International Journal of Research in Engineering, Science and Management
Volume-2, Issue-5, May-2019
www.ijresm.com | ISSN (Online): 2581-5792

Security in IoT using AES Algorithm and Secure Key Transmission using Gmail

Prathamesh More[1], Namrata Phadtare[2], Swamini Pednekar[3], Rajas Karnik[4]
[1,2,3,4]*B.E. Student, Department of Computer Engineering, Shah and Kutchhi Engineering College, Mumbai, India*

***Abstract*: With the advent of the Internet of Things(IoT), many electronic devices are interconnected. With these technological advancements, need has arised for devices to become smarter and transmit data. This can be achieved by securing the devices and communication channels using Encryption algorithms, Authentication, Profile matching, Digital Certificate, Digital Signature, Hashing etc. AES with algorithm produce encrypted code that can be reversible to achieve confidentiality. The main focus is to improve the security of the data being sent by preventing outside attacks. In the given presented system, we are providing security to Medicine Inventory System using AES Encryption algorithm. Data is collected from Medicine Inventory store. It is then encrypted and stored on online database. This data can be viewed on a Web based application.**

***Keywords*: AES-256, confidentiality, encryption, Key, Gmail, authentication.**

1. Introduction

In the given presented application, we are providing security to Medicine Inventory System using AES Encryption algorithm. Cryptography plays an important role in many electronic systems, to ensure the security of the confidential data being used especially when the medium used for the communication is unreliable and error prone. Encryption is the process of using an algorithm to transform information to make it unreadable for unauthorized users. This encoded data may only be decrypted or made readable with a key. Symmetric-key and asymmetric-key are the two primary types of encryption. Encryption is essential for ensured and trusted delivery of sensitive information [4]. Decryption is generally the reverse process of encryption. It is the process of decoding the data which has been encrypted into a secret format [3]. Medicine Inventory Management, in Government Hospitals is mostly done using pen-paper method. It is observed that in these hospitals most of the medicines are misplaced by the local authorities for monetary benefits. We are trying to provide security by automating this system as much as possible that will reduce human intervention and hence reduce the vulnerabilities. RFID tags can be attached to medicines so that we can keep a track on the entry and exit of the medicines. The data from the RFID tags is read by the Raspberry Pi. This information is encrypted using encryption algorithm and stored in real-time cloud server. On the other end the Web based Application will read the data. Since this data is in encrypted format, Web based Application will first decrypt it on request and then display it. Access to the data will be provided on the basis of hierarchy. Data can only be viewed, not manipulated.

2. AES algorithm

AES is announced as a federal information Processing standard by NIST (National institutes of standards and technology) in 2001. AES is recurrently used encryption technique due to its high security, efficiency and simplicity. It uses the same key for both encryption and decryption process and known as symmetric block cipher. It uses three block ciphers AES-192, AES-128, AES-256. There are different rounds of processing according to the block size such as 10 rounds for 128- bit key, 12 rounds for 192-bit key and 14 rounds for 256-bit key [1].

A. Sub bytes transformation

This transformation involves a byte to byte nonlinear substitution where the substitute byte is obtained from a 16 x 16 look up table known as Substitution Box (S Box). To find the substitute byte for a given input byte:

- Input byte needs to be divided into two 4-bit patterns and find the corresponding integer value between 0 and 15.
- Represent these by their hex values 0 through F.
- One of the hex values is used as a row index and the other as a column index.
- From the S box lookup table, find the substitute byte after locating the corresponding row index and column index.
- Replace the corresponding data byte with the substitute byte.

B. Shift row transformation

The transformation is made to the incoming state array in such a way that the first row of the state array is not shifted, second row is circularly shifted to the left by one byte, third row is circularly shifted to the left by two bytes and the last row is circularly shifted by three bytes to the left.

