Available online at www.sciencedirect.com

ScienceDirect

Energy Procedia 42 (2013) 299 – 307

Energy Procedia

The Mediterranean Green Energy Forum 2013, MGEF-13

Location Based Security for Smart Grid Applications

Eraj Khan [‡], Bamidele Adebisi [ɮ], Bahram Honary[‡]

[‡] School of Computing and Communication, Lancaster University, UK
Email: {e.khan, b.honary}@lancaster.ac.uk

[ɮ] School of Electrical Engineering, Manchester Metropolitan University, UK
Email: b.adebisi@mmu.ac.uk

Abstract

Smart Grid (SG) promises efficient, sustainable, green and reliable electrical delivery by combining the existing electrical distribution network assets with modern information and communications technologies (ICT) in order to transfer information and energy in both directions. Introduction of these intelligent devices will help the grid monitor, protect, and automatically optimise the operation of interconnected elements, in addition to interaction between energy suppliers and consumers. However, this exposes the future grid to new security challenges and risks. In this paper, a framework for protecting Smart Grid applications using geographic location of the devices connected to it is proposed. With this framework, each device on the grid adds an extra layer of security. The proposed scheme is not application or device specific which means it can be implemented on any communication node on the grid. The scheme uses an algebraic code based cryptosystems known as GPT (Gabidulin -Paramonov-Trejtakov), which provides a very strong protection while utilising the smallest key size as compared to other cryptosystems based on algebraic codes. As with other code based cryptosystems, the proposed security framework protects grid information against cyber threats as well as against channel impairments in the form of error protection codes.

Keywords—security; smart grid; GPT, RSA

doi:10.1016/j.egypro.2013.11.030

1. INTRODUCTION

Recently, there has been a lot of research interests in new ways of producing, distributing and managing electric energy in order to facilitate cleaner, environmentally friendly and sustainable consumption[1][2]. Future electric networks are also expected to be resilient, ensure delivery of efficient service, and have technologies that enable it to self-heal. Consumers would also be able to take part in production and decision making. A major component of such grid of the future is communication. By installing communication systems at different points within the electric network, the current 'dumb' electrical grid will be transformed into an improved and intelligent grid also known as Smart Grid. The complexity and scale of a SG poses some serious challenges. Some of these are related to security of the grid and privacy of the consumers. Although SG enhances better power distribution, and integration of renewable energies, it does so at a cost. The increase in the number of devices in the grid makes it more vulnerable to physical as well as cyber-attacks such as Trojans, denial of service attacks, viruses, malwares etc. ICT systems supporting critical infrastructure (e.g. energy plants, water plants, financial entities, public administrations, transport networks, etc.) are no longer separated and isolated entities. The interconnection with other public and open networks causes security problems, and successful attacks may have significant effects, such as, for example, energy blackouts. Moreover, the crucial information from monitoring systems may be delayed or even lost, preventing early warning systems from proper and on- time reaction.
Since SG communication technologies are in their development stage, hence new security features could still be included in the design and development of Smart Grids devices because introduction of such important components after deployment could be more expensive, in terms of cost, time, and customer satisfaction.

In this paper, a security framework for protecting SG infrastructure using location based encryption, also known as Geo-Encryption is proposed. This location based security is designed using one of the algebraic code-based cryptosystems called GPT named after its inventors; Gabidulin, Paramanov and Tretjakov [3]. The GPT cryptosystem is the third cryptosystem of its kind. The first code-based cryptosystem, based on Goppa codes was proposed in 1978 by McEliece [14]. In 1986, another code-based cryptosystem was proposed by Niederreiter [15]. These cryptosystems are based on the problem of decoding a general linear code, which is considered very difficult to solve in polynomial time. The cryptosystem proposed by McEliece is still unbroken with its original parameter but it has not gained wider acceptance because of its very large key size. Algebraic code-based cryptosystem has been chosen for Smart Grids instead of the existing systems, which are based on number theoretic approaches because research results published in [11] and [12] have shown that the existing systems may not survive future cyber threats. This is due to evolutionary developments in computing hardware and the introduction of quantum computing. The prime candidates for future security are code-based cryptosystems. Furthermore, public key cryptosystems based on algebraic coding theory are several times faster than the Ron Rivest, Adi Shamir, and Leonard Adleman (RSA) cryptosystems. The key attraction of GPT when compared with other code based cryptosystems is that it uses the smallest key size, which means lower complexity. This is because the GPT is based on rank error correcting codes and uses the Rank metric instead of Hamming metric. Rank codes are well-structured codes. Therefore, several structural attacks have been proposed against the

