Geo Security using GPT Cryptosystem

Eraj Khan[1], Abbas Khalid[2], Arshad Ali[3,*], Muhammad Atif[4], Ahmad Salman Khan[5]
Department of Computer Science & Information Technology, The University Of Lahore, Lahore, 55150, Pakistan[1, 2, 3, 4]
Department of Software Engineering, The University of Lahore, Lahore, 55150, Pakistan[5]

***Abstract*—This paper describes an implementation of location-based encryption using a public key cryptosystem based on the rank error correcting codes. In any code based cryptosystem, public and private keys are in the form of matrices based over the finite field. This work proposes an algorithm for calculating public and private key matrices based on the geographic location of the intended receiver. The main idea is to calculate a location specific parity check matrix and then corresponding public key. Data is encrypted using public key. Some information about the parity check matrix along with other private keys are sent to receiver as cipher-text, encrypted with another instance of the public or GPT cryptosystem using public key of the receiver. The proposed scheme also introduces a method of calculating different parity check matrix for each user.**

Keywords*—*Location based security; code based cryptosystem; cipher-text; GPT

I. Introduction

Companies all over the world are extending their business models and reaching out to the consumers across the globe. Digital content distribution has overtaken physical format to become the dominant stream for generating revenue. Meanwhile, proliferation of global network interconnections along with ultrahigh density storage devices have made millions of documents online. Although it has made knowledge sharing easy and efficient but this large storage of digital contents has presented unique challenges. Now, the chances of information theft have increased than ever before. Therefore protection of confidentiality, privacy and integrity of information from unauthorised access has become significant challenge for researchers. Encryption provides a way to protect integrity and confidentiality of data which ensures that data is protected from unauthorised access. Traditional cryptographic algorithms provide assurance that only the intended users can access the encrypted data. It is still useful to have an extra layer of security on top of the existing encryption that guarantees that the authorized user can only access the contents at the specific location. It provides information protection against an authorised user who is not at authorised location. If an authorised user tries to decrypt the cipher text at an unauthorised location such as airports, train stations and other public places, the decryption should fail. It can be achieved, by combining decryption key with the location of intended recipient. The idea of combining location of intended recipient with encryption and decryption process was first introduced in [1]. In this paper authors have proposed a geolocking mechanism to be used with traditional cryptographic algorithms. There is a wide range of cryptographic algorithms available which are based on different mathematical problems. Most popular public key cryptosystems are either based on hardness of factorization of large integers (RSA) or on finding discrete logarithms over various groups (ElGamal). Although these algorithms are still considered secure if used with recommended key size and other parameters but after the seminal paper of Peter Shor [2], algorithms based on these problems are known to be broken. In [2], author provided efficient randomized algorithms for solving these problems on hypothetical quantum computer with small probability of errors. Code-based cryptography is a strong candidate for post quantum security algorithms along with hash-based and lattice-based cryptographic algorithms [3]. It is based on that mathematical which can withstand an attack by the adversary equipped with quantum computer [4].

First code-based public key cryptosystem was proposed by Robert McEliece in 1978 [5]. The cryptosystem proposed by McEliece was based on the hardness of decoding a general linear code. In a linear binary code, the problem of finding a codeword is NP-complete. Although it was a very strong algorithm but due to its large and impractical key size which was 219 bits, it didn’t gain much of attention. In 1986, Herald Niederreiter [6] proposed another code-based public key cryptosystem. The proposed cryptosystem used the scrambled version of the parity check matrix H as the public key. Due to use of parity check matrix as public key the key size is reduced from 219 to 218. Both of these cryptosystems were based on Hamming metric for calculating code lengths. In 1991, Gabidulin, Paramanov and Tretjakov (GPT) [7] proposed that if rank metric is used instead of Hamming metric, then key size of the code based public key cryptosystem can be reduced further. Based on this idea they proposed another cryptosystem based on rank codes called GPT cryptosystem. Use of rank metric instead of Hamming metric provided two advantages to the GPT cryptosystem. First it has reduced the key size to 214. Secondly as compared to the cryptosystems proposed in [5] and [6] the GPT cryptosystem is much stronger against decoding attacks. As rank codes are well structured and due to this property, over the years several attacks have been launched against GPT cryptosystem. Initially there were series of attacks on the GPT cryptosystem are published in [8-11]. To defend against these attacks several variants of GPT cryptosystems are proposed as well [12-15]. There were some recent attacks on the GPT cryptosystem published in [16-18] but to withstand these attack recently another construction of GPT cryptosystem is proposed by Loidreau P. [19]. Although GPT cryptosystem is continuously under threats over the years. However, it gained so much popularity that it is still considered as a credible post-quantum alternative to traditional cryptography [20]. Various encryption approaches are discussed by research community [21-24].

*Corresponding Author.

This work proposed a technique for implementing geo encryption using a GPT public key cryptosystem based on rank error correcting codes. Variant of GPT cryptosystem proposed in [19] is considered in this work because it withstands all the attacks published against the system so far. As GPT cryptosystem is a code based cryptosystem therefore both public key and private key are in the matrix form. In this paper, a technique for calculating the public key and private key for GPT cryptosystem based on the receiver location is presented.

