science

in

Mathematics

by

PRABHAT KUMAR

Reg. No. 2020PGMHMH13

Under the Supervision of

Dr. Sumit Kumar Debnath

DEPARTMENT OF MATHEMATICS

NATIONAL INSTITUTE OF TECHNOLOGY JAMSHEDPUR

JAMSHEDPUR - 831014, INDIA

MAY, 2022

Declaration

I hereby declare that the work, which is being presented in this dissertation entitled **"Graph Theory-Envisioned New Directions in Multivariate Based Cryptography and Its Application to IoT Technology"**, in partial fulfilment of the requirement for the award of degree of **Master of Science in Mathematics** submitted in the **Department of Mathematics, National Institute of Technology Jamshedpur**. This record is an authentic record of my own work carried out from January, 2022 under the supervision of **Dr. Sumit Kumar Debnath, Assistant Professor, Department of Mathematics, National Institute of Technology Jamshedpur**. The matter embodied in this report has not been submitted by me for the award of any other degree in this Institute or any other Institute/University.

Date:
Place: NIT Jamshedpur

Prabhat Kumar
M.Sc. in Mathematics (Final year)
Department of Mathematics
NIT Jamshedpur

Certificate

This is to certify that the dissertation entitled **Graph Theory-Envisioned New Directions in Multivariate Based Cryptography and Its Application to IoT Technology**, submitted by **Prabhat Kumar** to the National Institute of Technology Jamshedpur, for the award of the degree of Master of Science in Mathematics is a record of bonafide project work carried under my supervision and guidance. To the best of my knowledge, the work in the thesis has not been submitted in part or full to any other Institution and University for the award of any degree or diploma. I consider it worthy of consideration for the award of the degree of Master of Science in Mathematics.

Date:

Place: NIT Jamshedpur

Dr. Sunil Kumar
Head of the Department
Department of Mathematics
NIT Jamshedpur

Dr. Sumit Kumar Debnath
Supervisor
Department of Mathematics
NIT Jamshedpur

Acknowledgements

Today I am standing at the stage where I can easily say that my project work has been completed very well. But, it was not easy as it seems to be. I had to work with a lot of hard work, dedication, and perseverance to achieve my goal. But, if I say that I have attained the objectives of the project alone, then it will be a total lie. My mentor and supervisor Dr. Sumit Kumar Debnath Sir, and Ph.D. scholar Vikas Srivastava Sir have played a big role in completing this task. During this project, whenever I faced difficulties and when all the paths ahead were blurred, I found Sumit Sir and Vikas Sir guiding me to the right path with the torch of knowledge. Any amount of thank you would not be enough to express my gratitude towards them.

But still I want to thank my supervisor Dr. Sumit Kumar Debnath Sir, and Vikas Srivastava Sir, for helping me achieve this task by taking time in the midst of their busy work schedule and guiding me through this project work. I am grateful to them from the deepest core of my heart.

My entire family also has a big role in the successful completion of this task. Seeing the hope they placed on me inspired me, and I was filled with energy, which has made my project work so well.

I would also like to thank some of my friends - Roshan, Abhishek, and Kushagra, who always kept me motivated and helped me directly and indirectly in completing this work.

Finally, I would like to thank all the students of my class and all the faculty members who helped me make this project work a success.

Prabhat Kumar

M.Sc in Mathematics

Abstract

The security of almost all of the currently used cryptographic primitives are under a serious threat due to attack by quantum algorithms. When large scale computers will become a reality, the entire communication and information network will become insecure and useless. Thus, there is a need of new generation of algorithms which are resistant to attack by quantum computers. This new direction of research is called post-quantum cryptography. Multivariate public key cryptography (MPKC) is one such candidate. MPKC is based on the hardness of MQ-problem. A system of multivariate polynomials is used as the public key of the MPKC. In this thesis, we took an effort to explore new directions in the context of multivariate based cryptography. As a first step towards my contribution to the project, I studied in deep about public key cryptography and graph theory. In the following, we learned about post-quantum cryptography and multivariate public key cryptography. As a main contribution towards this project, we explored and analyzed the connection between graph theory and multivariate public key cryptography. We found a novel method to construct a system of multivariate polynomial equations from the theory of graphs. In the end, we tried tried to design an authentication protocol based on the hardness assumption of graph isomorphism problem and MQ problem. We also noted down the possible application of the authentication protocol in IoT Technology.

