Kirkuk University Journal /Scientific Studies (KUJSS)
Volume 13, Issue 3, September 2018, pp. (154-173)
ISSN: 1992-0849 (Print), 2616-6801 (Online)

A New Security Method for the Internet of Things Based on Ciphering and Deciphering Algorithms

Ali Ayid Ahmad
Electrical Engineering Department, College of Engineering, Kirkuk University, Kirkuk, Iraq.
aliayid2013@gmail.com

Abstract

Internet of Things (IoT) is an advanced application scenario of Internet and becomes buzzword in current world of distributed application platform. Security is one of the biggest challenges to the private and sensitive data processed through the IoT. In this research paper a cryptographic algorithm technique is being proposed to provide more security of information flowing through the distributed network under IoT system The proposed cryptographic algorithm is capable of providing the enhanced security of data and information from the sensor or Radio Frequency Identification (RFID) sub systems of IoT to communication network such as Internet to cloud storage. The main features of this proposed algorithm is the simplicity, adaptability under distributed applications and its ability to encrypt and decrypt all types of data.

Keywords: (IoT) – Internet of Things;(IETF) – Internet Engineering Task Force; (RFID Radio Frequency Identification); Advanced Encryption Standard (AES).

طريقة جديدة لأمن إنترنت الأشياء المستندة إلى خوارزميات التشفير وفك التشفير

علي عايد احمد

قسم الهندسة الكهربائية، كلية الهندسة، جامعة كركوك، كركوك، العراق.

aliayid2013@gmail.com

الملخص

إنترنت الأشياء هو سيناريو تطبيق متقدم من الإنترنت، وسوف تصبح الكلمة الطنانة في العالم الحالي من منصة التطبيقات الموزعة. وان الأمن هو احد أكبر التحديات للبيانات الخاصة والحساسة والتي يتم معالجتها من خلال تقنيات إنترنت الاشياء. في هذه البحث تم اقتراح تقنية التشفير لتوفير المزيد من الأمن للمعلومات التي تتدفق من خلال الشبكة الموزعة لإنترنت الاشياء. وان خوارزمية التشفير المقترحة قادرة على توفير و تعزيز أمن البيانات والمعلومات من أجهزة الاستشعار أو أنظمة تحديد الترددات الراديوية الفرعية لإنترنت الاشياء الى شبكات الاتصالات مثل الإنترنت ومن ثم الى مناطق التخزين. والميزة الرئيسية لهذه الخوارزمية المقترحة هي بساطتها واعتمادها على التطبيق الموزع. وقابليتها على تشفير واستعادة جميع انواع البيانات.

الكلمات الدالة: إنترنت الاشياء، التطبيقات الموزعة، انظمة تحديد الترددات الراديوية، اجهزة الاستشعار.

1. Introduction

Internet of Things (IoT) is an integrated technology platform of Internet and sensor devices with communication medium. The communication of information is oriented with the sensor system. IoT is considered more advanced communication than the existing machine to machine communication. The dedicated physical objects such as different categories of devices have the embedded computing capabilities to sense and communicate the information for processing [1]. An ecosystem of things such as sensor devices, communication system, applications and data analytic capabilities are built together under the IoT system. Internet is the carrier communication of the whole information and data sensed from the external environment for which the sensor based objects are used. As, Internet is one of the major building block of connectivity of a vast range of devices like sensors, RFID tags for gathering the information as per the usability setup, a major concern goes onto the security of information [1, 2]. The open nature of the Internet communication, securing the information becomes a massive requirement for IoT. Some of the objects used with IoT such as self healing sensors are autonomic and self configurable that reduces the security issues little bit. Further, heterogeneity of objects those are connected with the Internet increases the challenges for securing the information in the network.

Massive application fields such as home automation, medicine, healthcare management, industrial quality control, smart cities and many more really creates a heterogeneous environment of technology that create the complexities to implement the secure digital communication system [3]. A wireless sensor network generate the information by sensing the objects surroundings through equipped sensors and sends the sensed information to central processing controller. The security issue arises from sensor to the network that carries the information.

