ISSN: 2350-0328

International Journal of Advanced Research in Science, Engineering and Technology

Vol. 4, Issue 10 , October 2017

Design and Development of Novel Secured Key Schema for RSA CRYPT System

Jagadheesan.V, Dr.S.Prema

M. Phil Research Scholar, Department of Computer Science (PG), K.S.Rangasamy College of Arts and Science (Autonomous), Thiruchengod, Tamilnadu, India

Associate Professor, Department of Computer Science (PG), K.S.Rangasamy College of Arts and Science (Autonomous), Thiruchengod, Tamilnadu, India

ABSTRACT: Network and Information are the two major power players of the era. Cryptography is the mechanism to store the information securely without the intervention of the third party. Many organizations work with public networks where the security breaches happen in wide manner. The process of encoding messages to make them non-readable by the third party users is the goal of the cryptography. Internet of Things (IoT) creates a profound impact in last few years. The concept implied is connecting every digital gadget with the internet and utilize the resource involved in it. In present day scenario, mobile phones which connect different gadgets such as pervasive devices are apt samples which depict the future focus. The need as of now is wavering from devices with high computational capability to restricted resources (Storage, CPU, energy etc.). As a consequence, it is necessary to make communications between hybrid networks (lossy and lossless), hybrid environments, and hybrid user behaviors. The necessity of focusing on cryptography schemes which may work on network of IoT is the problem focused on the proposed research. The key schemes can imply a major role while shifting data from hybrid environments. The proposed work focus on two main aspects: (1) unique key scheme variant which may be apt for IoT; (2) Framework to deploy the algorithm in IoT.

The distinctive technique used in communal key cryptography is the use of asymmetric key algorithms, where a key used by one gathering to perform encryption is not the same as the key used by another in decryption. All operators have a couple of cryptographic keys – a public encryption key and a private decryption key.RSA is the greatest general public key cryptography (PKC). It uses 2 vital cryptosystem, a public key which is known by the sender and the receiver and a private key which is known only by the receiver, so that two gatherings can engage in a protected announcement over a non-secure announcement station without having to stake key.RSA key is a private key constructed on RSA algorithm. Private Key is used for authentication and a symmetric key exchange during establishment of an SSL/TLS session. It is a part of the public key organization that is normally used in situation of SSL certificates.RSA is a cryptosystem which is recognized as one of the first attainable public-key cryptosystems and is broadly used for secure data transmission. In this outline, one key (the public key) is used to encrypt the message while a dissimilar key (the private key) is used to decrypt it.This research instigates a variant of the RSA algorithm utilizing varying protracted strategy for the key schemes. It can be utilized for the un-reliable parties. The algorithm uses a public key and private key. The RSA encrypted message can be upgraded for the un-reliable parties by using the intermediate keys in IoT. Considering the private and public keys present in hybrid environments of IoT, an new intermediate key will be generated.

The new intermediate key will be distributed to the unreliable parties without revealing the plain text. The novel secured key scheme for RSA (NSK_RSA) algorithm is implemented using the MATLAB. Evaluation metrics deployed are throughput, encryption time and decryption time. Results are simulated on hybrid environments. The performance of the proposed system seems to be promising.NSK_RSA schemes allow for a cipher text to be re-encrypted an unlimited number of times. For example, a cipher text might be re-encrypted from Bob to Charlie, and then again from Charlie to David and so on. Non-transitive schemes allow for only one (or a limited number) of re-encryptions on a given cipher text. The cryptographic key scheme proposed seems to be apt for the IoT where the need for the security improves in drastic manner.

I. INTRODUCTION

Present cryptography concerns itself with the succeeding four objectives are isolation, integrity, Non-repudiation, confirmation. Events and procedures that assemble a number of or all of the above criteria are known as cryptosystems. Cryptosystems are often consideration to refer only to numerical events and workstation programs; however, they also hold the instruction of human presentation, such as choosing hard-to-guess passwords, sorting off out of work schemes, and not conversing responsive proceedings with outsiders. The enlargement of the Internet and electronic business has brought to the front position the issue of isolation in electronic communication. Large volumes of individual and responsive information are automatically transmitted and stored every day. What an agreement does one have that a message sent to one more person is not intercepted and read without their knowledge or approval? Tools to make sure the isolation and privacy of paper-based message have existed for a long time. Similar tools exist in the electronic communications arena. Encryption is the normal technique for creation a communication personal. Anyone deficient to send a private message to another user encrypts (enciphers) the communication before transmitting it. Only the planned recipient knows how to properly decrypt (decipher) the communication. Anyone who was "eavesdropping" on the announcement would only see the encrypted communication. Because they would not know how to decrypt it successfully, the communication would make no sense to them. As such, privacy can be ensured in electronic communication. The Rivest-Shamir-Adleman (RSA) algorithm is one of the majorities accepted and secures public-key encryption techniques. The procedure capitalizes on the fact that there is no efficient way to factor very large (100-200 digit) numbers.