The rest of the paper is organized as follow: Section II provides related work. It consists of two parts. In first part geo encryption is discussed whereas in second part background information about rank codes is provided. The proposed scheme is described in Section III and results are discussed in Section IV. Finally paper is concluded in Section V.

II. Related Work

Related work section is divided in two sections. First one is about geo encryption and second is about GPT cryptosystem.

A. GEO Encryption

The term geo encryption or location based security refers to the encryption technique that restricts the access to the encrypted data to a specified location at specified time even for a legal user. This restriction can be based on location and time dependent parameters. The main idea is to ensure that data cannot be used other than the authorized location and time. In [1], Logan and Denning proposed a framework for the implementation of geo encryption for digital movie distribution as shown in Fig. 1. They proposed a hybrid approach to implement geo encryption for digital movie distribution which means both public key and private key algorithms are used. The actual data is encrypted using private key encryption algorithm and then the key used for encryption is XORed with a geo lock which is computed using location and decryption time of the intended receiver. This XORed data is then encrypted again using public key encryption algorithm. At the other end, the receiver will first decrypt the encrypted key using private key and then to get the session key the output will XORed with geo lock which is computed using the same function as used at the sender. The session key will be then used to decrypt the data.

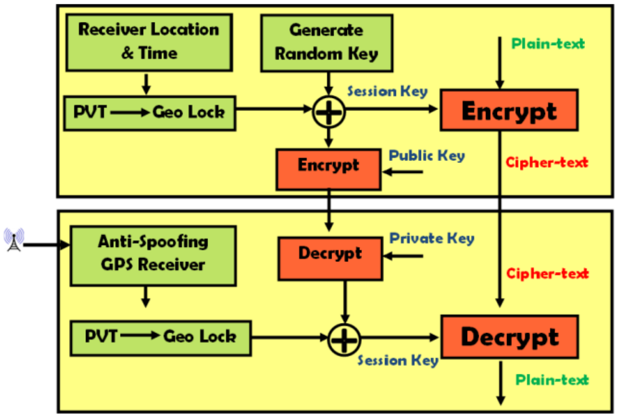

Fig. 1. Geo Codex.

B. Rank Codes

The rank distance codes is first provided in [25]. Let F_q and $F_q{}^N$ represent a finite base field of q elements and an extension field of degree N respectively. If $a = (a_1, a_2, a_3, \ldots, a_n)$ is a vector having coordinates from extension field then the Rank of a is defined as the maximal number of a_i, which are linearly independent over the base field and it can be denoted as $rk(A|F_q)$. The Rank distance between any two vectors a and b is the rank of the difference between a and b i.e. $d(a,b) = rk(a-b|F_q)$. In case of any matrix having all its elements from extension field, its column rank will be all those columns, which are linearly independent over base field. The column rank of any matrix A can be denoted as $rk(A|F_q)$.

In [26], the detailed description about the theory optimal MRD codes is given. The k x n generator matrix G of any MRD code is defined as

$$G = \begin{pmatrix} g_1 & g_2 & g_3 & \cdots & g_n \\ g_1^{[1]} & g_2^{[1]} & g_3^{[1]} & \cdots & g_n^{[1]} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ g_1^{[k-1]} & g_2^{[k-1]} & g_3^{[k-1]} & \cdots & g_n^{[k-1]} \end{pmatrix} \qquad (1)$$

where $g_1, g_2, g_3, \ldots, g_n$ are randomly chosen elements from extension field F_q^N. All elements of the first row of generator matrix must be linearly independent over the base field F_q, here $g^{[i]} := g^{q^{i \bmod N}}$ represents the i^{th} Frobenius power of *g*. If *q=2*, then each element of current row is square of the elements present in the same column in previous row. If $m = m_1, m_2, m_3, \ldots, m_k$ is a *k*-dimensional information vector then the corresponding code vector of dimension *n* will be:

$$g(m) = mG_k \qquad (2)$$

If $y = g(m) + e$ is a received code-word and if rank of the error vector e is, $rk(e|F_q) = s \le t = \left\lfloor \frac{d-1}{2} \right\rfloor$, then the information vector m can be easily gotten back by applying decoding algorithms on y. For decoding any MRD code, another ((n-k) × n) matrix called parity check matrix is need and it is denoted as H. The generator matrix G and parity check matrix H are orthogonal to each other, i.e. $G.H^T = 0$. A parity check matrix can be represented as

$$H = \begin{pmatrix} h_1 & h_2 & h_3 & \cdots & h_n \\ h_1^{[1]} & h_2^{[1]} & h_3^{[1]} & \cdots & h_n^{[1]} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ h_1^{[d-1]} & h_2^{[d-1]} & h_3^{[d-1]} & \cdots & h_n^{[d-1]} \end{pmatrix} \qquad (3)$$

where elements $h_1, h_2, h_3, \ldots, h_k$ are from F_q^N and like generator matrix all elements of the first row must be linearly independent over F_q. The notation $h^{[i]} := h^{q^{[i \bmod N]}}$ means the i^{th} Fresenius power of *h*. The detail of rank codes is described in [21].