Cryptographic Security technologies are considered as the robust solution to provide confidentiality, privacy and authenticity to the information of IoT [4]. There are so many cryptographic algorithms are used to secure the communication medium used by the IoT system. Some of the popular cryptographic algorithms commonly employed with IoT communication are secured socket layer cryptography, Authentication with Kerberos protocols, AES encryption algorithm to hide the meaning of information etc. These all cryptographic algorithms used to protect the sensitiveness of information also in the cyber space to secure the information and information system.

IoT is emerging technology having the huge potential of the use of Internet in the applications of the real world area. Security of data and information is primary requirements of the IoT communication network. A sensor based network is more prone to be attacked so that the enhance security framework of data security must be implemented to provide robust security to the IoT system.

2. Security Challenges to IoT and Network

There are massive ranges of the security challenges and issues related with the IoT systems and its communication network. These all security issues and challenges are faced while the designing the communication structure protocols and architectural framework for the communication of IoT through the Internet.

2.1 Huge Scaling Feature

The smart sensors, RFID tags are used in IoT are huge in numbers and also its future deployment is also unpredictable. Due to this feature of IoT the existing protocols and architecture as functioning becomes non functional. The requirements of bandwidth, processing capability, routing of the packets etc. becomes very complex and exiting communication network used with Internet becomes unable to handle the communication. This further imposes the security issues as a large amount of data. The huge scaling also leads the requirements of own energy source which depletes the charged power source.

2.2 Dependencies on the Architecture

Diverse nature and versatility of functioning of different categories of IoT devices require the different architectural framework to be combined into a central architectural framework. This is too complex and building a centralized architectural framework to provide ease of connection, controlling features, communication bandwidth, and relevant useful application support. The dependencies on the applications and architecture further leads the heterogeneous environment that reduces the security scope by providing the loop hole to the cyber criminals to hack the information. As many things such as devices and protocols are functional with their own application capabilities so that it creates the limitation to the secured protocol to handle the security implementation to the whole of the system. Shared environment and its implementation again provides opportunity to the intruder and hackers to steal the information as Internet is primary dependency to the user of the IoT system.

2.3 IoT and Big Data Application

IoT system generates enormous amount of data in fraction of second. Thus, a vast amount of raw data is required to be processed in same fraction of time. The network such as Internet connection also be capable to handle the communication of such amount of vast data. To get knowledge from the raw data needs the processing in time. Some of the critical system and application area of IoT such as healthcare monitoring system to monitor the health factor of patient, always requires efficient and secured communication environment with Internet. For, example, the captured raw data of heart rate, blood pressure, pulse rate etc must be processed timely to provide the knowledgeable information. Security issues in this scenario primarily concerned with the different categories of attacks such as denial of service attacks and unavailability of the Internet traffic.

2.4 Cloud Storage and its Severity

The cloud system is third party based architecture that is main system under IoT to store the huge amount of raw data for processing. Network is also cloud based so that security issues are more prevalent and there are so many security challenges come into existence [5]. The inherent security framework implemented with the cloud storage system is controlled by the involvement of third party. This is again a security risk of confidentiality of the data captured through the IoT devices and stored into the third party cloud storage. Monitoring of the access of the cloud storage is not possible by the real user of the IoT, thus confidentiality of stored information always poses the enhanced challenges to the stored data.

3. Security Needs of IoT and Network

Gartner proposed that twenty five billions IoT Application will be under use by 2020. According to him, the different industries who use the IoT till 2020 are summarized in Fig. 1.