Using an encryption key (e,n), the algorithm is as follows:

1. Represent the message as an integer between 0 and (n-1). Large messages can be broken up into a number of blocks. Each block would then be represented by an integer in the same range.
2. Encrypt the message by raising it to the eth power modulo n. The result is a cipher text message C.
3. To decrypt cipher text message C, raise it to another power d modulo n

The encryption key (e,n) is made public. The decryption key (d,n) is kept private by the user.

1. Choose two very large (100+ digit) prime numbers. Denote these numbers as p and q.
2. Set n equal to p * q.
3. Choose any large integer, d, such that GCD(d, ((p-1) * (q-1))) = 1
4. Find e such that e * d = 1 (**mod** ((p-1) * (q-1)))

Cryptographic techniques cannot be proven secure. As an alternative, the only test is to see if an important person can form out how to decipher a communication without having straight information of the decryption key. The RSA method's safety rests on the fact that it is enormously hard to factor very large numbers. If 100 digit numbers are used for p and q, the resulting n will be approximately 200 digits. The greatest known factoring algorithm would take far too long for an attacker to ever break the code. Other methods for determining d not including factoring n are uniformly as hard. Any cryptographic method which can refuse to accept a concerted attack is regarded as secure.

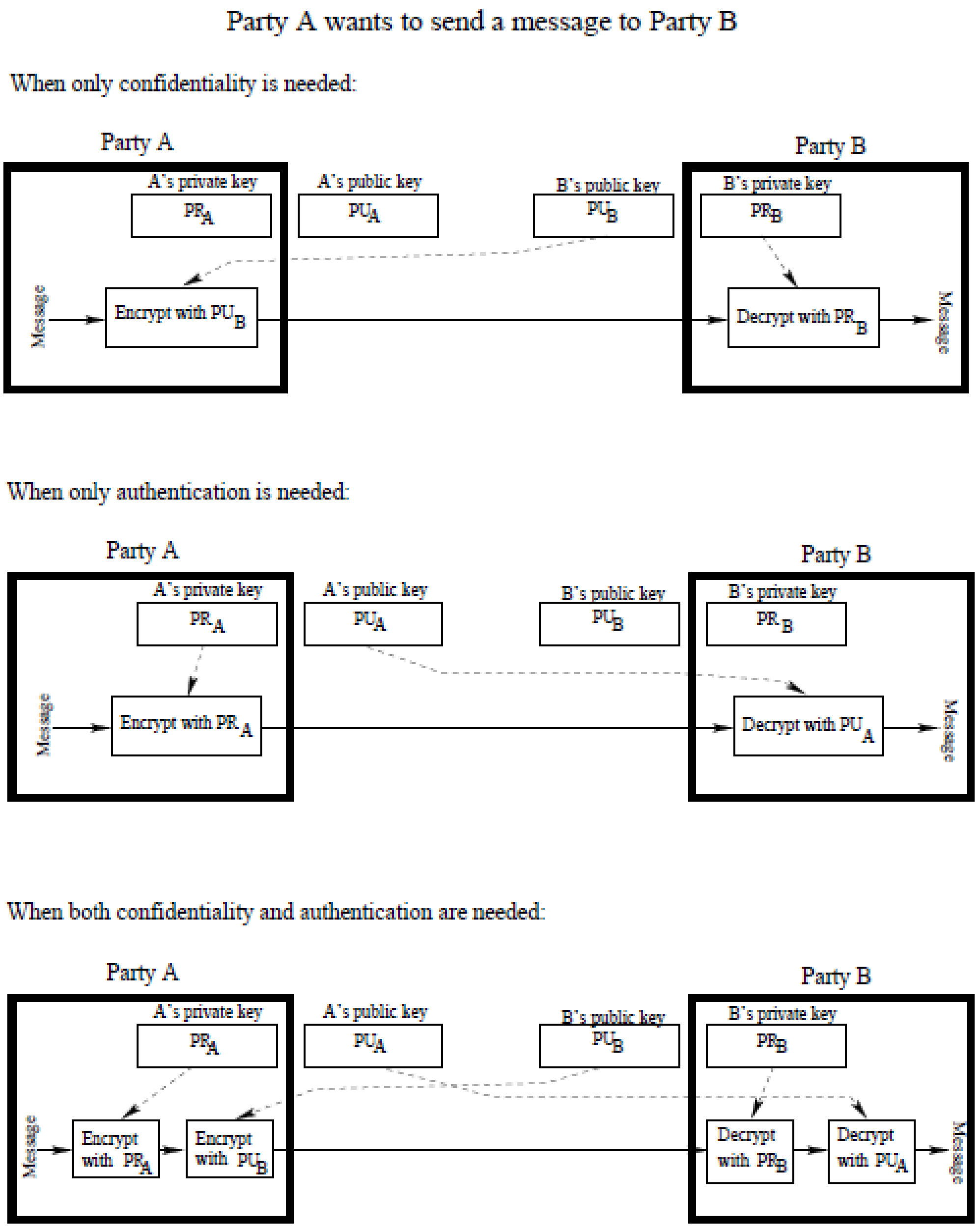

Figure 1.Sending and receiving of data

II. LITRATURE REVIEW

James Aspnes[1] is concerned in troubles concerned with securing enormous disconnected algorithms nearby trouble by unreliable donor. Using cryptographic methods, it may be possible to authorize in-between consequences in a scattered algorithm to be practiced separately of who presents them, reducing the problem of deciding which equipments to belief. These troubles twist out to be mostly necessary in systems, such as peer-to-peer networks, where association with the system is planned and cannot be imperfect only to equipment beneath the categorize of the algorithm luxurious.

Joan Feigenbaum[2] is concerned in the fundamentals of electronic business and in basic evils in difficulty suggestion that are forced by cryptology. One such complexity is the power of "instance-hiding" computations. The owner of a secret database uses the better dispensation power of one or more additional equipments without having to expose the record to that equipment. It is established that 1) the instance-hiding computations are imperfect in authority if the private-database proprietor can only discuss with a single other engine; 2) that they are enormously authoritative if the proprietor can discuss with numerous other equipments, and 3) that illustration thrashing is directly connected to some of the middle subjects of difficulty hypothesis.

