International Journal of Engineering Research and Development
e-ISSN: 2278-067X, p-ISSN: 2278-800X, www.ijerd.com
Volume 10, Issue 8 (August 2014), PP.60-66

Message Authentication and Source Privacy in Wireless Sensor Network using Rivest Cipher 6 Algorithm

[1]Ms. Sonam A. Bais, [2]Prof. Animesh R. Tayal
[1]*Computer Technology Priyadarshini College of Engineering Nagpur, India*
[2]*Computer Technology Priyadarshini College of Engineering Nagpur, India*

Abstract:- Message authentication is one of the most efficient ways to prevent unauthorized and corrupted messages from being forwarded in wireless sensor networks (WSNs). That's why, numerous message authentication proposals have been developed based on either symmetric-key cryptosystems or public-key cryptosystems. Many of them, have the restrictions of high computational and communication overhead in addition to lack of scalability and resilience to node compromise attacks. Wireless Sensor Networks (WSN) are being very popular day by day, however one of the main concern in wireless sensor network (WSN) is its limited resources. One have to look to the resources to generate Message Authentication Code (MAC) keeping in mind the feasibility of method used for the sensor network at hand. This paper the message authentication and source privacy in wireless sensor environment by using Rivest cipher version 6 (RC6) algorithms. It also compares various features in terms of computational overhead, energy consumption, message delay, memory consumption.

Index Terms:- Hop-by-hop message authentication, symmetric-key cryptosystem, public-key cryptosystem, source privacy, simulation, wireless sensor networks (WSNs), RC6 algorithm (Rivest cipher version 6).

I. INTRODUCTION

Message authentication performs a very important role in thwarting unauthorized and corrupted messages from being delivered in networks to save the valuable sensor energy [9]. Therefore, many authentication schemes have been proposed in literature to offer message authenticity and integrity verification for wireless sensor networks (WSNs) [4]–[9]. These approaches can largely be separated into two categories: public-key based approaches and symmetric-key based approaches[13].The symmetric-key based approach necessitates composite key management, lacks of scalability, not flexible to large numbers of node compromise attacks since the message sender and the receiver have to share a secret key. The shared key is handled by the sender to produce a message authentication code (MAC) for each transmitted message [8]. However, for this process the authenticity and integrity of the message can only be confirmed by the node with the shared secret key, which is usually shared by a group of sensor nodes [11]. An intruder can compromise the key by incarcerating a single sensor node. In addition, this method is not useful in multicast networks.

For the public-key based method, each message is transmitted along with the digital signature of the message produced using the sender's private key [10]. Every intermediate forwarder and the final receiver can authenticate the message using the sender's public key [16], [11]. One of the restrictions of the public key based method is the high computational overhead. To solve the scalability problem a secret polynomial based message authentication scheme was introduced. The idea of this scheme is similar to a threshold secret sharing where the threshold is determined, by the degree of the polynomial [7]. This approach offers information-theoretic security of the shared secret key when the number of messages transmitted is less than the threshold [5]. The intermediate nodes verify the authenticity of the message through a polynomial evaluation [6]. However, when the number of messages transmitted is larger than the threshold the polynomial can be fully recovered and the system is completely broken. An alternative solution was proposed in [4] to thwart the intruder from recovering the polynomial by computing the coefficients of the polynomial l[5]. The idea is to add a random noise called a perturbation factor to the polynomial that why, the coefficients of the polynomial cannot be easily solved [2]. However, a recent study shows that the random noise can be completely removed from the polynomial using error-correcting code techniques [15].

For the public-key based approach, each message is transmitted along with the digital signature of the message generated using the sender's private key. Every intermediate forwarder and the final receiver can authenticate the message using the sender's public key [11], [8]. One of the limitations of the public key based scheme is the high computational overhead [6]. By comparing all the limitation and drawback over the public

key cryptography, in this paper we propose the RC6 algorithm for providing high security in wireless sensor environment in terms of message authentication as well as source privacy.