Table of Contents

Chapter 1

Introduction and Objective

Peter Shor devised a method in 1994 that allows traditional cryptographic techniques like RSA and ECC to be defeated. The standard and long-established number theoretic hardness assumptions such as prime factorization and discrete logarithm are used to secure classical cryptographic primitives. Almost all of the currently used classical cryptography systems can be cracked in polynomial time using Shor's technique. It poses a serious danger to the current information system's security. Furthermore, it raises concerns about the security and integrity of current digital communication networks.

According to recent research, large-scale quantum computers will be a reality within a few decades. Even though the exact date of the advent of large quantum computers cannot be predicted, cryptographic building blocks are required to give resistance to this entirely new class of attacks. This looming danger has prompted academics all around the world to create the next generation of cryptography designs and systems that are resistant to quantum computer attacks. Post-quantum cryptography [1, 2] is the name given to this emerging study area.

Although there are multiple competitors for post-quantum cryptography, multivariate public key cryptography (MPKC) continues to lead the race. It's primarily owing to the fact that MPKC schemes in general are extremely quick, dependable, and need very little CPU power to implement. They are efficient, using simply modular field multiplications and additions as mathematical operations. The hardness of the MQ gives the theoretical security of MPKC. Even for tiny field GF(2), the

MQ problem is NP-hard.

In this project, we are aiming to explore new directions in the context of multivariate based cryptography. We want to analyze and explore the connection between graph theory and multivariate public key cryptography.

Chapter 2

Multivariate Public Key Cryptography

2.1 Multivariate Public Key Cryptography [3, 4]

In this section, we revisit some of the basic facts about MPKC. The basic objects of multivariate public-key cryptosystem are system of multivariate quadratic polynomial equations over a finite field $\mathbb{F}_q$. Let $\mathbb{F}_q$ denote the finite field of order q. A multivariate quadratic polynomial in n variables $z_1, \ldots z_n$ is of the form

$$f(\mathbf{z}) = \sum_{i,j} a_{ij} z_i z_j + \sum_i b_i z_i + c$$

with n-tuple $(z_1, \ldots, z_n)$ denoted by $\mathbf{z}$. a_{ij}, b_i and c belongs to the finite field $\mathbb{F}_q$.

The underlying idea and concept behind the design of MPKC is to select a system $\mathcal{W} : \mathbb{F}_q^n \to \mathbb{F}_q^m$ of m multivariate polynomials of degree two in n variables. We stipulate that this map $\mathcal{W}$, also known as central map, is easily inverted in the sense that finding preimage of y under $\mathcal{W}$ is easy. To obfuscate the structure of $\mathcal{W}$, we pick two affine invertible transformations $\mathcal{C}_1 : \mathbb{F}_q^m \to \mathbb{F}_q^m$ and $\mathcal{C}_2 : \mathbb{F}_q^n \to \mathbb{F}_q^n$. To find the public key of the cryptosystem, we move on by taking the composed map $\mathcal{X} = \mathcal{C}_1 \circ \mathcal{W} \circ \mathcal{C}_2 : \mathbb{F}_q^n \to \mathbb{F}_q^m$. The secret key of the MPKC is a three tuple $(\mathcal{C}_1, \mathcal{W}, \mathcal{C}_2)$.