Michael Fischer [3] is concerned in security harms connected with Internet collection and extra usually in hope and safekeeping in joint additions. He has been raising a replicated civilization in which faith has a precise algorithmic meaning. In this environment, faith can be well-informed and used for decision structure. Better choices lead to enhanced community accomplishment. This building agrees for the improvement and examination of some very easy algorithms for education and exploiting faith that are simply implementable in a variety of setting and are doubtfully like to what public frequently use in on a daily base life.

Zhong Shao[4] leads the FLINT group at Yale, which is raising a method for confined portable structure based on authentication logics, proof-carrying code, and type-based confirming compilers. Verification logics are official logics that authorize one to reason about the possessions of systems and measures that confirm the independence of users and make a decision whether or not to allow a range of processes. Modeling such schemes provides the common profits of formal study: hidden statements are made clear, unneeded features are uncovered, and errors in the system may be found. Proof-carrying code (PCC) allows a code creator to offer a (compiled) agenda to a host, the length of with an official proof of security. The host can recognize a safety policy and a set of adages for computation about guard the producer’s verification must be in conditions of those proverbs. Type-based certifying compilers are compilers that use unmoving type in run to help create provably safe target code. These technologies fit jointly naturally and form the basis for present confined mobile-code system.

III. SYSTEM ANALYSIS

Dworkin presents an in detail literature survey of accessible cryptographic organizer systems. The segment starts with an explanation of different plan goals and design parameters that be supposed to be measured while scheming a cryptographic file system. A brief explanation of a variety of ciphers and modes of operations used by accessible cryptographic file systems has been offered along with a exhaustive explanation of XEX-based Tweaked codebook mode with cipher text Stealing (XTS) that can be used by cryptographic file systems for better presentation. Then, existing cryptographic file systems at the block device level and at file system level in user-space and in kernel space are obtainable with their rewards and restrictions. Further, a brief appraisal of trusted computing technologies and profits of by means of them for key organization in cryptographic file systems has been explained. Finally, synopsis of the properties of accessible cryptographic file systems has been offered along with the problems recognized for shipping out explore job.

