Designing an Improved KISS Environment

Mohinder Singh[1], Harmandeep Singh[2]

Department of Computer Engineering, Punjabi University, Patiala, Punjab, India

Abstract: ***Wireless Sensor Networks consist of a number of wireless nodes connected to each other using wireless connections. Because these nodes are wirelessly connected with each other and base stations, they are highly prone to the hacking attacks. When the WSN nodes are in working condition, they need secure cryptographic keys for secure propagation of the sensitive information. Existing cryptographic key management and distribution technique usually consume higher amount of energy and put larger computational overheads on Wireless Sensor Nodes. An effective corporate key management and distribution policy is required to maintain the security of the wireless sensor nodes without compromising battery power. None of the current commercial systems (either based on software or hardware security modules) or research proposals effectively address both challenges. This paper represents improved key management architecture, called KISS for the WSNs, to enable comprehensive, trustworthy, user-verifiable, and cost-effective key management. KISS allows only authorized applications and/or users to use the keys. Using simple devices, administrators can remotely issue authenticated commands to KISS and verify system output.***

Keywords: Sensoe, Network, Directed Diffusion, BTS, Key

1. Introduction

A Wireless Sensor Network (WSN) uses radio communication in an autonomous and distributed manner. Nodes are distributed over a specific field, and are able to collect and relay information about the environment. A sensor node is typically equipped with one or more sensors that are used to capture events from the environment, an analog-digital converter, a radio transceiver, a central processing unit with limited computational capabilities, a small amount of memory and a battery power supply. Sensor devices collaborate with each other in order to perform basic operations such as sensing, communication and data processing.

Major applications using WSNs include: environmental monitoring, health care, mood-based services, positioning and animal tracking, entertainment, logistics, transportation, home and office, industrial and military applications. Health care applications enable people with certain medical conditions to receive constant monitoring through sensors. Military applications include surveillance, target tracking, counter-sniper systems and battlefield monitoring, in which information is propagated to soldiers and vehicles involved in combat.

A sensor network consists of an array of sensors scattered for distributed monitoring of real time events. The sensor networks have limited energy, as the sensor nodes are powered by batteries. The sensor nodes also have limited amount of memory and computational power although these can be deployed in remote areas. There has been a booming use of sensor networks for life critical applications such as monitoring patients in hospital, military applications etc. These applications require a good security environment for sensor networks. This is because the design of a security protocol for these areas is very challenging. In this paper security issues in Directed diffusion are considered. Directed Diffusion is a noble routing protocol for sensor networks. This provides consideration of possible attacks and possible measures. The paper poses a brief picture of those attacks as well as ways to prevent them.

The security of Wireless Sensor Networks (WSN) can be compromised in many ways. A remote end user accessing base station information can be prevented from doing so in a number of ways. Communication between the base station and sensor nodes can be blocked. This can be accomplished by analog jamming of signals or by digital jamming in the form of DoS(Denial of Service) attacks that flood the network, base stations or both. Targeted DoS attacks on strategic nodes in the WSN can also block communication of large parts of the network with the base station. Communication between base stations and other sensor nodes can be prevented by setting up incorrect routing information so that traffic goes to the wrong destination or loops. One way to do this is to spoof the base station and deceive nodes into rerouting all packets to the spoofed base station instead of the real base station.

Another way of breaching security is to destroy the base station itself. This can be accomplished by monitoring the volume and direction of packet traffic toward the base station so that the location is eventually revealed. Destruction can also be accomplished by listening to the RF signals to locate the base station. A third threat is eavesdropping. This is made easier by wireless hop-to-hop communication. Eavesdropping can be used to track and deduce the location of the base station for destruction. There are many other methods to breach the WSN security. In this paper, we are considering the whole WSN network in a new way.

2. Existing KISS key exchange scheme

The existing KISS scheme was to ensure the trusted and secure links between the servers to ensure the security of the data being shared or transferred between them. The existing scheme is based on client-server model for cryptographic key management and key sharing between the data or application servers. The existing scheme was proposed to protect or harden (to make more secure) the previous key management and key sharing scheme. Terminologies used in the existing scheme are KISS server, KISS client, KISS TAD (Trusted Administrator Devices), KISS Remote Managers.

