www.ioirp.com
International Journal of Innovative Research in Computer Science and Engineering (IJIRCSE)
ISSN: 2394-6364, Volume – 1, Issue – 6. May 2015

Survey on Message Authentication and Source Privacy in Wireless Sensor Networks

[1]Priyanka Jaikumar Upadhye, [2]Rakesh S
Department of Computer Science Engineering. Akshya Institute of Technology Tumkur, India
Computer Network and Engineering., Akshya Institute of Technology, Tumkur, India
[1]priyanka.pinku20@gmail.com

***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, however, 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 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 investigates different cryptographic approaches such as symmetric key cryptography and asymmetric key cryptography.**

***Keywords*— Message Encryption;WSN**

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. Therefore, many authentication schemes have been proposed in literature to offer message authenticity and integrity verification for wireless sensor networks (WSNs). These approaches can largely be separated into two categories: public-key based approaches and symmetric-key based approaches. The symmetric-key based approach necessitates composite key management, lacks of scalability, and is 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. 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. 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. Every intermediate forwarder and the final receiver can authenticate the message using the sender's public key [1]. One of the restrictions of the public key based method is the high computational overhead.

II. INSIDE VIEW ON WIRELESS SENSOR NETWORKS

Wireless sensor networks simplify the compilation and scrutiny of information from multiple locations [2]. 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. 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. This section recognizes the exclusive challenges of wireless sensor networks.

A. Characteristics of Wireless Sensor Networks: -

Wireless sensor networks are deployed for a varied diversity of applications, each characterized by an exclusive set of requirements. While the classical sensor network made up of homogeneous devices, contemporary sensor networks fit in modular design and make use of heterogeneous nodes that accomplish unique requirements. For example, some nodes contain a GPS sensor that other nodes can query to decide their location. Others may contain interfaces to the Internet through satellite or cellular communications. While radio frequency is the most general communication modality, data can also be transmitted via laser, sound, and diffuse light. These communication means carry an assortment of network infrastructures.

In a fundamental infrastructure-organized network, nodes can only converse with a base station. The reverse is true in an ad-hoc network where there is no base station or communication infrastructure. In this case, each node can converse with any other node. The communication infrastructure manipulates network topology. In some cases, each node must be inside radio range of any other node because messages can only voyage across a single hop. Networks planned into a graph-like topology permit routing of messages across multiple hops. Some applications can achieve their goals with a network of sparsely deployed sensors. Others require a densely populated network with redundant nodes accessible. Network topology and coverage requirements decide the network size. Networks may range in size from thousands of nodes to only a few.

B. Security in WSN: -

Security risks in wireless sensor networks contain threats to the confidentiality, integrity, and 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. The adoption of competent algorithms to alleviate security risks has not kept pace with the rate of miniaturization. This section underscores the challenges of securing sensor network communications and demonstrates general attacks against sensor networks.

- *Security Goals:* - Security assessments of any application spotlight on the five fundamental tenets of data security: confidentiality, origin integrity, data integrity, non-repudiation, and availability. The definitions used in this subsection are derived from [3]. Confidentiality means the camouflage of information from unauthorized entities. Mechanisms used to accomplish confidentiality include access control mechanisms and cryptography. Cryptography scrambles, or encrypts, information to produce cipher text inarticulate to any unauthorized viewer. The data can be made understandable to an authorized viewer who knows the secret key. Semantic security entails a stronger assurance of confidentiality. Semantic security needs that repeated encryption of a message M would yield unique cipher text each round. This confines the ability of an eavesdropper to understand the plaintext even after observing numerous encryptions of the identical message. Use of initialization vectors (IVs) seeded with a counter or a non-repeating nonce gives semantic security.