C. Description of Stndard GPT Cryptosystem

Several variants of GPT cryptosystem are proposed due to several attacks against the original system. The main structure of almost all the variant remains the same. The difference lies in the construction of various matrices comprising the public key. To be more precise, it's how the elements of various matrices must be chosen so that an attack could be ineffective against the system. For proposed scheme, parameters suggested in [19] are considered which withstands all the known attacks to date. The public key of GPT cryptosystem is given below:

$$G_{pub} = S\,G\,P^{-1} \tag{4}$$

The S is a k × k non-singular, row scrambler matrix over$F_q{}^N$. G is the generator matrix as given in eq (1). The matrix P is an invertible having entries from $F_q{}^N$ as described in [19].

III. GEO Encryption using GPT

The main advantage of using public key cryptosystems over private key cryptosystem is that in former, one does not need to transfer the private keys to the receiver to decrypt the cipher text instead one encrypts the message using public key of the receiver provided through any certificate authority or public directory. The challenge of implementing geo encryption using a public key cryptosystem is to restrict the receiver from decrypting the cipher text without being on the permitted location or at inappropriate time. It means that receiver must verify its location and time to accurately decrypt the cipher text. On the other hand, the receiver is unable to calculate the private keys based on its location and time parameters alone without knowing the structure of the public key. Therefore, to implement the location and time restrictions partial information about the private keys will be sent to receiver and to accurately calculate the private key and to verify its location and allowed decryption time, the receiver has to calculate the rest of the key based on its location and time parameters. Fig. 2 shows the overview of the proposed scheme. In GPT cryptosystem, all keys are in the form of matrices based over F_q^N.

At sender, first a parity check matrix will calculated based on the geographical coordinates of the receiver then a corresponding generator matrix and public key matrix will be calculated. A data will be encrypted using this public key and transmitted over any channel. Some information about the calculation of parity check matrix along with two other matrices which serve as private key will be encrypted together using another instance of GPT cryptosystem using the public key of the intended receiver which means that to decrypt this data we do not need to provide the private key to receiver as it already has it. At the receiver, first of all the encrypted keys will be decrypted to get the two private key matrices and some information about calculating parity check matrix, in parallel to this a key generation which will geographical location parameter to calculate remaining information needed to calculate the parity check matrix. The output of this function along with the information receiver from the sender will be used to calculate the parity check matrix.

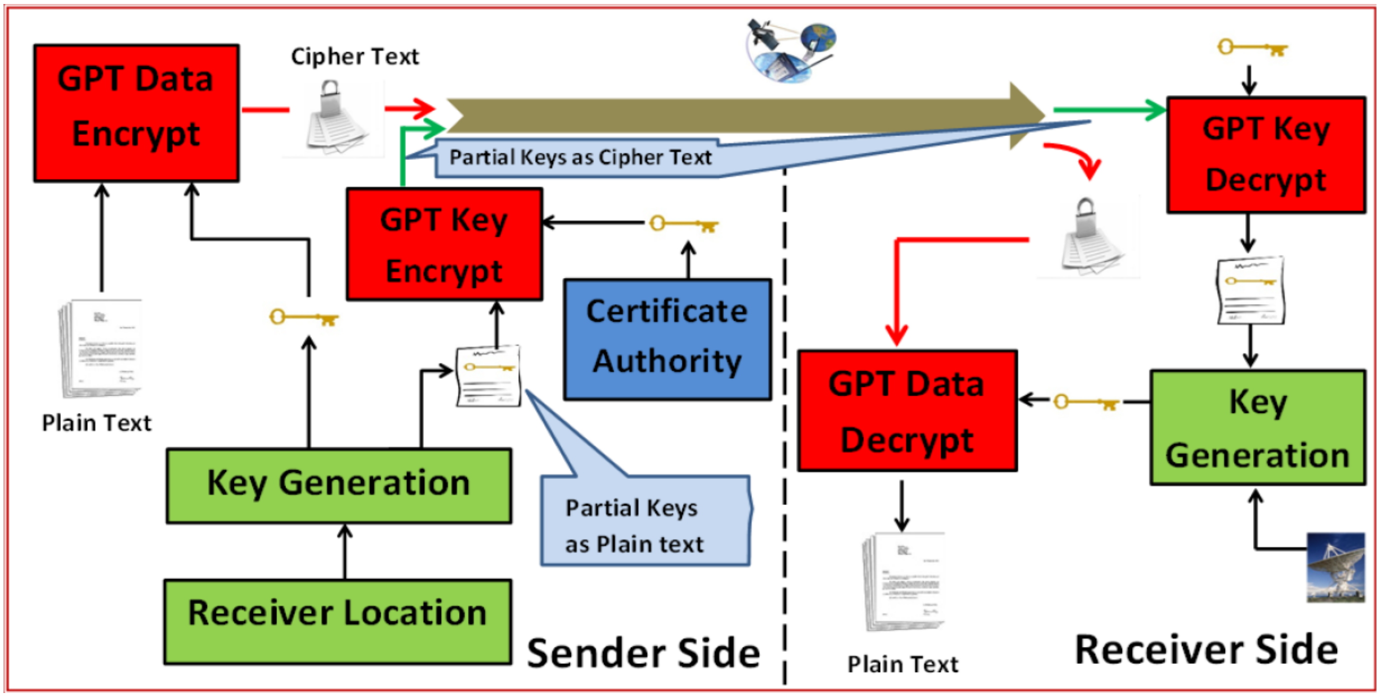

Fig. 2. Geo Encryption with GPT.

