Journal of Information Systems Engineering and Management
2025, 10(23s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

Enhancing IoT Security with Multilayer Authentication and End-to-End Encryption

Ashish Dwivedi[1], Santosh Kumar Sharma[2]
Department of Computer Science & Engineering, United University, Prayagraj, India
Department of Computer Science & Engineering, United University, Prayagraj, India
Email: dwivedi_ashish01@yahoo.com, sharma.santosh83@gmail.com

ARTICLE INFO

Received: 18 Dec 2024

Revised: 08 Feb 2025

Accepted: 21 Feb 2025

ABSTRACT

The internet of things is a sophisticated network of all the items that are controllable online. Owing to its advanced accessibility and management qualities, it is seen as one of the most sought-after pieces of technology in the upcoming periods. Its ability to support a wide range of things in daily life has demonstrated its effectiveness in serving human society. It must contend with security issues including peer-to-peer authentication and data secrecy from prying eyes. Due to the multicast user approach of the Internet of Things, there is constant contact between the user and the many objects. To protect our resources from any kind of security breach or vulnerability, a robust security system is needed. Thus, focusing on this specific issue, we suggested a method that combines two security techniques. The first is the Elgamal end-to-end security mechanism, which is followed by a tiered authentication strategy. This technique operates in two stages: in the first, the sub-server and the authentication procedure work together to provide authentication for access to the main data server, and in the second. When a user gets close to the actual server, he must use the public cryptography technique, which encrypts data using Elgamal cryptography to offer end-to-end security. Every aspect of the task is thoroughly examined using the cryptography and authentication factors at hand. Ultimately, the suggested solution will offer robust security to better facilitate communication and preserve data.

Keywords: cryptography, end-to-end security, authentic server, Elgamal, Cramer-Shoup, Blum Gold Wasser, CEILIDH cryptosystem.

INTRODUCTION

The team behind Internet standards and researchers working tirelessly to develop Internet-based technologies are progressively affecting people's work environments and lifestyles on a daily basis**.** At when it comes to IOT, we face a hard task in ensuring that every user that requests access to the network has secure housing and resource allocation in an unpredictable environment where several users may join and exit at any time. Here, the system we suggest is in light of the current situation, everyone wants to use the tools society has at its disposal to improve every aspect of human existence, including smart homes, smart businesses, and smart medical systems.

The user's ability to access and monitor their data resources from any location at any time is enhanced by the internet of things. With their exceptional work and insistence on working efficiently from a distance rather than leaving the office to complete a particular assignment, intelligent monitoring and surveillance systems for elderly parents as a smart health care system that runs around the clock, the patient can be in constant communication with their carer. It is predicted that by 2030, the Internet of Things will support 40-50 billion devices. Regarding the Internet of Things, the topic of whose complicated structure has made it so difficult to handle numerous connections and how to give authentication across multiple locations is discussed. Internet technology has effectively delivered services right to our doorstep. Peer authentication, virtual assaults, eavesdropping, and other potential security risks have been brought to light by the multicast user approach, which is one of the more sophisticated uses of wireless communication technologies in the Internet of Things. The internet of things offers a number of capabilities for controlling various items, like smart refrigerators and air conditioners. IoT devices should have sufficient security features to protect stored data, as maintaining privacy and security has become a difficult task in this realm. Even so, a number of security measures, including the onion layer security system, were implemented to offer security ranging from the physical to the application layers.

Semantic security, which isn't offered by any other methods, is how Elgamal encryption works. Elgamal encryption's strength is its ability to process a single piece of plain text and transform it into several different kinds of cypher text.

Other Public Cryptosystem

The Cramer-Shoup cryptosystem is another sophisticated public key encryption technique that has the ability to defend against inventive attacks and adaptive selected cypher text attacks. The operating premise of the Blum-Goldwasser cryptosystem is public cryptography, which ensures probabilistic and semantic security with constant cypher text size and expansion due to the semantic security system, all without the need for extra assumptions. Based on an algebraic discrete logarithm problem, CEILIDH public cryptography is another sophisticated cryptosystem. This system's main benefit is that it uses smaller keys while maintaining the same level of security.