2.2 Hardness assumption [7]

The theoretical security of a multivariate scheme relies on MQ problem - a NP hard problem from the field of algebraic geometry. It is mathematically formulated as

Definition 2.2.1. Given a system $\mathcal{Q} = (q_{(1)}(\phi_1, \ldots, \phi_n), \ldots, q_{(m)}(\phi_1, \ldots, \phi_n))$ of m quadratic equations in variables $(\phi_1, \ldots, \phi_n)$, find a n tuple $(\bar{\phi}_1, \ldots, \bar{\phi}_n)$ such that

$$q_{(1)}(\bar{\phi}_1, \ldots, \bar{\phi}_n) = \cdots = q_{(m)}(\bar{\phi}_1, \ldots, \bar{\phi}_n) = 0.$$

All the variables along with coefficients belong to $\mathbb{F}_q$.

Chapter 3

Graph Theory

In this chapter, we will revisit some of the basic terminology used throughout the thesis. We are following [12–14]

3.1 Bipartition and Isomorphism of Graphs

Definition 3.1.1. The graphs G_1=(V_1,E_1) and G_2=(V_2,E_2) are isomorphic if there exists a one-to-one and onto function f from V_1 to V_2 with the property that a and b are adjacent in G_1 if and only if $f(a)$ and $f(b)$ are adjacent in G_2 , for all a and b in V_1. Such a function f is called an isomorphism.

Figure 3.1: Two isomorphic graphs with ismoprhism given by $f(1) = a, f(2) = b, f(3) = c$, and $f(4) = d$

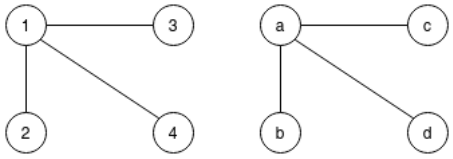


Figure 3.2: Two graphs which are non isomorphic to each other

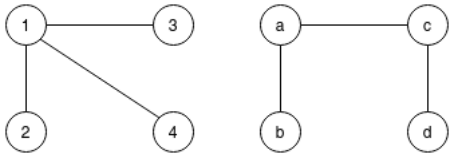
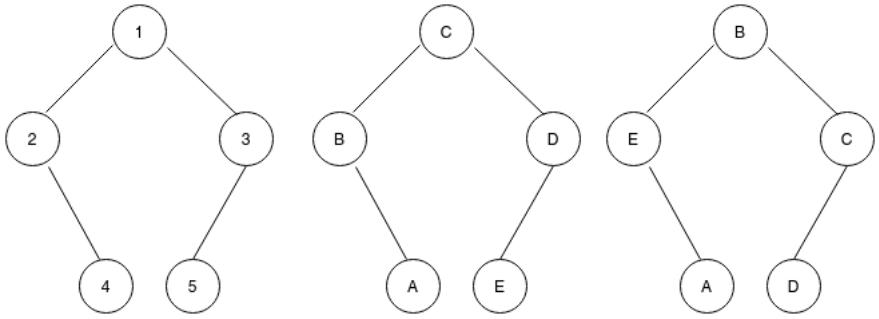
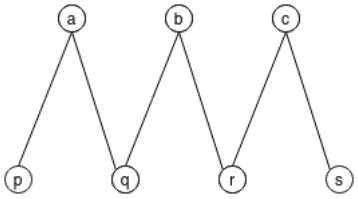


Figure 3.3: A relabeling of vertices of a graph is isomorphic to the graph itself. Consider the three isomorphic graphs illustrated below. The first two graphs illustrate a change of using letters to using numbers to label the graphs. The second pair of graphs are also isomorphic as only the labels were changed. We can match vertices in the second graph with those in the third graph to satisfy the isomorphism requirements.



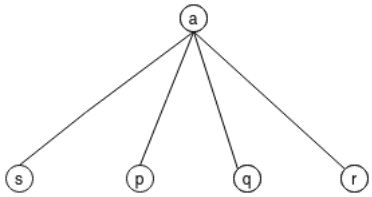
Definition 3.1.2. A simple graph G is called bipartite if its vertex set V can be partitioned into two disjoint sets V_1 and $V2$ such that every edge in the graph connects a vertex in V_1 and a vertex in V_2. When this condition holds,we call the pair (V_1,V_2) a bipartition of the vertex set V of G.

