Key Generation for Electrical Smart Meters using Hash Functions

Lincoln Kamau Kiarie
Department of telecommunicaiton and Informaiton Engineering
(Jomo Kenyatta University of Agriculture and Technology)
Nairobi, Kenya
kamaulincoln@jkuat.ac.ke

Philip Kibet Langat
Department of telecommunicaiton and Informaiton Engineering
(Jomo Kenyatta University of Agriculture and Technology)
Nairobi, Kenya
kibetlp@jkuat.ac.ke

Christopher Maina Muriithi
Electrical and Power Engineering Department
(Murang'a University of Technology)
Nairobi, Kenya
cmmuriithi@mut.ac.ke

***Abstract*—The electrical power grid has been undergoing improvements to turn it into a more efficient system known as the Smart Grid. A key element in this upgrade is the Smart Meter. A compromise on the cyber-security of the smart meter would have far-reaching effects affecting millions of users. Effective cryptography is needed to protect it. This work looks at how secret keys for use in encryption can be generated through the use of hash functions. An analysis is carried out on 100 generated keys to see if they have an obvious relationship. The average difference in the bits of the keys were found to be 49.578% indicating strong independence between the keys. This method would thus be a good approach of producing 128 bit keys for use in encryption.**

Keywords— Cyber security, Cryptography, Key Generation, Key Management, Smart grid, Smart meters

I. INTRODUCTION

As demand for electricity rises globally, it is crucial that the power generated is transmitted and utilized efficiently. Over the last few years, researchers and governments have been working to transform the traditional electrical grid into a more efficient system known as the Smart Grid. By integrating better two-way communication, Smart Grid allows for improved monitoring, better fault detection and correction, integration of renewable sources and a myriad of other benefits. As a result, utility companies incur fewer losses, are better aware of the situation at hand, need less manpower for data collection and can thus provide better services to their customers. Customers on the other hand receive more reliable power with faster recovery after blackouts. They save on overall power costs and can resell the extra power they produce (e.g. via solar) to the utility.

A key component of the Smart Grid is the smart meter. Smart meters not only provide operations and billing data at a faster rate than conventional meters, but they also offer value-added services to customers [1][2]. A contrast between typical energy meters and smart meters is shown in Figure 1(a) and (b).

In spite of all their benefits, advanced communication in Smart Grid introduce the major vulnerability of cyber security threats [3]. With rising automation and interconnectivity, cyber attackers have spread their targets from financial institutions to a wider range of systems. Smart Grid is an inviting target to malicious attackers for at least three reasons. First, most components of the electrical grid were developed when the level of interconnectedness was much lower. This means that even the basic security measures needed to survive in our digital world had not put in place, making the task of attackers quite easy.

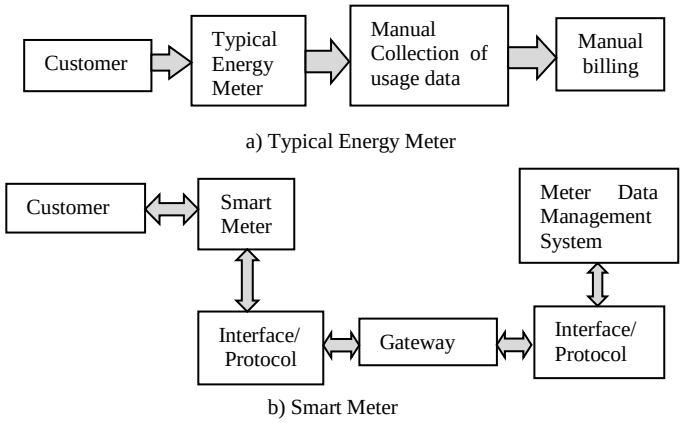


Figure 1 (a) Conventional energy meter and (b) smart meter architecture

Second, our heavy dependence on electricity means that an attack by a cyber-terrorist would have far-reaching effects. An attack affecting hospital equipment would endanger lives, telecommunications would be paralysed by disrupting power in the backbone architecture and tampering with traffic control systems would lead to severe traffic jams.