II. LITERATURE REVIEW

A .Wireless sensor networks

Wireless sensor networks simplify the compilation and scrutiny of information from multiple locations [3]. The term wireless sensor network (WSN) illustrates an association among miniaturized embedded communication devices that supervise and evaluate their surrounding environment. The network is composed of many minute nodes sometimes referred to as motes [5]. A node is made up of the sensor(s), the microcontroller, the radio communication component, and a power source. Wireless sensor nodes range in size from a few millimeters to the size of a handheld computer. Apart from of size, sensor nodes share general constraints.

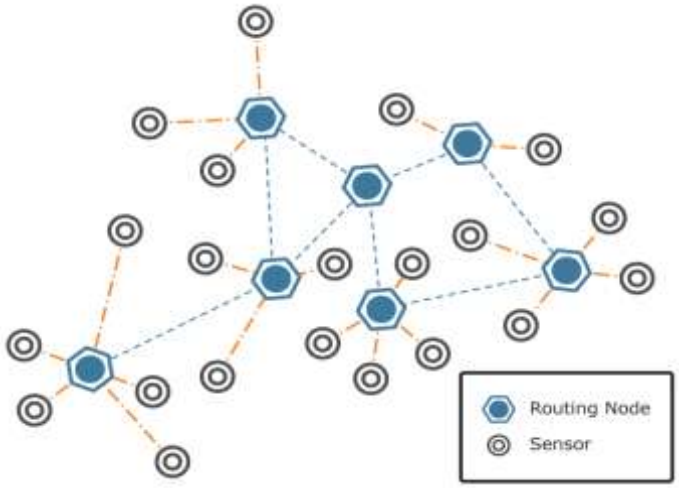

Figure 1: wireless sensor network environment

Security risks in wireless sensor networks contain threats to the confidentiality, integrity, and a availability of the system. Security methods used on the Internet are not simply adaptable to sensor networks because of the limited resources of the sensors and the ad-hoc feature of the networks. In this paper we propose hop by hop message authentication by using RC6 algorithm.

B. NS-2

NS (Version-2) is an object oriented, discrete event simulator. It was written in C++ with OTcl use as a front- end[10]. The simulator supports a class hierarchy in C++ (compiled hierarchy) and a similar class hierarchy within the OTcl interpreter (interpreted hierarchy) include. The two hierarchies are closely related to each other; from the user's point of view, there is a one-to-one relation between a class in the interpreted hierarchy and one in the compiled hierarchy.

Network simulator uses two languages because simulator has two different kinds of things it needs to do. On one side, detailed simulations of protocols requires a systems programming language which can efficiently manipulate bytes, packet headers, implement algorithms that run over large data sets[12]. For these tasks run-time speed is important and turn-around time (run simulation, find bug, fix bug, recompile, re-run) is less important. On the other side, a large part of network research involves slightly varying parameters or configurations, or quickly exploring a number of scenarios [10]-[12]. In these cases, iteration time (change the model and re-run) is more important. Since configuration runs once (at the beginning of the simulation) [5], run-time of this part of the task is less important.

Ns meets both of these needs with two languages, C++ and OTcl [5].C++ is fast to run but slow to change, make it suitable for detailed protocol implementation. An OTcl runs very slower but can be changed very quickly (and interactively), making it ideal for simulation configuration [10]. ns (via tclcl) provides glue to make objects and variables appear on both languages. The tcl interface can be used in cases where small changes in the scenarios are easily implemented. Similarly, the C++ code can be changes when processing of all incoming packets are done, or when changes in the behavior of the protocol is anticipated.

In ns, the advance of time depends on the timing of events which are maintained by a scheduler [12]. An event is an object in the C++ hierarchy with an unique ID, a scheduled time and a pointer to an object that handles the event. A scheduler keeps an ordered data structure with the events to be executed and fires them side by side, invoking the handler of the event [8]-[10].