Figure 3.4: An example of bipartite graph



Definition 3.1.3. A complete bipartite graph $K_{m,n}$ is a graph that has its vertex set partitioned into two subsets of m and n vertices respectively with an edge between two vertices if and only if one vertex is in the first subset and the other vertex is in the second subset.

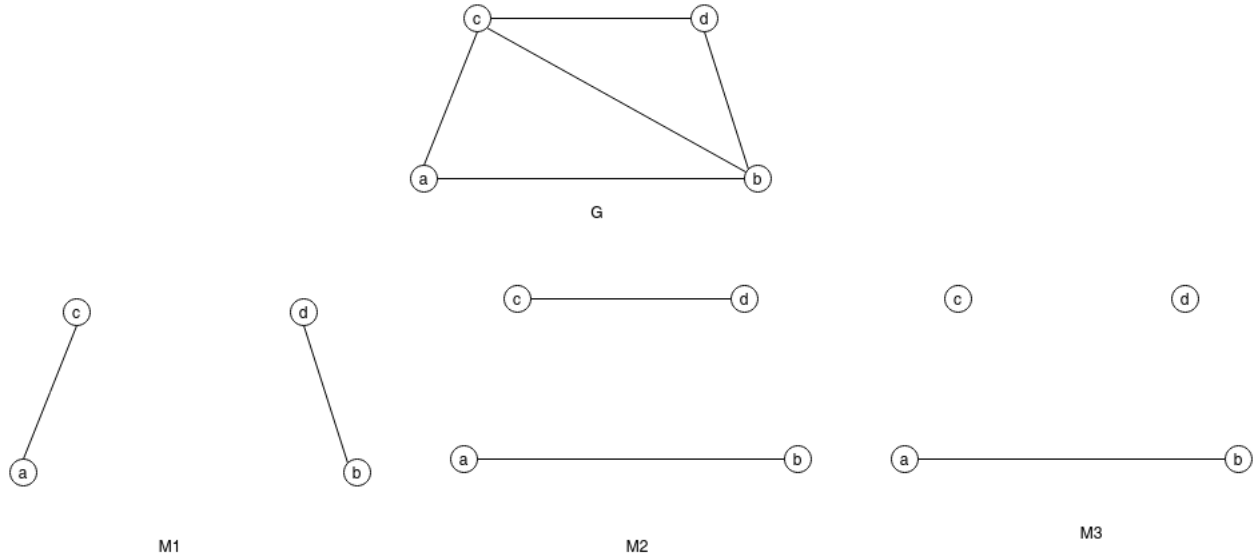
Figure 3.5: An example of complete bipartite graph



3.2 Matching

In graph theory, a matching in a graph is a set of edges that do not have a set of common vertices. In other words, a matching is a graph where each node has either zero or one edge incident to it. A matching is *perfect* if every vertex of the graph is incident to an edge of the matching.

Figure 3.6: An illustrative figure showing the concept of matching and perfect matching. Here M_1, M_2 and M_3 are examples of matching. M_1 and M_2 are perfect matching while M_3 is not an example of perfect matching.



3.3 Graph Isomorphism (GI) Problem

Recall that given a pair of graphs $G = (U, D)$ and $H = (V, E)$, a bijection $\phi : U ß V$ that preserves edges is an isomorphism between G and H. If such a bijection exists between G and H, they are said to be isomorphic. Thus, we can define the GI problem as the task of finding one of the possibly many isomorphisms between G and H or deciding that this bijection does not exist. The problem is not known to be solvable in polynomial time [10].

Chapter 4

Zero Knowledge Proofs

4.1 Zero Knowledge Proof

A zero-knowledge proof [8, 9, 11] is a unique method where a user can prove to another user that he/she knows an absolute value, without actually conveying any extra information. Here, the prover could prove that he knows the value z to the verifier without giving him any information other than the fact that he knows the value z. The main essence behind this concept is to prove possession of knowledge without revealing it. The primary challenge here is to show that you know a value z without saying what z is or any other info.

