*International Journal of Engineering & Technology*, 7 (4.41) (2018) 101-104

International Journal of Engineering & Technology

*Website: www.sciencepubco.com/index.php/IJET*

***Research paper***

# A Security Paradigm for IoT

**Priyanka Anurag Urla[1], Smitha N Pai[2]*, Girish Mohan[3]**

*[1,2,3] Dept. of ICT, Manipal Institute of Technology, Manipal Academy of Higher Education, Manipal, Karnataka, INDIA.*
**Corresponding author E-mail:smitha.pai@manipal.edu*

## Abstract

Current technology is offering lot of services at the finger tips. IoT is becoming a part of daily life without us being actually aware of the vulnerability to attacks. In this paper issues with choice of security algorithm is discussed. A secure computational approach is designed to overcome the issues with security.

***Keywords***: *IoT; Security; threats; attacks;*

## 1. Introduction

The internet has been around for a while, largely for the betterment of the human lives. The data, images, recordings, games, books and commerce has been created for the people, by the people and about the people. The internet is one of the most important and transformative technologies ever invented. The internet is not just about connecting people, it's about connecting things (devices), called as Internet of Things (IoT). A network of things connected to internet, capable of data collection and data exchange operating with the aid of sophisticated embedded sensors, together constitute IoT. This Internet of Things refers to connecting objects to the Internet. The objects refer to physical or virtual devices as in hand held devices, sensing nodes, gateway nodes etc. The data collected can be from X-ray, biomedical devices, pharmacy dispensing, pacemakers, which are all a part of IoT. Various applications and their vulnerability to attack is discussed in depth in [1]. The analysis of the data stored and sharing of large amount of monitored data could be at security risks[2]. This is possible in wearable devices as used by new born or the elderly health monitoring system. Tracking of pet's, human or vehicles using GPS and misguiding their directional movement could be of threat. The devices which provide convenience can also be a threat. In a rush to deploy the new technology the security of the devices was not adhered to. A compromised device can collect personal information, launch denial of service attack, life-threatening attacks. Medical data should be protected against disclosure of information, alteration, deletion or unauthorized access.

## 2. Literature Review.

Security and efficiency go hand in hand. They define each other in a very prominent manner. Security issues are highly disturbing and suggestive solutions have to be obtained to protect data in IoT environment from various sources like sensors and application[1][2]. The architecture of IoT being a very crucial aspect and the study of the communication protocols[3][4] existing in the IoT environment gives us an open platform to takes measures in the security area and also to render new solutions to mitigate security threats.[5][6] Gentry's methods are employed to convert a plain "somewhat homomorphic" encryption algorithm utilizing elementary modular arithmetic[7]. The main aim of the methodology is that it is conceptually simple. A new modulus technique of switching for the Dijk, Gentry, Halevi and Vaikuntanathan's (DGHV) methodology is described along with the foremost objective which is a compression technique where there is a reduction in the size of public key of DGHV's fully homomorphic scheme[8][9] .As an alternative to traditional cryptographic  namely RSA, the use of elliptical curve cryptography has an enormous amount of advantage in terms of performance in the wireless framework where-as the voluminous and vigilant study of the software administration on the workstations of the elliptical curves over binary fields is carried out and recommended by NIST[10][11]. For the reduction of the key size and to improve the efficiency, a new strategy for the compression of public key size is advocated using. Fully homomorphic encryption (FHE). It is  having its origin from quadratic parameters along with corrections incorporated with it. To operate on real numbers an advanced FHE scheme is proposed. This methodology gives us the advantage of accuracy along with efficiency[12][13]. An encryption scheme is formulated where it differentiates the credentials of private and public keys. End to end secure  connectivity can be obtained with protocols and apt architecture [14][15].

The work discussed in the related field deal with encryption in different domains where data is fragile. The current paper addresses the issues of security in the IoT environment providing enhanced security using multiple levels.

