An AES Based Secure Data Transmission in Internet of Thing

Deepika Khambra
Dept. of Computer Eng, U.I.E.T Kurukshetra University, Kurukshetrra, India

Poonam Dabas
Assistant Professor, Dept. of Computer Eng, U.I.E.T, Kurukshetra University, Kurkukshetra, India

Abstract – Internet of Things (IoT) is a collection of billion of devices interlinked together to share information between them. As numbers of devices were increases the chances of security violation also increases because of presence of malicious nodes. Too securely transmission between two devices is challenging task. There are numbers of existing cryptography algorithms are available such as DES, RSA and AES. In this paper we provide secure data transmission mechanism in which we uses AES to increases the security of data. To implement propose mechanism we uses MATLAB and to analyze performance we uses metrics like execution time and throughput.

Index Terms –: Internet of Things (IoT), Sensor, Attacks, Security and AES.

1. INTRODUCTION

The IoT constitutes a system of interrelated physical objects with ability to communicate with each other and also with their external environment. IoT has various applications in a number of areas including but not limited to health care, smart homes, smart cities, smart transport system, smart manufacturing and other industries [1]. The Internet of Things (IoT) is a vital point in technology industry, strategy, and designing circles and has progressed toward becoming feature news in both the strength press and the well known media. This technology was epitomized in a broad range of organized items, frameworks, and sensors, which exploit progressions in processing power, hardware scaling down, and arrange interconnections to offer new abilities not already conceivable. A wealth of gatherings, reports, and news articles talk about and face off regarding the planned effect of the "IoT upset"— from new market openings and plans of action to worries about security, protection, and specialized interoperability. The vast scale execution of IoT devices guarantees to change numerous parts of the system we live [2].

2. RELATED WORK

Shah *et al*. [3] proposed Home automation system based on loT used Reed Solomon codes to moderate risks and so enhancing security by providing error correction scheme both in the communication channel as well as the data store. Venkata *et al*. [4] discussed a new light weight transport method (LWTM) which used existing Advanced Encryption Standard-Counter (AES_CTR) and Advanced Encryption Standard- Cipher block chaining (AES_CBC) algorithms in a approach to reduced computational time drastically for IoT applications involving large data. Horton *et al.* [5] focused on the examination and enhancement of security between IoT enabled robots, specifically in this project, Turtle Bots, and the cloud infrastructure supported by them provided a combined set of security best practices for robotic file systems and communications. Kuusijarvi *et al.* [6] discussed the current security challenges of loT devices and proposed a solution to secure these devices via a trusted Network Edge Device. Shifa *et al*. [7] proposed lightweight encryption, which was preserve privacy and security of organizations and individuals by addressing the different levels of security required by multimedia applications at different phases in its operation. Dalipi and Yayilgan [8] represented a comprehensive survey of the most recent contributions on security and privacy aspects of IoT applications in smart grid and identify a number of the remaining challenges and vulnerabilities related to security and privacy. Tawalbeh and Somani [9] introduced the IoT concepts, applications, and challenges facing IoT. Then, they presented the recent timing and fault Side Channel Attacks on cryptosystem implemented for the most secure encryption algorithm (AES, ECC, and RSA).

3. PORPOSED MODELLING

3.1 Problem Statement

In IoT secure data transmission between devices is very challenging task. An attacker node may cause information during data transmission and access it to get its own benefits. An attack can be performed by sensing the communication between two nodes which is known as a man-in-the-middle attack. Among all of issues one important issue is the security performance for sending and receiving data on device and other cannot differentiate between the encrypted and unencrypted plain text. Various cryptographic algorithms have been developed that addresses the said matter, but their utilization in IoT is questionable as the hardware we deal in

the IoT are not suitable for the implementation of computationally expensive encryption algorithms.

3.2 Use of AES (Advanced Encryption Standard) Algorithm

AES is symmetric key algorithm. AES performs computations on bytes .AES used 128 bits of a plaintext block as 16 bytes. These 16 bytes are arranged in four columns and four rows. AES is variable depends on the length of key. An AES cipher specify the amount of repetitions of conversion rounds that exchange input that is called plaintext, and the final output is called ciphertext.

3.3 Proposed Algorithm

In this section algorithm for encryption and decryption has been presented with their detail description. To enhance security of private keys different rounds will be performed to generate a private key.

3.3.1 Data Sending

However any data D is sending it will include partition of data in to eight blocks by encrypt in some part size in bit. These parts are then encrypted by using algorithm AES and encrypted parts will be send to the receiver.