RELATED WORK

The two main issues with IOT are connectivity and security. However, we are solely paying attention to security. We shall talk more about the security challenges for the reasons listed below [1, 4]. In order to improve IOT security, Wang [5, 7] discusses BAN logic in conjunction with a heuristic technique, an improved end-to-end authentication procedure and user protocol. Contributions by Moon and Wan [7, 8] demonstrated the effectiveness of biometric authentication and block chain security. We need to take the heavy security mechanism into account when improving the internet of things' security system. It guards against several kinds of threats and attacks, like phishing attempts, eavesdropping, and on-off trust attacks. The technique of quantum cryptography has been presented by Qili & Wilson [11, 13, 14] as a solution to the problem of internet security.

The Internet of Things employs several cryptographic techniques to protect user data and other users from unskilled users. Batool & Christos [16, 17, 18] give the principle authentication activity management in place of cryptography technique. Researcher has a direct control over the activities of the malicious users with the help of authentication and activity management directly, with co-operative joining technique. IOT agents are also introduced in the context of network-based virtual clones (physical objects/devices). The agent's main responsibility is to operate as a standalone service and split from the physical layer. Additionally, any attempt to access from the publicly accessible open network must be limited in order to reduce the expense of security services. In addition to providing guidance on how to manage the sensor directly in certain situations, Gang & Liu [20, 23, 24] they explain how the cooperative jamming strategy functions.

Managing trust while allowing the user to address sensor situations Praveena & Sharma [19, 29] use a hybrid approach to cryptography to create a complex security layer structure that encrypts stored data using the MAC OMURA technique and ultra encryption technology. [28] To provide more sophisticated security for the Internet of Things, Sharma has focused more on the layered security strategy. Zhou & Wang [31, 32] they have talked about the various block chain security techniques for IOT, and they have also talked about trust management at every security tier. Zhou and Sheen [32] have put out the mythology surrounding quantum in the context of encryption and its power. In light of cutting-edge security, Wany & Ren [33] submitted their work to give an instantaneous encryption solution. Ran has talked about the use of attribute basis encryption in cryptography.

AUTHENTIC FRAME WORK

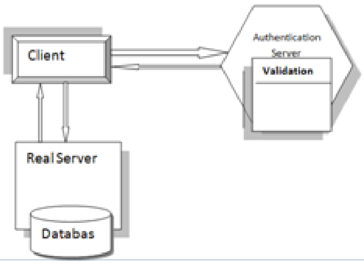

Fig.1. Security Architecture

DESCRIPTION

The client requests that the authentication server validate the details in the proposed system. Validation in the authentication server is carried out by employing certain cryptographic methods. We are encrypting and decrypting the data in our proposed work utilising Elgamal cryptography. In order to decrypt the data, the client puts the KPUB and KPRI here. Ultimately, the client receives the decrypted data at a scheduled session.

Table.1 Notations and Symbols used to Elaborate Algorithm & Description

Keywords	**DESCRIPTION**
$User, Uid, Pwd$	The ith user, identity, password
R_n	Random numbers
h (.)	hash function use to provide collision resistance for security consents
||	Addition of values
⊕	The bit-wise exclusive or operation
D_i	Details of the i^{th} user/client
A_S	Authentication Server
T_s	Token with time stamp
SK_i	Session key calculated and provided to user
V_d	Validation details
DB	Database
R_s	Real server which maintains DB
R_p	Repository
TDB	Transactional database
%	Mod
E_T	Encrypted text
D_T	Decrypted text
P_T	Plain text
P_N	Any large prime number
d_E	Member Element
Pr1, Pr2	Primitive roots
KPUB, KPRI	Public key, private key
CT1, CT2	Cipher text
a(x)	number generator