Third, data mining of electrical consumption data can reveal lifestyle habits of users and lead to privacy leakage at a time when there is growing awareness of the dangers [4]. Using only the overall electrical usage pattern, it is possible to extract information on when people sleep, leave the house, shower, watch TV, which gadgets they have and other lifestyle habits [5].

Securing our electrical grid requires carefully applied cryptography. Advanced solutions from the larger telecommunication and ICT disciplines may not be applied directly on Smart Grid, which is largely composed of light-weight devices such as smart meters. Additionally, unlike most internet traffic, some sections of Smart Grid are time-critical with very stringent latency requirements [6]. Cryptography is generally computationally involving and can slow down the system. Thus, in the context of Smart Grid, a compromise is needed between speed and security.

One major attempt at securing millions of smart meters in Europe was based on a protocol known as Open Smart Grid Protocol (OSGP) [7]. It applied the popular RC4 (Rivest Cipher 4) encryption but was found to be weak and as a result was broken [8], [9]. It was then recommended that the Advanced Encryption Standard (AES) be used in the next version of OSGP. *Spritz encryption* has potential to work well in the context of smart meters [10]. Great care is crucial in applying cryptography to protect our systems by having a suitable encryption method.

It would at first seem that having an undisclosed encryption algorithm would help solve improve security, but history has shown that this approach does not work well [11]. Through reverse engineering, extensive analysis, or a breach of trust, the details of an encryption method will sooner or later be discovered. The better approach is to make the details of an encryption method public and have as many people as possible trying to break it. This was how AES was selected and has proven itself to be a very strong encryption algorithm [12]. Only after surviving multiple attempts to break it by practitioners, researchers and hobbyist does the confidence level for using an encryption method rises. The secrecy of such a system is not based on hiding its details, but on the secrecy of the *key* being used. As long as the cryptographic key is unknown to attackers, a well-designed encryption method is secure.

Regardless of how good an encryption method is, once the key is discovered by an attacker, there is no security left. The best cryptographic methods are practically impossible to break, but they can all be bypassed if the key is leaked. This paper looks at the issue of key generation in the context of smart meters. The number of keys needed to secure all the smart meters in a typical country's grid is in the millions. A scalable solution of generating keys is essential for the security of the Smart Grid.

In this paper, we propose the use of secure hash functions for generating cryptographic keys for use in encryption. The keys can then be stored in an encrypted form within the smart meters while the utility securely keeps its copies. A pseudorandom generator can help generate keys in bulk [13]. This work proposes a method that uses *hash functions*, which are good at producing pseudorandom numbers.

The rest of this paper is organized as follows: Section II gives a brief background on cryptography, Section III describes our key generation procedure, Section IV gives an analysis of the results obtained, and Section V concludes the paper.

II. CRYPTOGRAPHY BACKGROUND

Encryption is a cryptographic procedure that ensures confidentiality between a message sender and a receiver. Encryption is classified as either symmetric or asymmetric, based on the method used in applying the key(s). In symmetric encryption, a common key is shared between the two and must be kept secret from anyone else. In asymmetric, a pair of keys (one public another private) is used and can provide other security features depending on the order of application. Since smart meters have limited computational abilities, when the goal is secrecy, symmetric encryption is preferred because it is less computationally involving [13]. After a suitable symmetric encryption algorithm is chosen, the next major concern is how the key will be generated.

Hash functions are algorithms which take an input of variable size and produce an output with a fixed number of bits as shown in Figure 2 [13]. The output is known as the *message digest* and varies depending on the input. Several hash functions exist, but the most reliable and widely used belong to a family of algorithms developed known as Secure Hash Algorithm (SHA) [14]. SHA hash functions are classified according to the size of the output they produced. The special properties of hash functions have the potential to help in generating keys that are very likely to be unique [13].

