Architecture for Application Autonomic Protection Principles in Wireless Sensor Network

Hosam Soleman
Department of Computer Engineering
Maleke-Ashtar University
Islamic Republic of Iran

Ali Payandeh
Department of Computer Engineering
Maleke-Ashtar University
Islamic Republic of Iran

ABSTRACT

Wireless sensor networks (WSNs) as an emerging technology face numerous challenges. Sensor nodes are usually resources constrained,also, they are vulnerable to physical attacks or node compromises. Autonomic Computing is a steadily emerging and promising research field. In the domain of simplifying interoperability, it aims to diminish the management complexity in several industries and systems. Self-protection in WSNshasn't been deeply studied before, because ofthe high rate of fails. The major concern in WSN is to maximize the network's Lifetime.In this paper,a framework that embeds autonomic capabilities into WSN systems is proposed. The proposed framework provides self-protection features in cases of unauthorized, inadvertent and intentional change in security parameters.

Keywords: Wireless sensor network, Autonomic system,self-protection, structure.

1. INTRODUCTION

Wireless sensor networks (WSNs) as an emerging technology faces numerous challenges. Sensor nodes are usually resource constrained. Sensor nodes are also vulnerable to physical attacks or node compromises [1]. Autonomic Computing is a steadily emerging and promising research field. It aims at simplifying interoperability to diminish the management complexity in several industries and systems. Self-protection in WSN (Wireless Sensor Networks) has not been deeply studied because WSN has a high rate of fails. The major concern in WSN is to maximize the network's Lifetime.

WSN administrators face increasingly more difficult challenges brought about by

The growing complexity of WSN, which stems from several sources:

- Increased using WSN.
- Advances in WSN functionality, connectivity, availability, and heterogeneity. WSN administrator must grapple with complex decisions about hardware platforms, schema design, constraints and referential integrity, primary keys, indexes, materialized views, the allocation of sensors.
- Ongoing maintenance.
- Using the WSN in sensitive applications. (Military applications, medical applications...etc.).

In the current worka framework is proposed. The introduced framework embeds autonomic capabilities into WSN systems to provide self-protection features in cases of Unauthorized, Inadvertent and Intentional change in security parameters.

2. WIRELESS SENSOR NETWORK

Wireless sensor networks are often used in military systems[2], but they are also employed in various other systems (in commerce, in the service industry, in medicine, etc.).

The WSN combines sensing, computation and communication in a single small device, called Sensor Node. The sensor node mainly contains radio, battery, microcontroller and power devices [3]. Figure 1 shows the Architecture of WSN [4, 5].

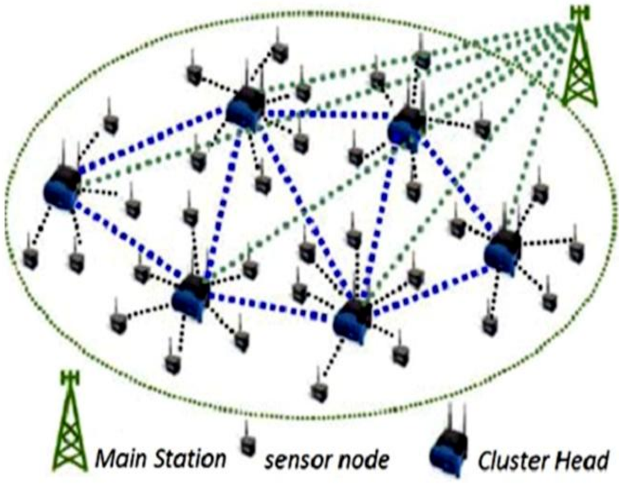

Fig1: Architecture of WSN

3. AUTONOMIC COMPUTING

Autonomic Computing is a steadily emerging and promising research field. It aims at simplifying interoperability to diminish the management complexity in several systems.

Autonomic computing systems consist of four attributes. As illustrated in the following 4-quadrant chart, they are [6]:

- Self-configuring (able to adapt to changes in the system)
- Self-healing (able to recover from detected errors)
- Self-optimizing (able to improve use of resources)
- Self-protecting (able to anticipate and cure intrusions)

4. FEASIBILITY OF BASIC SECURITY SCHEMES IN WIRELESS SENSOR NETWORK

Security is a broadly used term encompassing the characteristics of authentication, integrity, privacy, nonrepudiation, and anti-playback [7]. In this section discusses the network security fundamentals and how the techniques are meant for wireless sensor networks.

4.1 Cryptography

The encryption-decryption techniques devised for the traditional wired networks are not feasible to be applied directly for the wireless networks and in particular for wireless sensor networks. WSNs consist of tiny sensors which really suffer from the lack of processing, memory and battery power [8, 9].