The ElGamal encryption protocol is a public-key cryptosystem named after its inventor, Taher ElGamal. It is used for secure communication and digital signatures. Here's a brief overview of how it works:

1. Key Generation:
   a. Each user generates a pair of keys: a public key and a private key.
   b. The public key consists of parameters shared publicly and a large prime number, usually denoted as p, and a generator g of the multiplicative group modulo p. The public key is typically denoted as (p, g, a), where a is the private key exponent.
   c. The private key is a random number, typically denoted as a, chosen by the user.
2. Encryption: To send a message M to a recipient, the sender picks a random number k and computes $(c1, c2)$, where:

   $c1=g^k \text{mod}\, p$

   $c2=M\cdot y^k \text{mod}\, p$, here $y = g^x \text{ mod p}$

3. Decryption: Upon receiving ($c1$, $c2$), the recipient computes:

    i. M=c2$\cdot$S^{-1} mod p

    ii. S =C1^{x} mod p , such that S.S^{-1}=1 mod p

The security of ElGamal relies on the difficulty of the discrete logarithm problem, which is believed to be hard to solve efficiently. ElGamal can also be used for digital signatures, where the roles of public and private keys are reversed. The private key is used for signing and the public key is used for verification. Overall, ElGamal is a widely used cryptographic algorithm, especially in applications that require secure communication and digital signatures. The server assigns a token with a time stamp and key if the details are determined to be legitimate. Next, the client logs into the actual server to retrieve the database. The client can access the database they have requested with the aid of the access server. The database server houses the transactional database, which is accessible there. In transactional database the key distribution centre (KDC) reside in it, where a huge number of keys are present in it. In this transactional database the data is decrypted and provides the requested database to client. The decryption can be done.

Frame Work Algorithm - I

Communication between Authentication and Access Protocol:

Step 1: Start

Step 2: Define the entities involved:

a. *User: The entity seeking access.*

b. *Key (Ki): Key used for authentication.*

c. *Session Key (Ski): A temporary key used for securing the session.*

d. *Database (DB): The data repository.*

e. *Real Server (R_S): The actual server providing access to resources.*

Step 3: User provides authentication details.

User→ Authentication Server (A_S)

(Authentication details could include a username and password.)

Step 4: Initiate the authentication process.

Step 5: Generate a token with a timestamp.

a. *This token serves as proof of authentication and may have a limited validity period.*

b. *Token is generated with a timestamp.*

Step 6: Grant user access to the Real Server-

User→ Real Server (*R_S*)

(Upon successful authentication, the user is given access to the Real Server.)

Frame Work Algorithm - II

Step 1: Continue (Continuing the protocol)

Step 2: User is given access to the Transaction Database (TDB).

User→ Transaction Data Base (TDB)

The user is granted access to the Transaction Database to perform operations or retrieve data.

Step 3: TDB performs operations such as data encryption (E_T) and potentially involves a Key Distribution Centre (KDC). TDB may encrypt the data and may interact with a Key Distribution Centre for managing cryptographic keys.

Step 4: Decrypt the data using the keys Ki, Di, and Ski.

DATA=Ki+Di+Ski

Decrypt the data stored in the Transaction Database using the appropriate keys.

Step 5: User can access the decrypted data.

(Access is granted to the data once it has been successfully decrypted.)

Step 6: Stop ()

Algorithm for Elgamal Key Generation

Step 1: Select a large prime P_N and choose an integer x, which must be less than P_N-2

Step 2: Select d to be a member of the group, and Pr_1 and Pr_2, as primitive roots.

Compute Pr_2=(Pr_1^d) mod P_N.

Step 3: Announce the public key publicly and keep the private key secret.

a. Public key: K_{PUB}= ($Pr1$, $Pr2$, P_N)

b. Private key: K_{PRI}=d_E, here d_E=a^x mod P_N.

Step 4: Return the public and private keys.

Algorithm for Elgamal Encryption

This algorithm outlines the process of generating public and private keys for the ElGamal cryptosystem. The public key consists of the prime number P_N, along with two primitive roots $Pr1$ and $Pr2$, while the private key is kept secret.