Origin integrity, also recognized as authentication, refers to the trustworthiness of the source of information. It means that the receiver of a message can trust that the sender of the message is candidly who it claims. An intruder should be unable to propel a fabricated message and have it treated as a legitimate message from a trusted peer. Data integrity means that the user of the information can trust that the content of the information has not been altered in any way by an unauthorized intruder or improperly customized by an authorized user. Since a like mechanisms present origin integrity and data integrity, they are usually grouped under the moniker ―integrity. Integrity outshines other security goals because of its influence on the reliability of the system and its output. In a robust wireless sensor network, the data contained in a message grips a lower priority than the integrity and authenticity of the message. Non-repudiation means that the sender of a message should not be able to reject later that he ever sent that message. In the pre-digital scenario, one achieved non-repudiation with a simple hand-written signature. In cryptography, it implies that authentication and data integrity can be certified with a high level of guarantee and it cannot later be refuted. Non-repudiation is a serious security service and must be guaranteed in applications that engage financial and business transactions, where accountability of events is significant to guarantee success of the applications. Digital signatures offer non-repudiation. Availability implies that an authorized user should be able to employ the data or resource as required. In a wireless sensor network, the wireless communication link must remain obtainable for the network to sustain operations.

- *Challenges:* - The lack of proficient authenticated messaging exposes all layers of the sensor network protocol stack to potential compromise. Without link-layer authentication, an attacker may insert unauthorized packets into the network. This may be used to introduce collisions and force legitimate nodes into an infinite waiting state [4]. Network layer attacks against routing protocols give the attacker the ability to cause routing loops, delay messages, or selectively drop messages [5]. Wireless sensor networks deployed for tracking targets provide valuable application layer notifications about the location of the target. Without authentication, the attacker can perpetrate attacks such as dropping intruder notifications, spoofing intruder notifications to create a diversion, or forcing the entire network into a continual state of reorganization.

In wireless sensor networks, the need for integrity surpasses all other security goals. Data integrity and authentication create a foundation for a highly available and trustworthy network. While many authentication schemes have been conceived for wireless sensor networks, none of them is a panacea. Algorithms for unicast message authentication, for example, do not meet the

requirements for authenticating broadcast messages. Similarly, algorithms that mimic the asymmetry of public key systems by dividing time into slots violate the real-time constraints of intrusion notification systems.

- *Attacks against Sensor Networks:-*

Physical tampering poses a threat to sensors. If sensors are distributed in an unprotected area, an attacker could destroy the nodes or collect the sensors, analyze the electronics, and steal cryptographic keys. This complicates the process of bootstrapping newly deployed sensors with cryptographic keying material. To protect against this, sensors must be tamper-proof or they must erase all permanent and temporary storage when compromised. Secure key rotation mechanisms can also mitigate the threat of stolen cryptographic keys. Jamming attacks against wireless radio frequencies affect the availability of the network. While it is most efficient to program sensors to communicate on one specific wireless frequency, an attacker could easily broadcast a more powerful signal on the same frequency and introduce interference into the communications channel. Spread spectrum technologies such as frequency-hopping spread spectrum alleviate the impact of jamming; however, complex channel hopping patterns reduce battery life. Nodes could also try to detect jamming and sleep until the jamming stops, resulting in a temporary, self-induced denial of service (DoS). Link layer protocols face similarly challenging threats. Attackers can introduce collisions that force communicating nodes to retransmit frames. Following a collision, a node must back-off and wait for the channel to clear before attempting to resend. The attacker can continually introduce collisions until the victim runs out of power. While error-detecting mechanisms suffice for common transmission errors, they do not reduce the influence of maliciously generated collisions. Collisions maliciously injected near the end of a legitimate frame rapidly exhaust the resources of the legitimate node. Authentication cannot alleviate these physical and link layer attacks.