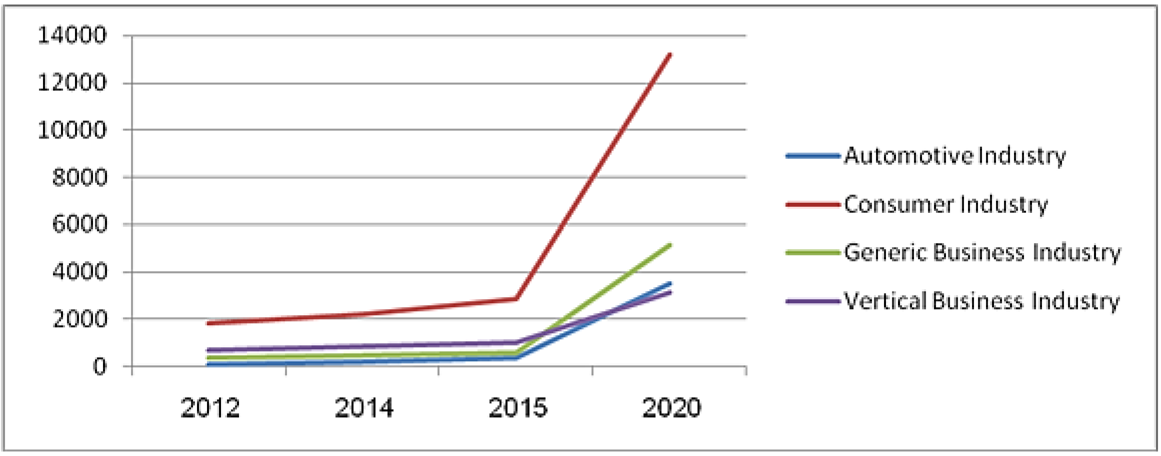


Fig. 1: IoT Installed till 2020 by Gartner

Fig. 1 Information shows that vast scaling of the IoT till 2020 needs the secured distributed applications to be developed to handle the secured communication over the Internet. The international institutions such as IEEE (Institute of Electrical and Electronics Engineering) and IETF (Internet Engineering Task Force) working together to develop such secured distributed applications to provide the robust secured environment of communication system for IoT systems to make it more reliable in the field of industrial applications [6].

4. IoT Security with Cryptographic Solution

Cryptography is considered as one of the robust security frameworks of the Internet communication. The different categories of existing cryptographic algorithms will not be capable of handling the requirement of IoT security in the coming future with rapid growth of application scenarios of IoT. The IoT networks such as sensor based network connecting all the sensing devices to capture the surrounding data, the common Internet to communicate that captured information to process elements and finally to store into the multi tenant based cloud storage system require, the fast and efficient routing and congestion control mechanisms having the secured framework.

It is also true that implementing the cryptographic system with each communication gateway point delays the communication due to additional processing involved with the encryption and decryption of data [7]. Therefore, simplified distributed application having the inherent cryptographic capabilities is more suitable for the IoT based network communication.

In this paper a cryptographic algorithm is proposed to process all types of data to provide a robust security with the distributed application used with IoT. The proposed Encryption and decryption of the sensor captured information are presented in block 4 and block 5.

5. Analysis of Cryptography Algorithm of Text Data

In this part of the research, cryptographic algorithm for processing text data is analyzed to provide security with the user's distributed application with IoT. Encryption and decryption of sensor text information captured are displayed in Block 1, Block 2 and Block 3 [8].

The cryptography algorithm works in the following way: -

Three Level Encryption

Sensor Captured data→

Convert all letters of Text Data in Upper Case

Level 1

Convert blank spaces to $ and # (Even Space to $ and odd space to #)

Level 2

Switch and replace all alphabets with complementary alphabets (A with Z and B with Y)