## 3. Methodology

Data is collected from sophisticated sensors. This is achieved by calibrating the sensors with Arduino/raspberry pi 3 boards. Figure 1 gives the flow chart of the project. For experimental purposes DHT11 which is a temperature and Humidity sensor and soil moisture sensor is used to collect data.
After the collection of the data. data is encrypted using the encryption process, by using the security paradigm as shown in  Figure 2. Using three security algorithms i.e., light weighted encryption technique, secured hash function, elliptical curve cryptography (ECC) and fully Homomorphic encryption (FHE) security issue

has been addressed. XOR is used to secure the data from brute force attacks. Secured hash function is used to ensure that the data is not modified.

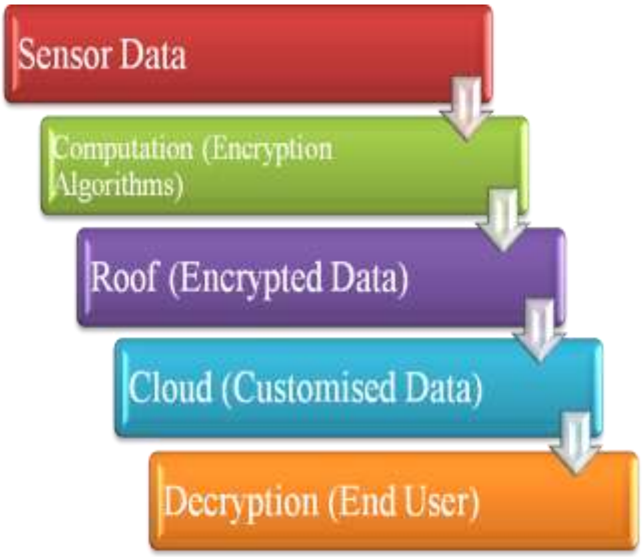


**Figure. 1:**Flow Chart.

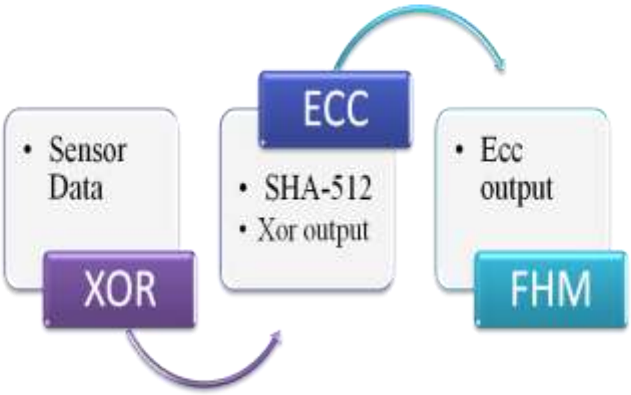


**Figure 2:** Computational flow.

The main focus of this project lies on two important algorithms providing multilayered security using ECC and FHE. The properties of Elliptical curves are used to generate the keys in ECC.

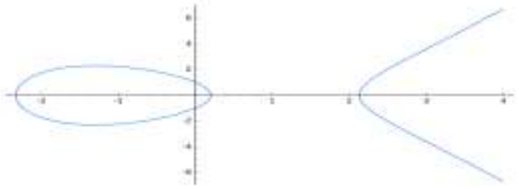


**Figure 3:** Elliptic curve of ECC.

Elliptical curves have two main properties, which are foundation for computing the data, i.e., A) with horizontal symmetry, at any point on the curve, it gets be reflected over the x-axis and it will still be on the curve. B) Any non-vertical line that is drawn on the curve will intersect this curve at the max, three times. Using these characteristics, the key generation, encryption and decryption are done using the formulas from equations (1) to (3).

Key generation:

$$Q = d * p \tag{1}$$

Where, Q is public key, P is the point on curve and d is the private key respectively.

Encryption formula:

$$C1 = K * p \,, C2 = M + K * Q \tag{2}$$

Decryption formula:

$$M = C2 - d * C1 \tag{3}$$

The most desirable feature of multilevel encryption is that it must be able to perform computations on encrypted data without the necessity to decrypt it. This is achieved by the using FHE.
FHE is an operation performed on a set of cipher texts such that decrypting the result of the operation is the same as the result of some operation performed on the plaintexts. The encryption and decryption process uses the equations (4) to (5)

Encryption formula:
Cipher text for $m$ € {0,1}

$$C = q.p + 2r + m \tag{4}$$

Where r and q are random integers, p is the secret key.