4.2 Example of Zero Knowledge Proof

Let p be a large prime, and suppose you choose an x at random which will be your secret ID number. Now choose a generator A and compute $B = A^x(\mod p)$. You can safely publish A, B, and p because an eavesdropper cannot compute x from that data if discrete log is hard. Now you want to prove that you have the knowledge of x without actually revealing any information about x.

1. Prover (you) chooses a random number $0 \leq r < p - 1$ and sends the verifier $h = A^r(\mod p)$.

2. Verifier sends back a random bit b.

3. Prover sends $s = (r + bx)(\mod (p-1))$ to verifier.

4. Verifier computes $A^s(\mod p)$ which should equal $hB^b(\mod p)$.

4.3 Discussion

The basic idea here is that if $b = 1$, the prover gives a number to the verifier (V) that looks random ($s = r + x(\mod (p-1))$). But V already knows $h = A^r$ and $B = A^x$ and can multiply these and compare them to A^s. We should be careful what is proved by that. What V actually sees are h and s, and so what V knows is that $s = dlog(h) + x(\mod (p-1))$, where $dlog(h)$ is the discrete log of h relative to A. The verifier knows s and so do you, the prover. Now if you also know $dlog(h)$, then its clear that you know x. So it remains for you to convince the verifier that you know $dlog(h)$. That's where the random bit comes in. If $b = 0$, you the prover just send $s = r$ back to V. V then checks that $h = A^r(\mod p)$, i.e. that r is the discrete log of h. So depending on the random bit, V gets either s or r but never both (because their difference is x). Thus V gets no information about x.

4.4 Properties of Zero Knowledge Proof

A zero-knowledge proof of some statement must satisfy three properties:

Completeness: if the statement is true, the honest verifier (that is, one following the protocol properly) will be convinced of this fact by an honest prover.

Soundness: if the statement is false, no cheating prover can convince the honest verifier that it is true, except with some small probability.

Zero-knowledge: if the statement is true, no verifier learns anything other than the fact that the statement is true.

4.5 Soundness/Knowledge Error

In the previous subsection, we saw the definition of soundness which says that if the statement is false, no cheating prover can convince the honest verifier that it is

true, ***except with some small probability.*** This probability of cheating is also known as knowledge error.

Theorem 4.5.1. *The knowledge error in the above given example in Section 4.2 is* $\frac{1}{2}$

Proof. **Case I** : A prover can try to cheat in one of two ways. If you dont know x, you can still pick a random r and send $h = A^r(\mod p)$ to V at the first step. If V picks $b = 0$, you are OK, because you can just send $s = r$ at step 3, and V will be able to check that $A^s = h(\mod p)$. But if V picks $b = 1$ the prover is stuck because you dont know x, and you cant easily compute an s that will satisfy $A^s = hB(\mod p)$ because that would be equivalent to finding the discrete log of hB.

Case II : On the other hand, you the prover might cheat by sending V a h whose discrete log dont know at step 1. A good candidate is $h = A^s B^{-1}$ for some random s. If the verifier picks $b = 1$, you send this s and it will satisfy $A^s = hB^b(\mod p)$. But if the verifier picks $b = 0$, we are stuck because we dont know an r such that $A^r = h(\mod p)$.

In either case, the verifier will discover that you cheated with 50% probability.

□

In order to reduce the knowledge error, we run the protocol k times so that the cheating probability reduces to $(\frac{1}{2})^k$.

Chapter 5

Constructing Multivariate Polynomial Sets From Graphs

In this chapter, we will see how to construct the multivariate polynomial sets from the graphs.