Level 3

Switch each second alphabet that is not blank with the following calculated code

$n = \text{ASCII}(\text{Char})$

$n = n + (\text{key})^2$

Append char to encoded series

These three Levels can process text-only data.

```
// ---------------- Pseudo code Three Level Encryption Text Data
Input : x  is an array of characters which represents plain text
```

```
Input: n is number of characters in array x
Output : y is an array of characters which represents cypher  text
// level 1
Set e to 0
For i=0 to n do
        Y[i]= upper case ( x[i])
        if( Y[i] equal to space) then
                if e is even number ) then
                set y[i]='$'
                else
                set y[i]='#'
                end if
          Add 1 to e
        end if
End for
// level 2
For i=0 to n do
    If Y[i] is alphabet  then
        Y[i] = complementary alphabets (A with Z and B with Y) of Y[i]
end if
End for
// level 3
Set t to 1
For i=0 to n do
If Y[i] is alphabet  then
        If t mod 2 equal to 0 then
        ch = ASCII(Y[i])
ch = ch + $(key)^2$
           Y[i]= char(ch)
        End if
          Add 1 to t
    End if
End for

   ➔ Encrypted Text Data



// ---------------- Pseudo code for  Encrypted Algorithm Text Data

Input : x  is an array of characters which represents plain text
Input: n is number of characters in array x
Output : y is an array of characters which represents cypher  text

For i=0 to n do
      Y[i]= upper case (  x[i] )
      if( Y[i] equal to space) then
            if( e is even number ) then
            set y[i]='$'
            else
            set y[i]='#'
            end if
       Add 1 to e
      end if
End for

For i=0 to n do
   If Y[i] is alphabet  then
      Y[i] = complementary alphabets (A with Z and B with Y) of Y[i]
end if
End for

Set t to 1
For i=0 to n do
```

```
        If Y[i] is alphabet  then
           if t mod 2 equal to 0 then
           ch = ASCII(Y[i])
ch = ch + $(key)^2$
              Y[i]= char(ch)
           End if
             Add 1 to t
        End if
End for
```

Block 1: Encryption Algorithm and Pseudo code of text data for IoT

Decryption Single Stage

Encrypted text data →

Convert encrypted text to upper case

Process each character

If odd character replace with complementary alphabet

If even character (Not $ and #) then subtract $(key)^2$ from member sequence

Cast the character and append to decoded data sequence

If ($ or #) then append a blank space decoded data sequence

➔ Decrypted text data sequence

```
// ----------------Pseudo code for Decrypted Algorithm text data

Input : x  is an array of characters which represents encrypted text
Input: n is number of characters in array x
Output : y is an array of characters which represents plain  text
```

```
For i=0 to n do
    Y[i]= upper case (  x[i] )
    If ( i is odd number) then
        Y[i] = complementary alphabets (A with Z and B with Y) of Y[i]
    Else
        If (Y[i] not equal to $ and #) then
        ch = ASCII(Y[i])
    ch = ch- (key)^2
    Y[i]= char(ch)
        End if
 If (Y[i]  i equal to $ or #) then
         Y[i]=' '
      End if
    End if
End for
```

Black 2: Decryption algorithm and Pseudo code of text data for IoT

6. Cryptographic Algorithm Text Data System for IoT

With reference to block 1 and block 2 of encryption and decryption algorithms the cryptographic system for the IoT system is designed. The designed cryptographic algorithm is used to be integrated with the distributed applications of IoT. The steps for encryption and decryption are presented in block 3.

Data Encryption:

Step 1. Enter sensor data

Step 2. Change all char to upper case

Step 3. Extract each of the character and perform following three level encryption on each character

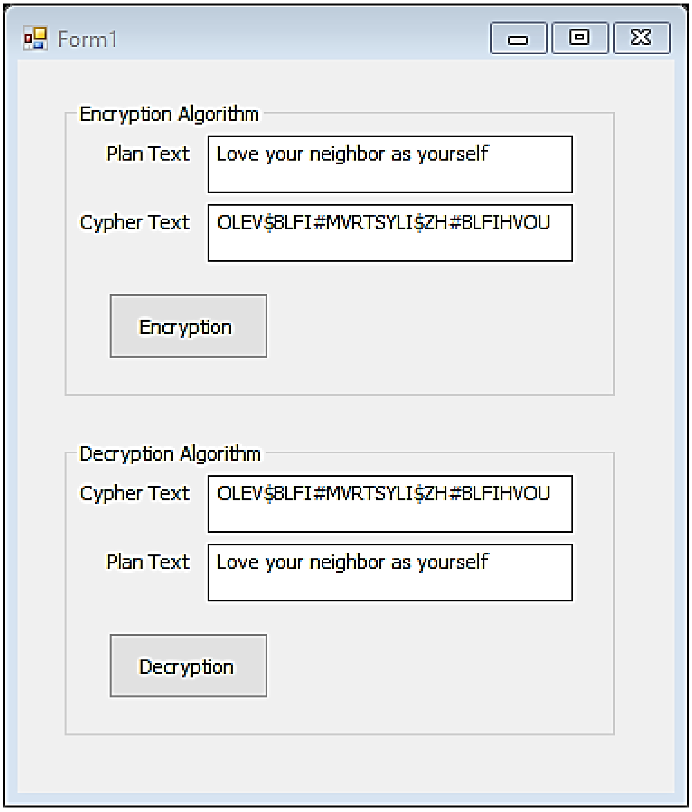
- Change even black space with $ and odd blank space with #

```
    - Switch the A with Z and B with Y and so on this switching
    - Change each of the 2nd alphabet which is not black space with following code
        o n = ASCII (char)
        o $n = n + (key)^2$
Step 4. Add the encoded char to encrypted char sequence

Data Decryption Text Data:

Step 1. Change cipher char to upper case
Step 2. If char is odd char then replace this with complementary char
Step 3. If char is even but not $ and # then subtract (key)2 from the number sequence
Step 4. Add the decoded char into decrypted sequence
Step 5. If char is $ and # then append a black space to the decrypted sequence
```