GPT cryptosystem [5] ,[6]. Some modifications to withstand these attacks are also proposed in [7], [8] and [9]. To the best of authors' knowledge these modifications withstand all the known attacks against the different variants of the cryptosystem. GPT cryptosystems do not only secure data against cyber threats; but also protect data against channel impairments in the form of error protection codes.

The rest of the paper is organised as follows. Section 2 briefly explains geo encryption and GPT cryptosystem. The proposed scheme is presented in section 3. Results and analysis are given in section 4 while section 5 concludes the paper.

2. GEO ENCRYPTION

In Geo encryption, a message is encrypted such that it can only be decrypted at a specified location and at a specified time. If any user tries to decrypt the cipher text at any location or time, other than the specified, the decryption fails. It can only be achieved using a location dependent decryption key. The idea of location based authentication was first proposed in [10]. The authors presented a mechanism for restricting the internet access for remote users based on their geographical location. In order to get access to the server, a geographical location based signature was created and appended to the data packet along with location data collected from satellites and then transferred to the server. At the receiver, the server verifies this signature by its own simultaneously collected satellite location information about the remote user and allows access to the user if he/she is within acceptable range.

A. GPT cryptosystem

Let **F2** be a finite field of q elements and let F2N be an extension field of degree N. Let a = (a1, a2... an) be a vector with coordinates in **F2N**.

The Rank norm of a is defined as the maximal number of ai, which are linearly independent over the base field **F2** and is denoted Rk(a | **F2**). In our scheme, we will consider the specifications of GPT cryptosystem proposed in [8]. The Public key is given in equation (1) below

$$G_{pub} = S(X|G)P \qquad (1)$$

Where **S** is a **k x k** non-singular row scrambler matrix based over the extension field. **X** is a **k x t_1** matrix called distortion matrix. The matrix is not required for decryption, so it can be deleted after encryption. The matrix **P** is a non-singular column scrambler matrix of order $(n+t_1)$, partially based over the extension field. The k x n generator matrix G is of the form given below

$$\mathbf{G} = \begin{bmatrix} g_1 & g_2 & \cdots & g_n \\ g_1^{[1]} & g_2^{[1]} & \cdots & g_n^{[1]} \\ g_1^{[2]} & g_2^{[2]} & \cdots & g_n^{[2]} \\ \vdots & \vdots & \ddots & \vdots \\ g_1^{[k-1]} & g_2^{[k-1]} & \cdots & g_n^{[k-1]} \end{bmatrix} \qquad (2)$$

the

Where $g_1, g_2 \ldots, g_n$ are any set of elements over the extension field $\mathbf{F}_2{}^N$ which are linearly independent over base field $\mathbf{F}_q$ and $g^{[i]} = g^{q^{i \bmod n}}$ means the i^{th} Frobenius

power of g. The code generated with (2) is called (n, k, d) code, where n is the code length, k is the information symbols and d which is equal to (n – k + 1) is the distance of the code. For such code there exists a fast decoding algorithm which can correct rank errors up to $t = \frac{d-1}{2}$. **Private Keys** are **S**, **G**, **P** matrices and fast decoding algorithm. **Plain-text** is the information vector of dimension k of the form m = (m_1, m_2, m_3, … , m_k) based over $\mathbf{F}_2^N$

3. Proposed Scheme

In order to generate a location-based key, two very important factors must be considered: area of decryption and accuracy of the location device, i.e., how large should the area for decryption be? It is very important that the area is not too large as to allow access to anyone equally, it should not be too small as to shut out some of the intended users. Secondly, before encrypting any data with intended receiver's geographical location parameters, the sender must consider the accuracy of the location device available at the receiver. Otherwise, even a legal receiver will be unable to calculate a valid decryption key. So there is a choice between either selecting the area of decryption large enough to cancel the accuracy errors of location device or the device must be accurate enough to give precise readings.