Network layer attacks take advantage of the ad-hoc organization of wireless sensor networks. Any node in the network can become a router, forwarding traffic from one node to another. By manipulating routing information, the attacker can shape the flow of traffic. The simplest attack compromises a routing node and forces it to drop messages, creating a network ―black hole‖. The attacker can also selectively delay messages routed by the compromised node. In a wormhole attack, the adversary tunnels messages destined for one part of the network through a path under enemy control. Wormhole attacks facilitates eavesdropping, message replay, or disconnection of a segment of the network. One technique to create black holes circumvents the way routing protocols organize the network. Nodes typically accept the router that broadcasts route advertisements with the strongest radio signal. This policy reduces the energy required for a node to converse with its default router. An attacker can influence this strategy to convince legitimate nodes that it necessitates the least communication overhead. Internet style attacks have their analogue in wireless sensor networks. Misdirection attacks, such as the Internet smurf attack, work in sensor networks. The attacker can propel multiple messages to broadcast addresses with a source address forged to the intended victim's address. The broadcast retorts will overwhelm the victim, flood its communication channel, and exhaust its power. Filtering the legitimate messages from the responses in a smurf attack needs a hierarchy not present in many wireless sensor network routing protocols.'

A alike attack, called a Sybil attack, objects systems that choose peers based on their reputation. In a Sybil attack, the adversary sends a large number of fabricated messages that emerge to be forwarded from other nodes. Legitimate nodes commence to trust the attacker because it seems to fairly route traffic. The legitimate nodes will eventually accept the adversarial node as their router.

Transport-layer protocols present end-to-end connectivity between nodes. Sequencing, such as that done in the Transmission Control Protocol (TCP), enhances the reliability of the connection. Protocols that apply sequencing may yield to Denial of Service (DoS) attacks. The classic TCP SYN flood concerns to sensor networks. An adversary can flood the victim with synchronization requests and bound the ability for other nodes to converse with the victim. One solution limits the number of synchronization needs accepted, but this limits both adversaries and allies. Client riddles, a more complex solution, require the client to construct a commitment to the server before it is allowed to begin a conversation. When the client opens a connection, the server will reply with a puzzle that the client must crack. The client must solve the puzzle and propel the answer to the server before the server will recognize a full connection. While these solutions defend the server from SYN floods, it may damage allies that have fewer computational resources than the adversary does. Origin authentication and message integrity can alleviate attacks at the network layer and above. Threats such as spoofing or fabrication of routing data validate the need for origin and data integrity of even the simplest HI

III. Review of Message Authentication Protocols

This section summarizes some of the most relevant proposals that incorporate origin integrity and data integrity in to wireless sensor network communications. Each proposal possesses exclusive qualities that persuade its applicability. Many merge schemes for origin integrity and message integrity with other security goals, such as confidentiality or replay protection. However, these features may use excessive processor, storage, or energy resources. An authentication protocol should be defiant to node compromise by permitting secure key management. The protocol may offer an integrated key-rotation mechanism or permit for key rotation by an external module. In addition, the protocol must have small computation overhead for both the sender and the recipient of a message. The protocol must also necessitate low communication overhead. Finally, messages supporting the authentication protocol must purpose in an unreliable network. Thus, the protocol should support the ability to immediately authenticate a message upon receipt.

A. View on Conventional Authentication: -

The roots of message integrity commence with cryptographic checksums, also known as hashes. These checksum functions acquire a message and compact it into a smaller message digest. The simplest example, the parity bit, calculates the number of 1-bits in a message to create a checksum of 1-bit in length. Strong cryptographic hash functions must own three desirable properties. First, the hash must be easy to calculate, not consuming major computational resources. Second, it should be computationally not feasible to reverse the hash function. This means that known the result of the hash *h (M)*, one should not be capable to decide *M*. A third advantageous property of hashing algorithms says that two distinct messages, when hashed, will acquiesce two distinct checksums. However, as per the pigeonhole principle, there is a possibility that two distinct messages *M* and *M'*, will acquiesce generate the same hash value, *h (M) = h(M')*. This condition, known as a collision, can be subjugated to overcome hash functions]. The MD5 and SHA-1 hash functions are engaged in several security applications and protocols. MD5 abbreviates a message into a hash of 16 bytes. SHA-1 abbreviates a message into a 20-byte hash. Both MD5 and SHA-1 have been established susceptible to collisions.