5.1 Construction

Let H_1 and H_2 be two isomorphic graphs of size e and order n with vertex sets

$$U = \{u_1, \ldots, u_n\}$$

and

$$V = \{v_1, \ldots, v_n\}$$

and edge sets D and E respectively. Let $K_{U,V}$ denote the complete bipartite graph with bipartition U, V. We get a perfect matching M in $K_{U,V}$ by selecting $u_i v_k, u_j v_l$ into M if and only if and only if $u_i u_j \in D$ and $v_k v_l \in E$.

Matching induces a bijection : We can identify any perfect matching M built in this way with a bijection ϕ that defines the isomorphism of graphs. From a set-theoretic point of view, ϕ is treated as a collection of pairs being their first coordinate elements that belong to the domain of the function, while the second ones belong

to the co-domain

From graph theory to multivariate polynomials : To generate the polynomial system, a user executes the following steps.

1. First we will consider the set of n^2 variables $\{X_{i,k}\}$ for $i, k = 1, \ldots, n$. We restrict any possible solution to the binary set $\{0, 1\}$ by introducing the following polynomials:
$$X_{i,k}^2 - X_{i,k} = 1$$

2. Now, the following polynomials are introduced to require that one and only one vertex v_i from U is connected to one vertex of V and vice versa. This links solutions to the fact that we have a perfect matching in M.
$$\sum_{i=1}^{n} X_{i,k} - 1 \; k = 1, \ldots n$$
$$\sum_{k=1}^{n} X_{i,k} - 1 \; i = 1, \ldots n$$

3. Finally, to ensure that the set of polynomials has a solution related to the chosen isomorphism, we introduce a third set of polynomials:
$$X_{i,k} X_{j,l}$$
for any i, j, k, l satisfying $(u_i u_j \in D \wedge v_k v_l \notin E) \vee (u_i u_j \notin D \wedge v_k v_l \in E)$.

This completes the construction of the polynomial set related to the given GI instance.

5.2 Toy Example

Let us consider the graph $G = (U, D)$ with $U = \{1, 2, 3, 4\}$, $D = \{(1, 2), (1, 4), (2, 3), (3, 4)\}$. Take $\sigma \in S_4$ to be $\sigma = (23)$. By applying σ to the vertex set U we get the isomorphic graph $H = (V, E)$ where $V = U$ and $E = \{(1, 3), (1, 4), (2, 3), (2, 4)\}$. We

start by building the polynomial set by fulfilling first condition, which appends 16 polynomials:

$$X_{i,j}^2 - X_{i,j} = \text{ for } \quad i,j \in \{1,2,3,4\}.$$

Subsequently, second condition is addressed by considering the polynomials

$$X_{i,1} + X_{i,2} + X_{i,3} + X_{i,4} - 1 \text{ for } i = 1,2,3,4$$

$$X_{1,j} + X_{2,j} + X_{3,j} + X_{4,j} - 1 \text{ for } j = 1,2,3,4.$$

To build the polynomial set, we start with the edges in G and $\bar{H}$. For instance, considering $(1,2) \in G$ and $(3,4) \in H$, we get the polynomials $X_{1,2}X_{3,4}$. Once we walk over all the edges of G in this fashion, we get the polynomials $X_{1,1}X_{2,2}, X_{1,1}X_{4,2}$, $X_{2,1}X_{3,2}$, $X_{3,1}X_{4,2}$ $X_{1,3}X_{2,4}$, $X_{1,3}X_{4,4}$, $X_{2,3}X_{3,4}$; $X_{3,3}X_{4,4}$. Now, by considering the edges $\bar{G}$ and H, we get another set of 8 polynomials: $X_{1,1}X_{4,3}, X_{1,1}X_{4,2}$, $X_{2,1}X_{3,2}, X_{3,1}X_{4,2}, X_{2,1}X_{3,3}, X_{2,3}X_{3,4}, X_{2,3}X_{3,4}; X_{3,3}X_{4,4}$.