Fig. 1 shows the framework of our proposed scheme. GPT is a public cryptosystem. Hence, the communicating parties do not need to exchange session keys, but in the proposed framework, we are using two instances of GPT cryptosystems. One instance is used to encrypt the actual plain-text using key matrices calculated by linearly combining intended receiver location information and randomly chosen key from the $\mathbf{F}_2^N$ whereas the second instance is used to encrypt the randomly chosen matrices using the public key of the receiver. At the other end, the receiver will calculate the part of the private keys using its location devices and calculate the final private keys by decrypting the keys using its own private keys and then linearly combining the two information. It should be noted that, in order to decrypt the original cipher-text, the receiver must correctly decrypt the encrypted key information from the sender. It must also calculate the correct keys by using mapping function. Using any one of the information alone will result in failed decryption.

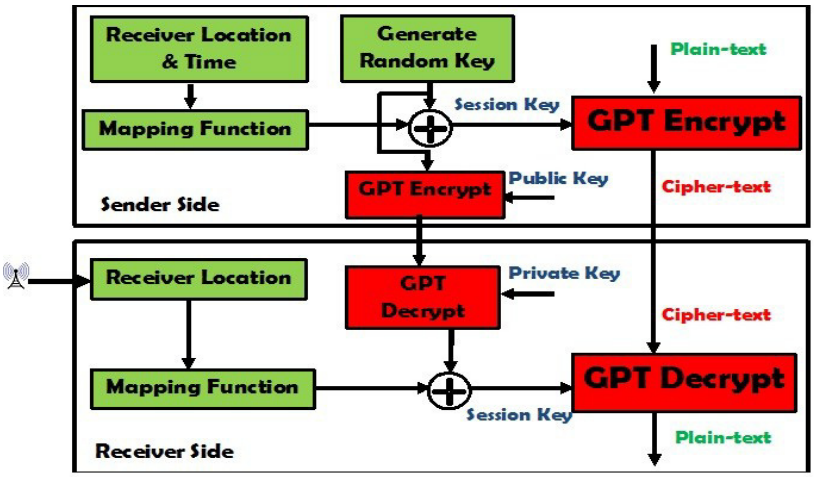

Figure 1. Proposed Framework

Steps for calculating at either end are explained below.

A. Sender side

1) Randomly choose private key matrices.
2) Take the geographical parameters of intended receiver. These parameters could be latitude, longitude and desired decryption time. Thencalculate location vector of length n.
3) Linearly combine the matrices calculated in step 1 and location vector calculated in step 2 to get the session keys for encrypting plain-text.
4) Encrypt the plain-text using session key calculated in step 3.
5) Send the cipher-text calculated in step 4 to the receiver.
6) Encrypt the randomly chosen private keys matrices using public key of the intended receiver.
7) Send the encrypted key information calculated in step 6 to the receiver.

B. Receiver side

The receiver will take the following steps to recover the plain-text from cipher-text.

1) Decrypt the encrypted key information using its private key.
2) Take the geographical parameters of intended receiver. These parameters could be latitude, longitude and desired decryption time, then calculate location vector of length n.
3) Linearly combine the matrices decrypted in step 1 and location vector calculated in step 2 to get the session keys for encrypting plain text.
4) Using the session key calculated in step 3, decrypts the cipher-text sent from the sender and recovers the plain text.

4. ANALYSIS AND DISCUSSION

A. Security Analysis

In any code-based cryptosystem, the public and private keys are in the form of matrices based over the finite field. Figure 2 shows different types of attacks against the cryptosystems. It has been shown that there could be two types of attacks against any code based cryptosystem; Structural attacks and Decoding attacks. In structural attacks, the adversary tries to recover private keys from the public key through mathematical transformations on the public key matrix. It was argued earlier that with the modifications proposed in [7], [8] and [9] GPT cryptosystem currently can withstand all the known structural attacks.