Sender side: The key generation algorithm at the sender side is presented in algorithm given below. First of all, the sender will check whether the message is already encrypted. If it is already encrypted, then the sender is not needed to calculate the parity check matrix, generator matrix and public key instead it will calculate an appropriate initial vector and encrypt it along S and P matrices and send it to the receiver along with already encrypted message. So there will be two cases:

Case 1: First time Encryption. The sender will calculate parity check matrix H by calculating an integer constant Φ using the location and time of the week parameters of the intended receiver using a pseudo random permutation. Any pseudo random permutation can be used which could take location and time as inputs and return a big integer as output. It should be noted that the size of the integer constant Φ must of $\leq 2^N - 1$, where N is the degree of the extension field. e.g. if N=8, then the largest value Φ can have is 255.

$$\varphi \leftarrow f(lat, lon, TOW) \tag{5}$$

After calculating Φ, write it in the multiplicative factors of powers of two.

$$\varphi = (x_{n-1}2^{n-1} + x_{n-2}2^{n-2} + \ldots + x_0 2^0) \tag{6}$$

```
**Input**: Location coordinates and time such as latitude, longitude and time of the week
**If** Message NOT Encrypted Before Then
        Compute $\varphi \leftarrow f(latitude, logitude, TOW)$
        Write $\varphi$ in the multiplicative factors of powers of 2
        Compute $h_{LV}$ by replacing all the powers of 2 for which coefficient is one
        With corresponding elements from $F_q^N$ and rest with zeros
        **Repeat**
            Compute $h_{LV}$ by choosing elements from $F_q^N$
            Compute final generating vector $h_f$ for parity check matrix as
                $h_f = h_{LV} \oplus h_{IV}$
        **Until** All elements of $h_f$ are not linearly independent
        Compute parity check matrix as given in eq 4
        Compute corresponding Generator matrix using matrix reduction algorithm
**Else**
        **If** Same location but different receiver
            Compute β by randomly choosing elements from $F_q^N$
            Compute $h_{f\,new} = \beta \times h_f$
            Compute $h_{IV} = h_{f\,new} \oplus h_{LV}$
        **Else**
            Compute ϕ ← $f(latitude, logitude, TOW)$
            Write $\varphi$ in the multiplicative factors of powers of 2
            Compute $h_{LV}$ by replacing all the powers of 2 for which coefficient is one
            With corresponding elements from $F_q^N$ and rest with zeros
            Compute $h_{IV} = h_f \oplus h_{LV}$
        **End if**
**End if**
```

Here x_i are the coefficient having values either 0 or 1 and $i = 0,1,\ldots,n-1$. In above equation where coefficient is 1, replace powers of 2 with the corresponding elements from the ${F_q}^N$ and insert 0 for rest all. Call it location generating vector h_{LV}. i.e. $h_{LV} = h_{LV_1}\ h_{LV_2}\ h_{LV_2} \ldots h_{LV_n}$

```
**Input**: $n \times n$ matrix
**For** $i = 1 \rightarrow n$ then
    **If** first element of first column $\neq 0$ **then**
        Do Nothing
    **Else**
        **If** left most of the rest of the columns having first element $\neq 0$ & diagonal element $= 0$ **then**
            Swap the first column with this column
        **Else If** left most of the rest of the columns having first element $\neq 0$ & diagonal element $\neq 0$
            Swap the first column with this column
        **End If**
    **End If**
    **If** first element of the first column $\neq 0$ then
        Divide the column by its first element
    **Else**
        Do Nothing
    **End If**
    Zero the first element of each column except the first column by subtracting an appropriate multiple of the first column
    Rotate rows upwards and column leftwards
**End for**
```

In next step choose an initial generating vector h_{IV}. There are two reason of choosing this generating vector. First, it will make sure that all the elements of final generating vector which is the first row of parity check matrix are linearly independent. Second, it will completely distort the elements of h_{LV}. To correctly calculate the parity check matrix at receiver, the sender will transmit h_{IV} to the receiver.

Calculate the final generating vector h_f for parity check matrix by taking XOR of h_{IV} and h_{LV}.

$$h_f = h_{IV} \otimes h_{LV} \tag{7}$$

The h_f is the first row of the parity check matrix, rest all rows are frobenious power of each element of the previous row. After calculating parity check matrix H, the sender will calculate a corresponding generator matrix G orthogonal to parity check using matrix reduction algorithm provided below which was Originally proposed in [13].

The message will be encrypted using equation 7 and sent to the receiver using any communication channel. As sender has encrypted the message without giving any prior information about private keys to the receiver, so it will also transmit S^{-1}, P^{-1} and h_{IV} to the receiver in the form of another cipher text encrypted using the public key of the receiver which can be obtained from certificate authority or public directory.