The security of the system relies on the difficulty of computing discrete logarithms in a finite field.

Step 1: Input the plaintext (P_T), prime number (P_N), and primitive roots ($Pr1$, $Pr2$).

Step 2: Select a random integer Rn in the group.

Step 3: Calculate the cipher text as follows:

Compute $CT1$=($Pr1^{Rn}$) mod P_N

Compute $CT2$=(PT×$Pr2^{Rn}$) mod P_N

Step 4: Return the cipher text CT1 and $CT2$.

Algorithm for Elgamal Decryption

Step 1: Input the private key dE, prime number (P_N), and cipher text components $CT1$ and $CT2$.

Step 2: Generate the plaintext as follows:

Calculate PT= ($CT2$× ($CT1$ dE) $^{-1}$) mod P_N

Step-3: Return Plain Text

COMMUNICATION BETWEEN OBJECTS (SERVERS)

Steps involved in this communication

1. User Initialization (<<M1>>):

Compute Rn=$Di \oplus h(Uid||Pwd)$

a. Select random numbers [[1, q-1], $r2$, and $r3$].

b. Initiate communication with the Authentication Server (As).

c. Generate session keys (SK_i), keys (K_i), and tokens (Ts).

d. Generate public and private keys (K_{PUB}, K_{PRI}).

$KPUB= (Pr1, Pr2, P_N), KPRI=d_E ,K_i= KPUB || KPRI$

2. User Authentication (<<M2>>):

   a. Compare Rs received from the Authentication Server with $DB+Rp$.

   b. Calculate $Rn=(Ki||a) \oplus x.$, $Pr2=Pr1d_E \%PN$, $E_T= (Pr1||Pr2 ||P_N||P_T)$

   c. Compute $CT_1$and CT_2 for encryption. , $CT1= Pr1^{Rn}\% P_N$, $CT2= (PT * Pr2^{Rn}) \% P_N$

   a. Encrypt the message using ElGamal encryption.

3. Transaction Handling (<< M3>>):

   a. Compute CT_1 and CT_2 for the transactional database.

   b. Verify the correctness of the encryption and decrypt the data using ElGamal decryption.

   $d_{E} = a^x\%P_N$, $D_T=(D_E.P_T,CT1,CT2)$, $PT=(CT2(CT1^{dE})^{-1})\%P_N$

4. Verification (<<M4>>): Verify the decrypted data against the encrypted data. $D_T= D_T\oplus E_T$

5. User Interaction with the Database: Access the database with the decrypted data. , $U_{ser}=DB$

Description Step by Step

This process consists of five phases Initialization, Login, Authentication, Key Distribution, and Cryptanalysis.

Initialization (Step 1)

Generation of groups & Parameters: The Authentication Server generates a group 'G' with generator number 'a', a member of the group'd', an integer 'x', and large prime 'P_N'.

Login (Step 2)

User Authentication Request:

The user chooses a random number Rn, their own identity Uid and password Pwd

a. Compute nonces $NIi=h(Rn||Uid)$

b. Compute $NPi=h(Rn||Pwd)$

c. Sends NIi, NPi, and Uid to the As.

d. Compute $Rn=Di\oplus h(Uid||Pwd)$ and sends <<M1>> to the As.

Authentication (Step 3)

a. Upon receiving the request, As computes $User=Rn\oplus h(Uid||Pwd)$

b. Checks user details in the database (DB) and activates the session by generating a key Ki.

Key Distribution (Step 4)

a. If user details are found and matched in DB, KDC generates a session key $Ki$and a token with timestamp Ts.

b. Generate $KPUB= (Pr1, Pr2, PN)$ and $KPRI=d_E$

c. Concatenate and pass the public and private keys to the user as a single key Ki

d. Send <<M2>> to the user.