Chapter 6

Authentication Protocol Based On Hardness of GI and MQ

Authentication is the process of determining if the person or entity accessing a computing system really is who they claim to be. Authentication systems make a binary decision. They allow or deny access based on credentials or other proof provided by those requesting access. Authentication typically works together with authorization systems, which determine what type or level of access a user should have.

The task of the authentication protocol is to specify the exact series of steps needed for execution of the authentication. It has to comply with the main protocol principles:

1. A Protocol has to involve two or more parties and everyone involved in the protocol must know the protocol in advance.
2. All the included parties have to follow the protocol.
3. A protocol has to be unambiguous - each step must be defined precisely.
4. A protocol must be complete - must include a specified action for every possible situation.

6.1 Authentication Protocol Based on GI and MQ

In this section, we present a possible authentication protocol based on the ideas of Zero knowledge proof developed in Chapter 4. The following steps are performed between Alice (the prover) and Bob (the verifier):

Key Generation :Alice picks a graph G and randomly generates a permutation of the set $\{1, \ldots, n\}$. This permutation is used to create the isomorphic graph H together with its isomorphism ψ. Then the public key F_1 is computed (See Chapter 5). The private key x_1 is a solution to the public system F_1.

Authentication: The following steps are executed between prover and the verifier for the authentication

1. Alice generates a permutation σ for the set $\{1, \ldots, n\}$ at random and computes the polynomial system F_2, which is sent to Bob.
2. Bob creates a challenge by selecting at random $b \in \{0, 1\}$. Bob sends b to Alice.
3. Once Alice has received b she must answer accordingly:
   (a) if $b = 0$, she sends the solution x_2 of the system F_2 to Bob,
   (b) if $b = 1$, she sends σ.
4. According to the value of b Bob performs the following to authenticate Alice:
   (a) if $b = 0$, he checks whether x_2 is a solution for F_2 or not,
   (b) if $b = 1$, he computes the system $F_2^{'}$ applying σ to F_1 and checks if he obtains the system F_2.

6.2 Discussion

Completeness Completeness property means that if the statement is true, the honest verifier (that is, one following the protocol properly) will be convinced of this fact by an honest prover. It is straightforward to see that if all the steps are followed correctly, the protocol always ends.

Soundness This property means that if the statement is false, no cheating prover can convince the honest verifier that it is true, *except with some small probability*. This small probability is also known as knowledge error.

Theorem 6.2.1. *The soundness/knowledge error of our authentication protocol is* $\frac{1}{2}$.

Proof. We will consider that a malicious entity (Eve) wants to play the role of Alice. Then she can try the following strategy. Eve flips a coin to decide which value b will send Bob as a challenge. If the result is $b = 0$, then she randomly generates a system $F_2^{'}$ with a known solution for her. Then Eve sends the system $F_2^{'}$ and waits for the challenge. If Bob selects $b = 0$ the Eve is able to provide an answer to the challenge. Otherwise, if $b = 1$ she will fail to provide the permutation. Now if the result of the flip is $b = 1$, the she selects a permutation at random to transform the system F_1 into $F_2^{'}$. Now she will have the answer for the challenge if Bob chooses to send $b = 1$, but she fails if this is not the case.

□

Zero Knowledge This property means that if the statement is true, no verifier learns anything other than the fact that the statement is true. We can see that during the execution of steps no knowledge about x_1 is leaked. Due to shortage of time, we are leaving formal proof as a future work.

6.3 Use of GI Problem and MQ Problem in Our Authentication Protocol

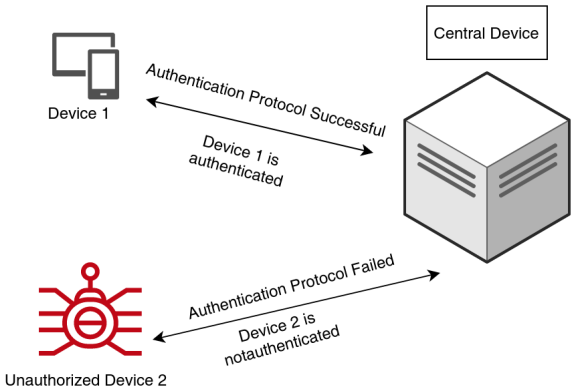
- MQ : An efficient polynomial system solver to find a solution for F_1 would break completely the scheme by exhibiting the private key.
- Solving the GI Problem. For this approach we need to retrieve the initial isomorphic graphs from the polynomial set and find an isomorphism.

