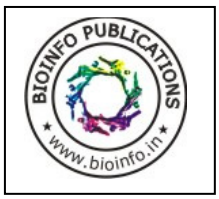
Journal of Information Systems and Communication, ISSN: 0976-8742, Volume 1, Issue 2, 2010, pp-01-06

Security issues in wireless sensor networks



Snehlata Yadav*[1], Kamlesh Gupta[2] and Sanjay Silakari[3]
*[1]Govt. Women's Polytechnic College, Bhopal MP, India, sneharshi2004@yahoo.co.in
[2]Jaypee University of Engineering and Technology, Guna, MP, India, kamlesh_rjitbsf@yahoo.com
[3]UIT, RGPV Bhopal, MP, India, ssilakari@yahoo.com

Abstract- Wireless sensor networks are a new type of networked systems, characterized by every constrained computational and energy resources, and an ad hoc operational environment Network security to Wireless Sensor Networks is a very essential requirement because they are easily susceptible to many threats like Denial-of-Service attacks [15]. The most important security services required are confidentiality and authentication. Many researchers have tried to provide security by using only symmetric key mechanisms thinking that public key cryptosystems are not feasible to implement in these networks because they are constrained with less resources. This paper studies the security aspects of these networks. The paper first introduces sensor networks, and then presents its related security problems, threats, risks and characteristics.

Introduction

Sensor networks refer to a heterogeneous system combining tiny sensors and actuators with general-purpose computing elements. These networks will consist of hundreds or thousands of self-organizing, low-power, low cost wireless nodes deployed en masse to monitor and affect the environment .Potential applications include burglar alarms, inventory control, medical monitoring and emergency response [11], monitoring remote or inhospitable habitats [9, 10], target tracking in battlefields [12],disaster relief networks, early fire detection in forests, and environmental monitoring. Sensor networks are typically characterized by limited power supplies, low bandwidth, small memory sizes and limited energy. This leads to a very demanding environment to provide security. Public-key cryptography is too expensive to be usable, and even fast symmetric-key ciphers must be used sparingly. Communication bandwidth is extremely dear: each bit transmitted consumes about as much power as executing 800–1000 instructions [13], and as a consequence, any message expansion caused by security mechanisms comes at significant cost. In [5], the authors point out that it seems unlikely that Moore's law will help in the foreseeable future. Because one of the most important factors determining the value of a sensor network comes from how many sensors can be deployed, it seems likely there will be strong pressure to develop ever-cheaper sensor nodes. In other words, we expect that users will want to ride the Moore's law curve down towards evercheaper systems at a fixed performance point, rather than holding price constant and improving performance over time. Thus, the resource-starved nature of sensor networks poses great challenges for security. However, in many applications the security aspects are as important as performance and low energy consumption. Besides the battlefield applications, security is critical in premise security and surveillance, building monitoring, burglar alarms, and in sensors in critical systems such as airports, hospitals.

Sensor Network Architecture

Sensor networks often have one or more points of centralized control called base stations. A base station is typically a gateway to another network, a powerful data processing or storage center, or an access point for human interface. They can be used as a nexus to disseminate control information into the network or extract data from it. Base stations have also been referred to as sinks. The sensor nodes establish a routing forest, with a base station at the root of every tree. Base stations are many orders of magnitude more powerful than sensor nodes. Typically, base stations have enough battery power to surpass the lifetime of all sensor nodes, sufficient memory to store cryptographic keys, stronger processors, and means for communicating with outside networks. Communication Architecture Generally, the sensor nodes communicate using RF, so broadcast is the fundamental communication primitive. The baseline protocols account for this property: on one hand it affects the trust assumptions, and on the other it is exploited to minimize the energy usage. In the sensor applications developed so far, the communication patterns within the network fall into the following categories: ·

Node to base station communication, e.g.sensor readings, specific alerts.· Base station to node communication, e.g.specific requests, key updations .· Base station to all nodes, e.g. routingbeacons, queries or reprogramming of the entire network. Communication amongst a defined cluster of nodes (say, a node and all its neighbors). Clustering can reduce the total number of messages sent and thus save energy

[14, 15, 16] by using in-network processing techniques such as data aggregation [24, 25] (an aggregation point can collect sensor readings from surrounding nodes and forward a single message representing an aggregate of the values) and passive participation (a node that overhears a neighboring sensor node transmitting the same reading as its own current reading can elect to not transmit the same).

