Implementation Of Diverse Encryption Algorithms In Iot Devices For Securing User Data

[1]Mrs. M.Sowmiya, [2] Mrs.C.Thilagavathi
Assistant Professor, Department of IT
M.KUMARASAMY COLLEGE OF ENGINEERING, KARUR

Abstract

Our world has adapted to the many new technologies that make the human reduce his/her effort of working. The environment around us has become more e-friendly so as to make our world to be digitalized. Internet of things is one such important emerging technology that has supported humans through various applications such as washing machines, fitness tracker, smart gardening, health monitoring system and so on. Since the IoT has proven to be more essential for today's development, it becomes essential to give protection to the different IoT devices that are carrying sensitive data. The data that the devices use must not be leaked at any cost, because they may threaten the lives of human sometimes. Hence this paper allows us to understand various security algorithms that can be implemented in IoT devices for ensuring the data privacy. The different algorithms are studied and they are compared among each other to identify the best suitable one.

1. Introduction:

Internet of things emerging Technology that has been developed for the purpose of connecting the world it created the possibility to connect various physical devices on the internet using sensors and various devices. Advent of various Technologies such as big data cloud computing it networks made important for the era. IOT makes sensors to communicate and interact to the real world.

It possesses various characteristics such as scalability, efficiency, adaptability and several others. This emerging trend has a high market share in business (40.2%), healthcare (30.3%), retail(8.3%) and security (7.7%). However, the security in IoT is still a complicated thing to be considered. The other fields represented above has shown a good advancement in the corresponding fields. Fortunately, all the fields are in need of inventing several connected devices for serving the human in their day to day activities through internet. Few such devices include ATM, Smart meters, digital locks, smart cities, smart vehicles and smart health care.

In modern days, IoT has many applications that help to reduce the burden of human such as tank level monitoring and intimation to smart parking of vehicles. Each of these applications are done with the help of sensors and actuators. Hence it is necessary for the IoT to support the systems with a high level of security to ensure data loss. Hence this paper discusses about the various security threats occurring in IoT and few of the cryptographic algorithms that are useful in getting rid of the security issues.

2. Security in IoT:

IoT makes use of the real world data that interpret the solutions. There exists a thirst of safeguarding the data that are vulnerable to internet attacks, thus threatening the lives of human. Since the data can be more sensible to the environment it is necessary to give high security to the data that are being processed in ant of the security device. To overcome such security issues we extend the help of cryptographic algorithms. There is also an expectation that the added encryption algorithm should not give additional burden to the IoT device that may degrade the performance. The important security requirements such as authentication, integrity, confidentiality has to be ensured throughout the connection. Such a security algorithm must possess the following requirement:

- Size of RAM
- Power consumption
- Processing speed (Throughput & Delay)

Each of the above mentioned requirements have to be considered while choosing the encryption algorithm. The encryption algorithm will have to give fullest security to the data as well as does not give extra burden to the device. Based on the above mentioned requirements our cryptographic algorithms are classified into symmetric and asymmetric algorithms. This paper shows the study on various cryptographic algorithms that has to safeguard the data.

Symmetric algorithms are ones that make use of the same key for encryption and decryption. Asymmetric algorithms will use two different keys to encrypt and decrypt the data. Some of the wellknown algorithms are Advanced Encryption Standards (AES),Data Encryption Standards (DES),Rivest Shamir Algorithm (RSA), Diffie Hellman Algorithm, Present, Curupria.Although these algorithms show good results individually, they differ in terms of computing power, battery utilization, power consumption, response time, key length / size. This paper analyses the performance of all above mentioned algorithms and gives a suggestion for the best algorithm to be chosen in any of the IoT application.

2.1 Advanced Encryption Standards (AES):

AES was established in the year 2001 by National Institute of Standard and Technology (NIST). AES is a symmetric encryption technique that follows a block cipher approach. The block size of the plain text can be 128 bits. The key length is often 16 or 24 or 32 bytes. Supported, the key length the AES can be termed as AES-128, AES-192, or AES-256. This algorithm performs operations on 8 – bit bytes.

It takes each block as a single 128bit block into 4 × 4 square matrix and stores it into an array called state array. State array is changed at every stage of encryption and decryption. Based on the size of key the algorithm follows different rounds such as:

a. 128 bits – 10 rounds
b. 192 bits – 12 rounds
c. 256 bit – 14 rounds.