Case 2: Message is already encrypted. If the message is already encrypted then the sender will check whether the intended receiver sharing the same location parameters with the previous receiver for which message was encrypted because in that case the sender will compute new parity check matrix. The parity check matrix of rank codes is quite structured. The generator matrix which is orthogonal to one parity check matrix is also orthogonal to any other parity check matrix which is calculated using the same generating vector multiplied with any randomly chosen element from extension field. It means, if h is the generating vector for a parity check matrix H which is orthogonal to a generator matrix G, then another parity check matrix $\tilde{H}$ which is generated using another generating vector $\tilde{h} = \beta \times h$ is also orthogonal to generator matrix G, where β is a randomly chosen element from extension field. The parity check matrix with $\tilde{h}$ will also be orthogonal to the generator matrix in eq. 3.

$$h_{f_{new}} = \beta \times h_f \qquad (8)$$

New h_{IV} will be calculated as

$$h_{IV} = h_{f_{new}} \otimes h_{LV} \qquad (9)$$

and sent to the receiver. If the receiver location is different, then the sender will calculate the h_{LV} and will XOR this with the h_f to get h_{IV}.

Receiver side: The algorithm for key generation at receiver is given below. All the steps of key generation algorithm at receiver are similar to steps at the sender except the elements of h_{IV} are not randomly chosen instead the receiver will use the h_{IV} provided by the sender.

```
Input: Location coordinates and time such as latitude, longitude and time of the week
Compute Matrix generating constants
Compute $\varphi \leftarrow f(latitude, logitude, TOW)$
Write $\varphi$ in the multiplicative factors of powers of 2
Compute $h_{LV}$ by replacing all the powers of 2 for which coefficient is one with corresponding elements from $F_q^N$ and rest with zeros
Decrypt the generating vector $h_{IV}$ received from the sender
Compute the final generating vector $h_f$ for parity check matrix as

$h_f = h_{LV} \oplus h_{IV}$
Compute the parity check matrix as given in eq. 4
Compute corresponding Generator matrix using matrix reduction algorithm
```

IV. Analysis and Discussion

In this section, different aspects of the proposed scheme will be analysed and discussed.

A. Security

In the proposed scheme, there are two types of messages which are transmitted from sender to receiver. First is the data itself and second are the private keys to decrypt this data. Both of these messages are encrypted using the GPT cryptosystem first and then transmitted over the channel. Therefore it can be said that the overall security of the proposed scheme is equal to that of security of the cryptosystem itself. Although the keys are transmitted from sender to receiver but these are not enough to decrypt the encrypted data. Only S^{-1}, P^{-1} and h_{IV} are provided. Using the PRP, the receiver has to calculate h_{LV} and combine it with h_{IV} to get parity check matrix. The only way in which an adversary can attack the system is to calculate h_{IV} by correctly guessing the location parameters and decryption time and then using the pseudo random permutation (PRP) to calculate h_{LV}. Therefore it is suggested that the user must use a secure PRP to get the Φ and it must be secret. Even though adversary correctly calculates the h_{LV}, it is not enough because he/she still needs the encrypted S^{-1}, P^{-1} and h_{IV} to decrypt the cipher text. One of the potential attack against any code based cryptosystem is the decoding attack. In decoding attack, an adversary tries to recover the plain text by correcting the errors using a general decoding algorithm without any knowledge of the structure of the code. The aim of the adversary is to try to decode the encoded/encrypted message to the nearest possible codeword. If the adversary is successfully to decode the encrypted message then he/she can recover the original plain text correctly. The general decoding algorithms do not consider the inherent structure of the code. They treat the published code as random. In [10], the authors published two general decoding algorithms to decode an arbitrary linear rank codes. These algorithms can correct errors of rank $t = \left\lfloor \frac{d-1}{2} \right\rfloor$ in $O^{(k+t)^3 t^3 q^{(t-1)(N-t)}}$ and $O^{(Nt)^3 q^{(t-1)(k+1)}}$ operations in F_q. Fig. 3 and Fig. 4 show the operation complexities of these algorithms with respect to key size.

The operation complexity is calculated for three different values of *n* and *k*. It can be seen in both Fig. 3 and Fig. 4 that when n=24 and k=20 the cryptosystem provides good information rate $k/n = 0.833$ but at the same time it is not secure at all and can be easily broken in about 2^{38} operations. For *n=28* and *k=14* the information rate will be $k/n = 0.5$

and the first algorithm requires about 2^{148} operations and second algorithm requires about 2^{113} operations which is quite secure with the current computing power.

B. Key Size and Information Rate

Although algebraic code based cryptosystems are considered as cryptosystems for post quantum computing but they are still not widely accepted for application development due to their huge key size and data expansion. As compared to McEliece [5] and Niederreiter [6], GPT cryptosystem has reduced key size with almost same level of security. Results in Fig. 5 shows the key size versus information rate for different values of *t* where $t \leq \left\lfloor \frac{n-k}{2} \right\rfloor$ is the error correcting capability of the code.

The proposed work randomly chooses h_f to completely mix all elements of generating vector of parity check matrix, so security of proposed system is same as the security of original GPT cryptosystem.

C. Decoding Speed

In [10], two fast decoding algorithms are proposed to decode any rank distance code. First one is Matrix Decoding Algorithm (MDA) and second is Decoding based on Right Euclidean Decoding Algorithm (DREDA). Space limitations discourage from going through each and every step of mentioned algorithms and arithmetic operations required in these steps, instead the arithmetic operations required in these algorithms are summarized in Table I.

Here *t* is error correcting capability of the code and is defined as $t = \left\lfloor \frac{d-1}{2} \right\rfloor$ and *n* is the code length. Table I shows that the number arithmetic operations depend on the size of *t*. Table II and Table III show the exact number of operations required for different values of *t*. Here *n=30* and *k* is changed for different information rates.