Cryptanalysis (Step 5)

a. Upon receiving keys from KDC, the user gains access to the real server (RS) where the database is located.

b. R_S is calculated as $R_S=DB+R_P$.

c. Using ElGamal cryptography, encrypted data or text E_T can be decrypted in the Transactional Database (TDB) system.

SAMPLE CODE

```
import sys
import json
from cipher.elgamal_cipher import ElGamal
def spt(stng):
    """Splits the string into chunks of size 2."""
    return [stng[i: i + 2] for i in range(0, len(stng), 2)]
def alphabet_position(text):
    """Converts text to positions based on predefined mapping."""
    dictt = {
        'a': '101', 'b': '102', 'c': '103', 'd': '104', 'e': '105', 'f': '106', 'g': '107',
        'h': '108', 'i': '109', 'j': '110', 'k': '111', 'l': '112', 'm': '113', 'n': '114',
        'o': '115', 'p': '116', 'q': '117', 'r': '118', 's': '119', 't': '120', 'u': '121',
        'v': '122', 'w': '123', 'x': '124', 'y': '125', 'z': '126', '1': '131', '2': '132',
        '3': '133', '4': '134', '5': '135', '6': '136', '7': '137', '8': '138', '9': '139',
        '0': '130', ' ': '141', ',': '142', '.': '143'
    }
    arr = [dictt[i] for i in text.lower() if i in dictt]
    return ''.join(arr)
print("Client Access Authentication Server")
username = 'user'
password = 'password'
user_input = input("Username:\n")
if user_input == username:
    pass_input = input("Password:\n")
    if pass_input == password:
        print("Valid Authentication Details.")
        # Initialize ElGamal and generate keys
        try:
            elgamal = ElGamal()
            print("Generating keys...")
            keyset = elgamal.generate_keys()
            # Read input file safely
            try:
                with open("input.txt", "r") as file1:
                    txt = file1.read()
            except FileNotFoundError:
                print("Error: 'input.txt' file not found.")
                sys.exit(1)
            # Process the text and write to another file
            inp = alphabet_position(txt)
            with open("data.txt", "w") as ftemp:
                ftemp.write(inp)
            # Read and convert the processed data
            try:
                with open("data.txt", "r") as file2:
                    data = file2.read()
                if data.isdigit():
                    value = int(data)
                else:
                    raise ValueError("Data contains non-numeric values.")
            except ValueError as e:
                print(f"Error: {e}")
                sys.exit(1)
            # Encrypt the value
            print("Encrypting...")
            c, ke = elgamal.encrypt(value)
            print("Encrypted.")
            print("KEY:", c)
        except Exception as e:
            print(f"An error occurred: {e}")
    else:
        print("Invalid password.")
else:
    print("Invalid username.")
```

RESULT AS ALGORITHM ANALYSIS

A comparison of various cryptographic algorithms based on different parameters such as block size, key length, security, and performance:

1. AES-Advance Encryption System

a. Block Size: 128 bits

b. Key Length: 128, 256 bits

c. Security: Good

d. Performance: High

2. RSA-Rivest-Shamir-Adelman

a. Block Size: Not applicable (RSA is an asymmetric algorithm)

b. Key Length: 512 bits (for encryption), 1024 bits (for signature)

c. Security: Less

d. Performance: Minimum

3. ECC-Elliptic Curve Cryptography

e. Block Size: Not applicable (ECC is a public-key algorithm)

f. Key Length: 256 bits (equivalent to RSA 3072-bit key)

g. Security: Good

h. Performance: High

4. DES-Data Encryption Standards

a. Block Size: 64 bits

b. Key Length: 56 bits

c. Security: Less

d. Performance: Minimum

5. Blowfish Algorithm

a. Block Size: 64 bits

b. Key Length: 32-448 bits

c. Security: Good

d. Performance: High

6. Elgamal Cryptography Technique

a. Block Size: Not applicable (ElGamal is an asymmetric algorithm)

b. Key Length: 512 bits (for encryption)

c. 1024 bits (for signature)

d. Security: Good

e. Performance: High

These assessments are relative and depend on various factors such as the specific application, computing resources available, and the threat model. Additionally, advancements in technology and cryptanalysis may impact the security and performance of these algorithms over time. Here the client gets access to the real server, where the transactional database resides in it. In this coding, with the help of key and using Elgamal cryptography technique the data is decrypted. It provides a comparison of different cryptographic algorithms based on their block size, key length, security, and performance.