Security Issues and Goals

1. Data Confidentiality

Confidentiality means keeping information secret from unauthorized parties. A sensornetwork should not leak sensor readings to neighboring networks. In many applications (e.g. key distribution) nodes communicate highly sensitive data. The standard approach for keeping sensitive data secret is to encrypt the data with a secret key that only intended receivers possess, hence achieving confidentiality. Since public-key cryptography is too expensive to be used in the resource constrained sensor networks, most of the proposed protocols use symmetric key encryption methods. The creators of TinySec [7] argue that cipher block chaining (CBC) is the most appropriate encryption scheme for sensor networks. They found RC5 and Skipjack to be most appropriate for software implementation on embedded microcontrollers. The default block cipher in TinySec is Skipjack. SPINS uses RC6 as its cipher.

2. Data Authenticity

In a sensor network, an adversary can easily inject messages, so the receiver needs to make sure that the data used in any decision making process originates from the correct source. Data authentication prevents unauthorized parties from participating in the network and legitimate nodes should be able to detect messages from unauthorized nodes and reject them. In the two-party communication case, data authentication can be achieved through a purely symmetric mechanism: The sender and the receiver share a secret key to compute a message authentication code (MAC) of all communicated data. When a message with a correct MAC arrives, the receiver knows that it must have been sent by the sender. However, authentication for broadcast messages requires stronger trust assumptions on the network nodes. The creators of SPINS [1] contend that if one sender wants to send authentic data to mutually untrusted receivers, using a symmetric MAC is insecure since any one of the receivers know the MAC key, and hence could impersonate the sender and forge messages to other receivers. SPINS constructs authenticated broadcast from symmetric primitives, but introduces asymmetry with delayed key disclosure and one-way function key chains. LEAP [8] uses a globally shared symmetric key for broadcast messages to the whole group. However, since the group key is shared among all the nodes in the network, an efficient rekeying mechanism is defined for updating this key after a compromised node is revoked. This means that LEAP has also defined an efficient mechanism to verify whether a node has been compromised.

3. Data Integrity

Data integrity ensures the receiver that the received data is not altered in transit by an adversary. Note that Data Authentication can provide Data Integrity also.

4. Data Freshness

Data freshness implies that the data is recent, and it ensures that an adversary has not replayed old messages. A common defense (used by SNEP [1]) is to include a monotonically increasing counter with every message and reject messages with old counter values. With this policy, every recipient must maintain a table of the last value from every sender it receives. However, for RAM constrained sensor nodes, this defense becomes problematic for even modestly sized networks. Assuming nodes devote only a small fraction of their RAM for this neighbor table, an adversary replaying broadcast messages from many different senders can fill up the table. At this point, the recipient has one of two options: ignore any messages from senders not in its neighbor table, or purge entries from the table. Neither is acceptable; the first creates a DoS attack and the second permits replay attacks. In [5], the authors contend that protection against the replay of data packets should be provided at the application layer and not by a secure routing protocol as only the application can fully and accurately detect the replay of data packets (as opposed to retransmissions ,for example). In [7], the authors reason that by using information about the network's topology and communication patterns, the application and routing layers can properly and efficiently manage a limited amount of memory devoted to replay detection. In [1], the authors have identified two types of freshness: weak freshness, which provides partial message ordering, but carries no delay information, and strong freshness, which provides a total order on a request response pair, and allows for delay estimation. Weak freshness is required by sensor measurements, while strong freshness is useful for time synchronization within the network.

5. Robustness and Survivability

The sensor network should be robust against various security attacks, and if an attack succeeds, its impact should be minimized. The

compromise of a single node should not break the security of the entire network.

Security Threats, Types of Attacks on Sensor Networks and Countermeasures

Wireless networks are vulnerable to security attacks due to the broadcast nature of the transmission medium. Furthermore, wireless sensor networks have an additional vulnerability because nodes are often placed in a hostile or dangerous environment where they are not physically protected.