4.2 Steganography

While cryptography aims at hiding the content of a message, Steganography aims at hiding the existence of the message. Steganography is the art of covert communication by embedding a message into the multimedia data (image, sound, video, etc.) [10].

4.3 Physical Layer Secure Access

Physical layer secure access in wireless sensor networks could be provided by using frequency hopping. A dynamic combination of the parameters like hopping set (available frequencies for hopping), dwell time (time interval per hop) and hopping pattern (the sequence in which the frequencies from the available hopping set is used) could be used with a little expense of memory, processing and energy resources [11].

5. ARCHITECTURE OF THE PROPOSED FRAMEWORK

The introduced framework is based upon the following steps: 1) Network architecture of WSN that have been adopted in the research. 2) Architecture of a Self-protecting WSN.3) Flowchart of the uninterrupted self-protection approach.

5.1 Network architecture of WSN that hasbeen adopted in the current research:

The traditional network architecture of WSN is self-organized and flat, so it is not easy to centralized monitor all the devices operation. The network architecture of WSN is declared as shown in Figure 2. It introduces the idea of central control and helps us grasp the network conditions in WSN.

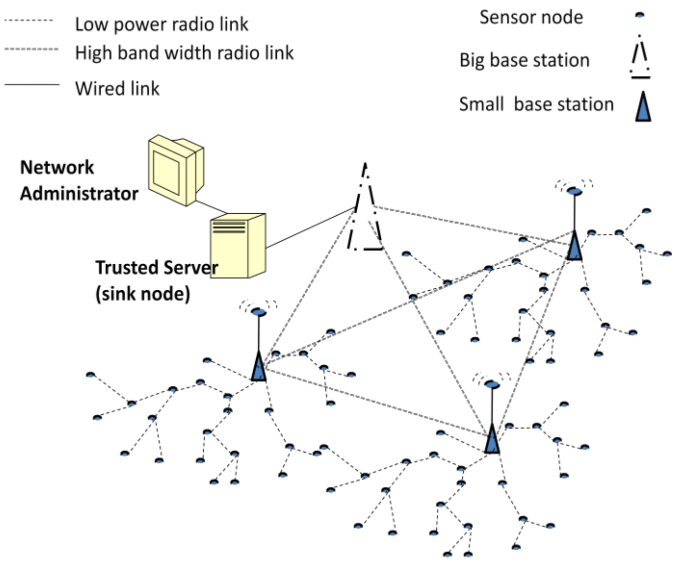

Fig2: Network Architecture of WSN

It contains lots of sensor nodes, several small base stations, one big base station, one trusted server (Sink Node) and one person monitor system. Here, a firewall to prevent inner WSN from outer internet attacks can be set up.

5.2Architecture of the Framework

The WSN owner and WSN administrator can send commands or queries to a Sink Node (BS) which spreads those commands or queries into the network. Therefore, it is important to build security policies within the BS to prevent the illegal commands and requests that issued by:

- Administrator of the network.
- Attackers (internal or external).

The proposed architecture consists of the following steps (see Figure3):

1. The system owner builds policies to support certain objectives.
2. These policies are stored in the same sink nods of WSN being protected and are used to verify requests (or attempts) to change system configurations and to enforce the policy mandates in case ofexisting apower user or a hacker.
3. Initiates an attempt to change security configurations.
4. The request goes through a process of verification before it can be processed. This step is carried out by sink node stored procedures that have built-in logic for checking the request against the policies.
5. If the request complies with the set policies that govern their scope of applicability.
6. The request is applied, and then the WSN system information is updated to reflect the change.
7. An audit trail is recorded.
8. Otherwise, the request is rejected and the system owner is alerted.
9. The user is notified.
10. And an audit trail is recorded.