FUTURE SCOPE AND CONCLUSION

The section concludes by summarizing the proposed security techniques implemented in the system, which involve authentication server and cryptography techniques. It highlights the importance of traversing through multiple verification modules to reach the real server and mentions the use of ElGamal security model for end-to-end security. The future work is aimed at enhancing the security architecture using AVISPA. Overall, this section provides a comprehensive analysis of cryptographic algorithms and their implications for system security. It emphasizes the importance of selecting appropriate algorithms based on security requirements and performance considerations. a list of research papers related to security in various aspects of technology, particularly focusing on the Internet of Things (IoT) and cyber-physical systems.

REFERENCES

[1] Shibi, S. Rimlon and Prabu, R. Thandaiah, Enhancing Security and Privacy in Healthcare IoT Through Multi-Layered Security Frameworks (November 15, 2024). Proceedings of the 3rd International Conference on Optimization Techniques in the Field of Engineering (ICOFE-2024),http://dx.doi.org/10.2139/ssrn.5088965

[2] Y. Sun, F. P. . -W. Lo and B. Lo, "Lightweight Internet of Things Device Authentication, Encryption, and Key Distribution Using End-to-End Neural Cryptosystems," in *IEEE Internet of Things Journal*, vol. 9, no. 16, pp. 14978-14987, 15 Aug.15, 2022, doi: 10.1109/JIOT.2021.3067036

[3] Ghosh, S.; Verma, S.K.; Ghosh, U.; Al-Numay, M. Improved End-to-End Data Security Approach for Cloud Computing. *Sustainability* **2023**, *15*, 16010. https://doi.org/10.3390/su152216010

[4] M. Shahid Dildar, A. Shahid Khan, I. A. Abbasi, R. Shaheen Naseem Akhtar, K. Al Ruqaishi and S. Ahmed Ghulam Sarwar, "End-to-End Security Mechanism Using Blockchain for Industrial Internet of Things," in *IEEE Access*, vol. 13, pp. 20584-20598, 2025, doi: 10.1109/ACCESS.2025.3535821.

[5] Wang, Guoai , Wenting, " A Secure and Anonymous Two-Factor Authentication Protocol in Multiserver Environment".PP.1-16,(2018).

[6] FrazBaig, Hassan, Ghani, Ashraf Chaudhry, Imran Khan, Usman Ashraf, "A lightweight and secure two factor anonymous authentication protocol for Global Mobility Networks". PP.1-21 (2018).

[7] Moon,SookLee, JiyeKim, DonghoWon,"Improving an Anonymous and Provably Secure Authentication Protocol for a Mobile User".PP.1-14,(2017).

[8] FerreiraJesus,Vanessa R.L.Chicarino, CélioV.N., deA.Rocha, "A Survey of How to Use Blockchain to Secure Internet of Things and the Stalker Attack".PP.1-28,(8 April 2018).

[9] Jeans, Angelo, Mirko Perkusich, "A Smart Trust Management Method to Detect On-Off Attacks in the Internet of Things". PP. 1-11, (15 April 2018).

[10] QingheDu, YingXu, WanyuLi,HoubingSong, "Security Enhancement for Multicast over Internet of Things by Dynamically Constructed Fountain Codes" .PP.1-12,(2018).

[11] TianqiZhou,JianShen , XiongLi, ChenWang,JunShen "Quantum Cryptography for the Future Internet and the Security Analysis".PP.1-8,(2018).

[12] ChenWang, JianShen, QiLiu, YongjunRen,TongLi, "A Novel Security Scheme Based on Instant Encrypted Transmission for Internet of Things".PP.1-8,(2018).