A. RSA cryptography

The a variety of comments immediately affirmed form the basis for the RSA public-key cryptosystem,
this was made-up at MIT in 1977 by Ronald Rivest, Adi Shamir and Leonard Adleman. The public key in this cryptosystem consists of the value n, which is called the modulus, and the value e, which is called the communal proponent. The private key consists of the modulus n and the value d, which is called the confidential proponent.

An RSA public-key / private-key pair can be generated by the following steps:

1. Produce a pair of large, random prime’s p and q.
2. Calculate the modulus n as n= pq.
3. Select an odd public exponent e between 3 and n-1 that is relatively prime to p-1 and q-1.
4. Compute the private exponent d from e, p and q.
5. Production (n, e) as the public key and (n, d) as the private key. The encryption operation in the RSA cryptosystem is exponentiation to the e^{th} power modulo n:

$$C=\mathrm{ENCRYPT}\ (m) = m^{e} \mathrm{mod}\ n$$

The input m is the message; the output c is the resulting cipher text. In practice, the message m is typically some kind of appropriately formatted key to be shared. The actual message is encrypted with the shared key using a traditional

encryption algorithm. This construction makes it possible to encrypt a message of any length with only one exponentiation.

The decryption operation is exponentiation to the dth power modulo n:

$$m = DECRYPT(c) = c^{d} mod\ n.$$

The relationship between the exponent's e and d ensures that encryption and decryption are inverses, so that the decryption operation recovers the original message m. Without the private key (n, d) (or equivalently the prime factors p and q), it's difficult to recover m from c. Consequently, n and e can be made public without compromising security, which is the basic requirement for a public-key cryptosystem. The fact that the encryption and decryption operations are inverses and operate on the same set of inputs also means that the operations can be employed in reverse order to obtain a digital signature scheme following Diffie and Hellman's model. A message can be digitally signed by applying the decryption operation to it, i.e., by exponentiation it to the dthpower:

$$s= SIGN\ (m) = m^{d} mod\ n$$

The digital signature can then be verified by applying the encryption procedure to it and evaluating the result with and/or improving the message:

$$m= VERIFY\ (s) = s^{e} mod\ n.$$

Rijndael developers sought a plan that has a low connection between input bits and output bits, and the goods that the output cannot be described as a simple mathematical utility of the input. In addition, the s-box has no permanent points (s-box (a) = a) and no contrary permanent points (s-box (a) =−a) where −a is the bitwise compliment of a. The s-box must be invertible if decryption is to be potential (Is-box[s-box (a)] = a) however it should not be its self inverse i.e. s-box (a) 6= Is-box (a).

IV. DRAWBACKS OF THE EXISTING SYSTEM

The lack of the security mechanism which is compatible for the IoT such as cryptography do possess the following drawbacks:

a. Loss of reputation. "The battery that you (manufacturer 'x') claimed as genuine has exploded in my laptop."
b. Loss of IP. "The terrific algorithm I've developed in my video decoder during the last five years has been copied and duplicated. And I did not patent it to avoid disclosure of my tricks!"
c. Loss of money. "Tens of payment terminals are hacked in my retail chain store, so fake transactions are performed and/or cardholder sensitive data are stolen. Customers are going to blame me and I will need to identify the hackers."
d. Loss of goods. "I just read about the hack of an energy meter published on the web and already thousands of dishonest subscribers are implementing it to pay a lower bill."
e. Loss of health. "My insulin pump does not dispense any more, or it dispenses too much. Who ordered a change in delivery times?"
f. Loss of control of vital infrastructures. "Who turned the lights off in the whole city?

V. NEED FOR PROPOSED WORK

Security is a primary concern for the Internet of Things, as important as minimum power consumption, affordability, and wireless connectivity. Hence IoT devices are optimized for little power consumption and affordability, many have fewer than optimal computing resources. The good information is there are quite a lot of options for using cryptography to make it more complicated for hackers to highjack user living room webcam, video doorbell or car.

For the IoT, authentication ensures that devices are interacting with authorized gateways and cloud services and they in turn authorize that they are dealing with authentic IoT nodes. The sender will use a hashing method and shared secret keys to produce a tag known as a message authentication code (MAC). The receiver act upon the same hashing algorithm to decode the MAC and evaluate it with one stored locally.

The key aspect of the MAC depends on the complexity of the hashing algorithm, the length of the key used and whether the key is shared secretly and stored securely. The current state-of-the-art hashing method for cryptographic aims is SHA-256 with 256-bit keys.