Algorithm 1: Existing Client-Server based KISS scheme
Assumptions:

1.) KISS Server
2.) KISS Client
3.) KISS TAD
4.) Remote Managers
5.) All of the nodes are connect to direct power sources
6.) KISS Server manages all key related operations

Algorithm Flow

1. **Server X** initializes communication with **Server Y**
2. **Server Y** requests Key Management Server $\mathbf{K_{server}}$.
3. $\mathbf{K_{server}}$ connects **Server X**.
4. $\mathbf{K_{server}}$ sends security key to **Server X.**
5. Server X replies with the calculated answer key
6. $\mathbf{K_{server}}$ attest the **Server X** as valid node
7. $\mathbf{K_{server}}$ updates the **Server Y**
8. **Server Y** starts communication with **Server X**

3. Proposed KISS scheme for WSN:

The proposed scheme is specially proposed for wireless sensor networks (WSN). The wireless sensor nodes are battery operated devices, Hence, having limited power sources. The WSN usually sends data to Base stations via long distance wireless communication at most of the times. Sometimes, WSNs does not send data anywhere to any BTS. Long distance wireless communications are not considered as the safe communications. Firstly considered option was to host the key management services on BTS. But it was not possible, because, if the key management services will be hosted on BTS, they are prone to hacking because of the long distance communication with the wireless sensor networks. In case BTS does not exist, the key management service will not work. Second consideration was to host the key management services on Cluster Heads, but it could choke the battery of cluster heads quickly, which may dent the performance of WSN. Then, there was a third and final option available which was to make a self-adaptive key management solution for WSN nodes. In the proposed solution, the self-adaptive key management scheme has been implemented.

Algorithm 2: Proposed KISS key exchange scheme
Assumptions

1.) All nodes are KISS scheme aware nodes
2.) All nodes can serve as client or server
3.) All nodes can encrypt decrypt the key information
4.) All nodes are battery operated nodes

ALGORITHM FLOW

1. Node X builds a secure key table
2. Node X sends its secure key table with Node Y
3. Node Y saves Node X table in its memory
4. Node X senses the data
5. Node X transmit the data to Node Y for first time
6. Node Y sends a secure key to Node X
7. Node X replies with the secure reply Key
8. Node Y matches the Key
    a. If key matches
        i. the communication takes place.
    b. Else
    The communication request rejected by Node Y

Key Generation Policy: Key generation policy under the proposed model is using the following mathematical algorithmic flow to populate the key table which is saved and being exchanged between the sensor nodes in the working cluster.

Algorithm 3: Random Function to generate random number

A. First, initialize the random number generator to make the results in this example repeatable.
B. Create a radii value for each point in the sphere. These values are in the open interval, $(0,3)$, but are not uniformly distributed. The values have been created using the mathematic equation:

$$f(x) = 3 * \int_{1}^{1000000} random * \frac{1}{3}$$

C. Randomly select and concatenate the coordinates or values to create the OTP.
D. Return OTP

4. Security Analysis

This section analyzes potential attacks on KISS and our defense mechanisms.

System Bootstrap

During the system bootstrap in WSN nodes, the hackers can attempt to inject the malicious content or malware on the sensor nodes. The proposed solution is self adaptive; hence, every node is made capable of taking care of its own security. The node will authenticate the nodes only after they will share a key table of certain measurement. Also, the received data will be verified using the randomized key from secure key table, which ensures the security against the penetration attacks for malware injection in the initial stages.

Key Life-cycle Operations

Another threat is generated when the hackers try to change the key tables saved on the sensor nodes or to change the key generation policy by tweaking into the node software. The WSN nodes carry a natural protection against software tweaking attack, because embedded system carry chip level programming which can't be changes on the fly. Also, the nodes will not flush or change the table on any of the hacking attempt by sending the data to the sensor nodes. The sensor nodes accept the data from the nodes replying with the reply keys. The key table is generated using high randomization based secure code generation technique, which does not hold any computational dependency on the base key. So the hacker can't gain the authorization and can't change key information table.