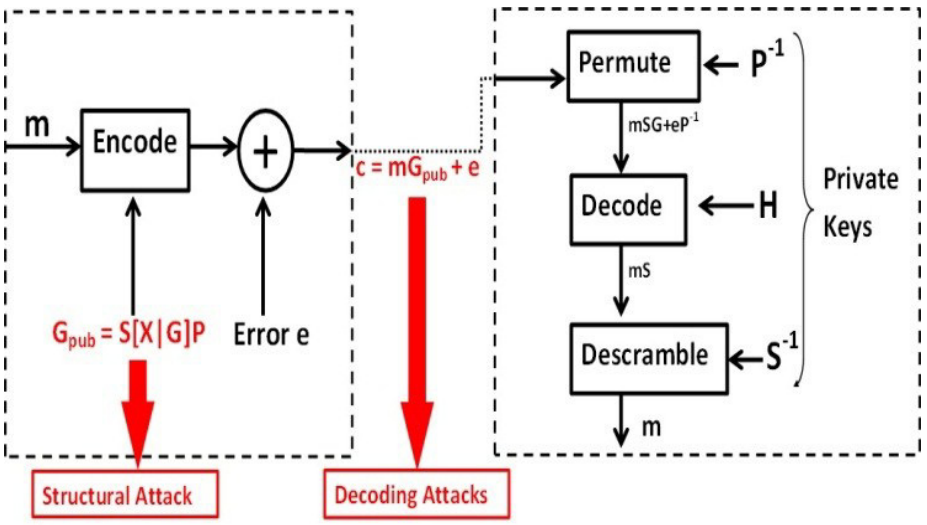

Figure 2. Different type of Attacks against GPT

In decoding an attack, a cryptanalyst tries to decode any (n, k, d) linear code without the knowledge of the structure of the codes. In [13], authors propose two algorithms for decoding an arbitrary (n ,k) linear rank codes over $\mathbf{F}_q^N$. These algorithms could be considered as the fastest decoding attacks against the GPT cryptosystem. The first algorithm correct t rank errors with $O((Nt)^3(q)^{(t-2)(k+1)})$ operations over the $\mathbf{F}_q^N$ whereas the second algorithm requires $O((k+t)^3t^3(q)^{(t-1)(N-t)})$ operations over the $\mathbf{F}_2^N$. Figure 3 and Figure 4, respectively show the comparison between the sizes of the public key with the complexity of the two algorithms. Results show that security of the cryptosystem is proportional to the size of the public key, i.e., if we increase the size of the public key, the operational complexity of the decoding attacks increases. With current computing power, the key size with complexity higher than 2^{90} is considered practically not feasible. These results show that the GPT cryptosystem having key size of around 5Kbits survives the first algorithm attack but to make the GPT cryptosystem completely secure against both of the algorithms proposed in [13], the key size must not be less than 7.5Kbits. This key size is quite big but still a lot smaller than the key sizes of the other two cryptosystems proposed in [14] and [15] which are 500Kbits and 48Kbits respectively.

