A ROBUST USER AUTHENTICATION SCHEME FOR WIRELESS SENSOR NETWORK

by

M. Zulfiker Ali

BASc in Electrical & Electronic Engineering
BUET, Dhaka, 1995

A thesis

presented to Ryerson University

in partial fulfillment of the

requirements for the degree of

Master of Applied Science

in the Program of

Electrical and Computer Engineering

Toronto, Ontario, Canada, 2013

AUTHOR'S DECLARATION FOR ELECTRONIC SUBMISSION OF A THESIS

I hereby declare that I am the sole author of this thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I authorize Ryerson University to lend this thesis to other institutions or individuals for the purpose of scholarly research.

I further authorize Ryerson University to reproduce this thesis by photocopying or by other means, in total or in part, at the request of other institutions or individuals for the purpose of scholarly research.

I understand that my thesis may be made electronically available to the public.

Abstract

Title	:	A Robust User Authentication Scheme for Wireless Sensor Network
Author	:	M. Zulfiker Ali
Degree	:	Master of Applied Science, 2013
Program	:	Electrical and Computer Engineering
University	:	Ryerson University, Toronto, Ontario, Canada

The primary requirements of a secure Wireless Sensor Network architecture are confidentiality, integrity and authentication of users and other participating entities. User Authentication for wireless sensor networks is a fundamental and important issue in designing dependable and secure systems. In this thesis, we have outlined the security model, functional requirements, assumptions and network setup for an authentication scheme in the first phase. Keeping in mind the security requirements as well as the flaws of past authentication schemes, we propose a robust user authentication method that inherits user anonymity, mutual authentication and password changing functionality of previous password-based schemes and improves security by resisting gateway bypass and replay attack, and many logged in user with the same ID threat. Our scheme is a variant of strong password based schemes that does not require strict network synchronization.

In the second phase of the thesis, we have analysed our authentication scheme from the perspective of security issues and functional requirements. The proposed scheme is modelled in SystemC. It is evaluated in different attack scenarios. The authentication latency, memory and functional requirements, and computational overhead are the metrics used to evaluate the scheme. The effect of multiple users on authentication latency in our scheme is also studied.

Some of the past representative schemes have also been modelled and evaluated in the same environment. A detailed comparison of over-head cost, authentication latency and security features are provided in this thesis. It is verified and confirmed by modeling that our scheme provides enhanced security without adding extra computation at the sensor node.

Acknowledgements

There is a person without whom, I am certain I never would have completed this work. She is my wife, Shahin, who always has belief in me and never stops encouraging me. I cannot imagine I could pursue and complete my MASc without her support and sacrifice.

I would like to express my deep appreciation to my supervisor, Dr. Gul N. Khan, for his expert guidance, insightful discussions and patient encouragement throughout my research and thesis writing. I would like to thank the Pervasive and Embedded Computing and Communication Systems (PECCS) review committee for their constructive suggestions. I do appreciate Mr. Masoud for continuously providing helpful advices for my research and career planning. Besides, I am also grateful to my colleagues in the Microsystems Research Lab and all my friends at Ryerson University for their help and support during my study.

I also acknowledge the financial support provided by NSERC and the Electrical & Computer Engineering Department at Ryerson University.

This thesis is dedicated to my wife

Abbreviations

ACL	Access Control List
ASIC	Application Specific Integrated Circuit
CH	Cluster Head
CSMA-CA	Carrier Sense Multiple Access - Collision Avoidance
DARPA	Defence Advanced Research Projects Agency
DoS	Denial of Service
DSN	Distributed Sensor Network
DSP	Digital Signal Processor
ECC	Elliptic Curve Cryptography
FPGA	Field Programmable Gate Array
ISM	Industrial, Scientific and Medical
IV	Initialization Vector
LAN	Local Area Network
LWIM	Low-power Wireless Integrated Micro-sensors
MAC	Medium Access Control
MAN	Metropolitan Area Network
QoS	Quality of Service
RF	Radio Frequency
TCP	Transmission Control Protocol
TDMA	Time Division Multiple Access
UA	User Authentication
WAN	Wide Area Network
WINS	Wireless Integrated Network Sensors
WPAN	Wireless Personal Area Network
WSN	Wireless Sensor Network

Table of Contents

List of Tables

List of Figures

Chapter 1: Introduction

Recent advances in integration between tiny embedded processors, wireless interfaces, and micro-sensors have led to the emergence of ***wireless sensor networks (WSN)***. This new kind of wireless networks has gained worldwide attention in recent years and has been integrated in several application domains. The background of WSN is outlined in this chapter. The motivation behind this research work and the objective of the research are discussed. A brief overview of the thesis concludes this chapter.

1.1 Background

The origins of WSN research can be traced back to the Distributed Sensor Networks (DSN) program at the Defence Advanced Research Projects Agency (DARPA) in the eighties where Arpanet (predecessor of the Internet) approach for communication was extended to sensor networks. DSNs were assumed to have many spatially distributed low-cost sensing nodes that collaborate with each other but operate autonomously. Technology components for a DSN included sensors (acoustic), communication and processing modules, and distributed software [66]. In the early eighties the state of the art real-time test bed was developed by MIT Lincoln Laboratory using Ethernet and microwave radio for acoustic tracking of low-flying aircraft [32]. DARPA acted as a pioneer in sensor network research by launching Low-power Wireless Integrated Micro-sensors (LWIM) project during mid-1990s and continued the initiative research program called SensIT [25]. It provided the present sensor networks with new capabilities such as ad hoc networking, dynamic querying and tasking, reprogramming and multitasking. Since 1993, the Wireless Integrated Network Sensors (WINS) project at the University of California at

Los Angeles covers almost every aspect of wireless sensor network design. The University of California at Berkeley started the PicoRadio program in 1999 to support the assembly of an ad-hoc wireless network of low-cost, low-energy sensor and monitor nodes [31]. The µAMPS program at MIT focused on the development of a complete system for wireless sensor networks, emphasizing the need for low power operation [15]. Several applications have been benefited from the advances in wireless sensor networks. Today such networks are used in Agriculture, Health Care, Defence, Wild-Life Habitat Monitoring, Under Water Monitoring, Disaster Management (Safety) and Industrial (monitoring, control, factory automation) applications. The current WSN research focuses on application-driven systems in order to address more concrete issues.

1.2 Motivation of the Research Work

Wireless Sensor Networks (WSNs) have attracted a large number of researchers due to its ubiquitous nature, easy deployment and a wide range of applications. In general, most of the queries in WSN applications are issued at the points of base stations or Gateway (GW) node of the network. However, one can foresee that there are greater needs to access the real-time data inside WSN. The user must be able to access the real-time data from sensor nodes when required. For some applications, the collected data is valuable and confidential. In many applications, integrity and confidentiality of collected data as well as the user privacy are critical. Security measures should be incorporated to protect the access to critical data and to restrict non-authorized users from acquisition of critical data. If the data is made available to the user on demand then user authentication must be ensured before allowing the data access.

Over the years, password based user authentication and two factor user authentications have been proposed by a number of researchers. Two-factor user authentication allows for a separation of roles at the expense of adding a multitude of implementation and deployment issues, which make them expensive. The two-factor user authentication assumes that WSNs are deployed in a confined area. The basic idea of the scheme is that during registration phase, a user receives a smart card from GW node. Then during login-authentication phase, the user can login to the sensor and access data with the aid of the user's password and smart card. In this case, the user must insert the smart card into the specific terminal to be able to login. However in many applications ad hoc topology of WSN is deployed in uncontrolled areas where using a specific terminal and smart card may not be feasible and will restrict the user mobility and utility of WSN. Password-based authentication is easy to integrate and at least it does not involve some incompressible extra costs. Therefore, we are motivated to develop a password based authentication scheme that eliminates the need for smart card and allows only the legitimate user to gain access to sensor data.

1.3 Objectives

Sensor nodes in a WSN are battery-powered and have limited communication, computation and storage capabilities. This requires that the security design must be lightweight and efficient regarding both communication and computation overheads. Due to wireless communication between nodes in a WSN, an adversary can eavesdrop the communication messages and launch different types of attacks. The unattended nature of some WSN makes it vulnerable to node compromise attack. The resource and network constraints together with different attacks impose many challenging requirements for the security design in WSNs. A sophisticated security or authentication scheme requires a balance among the requirements and its design must be robust against sensor compromise and different attacks.

All the password-based authentication schemes proposed so far have security weaknesses that make them unsuitable for wireless network. Moreover, all password-based schemes require strict time synchronization, which increases network overhead and makes the scheme vulnerable to replay attack within a certain time interval. The past schemes also suffer from many logged in users with same ID threat. Therefore, our main objective is to develop a robust user authentication method that inherits all the advantages of previous password-based schemes and improves security by resisting gateway bypass attack, replay attack and many logged in user with the same ID threat. In this thesis, we propose a robust user authentication method which ensures that only the legitimate user can get the access to the sensor data. The objectives of this thesis are as follows:

- Identify the security flaws in the existing user authentication schemes.
- Develop a user authentication method that eliminates the identified security flaws.
- Analyze the performance of the developed authentication scheme.
- Model the proposed authentication protocol in SystemC to verify the security claims.

1.4 Main Contributions

The main contributions of this thesis are summarized as follows:

- We identify the security vulnerability of some of the existing user authentication schemes in WSN. We have pointed out that the existing schemes suffer from replay, forgery and gateway bypass attacks. Then we propose a password based user authentication scheme that eliminates the identified security flaws of the past schemes. The proposed scheme minimizes computational and communication costs. We achieve our goal by integrating one way hash function and XOR operations in the proposed scheme.

- The proposed user authentication scheme does not rely on the transmission delay (ΔT) to resist replay attack. The scheme incorporates mechanism within the computation to resist replay attack and eliminates the need for strict time synchronization in WSN.
- We also analyze both the performance and security resilience of the proposed scheme. A quantitative delay analysis is given in detail and demonstrates the effectiveness and efficiency of the proposed scheme.
- The modeling of the proposed scheme is done in SystemC.

1.5 Overview of the Thesis

The remainder of the thesis is organized as follows. In Chapter 2, the security issues, threats and models of WSN and related works are studied and investigated. A detail security analysis of some of the representative schemes are also provided in this chapter. In the third chapter, a robust user authentication scheme is proposed. Chapter 4 provides the detail security and performance analysis of the proposed scheme. The security of the scheme is evaluated for replay, forgery, gateway by-pass attack as well as many logged in user threat. The performance of the scheme is evaluated based on the computational overhead and functional requirements. Chapter 5 provides the details of modeling of our protocol in SystemC. A detailed study of the authentication latency of the proposed scheme is also provided in this chapter. Some representative authentication schemes are also modeled and a comparison of the authentication latency among the schemes is provided. The security claims of the proposed scheme are also verified. Chapter 6 concludes the thesis and provides future research direction.

1.6 Conclusion

In this chapter, we have provided an overview of the evolution of wireless sensor networks. The hardware architecture, network architecture, network topology and the prevailing standards and specifications are discussed. Some of the promising applications are discussed and the motivation behind the research work is described. In the next chapter we will study the security issues of wireless sensor networks and review some of the representative schemes for user authentication in WSN.

Chapter 2: Wireless Sensor Network and Authentication

In this chapter we introduce Wireless Sensor Network (WSN). Then we discuss the security challenges faced in WSN. Different types of attacks in WSN are described in detail. The major factors of user authentication and authentication related works are elaborated in this chapter. A detail Security analysis of some of the representative schemes are presented at the end of this chapter.

2.1 Introduction to WSN

The emerging field of Wireless Sensor Networks [2] combines sensing, computation and communication into a single tiny device. A sensor network can be described as a collection of tiny, intelligent, low-cost and wirelessly connected sensor nodes which coordinate to collect and disseminate physical environmental data from remote locations to one or multiple sink nodes. Sensor networks may be composed of many different types of sensors such as seismic, low sampling rate magnetic, thermal, visual, infrared, acoustic and radar, etc. These sensors are able to monitor a wide variety of ambient conditions such as temperature, humidity, vehicular movement, lightning condition, pressure, noise levels, etc. WSN consists of spatially distributed autonomous sensors to monitor physical or environmental conditions. It is employed in various systems involved in surveillance supporting tracking, monitoring and control in urban/suburban areas, military and/or anti-terrorism operations, telemedicine, assistance of disabled and elderly people, environmental monitoring, localization of services and users, industrial process control, etc. The development of wireless sensor networks has been employed in military applications such as battlefield surveillance. The concept of micro-sensing and wireless connection of these nodes promise many new applications including environmental, medical, military, transportation, entertainment and homeland defence.

2.1.1 Hardware Platform of Sensor

The WSN is made of "nodes" – from a few to several hundreds or even thousands, where each node is connected to one (or sometimes several) sensor. A typical sensor network consists of a large number of multifunctional sensor nodes, which are equipped with sensing, information collecting, processing, and communicating components. Each sensor node can have several components: a radio transceiver with an internal antenna or connection to an external antenna, a microcontroller, internal and external memories, an electronic circuit for interfacing with the sensor and an energy source, usually a battery or an embedded form of energy harvesting. A block diagram of a typical sensor node is shown in Figure 2.1.

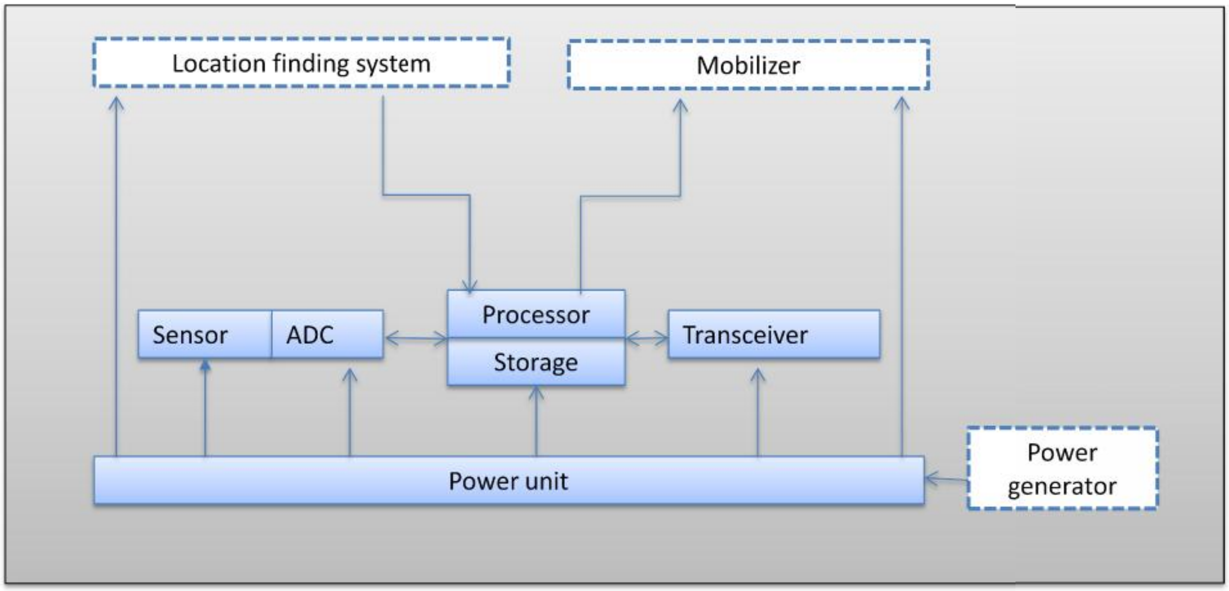


Figure 2.1: A Typical Sensor Node

Transceiver:

The transceiver serves the purpose of connecting the node to the network. It is responsible for providing wireless interface for transmission and reception operations. The nodes can communicate with each other using wireless communication usually at Radio Frequency (RF). A radio propagates signals which contain the data that it is sending through the air. Another radio can receive these signals and retrieve the transmitted information.

Processor and Memory:

The embedded processor that can be used in a sensor node includes Microcontroller, Digital Signal Processor (DSP), Field Programmable Gate Array (FPGA) or Application Specific Integrated Circuit (ASIC). Among all these alternatives, the Microcontroller has been the most used embedded processor for sensor nodes due to its low cost and flexibility to connect to other device parts. Memories in a sensor node include on-chip flash memory and RAM of a microcontroller and external flash memory. The processor operates with a simpler operating system specifically designed for micro-sensors such as TinyOS [63].

Sensor:

A sensor is a hardware device that produces a measurable response signal to a change in a physical condition such as temperature, pressure or humidity. The continual analog signal sensed by the sensor is digitized by an analog-to-digital converter and sent to the embedded processor for further processing.

Power Unit:

Sensor nodes consume power for sensing, communication and processing of data. Usually the sensors are battery powered. However, energy harvesting sensor nodes are also available [45].

2.1.2 Network Structure

A WSN can be infrastructure-based or ad-hoc. In an infrastructure-based WSN, some nodes form a relatively static infrastructure and are responsible for relaying traffic for other nodes. Smart Dust is an example of the network in this mode [21]. The WSN ad hoc networks are decentralized type of wireless networks that are able to organize themselves without predefined infrastructure. The ad-hoc network does not rely on a pre-existing infrastructure, such as routers in wired networks or access points in managed (infrastructure) wireless networks. Each sensor node participates in routing by forwarding data for other nodes, and so identification of the nodes that forward data is made dynamically. The decentralized nature of wireless ad-hoc networks improve the scalability of wireless ad-hoc networks compared to wireless managed networks. Minimal configuration and quick deployment make the ad-hoc networks suitable for emergency situations such as natural disasters or military operations. However, managing the network resources and quality of service provisioning in such networks is more difficult than in an infrastructure-based network. Some WSNs may combine the infrastructure-based and ad-hoc modes, where cluster of sensor nodes are inter-connected via some central access points referred to as Cluster Heads (CHs), and all the data packets collected by the sensor nodes are eventually followed by the CHs to the sinks.

2.1.3 Standards and Specification

The protocol stack used by the sink, cluster head and sensor nodes are shown in Figure 2.2.

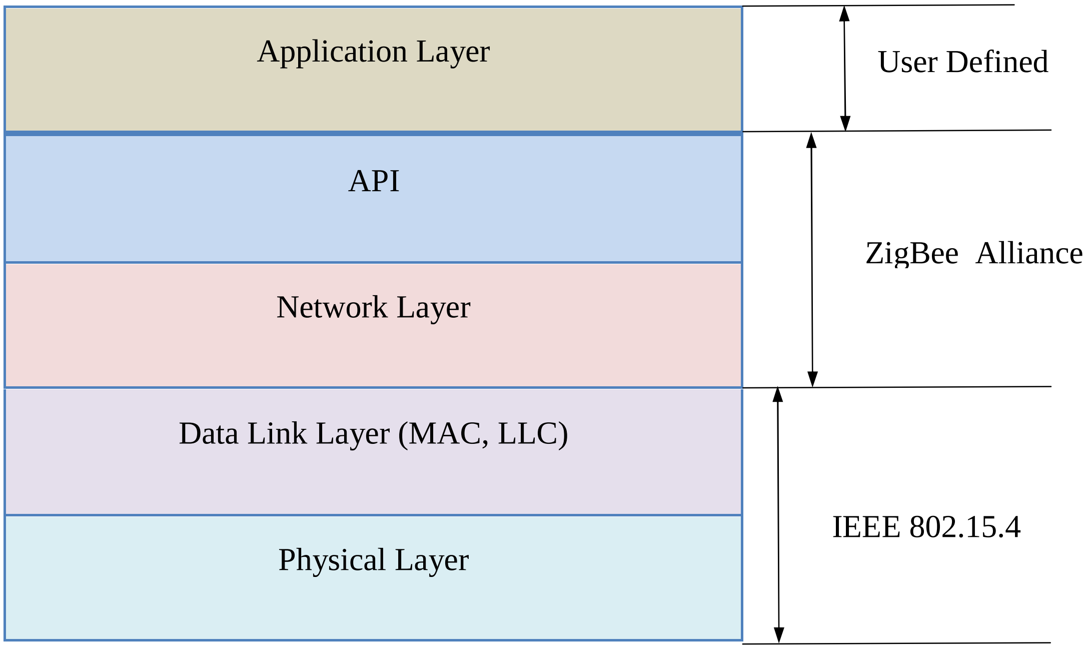


Figure 2.2: Protocol Stack of a Sensor Node

According to Akyildiz *et al.* the sensor network protocol stack is much like the traditional protocol stack, with the following layers: application, transport, network, data link, and physical [3].

Physical Layer:

The physical layer is responsible for frequency selection, carrier frequency generation, signal detection, modulation and data encryption.

Data Link Layer:

The data link layer is responsible for multiplexing data streams, data frame detection, medium access and error control. It ensures reliable point-to-point and point-to-multipoint connections in a communication network.

Network Layer:

The network layer takes care of routing the data supplied by the transport layer.

Transport Layer:

The transportation layer helps to maintain the data flow and may be important if WSNs are planned to be accessed through the Internet or other external networks.

Application Layer:

Depending on the sensing tasks, different types of application software can be set up and executed at the application layer.

IEEE 802.15.4 [65] and ZigBee [69] are among the pre-dominant standards and specifications for wireless sensor networks. IEEE 802.15.4 standard specifies the physical and Medium Access Control (MAC) layers for low-rate, low-power and flexible wireless personal area networks (WPANs). Main features of IEEE 802.15.4 standard are given below:

- Data rates of 250 kbps, 40 kbps and 20 kbps
- Two addressing modes: 16-bit short and 64-bit IEEE addressing
- Support for latency critical devices, such as joysticks
- CSMA-CA channel access

- Automatic network establishment by the coordinator
- Fully handshake protocol for transfer reliability
- Power management to ensure low power consumption
- 16 channels in the 2.4GHz ISM (Industrial, Scientific and Medical) band, 10 channels in the 915MHz ISM band and one channel in the 868MHz band.

ZigBee is a standard for a suite of high level communication protocols based on the IEEE 802.15.4 standard for low power and low data rate radio communications. The ZigBee specification is maintained and published by ZigBee Alliance, which is an association of companies working together to enable low-power, low-cost WPANs. A lot of WSNs currently studied in the literature do not follow the IEEE standards. In particular, TDMA-based WSNs have been reported extensively in the literature [40, 43, 50].

WSNs must also be aware of the management planes in order to function efficiently such as mobility, power, task, quality of service (QoS) and security. Among them, the functions of task, mobility and power management planes have been elaborated by Akyildiz *et al.*[3]. The power management plane is responsible for minimizing power consumption and may turn off the functionality in order to preserve energy. Most of the wireless sensor nodes have limited battery life and restoring batteries on these nodes is sometime almost impossible. Therefore, the lifetime of a sensor node can be the lifetime of a battery. Many protocols and algorithms deployed in WSNs are based on energy efficient and aware [7, 12, 16]. The mobility management plane detects and registers movement of nodes so a data route to the sink is always maintained. The task management plane balances and schedules the sensing tasks assigned to sensing field. Therefore, only the necessary nodes are assigned with sensing tasks and the remainder are able to focus on routing and data aggregation. QoS management in WSNs can be very important if there is a real-time requirement for the data services [18]. Security management is the process of managing, monitoring, and controlling the security related behaviour of a network. The primary function of security management is to control access points for critical or sensitive data. Security management also includes the seamless integration of different security function modules including encryption, authentication and intrusion detection. It is obvious that networking protocols developed for WSNs must address all of these management planes.

2.1.4 Network Topology

The design, development and deployment of wireless sensor networks have taken the traditional network topologies in new directions [12, 20, 55]. ZigBee Alliance defines three network topologies at the top of IEEE 802.15.4 physical and MAC layers that are mesh, star and the cluster-tree topology. Figure 2.3 shows different topologies for a WSN.

In a mesh topology, some of the nodes are connected with more than one adjacent node in the network, and packets determine their path to the destinations according to the routing algorithm. Mesh networks allow data to "hop" from node to node that enables the network to be self-healing. Each node is then able to communicate with each other as data is routed from node to node until it reaches the desired destination. This type of network is one of the most complex networks having a significant cost to deploy properly. Due to its cost and complexity, the mesh topology is usually used for networks with a small number of nodes.

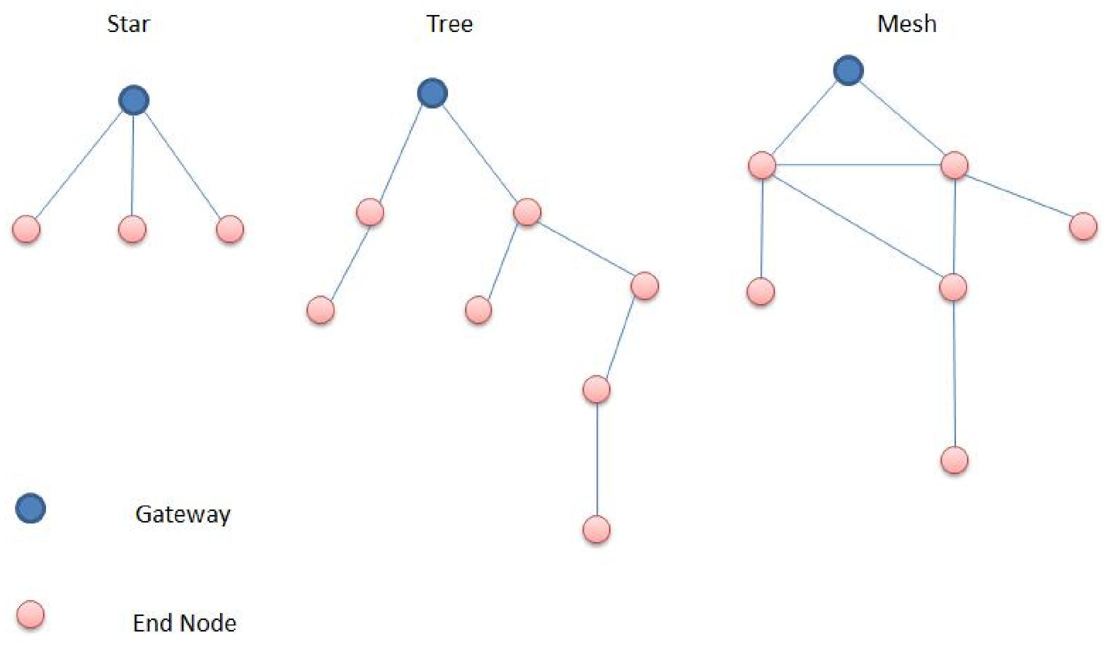


Figure 2.3: Sensor Network Topology

Star networks are connected to a centralized communications hub. Each node cannot communicate directly with one another and all communications is routed through the centralized hub. Each node is then a “client” while the central hub is the “server”. The star topology is easy to design and implement. The failure of each device or connection does not affect the entire network as long as the central nodes function properly.