Chapter 7

Applications to Internet of Things [5, 6]

In recent years, Internet of Things(IoT) gained popularity due to its enormous applications in many fields. IoT network comprises heterogeneous devices to a great scale, which creates numerous security threats. More often than not, in an IoT Network, it is required for a central IoT device to validate the peer devices. In the context of this problem, the design presented in this article could be used. Strong IoT device authentication is required to ensure connected devices on the IoT can be trusted to be what they purport to be. Consequently, each IoT device needs a unique identity that can be authenticated when the device attempts to connect to a gateway or central server. The Authentication Protocol of Chapter 6 can be used as a building block in an IoT network to provide a mechanism of authentication.

Figure 7.1: Application of authentication protocol in IoT



Conclusions and Future Directions

- We were not able to do any rigorous security analysis, so our first task would be to do a proper security analysis of the design.
- We would also like to do a performance analysis and calculate the length of keys required for achieving say 128 bit security level.
- We aim to move from GI problem to a more hard problem known as Subgraph Isomorphism Problem.
- Our goal is to read about Fiat Shamir Technique and understand whether this zero knowledge protocol can be extended to form signatures.
- We would like to analyze the nature of public polynomial constructed from Graph Theory.

Bibliography

[1] Bernstein, D. J., and Lange, T. Post-quantum cryptography. *Nature 549*, 7671 (2017), 188–194.

[2] Chen, L., Chen, L., Jordan, S., Liu, Y.-K., Moody, D., Peralta, R., Perlner, R., and Smith-Tone, D. *Report on post-quantum cryptography*, vol. 12. US Department of Commerce, National Institute of Standards and Technology . . . , 2016.

[3] Ding, J., Gower, J. E., and Schmidt, D. S. *Multivariate public key cryptosystems*, vol. 25. Springer, 2006.

[4] Ding, J., and Yang, B.-Y. Multivariate public key cryptography. In *Post-quantum cryptography*. Springer, 2009, pp. 193–241.

[5] El-Hajj, M., Chamoun, M., Fadlallah, A., and Serhrouchni, A. Analysis of authentication techniques in internet of things (iot). In *2017 1st Cyber Security in Networking Conference (CSNet)* (2017), IEEE, pp. 1–3.

[6] El-Hajj, M., Fadlallah, A., Chamoun, M., and Serhrouchni, A. A survey of internet of things (iot) authentication schemes. *Sensors 19*, 5 (2019), 1141.

[7] Garey, M. R., and Johnson, D. S. *Computers and intractability*, vol. 174. freeman San Francisco, 1979.

[8] Goldreich, O., and Krawczyk, H. On the composition of zero-knowledge proof systems. *SIAM Journal on Computing 25*, 1 (1996), 169–192.

[9] Goldreich, O., and Oren, Y. Definitions and properties of zero-knowledge proof systems. *Journal of Cryptology 7*, 1 (1994), 1–32.

[10] Grohe, M., and Schweitzer, P. The graph isomorphism problem. *Communications of the ACM 63*, 11 (2020), 128–134.

[11] Krantz, S. G. Zero knowledge proofs. *Expeditions in Mathematics 68* (2011), 249.

[12] Rahman, M. S., et al. *Basic graph theory*, vol. 9. Springer, 2017.

[13] Rosen, K. H., and Krithivasan, K. *Discrete mathematics and its applications: with combinatorics and graph theory.* Tata McGraw-Hill Education, 2012.

[14] West, D. B., et al. *Introduction to graph theory*, vol. 2. Prentice hall Upper Saddle River, 2001.