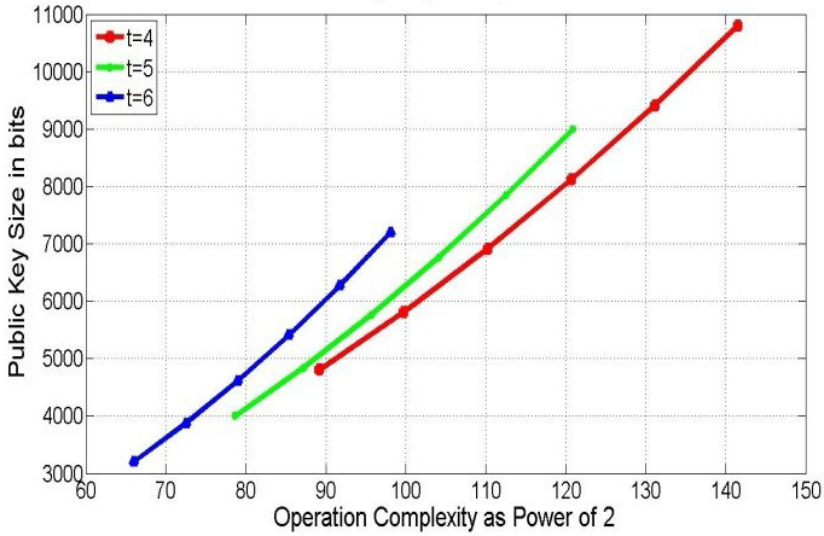

Figure 3. Public Key Size Vs Complexity

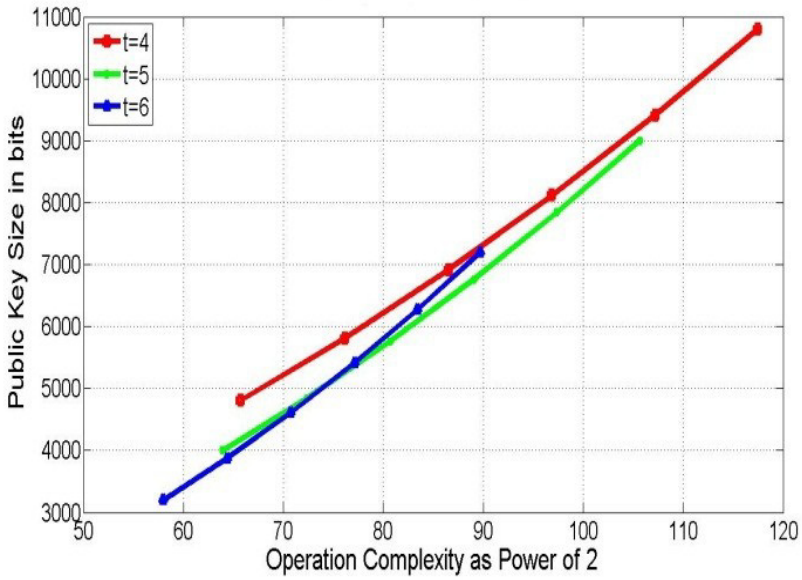

Figure 4. Public Key Size Vs Complexity

B. Information Rate

In any (n, k, d) linear codes, the length n of the code should be kept small for efficient transmission and more throughput whereas the k which is the dimension of the code should be kept large for sending wide variety of messages and better information rate. At the same time, the distance d, which depends on both n and k, should be large to correct more errors. Also, the size of the public key depends on these parameters. There is always a trade-off between choosing these parameters for optimum solution. Figure 5 shows the comparison between the various key sizes and their effect on the information rate for different error correcting capabilities. We can deduce from the results that increasing the throughput of the system will increase the size of the public key and to minimize the key size, we have to compromise on the information rate

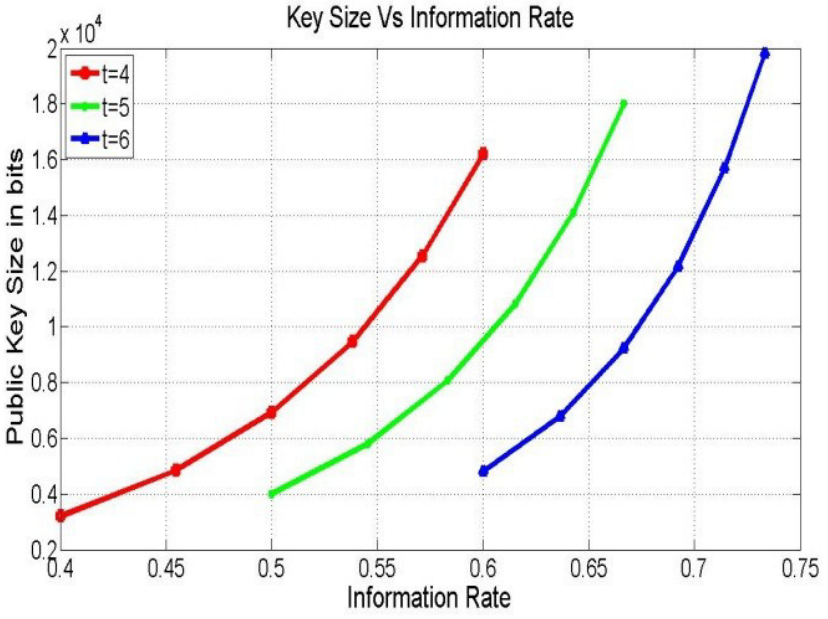

Figure 5. Public Key Size Vs Information Rate

C. Why GPT for Smart Grid

Although number theoretic based cryptosystems such as RSA provide better information rate and fixed key size, GPT belongs to the family of cryptosystems considered to be prime candidates for future cryptography because of their capability to survive even with under the speed and number crunching ability of quantum computing. Encryption and decryption of GPT is also multiple orders faster than RSA and it can provide error protection against channel impairments. By using very low key size, GPT saves cost in terms of the storage space required for storing the keys and high bandwidth communication

required for key transfer. GPT is therefore a smart choice to address security and privacy issues in Smart Grid.