The cluster-tree topology can be considered that integrates several small star topology networks together. The cluster-tree networks use a central hub (a *Root* node) as the main communications router.

The topology of a WSN affects many network features and qualities such as capacity, latency and scalability. Both the star and cluster-tree topologies can use beacon frames to synchronize the sensor devices to their parent node, and thus minimize power consumption of the devices with intermittent operations.

2.1.5 Applications of WSN

The sensing element embedded in the sensor node can detect ambient conditions such as sound, light, temperature, smell, vibration, etc. Then it transforms the sensed condition into electric signals. By analyzing the signals, one can extract useful information of the location where event happened under the collaborative effort of sensors. The development of wireless sensor networks was motivated by military applications such as battlefield surveillance. Today these networks are used in many industrial and consumer applications such as industrial process monitoring and control, machine health monitoring, and so on. With the advances in processor, memory and wireless communication, WSN has emerged as a promising technology to help human perform many tasks such as environment and habitat monitoring, health-care application, traffic control or wild ecological survey. The application and challenges of WSN are discussed in details elsewhere [1, 3, 17, 49]. Figure 2.4 shows some of the promising applications of WSN.

Figure 2.4: Typical Applications of WSN

2.2 Security Challenges in WSN

Resource constrained WSN nodes have limited processing capability, very low storage capacity and the network has a limited communication bandwidth. Due to these constraints, it is difficult to directly employ the conventional security mechanisms in WSNs. The security challenges of WSN are summarized as follows:

i) Minimizing resources and maximizing security

Energy is the biggest constraint for a WSN. It is used for sensing, communication and computation purposes. Communication is more costly than computation in WSNs. Higher security levels in WSN usually correspond to more energy consumption for cryptographic functions.

ii) Memory limitation

A sensor node has a small storage space for storing application programs, sensor data and intermediate results of computations. There is usually not enough space to run complicated algorithms after loading the OS and application code. Therefore, the security algorithm must have low computation and memory requirement.

iii) Unreliable communication

The broadcast nature of communication in WSN makes the communication unreliable. Packets may get damaged due to channel errors or, congestion in nodes. Sophisticated error handling in wireless communication means increased overhead.

iv) High communication latency

In a wireless sensor network, routing, network congestion and processing in the intermediate nodes may lead to higher latency in packet transmission. Achieving network synchronization becomes very difficult in such a network.

v) Unattended operation of network

Sensor network is deployed in an open, hostile and dynamic environment, which renders more link attacks ranging from passive eavesdropping to active interfering. This makes security in WSNs a particularly difficult task.

vi) Self-organization

The dynamic network topology and node mobility makes the security more complex.

2.3 Attacks in WSN

Wireless Sensor networks are vulnerable to security attacks because they are placed in a hostile environment where the nodes are not physically protected. Due to the broadcast nature of transmission, the nodes can have additional security threat. The small sensor nodes in a WSN are susceptible to many kinds of attacks. Various classifications of these attacks are discussed in the literatures [19, 42].

i) Attack based on the network environment

Depending on the location of attacker in a network, the attacks can be classified into two categories:

- *External Attack*

The attacker node is not an authorized participant of the sensor network. An outside attacker has no access to encrypted communication in a sensor network. External attack is further classified into two categories:

a) Active

The unauthorized attacker monitors, listens to the communication and disrupts network functionality by modifying the data stream. Some well-known active attacks are Routing Attacks [9, 14, 22], Denial of Service attacks [38], physical attack [4, 44], node outage attack [31], message corruption attack [34], jamming attack [9, 56], power exhaustion attack[22], Injecting faulty data into the WSN [31], Impersonating [41, 58], etc.

b) Passive

A passive attack involves unauthorized monitoring and listening to the communication message of the network. Eavesdropping [41, 51] and traffic pattern analysis [61] are passive type of attacks.

- *Internal Attack*

The legitimate user can share his/her user ID and password to another user or the attacker has the access to the verifier table in the gateway node of the network. Inside attacks are much harder to detect and defend against. Unauthorized login or multiple login with same user ID are some of the internal attacks of a WSN.

ii) Attack based on protocol layers

- *Physical layer*

WSNs are vulnerable against different physical attacks, such as jamming [56] and device tampering. Attackers can gain full access to the sensor nodes, extract and reveal sensitive information or launch DoS attacks against the WSN.

- *Medium Access Control*

The MAC protocol is responsible for managing the radio of a sensor node which is the main source of power consumption. The malicious collision attack [34, 58], unintelligent replay attack [22], unauthenticated broadcast attack [22], exhaustion attack [22] and intelligent jamming attack [22] are main MAC layer attacks.

- *Network layer*

By attacking network layer, attackers can absorb network traffic, inject traffic into the path between the source and destination, and thus control the network traffic flow.

Manipulating routing information [4], selective forwarding attack [22], Sybil attack [33], sinkhole (blackhole) attack [61], wormhole attack [4], hello flood attack [4], etc. are known network layer attacks.

- *Transport Layer*

  Similar to TCP protocols in the Internet, the WSN node is vulnerable to the classic SYN (synchronize) flooding attack or session hijacking attacks [29, 30, 57].

- *Application Layer*

  The application layer communication is most vulnerable in terms of security compared to other layers because the information an attacker seeks, ultimately resides within the application. Malicious code attacks [29] and repudiation attacks [4] are some of the known application layer attacks.

2.4 Authentication

Authentication is the ability to identify a system or a network user through the validation of a set of assigned credentials. In the context of WSNs, an authentication can be composed of three branches: Data authentication, Node authentication and User authentication.

2.4.1 Data authentication

In a wireless sensor network, an adversary can easily inject messages. The receiver needs to make sure that the data used in any decision-making process originates from the legitimate source. Data authentication prevents unauthorized parties from participating in the network and legitimate entities should be able to detect messages from unauthorized entities and reject them. Measures for protecting integrity are considered necessary to detect message alteration and to reject injected message. In the symmetric key cryptography, MACs are used to provide authentication. 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 original sender. In the public key cryptography, digital signatures are used to seal a message as a way of authentication. A digital signature is a mathematical scheme for demonstrating the authenticity of a digital message or a document. A valid digital signature gives a recipient reason to believe that the message was created by a known sender, and that it was not altered in transit. Digital signature involves much more computation overhead in signing, decrypting, verifying and encrypting operations than techniques used in symmetric cryptography.

2.4.2 Node Authentication

Authentication is necessary not only to data exchange processes but also to network administrative tasks in WSNs like the addition of new node to the networks. Several researchers have focused on node authentication before the nodes join the WSN such as protocol described by Manivannan *et al*. [27]. This protocol is based on congruence equations and number theory concepts to achieve secure authentication among nodes in WSNs. Most research projects on the node authentication and key distribution assume WSN as a static environment. Therefore, they only focus on the efficient initial authentication and key setup. However, considering the mobility of the nodes in WSN, other schemes have been proposed to take into consideration the mobility of nodes such as scheme presented by Han *et al.* [13]. The distributed node authentication (DNA) scheme uses geographic location and trust relationship among neighbouring sensor nodes to authenticate the identity of sensor nodes [59].

2.4.3 User authentication

User authentication is a mean of identifying the user and verifying that the user is allowed to access some restricted services [64]. User authentication means establishing a relation between the user and some identity. An identity is the individuality property of a user which ideally cannot be forged or copied. In practice, identities are implemented by items which users know (passwords), possess (secret keys or security tokens) or properties which they have (biometrics).

In WSN, access to the collected data will in general not be free since deployment of WSNs induces some costs of deployment. This means that the deployment agencies will make the sensed data available only to certain people, usually those who pay for receiving the service. In this case, a WSN must be able to distinguish legitimate users from the illegitimate ones. In authentication, a user sends his ID (e.g., name, IP address) and proof of his identity to a sensor so that the sensor can decide whether or not the identity is valid and in fact belongs to the user of that name. Upon successful authentication, the sensor authorizes the user who is granted access to the data.

2.4.4 Factors of Authentication

Existing authentication methodologies involve three basic “factors”:

- Something the user knows (e.g., password, PIN, passphrases);
- Something the user has (e.g., keys, badges, ID, tokens, ATM card, smart card); and
- Something the user is (e.g., biometric characteristic, such as DNA, fingerprints, voice match, aura, retinal scan).

2.5 Authentication Overview

Over the years, WSNs have attracted an increasing number of researchers due to its ubiquitous nature, easy deployment and a wide range of applications. Some schemes are suitable for wireless mobile devices and some for low-power devices. The IEEE 802.15.4 standard is the most appropriate communication scheme for low power sensor networks [68]. Sastry and Wagner observed the merits and limitations of the security aspects of IEEE 802.15.4 specification [39]. The specification allows a maximum of 255 Access Control List (ACL) entries, where within the ACL, there is no support for group keying and pair-wise keying. The specification suffers from IV (Initialization Vector) Management, Key management problems and insufficient integrity protection. The IEEE 802.15.4 API indicates two clear directions: (i) to go with the specification itself without adding more security patches, and (ii) to adopt add-on security service on top of the API according to application's requirement. Deploying low-power sensor nodes in an unattended environment makes the networks vulnerable to a variety of potential attacks, the inherent power and memory limitations of sensor nodes make conventional security solutions infeasible.

Password-based authentication schemes are the most widely used methods for remote User Authentication (UA). Existing schemes could be categorized into two types. One uses *weak-password* approach, while the other uses *strong-password* approach. The weak-password authentication approach is based on El Gamal cryptosystem [11]. The advantage of this scheme is that the remote system does not need to keep a user ID-password table to verify the validity of the user login. However, such a weak-password authentication approach leads to heavy computational load on the whole system. Therefore, this scheme cannot be applied to a WSN environment as remote sensor nodes cannot afford to do this heavy computation. Unlike the weak-password approach, strong-password authentication is mostly based on a one-way hash function and exclusive-OR operations (XOR). It requires much less computation and needs only simple operations. With this in mind, this scheme may have advantages when it is applied to a WSN environment.

A number of researchers have focussed on the user authentication schemes suited for WSNs. Benenson *et al.* introduced an *n*-authentication protocol in which the authentication succeeds if the user can successfully authenticate with any subset of *n*-sensors [5]. They also proposed a public key-based user authentication protocol using Elliptic Curve Cryptography (ECC) [6]. This authentication protocol is more reliable for WSNs than TinyPK [53]. Wong *et al.* proposed a strong password-based dynamic user authentication scheme [52]. It imposes light computational load as the protocol requires only one-way Hash function and Exclusive-OR operations. Their scheme comes with several advantages such as it allows legitimate users to query sensor data at any of the sensor nodes in an ad hoc manner. Moreover, it requires simpler computational operations. Tseng *et al.* [46] pointed out three security weaknesses of Wong *et al.'s* [52] scheme and proposed an improved dynamic user authentication scheme that not only fixed the weaknesses but also enhanced the security. Lee [26] also analyzed this scheme and proposed two simple dynamic user authentication methods that are variants of Wong *et al.*'s scheme [52]. Later on, Ko [24] pointed out some security flaws of Tseng *et al.'s* [46] scheme. Ko proposed a novel dynamic user authentication scheme that inherits all the advantages of Tseng *et al.'s* scheme and provides mutual authentication between the users, gateway and the sensor nodes [24]. However, Vaidya *et al.* argued that most of the schemes cannot preserve user anonymity [48]. In order to protect the real identity of the user, pseudonym can be used in WSNs. Random dynamic pseudonym, such as hashing-based ID random pseudonym can be the ideal solution for hiding real identity of the user. They proposed two dynamic user authentication schemes that are variations of the strong-password-based schemes with user privacy. These schemes use one-way hash functions and XOR operations to achieve lower computational and communication overheads. Furthermore, the schemes have not only user privacy but also mutual authentication.

Das presented the two-factor authentication concept for WSN to overcome many logged in users with the same login-id threat and stolen-verifier attack in previous schemes [8]. A two-factor authentication is a concept used to describe an authentication mechanism, where more than one factor (e.g., password and smart card) is required to authenticate the communicating party. Khan and Alghathbar [23] pointed out that Das's scheme is vulnerable to offline password guessing attack, sensor node compromising attack and GW-node by-passing attack. The scheme does not provide mutual authentication between GW-node and sensor node. It has the security threat of insider attack and does not have provision for changing or updating passwords of registered users. They incorporated enhanced security patches that allow the scheme to change or update user password, provide protection against insider attack, overcome the GW-node bypassing attack, and provides mutual authentication between GW-node and sensor node. Vaidya *et al.* [47] pointed out that several security pitfalls remain in both Das's and Khan-Algahathbar's schemes and proposed an improvement that results in high level of robustness and better security. Zhou *et al.* proposed a new dynamic user authentication based on nonce instead of timestamps [60]. However, in this scheme, query is directed to the gateway and it does not allow a user to login directly to a sensor node to retrieve real time data.

2.6 Analysis of Past Authentication Schemes

In this section, we analyze some of the representative authentication schemes from security point of view. These schemes have many advantages. However, we have identified a number of flaws in each of the schemes. Identification of security weaknesses of the representative schemes has led us to develop a more secure user authentication scheme for WSN. Table 2.1 shows the notations used in this section.

2.6.1 Tseng *et al.'s* Scheme

Tseng *et al.* [46] identified that Wong *et al.*'s scheme [52] cannot protect against replay and forgery attacks. The password could be revealed by any of the sensor nodes and the user cannot change his/her password freely. Tseng *et al.* presented a modification of Wong *et al.'s* scheme [52] and claimed that their scheme not only fixed the weaknesses but also enhanced the security.

Tseng *et al.* claimed that their scheme not only retains all advantages of Wong *et al.'s* authentication scheme but also enhances its security. However, Tseng *et al*.'s scheme still has several drawbacks such as Man-in-the-middle attacks, replay attack, stolen verifier attack and forgery attack. The scheme even does not achieve mutual authentication between the gateway and the sensor node, and between the user device and the sensor node.

Table 2.1: Notations Used in the Past Authentication Schemes

$\oplus$	Bit-wise Exclusive-OR (XOR) operation
$\|\|$	Bit-wise Concatenation
ACC_LOGIN	Accept Login message
GW	Registration Sensor Gateway
H(*d*)	Hash function of *d*
N	Random nonce
PW	Password chosen by the user
LN	Login node
Succ_Change	Successful Change message
Succ_Reg	Successful Registration message
t, T_i , T^*	Current time recorded by one of the nodes
TS	Timestamp for a particular user
ΔT	Time interval between sending and receiving a message
TID	Temporary User ID
UD	User's device such as PDA, etc.
UID	User's identity
x	Secret key known to the GW
X_s	Key generated by GW and known to user and sensor node

i) Replay attack within ΔT

- *The eavesdropper intercepts (UID, C, T, t) in the login phase*
- *The eavesdropper replays (UID, C, T, t)* to GW.
- Gateway computes , *A* and $C^*=H(A\oplus T^*)$
- As long as $(T - T^*) < \Delta T$ is valid, *C* will be same as *C** and the replay attack is successful.

ii) *Man In The Middle Attack*

- The eavesdropper intercepts (UID, A, t) and (UID, C, T, t) in the login phase.
- The eavesdropper computes $C^{*}=H(A \oplus T^{*})$.
- UID, C^{*}, T^{*}, t is forwarded to GW.
- Gateway computes A and $C'^{*}=H(A \oplus T^{*})$.
- As long as $(T - T^{*}) < \Delta T$ is valid, C^{*} will be same as C'^{*} and authentication will be successful.

iii) *Gateway bypass attack*

While transmitting ACC_LOGIN from GW to LN, the eavesdropper intercepts the message. Afterward, when a legitimate user sends login message *(UID, C, T, t)* to the *GW*, the eavesdropper can block ACC_LOGIN message by jamming attack and replay previously intercepted ACC_LOGIN message to the legitimate LN as pretending legal GW. Since LN does not check the correctness, it will also send ACC_LOGIN to UD. UD will accept ACC_LOGIN as it also does not check the correctness.

iv) *Replay attacks on* **ACC_LOGIN** *message from LN to UD*

While transmitting ACC_LOGIN from LN to UD, the eavesdropper intercepts the message. The eavesdropper can replay this message any time to a legitimate UD as pretending legal LN. Since UD does not check the correctness, it will accept ACC_LOGIN.

v) *Node capture attack*

The adversary first pretends to be a legitimate *LN*, allowing any users to send query to her. After receiving login message from the *UD*, the LN computes the computes *C*=*H(A*$\oplus$T*)* as the legitimate *SN* does and sends *(UID*, *C*, *T*, *t)* to the *GW*. Consequently, the checking process in the *GW* will be passed and the *GW* does not notice that the message is sent from the adversary. Then, the *GW* sends ACC_LOGIN message to the adversary, and the adversary sends ACC_LOGIN message to the *UD*, allowing *UD* to access the bogus sensor readings.

vi) No mutual authentication

Tseng *et al.'s* scheme does not achieve mutual authentication between the *GW* and *LN*. Therefore, the adversary can replay ACC_LOGIN message between GW and LN. Likewise, the scheme does not provide mutual authentication between the *UD* and the *LN*, which makes the scheme vulnerable to replay attack between LN and UD.

vii) Multiple Login with same ID

The scheme suffers from multiple login with same login ID threats. The legitimate user can disclose his user ID and password to an unauthorized user in the network. Since the GW does not have any protection against the multiple login using same user ID and password, the login will be successful.

viii) Time synchronization

The security of the scheme relies on strict time synchronization in the network. However, achieving time synchronization is a difficult task in an ad-hoc WSN network.

2.6.2 Ko*'s* Scheme

Ko [24] proposed an improved scheme by modifying the Tseng *et al.'s* technique. In addition to all the advantages of Tseng *et al.'s* method, Ko*'s* novel scheme provides additional security strength. However Ko's scheme suffers from multiple login attack with same ID and stolen verifier attack. The scheme has a heavy computational overhead for the sensor node and makes the scheme practically infeasible.

i) Multiple login attack with same ID

- The legitimate user uses (UID, A, t_1) to login the sensor node.
- The legitimate user shares his UID and password with a user who is not registered to GW.
- The unauthorized user uses (UID, A, t_2) to login the sensor node.
- UID is registered in the GW but (UID, t_2) is not in the database. So, the login by the unauthorized user will be successful and the scheme suffers from multiple login attack with same login ID threat.

ii) Stolen verifier attack with node capture attacks

- A mercenary breaks LN to get (UID, N, TS).
- The mercenary steals $h(PW)$ for UID from GW.
- Computes $h(x \oplus UID)=N \oplus h(PW)$.
- During the password change phase, $h(PW)$ is changed to $h(PW_1)$ for UID.
- The mercenary again breaks LN to get (UID, N', TS').
- Compute $h(PW')=N' \oplus h(x \oplus UID)$.

- Since the new password and legitimate *UID* is known, it can generate $A_e=h(h(PW')\oplus t_{1e})$.
- The mercenary sends login message (UID, A_e , t_{1e}) to LN.
- As long as validity of message is within allowed time interval, this kind of attack will be successful.

iii) Time synchronization

The successful communication flow of Ko's scheme depends on strict network synchronization. Without network synchronization, even the login attempt by the legitimate user fails. However achieving network synchronization in a dynamic network is a very difficult task and requires a large network overhead.

iv) Computational cost on Sensor node

The scheme uses too many bit wise XOR and Hash operations in sensor node. This produces a computational burden on the resource constrained sensor node.

2.6.3 Vaidya *et al.'s* Authentication Methodology

Vaidya *et al*. argued that most of the representative schemes cannot preserve user anonymity [48]. In order to protect the real identity of the user, pseudonym can be used in WSNs. Random dynamic pseudonym, such as hashing-based ID random pseudonym can be the ideal solution for hiding real identity of the user. They proposed two dynamic user authentication schemes that are variations of the strong-password-based schemes. As compared to their first scheme, the second scheme has advantage of providing resistance to the attack of an intruder impersonating the GW to grant access right to illegitimate users. The schemes are composed of four phases: the registration phase, the login phase, the authentication phase, and the password change phase.

Although Vaidya *et al.'s* scheme [48] provides user privacy, mutual authentication and light computation cost, it suffers from replay attack, node compromise attack, gateway bypass attack and multiple login with same ID threat. The scheme does not provide mutual authentication between UD and GW.

i) Replay attack within ΔT interval:

An adversary can cheat the login node to send the Acc_Login message to user by replaying and blocking some messages.

- The adversary eavesdrop the login message (*TID*, *A*, *t*) in step L2 and (*Acc_Login*, V_M, T_1) in step A2.
- Replays the message (*TID*, *A*, *t*) to login node within time interval ΔT. LN checks that TID is a valid user.
- The adversary blocks the message that is sent from the *SN* to the *GW* in step L4 preventing the *GW* from receiving this message.
- The adversary replays (*Acc_Login*, V_M, T_1) message in step A2 to the *SN*.
- LN receives message at T_2 and computes V'_M
- If T_2-$T_1 \leq \Delta T$, LN computes that $V_M = V'_M$
- The *LN* sends *Acc_Login* message to the adversary, allowing adversary to access its data. Therefore, scheme is vulnerable to replay attack within *ΔT* interval.

ii) *Gateway bypass attack due to node capture attack*

Typically, WSN are deployed in an unattended and hostile environment. One could easily capture a node and try to collect some secret information from it about the network. Implementation of one-time sensors can prevent this attack, but it is limited to some applications such as fire alarm, where confidentiality of the transmitted data is not required or important. When confidentiality of data is a concern, it is a difficult task to prevent this attack if sensor nodes are not tamper-proof and the environment is unattended. The GW-node, however, can monitor periodically whether any node is captured or not. If user authentication and data access from node are allowed to the user directly without GW node's notice then the impact of "node compromise" attack is very high. Vaidya *et al.*'s scheme [48] suffers from node compromise attack in the following way:

- Assume that UD_i sends the login message (TID, A, t) to the captured login node.
- The adversary knows (TID, X, TS)
- Adversary computes the following:

  Compute $V'_M = H(X||A||T_{1e})$

  Compute $Y_K = H(V'_M\ ||T_{2e})$
- Adversary sends (Acc_login, Y_K, T_{1e}, T_{2e}) to UD
- UD_i computes the following:

  Compute $V''_M=H(X||A||T_{1e})$

  Compute $Y'_K=H(V''_M\ ||\ T_{2e})$
- UD_i verifies that $Y_K = Y'_K$
- UD_i login to the compromised node and receives fake data.

iii) Many logged in user with same ID threat

The scheme is vulnerable to *many* logged in users with the *same* login-id threat. If a valid user shares his *TID* and password with second user, then the second user can generate the login message (*TID, A*, t_m) and sends to login node. Login node sends (*TID*, C_K, T_0, t_m) to GW. The GW checks that *TID* is a valid user and (*TID*, t_m) are not in the database. So the gateway will allow the login.

iv) No mutual authentication between UD_i and GW

The scheme can provide mutual authentication between GW and LN. The LN gives C_K and the GW gives *X* during login phase and during registration phase, respectively. Therefore, LN and the GW can use *X* and C_K, respectively, to realize mutual authentication. UD_i is authenticated to LN and GW by *TID* and *A*. LN uses *X* to authenticate itself to the UD_i. However, GW is not authenticated to UD_i . As a result GW bypass attack becomes possible in Vaidya *et al.*'s scheme.

v) Stolen verifier attack:

The server stores verifiers of users' passwords instead of the clear text of pass-words. In the stolen-verifier attack, the adversary who has stolen the password-verifier from the server uses it directly to masquerade as a legitimate user. The scheme suffers from stolen-verifier attack in the following way:

- Steal *vpw* and *TID* from GW.
- Compute $A=H(vpw||t)$.

- Send *TID, A, t* as login message to LN as pretending legal UD and the attack will be successful.

2.7 Conclusion

In this chapter we have presented various challenges and different types of attacks in WSN. An inside view of the requirements of authentication in context of WSN is provided. The chapter elaborates the need for a robust user authentication scheme. A detail discussion of related research works has been provided and finally we have provided a detail cryptographic analysis of some of the representative schemes. In the next chapter we will propose a robust user authentication scheme.