The encryption algorithm uses the set of following parameters:

Ey: Encryption, MC: Mix column, SR: Shift Rows, XOR: Exclusive OR operation, K: Key, HX: Hexadecimal, D: Data

Algorithm of Encryption

1. Start
2. Input data for encryption.
3. Transform this data into hexadecimal number.
4. Now perform shift rows operations to transform these hexadecimal numbers into rows and column.
5. Perform mix column transformation to transform each column into a new column.
6. Now add some round key to each column to perform addition of matrix.
7. At last XOR the output of the addition of matrix.
8. Generate encrypted key.
9. End

Description of algorithm

In the proposed algorithm firstly input data is taken for encryption. After that transform this data into hexadecimal number and then perform shift rows operation to transform this hexadecimal number into rows and column for encryption. In Shift rows operation the initial queue is gone unchanged. Every byte of the next string is shift individual near the gone. In the same way, third and fourth queue be shifted. In shift rows transformation is a simple permutation. After that perform mix column transformation to convert each column keen on an original line. In column conversion make original column standards by apply expressions to convert input column. An expression is able to contain variables, function, operator and column from the transformation input. And then add some round key to each column to perform addition of matrix. At last XOR the output of the addition of matrix with key. XOR is exclusive or exclusive disjunction is logical operator that results true only when input is different (one is true, other is false).

3.3.2 Data Receiving

This algorithm used four steps; in the first step, the algorithm decrypts the data D into blocks. in the send step, the shift rows operation perform into rows and column. In the third step, XOR operation performed. In the fourth step, strong cipher key will be generated.

The decryption algorithm uses the set of following parameters:

Dy: Decryption, MC: Mix column, SR: Shift Rows, XOR: Exclusive OR operation, K: Key, HX: Hexadecimal

Algorithm of Decryption

1. Start
2. Input data for decryption.
3. Transform this data into hexadecimal number.
4. Now perform shift rows operation to transform these hexadecimal numbers into rows and column.
5. Perform mix column transformation to transform each column into a new column.
6. Now add some round key to each column to perform addition of matrix.
7. At last XOR the output of the addition of matrix.
8. Generate original text.
9. End

Description of algorithm

In the proposed algorithm firstly input data is taken for decryption. After that transform this data into hexadecimal number and then perform shift rows operation to transform this hexadecimal number into rows and column for decryption. In Shift rows operation the initial queue is gone unchanged. Every byte of the next string is shift individual near the gone. In the same way, third and fourth queue be

shifted. In shift rows transformation is a simple permutation. After that perform mix column transformation to convert each column keen on an original line. In column conversion make original column standards by apply expressions to convert input column. An expression is able to contain variables, function, operator and column from the transformation input. And then add some round key to each column to perform addition of matrix. And at last XOR the output of the addition of matrix with key. XOR is exclusive or exclusive disjunction is logical operator that results true only when input is different (one is true, other is false).

3.3.3 Working of Proposed Work

Figure1 shows system model of proposed work. So to solve this here in this section we presented our proposed mechanism. In proposed mechanism increase throughput and execution time with AES. AES (advanced Encryption Standard) used 128 bit key mechanism to perform encryption and decryption on data before sends one device to another. After the generation of round keys the encryption process can be started. For the purpose of creating confusion and diffusion this process is composed of some logical operations, left shifting, swapping and substitution. The method of encryption is for the first round an array of 128 bit plain text (pt) is first faceted into eight segments of 16bits. As the bits progresses in each round the swapping operation applied so as to diminish the data originality by altering the direct of bits, essentially increasing confusion in cipher text. Bitwise XOR operation is performed between the respective round key Ki. After processing every one key of a strong cipher key will be generated by the combinations of all eight keys.

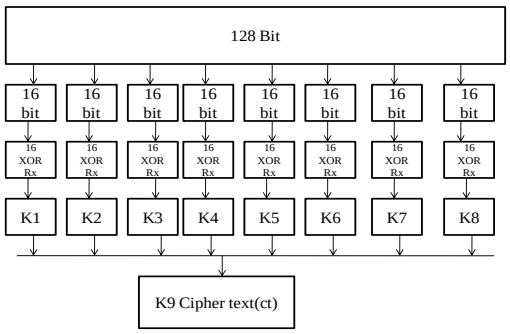

Figure 1 System model of proposed work

4 RESULTS AND DISCUSSIONS

Tool used: to analyze proposed mechanism we use MATLAB. MATLAB is a matrix laboratory. MATLAB is a muti-paradigm numerical computing environment and fourth generation programming language. A programming language developed by MathWorks.

Performance Matrices:

a) Throughput: It depicts amount of message successfully delivered in perspective of whole amount of messages created towards destination within given time.
b) Execution Time: It depicts the whole amount of time taken by device to perform successful transmission of messages in IoT.