Block 3: Cryptography Algorithm of Text Data for IoT.

7. Implementation of Encryption and Decryption Algorithm Text Data and Results

A new algorithmic program (C# language) is created to check the encryption Block 1 and decryption Block 2 algorithm text-only statement: I love your relative as yourself. The created program expressed the encryption algorithm. The following result (OLEV$BLFI#MVRTSYLI$ZH#BLFIHVOU) was accordingly obtained as in Fig. 2. When this encrypted data was entered into the decryption section of the same program for processing, the same original statement (I love your neighbor as yourself) was accordingly retrieved as in Fig. 2.



Encryption Algorithm

Plan Text: Love your neighbor as yourself

Cypher Text: OLEV$BLFI#MVRTSYLI$ZH#BLFIHVOU

Decryption Algorithm

Cypher Text: OLEV$BLFI#MVRTSYLI$ZH#BLFIHVOU

Plan Text: Love your neighbor as yourself

Fig. 2: Implementation of Encryption and Decryption algorithm Text Data Program

8. IoT Security with Cryptographic Solution for all Types of Data

Using the algorithm encryption and decryption Blocks 1 and 2 and its implementation in the above program Fig. 2, analysis shows that such algorithm can only process text data algorithms. Where text data was entered into the encryption algorithm and the same text was retrieved in the decryption algorithm. Therefore, the algorithms do not meet the purpose and stand unable to process all kinds of data from the encryption and decryption process. Therefore, a new algorithm is proposed to solve the encryption/decryption of all types of data (texts, digits, symbols, video and sensor signals, etc.). Since all types of data are represented as successive bites, the newly proposed algorithm is designed to process the encryption/decryption of all sets of successive bytes. This cryptographic algorithm creates codes for the entered data in a precise, easy, uncomplicated and fast manner. In the decryption algorithm, the data is retrieved with 100% correct results. This is the reason for constructing this new algorithm with the details shown in Blocks 4 and 5:-

Two Level Encryption

Sensor Captured data→

Level 1

A) Data were configured according to ASCII Code, as developed in Figure 2.

All the above bytes of ASCII code data were switched to different bytes using the (XOR) gate:

Y[0] = X[0]; for i = 1 to n-1 For i=1 to n-1 do Y[i] = X[i] XOR Y[i-1]

B) The first byte of the ASCII data is retained unchanged. The XOR gate was applied to the first and the second bytes, and the result gets automatically switched into the position of the second byte. The XOR gate was applied to the second and third bytes, and the result gets automatically switched into the position of the third byte. And so the process continues for the rest of the bytes.

Switch and replace the second byte is replaced by the XOR gate between the first and the second bytes. The third byte is replaced by the XOR gate between the second and the third bytes, and so on to the end of the string of bytes that make up the data.