Chapter 3: Proposed Authentication Scheme

3.1 Introduction:

In most of the applications, nodes in a wireless sensor network (WSN) are deployed in an open and hostile environment. The random deployment and unattended nature make the WSN prone to attack by adversaries. Therefore, the need of ensuring security while communicating in the presence of such attacks is of utmost importance. However, traditional security measures designed for the resource-rich networks such as LAN, WAN, MAN etc. are not suitable for a resource-constrained WSN network. These constraints arise out of limitation in computational power of the nodes, limitation in communication over the open medium and limitation in deployment strategies. In the presence of such limitations, it becomes mandatory to devise lightweight security solutions for wireless sensor networks (WSNs). In this chapter, we provide a detail description of the security goal and model of an authentication scheme. Our proposed authentication scheme is also described in this chapter.

3.2 Security Goals

Wireless sensor networks are vulnerable to many attacks due to the broadcast nature of transmission medium, resource limitation of sensor nodes and uncontrolled environments where they are left unattended. When dealing with security in WSNs, we mainly focus on the problem of achieving the following contributions or services [28].

- ***Confidentiality***: Confidentiality is the ability to conceal messages from a passive attacker so that any message communicated via the sensor network remains confidential. Confidentiality or secrecy protects secret data and makes information inaccessible to unauthorized users or entities. In many applications, sensor nodes accommodate and communicate highly sensitive data. Especially, in military applications the data stored in the sensor node may be highly sensitive. A sensor network must not leak sensitive sensor data to its neighbours. This is the most important issue in network security.

- ***Availability***: Availability ensures the survivability of network services to authorized parties when needed despite denial-of-service attacks. Availability determines whether a node has the ability to use the resources and whether the network is available for the messages to communicate. However, failure of the base station's availability will eventually threaten the entire sensor network. Therefore, availability is of primary importance for maintaining an operational network.
- ***Integrity***: Integrity measures ensure that the received message has not been altered in transit by malicious nodes or adversaries. Data integrity in sensor networks is needed to ensure the reliability of data and refers to the ability to confirm that a message has not been tampered with, altered or changed. Even if the network has confidentiality measures, there is still a possibility that the data integrity has been compromised by alterations. The integrity of the network will be in trouble when a malicious node present in the network injects the false data.
- ***Authentication***: Authentication enables a node to ensure the identity of the peer node with which it is communicating. Data authentication verifies the identity of the senders and receivers. It is achieved through symmetric or asymmetric mechanisms where sending and receiving nodes share secret keys. Due to the wireless nature of the media and unattended sensor networks, it is extremely challenging to ensure authentication.

- ***Authorization***: Authorization ensures that only authorized nodes can access the network services or resources and only the authorized entities are able to perform certain operations in the network (e.g. information providing, system controlling, etc.).
- ***Freshness***: Data freshness implies that each data is recent, and it ensures that no adversary will replay old messages. Since all sensor nodes provide some forms of time varying measurements, we must ensure each message is fresh. To solve this problem a nonce, or another time related counter, can be added into the packet to ensure data freshness.
- ***Scalability:*** The sensor nodes are becoming cheaper day by day. The scalability ensures that message can be conveyed to the network entities with the desired fidelity as the number of nodes grows without bound.
- ***Efficiency:*** The main network entities i.e. sensor nodes are constrained in terms of computational capabilities, memory, communication bandwidth and battery power. As a result, it is challenging to implement and use the cryptographic algorithms and protocols required for security services. The storage requirement, computation and communication limitations on sensor nodes must be taken into consideration while designing a scheme for the network. Therefore, the size for all the security related code must also be concise and small.
- ***Privacy and Anonymity***: These security properties are very important in those cases where the location and identity of the base station or the sensor nodes providing information data is hidden or protected. For example, any network that monitors endangered species should provide no clues on their physical location. This property can transcend beyond the technological dimensions and affect their social environment as sensor networks could be used as a surveillance tool to collect data about the behaviour of human beings.

3.3 Security model

Defining a security model requires the *security requirements* and the *threat model*. The security requirements identify the properties that have to be enforced in the scheme. The threat model formulates the hypothesis regarding the attacker's capabilities and its possible behaviour.

3.3.1 Security requirements

In designing our user authentication scheme, we focus on enforcing two fundamental security properties. Those are:

i) Confidentiality of message, and

ii) Mutual authentication among communicating entities.

Base stations in WSNs are usually regarded as trustworthy. Most research studies focus on secure routing between the sensor nodes and the base station. We assume that messages are encrypted using symmetric key algorithms and consequently, the confidentiality of messages translates into confidentiality of the secret keys. Moreover, the main requirement of our authentication scheme requires that the algorithms and protocols should be adaptive. In other words, the security level guaranteed within the system should be dynamically and efficiently tuneable in order to meet the dynamic security requirements. This is particularly important in WSNs, since higher security generally means higher energy consumption. It is therefore desirable to select the level of security on the basis of actual current threat and the sensitivity of information in the network.

3.3.2 Threat model

A distinguishing feature of WSNs is that sensors may be unattended. A common assumption is that the attacker is compliant with the Dolev-Yao model [10]. It means that the attacker can perform the following actions:

i) Intercept and learn any message;

ii) Introduce forged messages into the system using all the available information; and

iii) The attacker can capture a sensor; and

iv) Once a node is compromised, the attacker is capable of stealing the key materials contained within that node.

To cope with the latter action, it could be possible to assume that sensors are tamper-proof as discussed by Anderson *et al.* and Xu *et al.*[4, 35]. However, a very large number of sensors can be built only if sensors are low-cost devices that make it difficult for manufacturers to make them tamper-proof. Therefore, we will assume that sensors do not have tamper-proof components and that they can be captured.

3.4 Network Set-up

The proposed authentication scheme assumes the WSN setup as given in Figure 3.1. The randomly distributed sensor nodes sense the environment and communicate the information gathered from the monitored field through wireless links. For the simplicity of our scheme, we have combined the gateway and base station into one entity and named as gateway. The gateway is responsible for communication with other networks. The base station is the master radio in the wireless sensor network and serves as the interface between the gateway and the wireless sensors. It holds information for the routes to all the sensors in the network and is responsible for polling the sensors. The data in the sensor node is forwarded via multiple hops to the gateway that can use it locally or communicate it to other networks. The base station communicates with all of the wireless sensors in the network. Any sensor can serve as an RF repeater to communicate with other wireless sensors. The nodes can be stationary or moving. Authorized users can access the WSN anywhere in the network using mobile devices. The mobile device is assumed to have the ability to communicate with any sensor node, SN.

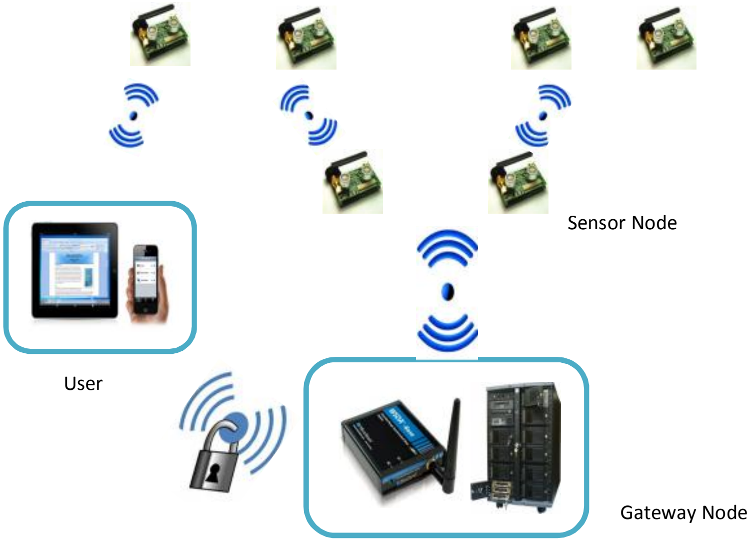


Figure 3.1: Network Setup for the Proposed Authentication Scheme

3.5 Notations

The notations used in this thesis to present our user authentication scheme are given in Table 3.1.

Table 3.1: Notations Used in the Proposed Scheme

$\oplus$	Bit-wise Exclusive-OR (XOR) operation
$\|$	Bit-wise Concatenation
A	Access request message computed by UD
ACC_LOGIN	Accept Login message
C_k, C'_k	Confirmation request message in SN and GW respectively
GW	Registration Sensor Gateway
H(*d*)	Hash function of *d*
N	Random nonce
PW	Password chosen by the user
LN	Login node
Succ_Change	Successful Change message
Succ_Reg	Successful Registration message
t, T_i , $T*$	Current time recorded by one of the nodes
TS	Timestamp for a particular user
ΔT	Time interval between sending and receiving a message
TID	Temporary User ID
UD	User’s device such as PDA, etc.
UID	User’s identity
vpw	Virtual password (SHA-4 of user password)
V_M, V'_M	Validation message computed by GW and LN
x_s	Secret key known to the GW
X_s	Key generated by GW and known to user and sensor node
Y_K, Y'_K	Authentication message computed by LN and UD

3.6 Assumptions

We assume the following working hypothesis:

- The cryptographic primitives that are employed in our scheme are computationally secure.
- The algorithms, protocols and mechanisms that are employed to secure the WSN are publicly known. Only keys in the sensors and gateway are secrets.
- The users can access the network using some type of mobile device that includes a ZigBee interface.
- The users are mobile, but during a particular querying process they have to remain in place.
- The communication between UD and GW is secure.
- The verifier table stored in GW is secure.
- The hash algorithm is either pre-deployed or the participating nodes agree on hash algorithm during communication.

3.7 Phases of the Proposed Scheme

The proposed scheme is a variant of the strong password-based authentication method. Our scheme is composed of four phases: the registration, login, authentication and the password change phase. Although authentication is a continuation of login phase, we define it separately to emphasise its importance. Before issuing any queries, a user must register with a name and a password at the gateway (GW) node. Upon successful registration, the user can submit a query to a sensor node (SN) any time within a predefined or administrative configurable period. The allocation of the secret key to user and sensor node is dynamic and performed during the registration phase. Every time a user registers with the GW node, the key is refreshed. A flag is maintained for each user in the GW to prevent multiple login.

3.7.1 Registration Phase

There are seven steps of the registration phase R1-R7 as given below:

R1: A registration interface is launched by a user's mobile device (UD), and a user inputs his/her ID (*UID*) and a chosen password *PW.* Although we assume that the connection between the UD and GW is secure, it is not feasible to send the user password to GW in plain text to avoid stolen verifier attack. Therefore, the user computes the Hash of *PW* and 512 bit output is stored as *vpw*.

UD: *Compute vpw=H(PW).*

R2: UD submits its identity *UID* and *vpw* to GW node in a secure way.

UD⇒GW: *UID, vpw*

R3: The GW node has the pre-installed secret key x_s stored in it and only an authorized system administrator has the permission to access the GW data. The GW generates a random nonce N_0 and computes *TID* and X_s. Every time a user registers to GW, a fresh nonce ensures a fresh Temporary ID (TID). The secret key *Xs* computed by GW is also refreshed. Then GW stores (*TID, vpw,* X_s *and TS*) in the user database. *TS* represents the timestamp that the gateway recorded when a user was doing the registration. A one bit flag is associated with each *TID* to keep the track of login. The flag value for the user is set to zero.

GW: *Compute* $g = H(UID)$

Compute $TID = g \oplus N_0$

Compute$X_s = H(TID||x_s)$

Store TID, vpw, X_s , TS

R4: GW replies the user for a successful registration with N_0 and X_s.

$GW \Rightarrow UD$: $Succ\_Reg(N_0,\ X_s)$

R5: Upon receiving N_0 and X_s from GW, UD computes *TID* and stores *TID* and X_s for future use.

UD: *Compute* $g=H(UID)$

Compute $TID=g \oplus N_0$

Store TID, X_s

R6: GW distributes (TID, X_s , TS) to those sensor nodes that are able to provide login interface to the user.

$GW \Rightarrow SNs$: TID, X_s , TS

R7: A small database is maintained in the sensor node to store (TID, X_s, TS) for the user.

SN: *Store* TID, X_s, TS

The overall communication flow of the registration phase is shown in figure 3.2.

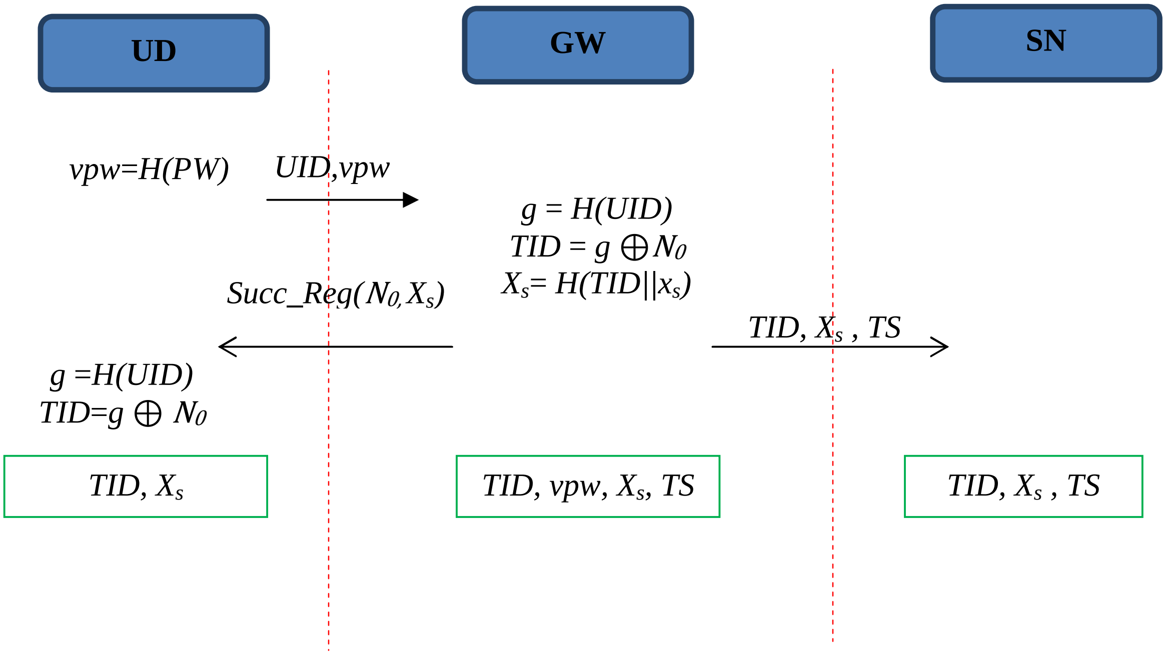


Figure 3.2: Communication Flow of Registration Phase

3.7.2 Login Phase

If the user wants to do some queries of sensory information, he/she needs to login to a dedicated sensor node. The main steps of login phase are as follows:

L1: At time *t*, the user initiates the login phase and computes Access request message *A*, which is the hash of *vpw* in concatenation with time *t*. The bitwise concatenation provides data freshness, prevents replay attack and provides mutual authentication between user and the gateway.

UD: *Compute* $A=H(vpw||t)$

L2: The user submits *(TID, A, t)* to a sensor node.

UD⇒*SN*: *TID, A, t*

L3: Upon receiving the login request at time T_0, the sensor node checks its database to verify the validity of *TID*. If *TID* is not valid, the login request is rejected. Otherwise, the sensor node retrieves the corresponding value of *A* and computes C_k as follows:

SN: *Check TID*

Compute $C_k=H(X_s \oplus A \oplus T_0)$

L4: Sensor node sends (TID, C_k, T_0, t) to the GW.

$SN \Rightarrow GW$: TID, C_k, T_0, t

The communication flow of the login phase is shown in figure 3.3.

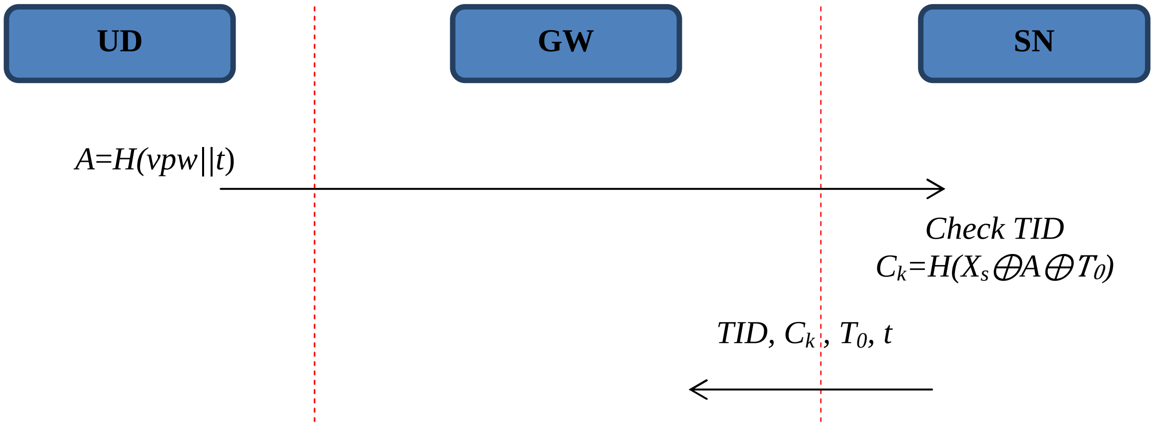


Figure 3.3: Communication Flow of Login Phase

3.7.3 Authentication Phase

At time T_1, GW receives (*TID*, C_k, T_0 and *t*) from the sensor node. The steps of the authentication phase (A1 - A5) are as following:

A1: GW checks the database whether *TID* is a valid user or not. If *TID* is not valid then login is declined. Otherwise, GW checks whether time *t* is recorded in the database. If *t* is already recorded, login is rejected to prevent a replay attack. If not, GW checks the flag associated with the *TID*. If the flag is set, login is rejected due to multiple login attempts. Otherwise, GW retrieves *vpw* from the database and computes A' and C'_k. GW verifies ($C_k = C'_k$) to authenticate the sensor node and the user. Otherwise, a reject message is sent to the sensor node. GW computes V_s and V_u. Time *t* is recorded in the database, and the flag associated with the *TID* is set to one.

GW: *Check TID, t, flag*

Compute $A'=H(vpw||t)$

Compute $C'_k=H(X_s \oplus A' \oplus T_0)$

Verify $C_k = C'_k$

Compute $V_s=H(X_s||A'||T_0||T_1)$ and $V_u= H(vpw||A')$

Store time t

A2: GW sends the accept message $(Acc\_Login, V_s, V_u, T_1)$ to the sensor node.

$GW \Rightarrow SN$: Acc_Login, V_s, V_u, T_1

A3: At time T_2, the sensor node receives message from the GW, computes V'_s and verifies ($V_s=V'_s$) to authenticate the GW node and the user. Sensor node then computes $Y_u=H(V_u||X_s||T_2)$.

Compute $V'_s=H(X_s||A||T_0||T_1)$

Verify $V_s=V'_s$

Compute $Y_u=H(V_u||X_s||T_2)$

A4: The sensor node sends *(Acc_Login*, Y_u, T_2) to the UD.

$SN \Rightarrow UD$: *Acc_login*, Y_u, T_2

A5: Upon receiving the message at time T_3, the user device retrieves the corresponding *A*, performs $V'_u=H(vpw||A)$ and $Y'_u=H(V'_u||X_s||T_2)$, and checks if $Y_u=Y'_u$. If it is true, sensor node and GW node is authenticated and UD starts obtaining data. Otherwise, *Acc_login* message is rejected.

Compute $V'_u=H(vpw||A)$

Compute $Y'_u=H(V'_u||X_s||T_2)$

Verify $Y_u=Y'_u$

The communication flow of the authentication phase is shown in Figure 3.4.

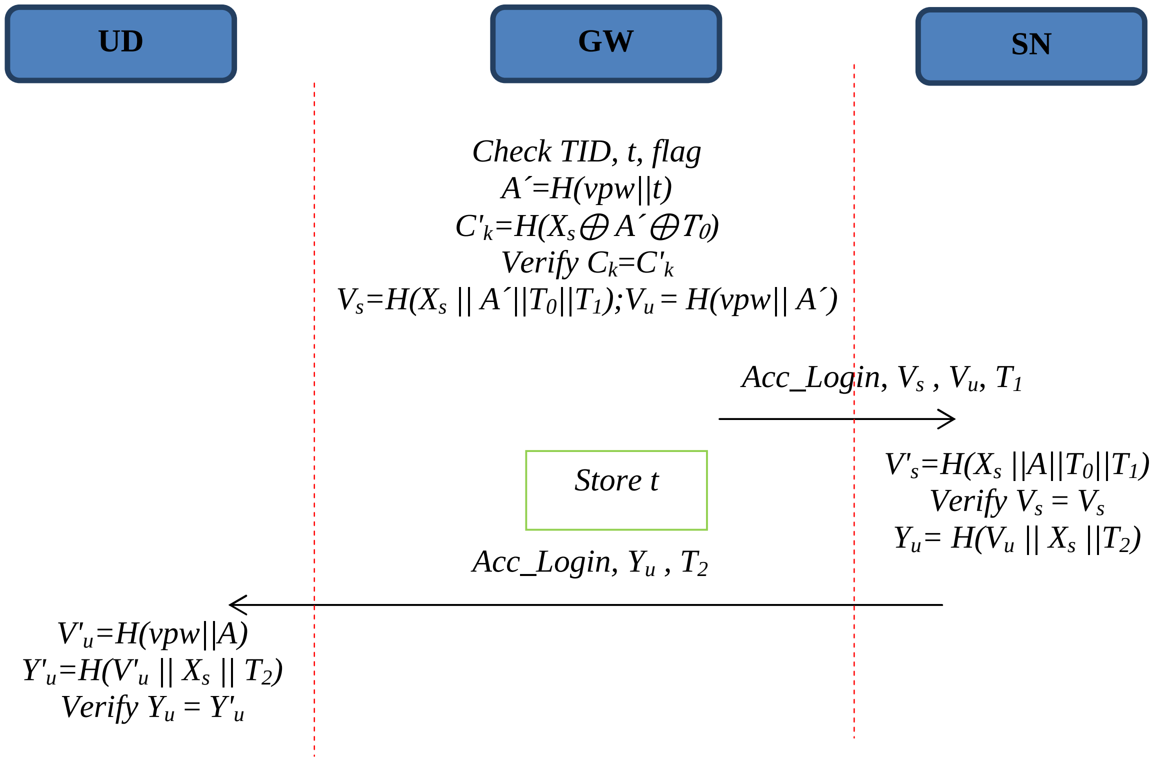

UD
GW
SN
Check TID, t, $flag$
$A´=H(vpw||t)$
$C'_k=H(X_s \oplus A´ \oplus T_0)$
Verify $C_k=C'_k$
$V_s=H(X_s || A´||T_0||T_1);V_u = H(vpw|| A´)$
$Acc\_Login, V_s , V_u, T_1$
Store t
$V'_s=H(X_s ||A||T_0||T_1)$
Verify $V_s = V_s$
$Y_u= H(V_u || X_s ||T_2)$
Acc_Login, Y_u , T_2
$V'_u=H(vpw||A)$
$Y'_u=H(V'_u || X_s || T_2)$
Verify $Y_u = Y'_u$

Figure 3.4: Communication Flow of Authentication Phase

3.7.4 Password change phase

The main steps of password change phase (P1 – P7) are as follows:

P1: UD chooses the new password $PW_1$and computes vpw_1which is the hash of PW_1.

UD: Compute $vpw_1=H(PW_1)$

P2: UD sends the triplet (TID, vpw, vpw_1) to GW via a secure channel.

$UD \Rightarrow GW$: TID, vpw, vpw_1

P3: After verification of *TID* and *vpw*, GW generates nonce $N_1$and computes TID_1, X_{1s} *and* TID'_1 as shown below. GW updates *TID*, *vpw*, X_s and *TS.*

GW: *Generate* N_1

Compute $TID_1 = g \oplus N_1$

Compute $X_{1s} = H(TID_1 || x_s)$

Compute $TID'_1 = TID_1 \oplus X_s$

Update TID, vpw X_s, *TS*

P4: GW sends success change, *Succ_Change* (N_1, X_{1s}) to the UD.

$GW \Rightarrow UD$: *Succ_Change*(N_1, X_{1s})

P5: UD computes TID_1 and updates *TID* and X_s.

UD: *Compute* $g = H(UID)$

Compute $TID_1 = g \oplus N_1$

Update TID, X_s

P6: The GW distributes $(TID, TID'_1, X_{1s}, TS_1)$ to all the sensor nodes.

$GW \Rightarrow SNs$: $TID, TID'_1, X_{1s}, TS_1$

P7: Upon receiving updates, sensor node checks *TID* and computes TID_1. Sensor node updates *TID*, X_s and *TS*.

Verify TID

SN: *Obtain* $TID_1 = TID'_1 \oplus X_s$

Update TID, X_s, *TS*

The communication flow of the password change phase is shown in Figure 3.5.