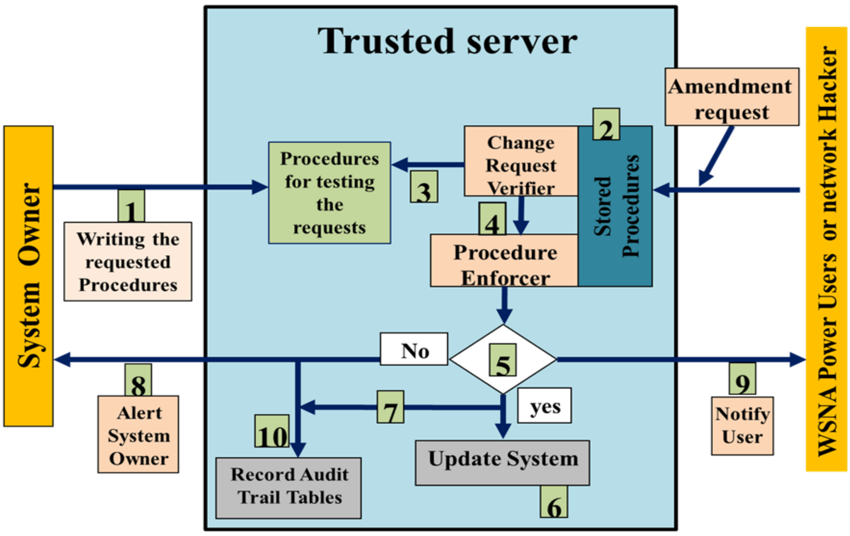

Fig3: Architecture of a Self-protecting WSN

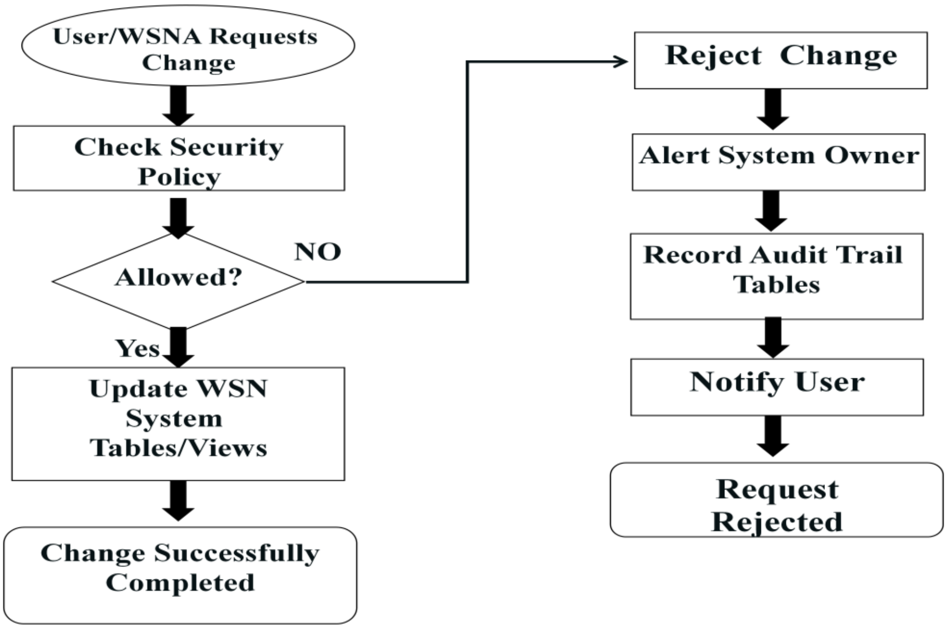
User/WSNA Requests Change
Check Security Policy
Allowed?
NO
Yes
Update WSN System Tables/Views
Change Successfully Completed
Reject Change
Alert System Owner
Record Audit Trail Tables
Notify User
Request Rejected

Fig4: Flowchart of the uninterrupted self-protection approach

Figure4,shows the flowchart of the processes involved in providing uninterrupted self-protection through embedding autonomic capabilities into the WSN.

Briefly, the self-protection mechanism ensures that all requests to the system are verified against the security policy. Every request submitted to the system must comply with the security policy for it to be applied. Requests that are intended to affect the security policy itself can only come from the super system owner. This ensures that a system owner alone cannot make changes to the vital security policy of the system. It has to be a collective agreement among the system owners. Based on the nature of the request submitted to the system, one of the following processes is carried out:

- Verify Security Policy: marks a request as safe to execute or reject.
- Process Request: formulates an execution plan and prepares the appropriate system processes or code commands and executes them against the system in order to process a user’s request.
- Reject Request: ensures that all requests marked as reject are not applied to the system.
- Alert System Owner: alerts the system owners of suspicious requests or activities.
- Record Audit Trail: records all actions submitted to the system in an audit trail repository for usage by the integrated business intelligence capability of the framework.
- Inform User: sends informative messages to the user regarding the status of their requests.

This structure can be used in the detection of hostile nodes when they want access the network, as follows:

1. Hostile node sends a joining request to the network.
2. When sensors receive the request they send this request to the sink node.

3. The sink node tests that request.
4. Due to that test, the sink node sends a message to the all nodes that have received this hostile request. That message determines if a node is hostile or not.

5.3 Types of the Requests

The proposed architecture has to protect the system from the following requests:

- Switching node from active mode to non-active mode.
- Changing the radio range.
- Deleting node.
- Deleting base station.
- Adding node.
- Changing identifier of (node, base station, and sink node).
- Changing (encryption key, public key, and coding key…).
- ….etc.