C. Mix column transformation

The transformation operates on the State matrix column-by-column individually. Here each byte of a column is replaced by

a function of all the bytes in the same column as two times that byte, plus three times the next byte, plus the byte that comes next, plus the byte that follows. In simple matrix form this operation can be represented as:

$$\begin{bmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{bmatrix} \begin{bmatrix} S_{0,0} & S_{0,1} & S_{0,2} & S_{0,3} \\ S_{1,0} & S_{1,1} & S_{1,2} & S_{1,3} \\ S_{2,0} & S_{2,1} & S_{2,2} & S_{2,3} \\ S_{3,0} & S_{3,1} & S_{3,2} & S_{3,3} \end{bmatrix} = \begin{bmatrix} S'_{0,0} & S'_{0,1} & S'_{0,2} & S'_{0,3} \\ S'_{1,0} & S'_{1,1} & S'_{1,2} & S'_{1,3} \\ S'_{2,0} & S'_{2,1} & S'_{2,2} & S'_{2,3} \\ S'_{3,0} & S'_{3,1} & S'_{3,2} & S'_{3,3} \end{bmatrix}$$

Here the additions and multiplications are performed in GF (2^8).

D. Add Round Key transformation

This transformation is a simple bit wise XOR operation between the state matrix and the corresponding round key generated from the key scheduling module. This operation is also performed in decryption stage where each round performs four transformations: Inverse Shift Rows, Inverse Sub Bytes, Inverse Mix Columns and Add Round Key, and the last round do not have the Inverse Mix Columns transformation.

3. Implementation

With the ever increasing demand of smarter devices and rapid advancements in the field of IoT the only major hurdle that lies in its development is security and privacy issues. Security of the user's sensitive data that is sent over the internet is of utmost importance. This data is vulnerable to hackers and other unauthorized users. Our goal is to make this data as much secure as possible before sending it. Need has arisen for the devices to become smarter before transmitting the data.

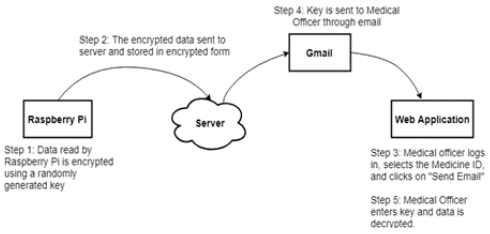

Fig. 1. System block diagram

As we know that the medicine inventories in government hospitals are not automated, there is a high risk of these medicines being misplaced and misused for monetary benefits by the local authorities. To overcome this problem, it is important that the system is fully automated without any human intervention and the concerned higher official gets regularly notified about anything happening inside the inventory. In order to keep a track on the entry and exit of medicines and to keep the higher authority informed we have made an attempt to design a system that is efficient enough to handle this situation.

The detailed steps involved in the system to store data at server side are as follows:

1. Run python script at the sender side and input medicine data.
2. Unique key is randomly generated and data is encrypted using this key.
3. The encrypted data is sent to the server and stored in encrypted form.

The steps involved in to retrieved data from server at Web application:

1. Enter credentials to login home page.
2. Enter medical id and database id and select "Send Email" button.
3. The key is sent to Medical Officer through Gmail.
4. Enter key and click on "Submit" button, the decrypted data is shown.

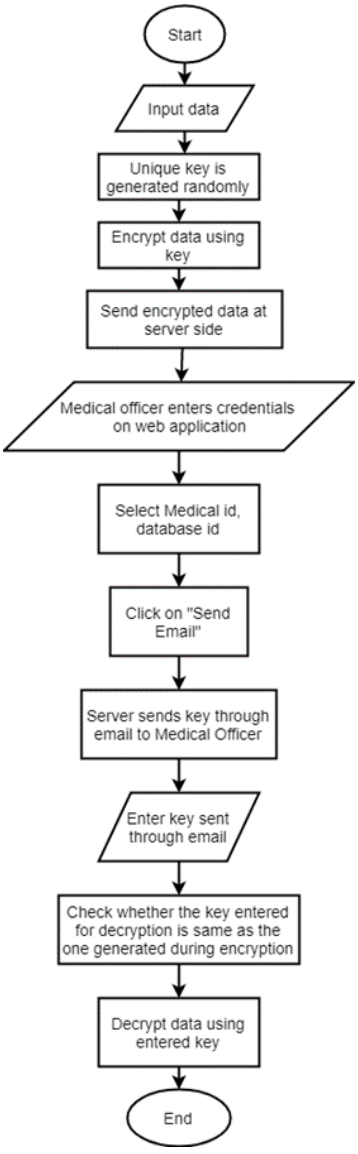

Fig. 2. Flow chart

4. Result and analysis

1. This project automates the traditional pen- paper based Medicine Inventory Management System.
2. We have used the Advanced Encryption Standard (AES) algorithm along with Gmail Security for Secure

Key transmission.