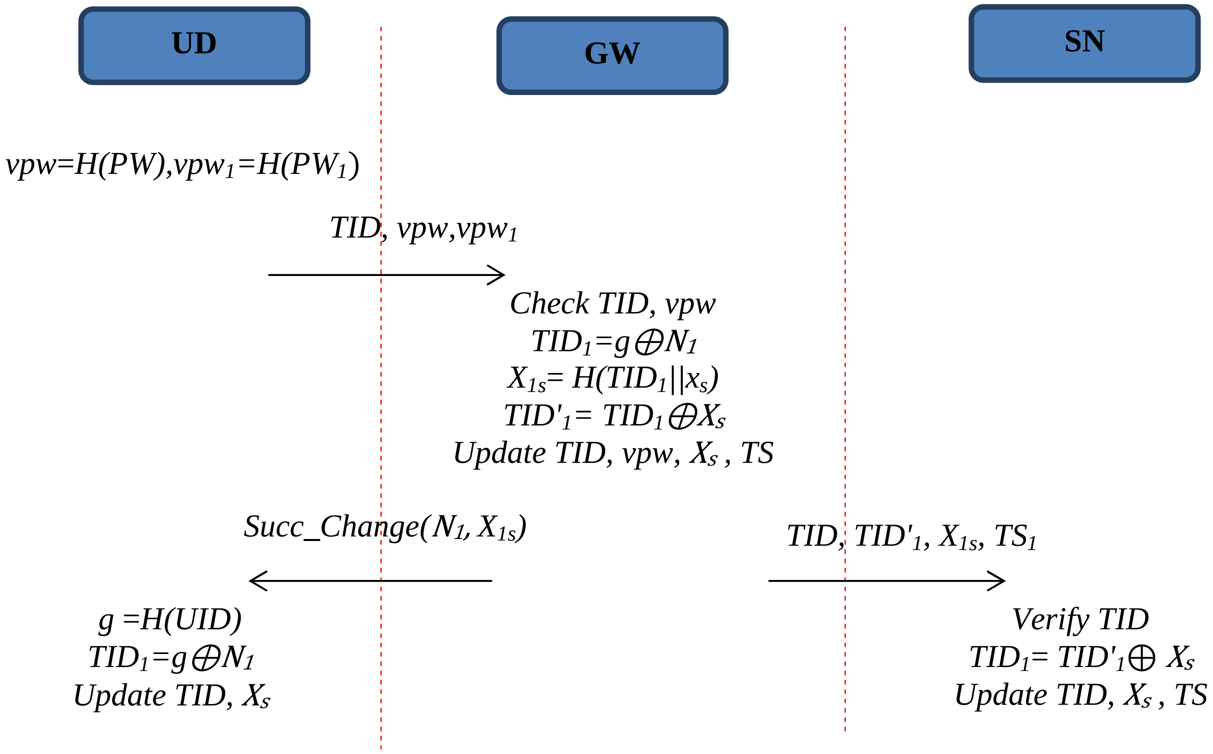

UD
GW
SN
$vpw=H(PW), vpw_1=H(PW_1)$
TID, vpw, vpw_1
Check TID, vpw
$TID_1=g\oplus N_1$
$X_{1s}= H(TID_1||x_s)$
$TID'_1= TID_1\oplus X_s$
Update TID, vpw, X_s , TS
$Succ\_Change(N_1, X_{1s})$
$TID, TID'_1, X_{1s}, TS_1$
$g =H(UID)$
$TID_1=g\oplus N_1$
Update TID, X_s
Verify TID
$TID_1= TID'_1\oplus X_s$
Update TID, X_s , TS

Figure 3.5: Communication Flow of Password Change Phase

3.8 Conclusion

In this chapter we have discussed the general security goal of an authentication scheme and then the security model of our proposed scheme is presented. In the security model, we have pointed out the security requirement for the proposed scheme. The threat model elaborates the capability of an attacker. While developing the scheme, we have assigned some security goals that we want to achieve. In achieving the goals, we have made some reasonable assumptions in our scheme. Finally, we have proposed a robust password-based user authentication scheme. In the next chapter we analyze our scheme on the basis of security and performances.

Chapter 4: Analysis of the Proposed Scheme

4.1 Introduction

We have proposed a robust user authentication scheme for WSN in the last chapter. Our scheme provides application layer security for WSN system. The success of the scheme depends on how it provides effective security and meets the functional requirements. In this chapter, we present the analysis of the proposed scheme from the perspective of security and performance.

4.2 Security Analysis

To analyze the security of our authentication method, we assume that the adversary has the ability to replay, block or forge any network traffic. We also assume that it is computationally infeasible to break the underlying cryptographic function (e.g. cipher). The proposed scheme has most of the security features of different authentication schemes such as user anonymity and resistance to password guessing, impersonation and replay attack. Moreover, our scheme incorporates mechanism to remove the flaws identified in different representative schemes in chapter 2.

4.2.1 Replay Attack

A replay attack is a breach of security in which information is stored without authorization and then retransmitted to trick the receiver into unauthorized operations such as false identification, authentication or a duplicate transaction. A message from an authorized user, who is logging into a network, may be captured by an attacker and it is replayed next time. The messages may be encrypted and the attacker does not know the actual keys and passwords. However, retransmission of a message can also be problematic in some cases. For example, retransmission of valid logon messages is sufficient to gain access to the network. The resistance to replay of different communication message in our proposed scheme is discussed as follows:

i) Replay attack of login message in the login step L2:

The proposed scheme can resist replay attack of login message in the login step L2 of our scheme. Let us assume that an adversary eavesdrops the login message (*TID, A, t*) sent by UD_i in login step L2 and uses it to impersonate an UD_i while logging into sensor node SN in a later session. Our scheme resists the replay attack in the following two ways:

- The adversary replays the same previous message without modification of timestamp t. However, the replay of UD_i's previous login message will be detected by the GW since the timestamp t has already been recorded against the *TID* and the login will be denied by the GW.
- The adversary can modify the message and send the triplet (TID, A, t_e) to the SN. The GW node computes A'=H($vpw||t_e$). As A' is not the same as A=H(vpw||t), the verification of (C_k= C'_k) will fail at the GW node level and login will be denied.

ii) Replay attack of accept login message in the Authentication step A2:

The proposed scheme can resist a replay attack on the accept-login message in the authentication step A2 in the following ways:

- While transmitting (Acc_Login, V_s, V_u, T_1) from GW to SN, the malicious party can intercept the message before forwarding it.

- In the next session, when a legitimate SN sends (*TID*, C_k, T_0, *t)* to GW, the malicious party intercepts and drops that message to replay the captured *Acc_Login* message to SN pretending itself a legal GW.
- Since A' in V_s i.e. $H(X_s|| A'||T_0||T_1)$ is different from A in V'_s, the verification of (V_s=V'_s) will fail and the login will be rejected in SN.
- Alternatively, the malicious party can modify the *Acc_Login* message by replacing T_1 with T_{1e} and send the message to SN. Since T_1 in V_s is different from T_{1e} in V'_s, the verification of (V_s=V'_s) will fail and the login will be rejected in SN.

iii) Replay attack of accept login message in authentication step A4:

The scheme can resist the replay of accept login message in authentication step A4 in the following way:

- While transmitting *(Acc_Login,* Y_u*,* T_2*)* from SN to UD_i in the authentication step A4, an adversary eavesdrop the message.
- In the next session, the login message *(TID, A, t)* from UD_i can be blocked by the adversary.
- The captured *Acc_Login* message is replayed to UD_i pretending a message from a legal SN.
- Since A' in V_u i.e. H(*vpw*‖*A'*)is different from A in V'_u, the verification of (Y_u=Y'_u) will fail and the login will be rejected in UD_i.
- Alternatively, if T_2 is modified with T_{2e} then T_{2e} in Y'_u will be different from T_2 in Y_u. As a result the verification of (Y_u=Y'_u) will fail and the login will be rejected in UD_i.

iv) Replay attack of messages in login step L2 and authentication step A2:

The adversary eavesdrops the login message *(TID, A, t)* in login step L2 and (*Acc_Login*, V_s,V_u,T_1) in the authentication step A2.

- The adversary replays the message *(TID, A, t)* to the sensor node, SN. SN checks that the *TID* is a valid user and computes C_{ke} at time T_{0e}.
- The adversary blocks the message sent from the SN to GW in the login step L4 preventing the GW from receiving this message.
- The adversary replays (*Acc_Login*, V_s,V_u, T_1) message in the authentication step A2 to SN.
- SN receives message at T_{2e} and computes $V_s=H(X_s||A||T_0||T_1)$ that is different from $V'_s=H(X_s||A'||T_{0e}||T_1)$ in the replayed message. In this way, the login is denied in sensor node.

4.2.2 Forgery Attack

Forgery in wireless sensor network is the process of making, adapting, or imitating communication messages with the intent to deceive. A forgery is essentially concerned with a produced or altered message. In forgery attack, the attacker can use the captured message or part of the message to compute a new message or alter part of the message to produce a new message. The proposed scheme resists the forgery attack in two ways as given below.

i) The adversary eavesdrops or intercepts the login message *(TID*, C_k, T_0, *t)* from SN to GW in the login step L4. Then it can capture SN to get (*TID*, X_s, *TS)*. However, it cannot compute C_k since *A* is not known.

ii)

- Captures SN to get (TID, X_s, TS).
- Eavesdrop the login message (TID, A, t) from UD_i to SN in the login step L2
- The adversary computes $C_{ke}=(X_s \oplus A \oplus T_e)$ with timestamp T_e and sends (TID, C_{ke}, T_e, t) to GW
- Since TID and t is already recorded in the database, the login will be denied by the GW. If t is replaced by t_e then t_e in A' will be different from t in A. As a result, the verification of (C_k= C'_k) will fail and the login will be denied.

4.2.3 Gateway By-pass Attack

If a user is allowed to access data from sensor node directly without GW node's notice then the impact of "node compromise" attack is very high and the scheme is vulnerable to gateway node bypass attack. Our scheme resists gateway bypass attack in the following way:

- Assuming UD_i sends the login message (TID, A, t) in the login step L2 to the captured sensor node.
- The adversary will know (TID, X_s, TS).
- Adversary can compute $V'_s=H(X_s||A||T_{1e})$ in the authentication step A3 but it cannot compute V_u since vpw is not known. In this way, the gateway bypass attack is prevented in our proposed scheme.

4.2.4 Many Logged-in User Threat

Password-based schemes are vulnerable to many logged-in users with the same login-id threat. If a valid user shares his *TID* and password with a second user, then the second user can generate the login message and gain access to the login node. Our scheme can resist many logged in users with the same login-id threat as outlined below.

- Assume that a valid user generates *(TID, A, t)* and logs in to SN.
- GW records the time t against the *TID* and the flag of *TID* is changed from zero to one indicating that a user with this *TID* has logged into the network.
- A user shares its *TID* and password with the second user.
- The second user generates the login message (TID, A, t_m) and sends to SN.
- Sensor node sends (TID, C_k, T_0, t_m) to GW.
- GW checks that the TID is a valid user and t_m is not recorded in the database. However, the flag is set indicating that another user with the same *TID* has already been logged into the network. GW will decline the login request.

4.2.5 Mutual Authentication

Mutual authentication is a process in which both entities in a communications link authenticate each other. In a wireless sensor network environment, the user, gateway and sensor nodes authenticate each other. In this way, network users can be assured that they are doing business exclusively with the legitimate entities and gateway must be sure that all would-be users are attempting to gain access for legitimate purposes. Mutual authentication is gaining acceptance as a tool that can minimize the risk of different attacks in WSN. The proposed scheme provides mutual authentication in the following ways:

- Authentication of UD_i by gateway:

  The user device UD_i computes A = H(vpw||t) during login step L1. In authentication step A1, the gateway computes $A'=H(vpw||t)$ where vpw is the hash of the password of a legitimate user. Gateway uses A' to compute C'_k. When gateway verifies $C_k=C'_k$, it authenticates that vpw in A = H(vpw||t) is from a legitimate user. Therefore, UD_i is authenticated by the gateway.

- Authentication of GW by UD_i :

  In authentication step A1, the gateway computes $V_u= H(vpw||A')$. In authentication step A5, the user device computes $V'_u=H(vpw||A)$. If the UD_i verifies $Y_u=Y'_u$, then V_u and V'_u must be equal. Therefore the user becomes sure that V_u is generated by a legitimate entity and since only the gateway knows the vpw, gateway is authenticated to UD_i .

- Authentication of UD_i by SN :

  The user sends TID to sensor node during the login step L1. The sensor node verifies TID in the lookup table during step L3 of the login phase and authenticates the user.

- Authentication of SN by UD_i:

  Sensor node computes $Y_u=H(V_u||X_s||T_2)$ during step A3 of the authentication phase. The secret key X_s is only known to sensor node and the user. In step A5 of the authentication phase, UD_i computes $Y'_u=H(V'_u||X_s||T_2)$ and checks if $Y_u=Y'_u$. If it is true, X_s in message Y_u is generated by a legitimate sensor node that will be authenticated to the user device UD_i.

- Authentication of SN by GW:

  In login step L3, the sensor node (SN) computes $C_k=H(X_s \oplus A \oplus T_0)$. Here SN uses the secret key X_s to compute C_k. In authentication step A1, the gateway computes $C'_k=H(X_s \oplus A' \oplus T_0)$ and gateway uses the secret key X_s stored in the database of GW node. If GW verifies $C_k=C'_k$, gateway confirms the identity of sensor node and SN is authenticated by the GW.

- Authentication of GW by SN:

  In authentication step A1, GW computes $V_s=H(X_s||A'||T_0||T_1)$ where X_s is the secret key stored in the database of GW. During authentication step A3, SN computes $V'_s=H(X_s||A||T_0||T_1)$ where X_s is the secret key stored in SN’s lookup table. If SN verifies ($V_s=V'_s$), GW is authenticated to the SN.

4.3 Performance Analysis

Good Performance is very important for any authentication algorithm or scheme. A slow authentication program is almost as useless as an incorrect one. The two most important metrics that are used for evaluating performance of a scheme is the time to execute the algorithm and the cost of the scheme in terms of memory usage. However, designing a great scheme is not just about performance. There are a number of factors such as security, functionality, robustness, simplicity, scalability and reliability that can be more important than performance. The evaluation of the performance of our scheme is presented in the next chapter. In this section we present the analysis of our scheme on the basis of computational overhead and functional requirements.

4.3.1 Computational Overhead

We have used computational overhead as a metric to evaluate the performance of our proposed authentication scheme as compared to the past representative schemes [24, 46, 48]. The comparison of the computational overhead is presented in Table 4.1.

The number of elements contained in the messages is not considered for comparison. Tseng *et al.*'s scheme has lowest computational cost [46]. However, their scheme suffers from a number of security threats. Ko's algorithm [24] provides better security as compared to Tseng *et al.*'s scheme at an expense of higher computational overhead. Our proposed scheme slightly adds some computational overhead at GW than the Vaidya *et al.'s* scheme [48]. However, in most WSN applications, the computational capability of the GW and the user devices are more powerful than SN. The one-way Hash function and the XOR operation are considered lightweight for these two devices. Therefore, the computational load increase for GW and UD is negligible. It means that without adding any extra computational load for the SN, our proposed scheme provides higher security than the Vaidya *et al.'s* method [48].

Table 4.1: Comparison of Computational Overhead

Overhead cost (registration, login and authentication)				
Scheme	User	Gateway Node	Sensor Node	Total
Tseng *et al.'s* scheme	$2T_H + 1T_{XOR}$	$2\ T_H + 2T_{XOR} + (K+1)C_{MH}$	$1\ T_H + 1T_{XOR} + 1C_{MH}$	$5\ T_H + 4T_{XOR} + (K+2)C_{MH}$
Ko*'s* scheme	$4\ T_H + 3T_{XOR}$	$6T_H + 8T_{XOR} + (K+1)\ C_{MH}$	$4T_H + 5T_{XOR} + 1C_{MH}$	$14T_H + 16T_{XOR} + (K+2)C_{MH}$
Vaidya *et al.'s* scheme	$5\ T_H + 1T_{XOR}$	$5\ T_H + 3T_{XOR} + (K+1)C_{MH}$	$3T_H + 2T_{XOR} + 1C_{MH}$	$13T_H + 6T_{XOR} + (K+2)C_{MH}$
Proposed scheme	$5T_H + 1T_{XOR}$	$6\ T_H + 3T_{XOR} + (K+1)C_{MH}$	$3T_H + 2T_{XOR} + 1C_{MH}$	$14T_H + 6T_{XOR} + (K+2)C_{MH}$

4.3.2 Functional Requirements

The comparison of functional requirements between some of the past protocols and our proposed protocol is also presented in Table 4.2. From the comparison, one can observe that the past schemes [24, 46, 48] do not resist multiple login and replay attack within a time interval, ΔT. On the other hand, our proposed scheme provides better security features without adding any extra computational overhead at the SN level.

All the currently available authentication schemes use timestamp to avoid replay attack [24, 46, 48]. However, implementation of strict time synchronization is very difficult and increases the network overhead. If a setting of the transmission delay interval is too short, it will cause the failure of a legal user's login. On the other hand, setting a large transmission delay will lead to replay attacks. Our proposed scheme inherently resists replay attack because it does not rely on network synchronization. The authentication latency time of our scheme is low due to the fact that the scheme does not require checking time interval ΔT in any network entity.

Table 4.2 Comparison of Functional Requirements

Function	Representative schemes			
	Tseng	Ko	Vaidya	Proposed
Password changing	Yes	Yes	Yes	Yes
Mutual authentication between GW and SN	No	Yes	Yes	Yes
Mutual authentication between GW and UD	No	Yes	Partial	Yes
User anonymity	No	No	Yes	Yes
Resist replay attack within ΔT	No	No	No	Yes
Resist GW bypass attack	No	Yes	No	Yes
Resist multiple login	No	No	No	Yes

4.3.3 Simplicity

Developers should design for simplicity by looking for ways to break up the scheme into small and straightforward cooperating pieces. This rule aims to discourage developers' affection for writing "intricate and beautiful complexities" that are in reality bug prone programs. Our proposed scheme is simple and divided into four phases. The scheme uses simple hash and bitwise XOR operations. The communication steps are clearly elaborated and easily readable to the reader.

4.3.4 Robustness

Robustness describes how reliable the scheme is, especially under extreme conditions such as extreme workload and bad or unpredictable user inputs. Ideally, our scheme never crashes no matter what the user inputs. Even if the user enters invalid data, our scheme handles it properly. The worse thing it can do is to terminate the program gracefully.

4.3.5 Scalability

The scheme does not require the pre-deployment of key in any sensor node. Only the Gateway node has the initial secret key. When a user registers with the gateway, it distributes a computed secret key to the sensor nodes. During the expansion of network, the scheme does not require any modification.

4.4 Conclusion

In this chapter we have discussed the security and performance analysis of our proposed scheme. The security analysis mainly focuses on the identified security flows of some past schemes. The performance analysis is based on functional requirement, computational cost, simplicity, robustness and scalability of the scheme. In chapter 5 we provide the modeling and performance evaluation of our scheme.

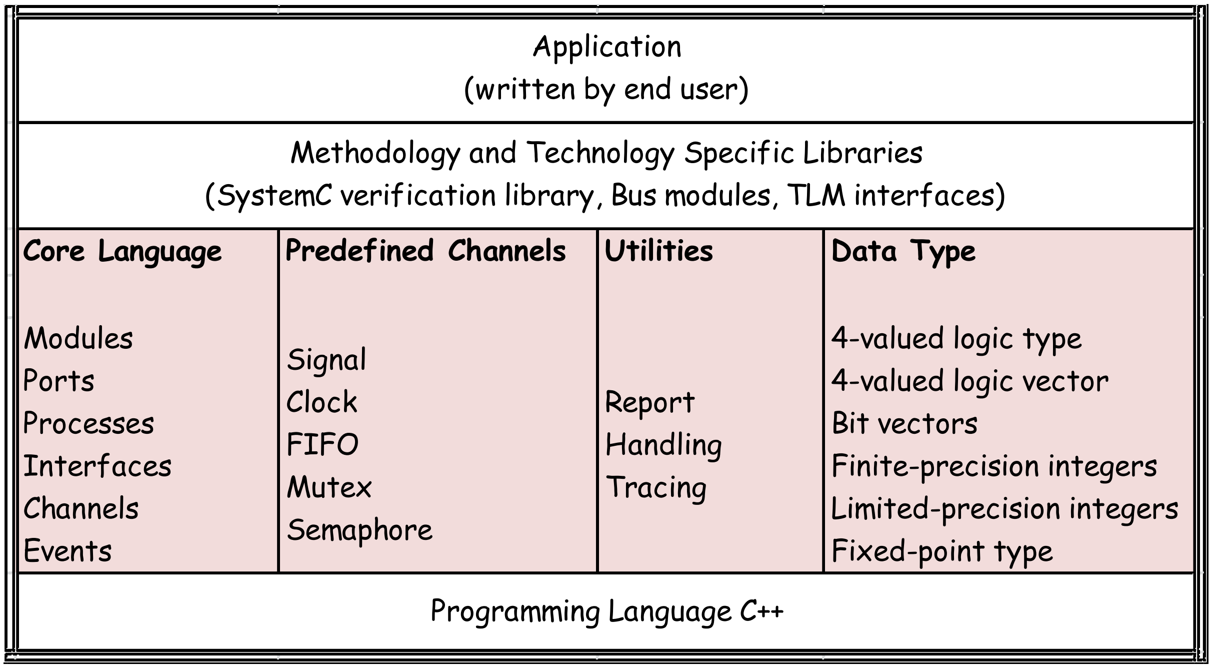
Chapter 5: Protocol Modeling for Performance Evaluation

5.1 Introduction

In this chapter we elaborate the modeling of our proposed scheme in SystemC. The security features are verified and the performance of the scheme is evaluated on the basis of execution time and cost (storage requirement). A comparison of authentication latency and storage requirement of some of the representative schemes are also provided in this chapter.

5.2 Protocol Architecture

We have modelled a WSN environment using SystemC [67]. The model is based on the overall conceptual framework and our proposed security scheme. SystemC is a class library for C++ that allows the functional modeling of embedded systems. It is a hierarchical decomposition of a system into modules where the connectivity between those modules uses established communication ports. SystemC schedules and synchronizes concurrent processes using events and clock sensitivity. It also separates computation (processes) from communication (channels). The language architecture of SystemC is shown in Figure 5.1.



Application
(written by end user)

Methodology and Technology Specific Libraries
(SystemC verification library, Bus modules, TLM interfaces)

Core Language	**Predefined Channels**	**Utilities**	**Data Type**
Modules Ports Processes Interfaces Channels Events	Signal Clock FIFO Mutex Semaphore	Report Handling Tracing	4-valued logic type 4-valued logic vector Bit vectors Finite-precision integers Limited-precision integers Fixed-point type

Programming Language C++

Figure 5.1: Language Architecture of SystemC

Our authentication protocol specification consists of SystemC modules communicating through channels. Modules are the basic building blocks for partitioning the design. A module is a structural entity, which can contain *processes, ports, channels* and other modules. It is the foundation of structural hierarchy that allow designers to hide internal data representation and algorithms from other modules. Designers are forced to use public interfaces to other modules, thus making the entire system easier to change and maintain. Module ports are used to pass data to and from the processes of a module. A port data type can be any C++, SystemC or user-defined type. Channels are connected to the module ports. The module constructor SC_CTOR is a macro that performs the initialization, registration of processes with the SystemC kernel and provides static sensitivity. Each module contains at least one process. Processes are activated according to a sensitivity list defined statically or dynamically.

The module specification has two parts: communication model and the software model. The communication model consists of header files which contain description of SystemC modules with emphasis on communication aspects. The most essential parts of the communication model are given below.

- The processes located in various SystemC modules.
- Ports of each module that are used by these processes.
- Channels that connect the modules via ports.

Figure 5.2 lists the gateway module header file, gateway.h.

```
#include "packet.h"
#define ID_G "1"
#define Succ_Reg 200
#define Acc_login 210
#define Succ_Change 220
#define secret_x "1234567890abcdeffedcba0987654321"
#define del_T 200
#define TIMEOUT 1000
SC_MODULE(gateway) {
        sc_in<packet_type>gupackin; // input port
        sc_in<packet_type>glpackin; // input port
        sc_out<packet_type>gupackout; // output port
        sc_out<packet_type>glpackout; // output port
        sc_in<bool>gclk;
        packet_typeguin, glin, guout, glout;
        packet_typegupackold, glpackold, s, epackold;
        std::string ID_U,TID, X, vpw;
        vector<string>MultiID_U, MultiTID, MultiX, Multivpw;
        longint TS, t_pre, log_count;
        vector<int>MultiTS, Multilog_count;
        voidget_data_user();
        voidget_data_ln();
        voidreplay_nc();
        voidreplay_c();
        vector<int>Multit_pre;
        // Constructor
        SC_CTOR(gateway) {
                SC_METHOD(get_data_user); // Method Process
                sensitive<<gupackin;
                //sensitive_pos<<gclk;
                SC_METHOD(get_data_ln); // Method Process
                sensitive<<glpackin;
                //sensitive_pos<<gclk;

        }
};
```

Figure 5.2: A Sample Header File (gateway.h)

Software model consists of program files that translate module specifications into C++ programs. Figure 5.3 shows a sample program file, gateway.cpp.

```
#include "gateway.h"
void gateway::get_data_user() {
    //Sleep(70);
    std::string  Vs;
    longint nonce0, nonce1;
    std::string con_cat;
    std::stringCk;
    std::string Ck_;
    std::string v;
    //int diff1, diff2,T0, t;
    std::string i, n0, n1;
    std::string uid_str, g_str, g_bstr, N0_bstr, T0_str, T_str, T1_str, t_cstr;
    std::bitset<512>  N1, TID_bit, A_bit, T0_bit, Ck_bit, TID1_bit, X1_bit, TID11_bit;
    intp,l,y;

    guin=gupackin;
    if (guin == gupackold) {
        return;
    }

    if (guin.mid == 10) {

        /*cout<< "GW : Packet received from user, ID = " << guin.mid << "\n";
        cout<<"GW : packet.f1(vpw) = "<<guin.f1 << "\n";
        cout<<"GW : packet.sid(UID) = " <<guin.sid<< "\n";*/
        ID_U = guin.sid;                                          //user ID

        for(l=0;l!=MultiID_U.size();l++){
            if(MultiID_U[l]==ID_U){
                break;
            }
        }
```

Figure 5.3: A Sample C++ File (gateway.cpp)

The user module generates the registration packet from the data provided by the user and sends the packet to GW node. The packet size used in our scheme is of 382 bytes length. We have used secure Hash function SHA-512 to generate the message digest in our scheme. However, 160-bit SHA-1 Hash function could be used to address the bandwidth constrain among different modules of our simulation. This would also reduce the storage (memory) requirement for the Gateway (GW) and Sensor (SN) modules. In fact, a range of message digest can be employed depending on the security requirement and resource constraints in GW and sensor nodes.