A. ADVANTAGES OF THE PROPOSED SYSTEM

The cryptographic method defined and implemented in the proposed system has considered the following aspects:

1. Perform the legal/contractual loom. This possibility is always cost effective and worth setting up. A device manufacturer can ask subcontractors (e.g., manufacturing plants) to sign a nondisclosure agreement (NDA) and to assure to be honest and faithful.
2. Execute technical countermeasures. These steps will care for devices against deceitful partners, subcontractors, and outlaw/unreachable attackers.
3. Technical countermeasures assure that a device's banal behavior and functions are forbidden, sealed, and sanctioned by the manufacturer. Nothing can then either amend defined operation or entrée protected functions.

B . MOTIVATION OF THE RESEARCH

Encryption has been worn for millennia. Ancient Greek generals accepted messages to each other encoded on leather strips. Today AES is the established standard to encrypt and decrypt messages using digital keys. Symmetric key cryptography deploys the same key to encrypt and decrypt the message, making it critical to keep the key secret. Asymmetric cryptography deploys a shared, public key and a private key which is been secret.

While asymmetric key cryptography has the benefit of added security over insecure channels, it's more than 1,000 times more computationally expensive than symmetric key cryptography. Asymmetric cryptography can be employed to establish a secure channel to handover secret keys which can be used for succeeding symmetric methods. On the other hand, symmetric key cryptography used along with Diffie–Hellman key exchange is frequently secure enough for different embedded applications.

In IoT devices, hardware acceleration creates ample information. Authentication chips or cryptographic co-processors can carry out complicated encryption and authentication competently in hardware, saving battery life and processor cycles. It consumes more effort to secure any allied computing device, but in the long run, it's the right thing to do.

VI. IMPLEMENTATION OF ALGORITHM

A. RSA cryptography – formal method of working

The working of RSA cryptography method is explained as follows:
Step 1: Start the process:
Step 2: Consider two large prime numbers x & y.
Step 3: Calculate: N = x*y Where N is the factor of two large prime number.
Step 4: Choose an Encryption key (E) subject to the condition that it should not be a factor of (x-1)*(y-1) and assign Ø(n)= (x-1)*(y-1)
The assumption for the Encryption Key E should be, 1< E < Ø(n) and gcd(E, Ø(n)=1
The intention behind gcd calculation is that E & Ø(n) has to be relative prime.
Step 5: Choose the Decryption key (D), which satisfy the condition that D*E mod (x-1)*(y-1) = 1
Step 6: Encryption Phase: Cipher Text= (Plain Text) E mod N
Step 7: Decryption Phase: Plain Text= (Cipher Text) E mod N
Step 8: Stop the process

Table 1: Comparison of Message Size and Time taken

Message Size	RSA	Enhanced RSA	Proposed Method
10 KB	0.00578	0.00057	0.00226
20 KB	0.03859	0.00223	0.00246
30 KB	0.01946	0.0035	0.00379
40 KB	0.02767	0.00624	0.00659
50 KB	0.03322	0.00686	0.00729
100 KB	0.07127	0.01432	0.01569
200 KB	0.15799	0.03022	0.03086