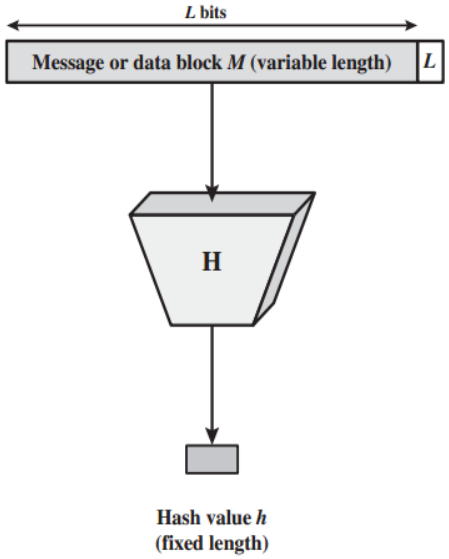


Figure 2 Description of a general hash function, H

To examine if the keys produced from adjacent meter numbers are related, the difference in bits can be examined. A large number of similar bits indicate a high possibility of an attacker being able to deduce the key and thus the key generation method is insecure.

A useful quantity in comparing keys this way is the *hamming distance*. It is the number of corresponding bits that are different between two binary values of equal length [15].

Hamming distance is denoted by $d(\mathbf{v}_1,\mathbf{v}_2)$. As an example, if $\mathbf{v}_1 = 1101$ $\mathbf{v}_2 = 1010$, then the hamming distance is $d(\mathbf{v}_1,\mathbf{v}_2) = 3$, since there are 3 positions where the two are different (i.e. at the 2^{nd}, 3^{rd} and 4^{th} bits). If only a few bits are different between two adjacent keys, the generation method is insecure since it can provide an attacker with information on finding other keys. If all the bits were different, this would imply a bitwise negation of the key (with all 0s interchanged with 1s and vice versa) and would also help an an attacker in identifying a pattern. However, if the hamming distance is approximately half the length of the two keys, the generation technique can be considered secure.

III. KEY GENERATION PROCEEDURE

The generation of a key is a very important aspect of securing a system and is comparable to coming up with a good password. To keep out attackers who would want to use trial and error, it must be long enough. It must also be difficult to predict and weak passwords are discouraged as they form easy targets. Similarly, automated key generation must avoid statistical properties that a clever attacker might identify.

To secure a system from an attacker who attempts every possible key (i.e. brute force), a key of 128 bits in length is currently considered adequate. This would result in 2^{128} (approximately 3×10^{38}) possible options, which are impractical to test. At 1 billion test per second would take 5.4×10^{21}years to break such a key. 128 bits is also a good choice since a number of symmetric ciphers, including both AES and RC4, support this length of key. The procedure described here will generate keys of 128 bits in length but it can be modified to cater for other key lengths.

Our procedure provides a method that can be used to create many keys which are unrelated to each other, thus

making it hard for an attacker to guess the right key. This would even protect other keys from an attacker who was somehow able to obtain the key of one meter.

To demonstrate our technique, an 11-digit meter number was used for testing the procedure described below. This number of digits was based on an actual meter number from the Kenyan utility company. To preserve the privacy of the original value, a different number was chosen using a uniform random distribution for the purpose of demonstration. The random number representing our meter number was *24866272461*.

A utility company typically has millions of meters installed at users' premises. Since smart meters already have meter numbers allocated, we propose the use of the hashing to provide a key corresponding to each meter. It should however not be the case that knowledge of the key of one-meter number would result in the recovery of the keys belonging to adjacent meters (e.g. meters with consecutive numbers). To examine if this was the case, keys were generated for the next 100 meter numbers (i.e. from 24866272461 to 24866272561).