3. It is very essential to secure the data being used especially when it is confidential in nature. AES has been adopted by many Government and business firms nowadays to secure their data.
4. AES Encryption will make the system more secure and the time needed to crack the system increases further as the key to it cannot be easily obtained. Breaking a symmetric 256-bit key by brute force requires 2128 times more computational power than a 128-bit key. Fifty supercomputers that could check a billion billion (1018) AES keys per second (if such a device could ever be made) would, in theory, require about 3×1051 years to exhaust the 256-bit key space. Then the earth's population can crack one encryption key in 77,000,000,000,000,000,000,000,000 years! The bottom line is that if AES could be compromised, the world would come to a standstill.
5. We have also used Gmail Security for key transmission which is one of the largest and most popular email service providers. With all types of information traveling between its users, security and privacy are of utmost importance. The company has set numerous security measures into place to ensure information is secure and your personal data is not accessed illegally.
6. The Medical Officer can also follow some more security checks like:
   a) Using Strong Passwords
   b) Applying 2-Step Verification
   c) Complete the Security Checklist
   d) Enabling HTTPS Security
   e) Regularly Updating the Browser
   f) Utilizing the Safe Browsing Feature
   g) Regularly Checking Device Activity

This will ensure that the key sent is not illegally accessed at any point during its transfer from the server to the client.

1. Last but not the least we have provided the username and password authentication mechanism for anyone and everyone who tries to access the web application. The user name and password authentication mechanism (HTTP Basic Authentication) authenticates users with their user name and password credentials that are stored in the Access Manager user repository.
2. HTTP Basic Authentication - If a browser or program sends a request for a web page that requires Basic authentication, the server responds with an error that contains a 'WWW-authenticate' attribute in the header. The user then enters a username and password, which is sent to the server in a Base64-encoded form.
   - It is very lightweight authentication mechanism.
   - Most web servers and platforms provide built-in support; thus making the implementation very simple.
3. As we can analyze that the information security of the data being sent over the internet is achieved in our application by applying different levels of security

5. Future

- *Profile matching:* It ensures that the data sent to the server is through an entitled reliable source. This can be done by matching the IP address of the sender to the one stored in the database. As the IP address is dynamic over the internet one needs to purchase an independent IP address in order to implement profile matching. But this is practically and economically not feasible to demonstrate.
- *Rogue node detection:* To improve the security even better and to take it to a next level altogether the concept of rogue node detection can be implemented. This method detects and ensures whether the source node that is reading the data and sending it to the server is not being hampered or operated by any other unauthorized entity.

6. Conclusion

The paper presented an overview on Security in IoT using AES algorithm and secure key transmission using Gmail

References

[1] Samiksha Sharma, Vinay Chopra, “Analysis of AES Encryption with ECC,” Proceedings of International Interdisciplinary Conference on Engineering Science & Management, 17th - 18th December 2016, pp. 195-201.
[2] R. K. Ibraheem, R. A. J. Kadhim and A. S. H. Alkhalid, "Anti-collision enhancement of a SHA-1 digest using AES encryption by LABVIEW," *2015 World Congress on Information Technology and Computer Applications (WCITCA)*, Hammamet, 2015, pp. 1-6.
[3] https://www.defit.org/decryption/
[4] https://www.techopedia.com/definition/5507/encryption
[5] S. Vashi, J. Ram, J. Modi, S. Verma and C. Prakash, "Internet of Things (IoT): A vision, architectural elements, and security issues," *2017 International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC)*, Palladam, 2017, pp. 492-496.