[13] QiLi, HongboZhu, Zuobin,TaoZhang, "Traceable Cipher text-Policy Attribute-Based Encryption with Verifiable Outsourced Decryption in eHealth Cloud".PP.1-13,(2018).

[14] Wilson S. MeloJr., RaphaelC.S.Machado, "Using Physical Context-Based Authentication against External Attacks: Models and Protocols" .PP.1-15,(2018).

[15] QianmuLi, Hiuk, "A Comparative Analysis of Information Hiding Techniques for Copyright Protection of Text Documents".PP . 1-23,(2018).

[16] Batool, MuazzamKhan, "Internet of Things Data Analytics for User Authentication and Activity Recognition",PP . 1-5(2017).

[17] Vimal Jerald, Albert Rabara, Daisy Premila, "Algorithmic Approach to security Architecture for Integrated IoT Smart Services Environment", PP.1-6(2017).

[18] Christos,Kostas.psamis, plagres, "Architecture for Security monitoring in IOT environment",PP.1-4,(2017).

[19] Praveena, "Achieving Data Security in Wireless Sensor Networks Using Ultra Encryption Standard Version – IV Algorithm", PP.1-5, (2017) .

[20] Mauricio, Samy El-Tawab, Heydari, "IoT Security Attacks Using Reverse Engineering Methods on WSN Applications", PP.1-6, (2017).

[21] Israr Ahmed, Saleel A.P, BabakBeheshti, Zahoor Ali Khan, Imtiaz, "Security in the Internet of Things (IoT)", PP.1-7(2017). 18. NagasimhaSwamy, DiptiJadhav, Nikita, "Security Threats in the Application layer in IOT Applications", PP.1-4, (2017)

[22] Bei Gong,YuboWang,Liu, Fazhi Qi,ZhihuiSun,"A Trusted Attestation Mechanism for the Sensing Nodes of Internet of Things Based on Dynamic Trusted Measurement",PP.122,(February 2018).

[23] Lin Hu, Wen, Bin Wu, Fei Pan, Liao, Huanhuan, Jie Tang, Wang, "Cooperative Jamming for Physical Layer Security Enhancement in Internet of Things", PP. 110, (2017).

[24] Vinay, Abhishek, Bose, "Channel-Based Mapping Diversity for Enhancing the Physical Layer Security in the Internet of Things", PP.16, (2017).

[25] AndreAsBurg, AnupAmCh, KwoK "Wireless Communication and Security Issues for Cyber– Physical Systems and the Internet-ofThings", PP1-23, (2018).

[26] Rachad, Liu, Jonathan Ashdown, "A Physical Layer Security Scheme for Mobile Health Cyber-Physical Systems", PP.1-15, (2017).

[27] Sharma, "A Survey on Layered Approach for Internet of Things Security", SERSC, ASTL, SMART DSC-2017, vol. 147, pp. 26-33, (2017)

[28] S.K.Sharma, Namkyun Baik, Bonomail Khuntia."Encrusted Security for Internet of Things using MAC-OMURA".IJCA,SERSC- Australia.PP.45-54, (2018).

[29] Jean Caminha, Mirko Perkusich, "A Smart Trust Management Method to Detect On-Off attacks in the Internet of Things". pp. 1-10, (2018)

[30] Qinghe Du, Houbing, "Security Enhancement for Multicast over Internet of Things by Dynamically Constructed Fountain Codes, pp.111, (2018).

[31] Zhou, Jian, Shen, Xiong Li,Chen Wang, "Quantum Cryptography for the Future Internet and the Security Analysis".pp.1-7,(2018).

[32] Wang, Ren, Tong Li "A Novel Security Scheme Based on Instant Encrypted Transmission for Internet of Things".pp. 1-7, (2018).

[33] Qi Li, HongboZhu,Tao Zhang, "Traceable Cipher text-Policy Attribute-Based Encryption with Verifiable Outsourced Decryption in eHealth Cloud".pp.1-12,(2018).