In propose scheme we used OTcl for message authentication, source privacy and When providing encryption and decryption by using RC6 then C++ language is used.

III. PROPOSED APPROACH

Our proposed authentication scheme aims at achieving the following goals:

- **Message authentication:** The message receiver should be able to verify whether a received message is sent by the node that is claimed or by a node in a particular group. In other words, the adversaries cannot pretend to be an innocent node and inject fake messages into the network without being detected.
- **Hop-by-hop message authentication**: Every forwarder on the routing path should be able to verify the authenticity and integrity of the messages upon reception [1].
- **Identity and location privacy**: The adversaries cannot determine the message sender's ID and location by analyzing the message contents or the local traffic [3].
- **Efficiency**: The scheme should be efficient in terms of both computational and communication overhead [15].

A Rivest Cipher 6

RC6 is a block cipher based. RC6 is a parameterized algorithm where the block size, the key size, and the number of rounds are variable.[12] The upper limit on the key size is 2040 bits. RC6 adds two features to RC5:-First the inclusion of integer multiplication. Second is the use of four 4-bit working registers instead of RC5's two 2-bit registers [4].

RC6 is a completely parameterized family of encryption algorithms system. A version of RC6 is more precisely specified as RC6-w/r/b where the word size is w bits, encryption has nonnegative number of rounds r and b denoting the length of the encryption key in bytes [10]. Since the AES submission is aimed at w = 32 and r = 20, it can use RC6 as shorthand to consider to such versions. When any other value of w or r is intended in the text, the parameter values will be specified as RC6-w/r [7]. Of meticulous relevance to the AES attempt will be the versions of RC6 with 16-, 24- and 32-byte keys. For all variants, RC6-w/r/b works on units of four w-bit words using the following fundamental operations [2].

The operations used in RC6 are given fundamental operation:

- A+B= integer addition modulo 2w.
- A-B= integer subtraction modulo 2w.
- A⊕B= bitwise exclusive-or of w-bit words size.
- A*B= integer multiplication modulo 2w.
- A<<<B= rotation of the w-bit word A to the left by the amount given by, the least significant lg w bits of B.
- A>>>B = rotation of the w-bit word A to the right by the amount given by, the least significant lg w bits of B f(x) = x(2x+1)mod2w.

A.1 Encryption with RC6-

Input: Plaintext stored in four w-bit input registers A, B, C, D
Number r of rounds
w-bit round keys $S[0, \ldots, 2r+3]$

Output: Ciphertext stored in A, B, C, D

Procedure:

$B = B + S[0]$
$D = D + S[1]$
for $i = 1$ **to** r **do**
{
$t = (B \times (2B + 1)) \lll \lg w$
$u = (D \times (2D + 1)) \lll \lg w$
$A = ((A \oplus t) \lll u) + S[2i]$
$C = ((C \oplus u) \lll t) + S[2i + 1]$
$(A, B, C, D) = (B, C, D, A)$
}
$A = A + S[2r + 2]$
$C = C + S[2r + 3]$

A.2 Decryption with RC6-

Input: Ciphertext stored in four w-bit input registers A, B, C, D
Number r of rounds
w-bit round keys $S[0, \ldots, 2r+3]$

Output: Plaintext stored in A, B, C, D

Procedure:
$C = C - S[2r+3]$
$A = A - S[2r+2]$
for $i = r$ **downto** 1 **do**
{
$(A, B, C, D) = (D, A, B, C)$
$u = (D \times (2D+1)) \lll \lg w$
$t = (B \times (2B+1)) \lll \lg w$
$C = ((C - S[2i+1]) \ggg t) \oplus u$
$A = ((A - S[2i]) \ggg u) \oplus t$
}
$D = D - S[1]$
$B = B - S[0]$