5. CONCLUSION

In this paper, a framework for protecting Smart Grid infrastructure using GPT cryptosystem was presented. As Smart Grid is a very large distributed network connecting various heterogeneous devices and networks, the scheme is not targeted at specific application or service within a network; instead, it provides a method of protecting different critical communication infrastructures of Smart Grid using their geographical location parameters such as latitude and longitude. Examples of applications that could benefit from the scheme include, but are not limited to, communication links between Smart Meters and the substations, software upgrades, and distribution automation. For practical implementation of this scheme, location sensors or any other location device is required.

References

[1] B.Adebisi, A. Haidine, et.al. " IP-centric High rate Narrowband PLC for Smart Grid Applications, IEEE Communications Magazine Feature Topic Issue Power Line Communications for Automation Networks and Smart Grid, vol. 49, No. 12, pp. 46-64, December, 2011

[2] A. Haidine, B. Adebisi, A. Treytl, H. Pille, B. Honary, A. Portnoy, "High-Speed Narrowband PLC in Smart Grid Landscape – State-of-the-art", in Proc. of the 14th International Symposium on Power Line Communications and its Applications (IEEE ISPLC), Udine, Italy, April, 2011, pp.468-473

[3] E.M. Gabidulin, A.V. Paramonov, O.V. Tretjakov, “Ideals over a Non-commutative Ring and Their Application in Cryptology,” in Advances in Cryptology --- Eurocrypt '91, Editor: D.W. Davies, Lecture Notes in Computer Science, No.547, pp. 482--489, Berlin and Heidelberg: Springer-Verlag, 1991.

[4] 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.

[5] R. Overbeck, “A new brute-force attack for GPT and variants,” In: Proc. of Mycrypt 2005, vol. 3517 of LNCS, pp. 5-63, Springer-Verlag, 2005.

[6] Overbeck R., “Brute-force attacks Public Key Cryptosystem Based on Gabidulin codes,” J. Cryptology, 21(2):280-301, 2008.

[7] E. M. Gabidulin, “Attacks and counter-attacks on the GPT public key cryptosystem,” Designs Codes and Cryptography. Vol.48, No. 2, pp. 171-177, Springer Netherlands, August 2008.

[8] E.M. Gabidulin, H. Rashwan, B. Honary, `On Improving Security of GPT Cryptosystems', Int. Symposium on Information Theory, ISIT 2009 pp. 1110-1114, 2009.

[9] E. Khan, E. Gabidulin, B. Honary, H. Ahmed ` Modified Niederreiter Type of GPT Cryptosystem Based on Reducible Rank Codes', Designs Codes and Cryptography, Springer, DOI:10.1007/s10623-012-9757-4, 2012.

[10] Denning, D. E. and MacDoran, P. F., Location-Based Authentication: Grounding Cyberspace for Better Security, Computer Fraud & Security, Vol 2, pp. 12-16. Feb. 1996.

[11] P. Zawadzki, An Improved Estimation of the RSA Quantum Breaking Success Rate, in Proc. NDT (1), pp.234-240, 2010.

[12] H. Dinh, C. Moore, A. Russell: `McEliece and niederreiter cryptosystems that resist quantum fourier sampling attacks'. In Proceedings of the 31st annual conference on Advances in cryptology (CRYPTO'11), Phillip Rogaway (Ed.). Springer-Verlag, Berlin, Heidelberg, 761-779, 2011.

[13] 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), 237-246.

[14] McEliece, R. J., : ``A Public Key cryptosystem Based on Algebraic coding theory”, JPL Deep Space Network Progress Report, Feb 1978, pp114-116

[15] Niederreiter, H.,:``Knapsack-type cryptosystems and algebraic coding theory”, Prob. of Control and Inf. Theory, 15, 1986, pp-19-34 139