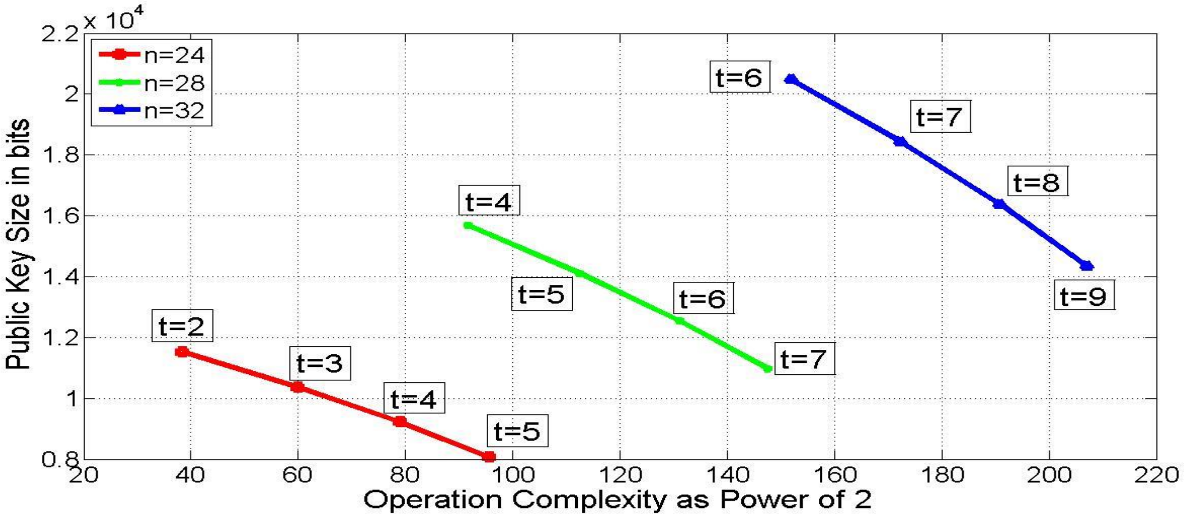

Fig. 3. Key Size vs Complexity of First Decoding Algo.

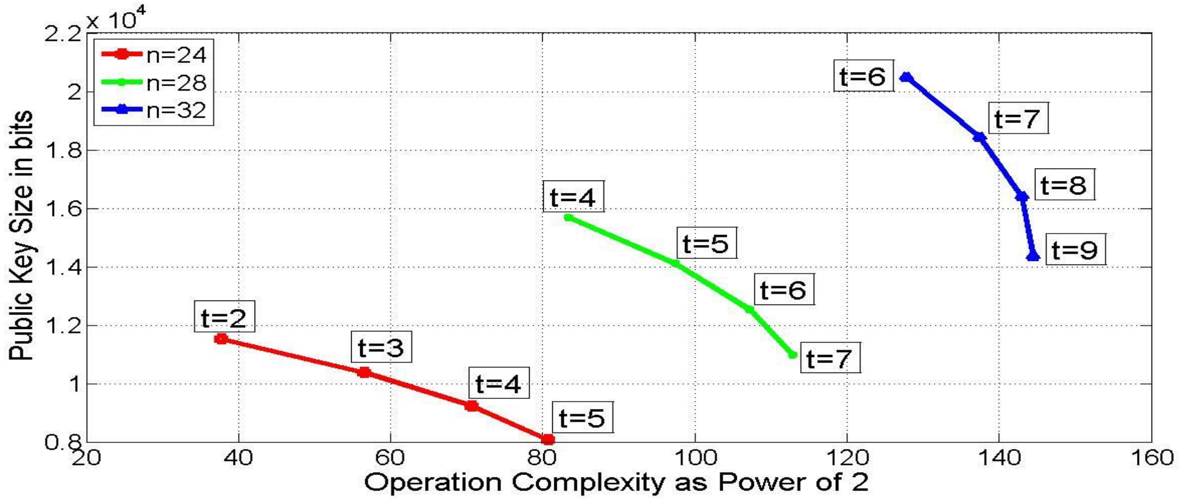

Fig. 4. Key Size vs Complexity of Second Decoding Algo.