The procedure below was used for each of the meter numbers:

1. The 11-digit meter number was expressed as a string of characters.
2. To ensure that the value is not deterministic, the meter number was concatenated (combined) with the time (number of seconds since the epoch, i.e. 1 Jan 1970).
3. The hash value of the above value was computed using SHA-512.
4. The first 128 bits of the hash value obtained were chosen as the key (the one produced by the original meter number will henceforth be called *Key1*).

The results for the first 5 keys are shown in Table 1.

Table 1 Values of first 5 keys generated

Key	Value in hexadecimal notation
Key1	0x457DB2A89DF2D052398AE327158D584B
Key2	0xCF6D81BC52E4D0A54A0B34B454FE1B2A
Key3	0x0E9DBEA35DAD6824DACDDA76014280C0
Key4	0x623B540CD31AFC06C0B18A4FB89F399E
Key5	0x1AB53ED3C3700DE6807188E98B8B9324

By observing the results, most of the digits appear to be different. The following section uses the hamming distance to provide an objective test of the difference in the bits.

IV. Analysis

Using the above procedure, it is important that the relationship between the meter number and the corresponding key should not be obvious. If the meter number changes even by a small value, the bits in the key produced should change significantly. If this does not happen, an attacker might find a way of using the key of their meter to compute the cryptographic keys of neighbouring meters.

To test if our technique is resistant to such an attack, the hamming distance between the original meter number and the next 100 was computed. The expected number of bits that should change should be approximately half. For the first meter, the hamming distance between the original meter and the one immediately after it (i.e. *Key1* and Key2) was found to be 58 bits (48.44%), which is fairly close to 50%.

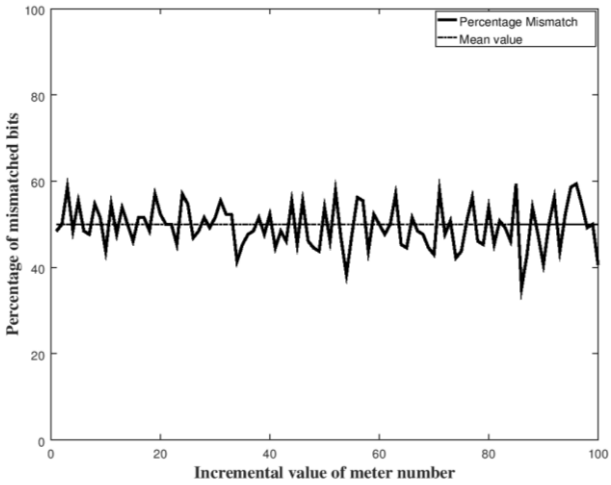


Figure 3 Percentage variation in hamming distance as meter number increases

A plot of the percentage variation in hamming distance between *Key1* and the 100 consecutive keys after it is shown in Figure 3. The average value was 49.578%. This is a good sign that proposed technique produces keys that are difficult for an attacker to predict, even if they have access to a key obtained from a particular meter number.

V. Discussion and Conclusion

This work proposes a reliable method for generating a large number of keys that do not produced related keys. The method described can produce 128 bit keys which are compatible with many encryption ciphers, including AES.

It is worth noting that anyone who wants to implement this method in practice should modify it since with the publication of this work, the description is accessible to any attacker. Modification can be done in several ways. The hash function can be used recursively, a different hash function can be used, a different subset of the hash can be selected for keys (e.g. the last 128 bits), etc. Whichever method is chosen for creating they keys must be kept secret, otherwise the encryption process will be rendered useless.

The method given here can be used to examine the keys generated to test if they have desirable properties that would make them more secure from attack.

Acknowledgment

The authors would like to thank Jomo Kenyatta University of Agriculture and Technology for making this work possible.

References

[1] S. S. S. R. Depuru, L. Wang, and V. Devabhaktuni, "Smart meters for power grid: Challenges, issues, advantages and status," *Renew. Sustain. Energy Rev.*, vol. 15, no. 6, pp. 2736–2742, 2011.