IV. COMPARE VARIOUS FEATURE BY USING GRAPHICAL FORMATE

A. Delay combime graph

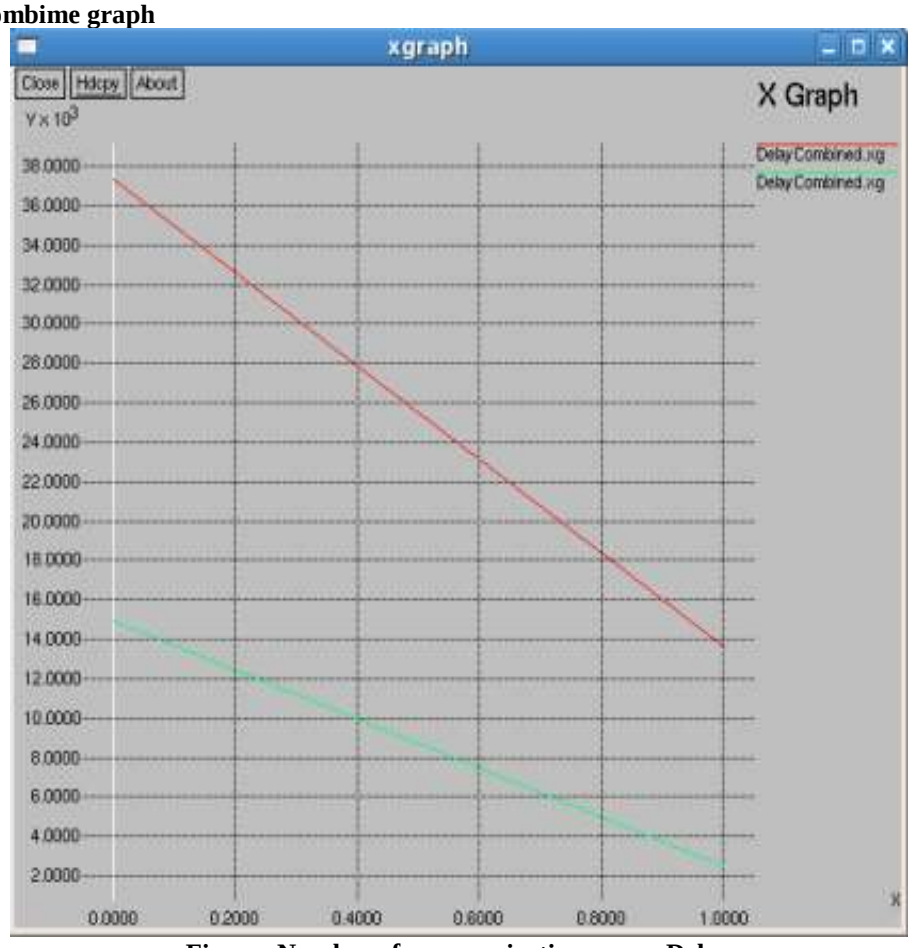

Figure: Number of communication verses Delay

B. Packetlosscombine graph

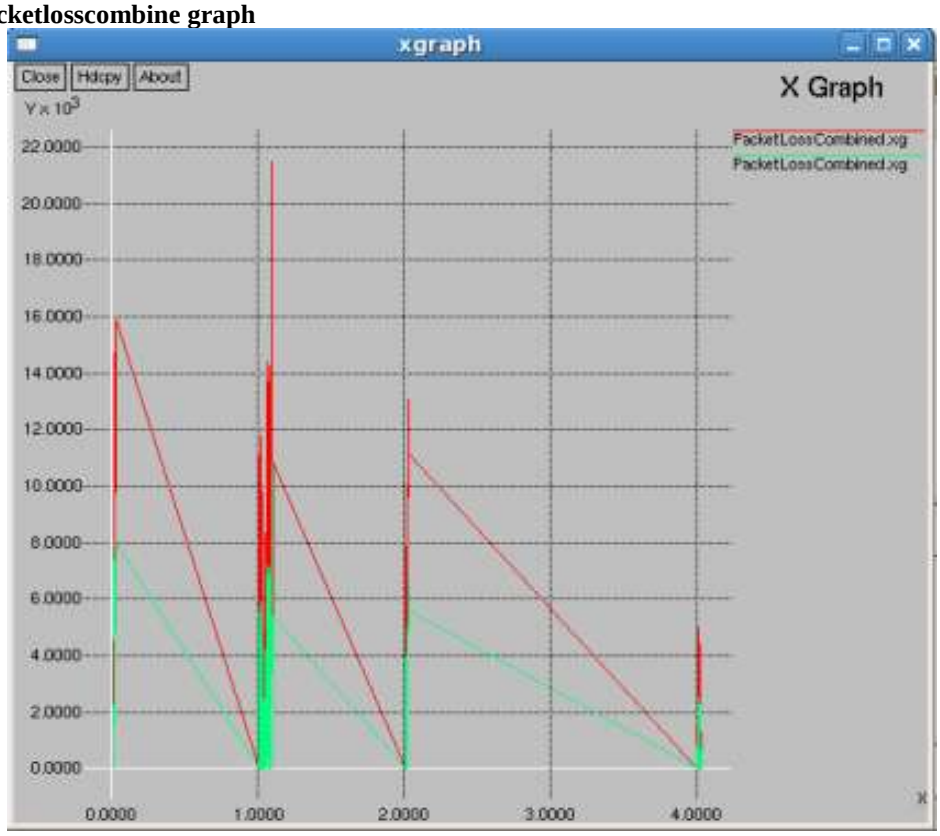

Figure: Simulation time verses Packet loss

C. Network loss combine graph