Level 2

The following formula is applied with each byte:

Y[i]= (y[i] + $(key)^2$) mod 255

Append byte to encoded series

➔ Encrypted Data

```
// ---------------- Pseudo code for  Encrypted Algorithm
Input:  x an array of data
Input: n  is the number  of bytes in array x
Input: i  is the index  of bytes in array x
Input : Key is a factor used for maximizing the security of data, making the description almost impossible
Output: y an array of encrypted data of array x
y[0]=x[0];
For i=1 to n-1 do
y[i] = x[i]  XOR  y[i-1]
end for
```

```
For i=0 to n-1 do
 $Y[i]= ( y[i] + (key)^2) \bmod 255$
end for
end algorithm
   ➔ Encrypted Data
```

Block 4: Cryptographic and Encryption algorithm and Pseudo code of Sensor all data types for IoT

Decryption Single Stage

Level 1

The formula $Y[i] = Y[i] - (key)^2$ was applied to all of the encrypted (pseudo) bytes.

If $Y[i]< 0$ then

$Y[i] =Y[i] + 255$

Level 2

A)

All the above bytes of ASCII code data were switched to different bytes using the (XOR) gate: $X[0] = Y[0]$; for i= 1 to n-1 do $X[i] =Y[i]$ XOR $Y[i-1]$

B) The first byte of the ASCII data is retained unchanged. The XOR gate was applied to the first and the second bytes, and the result gets automatically switched into the position of the second byte. The XOR gate was applied to the second and third bytes, and the result gets automatically switched into the position of the third byte. And so the process continues for the rest of the bytes. The algorithm redevelops of the original plain data according to ASCII code.

Decrypted data →

```
// ----------------Pseudo code for Decrypted Algorithm
Input: y an array of encrypted
Input: n  is the number  of bytes in array y
Input: i  is the index  of bytes in array x
Input : Key is a factor used for maximizing the security of data, making the description almost impossible.
```

```
Output: x an array of decrypted data
For i=0 to n-1 do
          y[i] =  y[i] -  $(ke y)^2$
  If y[i]< 0 then
          y[i] = y[i] + 255
 End if
end for
          x[0]=y[0];
For i=1 to n-1 do
         x[i] = y[i]  XOR  x[i-1]
end for
end algorithm
      ➔ Decrypted data sequence
```

Block 5: Encryption algorithm and Pseudo code   to Sensor all data types for IoT

9. Implementation of Encryption and Decryption Algorithm all Data Types Result

A program was built using the C# language, as a model to implement the proposed algorithm to encrypt all data types mentioned in Block 4, where the following data was used : (1 - Love your neighbor as yourself. @).These data contain text, special symbols and numbers. It is possible to use any data for the purpose of encryption and decrypting. However, the statement data is chosen for easy understanding of the reader. When these data (letters, symbols and numbers) are represented by the corresponding ASCII code, to obtain a set of bytes that correspond to their ASCII code:-

(49 32 45 32 76 111 118 101 32 121 111 117 114 32 110 101 105 103 104 98 111 114 32 97 115 32 121 111 117 114 115 101 108 102 46 32 64**).**

These data becomes ready to begin the encryption algorithm Level 1 – Block 4. When this data is entered to the first level of the proposed encryption algorithm mentioned in Block 4, we obtained the following results**:-**

(49 17 60 28 80 63 73 44 12 117 26 111 29 61 83 54 95 56 80 50 93 47 15 110 29 61 68 43 94 44 95 58 86 48 30 62 126**).**