Decryption formula:

$$(C \; mod \; p) mod \; 2 = \; m \tag{5}$$

After the whole decryption process the data is mounted on to the Roof. Roof is an amalgamated networking and computational archetype for IoT, which is invariably accessible for real time context building, on site operation facilitation. As a software platform roof is implemented on various devices that proxy the things and also their IoT services to the remaining part of the world.

Data is collected at the local server. Values above and below the required threshold are being updated on to the cloud on encryption. The end user can access this data through an end device as and when needed (continuously or at specific time interval, or averaged data within some interval). An app is designed to provide the user the ease of access. When the end user accesses the data it is decrypted in the same sequence as it was encrypted before.

## 4. Implementation.

Data is read and collected from sensors interfaced with raspberry Pi 3/ Arduino board and stored in the system in the form of a file. Application specific sensors can be used. Other devices which can send text, image etc. can also be taken into account. In the current scenario, humidity, temperature and soil moisture content detector are used. The baud rate chosen in this example is 3600baud. Figure 4A and 4B illustrate the set up of sensors with Arduino. The Data collected is shown in the figures 5 and 6. Experiments are carried out using Raspberry Pi 3 board as well.

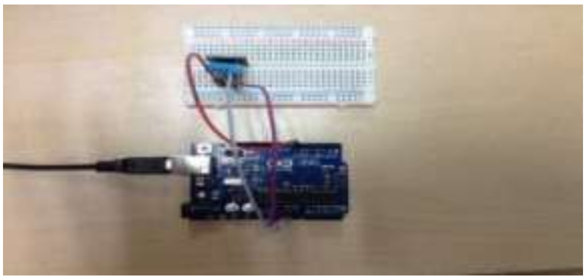


**Figure 4: A.**Temperature and humidity sensor connectivity with Arduino.

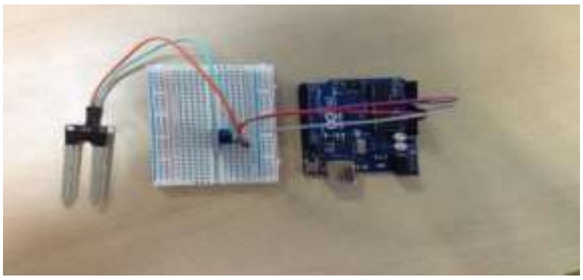
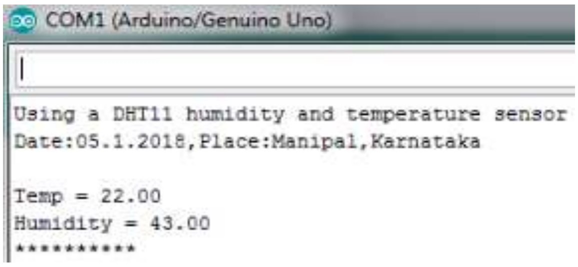


**Figure 4: B.** Moisture sensor connectivity with Arduino.

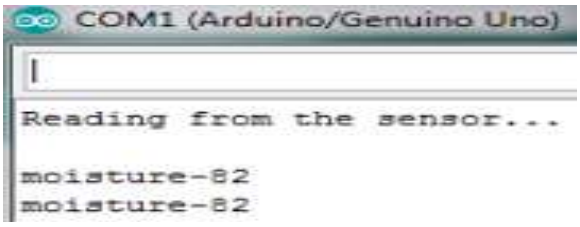


```
COM1 (Arduino/Genuino Uno)

Using a DHT11 humidity and temperature sensor
Date:05.1.2018,Place:Manipal,Karnataka

Temp = 22.00
Humidity = 43.00
**********
```

**Figure 5:** Measured temperature and humidity sensor data.



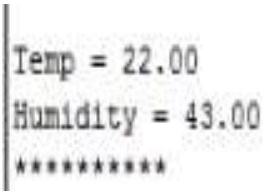
```
COM1 (Arduino/Genuino Uno)

Reading from the sensor...

moisture-82
moisture-82
```

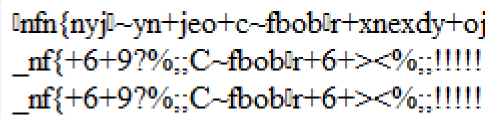
**Figure 6:** Moisture sensor data.