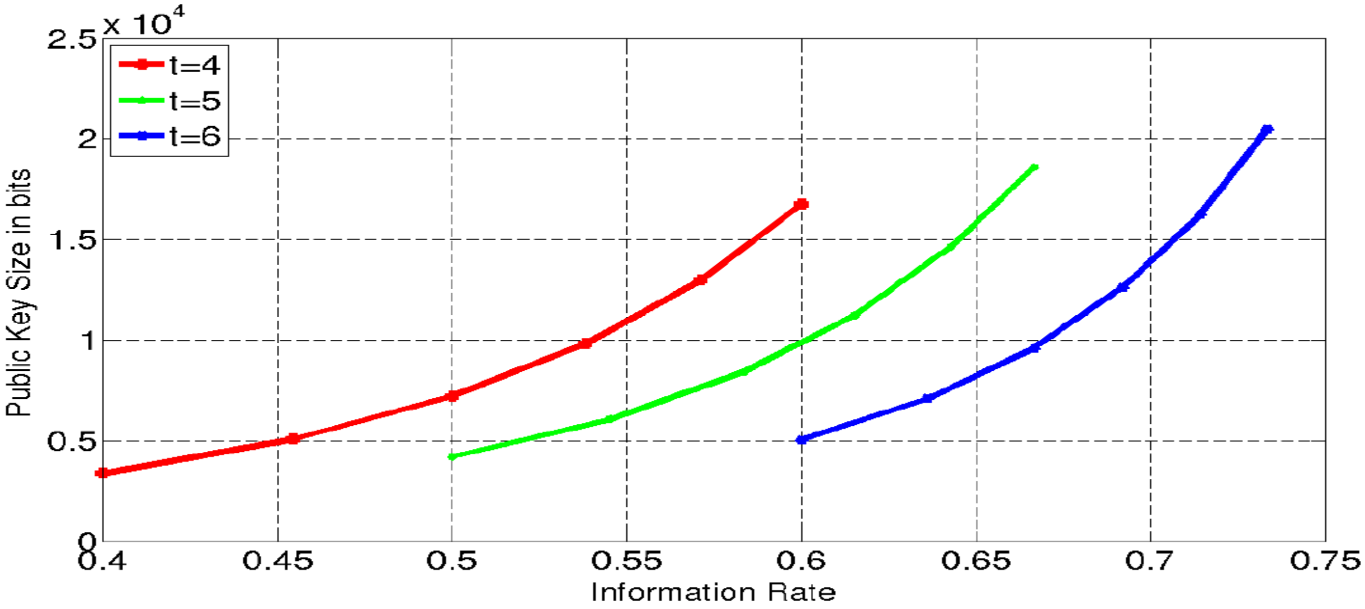

Fig. 5. Key Size vs Information Rate.

TABLE. I. ARITHMETIC OPERATIONS REQUIRED IN FAST DECODING ALGORITHMS

Operations	**MDA**	**DREDA**
Multiplications	$2tn + t!\,(t-1) + 2t(t-1)^2$	$2tn + 2t(2t+1) + t(t-1)^2$
Additions	$t! + 2t^2(t-2) + 3(t-1)n$	$t^2(t + {}^{3}/_{2}) + (3t-1)n + {}^{t}/_{2}$
Divisions	$2t(t-1)$	$t(t-1) + 2t + 1$
Squares	0	$\frac{t(7t+3)}{2}$
Square Roots	$t(2t+1)$	0

TABLE. II. ARITHMETIC OPERATIONS REQUIRED IN MDA ALGORITHM

K	**T**	**MDA**			
		Mul	Add	Division	SQ
20	4	368	396	24	36
18	5	920	662	40	55
16	6	4236	1484	60	78
14	7	31136	6090	84	105

TABLE. III. ARITHMETIC OPERATIONS REQUIRED IN DREDA ALGORITHM

K	**T**	**DREDA**			
		Mul	Add	Division	SQ
20	4	332	396	21	63
18	5	470	557	31	95
16	6	642	749	43	135
14	7	854	980	57	182

V. CONCLUSION

In this paper, an algorithm for implementing geo encryption using one of the algebraic code based cryptosystem called GPT cryptosystem is proposed. The algorithm proposed a new technique for calculating location based parity check matrix and corresponding public key. Although the key is calculated using geographic location but still it is completely randomized by mixing it with random elements from extension field thus the level of security of the proposed system is equal to that of the underlying GPT public key cryptosystem. This work introduced an idea of encrypting with one public key and decrypted with multiple different private keys but calculating different parity check matrix for each user.

REFERENCES

[1] Scott, L., Denning, D.E., “A Location Based Encryption Technique and Some of Its Applications”, Proceedings of the 2003 National Technical Meeting of The Institute of Navigation, Anaheim, CA, pp. 734-740, January 2003. http://faculty.nps.edu/dedennin/publications/location basedencryption-ion2003.pdf.

[2] Peter W. Shor, “Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer”, SIAM Journal on Computing, vol. 26, issue. 5, pp. 1484-1509, October 1997.http://dx.doi.org/10.1137/S0097539795293172.

[3] Nicolas Sendrier, Jean-Pierre Tillich, "Code-based Cryptography: New Security solutions against a quantum adversary", ERCIM News, ERCIM, 2016, Special Theme Cybersecurity (106), July 2016 https://hal.archives-ouvertes.fr/hal-01410068/file/codebased-final.pdf.

[4] Nicolas Sendrier,"Code-Based Cryptography: State of the Art and Perspectives", IEEE Security & Privacy, vol.15, issue 4, pp.44-50, 2017 https://doi.org/10.1109/MSP.2017.3151345.

[5] McEliece, R. J., "A Public Key Cryptosystem Based on Algebraic Coding Theory", JPL DSN Progress Rep. 42-44. https://tmo.jpl.nasa.gov/progress_report2/42-44/44N.PDF.

[6] Niederreiter, H., "Knapsack-Type Cryptosystem and Algebraic Coding Theory", Probl. Control and Inform. Theory, vol. 15, pp. 19-34, 1986.

[7] Gabidulin, E. M., Paramonov, A.V., Tretjakov, O.V., "Ideals over a non-commutative ring and their application in cryptology", advances in

Cryptology, Proc. EUROCRYPT' 91, LNCS 547, D. W. Davies, Ed. Springer-Verlag, 1991, pp. 482-489. https://doi.org/10.1007/3-540-46416-6_41.