Key Generation and Usage Control

The key generation policy used in the proposed model is based on the high randomization and mathematical array value shuffling operation, which creates highly randomized and undependable numeric keys. Any of the key in the key table can't be calculated mathematically to find the next key in the table. Unauthorized applications and hackers cannot bypass the KISS scheme running on the sensor nodes because, to gain the authority to send the data to the sensor nodes, one has to obtain the authorization by sending a reply or response key in return to the request or question key sent they sensor node on receiving a data stream. However, this administrative operation can be recorded in the sensor node audit log and held accountable.

5. Comparisons and Improvements

Table 1: KISS System Operation Categorization for proposed and existing schemes

Category	*Operations*	*Proposed Scheme*			*Existing Scheme*		
		Local or Remote	*Quorum or Any*	*Manual or Automatic*	*Local or Remote*	*Quorum of Any*	*Manual or Automatic*
1	Node Bootstrap, adding neighbors/administrators	Local	Either	Automatic	Local	Quorum	Manual
2	Removing Neighbors/Administrators	Local	Either	Automatic	Either	Quorum	Manual
3	Client Bootstrap	Local	Either	Automatic	Local	Either	Manual
4	Client Registration	Local	Either	Automatic	Either	Either	Manual
5	Server/client key-life cycle operations	Local	Either	Automatic	Either	Either	Either

References

[1] Zongwei Zhou, Jun Han, Yue-Hsun Lin, Adrian Perrig, Virgil Gligor, “KISS: Key it Simple and Secure Corporate KeyManagement”, Trust and Trustworthy Computing Lecture Notes in Computer Science, volume 7904, pp. 1-18, Springer, 2013.

[2] N. Suganthi, V. Sumathy, “Energy Efficient Key Management Scheme for Wireless Sensor Networks”, vol 9, issue 1, pp. 71-78, INT J COMPUT COMMUN, 2014.

[3] Ivan Damgård, Thomas P. Jakobsen, JesperBuus Nielsen, and Jakob I. Pagter, “Secure Key Management in the Cloud”, Cryptography and Coding Lecture Notes in Computer Science, volume 8306, pp. 270-289, Springer, 2013.

[4] RamaswamyChandramouli, Michaela Iorga, SantoshChokhani, “ Cryptographic Key Management Issues & Challenges in Cloud Services”, Computer Security Division Information Technology Laboratory, NIST, 2013.

[5] Marco Tiloca, Domenico De Guglielmo, GianlucaDini and Giuseppe Anastasi, “SAD-SJ: a Self-Adaptive Decentralized solution against Selective jamming attack in Wireless Sensor Networks”, ETFA, vol. 18, pp. 1-8, IEEE, 2013.

[6] Md. MonzurMorshed, Md. Rafiqul Islam, “CBSRP: Cluster Based Secure Routing Protocol”, IACC, vol. 3, pp. 571-576, IEEE, 2013.

[7] Haowen Chan, Adrian Perrig, and Dawn Song. Random key predistribution schemes for sensor networks. In SP ’03: Proceedings of the 2003 IEEE Sympo- sium on Security and Privacy, page 197, Washington, DC, USA, 2003. IEEE Computer Society.

[8] Haowen Chan, Virgil D. Gligor, Adrian Perrig, and Gautam Muralidharan. On the distribution and revocation of cryptographic keys in sensor networks. IEEE Trans. Dependable Sec. Comput., 2(3):233–247, 2005.

[9] Seyit Ahmet C¸ amtepe and Bu¨lent Yener. Combinatorial design of key distri- bution mechanisms for wireless sensor networks. In ESORICS, pages 293–308, 2004.

[10] Jooyoung Lee; D.R. Stinson. A combinatorial approach to key predistribution for distributed sensor networks,. Wireless Communications and Networking Conference, 2:1200–1205, 13-17 March 2005.

[11] Jooyoung Lee; D.R. Stinson. Common intersection designs,. Journal of Com- binatorial Designs, 14:251–269, 2006.

[12] Sushmita Ruj and Bimal Roy. Key predistribution schemes using codes in wireless sensor networks. pages 275–288, Berlin, Heidelberg, 2009. Springer-Verlag.