1. Passive Information Gathering

An intruder with an appropriately powerful receiver and well designed antenna can easily pick off the data stream. Interception of the messages containing the physical locations of sensor nodes allows an attacker to locate the nodes and destroy them. Besides the locations of sensor nodes, an adversary can observe the application specific content of messages including message IDs, timestamps and other fields. To minimize the threats of passive information gathering, strong encryption techniques needs to be used.

2. Subversion of a Node

A particular sensor might be captured, and information stored on it (such as the key) might be obtained by an adversary. If a node has been compromised then how to exclude that node, and that node only, from the sensor network is at issue (LEAP [8] defines an efficient way to do so).

3. False Node and malicious data An intruder might add a node to the system that feeds false data or prevents the passage of true data. Such messages also consume the scarce energy resources of the nodes. This type of attack is called “sleep deprivation torture” in [17]. Insertion of malicious code is one of the most dangerous attacks that can occur. Malicious code injected in the network could spread to all nodes, potentially destroying the whole network, or even worse, taking over the network on behalf of an adversary. A seized sensor network can either send false observations about the environment to a legitimate user or send observations about the monitored area to a malicious user. By spoofing, altering, or replaying routing information, adversaries may be able to create routing loops, attract or repel network traffic, extend or shorten source routes, generate false error messages, partition the network, increase end-to-end latency, etc. Strong authentication techniques can prevent an adversary from impersonating as a valid node in the sensor network.

4. The Sybil attack

In a Sybil attack [18], a single node presents multiple identities to other nodes in the network. They pose a significant threat to geographic routing protocols, where location aware routing requires nodes to exchange coordinate information with their neighbors
to efficiently route geographically addressed packets. Authentication and encryption techniques can prevent an outsider to launch a Sybil attack on the sensor network. However, an insider cannot be prevented from participating in the network, but (s)he should only be able to do so using the identities of the nodes (s) he has compromised. Using globally shared keys allows an insider to masquerade as any (possibly even nonexistent) node. Public key cryptography can prevent such an insider attack, but it is too expensive to be used in the resource constrained sensor networks. One solution is to have every node share a unique symmetric key with a trusted base station. Two nodes can then use a Needham- Schroeder like protocol to verify each other’s identity and establish a shared key. A pair of neighboring nodes can use the resulting key to implement an authenticated, encrypted link between them. An example of a protocol which uses such a scheme is LEAP [8], which supports the establishment of four types of keys.

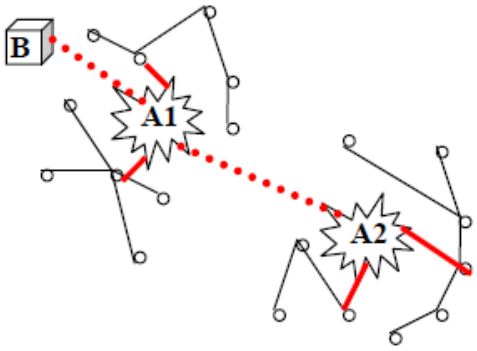
Sinkhole attacks

In a sinkhole attack, the adversary’s goal is to lure nearly all the traffic from a particular area through a compromised node, creating a metaphorical sinkhole with the adversary at the center. Sinkhole attacks typically work by making a compromised node look especially attractive to surrounding nodes with respect to the routing algorithm. For instance, an adversary could spoof or replay an advertisement for an extremely high quality route to a base station. Due to either the real or imagined high quality route through the compromised node, it is likely each neighboring node of the adversary will forward packets destined for a base station through the adversary, and also propagate the attractiveness of the route to its neighbors. Effectively, the adversary creates a large “*sphere of influence*” [5], attracting all traffic destined for a base station from nodes several hops away from the compromised node.