5.4 The general structure of the request

The structure of the request contains the following fields:

- Request_num: the request number.
- Request_name: the name of request (adding, deleting, switching, changing …etc).
- Request_param: the request parameters.
- Request_date: the request sending date.

5.5 Policies

Policy-based computing is one of several techniques which used to implement autonomic capabilities in computer systems. In the current research, the using of policies to implement autonomic capabilities into a WSN has been applied.The system owner is allowed to create WSN configuration specific policies which decide the actual run-time behavior of the WSN.

These policies control and decide which changes are allowed and which ones are not. This is based on the WSN being aware of its operational state being able to defend itself against input from various environmental sources such as users, malicious code, or external software systems.

In general, policies provide the capability of controlling who can do what, when, where, and how. But the use of policies can also be exploited to predict the reason of performing a certain action. Based on the specifics of the action, the system can gather and formulate intelligence that would be the basis for deducing the motive behind the action.

The policies (Algorithms),that have beenbuilt in the second work have to verify the security's goals, are:

1- Policy for protect WSN from deleting node illegally.
2- Policy for protect WSN from adding node illegally.
3- Policy for avoiding changing attributes of WSN elements illegally from:

- Administrator.
- Attackers.
- WSN users.

4- Policy for protect Sink node attributes:

- Transmission range.
- Nod ID.

6. CONCLUSION:

A new structure for application autonomic principles in WSNs is proposed. The introduced structure verifies the following features:

1. A central command for the network.
2. Protection from internal attacks that affect the performance and specifications of the network.
3. Detect a malicious entry to the network.
4. The energy consumption in sensor nodes is low in the event of the discovery process to enter malicious nodes to the network.
5. The sending data in the event of detection are low.
6. There is no needfor more memory's size in the case of intrusion detection.
7. There is no processing in the sensor nodes in the case of intrusion detection. (The task of the sensor node is sending the access request to the sink node).
8. Analysis and processing of the security requirements are performed in the sink node. This node has the following features:
   - ➢ Sophisticated processor.
   - ➢ Large capacity memory.
   - ➢ A lot of power, and when it is available:
     - Change the power supply.
     - Recharged the power supply continuously.

7. REFERENCES

[1] Lili Yang, "Determining Sink Node Locations in Wireless Sensor Networks", IEEE, pp3400-3404.2006,

[2] Chien-Chung Shen ,ChaipornJaikaeo , Chavalit ," Sensor Network Architecture and Applications ", IEEE Personal Communications Magazine, 2005.

[3] QutaibaIbrahem Ali, AkramAbdulmaowjod, Hussein Mahmood Mohammed, "Simulation & Performance Study of Wireless Sensor Network (WSN) Using MATLAB", IEEE, pp307-314. 2010,

[4] Kurak, C and McHugh, J, "A Cautionary Note on Image Downgrading in Computer Security Applications", Proceedings of the 8th Computer Security Applications Conference, San Antonio, December, pp. 153-159.1992.

[5] Mokowitz, I. S., Longdon, G. E., and Chang, L., "A New Paradigm Hidden in Steganography", Proc. of the 2000 workshop on New security paradigms, Ballycotton, County Cork, Ireland, pp. 41 – 50.2001.

[6] A Practical Guide to the IBM Autonomic Computing Toolkit (Red Book).

[7] Undercoffer, J., Avancha, S., Joshi, A., and Pinkston, J., "Security for Sensor Networks", CADIP Research Symposium, 2002, available at, http://www.cs.sfu.ca/~angiez/personal/paper/sensor-ids.pdf.

[8] Jolly, G., Kuscu, M.C., Kokate, P., and Younis, M., "A Low-Energy Key Management Protocol for Wireless Sensor Networks", Proc. Eighth IEEE International Symposium on Computers and Communication, 2003. (ISCC 2003).vol.1, pp. 335 – 340.

[9] Rabaey, J.M., Ammer, J., Karalar, T., Suetfei Li., Otis, B., Sheets, M., and Tuan, T., "PicoRadios for wireless sensor networks: the next challenge in ultra-low power design" IEEE International Solid-State Circuits Conference (ISSCC 2002), Volume 1, 3-7 Feb., pp. 200 – 210.

[10] Kim, C. H., O, S. C., Lee, S., Yang, W. I., and Lee, H-W., "Steganalysis on BPCS Steganography", Pacific Rim Workshop on Digital Steganography (STEG'03), July 3-4, Japan, 2003.

[11] Orihashi, M., Nakagawa, Y., Murakami, Y., and Kobayashi, K., "Channel synthesized modulation employing singular vector for secured access on physical layer", IEEE GLOBECOM 2003, Volume 3, 1-5 December, pp. 1226 – 1230.2003.