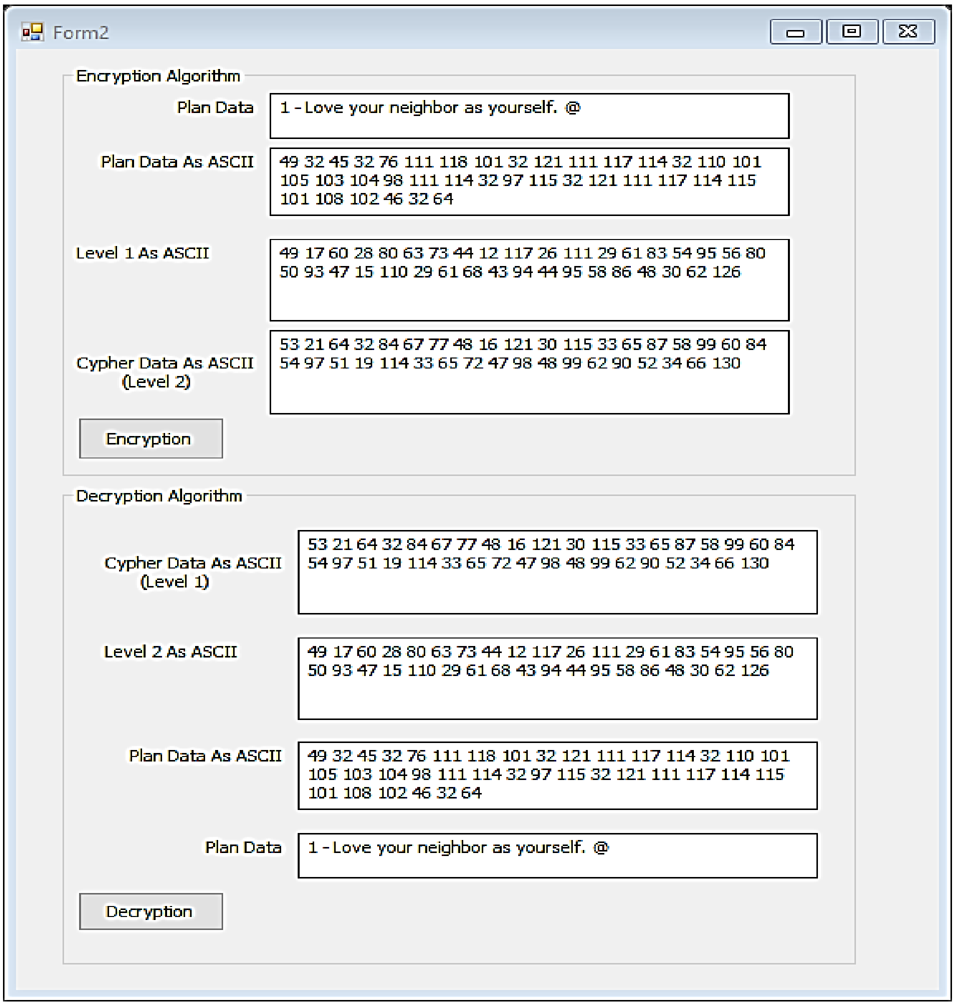
When the outputs of Level 1 were entered to the Level 2 of the proposed cryptographic algorithm, the following encoded data was obtained:
(53 21 64 32 84 67 77 48 16 121 30 115 33 65 87 58 99 60 84 54 97 51 19 114 33 65 72 47 98 48 99 62 90 52 34 66 130)

All the above implementations are shown in Fig. 3. When applying the proposed decryption algorithm, the following encrypted data were entered to Level 1, Block 5:
(53 21 64 32 84 67 77 48 16 121 30 115 33 65 87 58 99 60 84 54 97 51 19 114 33 65 72 47 98 48 99 62 90 52 34 66 130)

In the second level of the decryption algorithm, the equation is applied to the date in Level 1 to obtain the data shown in Level 2:
(49 17 60 28 80 63 73 44 12 117 26 111 29 61 83 54 95 56 80 50 93 47 15 110 29 61 68 43 94 44 95 58 86 48 30 62 126).

When applying the second equation to the outputs of Level 2, the following data are obtained as shown in the ASCII decoded section:
(49 32 45 32 76 111 118 101 32 121 111 117 114 32 110 101 105 103 104 98 111 114 32 97 115 32 121 111 117 114 115 101 108 102 46 32 64).

The program restores the original data (letters, symbols and numbers) from the corresponding ASCII codes as (1 - Love your neighbor as yourself. @). All the above implementation of decryption is shown in Fig. 3.