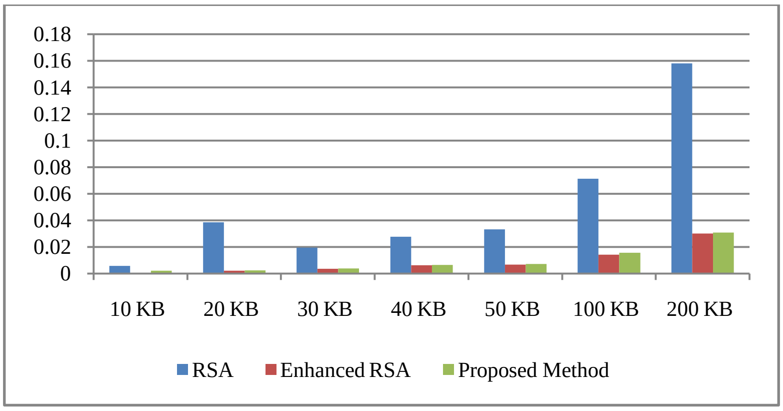

Figure 3. Comparison of Message Size and Time taken

Table 2: Comparison of the Mean time

Message Size	RSA	Enhanced RSA	Proposed Method
Mean Time	0.04666	0.00913	0.00985

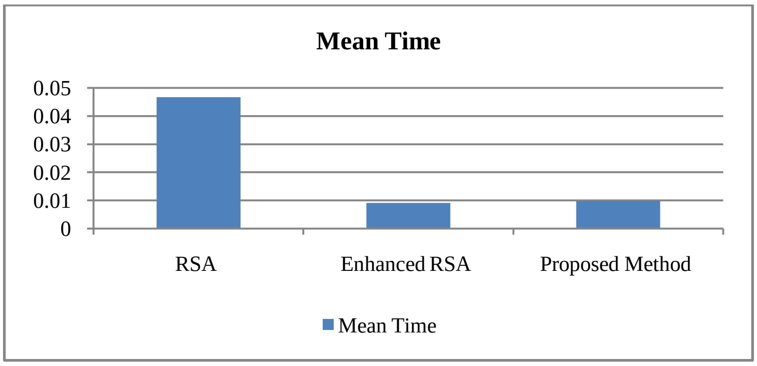

Figure 4. Comparison of the Mean time

Table 3: Throughput comparison

Message Size	RSA	Enhanced RSA	Proposed Method
Throughput (MBPS)	4.04969	0.92116	4.33759

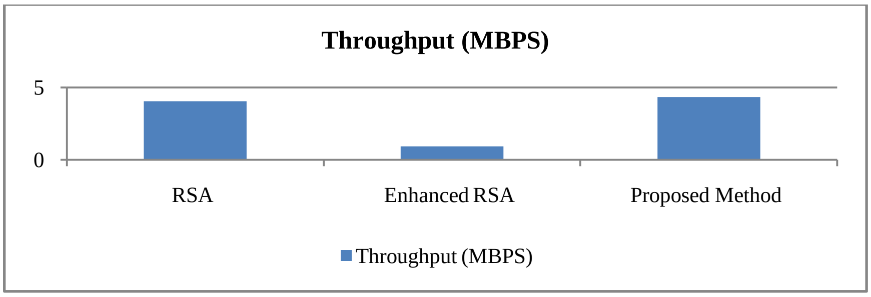

Figure 5. Throughput comparison

VII. CONCLUSION

In the near future Internet of Things will be an essential element of our daily lives. Copious energy embarrassed devices and sensors will always be communicating with each supplementary the security of which be obliged to not be compromised. For this rationale a lightweight security method is proposed in this research named as NSK_RSA for IoT. The accomplishment show capable results making the method an apposite entrant to be adopted in IoT applications. In the near future we are interested in the detail performance evaluation and cryptanalysis of this algorithm on different hardware and software platforms for possible attacks.

VIII. FUTURE WORK

For prospect exploration, the accomplishment of the method on hardware and software in a variety of computation and network upbringing is under contemplation. Likewise, the method can be tuned in order to augment the performance according to variant hardware environments.

REFERENCES

[1] James Aspnes, PhD (CS) 1992, Carnegie-Mellon University; SM (EECS) 1987, Massachusetts Institute of Technology; SB (Math) 1987, Massachusetts Institute of Technology.

[2] Joan Feigenbaum, B.A., Mathematics, Harvard, 1981 Ph.D., Computer Science, Stanford, 1986. Joined Yale Faculty 2000.

[3] Michael Fischer, B.S., University of Michigan, 1963 M.A., Ph.D., Harvard University, 1965, 1968.

[4] Zhong Shao, B.S., University of Science and Technology of China, 1988 M.A., Ph.D., Princeton University, 1991, 1994.

[5] PriyaTrivedi, Sanya Harneja , "Network Security Issues and Cryptography", International Journal of Research (IJR) Vol-1, Issue-10 November 2014

[6] Avi Kak, Public-Key Cryptography and the RSA Algorithm, Lecture Notes on "Computer and Network Security", February 16, 2017

[7] Chaitra B,Kiran Kumar V.G,,Shatharama Rai C, "A Survey on Various Lightweight Cryptographic Algorithms on FPGA", IOSR Journal of Electronics and Communication Engineering (IOSR-JECE), Volume 12, Issue 1, Ver. II (Jan.-Feb. 2017), PP 54-59

[8] Vermesan, Ovidiu; Friess, Peter (2013). Internet of Things: Converging Technologies for Smart Environments and Integrated Ecosystems (PDF). Aalborg, Denmark: River Publishers. ISBN 978-87-92982-96-4.

[9] Brown, Eric (13 September 2016). "Who Needs the Internet of Things?", Linux.com. Retrieved 23 October 2016.

[10] Hwang, Jong-Sung; Choe, Young Han (February 2013). "Smart Cities Seoul: a case study" (PDF). ITU-T Technology Watch. Retrieved 23 October 2016.