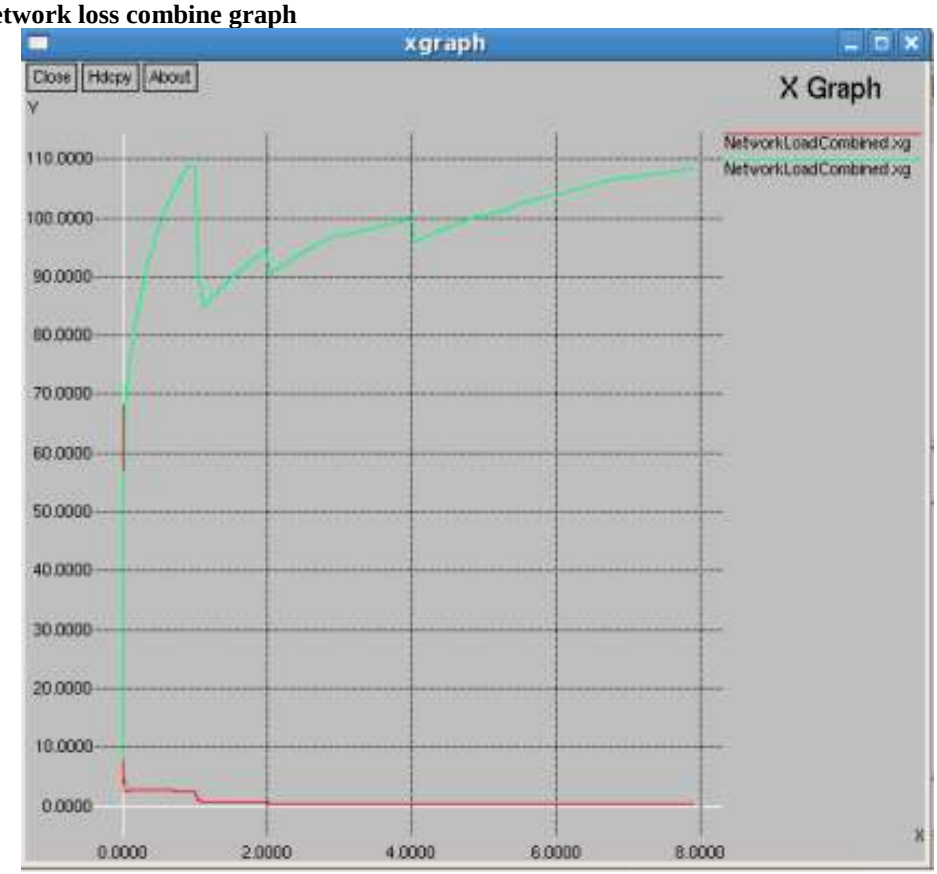

Figure: Simulation time verses Network loss

D. Cost combine graph

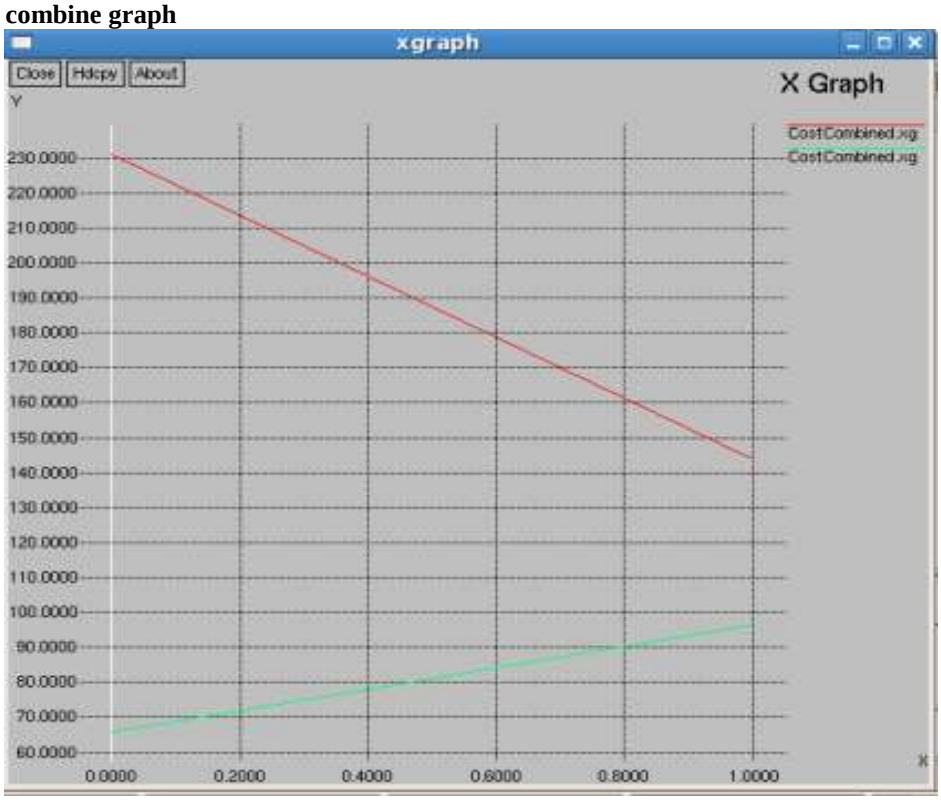

Figure: Number of communication verses Cost

ACKNOWLEDGMENT

To provide hop-by-hop message authentication, the weakness of the built in threshold of the polynomial-based scheme. We propose a hop-by-hop message authentication scheme based on the RC6. Both theoretical and simulation results show that, in comparable scenarios, our proposed scheme is more efficient than the bivariate polynomial-based scheme in terms of computational overhead, energy consumption, message delay and memory consumption.