[2] M. R. Asghar, G. Dan, D. Miorandi, and I. Chlamtac, "Smart Meter Data Privacy: A Survey," *IEEE Commun. Surv. Tutorials*, pp. 1–1, 2017.

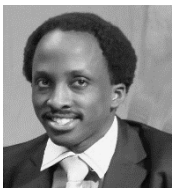
[3] A. Humayed, J. Lin, F. Li, and B. Luo, "Cyber-Physical Systems Security—A Survey," *IEEE Internet Things J.*, vol. 4, no. 6, pp. 1802–1831, 2017.

[4] G. Eibl and D. Engel, "Influence of data granularity on smart meter privacy," *IEEE Trans. Smart Grid*, vol. 6, no. 2, pp. 930–939, 2015.

[5] K. Zhang, J. Ni, K. Yang, X. Liang, J. Ren, and X. S. Shen, "Security and Privacy in Smart City Applications: Challenges and Solutions," *IEEE Commun. Mag.*, vol. 55, no. 1, pp. 122–129, 2017.

[6] W. Wang and Z. Lu, "Cyber security in the Smart Grid: Survey and challenges," *Comput. Networks*, vol. 57, no. 5, pp. 1344–1371, 2013.
[7] Protocol, "ETSI GS OSG 001: Open Smart Grid Protocol (OSGP)," vol. v1. 2012.
[8] K. Kursawe and C. Peters, "Structural weaknesses in the open smart grid protocol," *Proc. - 10th Int. Conf. Availability, Reliab. Secur. ARES 2015*, pp. 1–10, 2015.
[9] P. Jovanovic and S. Neves, "Practical Cryptanalysis of the Open Smart Grid Protocol Dumb Crypto in Smart Grids Smart Grids," *Int. Work. Fast Softw. Encryption*, pp. 297–316, 2015.
[10] L. K. Kiarie, P. K. Langat, and C. M. Muriithi, "Application of Spritz Encryption in Smart Meters to Protect Consumer Data," *J. Comput. Networks Commun.*, vol. 2019, pp. 1–10, 2019.
[11] H. Abelson, K. Ledeen, and H. Lewis, *Blown to Bits: your life, liberty, and happiness after the digital explosion*. Addison-Wesley Professional, 2008.
[12] J. Daemen and V. Rijmen, *The Design of Rijndael*. Springer-Verlag, 2002.
[13] W. Stallings, *Cryptography and Network Security Principles and Practice*, 5th ed. Prentice Hall, 2011.
[14] F. PUB., "Secure Hash Standard," 2012.
[15] W. Stallings, *Data and computer communications*, 8th ed. Prentice Hall, 2007.

BIOGRAPHIES



Lincoln Kamau Kiarie received his BSc. Degree in Telecommunications and Information Engineering, BSc. In Electrical and Electronics Engineering and MSc in Telecommunication Engineering from Jomo Kenyatta University of Agriculture and Technology (JKUAT). His current research interests include Smart Grid Communications, Cyber Security, Cryptography and Image Processing.

Philip Kibet Langat has a PhD in Electronic Engineering from Stellenbosch University, South Africa, an MSc. in Telecommunication Engineering and a BSc. in Electrical & Electronic Engineering, both from JKUAT. His research interests include applied electromagnetics, radiation and scattering problems, RF and EM computations, EMI/EMC, RFI mitigation, and multi-antenna communications and application.

Christopher Maina Muriithi is an Associate Professor in Electrical Engineering, a professional Electrical Engineering Technologist specializing in Power Systems Stability, Artificial Intelligence Applications and Renewable Energy Technologies. He is also a Qualified Solar Photovoltaic Installer and Trainer of Trainers in Solar Home Systems and Grid connected Renewable energy hybrid systems. Current research interests include Renewable Energy Technologies, Voltage Stability and Smart grids.