[8] Gibson J. K., “Severely denting the Gabidulin version of the McEliece public key Cryptosystem”, Designs Codes and Cryptography, 6(1), 1995, pp.37-45. https://doi.org/10.1007/BF01390769.

[9] Gibson J. K., “The security of the Gabidulin public-key cryptosystem”, U. M. Maurer (Ed.), Advances in Cryptology --EUROCRYPT'96, LNCS vol 1070, Springer, Berlin, 1996, pp. 212-223. https://doi.org/10.1007/3-540-68339-9_19.

[10] Ourivski A. V., Johansson T., “New Technique for Decoding Codes in the Rank Metric and Its Cryptography Applications”, Problems of Information Transmission, 38(3), 2002, pp. 237-246. https://doi.org/10.1023/A:1020369320078.

[11] Overbeck, R., “Structural attacks for Public Key Cryptosystem Based on Gabidulin codes”, Journal of Cryptology, 21(2), 2008, pp.280-301. https://doi.org/10.1007/s00145-007-9003-9.

[12] Gabidulin E. M., "Attacks and counter-attacks on the GPT public key cryptosystem", Designs Codes and Cryptography, Springer Netherlands, (48) 2, August 2008, pp.171-177. https://doi.org/10.1007/s10623-007-9160-8.

[13] Gabidulin, E.M., Rashwan, H., Honary,B., "On Improving Security of GPT Cryptosystems", Int. Symposium on Information Theory, pp.1110-1114. 2009 https://doi.org/10.1109/ISIT.2009.5206029.

[14] Rashwan, H., Gabidulin, E. M. and Honary, B. (2011), “Security of the GPT cryptosystem and its applications to cryptography”. Security Comm. Networks, 4: 937–946. 2011 http://dx.doi.org/10.1002/sec.228.

[15] Khan, E., Gabidulin, E.M., Honary, B., Ahmed H., " Modified Niederreiter Type of GPT Cryptosystem Based on Reducible Rank Codes", Designs Codes and Cryptography, Springer, vol(70) 1, pp. 231-239, 2014 https://doi.org/10.1007/s10623-012-9757-4.

[16] Anna-Lena Horlemann-Trautmann, Kyle Marshall and Joachim Rosenthal, "Extension of Overbeck's attack for Gabidulin based cryptosystems", Designs Codes and Cryptography, 2017. https://doi.org/10.1007/s10623-017-0343-7.

[17] Ayoub Otmani, Herve Tale Kalachi, Selestin NDJEYA, "Improved Cryptanalysis of rank metric schemes based on Gabidulin Codes", Designs, Codes and Cryptography, 2017. https://doi.org/10.1007/s10623-017-0434-5.

[18] Philippe Gaborit, Ayoub Otmani, Herve Tale Kalachi, "Polynomial-time Key recovery attack on the Faure-Loidreau scheme based on Gabidulin Codes", Designs, Codes and Cryptography, 2017. https://doi.org/10.1007/s10623-017-0402-0.

[19] Pierre Loidreau, "A new rank metric codes based encryption scheme", Post-Quantum Cryptography : 8th International Workshop, PQCrypto 2017, Utrecht, The Netherlands, June 26-28, 2017, Proceedings, pp. 3-17, 2017. https://doi.org/10.1007/978-3-319-59879-6_1.

[20] Philippe Gaborit, Oliver Ruatta, Julien Schrek, Jean-Pierre Tillich, "Rank based cryptography: a credible post-quantum alternative to classical cryptography", NIST workshop on cybersecurrity in a Post-Quantum World 2015. https://csrc.nist.gov/csrc/media/events/workshop-on-cybersecurity-in-a-post-quantum world/documents/papers/session1-gaborit-paper.pdf.

[21] Singh KJ and Gagneja K. Overview of securing multimedia content using efficient encryption methods and modes. International Journal of Advanced and Applied Sciences, 2017; 4(10): 84-96.

[22] Arboleda. ER, Fenomeno CE and Jimenez JZ. KED-AES algorithm: combined key encryption decryption and advance encryption standard algorithm. Int. J. of Adv. in Appl. Sci. 2018; 8(1): 44-53.

[23] Nagavalli S, Ramachandran G. A Secure Data Transmission Scheme using Asymmetric Semi-Homomorphic Encryption Scheme. Int. J. of Adv. in Appl. Sci. 2018;7(4): 369-376.

[24] Pushpa K, Lakshmi L, Sabitha Ch.,Dhana B and Sreeja S.Top-K search scheme on encrypted data in cloud. Int. J. of Adv. in Appl. Sci. 2019;9(1): 67-69.

[25] E. M. Gabidulin, "The theory of codes with maximum rank distance", Problems Inform. Transmission 21 (1), pp. 1-12, 1985. https://www.researchgate.net/publication/235008632_Theory_of_codes_with_maximum_rank_distance_translation.

[26] E.M. Gabidulin,: ``Public-Key Cryptosystems Based on Linear Codes over Large Alphabets"; Efficiency and Weakness, in:Codes and Ciphers, Editor: P.G. Farrell, pp. 17--32, Essex: Formara Limited, 1995. http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.28.4876&rep=rep1&type=pdf.