REFERENCES

[1]. Jian Li Yun Li Jian Ren Jie Wu, ―Hop-by-Hop Message Authentication and Source Privacy in Wireless Sensor Networks‖, IEEE Transactions On Parallel And Distributed Systems, pp 1-10, 2013

[2]. Sadaqat Ur Rehman, Muhammad Bilal, Basharat Ahmad, Khawaja Muhammad Yahya, Anees Ullah, Obaid Ur Rehman, ―Comparison Based Analysis of Different Cryptographic and Encryption Techniques Using Message Authentication Code (MAC) in Wireless Sensor Networks (WSN)‖, IJCSI International Journal of Computer Science Issues, Vol. 9, Issue 1, No 2, January 2012, pp 96-101

[3]. Harsh Kumar Verma, Ravindra Kumar Singh, ―Performance Analysis of RC6, Twofish and Rijndael Block Cipher Algorithms‖, International Journal of Computer Applications (0975 – 8887) Volume 42– No.16, March 2012, pp 1-7

[4]. M. Albrecht, C. Gentry, S. Halevi, and J. Katz, ―Attacking cryptographic schemes based on ‖perturbation polynomials‖,‖ Cryptology ePrint Archive, Report 2009/098, 2009, http://eprint.iacr.org.

[5]. Dunfan Ye,DaoliGong,WeiWang ―Application of Wreless Sensor Networks in Environmental Monitoring‖, 2009 2nd International Conference on Power Electronics and Intelligent Transportation System.

[6]. Ling Tan, Shunyi Zhang, and Yanfeng Sun, Jing Qi ―Application of Wireless Sensor Networks in Energy Automation‖, Sustainable Power Generation and Supply, 2009. Supergen ’09. International conference

[7]. H. Wang, S. Sheng, C. Tan, and Q. Li, ―Comparing symmetric-key and public-key based security schemes in sensor networks: A case study of user access control,‖ in IEEE ICDCS, Beijing, China, 2008, pp. 11–18.

[8]. Ian F. Akylidiz, Fellow IEEE, Tommaso Melodia, Member IEEE, and Kaushik R. Chowdhury, Student Member IEEE ―Wireless Multimedia Sensor Networks: Applications and Testbeds‖, Proceedings of the IEEE. Vol. 96, No. 10, October 2008

[9]. W. Zhang, N. Subramanian, and G. Wang, ―Lightweight and compromise resilient message authentication in sensor networks,‖ in IEEE INFOCOM, Phoenix, AZ., April 15-17 2008.
[10]. Raymond Sbrusch, ―Authenticated Messaging In Wireless Sensor Networks Used For Surveillance‖, Thesis, The University Of Houston-Clear Lake, May, 2008
[11]. Chung-Kuo Chang, J. Marc Overhage, Jeffrey Huang ―An Application of Sensor Networks for Syndromic Surveillance‖ 2005 IEEE
[12]. F. Ye, H. Lou, S. Lu, and L. Zhang, ―Statistical en-route filtering of injected false data in sensor networks,‖ in IEEE INFOCOM, March 2004.
[13]. S. Zhu, S. Setia, S. Jajodia, and P. Ning, ―An interleaved hop-by-hop authentication scheme for filtering false data in sensor networks,‖ in IEEE Symposium on Security and Privacy, 2004.
[14]. A. Perrig, R. Canetti, J. Tygar, and D. Song, ―Efficient authentication and signing of multicast streams over lossy channels,‖ in IEEE Symposium on Security and Privacy, May 2000.
[15]. C. Blundo, A. De Santis, A. Herzberg, S. Kutten, U. Vaccaro, and M. Yung, ―Perfectly-secure key distribution for dynamic conferences,‖ in Advances in Cryptology - Crypto'92, ser. Lecture Notes in Computer Science Volume 740, 1992, pp. 471–486.
[16]. T. A. ElGamal, ―A public-key cryptosystem and a signature scheme based on discrete logarithms,‖ IEEE Transactions on Information Theory, vol. 31, no. 4, pp. 469–472, 1985.
[17]. R. Rivest, A. Shamir, and L. Adleman, ―A method for obtaining digital signatures and public-key cryptosystems, ‖ Communications. of the Assoc. of Comp. Mach., vol. 21, no. 2, pp. 120–126, 1978.