Encryption Algorithm

Plan Data: 1 - Love your neighbor as yourself. @

Plan Data As ASCII:
```
49 32 45 32 76 111 118 101 32 121 111 117 114 32 110 101
105 103 104 98 111 114 32 97 115 32 121 111 117 114 115
101 108 102 46 32 64
```

Level 1 As ASCII:
```
49 17 60 28 80 63 73 44 12 117 26 111 29 61 83 54 95 56 80
50 93 47 15 110 29 61 68 43 94 44 95 58 86 48 30 62 126
```

Cypher Data As ASCII (Level 2):
```
53 21 64 32 84 67 77 48 16 121 30 115 33 65 87 58 99 60 84
54 97 51 19 114 33 65 72 47 98 48 99 62 90 52 34 66 130
```

Decryption Algorithm

Cypher Data As ASCII (Level 1):
```
53 21 64 32 84 67 77 48 16 121 30 115 33 65 87 58 99 60 84
54 97 51 19 114 33 65 72 47 98 48 99 62 90 52 34 66 130
```

Level 2 As ASCII:
```
49 17 60 28 80 63 73 44 12 117 26 111 29 61 83 54 95 56 80
50 93 47 15 110 29 61 68 43 94 44 95 58 86 48 30 62 126
```

Plan Data As ASCII:
```
49 32 45 32 76 111 118 101 32 121 111 117 114 32 110 101
105 103 104 98 111 114 32 97 115 32 121 111 117 114 115
101 108 102 46 32 64
```

Plan Data: 1 - Love your neighbor as yourself. @

Fig. 3: Implement For Encryption And Decryption Algorithm

10.Advantages and Scope of the Proposed IoT Cryptographic System

The proposed cryptographic algorithm of IoT security with its distributed application provides both confidentiality and integrity of the data and information, which flow through the network from the sensors and RFID and also stored into the cloud storage system[8][9]. Further, this algorithm also caters the simplicity and reduces the complexities involved in the processing and computation. The algorithm has the following features

- Simplicity
- Flexibility
- Data confidentiality and integrity
- Robustness and scalability
- Efficiency in security services of IoT

11.Conclusion

In this paper a simplified cryptographic algorithm is proposed to provide an enhanced double security to the IoT, which is an emerging technological framework for the real world applications. The proposed algorithm has the capability of processing all types of data, including, texts, numbers, special symbols, signals, etc. Internet is primary communication framework of data and information captured by IoT implemented sensors and RFID tags. The basic communication framework of IoT system is IP based, rendering the IP security is of a primary concern. Further, enhanced security to the complex distributed application requires the simplified version of integrated cryptographic application to process the data to make it more secured.

References

[1] Tapalina Bhattasali, Chaki Rituparna, and Chaki Nabendu. "***Study of security issues in pervasive environment of next generation internet of things***." Computer Information Systems and Industrial Management, Lecture Notes in Computer Science, 8014, 206 (2013).

[2] Hui Suo, Wan Jiafu, Zou Caifeng, and Liu Jianqi. "***Security in the internet of things: a review***.", international conference on Computer Science and Electronics Engineering (ICCSEE), (2012).

[3] Amirhossein Farahzadi, Shams Pooyan, Rezazadeh Javad, and Farahbakhsh Reza. "***Middleware Technologies for Cloud of Things-a survey***." Digital Communications and Networks" Available Online 18 April (2017).

[4] Riahi Arbia Sfar, Natalizio Enrico, Challal Yacine, and Chtourou Zied, "***A roadmap for security challenges in the Internet of Things.***" Digital Communications and Networks, 4(2), 118 (2018).

[5] K. Taira, "***2-2 Research and Development of Security Architecture : Overview,*** "Journal of the National Institute of Information and Communications Technology, 63 (2), 11 (2016).

[6] P. P. Ray, "***A survey on Internet of Things architectures***.", Journal of King Saud University-Computer and Information Sciences, 30(3), 291 (2018).

[7] Ghofrane Fersi, "***A distributed and flexible architecture for Internet of Things.***" Procedia Computer Science, (73), 130 (2015).

[8] Abhishek Anand, Raj Abhishek, Kohli Rashi, and Bibhu Vimal. "***Proposed symmetric key cryptography algorithm for data security***." In International Conference on Innovation and Challenges in Cyber Security (ICICCS-INBUSH), India (2016).

[9] Tae Jung ,Kim. "***Requirement of Security for IoT Application based on Gateway System.***" International Journal of Security and Its Applications ,9(10),201(2015).