5.3 Delay Analysis

A major problem in deploying WSNs is their dependence on the limited battery power. A main design criterion is to extend the lifetime of the network without jeopardizing reliable and efficient communications from sensor to other nodes as well as gateway. Optimizing every facet of the communication protocols is therefore vital. The aim of a WSN design is to guarantee its longevity under the given energy and complexity constraints. The MAC (Medium Access Control) plays a central part in this design since it controls the active and sleeping state of each node. The MAC protocols needs to trade longevity, reliability, fairness, scalability and latency where throughput is rarely a primary design factor.

Most proposals for energy saving MAC schemes for sensor networks introduce a sleeping mode for nodes, during which no energy is consumed. Introducing a sleeping mode does however come at some cost. A solution is to use a Time Division Multiple Access (TDMA) scheme but this requires node synchronization tightly. This can be a quite complex task in large networks with random node locations and imperfect (drifting) clocks. The complexity can be reduced by letting the nodes to set their wake-up and sleeping times in a decentralized way. However, it increases the delay or latency to transfer information between the sink and a distant node. For some applications, such as spatial data collection for statistical purposes, this is acceptable but not for many others that are more time-critical. Even if some fixed amount of latency can be tolerated, a highly variable latency due to the random position of the nodes, random radio range, non-synchronized or random sleeping and active periods will be much more problematic.

WSNs usually use a single channel, which results in a long latency due to high interference, especially in high-density networks. When two or more sensor nodes send data to a common neighbour at the same time, data collision occurs at the common neighbour preventing it from successfully receiving any data. The data sent by a sender should be received by a corresponding receiver with no collisions. The receiver aggregates the incoming data with its own data, and stores the aggregated data as its new data. The time consumed by a single sending-receiving-aggregating-storing is normalized to one, and parallel sending-receiving is preferred for reducing network delay. Since the scheduling and associated delays happen at the MAC level, each application may require a different solution at the link layer.

5.3.1 Application Layer Delay

We define latency as the difference between the time a login packet is sent to the sensor node and the time a success_login message is received by the user. This is the time required to execute the authentication code in the application layer. In our simulation, there is no node to node communication delay since modules are directly connected through ports by signals. The authentication latency is a function of the number of users and encryption cipher used in the scheme. The proposed scheme can be integrated to any lower level MAC protocol specifically designed for wireless sensor network. In that case the latency due to lower level will be determined by that particular protocol. We have also studied the effect of multiple users on authentication latency for our proposed authentication scheme. Figure 5.4 shows the plot of application layer latency against the number of users in our simulation.

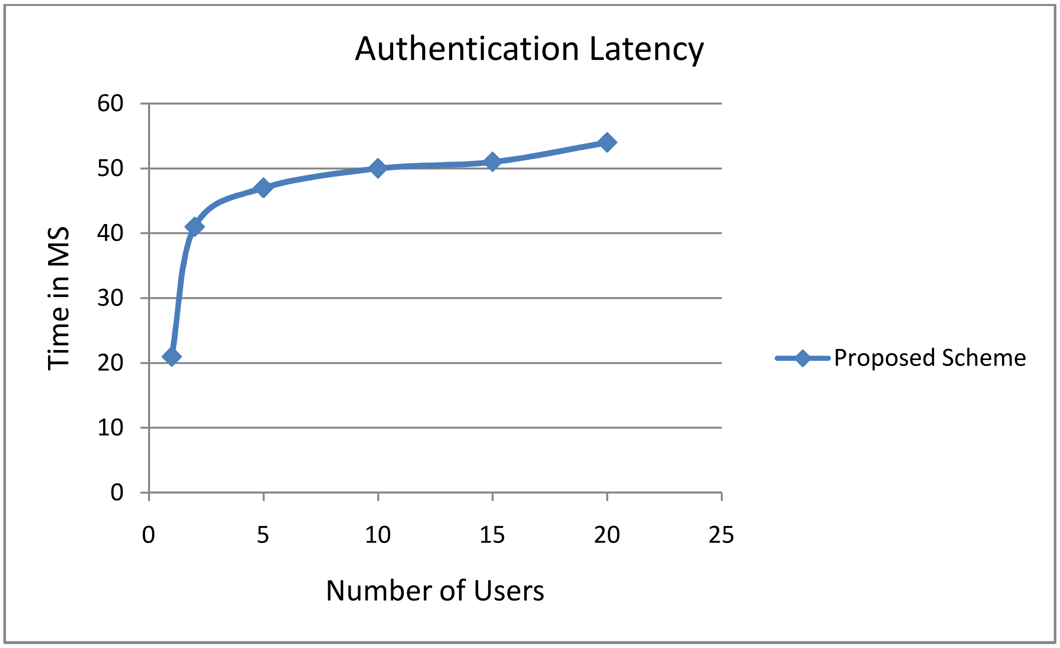


Figure 5.4: Authentication Latency

From the graph of Figure 5.4 one can observe that the latency increases as the number of users grow which is due to the increase in computational load of the gateway.

A comparison of the application layer latency against the number of users among the proposed and existing schemes is shown in Figure 5.5. Ko's scheme has a higher latency time due to higher computational load on sensor and gateway modules. The latency of our proposed scheme is comparable to Vaidya *et al.*'s scheme. Moreover, the scheme provides better security than all the existing password based protocols. We observe that the latency increases with an increase of computational load on gateway as the number of users grow in the network.

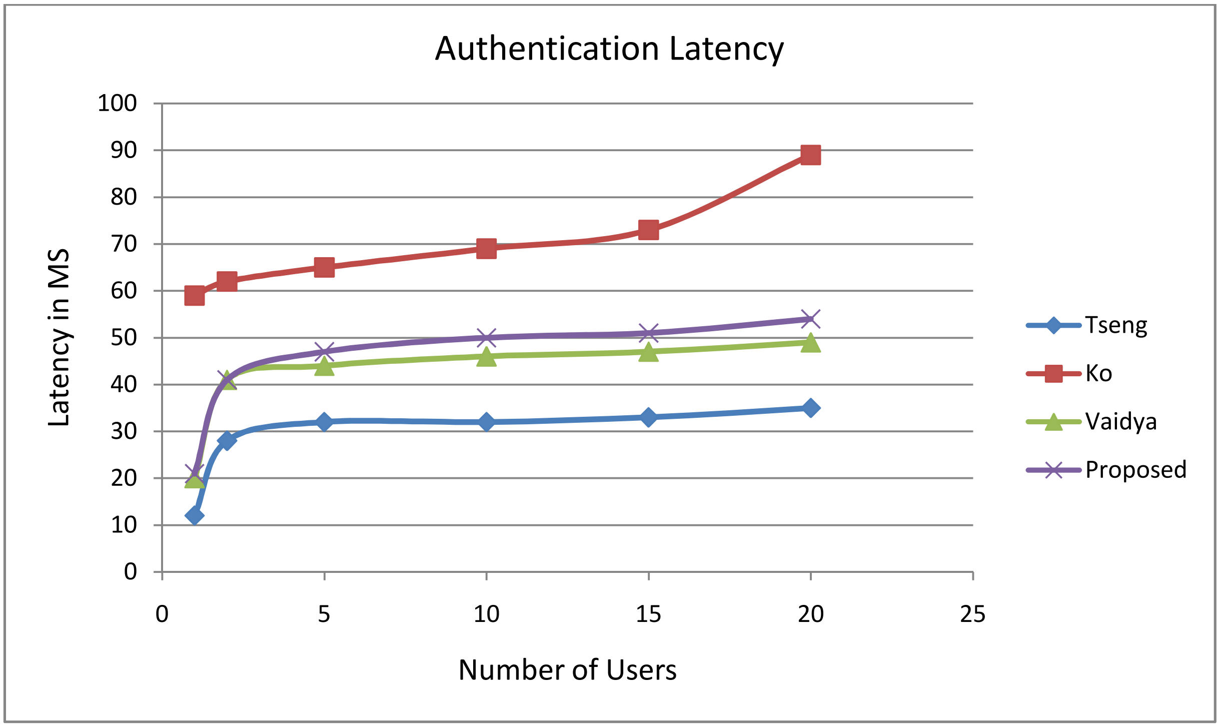


Figure 5.5: Comparison of Authentication Latency

5.3.2 Network Layer Delay

In most of the scenarios of relevant interest, wireless multi-hop networks do not have fixed communication paths. However, an end-to-end path followed by the packets is established according to a dynamic selection of hops. Indeed, it is often impossible to build fixed routing tables for these networks due to time-varying communication channel and network topology. In such networks, a packet is routed to a particular node (next-hop) as long as the node selected for a hop ensures progress towards the destination. Furthermore, a major technique to ensure power savings and longer network lifetime is to turn off a node whenever its presence is not strictly required for the correct operation of network. In this case, each node goes to sleep for a random amount of time depending on the traffic and network conditions, which means that the network topology is changing randomly. Ping has identified the following sources of delay in WSN [36].

- ***Sender processing delay***: This is the time elapsed from the moment a timestamp is taken to the point it is buffered in a sensor related RF device.
- ***Media Access delay***: This is the duration for a timestamp message stays in the RF device buffer. For TDMA system, this is the time spent waiting for a designated time slot. For CSMA system, it is the delay waiting for a clear channel to transmit.
- ***Transmit time:*** This is the time for a radio device to transmit a packet over a radio link. When a packet has a fixed length and transmission speed is constant, then the transmission time can be easily estimated.
- ***Radio propagation time:*** This is the time for a signal to propagate over the air to reach a receiver. Radio propagation speed is 300 meters per microsecond. Since the radio coverage of a wireless sensor network device is short and usually less than 100 meters, this delay is negligible.
- ***Receiver processing time:*** Time consumed on the receiver side to pass the received packet from RF device buffer to the application module that is responsible for processing the packet.

Among these delays, transmit time and radio propagation time could be considered symmetric to the paths of different directions. Media access time is the key uncertainty. In addition to these, a packet can be corrupted or lost along the path. If the MAC protocol retransmits the packet, the round trip time estimation error will increase significantly. Figure 5.6 shows the delay time line between two nodes.

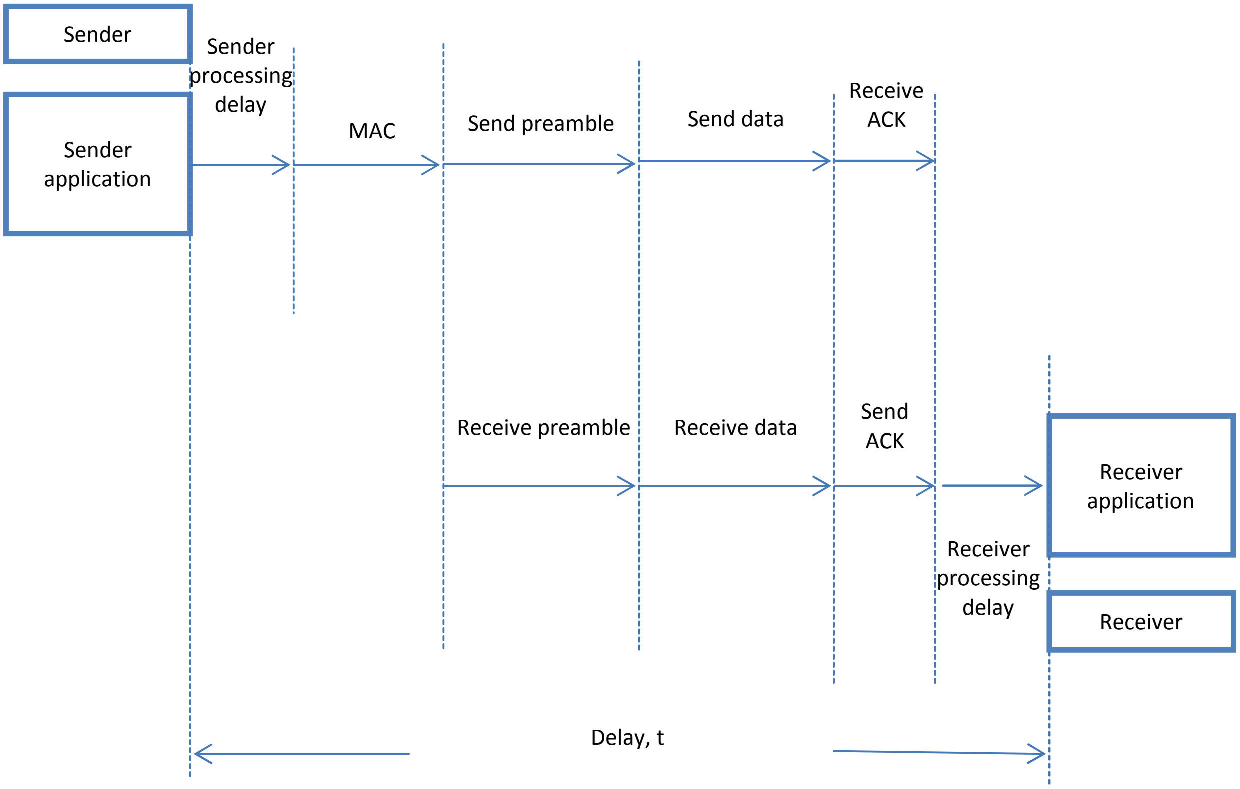


Figure 5.6: Delay Time Line Between Two Nodes

Witrant *et al.* setup a WSN and conducted an experiment based on the Breath routing protocol using sensor nodes [54]. They observed that the average end-to-end network induced delay for a two-hop and four-hop WSN are 70ms and 150ms respectively. Therefore, the application layer latency of our proposed scheme is reasonably lower than the network induced delay especially for a multi-hop network.

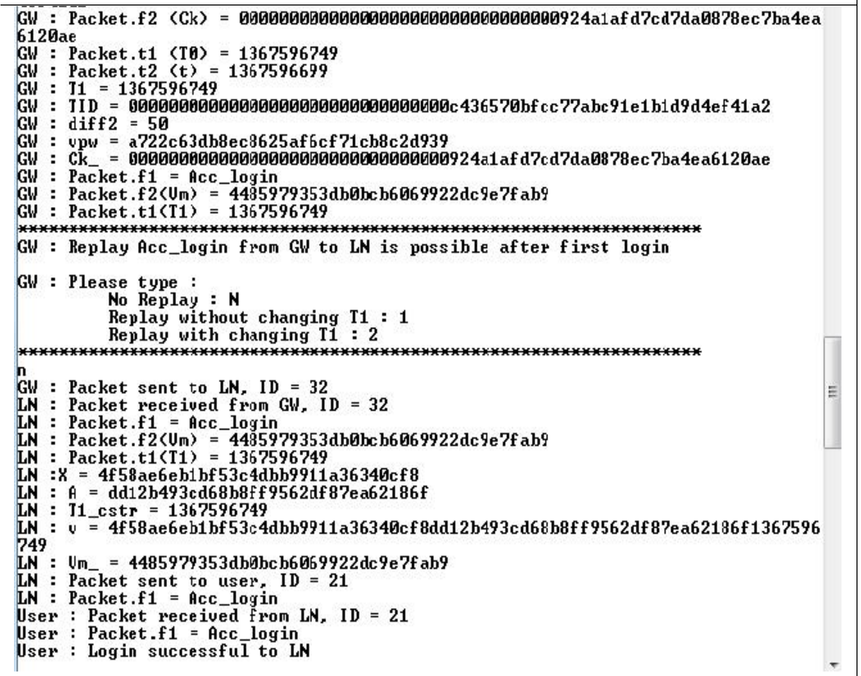
5.4 Security evaluation of some Representative Schemes

The securities of the existing schemes [24, 46, 48] are investigated by simulating the schemes in SystemC. We have observed the following about these schemes.

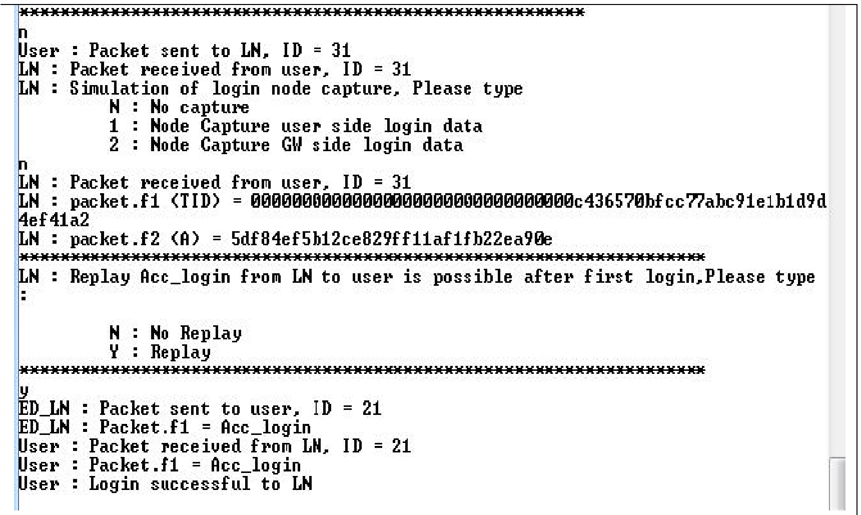
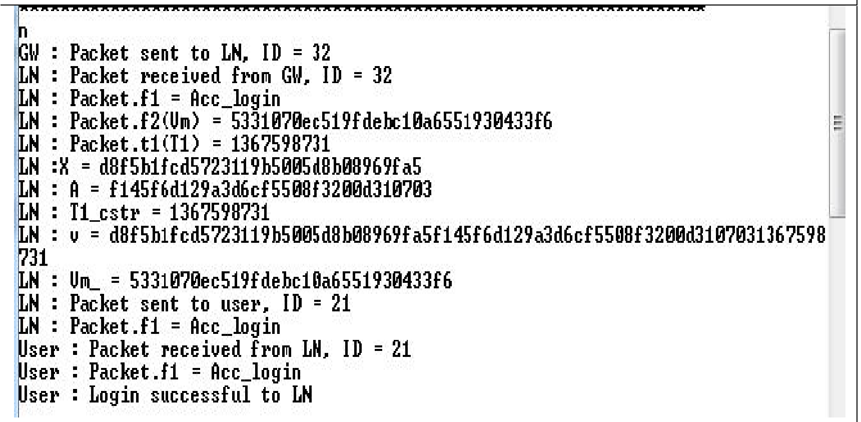
- Tseng *et al.*'s scheme cannot resist replay attack, man in the middle attack, stolen verifier attack, node capture attack, forgery attack and multiple login attack.
- Ko's scheme suffers from replay attack, stolen verifier attack and multiple login attack.
- Vaidya *et al.*'s scheme suffers from replay attack within ΔT interval, gateway bypass attack due to node capture attack and many logged in user threat.

Table 5.1 shows the simulation results of some of the security flaws of Vaidya *et al.*'s scheme.

Table 5.1 : Simulation of Attacks on Vaidya *et al.'s* Scheme

Security flaws	***Result from simulation***
The scheme cannot resist replay attack of login message L1 within ΔT	

```
GW : Packet.f2 (Ck) = 00000000000000000000000000000000924a1afd7cd7da0878ec7ba4ea
6120ae
GW : Packet.t1 (T0) = 1367596749
GW : Packet.t2 (t) = 1367596699
GW : T1 = 1367596749
GW : TID = 00000000000000000000000000000000c436570bfcc77abc91e1b1d9d4ef41a2
GW : diff2 = 50
GW : vpw = a722c63db8ec8625af6cf71cb8c2d939
GW : Ck_ = 00000000000000000000000000000000924a1afd7cd7da0878ec7ba4ea6120ae
GW : Packet.f1 = Acc_login
GW : Packet.f2(Vm) = 4485979353db0bcb6069922dc9e7fab9
GW : Packet.t1(T1) = 1367596749
*******************************************************************
GW : Replay Acc_login from GW to LN is possible after first login

GW : Please type :
        No Replay : N
        Replay without changing T1 : 1
        Replay with changing T1 : 2
*******************************************************************
n
GW : Packet sent to LN, ID = 32
LN : Packet received from GW, ID = 32
LN : Packet.f1 = Acc_login
LN : Packet.f2(Vm) = 4485979353db0bcb6069922dc9e7fab9
LN : Packet.t1(T1) = 1367596749
LN :X = 4f58ae6eb1bf53c4dbb9911a36340cf8
LN : A = dd12b493cd68b8ff9562df87ea62186f
LN : T1_cstr = 1367596749
LN : v = 4f58ae6eb1bf53c4dbb9911a36340cf8dd12b493cd68b8ff9562df87ea62186f1367596
749
LN : Vm_ = 4485979353db0bcb6069922dc9e7fab9
LN : Packet sent to user, ID = 21
LN : Packet.f1 = Acc_login
User : Packet received from LN, ID = 21
User : Packet.f1 = Acc_login
User : Login successful to LN
```

The scheme cannot resist gateway bypass attack	 ** n User : Packet sent to LN, ID = 31 LN : Packet received from user, ID = 31 LN : Simulation of login node capture, Please type N : No capture 1 : Node Capture user side login data 2 : Node Capture GW side login data n LN : Packet received from user, ID = 31 LN : packet.f1 (TID) = 00000000000000000000000000000000c436570bfcc77abc91e1b1d9d 4ef41a2 LN : packet.f2 (A) = 5df84ef5b12ce829ff11af1fb22ea90e ** LN : Replay Acc_login from LN to user is possible after first login,Please type : N : No Replay Y : Replay ** y ED_LN : Packet sent to user, ID = 21 ED_LN : Packet.f1 = Acc_login User : Packet received from LN, ID = 21 User : Packet.f1 = Acc_login User : Login successful to LN
The scheme suffers from multiple login with same ID attack.	 ** n GW : Packet sent to LN, ID = 32 LN : Packet received from GW, ID = 32 LN : Packet.f1 = Acc_login LN : Packet.f2(Vm) = 5331070ec519fdebc10a6551930433f6 LN : Packet.t1(T1) = 1367598731 LN :X = d8f5b1fcd5723119b5005d8b08969fa5 LN : A = f145f6d129a3d6cf5508f3200d310703 LN : T1_cstr = 1367598731 LN : v = d8f5b1fcd5723119b5005d8b08969fa5f145f6d129a3d6cf5508f3200d3107031367598 731 LN : Vm_ = 5331070ec519fdebc10a6551930433f6 LN : Packet sent to user, ID = 21 LN : Packet.f1 = Acc_login User : Packet received from LN, ID = 21 User : Packet.f1 = Acc_login User : Login successful to LN

5.5 Evaluation of Proposed Scheme

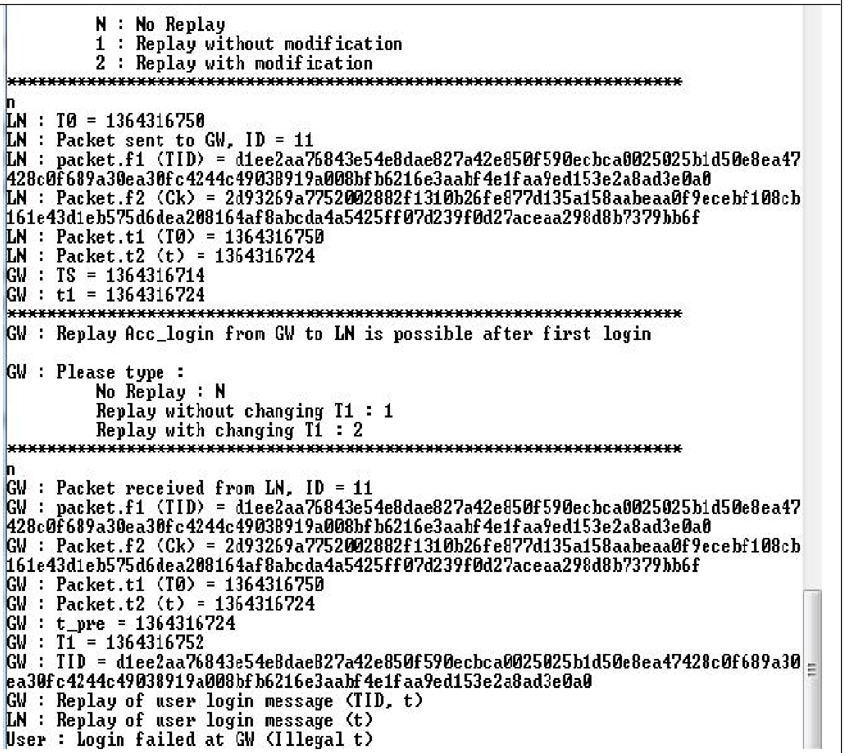
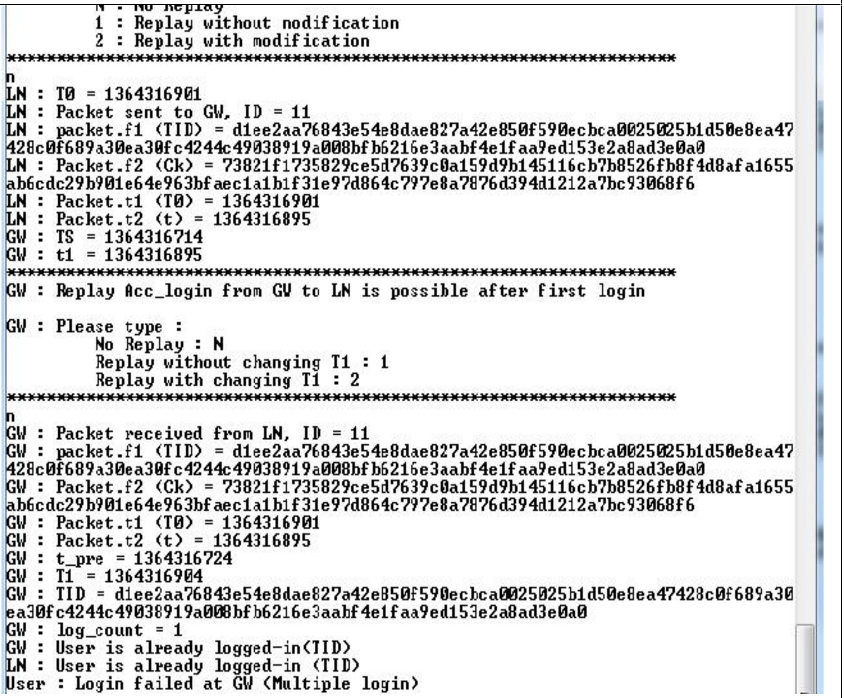
We have evaluated our scheme in different attack scenarios to verify the security claims. The delay in the application layer is also an important metric for the performance evaluation of our authentication scheme. This delay has been discussed in detail in the previous section. In this section, we evaluate the performance of our scheme from security and cost point of view.