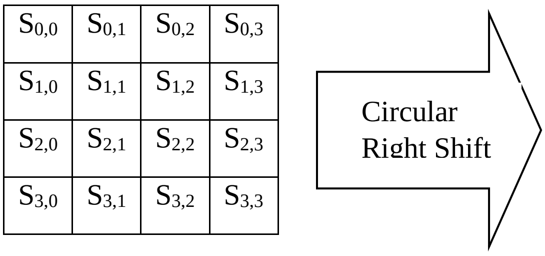
The key is expanded and used in AES. The AES algorithm performs its task by using four transformation functions. They are explained as follows;

i. Substitute Bytes:

S-Box implementation is used in substitute bytes transformation. The size of the S-Box is 16 × 16 matrix. The size of input and output is 8 bits. The first four bits are assumed to be the row number and the next four bits are assumed to be the column number.

ii. Shift Rows:

This transformation function performs circular right shift operation on the state array input. The shifting is done is row wise.



$S_{0,0}$	$S_{0,1}$	$S_{0,2}$	$S_{0,3}$
$S_{1,0}$	$S_{1,1}$	$S_{1,2}$	$S_{1,3}$
$S_{2,0}$	$S_{2,1}$	$S_{2,2}$	$S_{2,3}$
$S_{3,0}$	$S_{3,1}$	$S_{3,2}$	$S_{3,3}$

ISSN: 2233-7857 IJFGCN

$S_{0,0}$	$S_{0,1}$	$S_{0,2}$	$S_{0,3}$
$S_{1,1}$	$S_{1,2}$	$S_{1,3}$	$S_{1,0}$
$S_{2,2}$	$S_{2,3}$	$S_{2,0}$	$S_{2,1}$
$S_{3,3}$	$S_{3,0}$	$S_{3,1}$	$S_{3,2}$

iii. Mix Columns:

The output of the previous transformation function is stored in the sate array and processed in this step as follows. The function considers each column input as one word. It multiplies it with a predefined matrix and again stores it into state array.

iv. Add Round key:

This is the stage where the key is actually used. It performs XOR operation with the state array and the key. Thus, this stage gives the final output of the algorithm.

The decryption of AES algorithm is made reversing each stages of the transformation function. The Substitute byte, Shift Rows and Mix Column functions makes use of the inverse functions whereas the XOR operation done at the Add Round Key stage is done as such, without any inverse function.

2.2 Data Encryption Standards (DES):

DES was one of the commonly used encryption schemes in the earlier days, before the establishment of AES algorithms. DES was developed in the year 1977 by National Institute of Standard and Technology. It is form of block cipher approach, in which the input is taken as 64bit block and the size of the key 56bits. This technique uses the equivalent key for both encryption and decryption. The algorithm begins by passing the plain text into initial permutation (IP), which rearranges the bits of plaintext. This algorithm performs permutation and substitution operation in sixteen rounds. For each round, a subkey is formed by performing permutation and left circular shift on the keys. The leftward and rightward halves of the output of the last round is swapped to form a preoutput. This is then passed to IP^{-1} to get the final output.

The decryption process of DES is the use of same algorithm excepting that the application of subkeys is reversed. Also, the primary and concluding permutations are inverted.

2.3 RSA Algorithm:

RSA algorithm is a new approach of public key cryptosystems. The algorithm was developed in the year 1977 and published in the year 1978 by Rivest, Adi Shamir and Len Adleman. This algorithm takes numbers as input of size 0 to n-1 for some n.

Typically, the size of n is 1024 bits. The encryption algorithm is as follows:

[1] Let the plain text be $M < n$

[2] Select two prime numbers p, q

[3] Compute $n = p \times q$

[4] Compute $\phi(n) = (p-1)(q-1)$

[5] Choose integer e

[6] Compute $d = e^{-1} (\mathrm{mod}\ \phi(n))$

[7] Public key – {e,n}

[8] Private key – {d,n}

Therefore, the ciphertext is $C = M^e \bmod n$

The decryption process computes the plain text from the received cipher text:

$M = C^d \bmod n$.

RSA algorithm shows satisfactory level of security if the size of the key is high.

2.4Diffie Hellman Key Exchange Algorithm:

Diffie Hellman Key Exchange algorithm is used to enable two users to exchange the secret key in a secure manner. The procedure itself is limited to exchange of secret values.

The algorithm works by choosing a prime number q and an integer α which is the primitive root of q. These two values are publicly known numbers to the user A and user B. Then both the users choose random integers as their respective private keys such that $X_A<q$ and $X_B<q$.

Then the users compute public key using the formula, $Y_A = \alpha^{X_A} \bmod q$ & $Y_B = \alpha^{X_B} \bmod q$.