Table 1: Simulation Parameters

Parameters	Values
Simulation Area	100X100
Number of Nodes	[100;200;300]
Rounds	50
Initial Energy	0.5 J
Operating System	Windows 7

Table 2: Throughput of without AES and Proposed Mechanism

Number of Nodes	100	200	300
Without AES	1.7102	2.6391	3.0638
Proposed	2.8630	4.2553	5.2632

Fig 2 shows depicts throughput of proposed mechanism and default routing process of IoT without AES which is measured by varying number of nodes from 100 to 300. In proposed mechanism throughput is high as compare to default routing without AES

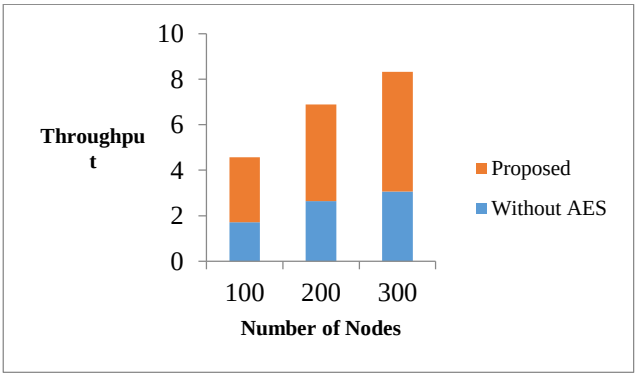

Figure 2 Throughput v/s Number of Nodes

Table 3: Execution time of without AES and Proposed Mechanism

Number of Nodes	100	200	300
Without AES	10.2239	13.2843	18.2878
Proposed	76.5304	80.8851	85.7570

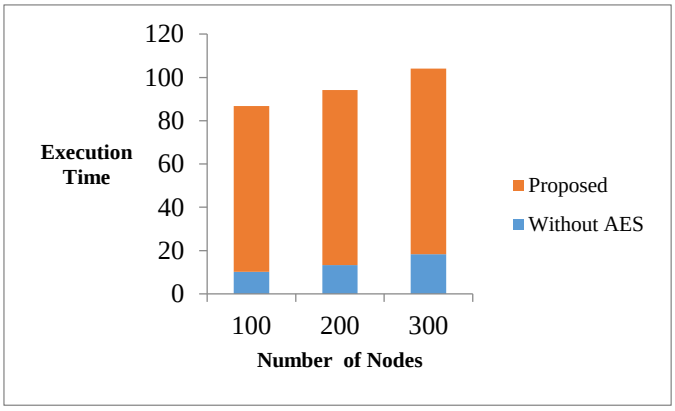

Figure 3 Execution v/s Numbers of Nodes

4. CONCLUSION

Secure data transmission between IoT is a very challenging task. There are numbers of existing algorithms available that provides secure data transmission out of them AES is much secure then other algorithms. On the basis of these algorithms, in this paper we provide mechanism which uses enhanced AES algorithm in which number of rounds or generation of private key increases that will help in generation of more secure encrypted key through which devices can transmit data in secure manner. Results show that in our mechanism throughput of data transmission system increases. In future try to enhanced proposed mechanism and focus on reducing end to end delay occurred during secure data transmission.

REFERENCES

[1] S. Raza, Shafagh, H. Hewage, K. Hummen, R., and T. Voigt " Light Weight Secure CoAP for the internet of Thing" IEEE Sensors Journal, 13(10), 3711-3720

[2] O. SINTEF, Norway, P. Friess EU, Belgium, "Internet of Things–From Research and Innovation to Market Deployment, river publishers' series in communications, 2014.

[3] I. A. Shah, F. A. Malik and S. A. Ahmad, "Enhancing Security in loT based Home Automation using Reed Solomon Codes" IEEE Conference, Chennai India, 2016, pp.1639-1642

[4] S. B. Venkata, P. Vellai, G. D. Verma, A. Lokesh, A. KS and S. S. Sanahapati, "A new light weight transport method for secured transmission of data for Internet of Thing," IEEE Conference, Bangalore India, 2016, pp.1-6

[5] M. Horton, L. Chen and B. Samanta, "Enhancing the Security of IoT Enabled Robotics: Protecting TurtleBot File System and Communication", IEEE Workshop on Computing, Networking and Communications (CNC), Santa Clara CA USA, 2017, pp.1-5

[6] J. Kuusijarvi, R. Savola, P. Savolainen and A. Evesti, "Mitigating loT Security Threats with a Trusted Network Element", 11th IEEE International Conference for Internet Technology and Secured Transactions, Barcelona Spain, 2016, pp.260-265

[7] A. Shifa, M. N. Asghar and M. Fleury, "Multimedia Security Perspectives in IoT", 6th IEEE international Conference on Innovative Computing Technology, Dublin Ireland, 2016, pp. 550-555

[8] F. Dalipi and S. Y. Yayilgan, "Security and Privacy Considerations for IoT Application on Smart Grids: Survey and Research Challenges", 4th International Conference on Future Internet of Things and Cloud, Vlenna Austria, 2016, pp. 63-68

[9] Lo'ai A. Tawalbeh, and T. F. Somani, "More Secure Internet of Thing using Robust Encryption algorithm Against Side Channel Attacks", IEEE Conference, Agadir Morocco, 2016, pp.1-6