5.5.1 Security Evaluation

Replay, node capture, gateway bypass and multiple login attacks are further investigated by simulating our authentication scheme. It is verified that the proposed protocol is resistant against all the above attacks that re-affirms our claims of chapter 4. The simulation of the security claims of our proposed scheme are shown in Table 5.2a, 5.2b and 5.2c.

Table 5.2a shows the simulation result of replay attack. In replay attack, a valid communication message is captured and replayed at a later time to gain access to the sensor data. The simulation result confirms that our proposed authentication scheme successfully resists replay attack.

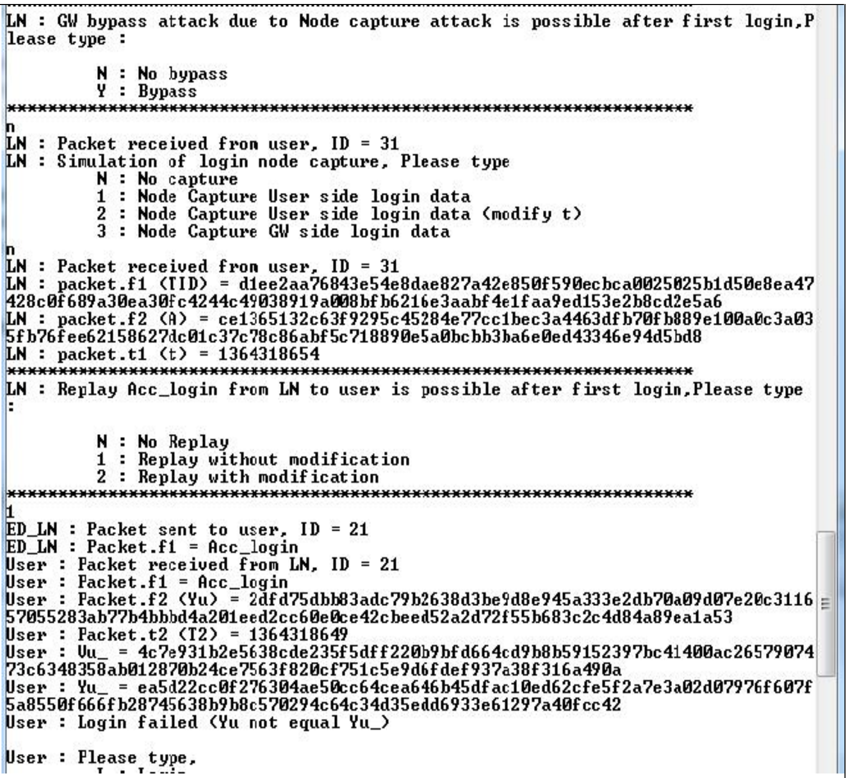
Table 5.2a: Simulation of Replay Attack

Claim	***Result from simulation***
The proposed scheme can resist Replay attack of login message in the login phase L1.	 

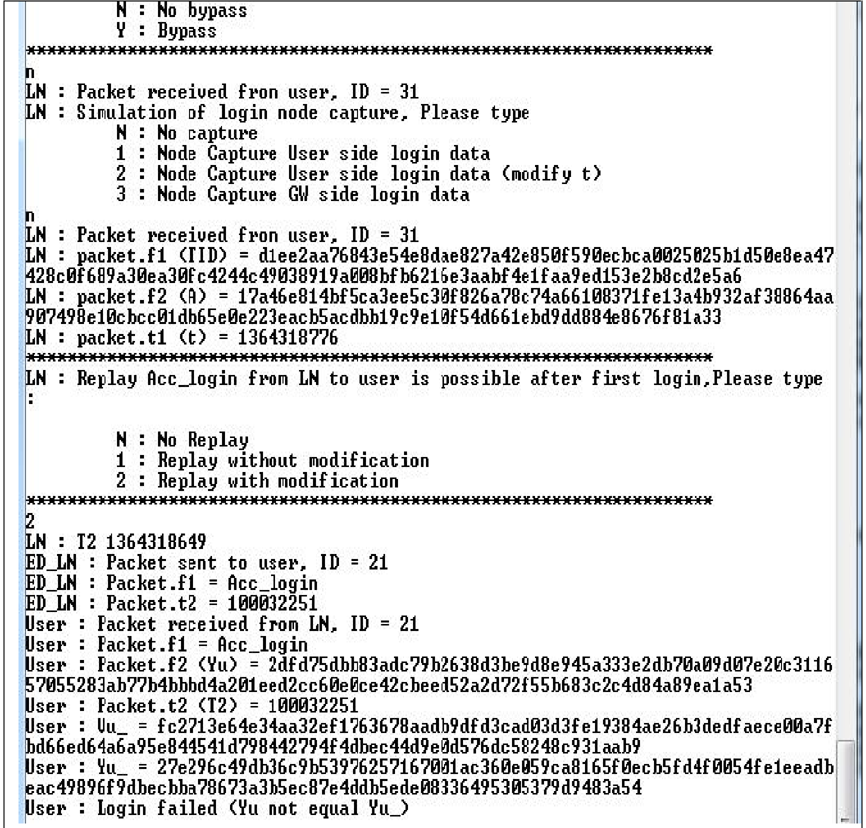
```
        N : No Replay
        1 : Replay without modification
        2 : Replay with modification
********************************************************************
n
LN : T0 = 1364316750
LN : Packet sent to GW, ID = 11
LN : packet.f1 (TID) = d1ee2aa76843e54e8dae827a42e850f590ecbca0025025b1d50e8ea47
428c0f689a30ea30fc4244c49038919a008bfb6216e3aabf4e1faa9ed153e2a8ad3e0a0
LN : Packet.f2 (Ck) = 2d93269a7752002882f1310b26fe877d135a158aabeaa0f9ecebf108cb
161e43d1eb575d6dea208164af8abcda4a5425ff07d239f0d27aceaa298d8b7379bb6f
LN : Packet.t1 (T0) = 1364316750
LN : Packet.t2 (t) = 1364316724
GW : TS = 1364316714
GW : t1 = 1364316724
********************************************************************
GW : Replay Acc_login from GW to LN is possible after first login

GW : Please type :
        No Replay : N
        Replay without changing T1 : 1
        Replay with changing T1 : 2
********************************************************************
n
GW : Packet received from LN, ID = 11
GW : packet.f1 (TID) = d1ee2aa76843e54e8dae827a42e850f590ecbca0025025b1d50e8ea47
428c0f689a30ea30fc4244c49038919a008bfb6216e3aabf4e1faa9ed153e2a8ad3e0a0
GW : Packet.f2 (Ck) = 2d93269a7752002882f1310b26fe877d135a158aabeaa0f9ecebf108cb
161e43d1eb575d6dea208164af8abcda4a5425ff07d239f0d27aceaa298d8b7379bb6f
GW : Packet.t1 (T0) = 1364316750
GW : Packet.t2 (t) = 1364316724
GW : t_pre = 1364316724
GW : T1 = 1364316752
GW : TID = d1ee2aa76843e54e8dae827a42e850f590ecbca0025025b1d50e8ea47428c0f689a30
ea30fc4244c49038919a008bfb6216e3aabf4e1faa9ed153e2a8ad3e0a0
GW : Replay of user login message (TID, t)
LN : Replay of user login message (t)
User : Login failed at GW (Illegal t)
```

```
        N : No Replay
        1 : Replay without nodification
        2 : Replay with modification
********************************************************************
n
LN : T0 = 1364316901
LN : Packet sent to GW, ID = 11
LN : packet.f1 (TID) = d1ee2aa76843e54e8dae827a42e850f590ecbca0025025b1d50e8ea47
428c0f689a30ea30fc4244c49038919a008bfb6216e3aabf4e1faa9ed153e2a8ad3e0a0
LN : Packet.f2 (Ck) = 73821f1735829ce5d7639c0a159d9b145116cb7b8526fb8f4d8afa1655
ab6cdc29b901e64e963bfaec1a1b1f31e97d864c797e8a7876d394d1212a7bc93068f6
LN : Packet.t1 (T0) = 1364316901
LN : Packet.t2 (t) = 1364316895
GW : TS = 1364316714
GW : t1 = 1364316895
********************************************************************
GW : Replay Acc_login from GV to LN is possible after first login

GW : Please type :
        No Replay : N
        Replay without changing T1 : 1
        Replay with changing T1 : 2
********************************************************************
n
GW : Packet received from LN, ID = 11
GW : packet.f1 (TID) = d1ee2aa76843e54e8dae827a42e850f590ecbca0025025b1d50e8ea47
428c0f689a30ea30fc4244c49038919a008bfb6216e3aabf4e1faa9ed153e2a8ad3e0a0
GW : Packet.f2 (Ck) = 73821f1735829ce5d7639c0a159d9b145116cb7b8526fb8f4d8afa1655
ab6cdc29b901e64e963bfaec1a1b1f31e97d864c797e8a7876d394d1212a7bc93068f6
GW : Packet.t1 (T0) = 1364316901
GW : Packet.t2 (t) = 1364316895
GW : t_pre = 1364316724
GW : T1 = 1364316904
GW : TID = d1ee2aa76843e54e8dae827a42e850f590ecbca0025025b1d50e8ea47428c0f689a30
ea30fc4244c49038919a008bfb6216e3aabf4e1faa9ed153e2a8ad3e0a0
GW : log_count = 1
GW : User is already logged-in(TID)
LN : User is already logged-in (TID)
User : Login failed at GW (Multiple login)
```

The proposed scheme can resist a replay attack on accept login message in the authentication phase A2

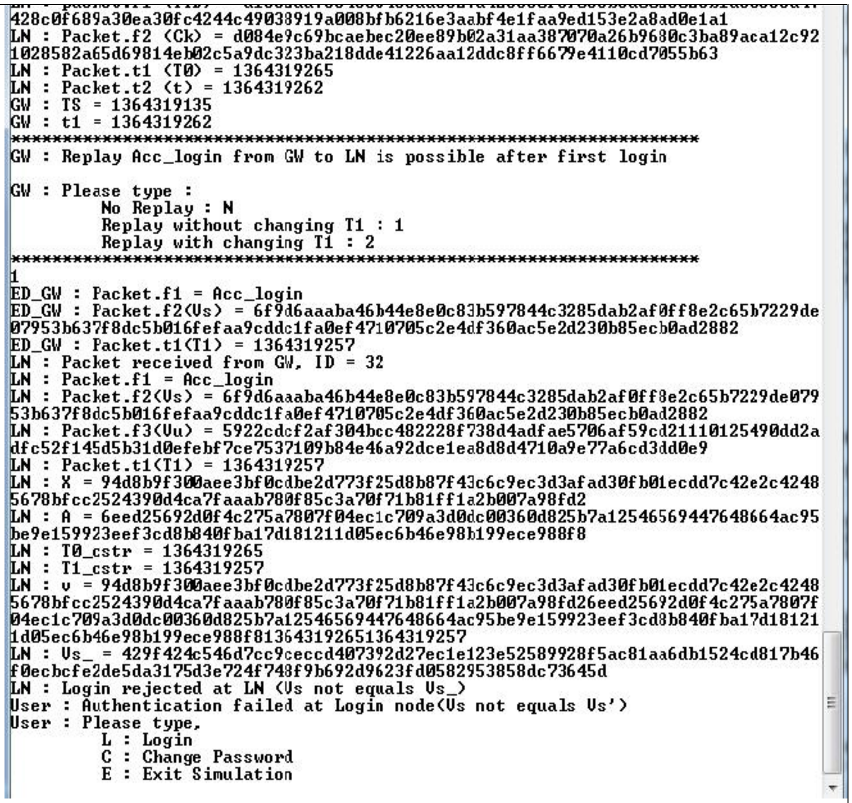


```
LN : GW bypass attack due to Node capture attack is possible after first login,P
lease type :

        N : No bypass
        Y : Bypass
*********************************************************************
n
LN : Packet received fron user, ID = 31
LN : Simulation of login node capture, Please type
        N : No capture
        1 : Node Capture User side login data
        2 : Node Capture User side login data (modify t)
        3 : Node Capture GW side login data
n
LN : Packet received fron user, ID = 31
LN : packet.f1 (TID) = d1ee2aa76843e54e8dae827a42e850f590ecbca0025025b1d50e8ea47
428c0f689a30ea30fc4244c49038919a008bfb6216e3aabf4e1faa9ed153e2b8cd2e5a6
LN : packet.f2 (A) = ce1365132c63f9295c45284e77cc1bec3a4463dfb70fb889e100a0c3a03
5fb76fee62158627dc01c37c78c86abf5c718890e5a0bcbb3ba6e0ed43346e94d5bd8
LN : packet.t1 (t) = 1364318654
*********************************************************************
LN : Replay Acc_login from LN to user is possible after first login,Please type
:

        N : No Replay
        1 : Replay without modification
        2 : Replay with modification
*********************************************************************
1
ED_LN : Packet sent to user, ID = 21
ED_LN : Packet.f1 = Acc_login
User : Packet received from LN, ID = 21
User : Packet.f1 = Acc_login
User : Packet.f2 (Yu) = 2dfd75dbb83adc79b2638d3be9d8e945a333e2db70a09d07e20c3116
57055283ab77b4bbbd4a201eed2cc60e0ce42cbeed52a2d72f55b683c2c4d84a89ea1a53
User : Packet.t2 (T2) = 1364318649
User : Vu_ = 4c7e931b2e5638cde235f5dff220b9bfd664cd9b8b59152397bc41400ac26579074
73c6348358ab012870b24ce7563f820cf751c5e9d6fdef937a38f316a490a
User : Yu_ = ea5d22cc0f276304ae50cc64cea646b45dfac10ed62cfe5f2a7e3a02d07976f607f
5a8550f666fb28745638b9b8c570294c64c34d35edd6933e61297a40fcc42
User : Login failed (Yu not equal Yu_)

User : Please type,
```

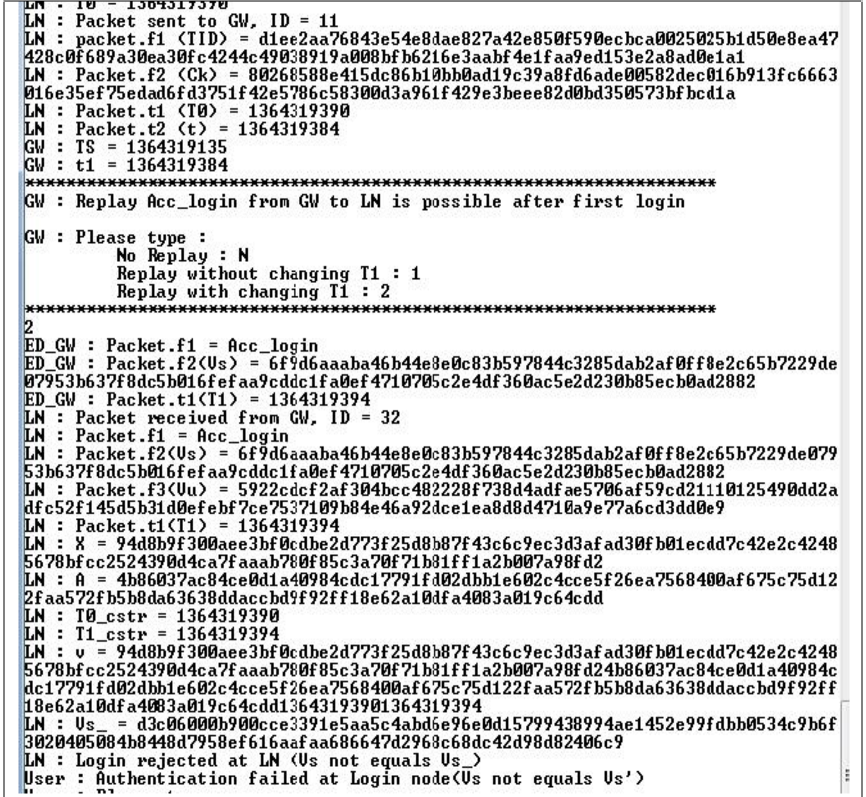


```
        N : No bypass
        Y : Bypass
*********************************************************************
n
LN : Packet received fron user, ID = 31
LN : Simulation of login node capture, Please type
        N : No capture
        1 : Node Capture User side login data
        2 : Node Capture User side login data (modify t)
        3 : Node Capture GW side login data
n
LN : Packet received fron user, ID = 31
LN : packet.f1 (TID) = d1ee2aa76843e54e8dae827a42e850f590ecbca0025025b1d50e8ea47
428c0f689a30ea30fc4244c49038919a008bfb6216e3aabf4e1faa9ed153e2b8cd2e5a6
LN : packet.f2 (A) = 17a46e814bf5ca3ee5c30f826a78c74a66108371fe13a4b932af38864aa
907498e10cbcc01db65e0e223eacb5acdbb19c9e10f54d661ebd9dd884e8676f81a33
LN : packet.t1 (t) = 1364318776
*********************************************************************
LN : Replay Acc_login from LN to user is possible after first login,Please type
:

        N : No Replay
        1 : Replay without modification
        2 : Replay with modification
*********************************************************************
2
LN : T2 1364318649
ED_LN : Packet sent to user, ID = 21
ED_LN : Packet.f1 = Acc_login
ED_LN : Packet.t2 = 100032251
User : Packet received from LN, ID = 21
User : Packet.f1 = Acc_login
User : Packet.f2 (Yu) = 2dfd75dbb83adc79b2638d3be9d8e945a333e2db70a09d07e20c3116
57055283ab77b4bbbd4a201eed2cc60e0ce42cbeed52a2d72f55b683c2c4d84a89ea1a53
User : Packet.t2 (T2) = 100032251
User : Vu_ = fc2713e64e34aa32ef1763678aadb9dfd3cad03d3fe19384ae26b3dedfaece00a7f
bd66ed64a6a95e844541d798442794f4dbec44d9e0d576dc58248c931aab9
User : Yu_ = 27e296c49db36c9b53976257167001ac360e059ca8165f0ecb5fd4f0054fe1eeadb
eac49896f9dbecbba78673a3b5ec87e4ddb5ede08336495305379d9483a54
User : Login failed (Yu not equal Yu_)
```

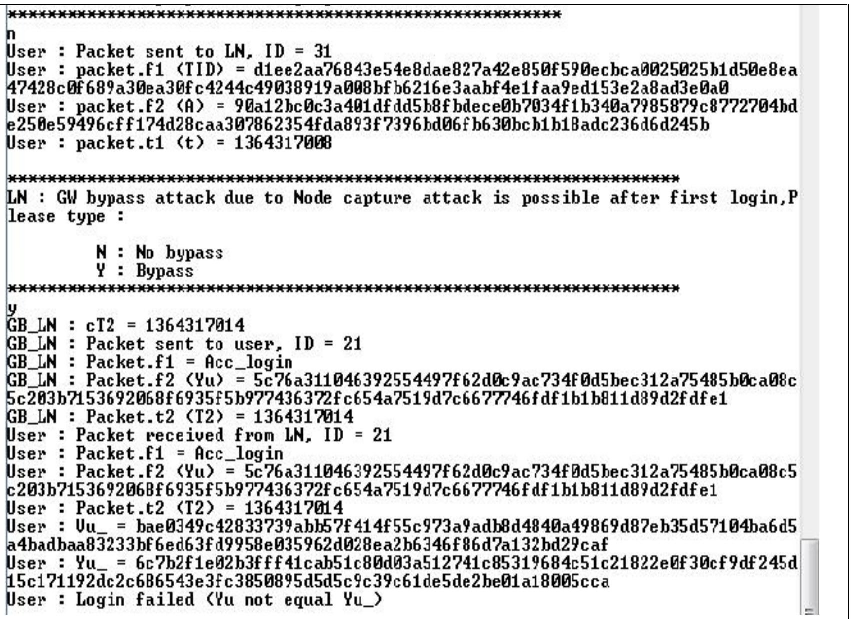
The proposed scheme can resist a replay attack of accept login message in authentication phase A4.



```
428c0f689a30ea30fc4244c49038919a008bfb6216e3aabf4e1faa9ed153e2a8ad0e1a1
LN : Packet.f2 (Ck) = d084e9c69bcaebec20ee89b02a31aa387070a26b9680c3ba89aca12c92
1028582a65d69814eb02c5a9dc323ba218dde41226aa12ddc8ff6679e4110cd7055b63
LN : Packet.t1 (T0) = 1364319265
LN : Packet.t2 (t) = 1364319262
GW : TS = 1364319135
GW : t1 = 1364319262
**********************************************************************
GW : Replay Acc_login from GW to LN is possible after first login

GW : Please type :
         No Replay : N
         Replay without changing T1 : 1
         Replay with changing T1 : 2
**********************************************************************
1
ED_GW : Packet.f1 = Acc_login
ED_GW : Packet.f2(Vs) = 6f9d6aaaba46b44e8e0c83b597844c3285dab2af0ff8e2c65b7229de
07953b637f8dc5b016fefaa9cddc1fa0ef4710705c2e4df360ac5e2d230b85ecb0ad2882
ED_GW : Packet.t1(T1) = 1364319257
LN : Packet received from GW, ID = 32
LN : Packet.f1 = Acc_login
LN : Packet.f2(Vs) = 6f9d6aaaba46b44e8e0c83b597844c3285dab2af0ff8e2c65b7229de079
53b637f8dc5b016fefaa9cddc1fa0ef4710705c2e4df360ac5e2d230b85ecb0ad2882
LN : Packet.f3(Vu) = 5922cdcf2af304bcc482228f738d4adfae5706af59cd21110125490dd2a
dfc52f145d5b31d0efebf7ce7537109b84e46a92dce1ea8d8d4710a9e77a6cd3dd0e9
LN : Packet.t1(T1) = 1364319257
LN : X = 94d8b9f300aee3bf0cdbe2d773f25d8b87f43c6c9ec3d3afad30fb01ecdd7c42e2c4248
5678bfcc2524390d4ca7faaab780f85c3a70f71b81ff1a2b007a98fd2
LN : A = 6eed25692d0f4c275a7807f04ec1c709a3d0dc00360d825b7a1254656944764866 4ac95
be9e159923eef3cd8b840fba17d181211d05ec6b46e98b199ece988f8
LN : T0_cstr = 1364319265
LN : T1_cstr = 1364319257
LN : v = 94d8b9f300aee3bf0cdbe2d773f25d8b87f43c6c9ec3d3afad30fb01ecdd7c42e2c4248
5678bfcc2524390d4ca7faaab780f85c3a70f71b81ff1a2b007a98fd26eed25692d0f4c275a7807f
04ec1c709a3d0dc00360d825b7a125465694476486 64ac95be9e159923eef3cd8b840fba17d18121
1d05ec6b46e98b199ece988f813643192651364319257
LN : Vs_ = 429f424c546d7cc9ceccd407392d27ec1e123e52589928f5ac81aa6db1524cd817b46
f0ecbcfe2de5da3175d3e724f748f9b692d9623fd0582953858dc73645d
LN : Login rejected at LN (Vs not equals Vs_)
User : Authentication failed at Login node(Vs not equals Vs')
User : Please type,
         L : Login
         C : Change Password
         E : Exit Simulation
```