Four levels of encryption are applied to the collected data. As an example the following is the input, figure 7.



```
Temp = 22.00
Humidity = 43.00
**********
```

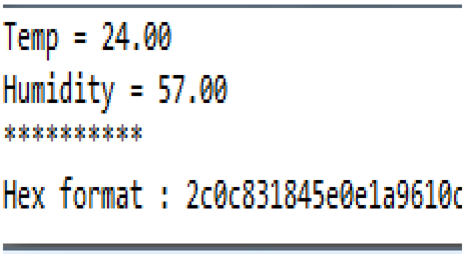
**Figure 7:** Input to the algorithms.



```
▯nfn{nyj▯~yn+jeo+c~fbob▯r+xnexdy+oj
_nf{+6+9?%;;C~fbob▯r+6+><%;;!!!!!
_nf{+6+9?%;;C~fbob▯r+6+><%;;!!!!!
```

**Figure 8:** Level 1 encrypted data.

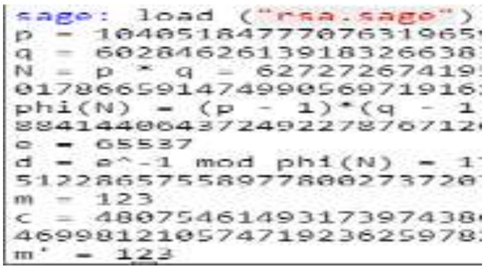
At the first level a simple light weight encryption standard is used whose output is as shown in figure 8. The input to the file is the humidity and temperature information. The security at the next level is obtained using the secure hash function whose input is as from figure 8. The output generated by the SHA algorithm as shown in figure 9.



```
Temp = 24.00
Humidity = 57.00
**********

Hex format : 2c0c831845e0e1a9610c
```

**Figure 9:** Second level of encryption.



```
sage: load ("rsa.sage")
p = 1040518477707631965
q = 6028462613918326638
N = p * q = 627272674195
0178665914749905697191 6
phi(N) = (p - 1)*(q - 1
8841440064372492278767 12
e = 65537
d = e^-1 mod phi(N) = 1
51228657558977800273720
m = 123
c = 48075461493173974 38
46998121057471923625978
m' = 123
```

**Figure 10:** ECC Output.

The output generated by the hash function is verified for modification and the output of the light weight encryption is fed to elliptical curve cryptosystem. The Elliptical curve cryptosystem output is as shown in figure 10 and is fed as the input to the last and final level of security; fully homomorphic encryption algorithm.

As the computation process is completed and the data is encrypted, the obtained encrypted data is put on to the roof, shown in figure 11,12 and 13.