The key to be exchanged is then computed using the following formula:

User A: $K = (Y_B)^{X_A} \bmod q$

User B: $K = (Y_A)^{X_B} \bmod q$

The results of both the users are exchanged as a secret value.

2.5Present:

Present is a light weight cryptographic algorithm, which was introduced in the year 2007. It is called as light weight cryptography algorithm since it is 2.5 times smaller than AES algorithm. The compact size of the algorithm becomes a great advantage of this scheme when compared to all other schemes.

The input block size is 64 bits. The key length varies between 80 bit or 128 bits. It performs the encryption using a single S-box with hardware optimization. This type of algorithm is used in situations where low power consumption and high chip efficiency is expected.

2.6Curupira:

Curupira algorithm follows Wide Trail strategy and it was established in the year 1995. The block size of the input data is given as a set of byte arrays. The key length varies between 96 bits, 144 bits or 192 bits. The implemented S box of this algorithm is 8*8 bit S-box, and it is divided into two 4*4 bit S-box. This in turn makes us to reduce the space required to store the S-boxes.

3. Results &Discussions:

As IoT is one of the emerging technologies and it is also the new research fields, many questions are arsing in terms of providing security to the devices. The IoT devices may handle more sensitive data at some applications that it may even threaten the lives of human when it is leaked to others. Hence it is the need of the day to give maximum protection for the data that is being transferred among IoT devices. Thus, this paper introduces few of the IoT Security algorithms that can be applied to the IoT sensor devices. The comparison of the above given algorithms is given below.

Name of the Algorithm	Throughput (%)
AES	75
DES	58
RSA	75
DIFFIE HELLMAN	85

PRESENT	90
CURUPRIA	94

Table 1: Throughput representation of different algorithms

From the above table, it is evident that the light weight algorithms show a good throughput percentage as well as it utilizes the battery more efficiently. Let us view its graphical illustration from the below graph.

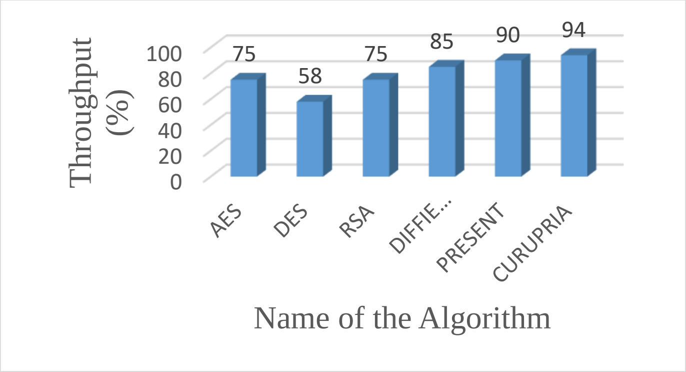


Figure 1: Comparison graph of diverse encryption algorithms over throughput

4. Conclusion:

IoT has become the most advanced and emerging one among several technologies in the world. Also, it can add up with many budding technologies to show high efficiency in real world applications, thus making the world to be digitalized. As it grows rapidly, it becomes necessary to provide protection. In this paper we saw the evolution of IoT, its advancements and the various security issues related to it. We also studied the different cryptographic algorithms that can be applied onto the IoT devices. As the technology grows, more security issues also arise. Hence it is necessary to focus on the security issues keenly and researches must be done efficiently by considering it to be the primary task.

5. References:

[1] Deena Nath Gupta, Rajendra Kumar, “Lightweight Cryptography: an IoT Perspective”, ISSN: 2278-3075, Volume-8 Issue-8 June, 2019.
[2] Indira Kalyan Dutta, Bhaskar Ghosh, Dr. Magdy Bayoumi, “Lightweight Cryptography for Internet of Insecure Things: A Survey”,2019.
[3] Susha Surendran (NYIT, Abu Dhabi, UAE), Amira Nassef (NYIT, Abu Dhabi, UAE), Babak D. Beheshti (NYIT, Old Westbury, New York), “A Survey of Cryptographic Algorithms for IoT Devices”, 2016.
[4] Rwan Mahmoud, Tasneem Yousuf, Fadi Aloul, Imran Zualkernan, “Internet of Things (IoT) Security: Current Status, Challenges and Prospective Measures”, 2015.
[5] Hui Suoa, Jiafu Wan, Caifeng Zoua, Jianqi Liu, " Security in the Internet of Things: A Review", DOI 10.1109/ICCSEE.2012.373, 2012.
[6] Rodrigo Roman, Pablo Najera, and Javier Lopez, University of Malaga, Spain, “Securing the Internet of Things”, 0018-9162/11/$26.00, 2011.