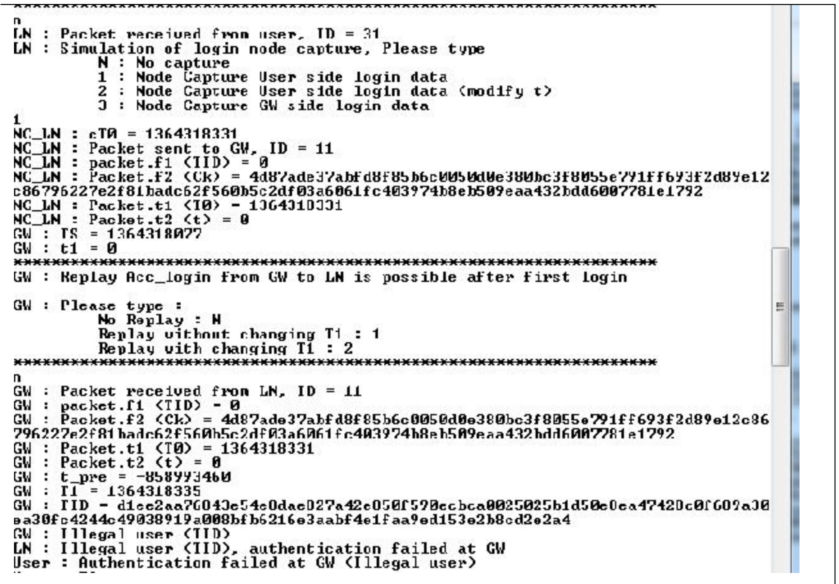
```
LN : T0 = 1364319390
LN : Packet sent to GW, ID = 11
LN : packet.f1 (TID) = d1ee2aa76843e54e8dae827a42e850f590ecbca0025025b1d50e8ea47
428c0f689a30ea30fc4244c49038919a008bfb6216e3aabf4e1faa9ed153e2a8ad0e1a1
LN : Packet.f2 (Ck) = 80268588e415dc86b10bb0ad19c39a8fd6ade00582dec016b913fc6663
016e35ef75edad6fd3751f42e5786c58300d3a961f429e3beee82d0bd350573bfbcd1a
LN : Packet.t1 (T0) = 1364319390
LN : Packet.t2 (t) = 1364319384
GW : TS = 1364319135
GW : t1 = 1364319384
**********************************************************************
GW : Replay Acc_login from GW to LN is possible after first login

GW : Please type :
         No Replay : N
         Replay without changing T1 : 1
         Replay with changing T1 : 2
**********************************************************************
2
ED_GW : Packet.f1 = Acc_login
ED_GW : Packet.f2(Vs) = 6f9d6aaaba46b44e8e0c83b597844c3285dab2af0ff8e2c65b7229de
07953b637f8dc5b016fefaa9cddc1fa0ef4710705c2e4df360ac5e2d230b85ecb0ad2882
ED_GW : Packet.t1(T1) = 1364319394
LN : Packet received from GW, ID = 32
LN : Packet.f1 = Acc_login
LN : Packet.f2(Vs) = 6f9d6aaaba46b44e8e0c83b597844c3285dab2af0ff8e2c65b7229de079
53b637f8dc5b016fefaa9cddc1fa0ef4710705c2e4df360ac5e2d230b85ecb0ad2882
LN : Packet.f3(Vu) = 5922cdcf2af304bcc482228f738d4adfae5706af59cd21110125490dd2a
dfc52f145d5b31d0efebf7ce7537109b84e46a92dce1ea8d8d4710a9e77a6cd3dd0e9
LN : Packet.t1(T1) = 1364319394
LN : X = 94d8b9f300aee3bf0cdbe2d773f25d8b87f43c6c9ec3d3afad30fb01ecdd7c42e2c4248
5678bfcc2524390d4ca7faaab780f85c3a70f71b81ff1a2b007a98fd2
LN : A = 4b86037ac84ce0d1a40984cdc17791fd02dbb1e602c4cce5f26ea7568400af675c75d12
2faa572fb5b8da63638ddaccbd9f92ff18e62a10dfa4083a019c64cdd
LN : T0_cstr = 1364319390
LN : T1_cstr = 1364319394
LN : v = 94d8b9f300aee3bf0cdbe2d773f25d8b87f43c6c9ec3d3afad30fb01ecdd7c42e2c4248
5678bfcc2524390d4ca7faaab780f85c3a70f71b81ff1a2b007a98fd24b86037ac84ce0d1a40984c
dc17791fd02dbb1e602c4cce5f26ea7568400af675c75d122faa572fb5b8da63638ddaccbd9f92ff
18e62a10dfa4083a019c64cdd13643193901364319394
LN : Vs_ = d3c06000b900cce3391e5aa5c4abd6e96e0d15799438994ae1452e99fdbb0534c9b6f
3020405084b8448d7958ef616aafaa686647d2968c68dc42d98d82406c9
LN : Login rejected at LN (Vs not equals Vs_)
User : Authentication failed at Login node(Vs not equals Vs')
```

The proposed scheme can resist a Replay attack of messages in login phase L2 and authentication phase A2.	

```
************************************************************
n
User : Packet sent to LN, ID = 31
User : packet.f1 (TID) = d1ee2aa76843e54e8dae827a42e850f590ecbca0025025b1d50e8ea
47428c0f689a30ea30fc4244c49038919a008bfb6216e3aabf4e1faa9ed153e2a8ad3e0a0
User : packet.f2 (A) = 90a12bc0c3a401dfdd5b8fbdece0b7034f1b340a7985879c8772704bd
e250e59496cff174d28caa307862354fda893f7396bd06fb630bcb1b18adc236d6d245b
User : packet.t1 (t) = 1364317008

**********************************************************************
LN : GW bypass attack due to Node capture attack is possible after first login,P
lease type :

         N : No bypass
         Y : Bypass
**********************************************************************
y
GB_LN : cT2 = 1364317014
GB_LN : Packet sent to user, ID = 21
GB_LN : Packet.f1 = Acc_login
GB_LN : Packet.f2 (Yu) = 5c76a311046392554497f62d0c9ac734f0d5bec312a75485b0ca08c
5c203b7153692068f6935f5b977436372fc654a7519d7c6677746fdf1b1b811d89d2fdfe1
GB_LN : Packet.t2 (T2) = 1364317014
User : Packet received from LN, ID = 21
User : Packet.f1 = Acc_login
User : Packet.f2 (Yu) = 5c76a311046392554497f62d0c9ac734f0d5bec312a75485b0ca08c5
c203b7153692068f6935f5b977436372fc654a7519d7c6677746fdf1b1b811d89d2fdfe1
User : Packet.t2 (T2) = 1364317014
User : Vu_ = bae0349c42833739abb57f414f55c973a9adb8d4840a49869d87eb35d57104ba6d5
a4badbaa83233bf6ed63fd9958e035962d028ea2b6346f86d7a132bd29caf
User : Yu_ = 6c7b2f1e02b3fff41cab51c80d03a512741c85319684c51c21822e0f30cf9df245d
15c171192dc2c686543e3fc3850895d5d5c9c39c61de5de2be01a18005cca
User : Login failed (Yu not equal Yu_)
```

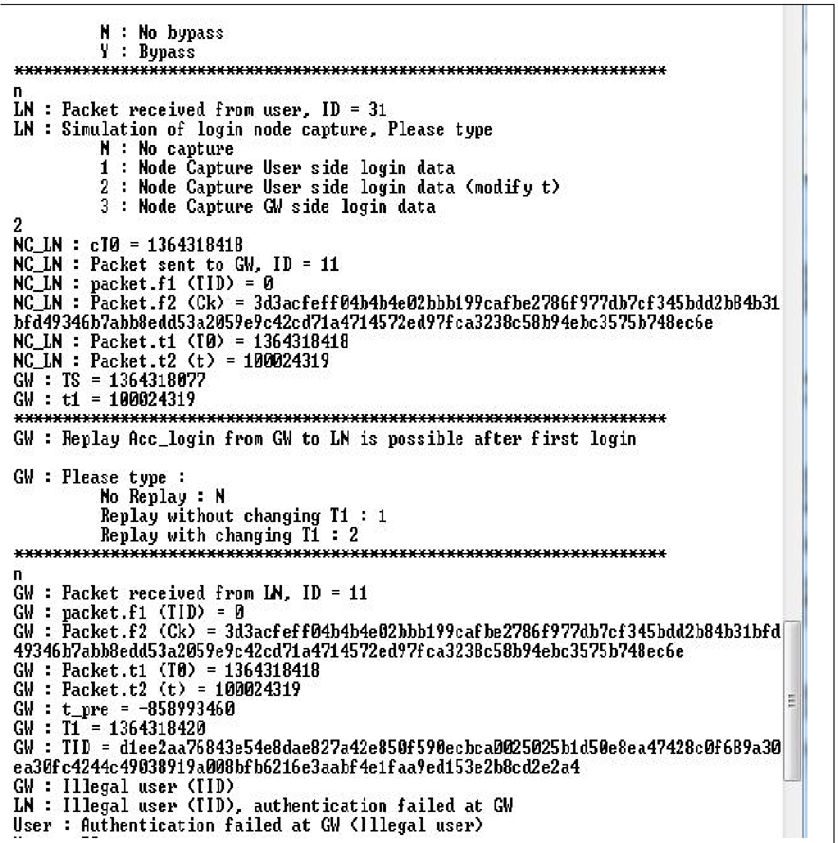
Table 5.2b shows the simulation result of forgery attack with node capture attack. In forgery attack, either the captured message is modified or a new message is computed with the help of captured node at different communication levels. The simulation confirms that the proposed authentication scheme resists forgery attack with node capture attack.

Table 5.2b: Simulation of Node Capture Attack

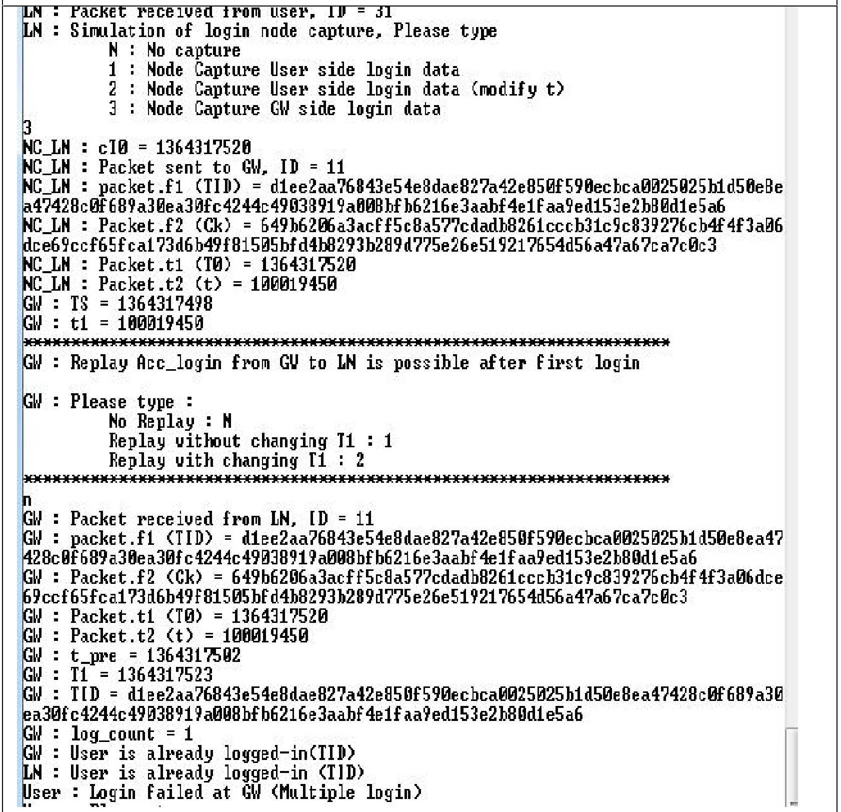
The proposed scheme can resist forgery attack with node capture attack.	

```
**********************************************************************
n
LN : Packet received from user, ID = 31
LN : Simulation of login node capture, Please type
         N : No capture
         1 : Node Capture User side login data
         2 : Node Capture User side login data (modify t)
         3 : Node Capture GW side login data
1
NC_LN : cT0 = 1364318331
NC_LN : Packet sent to GW, ID = 11
NC_LN : packet.f1 (TID) = 0
NC_LN : Packet.f2 (Ck) = 4d87ade37abfd8f85b6c0050d0e380bc3f8055e791ff693f2d89e12
c86796227e2f81badc62f560b5c2df03a6061fc403974b8eb509eaa432bdd6007781e1792
NC_LN : Packet.t1 (T0) = 1364318331
NC_LN : Packet.t2 (t) = 0
GW : TS = 1364318077
GW : t1 = 0
**********************************************************************
GW : Replay Acc_login from GW to LN is possible after first login

GW : Please type :
         No Replay : N
         Replay without changing T1 : 1
         Replay with changing T1 : 2
**********************************************************************
n
GW : Packet received from LN, ID = 11
GW : packet.f1 (TID) = 0
GW : Packet.f2 (Ck) = 4d87ade37abfd8f85b6c0050d0e380bc3f8055e791ff693f2d89e12c86
796227e2f81badc62f560b5c2df03a6061fc403974b8eb509eaa432bdd6007781e1792
GW : Packet.t1 (T0) = 1364318331
GW : Packet.t2 (t) = 0
GW : t_pre = -858993460
GW : T1 = 1364318335
GW : TID = d1ee2aa76843e54e8dae827a42e850f590ecbca0025025b1d50e8ea47428c0f689a30
ea30fc4244c49038919a008bfb6216e3aabf4e1faa9ed153e2b8cd2e2a4
GW : Illegal user (TID)
LN : Illegal user (TID), authentication failed at GW
User : Authentication failed at GW (Illegal user)
```

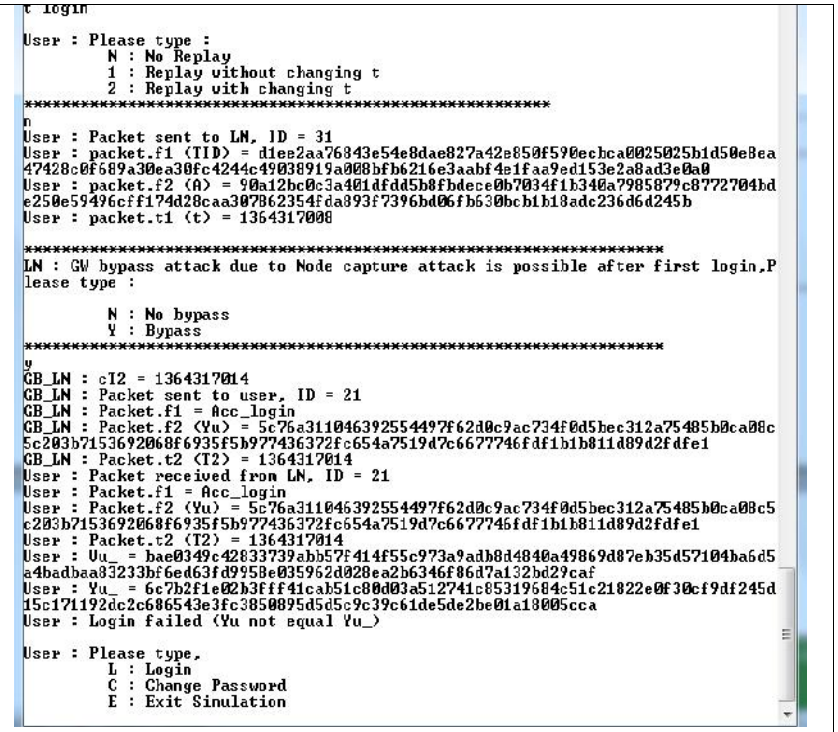
The proposed scheme can resist forgery attack with node capture attack.



```
        N : No bypass
        Y : Bypass
********************************************************************
n
LN : Packet received from user, ID = 31
LN : Simulation of login node capture, Please type
        N : No capture
        1 : Node Capture User side login data
        2 : Node Capture User side login data (modify t)
        3 : Node Capture GW side login data
2
NC_LN : cT0 = 1364318418
NC_LN : Packet sent to GW, ID = 11
NC_LN : packet.f1 (TID) = 0
NC_LN : Packet.f2 (Ck) = 3d3acfeff04b4b4e02bbb199cafbe2786f977db7cf345bdd2b84b31
bfd49346b7abb8edd53a2059e9c42cd71a4714572ed97fca3238c58b94ebc3575b748ec6e
NC_LN : Packet.t1 (T0) = 1364318418
NC_LN : Packet.t2 (t) = 100024319
GW : TS = 1364318077
GW : t1 = 100024319
********************************************************************
GW : Replay Acc_login from GW to LN is possible after first login

GW : Please type :
        No Replay : N
        Replay without changing T1 : 1
        Replay with changing T1 : 2
********************************************************************
n
GW : Packet received from LN, ID = 11
GW : packet.f1 (TID) = 0
GW : Packet.f2 (Ck) = 3d3acfeff04b4b4e02bbb199cafbe2786f977db7cf345bdd2b84b31bfd
49346b7abb8edd53a2059e9c42cd71a4714572ed97fca3238c58b94ebc3575b748ec6e
GW : Packet.t1 (T0) = 1364318418
GW : Packet.t2 (t) = 100024319
GW : t_pre = -858993460
GW : T1 = 1364318420
GW : TID = d1ee2aa76843e54e8dae827a42e850f590ecbca0025025b1d50e8ea47428c0f689a30
ea30fc4244c49038919a008bfb6216e3aabf4e1faa9ed153e2b8cd2e2a4
GW : Illegal user (TID)
LN : Illegal user (TID), authentication failed at GW
User : Authentication failed at GW (Illegal user)
```



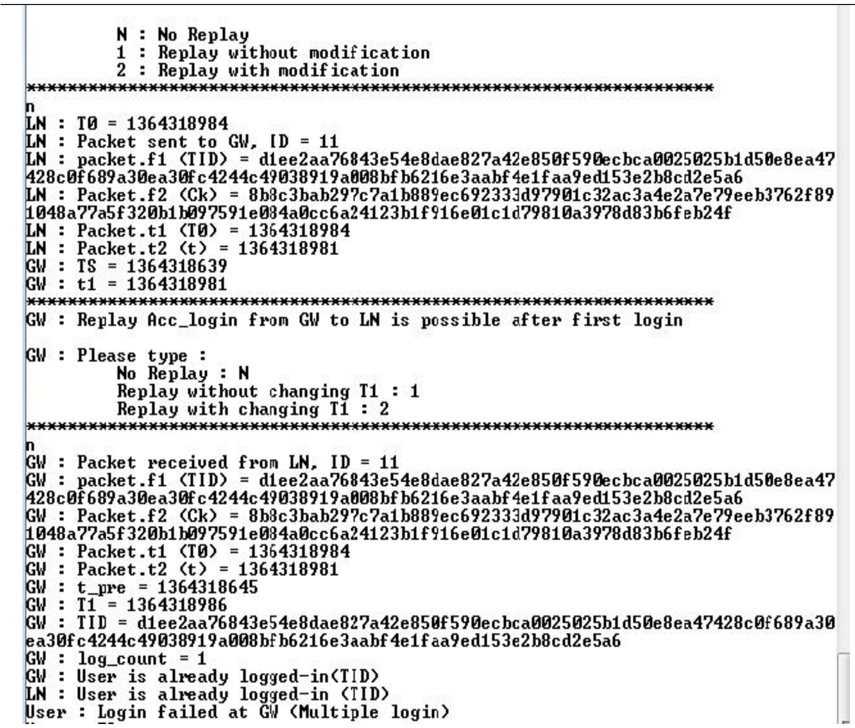
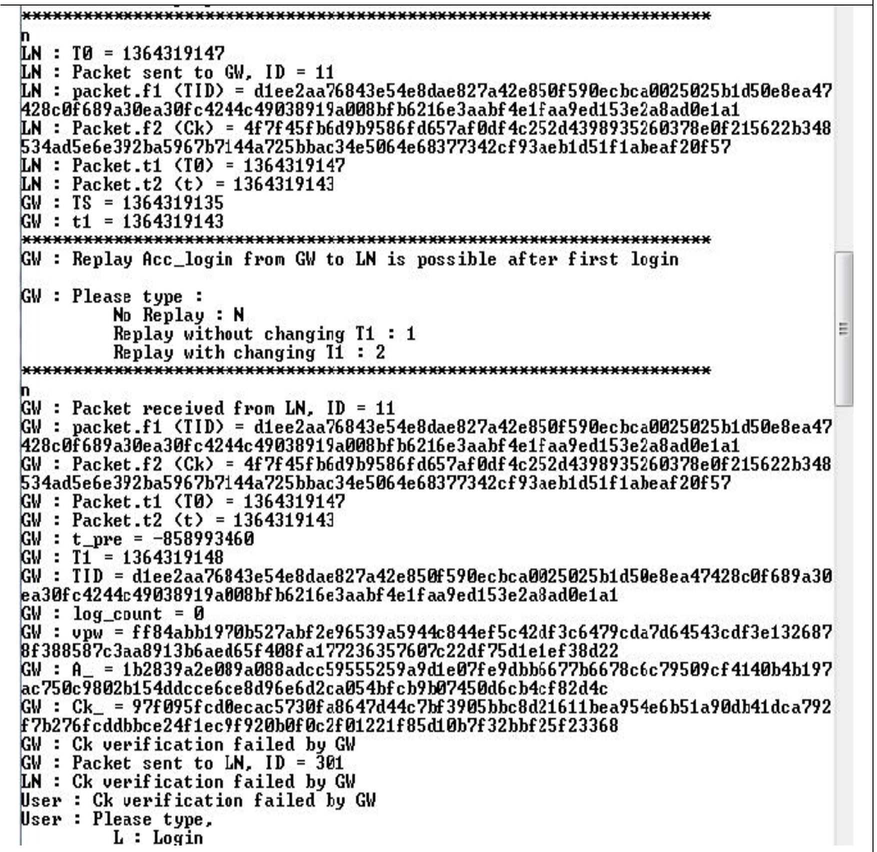
```
LN : Packet received from user, ID = 31
LN : Simulation of login node capture, Please type
        N : No capture
        1 : Node Capture User side login data
        2 : Node Capture User side login data (modify t)
        3 : Node Capture GW side login data
3
NC_LN : cT0 = 1364317520
NC_LN : Packet sent to GW, ID = 11
NC_LN : packet.f1 (TID) = d1ee2aa76843e54e8dae827a42e850f590ecbca0025025b1d50e8e
a47428c0f689a30ea30fc4244c49038919a008bfb6216e3aabf4e1faa9ed153e2b80d1e5a6
NC_LN : Packet.f2 (Ck) = 649b6206a3acff5c8a577cdadb8261cccb31c9c839276cb4f4f3a06
dce69ccf65fca173d6b49f81505bfd4b8293b289d775e26e519217654d56a47a67ca7c0c3
NC_LN : Packet.t1 (T0) = 1364317520
NC_LN : Packet.t2 (t) = 100019450
GW : TS = 1364317498
GW : t1 = 100019450
********************************************************************
GW : Replay Acc_login from GV to LN is possible after first login

GW : Please type :
        No Replay : N
        Replay vithout changing T1 : 1
        Replay vith changing T1 : 2
********************************************************************
n
GW : Packet received from LN, ID = 11
GW : packet.f1 (TID) = d1ee2aa76843e54e8dae827a42e850f590ecbca0025025b1d50e8ea47
428c0f689a30ea30fc4244c49038919a008bfb6216e3aabf4e1faa9ed153e2b80d1e5a6
GW : Packet.f2 (Ck) = 649b6206a3acff5c8a577cdadb8261cccb31c9c839276cb4f4f3a06dce
69ccf65fca173d6b49f81505bfd4b8293b289d775e26e519217654d56a47a67ca7c0c3
GW : Packet.t1 (T0) = 1364317520
GW : Packet.t2 (t) = 100019450
GW : t_pre = 1364317502
GW : T1 = 1364317523
GW : TID = d1ee2aa76843e54e8dae827a42e850f590ecbca0025025b1d50e8ea47428c0f689a30
ea30fc4244c49038919a008bfb6216e3aabf4e1faa9ed153e2b80d1e5a6
GW : log_count = 1
GW : User is already logged-in(TID)
LN : User is already logged-in (TID)
User : Login failed at GW (Multiple login)
```

The scheme can resist gateway bypass attack due to node-capture attack.	

```
t login

User : Please type :
         N : No Replay
         1 : Replay vithout changing t
         2 : Replay vith changing t
*********************************************************
n
User : Packet sent to LN, ID = 31
User : packet.f1 (TID) = d1ee2aa76843e54e8dae827a42e850f590ecbca0025025b1d50e8ea
47428c0f689a30ea30fc4244c49038919a008bfb6216e3aabf4e1faa9ed153e2a8ad3e0a0
User : packet.f2 (A) = 90a12bc0c3a401dfdd5b8fbdece0b7034f1b340a7985879c8772704bd
e250e59496cff174d28caa307862354fda893f7396bd06fb630bcb1b18adc236d6d245b
User : packet.t1 (t) = 1364317008

**********************************************************************
LN : GW bypass attack due to Node capture attack is possible after first login,P
lease type :

         N : No bypass
         Y : Bypass
**********************************************************************
y
GB_LN : cT2 = 1364317014
GB_LN : Packet sent to user, ID = 21
GB_LN : Packet.f1 = Acc_login
GB_LN : Packet.f2 (Yu) = 5c76a311046392554497f62d0c9ac734f0d5bec312a75485b0ca08c
5c203b7153692068f6935f5b977436372fc654a7519d7c6677746fdf1b1b811d89d2fdfe1
GB_LN : Packet.t2 (T2) = 1364317014
User : Packet received from LN, ID = 21
User : Packet.f1 = Acc_login
User : Packet.f2 (Yu) = 5c76a311046392554497f62d0c9ac734f0d5bec312a75485b0ca08c5
c203b7153692068f6935f5b977436372fc654a7519d7c6677746fdf1b1b811d89d2fdfe1
User : Packet.t2 (T2) = 1364317014
User : Vu_ = bae0349c42833739abb57f414f55c973a9adb8d4840a49869d87eb35d57104ba6d5
a4badbaa83233bf6ed63fd9958e035962d028ea2b6346f86d7a132bd29caf
User : Yu_ = 6c7b2f1e02b3fff41cab51c80d03a512741c85319684c51c21822e0f30cf9df245d
15c171192dc2c686543e3fc3850895d5d5c9c39c61de5de2be01a18005cca
User : Login failed (Yu not equal Yu_)

User : Please type,
         L : Login
         C : Change Password
         E : Exit Sinulation
```

Table 5.2c shows the simulation result of many logged in user with same login ID threat and password guessing attack. The result verifies our claim as discussed in chapter 4.