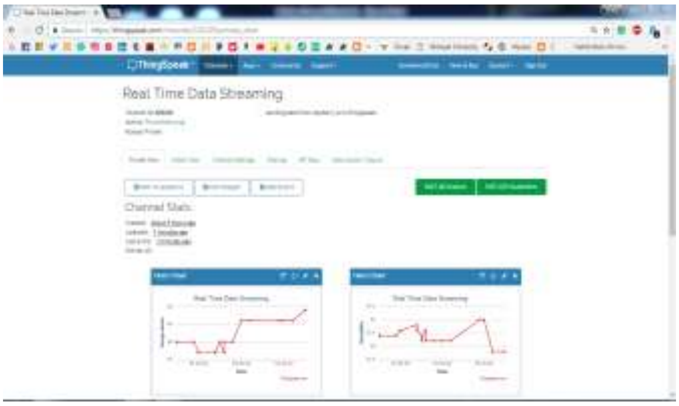


**Figure 11:** Screen shot of data on the roof.

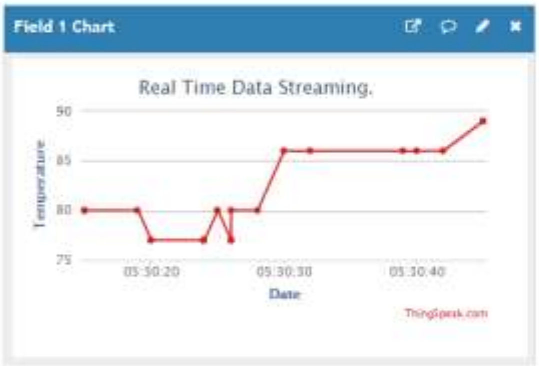


**Figure 12:** Screen shot of uploaded temperature data

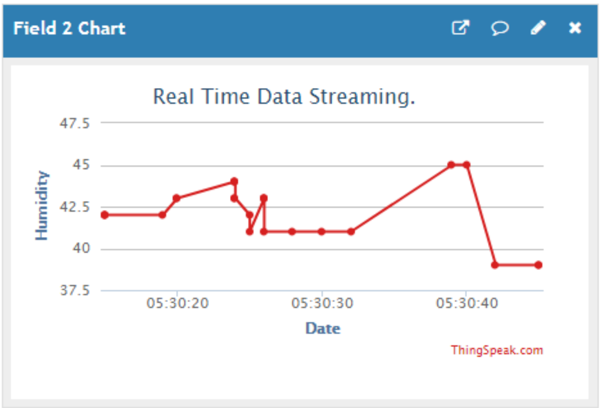
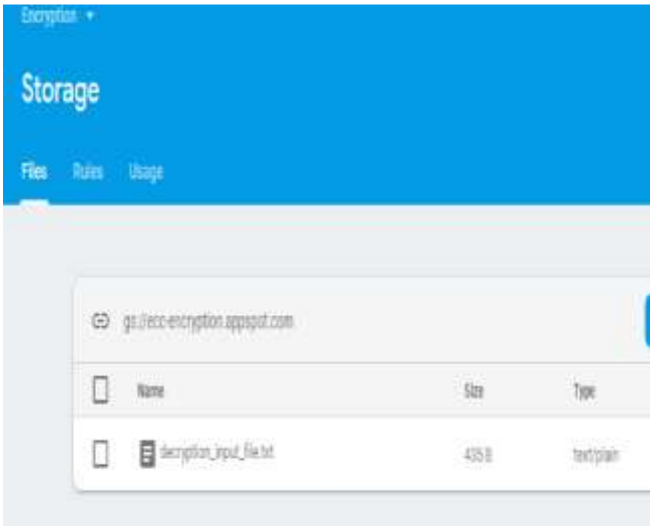


**Figure 13:** Screen shot of uploaded humidity data

The data above or below the threshold is then transferred on to the cloud as shown in the figure 14.

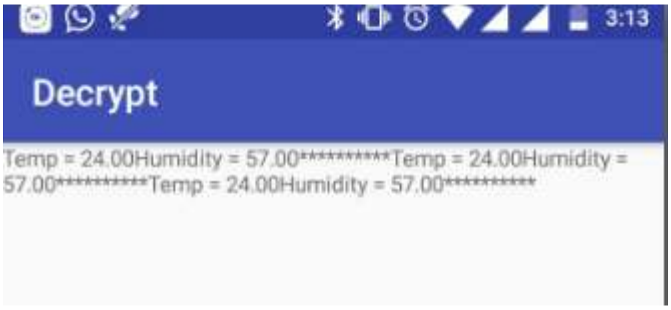


gs://ecc-encryption.appspot.com

| Name | Size | Type |
|---|---|---|
| decryption_input_file.txt | 435 B | text/plain |

**Figure 14:** Data on the cloud.

When the user/client wants to access the data, the decryption process is carried out in the reverse order to retrieve the original data. To make it available to any user an app is developed as shown in figure 15, with the decryption algorithms running in the background providing the user the availability of data by a single click.



```
Temp = 24.00Humidity = 57.00**********Temp = 24.00Humidity =
57.00**********Temp = 24.00Humidity = 57.00**********
```