Hash functions give a level of message integrity between communicating peers. A sender organizes a message *M* and computes the checksum *x = h (M).* It then propels the checksum along with the message to the recipient. When the recipient obtains message *M*, he can recomputed the checksum on the received message *M*. If the checksum added to the message matches the checksum calculated by the recipient, then the recipient can be assured of message integrity. Cryptographic checksums cannot give assurance that messages reach without modification or that they initiate from an authentic sender. Since an attacker may recognize the hashing algorithm in use, an attacker could just restore message *M* with message *M'*, calculate the hash *x' = h(M')*, and send the concatenation of the message *M'* and the hash *x'*. The recipient will compute the hash of *M'*, which will match the *x'* sent by the attacker. Thus, the recipient cannot authenticate that authenticity of the message. Message authentication codes (MAC), an instantiation of hashes that applies a unique key, give both the data integrity of checksums and origin integrity provided by a secret key. Both the sender and receiver should share the key. If an adversary finds out the secret key, the hashing function is compromised. A MAC is generated by encrypting a message with a block cipher in Cipher Block Chaining (CBC) or Cipher Feedback Modes (CFB) [14]. Use of the Cipher Block Chaining mode to create a MAC is commonly known as CBCMAC. Several WSN authentication mechanisms utilize CBC-MAC. However, the CBCMAC operation has been shown to be apprehensive for variable length messages.

B. Unicast vs. Broadcast Authentication: -

Unicast authentication gives the assertion of origin integrity when a message is delivered from one sender to one receiver. A message authentication code (MAC), created by the sender/creator of the message by using a secret key, can be used to guarantee origin integrity. For unicast messages, static symmetric (shared) key cryptography gets the requirements because the two peers are trusted not to disclose the key. The speed and effectiveness of symmetric key cryptography suit the constraints of wireless motes.

Broadcast authentication guarantees that multiple recipients of a message can authenticate its origin integrity. If using MACs to make sure broadcast authentication, all recipients of the message must share the symmetric key. The exclusive challenge for broadcast authentication engages the management of that shared key. If the key is broadcast to probable recipients,

an adversary could eavesdrop on the key broadcast, detain the key, and produce a legitimate MAC for a forged message. Public key cryptography explains the problem of securely sharing a key for conventional Internet computing systems. However, public key cryptosystems use far too a lot of storage, computation.

C. Block Ciphers: -

Symmetric key cryptography have two categories of ciphers: block ciphers and stream ciphers. Stream ciphers work on a single bit or byte at a time. Block ciphers function on groups of bits called blocks [37]. Common block ciphers considered for wireless sensor networks admit block sizes of 32, 64, and 128 bits. Authentication mechanisms typically utilize block ciphers because they can be used to create MAC.

Table 1 summarizes the block and key sizes of common block ciphers.

Ciphers	Key Size(b)	Block Size(b)
AES	128/192/256	128
RC5	0 ~ 2040	32/64/128
RC6	128/192/256	128
Twofish	128/192/256	128
Skipjack	80	64
XTEA	128	64

Cipher Key Size (b) Block Size (b)

Symmetric key encryption is frequent to ensure data confidentiality, it utilizes shared key for both encryption of plain text and decryption of cipher text. In cryptography, the Advanced Encryption Standard (AES) is an encryption standard adopted by the U.S. government. A combination of factors such as security, performance, efficiency, easiness of implementation and flexibility contributed to the assortment of this algorithm as the AES.

IV. PRIOR STUDY WORK

Ye et al. [6] presented a Statistical En-route Filtering (SEF) mechanism that can detect and drop such false reports. SEF requires that each sensing report be validated by multiple keyed message authentication codes (MACs), each generated by a node that detects the same event. As the report is forwarded, each node along the way verifies the correctness of the MACs probabilistically and drops those with invalid MACs at earliest points.

Zhang et al. [7] proposed in the past for protecting communication authenticity and integrity in wireless sensor networks. Most of them however have following limitations: high computation or communication overhead, no resilience to a large number of node compromises, delayed authentication, lack of scalability, etc.