Table 5.2c: Simulation of Many Logged in User with Same Login ID Threat

The proposed scheme resists many logged in user with same login ID threat.	
The proposed scheme resists password guessing at login phase L1	

```
        N : No Replay
        1 : Replay without modification
        2 : Replay with modification
**********************************************************************
n
LN : T0 = 1364318984
LN : Packet sent to GW, ID = 11
LN : packet.f1 (TID) = d1ee2aa76843e54e8dae827a42e850f590ecbca0025025b1d50e8ea47
428c0f689a30ea30fc4244c49038919a008bfb6216e3aabf4e1faa9ed153e2b8cd2e5a6
LN : Packet.f2 (Ck) = 8b8c3bab297c7a1b889ec692333d97901c32ac3a4e2a7e79eeb3762f89
1048a77a5f320b1b097591e084a0cc6a24123b1f916e01c1d79810a3978d83b6feb24f
LN : Packet.t1 (T0) = 1364318984
LN : Packet.t2 (t) = 1364318981
GW : TS = 1364318639
GW : t1 = 1364318981
**********************************************************************
GW : Replay Acc_login from GW to LN is possible after first login

GW : Please type :
        No Replay : N
        Replay without changing T1 : 1
        Replay with changing T1 : 2
**********************************************************************
n
GW : Packet received from LN, ID = 11
GW : packet.f1 (TID) = d1ee2aa76843e54e8dae827a42e850f590ecbca0025025b1d50e8ea47
428c0f689a30ea30fc4244c49038919a008bfb6216e3aabf4e1faa9ed153e2b8cd2e5a6
GW : Packet.f2 (Ck) = 8b8c3bab297c7a1b889ec692333d97901c32ac3a4e2a7e79eeb3762f89
1048a77a5f320b1b097591e084a0cc6a24123b1f916e01c1d79810a3978d83b6feb24f
GW : Packet.t1 (T0) = 1364318984
GW : Packet.t2 (t) = 1364318981
GW : t_pre = 1364318645
GW : T1 = 1364318986
GW : TID = d1ee2aa76843e54e8dae827a42e850f590ecbca0025025b1d50e8ea47428c0f689a30
ea30fc4244c49038919a008bfb6216e3aabf4e1faa9ed153e2b8cd2e5a6
GW : log_count = 1
GW : User is already logged-in(TID)
LN : User is already logged-in (TID)
User : Login failed at GW (Multiple login)
```

```
**********************************************************************
n
LN : T0 = 1364319147
LN : Packet sent to GW, ID = 11
LN : packet.f1 (TID) = d1ee2aa76843e54e8dae827a42e850f590ecbca0025025b1d50e8ea47
428c0f689a30ea30fc4244c49038919a008bfb6216e3aabf4e1faa9ed153e2a8ad0e1a1
LN : Packet.f2 (Ck) = 4f7f45fb6d9b9586fd657af0df4c252d4398935260378e0f215622b348
534ad5e6e392ba5967b7144a725bbac34e5064e68377342cf93aeb1d51f1abeaf20f57
LN : Packet.t1 (T0) = 1364319147
LN : Packet.t2 (t) = 1364319143
GW : TS = 1364319135
GW : t1 = 1364319143
**********************************************************************
GW : Replay Acc_login from GW to LN is possible after first login

GW : Please type :
        No Replay : N
        Replay without changing T1 : 1
        Replay with changing T1 : 2
**********************************************************************
n
GW : Packet received from LN, ID = 11
GW : packet.f1 (TID) = d1ee2aa76843e54e8dae827a42e850f590ecbca0025025b1d50e8ea47
428c0f689a30ea30fc4244c49038919a008bfb6216e3aabf4e1faa9ed153e2a8ad0e1a1
GW : Packet.f2 (Ck) = 4f7f45fb6d9b9586fd657af0df4c252d4398935260378e0f215622b348
534ad5e6e392ba5967b7144a725bbac34e5064e68377342cf93aeb1d51f1abeaf20f57
GW : Packet.t1 (T0) = 1364319147
GW : Packet.t2 (t) = 1364319143
GW : t_pre = -858993460
GW : T1 = 1364319148
GW : TID = d1ee2aa76843e54e8dae827a42e850f590ecbca0025025b1d50e8ea47428c0f689a30
ea30fc4244c49038919a008bfb6216e3aabf4e1faa9ed153e2a8ad0e1a1
GW : log_count = 0
GW : vpw = ff84abb1970b527abf2e96539a5944c844ef5c42df3c6479cda7d64543cdf3e132687
8f388587c3aa8913b6aed65f408fa177236357607c22df75d1e1ef38d22
GW : A_ = 1b2839a2e089a088adcc59555259a9d1e07fe9dbb6677b6678c6c79509cf4140b4b197
ac750c9802b154ddcce6ce8d96e6d2ca054bfcb9b07450d6cb4cf82d4c
GW : Ck_ = 97f095fcd0ecac5730fa8647d44c7bf3905bbc8d21611bea954e6b51a90db41dca792
f7b276fcddbbce24f1ec9f920b0f0c2f01221f85d10b7f32bbf25f23368
GW : Ck verification failed by GW
GW : Packet sent to LN, ID = 301
LN : Ck verification failed by GW
User : Ck verification failed by GW
User : Please type,
        L : Login
```

The main focus of our model is to provide application layer security. However, the security can further be enhanced by incorporating IEEE 802.15.4 specification into our scheme in order to provide confidentiality on frame at MAC sub-layer for all the four phases of our protocol.

5.5.2 Memory Cost Evaluation

The storage cost can be estimated by the memory usage. One way of estimating the memory usage is to count up the number of variables and weigh them by the number of bytes according to their types. A comparison is made between our proposed scheme with the existing schemes [24, 46, 48] based on memory requirement to store the user data that is presented in Table 5.3. The storage overhead of sensor node for our proposed scheme is slightly higher than that of the existing protocols [24, 46, 48]. However, our authentication protocol provides enhanced security by eliminating the flaws of previous schemes such as replay attack, gateway bypass attack and many logged in user threat identified in the previous protocols.

Table 5.3 Comparison of Memory Requirement

Storage overhead per user (bits)			
Scheme	*UD*	*GW*	*SN*
Tseng *et al.'s* scheme	640	704	160
Ko's scheme	1152	1216	672
Vaidya *et al.'s* scheme	2304	1600	1056
Our Proposed scheme	2304	1601	1088

5.6 Conclusion

In this chapter we have discussed the modeling of our proposed scheme in SystemC. The performance of the authentication scheme is evaluated from security, latency and cost perspective. A detailed comparative evaluation is provided to establish the supremacy of the proposed scheme. In chapter 6 we conclude the thesis and provide some directions for future research.

Chapter 6: Conclusion and Future Work

WSNs have attracted an increasing number of researchers due to its ubiquitous nature, easy deployment and a wide range of applications. In many applications, integrity and confidentiality of collected data as well as the user privacy are very critical. Security measures should be incorporated to protect the access to critical data and to restrict non-authorized users from gaining the data access. However, the provision of perfect security in WSNs is a challenging task due to various network and resource constraints and malicious attacks. In this thesis, we have proposed a robust user authentication method that ensures that only the legitimate user can access the sensor data.

In order to get some basic understanding of WSN, we have provided a brief introduction WSN, its hard-ware platform, network structure, standards and specification, network topology and some interesting and promising applications. Then we have discussed the security challenges and different types of attacks in a WSN. A detailed security analysis of some of the past representative authentication schemes is presented in order to review their benefits and avoid their limitations in our proposed scheme. In this thesis, we have proposed a robust user authentication scheme which is an improved password-based authentication methodology. The scheme is built upon the past authentication techniques put forward by Tseng *et al.*, Ko and Vaidya *et al.* [24, 46, 48]. During the design of our security protocol we considered several security requirements such as data confidentiality, integrity, freshness, authentication and the scalability of sensor networks. We have identified that most of the past schemes are subject to several security flaws [24, 46, 48]. To overcome these flaws, we have proposed an improved authentication scheme that not only retains all the advantages of past schemes [24, 46, 48] but also enhances the security by eliminating their weaknesses. Our proposed scheme possesses many advantages including resistance to replay attack, GW node bypass and many logged in user attacks. It also provides mutual authentication among all the entities of WSN. The comparison between the past schemes and our proposed scheme indicates that our scheme can provide better security with no additional computation and almost negligible addition to the memory storage overhead at the level of sensor nodes. Our scheme resists the replay attack inherently and does not need strict network synchronization. In fact, the timestamp used in our scheme works as a nonce. Our scheme successfully resists many logged in users with the same ID attack.

We have modeled our proposed authentication scheme and the past representative schemes [24, 46, 48] using SystemC. The modeling and simulation of the past schemes provides sufficient evidence that the schemes are vulnerable to a number of security threats. We have verified that our scheme eliminates the known flaws of the past schemes and provides better security than all the previous password based authentication methods.

In our scheme, we have assumed that the database is securely stored in GW node and failing to meet this condition may make our scheme vulnerable to stolen verifier attack. Again none of the past schemes provide an inherent method to detect a compromised node. These issues open some future directions to our work to mitigate the stolen verifier and node compromise attacks.

As a future work, one can investigate for the following:

- Design a protocol that inherently detects a compromised node.
- Optimize the code and message size to appropriately optimize the memory usage and latency time.
- Implement a WSN on a real platform and integrate the user authentication at the application layer.

Bibliography

[1] J. Agre and L. Clare, “An Integrated Architecture for Cooperative Sensing Networks,” *IEEE Computer,* Vol. 33, No. 5, 2000, pp. 106 – 108.

[2] I. F. Akyildiz, T. Melodia and K. Chowdhury, “A Survey on Wireless Multimedia Sensor Networks”, *Computer Networks,* Vol. 51, No. 4, March 2007, pp. 921-960.

[3] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam and E. Cayirci, “A survey on sensor networks”, *IEEE Communications Magazine*, Vol. 40, No. 8, 2002, pp. 102–114.

[4] R. J. Anderson and M. G. Kuhn, “Low cost attacks on tamper resistant devices”, *In Proceedings of the 5th International Workshop on Security Protocols,* London, UK, pp. 125–136, 1998.

[5] Z. Benenson, F. Gartner and D. Kesdogan, "User authentication in sensor networks, *"In Proceedings of Informatik Workshop on Sensor Networks*, Ulm, Germany, pp. 1-5, September 2004.

[6] Z. Benenson, N. Gedicke and O. Raivio, "Realizing robust user authentication in sensor networks", *In Proceedings of Workshop on Real-World Wireless Sensor Networks*, Stockholm, Sweden, pp. 1-5, June 2005.

[7] W. Chen and L. Sha, “An Energy-Aware Data-Centric Generic Utility Based Approach in Wireless Sensor Networks”, *In proceedings of the 3rd International Symposium on Information Processing in Sensor Networks*, Berkeley, California, USA, pp. 215 – 224, April, 26 – 27, 2004.

[8] M. L. Das, "Two-Factor User Authentication in Wireless Sensor Networks", *IEEE Transactions on Wireless Communications,* Vol. 8, No. 3, pp. 1086-1090, 2009.

[9] J. Deng, R. Han and S. Mishra, “Defending against Path-based DoS Attacks in Wireless Sensor Networks”, *In Proceedings of the 3rd ACM Workshop on Security of Ad Hoc and Sensor Networks*, Virginia, USA, pp. 89-96, 2005.

[10] D. Dolev and A.C. Yao, “The security of public key protocols”, *IEEE Transactions on Information Theory*, Vol. 29, No. 2, pp. 198-208, 1983.

[11] T. 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.

[12] P. Hainga, G. Smit, and M. Bos, “Energy-Efficient Adaptive Wireless Network Design”, *In proceedings of the 5th IEEE Symposium on Computers and Communications*, Antibes, France, pp. 502 – 507, 2000.

[13] K. Han I, K. Kim I and T. Shon, “Untraceable Mobile Node Authentication in WSN”, *Sensors*, Vol. 10, pp. 4410-4429, 2010.

[14] Z. M. Hanapi, M. Ismail, K. Jumari, and H. Mirvaziri, “Analysis of routing attacks in wireless sensor network” *In Proceedings of the 28th International Cryptology Workshop and Conference*, Santa Barbara, CA, USA, pp. 202-214, 2008.

[15] W. R. Heinzelman, A. Chandrakasan and H. Balakrishnan, "Energy-efficient communication protocol for wireless microsensor networks", *In Proceedings of the 33rd Annual Hawaii International Conference on Systems Sciences*, Maui, Hawaii, Vol.2, No. 10, pp. 4-7, January 2000.

[16] W. Heinzelman, A. Chandrakasan, and H. Balakrishnan, “An Application-Specific Protocol Architecture for Wireless Microsensor Networks”, *IEEE Transactions on Wireless Communications*, Vol. 1, No. 4, 2002, pp. 660 – 670.

[17] J. Hellerstein, W. Hong and S. Madden, “The Sensor Spectrum: Technology, Trends, and Requirements,” *SIGMOD Record*, Vol. 32, No. 4, 2003, pp. 22 – 27.

[18] I. Howitt, W. W. Manges, P. T. Kuruganti, G. Allgood, J. A. Gutierrez and J. M. Conrad, “Wireless industrial sensor networks: Framework for qos assessment and qos management” , *ISA Transactions*, Vol. 45, No. 3, pp. 347–359, 2006.

[19] R. Hunt, “Network Security: The Principles of Threats, Attacks and Intrusions, part1 and part 2”, *APRICOT*, 2004.

[20] A. Iwata, C. Chiang, G. Pei, M. Gerla and T. Chen, “Scalable Routing Strategies for Ad Hoc Wireless Networks”, *IEEE Journal on Selected areas in Communications*, Vol. 17, No. 8, 1999, pp. 1369 – 1379.

[21] J. M. Kahn, R. H. Katz and K. S. J. Pister, "Emerging Challenges: Mobile Networking for Smart Dust", *Journal of Communications and Networks*, September 2000, Vol. 2, No. 3, pp. 188-196.

[22] C. Karlof and D. Wagner, “Secure routing in wireless sensor networks: attacks and countermeasures”, *Ad Hoc Networks Journal: Special Issue on Sensor Network Applications and Protocols*, Vol.1, pp. 293-315, 2003.

[23] M. K. Khan and K. Alghathbar, "Cryptanalysis and Security Improvements of Two-Factor User Authentication in Wireless Sensor Networks", *Sensors*, Vol. 10, No. 3, pp. 2450-2459, 2010.

[24] L. C. Ko, "A Novel Dynamic User Authentication Scheme for Wireless Sensor Networks", *In Proceedings of IEEE ISWCS*, Reykjavik, Iceland, pp. 608-612, 2008.

[25] S. Kumar and D. Shepherd, “Sensit: Sensor information technology for the war-fighter”, *In Proceedings of the 4th International Conference on Information Fusion*, Montreal, Canada, pp. 3–9, 2001.

[26] T. H. Lee, "Simple Dynamic User Authentication Protocols for Wireless Sensor Networks", *IEEE SENSORCOMM*, Vol. 43, pp. 657-660, 2008.

[27] D. Manivannan, B. Vijayalakshmi and P. Neelamegam, “An efficient authentication protocol based on congruence for Wireless Sensor networks”, *In Proceedings of International Conference on Recent Trends in Information Technology*, Chennai, India, pp. 549-553, 2011.

[28] V.C. Manju, “A Survey on Wireless Sensor Network Attacks”, *International Journal of Engineering and Innovative Technology*, Vol. 2, No. 2, pp. 23-28, August 2012.

[29] D. McPherson, “BGP Security Techniques”, *APRICOT*, 2005.

[30] T. Mizuguchi and T. Yoshida, “BGP Route Hijacking”, *APRICOT*, 2007.

[31] S. Mohammadi and H. Jadidoleslamy, “A Comparison of Physical Attacks on Wireless Sensor Networks”, *International Journal of Peer to Peer Networks*, Vol.2, No.2, pp. 24-42, April 2011.

[32] C. Myers, A. Oppenheim, R. Davis and W. Dove, “Knowledge-based speech analysis and enhancement”, *In Proceedings of the International Conference on Acoustics, Speech and Signal Processing*, San Diego, California, USA, pp. 162-165, 1984.

[33] J. Newsome, E. Shi, D. Song and A. Perrig, “The sybil attack in sensor networks: analysis & defences”, *In Proceedings of Third International Symposium on Information Processing in Sensor Networks*, Berkeley, CA, USA, pp. 259-268, 2004.

[34] D. G. Padmavathi and M. D. Shanmugapriya, “A Survey of Attacks, Security Mechanisms and Challenges in Wireless Sensor Networks”, *International Journal of Computer Science and Information Security*, Vol. 4, No. 1 & 2, pp. 1-9, 2009.

[35] R. D. Pietro, L.V. Mancini and S. Jajodia, “Providing secrecy in key management protocols for large wireless sensors networks”, *Journal of AdHoc Networks*, Vol.1, No. 4, pp. 455–468, 2003.

[36] S. Ping, “Delay Measurement Time Synchronization for Wireless Sensor Networks”, *Intel Research Berkeley Lab*, June 2003.

[37] J. Rabaey, J. Ammer, J. L. da Silva and D. Patel, "PicoRadio: Ad-hoc wireless networking of ubiquitous low-energy sensor/monitor nodes", *In Proceedings of IEEE Computer Society Workshop on VLSI*, Orlando, Florida, USA, pp. 9-12, 2000.

[38] D. R. Raymond and S. F. Midkiff, “Denial of Service in Wireless Networks: Attacks and Defences”, *IEEE Pervasive Computing*, Vol. 7, No. 1, pp. 74-81, 2008.

[39] N. Sastry and D. Wagner, "Security considerations for IEEE 802.15.4 networks", *In Proceedings of 3rd ACM Workshop on Wireless Security,* Philadelphia, USA, pp. 32-42, 2004.

[40] C. Shanty and A. Shoo, "DGRAM: A Delay Guaranteed Routing and MAC Protocol for Wireless Sensor Networks", *IEEE Transactions on Mobile Computing,* Vol. 9, No.10, pp.1407-l423, October 2010.

[41] K. Sharma and M. K. Ghost, “Wireless Sensor Networks: An Overview on its Security Threats”, *Special Issue on Mobile Ad-hoc Networks MANETs,* CSE Department, SMIT, Sikkim, India, 2010.

[42] E. Shi and A. Perrig, “Designing secure sensor networks”, *Wireless Communication Magazine,* Vol. 11, No. 6, pp. 38-43, December 2004.

[43] L. Shi and A.O. Fapojuwo, "TDMA Scheduling with Optimized Energy Efficiency and Minimum Delay in Clustered Wireless Sensor Networks", *IEEE Transactions on Mobile Computing*, Vol. 9, No. 7, pp.927-940, July 2010.

[44] S. P. Skorobogatov, “Semi-invasive attacks - a new approach to hardware security analysis”, *Technical report*, University of Cambridge, Computer Laboratory, April 2005.

[45] S. Sudevalayam and P. Kulkarni, “Energy Harvesting Sensor Nodes: Survey and Implications”, *Technical Report TR-CSE-2008-19,* 2008, Department of Computer Science and Engineering, Indian Institute of Technology Bombay, India.

[46] H. R. Tseng, R. H. Jan and W. Yang, "An improved dynamic user authentication scheme for wireless sensor networks", *In Proceedings of IEEE GLOBECOM*, Washington DC, USA, pp. 986–990, November 2007.

[47] B. Vaidya, D. Makrakis and H. Mouftah, "Improved two-factor user authentication in wireless sensor networks", *In Proceedings of International Workshop on Network Assurance & Security Services in Ubiquitous Environments*, Niagara Falls, ON, Canada, pp. 600-605, December 2010.

[48] B. Vaidya, J. J. Rodrigues and J. H. Park, "User authentication schemes with pseudonymity for ubiquitous sensor network in NGN", *International Journal of Communication Systems,* Vol. 23, pp. 1201-1222, December 2009.

[49] A. Wadaa, S. Olariu, L. Wilson, M. Eltoweissy and K. Jones, “Training a Wireless Sensor Networks”, *Mobile Networks and Applications,* Vol. 10, No. 1-2, pp. 151 – 168, 2005.

[50] H. Wang, "Network Lifetime Optimization in Wireless Sensor Networks", *IEEE Journal on Selected Areas in Communications,* Vol. 28, No. 7, pp.1127-1137, September 2010.

[51] Y. Wang, G. Attebury and B. Ramamurthy, “A Survey of Security Issues in Wireless Sensor Networks”, *IEEE Communications Surveys and Tutorials,* Vol. 8, No. 2, pp. 2-23, 2006.

[52] K. H. Wong, Y. Zheng, J. Cao and S. Wang, "A dynamic user authentication scheme for wireless sensor networks", *In Proceedings of IEEE SUTC*, Taichung, Taiwan, vol. 1, pp. 318–327, June 2006.

[53] R. Watro, D. Kong, S. Cuti, C. Gardiner, C. Lyn and P. Kruus, "TinyPK: securing sensor networks with public key technology", *In Proceedings of ACM Workshop on Security of Ad Hoc & Sensor Networks,* Washington DC, USA, pp. 59-64, October 2004.

[54] E. Witrant, P. Park and M. Johansson, "Time-delay estimation and finite-spectrum assignment for control over multi-hop WSN", *In Wireless Networking Based Control*, S.K. Mazumder (Ed.) (2011) 135-152.

[55] W. Ye, J. Heidemann and D. Estrin, "An Energy-Efficient MAC Protocol for Wireless Sensor Networks", *In Proceedings of the IEEE Infocom*, New York, USA, pp. 1567 – 1576, June, 2002.

[56] W. Xu, K. Ma, W. Trappe and Y. Zhang, "Jamming Sensor Networks: Attack and Defence Strategies", *IEEE Network*, Vol. 20, No. 3, pp. 41-47, 2006.

[57] S. C. Yang, "Flow-based Flooding Detection System", *APRICOT*, 2004.

[58] J. Yick, B. Mukherjee and D. Ghosal, "Wireless Sensor Network Survey", *Computer Networks Journal,* Vol. 52, pp. 2292-2330, 2008.

[59] Q. Zhang, X. Zhou and F. Yang, "Distributed Node Authentication in Wireless Sensor Networks", *In proceedings of 5th International Conference on Wireless Communications, Networking and Mobile Computing*, Beijing, China, pp. 1-4, 2009.

[60] X. Zhou, Y. Xiong, F. Miao and M. Li, "A new dynamic user authentication scheme using smart cards for wireless sensor network", *In Proceedings of IEEE 2nd International Conference on Computing, Control & Industrial Engineering,* Wuhan, China, vol. 2, pp.1-4, Aug 2011.

[61] S. Zhu, S. Setia, and S. Jajodia, “Leap: Efficient security mechanisms for large-scale distributed sensor networks”, *In Proceedings of the10th ACM Conference on Computer and Communications Security*, Washington D.C, USA, pp. 62-72, October 2003.

[62] W. Znaidi, M. Minier and J. P. Babau, “An Ontology for Attacks in Wireless Sensor Networks”, *Institut National De Reccherche En InformatiqueEt En Automatique (INRIA)*, Oct 2008.

[63] Tinyos operating system, http://www.tinyos.net/.

[64] http://www.columbia.edu/acis/rad/authmethods/whatisit.html

[65] "IEEE Std. 802.15.4-2006, IEEE Standard for Local and Metropolitan Area Networks part 15.4, Wireless Medium Access Control (MAC) and Physical Layer (PRY) Specifications for Low-Rate Wireless Personal Area Networks (LR-WPANs)", *IEEE Press,*2006.

[66] C.Y. Chong and P.K. Srikanta," Sensor Networks: Evolution, Opportunities and Challenges", *In Proceedings of the IEEE Infocom*, San Francisco, CA, USA, Vol. 91, No. 8, pp 1247-1256, 2003.

[67] SystemC: The Language for System-Level Modeling, Design and Verification, IEEE Std. 1666-2011. http://www.accellera.org/downloads/standards/systemc

[68] Wireless medium access control and physical layer specifications for low-rate wireless personal area networks, IEEE Standard, 802.15.4-2003, May 2003

[69] "ZigBee Specification Version 1.0", *ZigBee Alliance, 2005.*