**Figure 15:** Screen shot of the app

## 5. Conclusions

A system of multilevel security for the data collected from IoT devices is proposed in this paper. In doing so, the security level is enhanced mitigating the attacks i.e. masquerade, man in the middle, data modification, denial of service etc. The first level of encryption ensures security from brute force attacks. The second level of encryption prevents data being modified. The third level of encryption increases security using public key for encryption operation. The key generation grows exponentially. Fourth level ensures that other attacks are not possible as it employs relatively a shorter key. The short key is faster, entails less computation of power and ensures transfer of the data faster. An app is developed for ease of access of data for the user from the cloud. The cloud obtains the customized data from the roof. The roof collects the data as per the required data rate from the sensing device after the first level of encryption. The whole process ensures that the data is securely transmitted and received without any modification of data during the process.

## Acknowledgement

The authors would like to thank the HOD of information and communication technology, MIT, Manipal. Guide and Various faculty and staff who help us with the hardware simulation etc. Lastly to the resource persons who shared valuable information in workshops and seminars related to IoT.

## References

[1] H. Suo, J. Wan, C. Zou, and J. Liu, “Security in the internet of things: A review,” in *Proceedings - 2012 International Conference on Computer Science and Electronics Engineering, ICCSEE 2012*, 2012.

[2] G. Gan, Z. Lu, and J. Jiang, “Internet of Things Security Analysis,” in *2011 International Conference on Internet Technology and Applications*, 2011.

[3] I. Yaqoob *et al.*, “Internet of Things Architecture: Recent Advances, Taxonomy, Requirements, and Open Challenges,” *IEEE Wirel. Commun.*, vol. 24, no. 3, pp. 10–16, 2017.

[4] Y. Pan, X. Cheng, and C. Xia, “World Symposium on Mechanical and Control Engineering ( WSMCE ) Research and Design of Lightweight Encryption for Mqtt Protocol,” vol. 1, no. 2, pp. 143–145, 2017.

[5] K. Zhao and L. Ge, “A survey on the internet of things security,” *Proc. - 9th Int. Conf. Comput. Intell. Secur. CIS 2013*, pp. 663–667, 2013.

[6] J. Granjal, E. Monteiro, and J. Sa Silva, “Security for the Internet of Things: A Survey of Existing Protocols and Open Research Issues,” *IEEE Commun. Surv. Tutorials*, vol. 17, no. 3, pp. 1294–1312, 2015.

[7] M. Van Dijk and C. Gentry, “Fully homomorphic encryption over the integers,” *Adv. Cryptology– …*, pp. 1–28, 2010.

[8] J.-S. Coron, D. Naccache, and M. Tibouchi, “Public Key Compression and Modulus Switching for Fully Homomorphic Encryption over the Integers.”

[9] X. Yi, R. Paulet, and E. Bertino, *Homomorphic Encryption and Applications*. 2014.

[10] K. Lauter, “The advantages of elliptic curve cryptography for wireless security,” *IEEE Wirel. Commun.*, vol. 11, no. 1, pp. 62–67, 2004.

[11] D. Hankerson, J. L. Hernandez, and A. Menezes, “Software implementation of elliptic curve cryptography over binary fields,” in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2000.

[12] L. Chen, M. Lim, and Z. Fan, “A Public Key Compression Scheme for Fully Homomorphic Encryption Based on Quadratic Parameters With Correction,” *IEEE Access*, vol. 5, pp. 17692–17700, 2017.

[13] K. Gai, M. Qiu, Y. Li, and X.-Y. Liu, “Advanced Fully Homomorphic Encryption Scheme Over Real Numbers,” in *2017 IEEE 4th International Conference on Cyber Security and Cloud Computing (CSCloud)*, 2017.

[14] P. Sha and Z. Zhu, “The Modification of RSA Algorithm to Adapt Fully Homomorphic Encryption Algorithm in Cloud Computing.”

[15] Y. Chahid, M. Benabdellah, and A. Azizi, “Internet of things security,” in *2017 International Conference on Wireless Technologies, Embedded and Intelligent Systems, WITS 2017*, 2017.