Wormholes

In the wormhole attack [3], an adversary tunnels messages received in one part of the network over a low latency link and replays them in a different part. The simplest instance of this attack is a single node situated between two other nodes forwarding messages between the two of them. However, wormhole attacks more commonly involve two distant malicious nodes colluding to understate their distance from each

other by relaying packets along an out-of-bound channel available only to the attacker. An adversary situated close to a base station may be able to completely disrupt routing by creating a well-placed wormhole. An adversary could convince nodes who would normally be multiple hops from a base station that they are only one or two hops away via the wormhole. This can create a sinkhole: since the adversary on the other side of the wormhole can artificially provide a high quality route to the base station, potentially all traffic in the surrounding area will be drawn through her if alternate routes are significantly less attractive. The following diagram shows an example of a wormhole being used to create a sinkhole:



Adversaries A1 and A2 combine to form a sinkhole-wormhole attack. The nodes near A2 believe that the Base Station B is closer via the sinkhole A1. Hence, the wormhole convinces two distant nodes that they are neighbors by relaying packets between the two of them. A technique for detecting wormhole attacks is presented in [20], but it requires extremely tight time synchronization and is thus infeasible for most sensor networks.

SNEP: Confidentiality, Authentication, Integrity, and Freshness

SNEP uses encryption to achieve confidentiality and message authentication code (MAC) to achieve two-party authentication and data integrity. Apart from confidentiality, another important security property is semantic security, which ensures that an eavesdropper has no information about the plaintext, even if it sees multiple encryptions of the same plaintext [21]. The basic technique to achieve this is randomization: Before encrypting the message with a chaining encryption function (i.e. DESCBC), the sender precedes the message with a random bit string (also called the *Initialization Vector*). This prevents the attacker from inferring the plaintext of encrypted messages if it knows plaintext-ciphertext pairs encrypted with the same key. To avoid adding the additional transmission overhead of these extra bits, SNEP uses a shared counter between the sender and the receiver for the block cipher in counter mode (CTR). The communicating parties share the counter and increment it after each block. SNEP offers the following properties:

Semantic security: Since the counter value is incremented after each message, the same message is encrypted differently each time.The counter value is long enough that it never repeats within the lifetime of the node.

Data authentication: If the MAC verifies correctly, a receiver can be assured that the message originated from the claimed sender.
Replay protection: The counter value in the MAC prevents replaying old messages. Note that if the counter were not present in the MAC, an adversary could easily replay messages.

Data freshness: If the message verified correctly, a receiver knows that the message must have been sent after the previous message it received correctly (that had a lower counter value). This enforces a message ordering and yields weak freshness.

Low communication overhead: The counter state is kept at each end point and does not need to be sent in each message.

Conclusion

In this paper, we introduce sensor networks, its related security problems, threats, risks and characteristics, and a brief introduction to SNEP . SNEP provides security for base station-to-sensor communication at some degree .Attacks on WSNs stimulate the design of secure routing protocols .For implementation details and performance evaluation of these protocols, please refer to the [1, 7, 8]. Adding security in a resource constrained wireless sensor network with minimum overhead provides significant challenges, and is an ongoing area of research.

References

[1] Adrian Perrig, Robert Szewczyk, Victor Wen, David Culler, Tygar J. D. (2001) Security Protocols for Sensor Networks. In *The Seventh Annual International Conference on Mobile Computing andNetworking (MobiCom 2001).*

[2] Sasha lijepcevic, Miodrag Potkonjak, Vlasios Tsiatsis, Scott Zimbeck, Mani B.Srivastava (2002) On Communication Security in Wireless Ad-Hoc Sensor Networks. In The Proceedings of the *Eleventh IEEE. International Workshops on Enabling Technologies: Infrastructure*

for Collaborative Enterprises (WETICE-02).

[3] Hu Y.C., Perrig A., and Johnson D. B. (2002) *"Wormhole detection in wireless ad hoc networks,"* Department of Computer Science, Rice University, Tech. Rep.TR01- 384.

[4] Jeffery Under coffer, Sasikanth Avancha, Anupam Joshi and John Pinkston. In *Security for Sensor Networks*.

[5] Chris Karl, David Wagner. In Secure Routing in Wireless Sensor Networks: Attacks and Countermeasures.

[6] Wadaa, Olariu S.,Wilson L., Eltoweissy M., Jones K. (2004) On Providing Anonymity in Wireless Sensor Networks. In Proceedings of the *Tenth International Conference on Parallel and Distributed Systems (ICPADS'04).*

[7] Chris Karlof, Naveen Sastry, David Wagner. (2004) *TinySec: A Link Layer Security Architecture for Wireless Sensor Networks. ACM SenSys*. November 3-5.