Perrig et al. [8] demonstrated two efficient schemes, TESLA and EMSS, for secure lossy multicast streams. TESLA, short for Timed Efficient Stream Loss-tolerant Authentication, offers sender authentication, strong loss robustness, high scalability, and minimal overhead, at the cost of loose initial time synchronization and slightly delayed authentication. EMSS, short for Efficient Multi-chained Stream Signature, provides no repudiation of origin, high loss resistance, and low overhead, at the cost of slightly delayed verification.

Albrecht et al. [9] demonstrated attacks on several cryptographic schemes that have recently been proposed for achieving various security goals in sensor networks. Roughly speaking, these schemes all use “perturbation polynomials” to add “noise” to polynomial-based systems that offer information theoretic security, in an attempt to increase the resilience threshold while maintaining efficiency.

Rivest et al. [10] presented with the novel property that publicly revealing an encryption key does not thereby reveal the corresponding decryption key. This has two important consequences: Couriers or other secure means are not needed to transmit

keys, since a message can be enciphered using an encryption key publicly revealed by the intended recipient. Only he can decipher the message, since only he knows the corresponding decryption key.

Wang et al. [11] introduced large memory and communication overhead. On the contrary, public key based schemes have simple and clean key management, but cost more computational time. The recent progress of elliptic curve cryptography (ECC) implementation on sensors motivates us to design a public-key scheme and compare its performance with the symmetric-key counterparts.

Pointcheval and Stren [12] addressed the question of providing security proofs for signature schemes in the so-called random oracle model. In particular, they establish the generality of this technique against adaptively chosen message attacks.

V. CONCLUSION

This paper discusses an overview on message authentication in wireless sensor networks. Message authentication performs a key role in thwarting unauthorized and corrupted messages from being forwarded in networks it investigates that public key is not energy efficient and is costly in terms of both computation and communication as compared to symmetric key. Sensor networks have limited resources, therefore most of the researcher considered symmetric key to create MAC in WSNs.

References

[1] 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.

[2] Raymond Sbrusch, ―Authenticated Messaging In Wireless Sensor Networks Used For Surveillance‖, Thesis, The University Of Houston-Clear Lake, May, 2008

[3] Bishop, M., Computer security: art and science. Boston, MA: Addison-Wesley, 2003

[4] Wood, A. D. and Stankovic, J. A., "Denial of service in sensor networks," IEEE Computer, vol. 35, pp. 54-62, 2002.

[5] Wood, A. D., Fang, L., Stankovic, J. A., and He, T., "SIGF: a family of configurable, secure routing protocols for wireless sensor networks," Proceedings of the fourth ACM workshop on Security of ad hoc and sensor networks, pp. 35-48, 2006

[6] F. Ye, H. Lou, S. Lu, and L. Zhang, "Statistical En-Route Filtering of Injected False Data in Sensor Networks," Proc. IEEE INFOCOM, Mar. 2004

[7] W. Zhang, N. Subramanian, and G. Wang, "Lightweight and Compromise-Resilient Message Authentication in Sensor Networks," Proc. IEEE INFOCOM, Apr. 2008

[8] A. Perrig, R. Canetti, J. Tygar, and D. Song, "Efficient Authentication and Signing of Multicast Streams over Lossy Channels," Proc. IEEE Symp. Security and Privacy, May 2000.

[9] M. Albrecht, C. Gentry, S. Halevi, and J. Katz, "Attacking Cryptographic Schemes Based on 'Perturbation Polynomials'," Report 2009/098, http://eprint.iacr.org/, 2009

[10] R. Rivest, A. Shamir, and L. Adleman, "A Method for Obtaining Digital Signatures and Public-Key Cryptosystems," Comm. ACM, vol. 21, no. 2, pp. 120-126, 1978.

[11] 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," Proc. IEEE 28th Int'l Conf. Distributed Computing Systems (ICDCS), pp. 11-18, 2008.

[12] D. Pointcheval and J. Stern, "Security Proofs for Signature Schemes," Proc. Advances in Cryptology (EUROCRYPT), pp. 387- 398, 1996.