[8] Sencun Zhu, Sanjeev Setia, Sushil Jajodia. (2003) LEAP: Efficient Security Mechanisms for Large-Scale Distributed Sensor Networks. In The Proceedings of the *10th ACM conference on Computer and communications security.*

[9] Alan Mainwaring, Joseph Polastre, Robert Szewczyk and David Culler. (2002) Wireless sensor networks for habitat monitoring. In *First ACM International Workshop on Wireless Sensor Networks and Applications.*

[10] Robert Szewczyk, Joseph Polastre, Alan Mainwaring and David Culler. (2004) Lessons from a sensor network expedition. In *First European Workshop on Wireless Sensor Networks (EWSN-04).*

[11] Matt Welsh, Dan Myung, Mark Gaynor and Steve Moulton. (2003) *Resuscitation monitoring with a wireless sensor network*. In Supplement to Circulation: *Journal of the American Heart Association.*

[12] Duckworth G.L., Gilbert D.C. and Barger J.E. (1993) Acoustic counter-sniper system. In SPIE *International Symposium on Enabling Technologies for Law Enforcement and Security.*

[13] Hill J., Szewczyk R., Woo A., Hollar S.,Culler D. and Pister K. (2000) System architecture directions for networkedsensors. In Proceedings of *ACM ASPLOS IX.*

[14] Intanagonwiwat C., Govindan R. and Estrin. D. (2000) Directed diffusion: A scalable and robust Communication paradigm for sensor networks In Proceedings of *MobiCOM-00*,

[15] Boston, Massachussetts, Karlof C., Li Y. and Polastre J. (2003) ARRIVE: *An Architecture for Robust RoutingIn Volatile Environments.Technical Report* UCB/CSD-03-1233, University of Californi at Berkeley,

[16] Madden S., Szewczyk R., Franklin M. and Culler D. (2002) Supporting Aggregate Queries over Ad-Hoc Wireless Sensor Networks. In *4th IEEE Workshop on Mobile Computing Systems and Applications*.

[17] Stajano F. and Anderson R. (1999) "The Resurrecting Duckling: Security Issues for Ad-hoc Wireless Networks", *3rd AT&T Software Symposium*, Middletown, Newzealand.

[18] Douceur J. R. (2002) "The Sybil Attack," in *1st International Workshop on Peer-to- Peer Systems (IPTPS-02).*

[19] Rivest R. L., Robshaw M.J.B., Sidney R. and Yin Y.L., "The RC6 Block Cipher", AES submission http://theory.lcs.mit.edu/~rivest/rc6. pdf

[20] Hu Y.C., Perrig A. and Johnson D. B.(2002) "*Wormhole detection in wireless ad hoc networks*", Department of ComputerScience, Rice University.

[21] Shafi Goldwasser and Silvio Micali. (1984) Probabilistic encryption. *Journal ofComputer Security*, 28:270–299.

[22] Adrian Perrig, Ran Canetti, Dawn Song,and Tygar J. D. (2001) *Efficient and securesource Authentication for multicast. In Network and Distributed System Security Symposium, NDSS-01.*

[23] Adrian Perrig, Ran Canetti, Tygar J.D. and Dawn Song N. (2000) Efficient authentication and signing of multicast streams over lossy channels. In *IEEESymposium on Security and Privacy.*

[24] Samuel R., Michael J., Franklin, Joseph M.,Hellerstein and Wei Hong (2002) TAG: A tiny aggregation service for ad-hoc sensor networks. In The *Fifth Symposium on Operating Systems Design andImplementation.*

[25] Samuel R., Madden, Robert Szewczyk,Michael Franklin J. and David Culler (2002)Supporting Aggregate Queries over ad-hocwireless sensor networks. In Workshop on *Mobile Computing and SystemsApplications*.

[26] Bellare M., Desai A., Jokipii E. andRogawayP. (1997) A concrete security treatment ofSymmetric encryption: Analysis of the DESModes

of operation. In Proceedings of *38th Annual Symposium on Foundations ofComputer Science (FOCS-97).*

[27] Bruce Schneier (1996) AppliedCryptography, *Second Edition. John Wiley & Sons.*

[28] Mihir Bellare, Joe Kilian and Phillip Rogaway (2000) the security of the cipher block chaining message authenticationcode.*Journal of Computer and SystemSciences*.